5. Process Improvement

5. Process Improvement - Detailed Table of Contents [5.]

5. Process Improvement

5.1. Introduction

This section describes the basic concepts of the Design of Experiments (DOE or DEX)

This section introduces the basic concepts, terminology, goals and procedures underlying the proper statistical design of experiments. Design of experiments is abbreviated as *DOE* throughout this chapter (an alternate abbreviation, *DEX*, is used in DATAPLOT).

Topics covered are:

- What is experimental design or DOE?
- What are the goals or uses of DOE?
- What are the steps in DOE?

5.1.1. What is experimental design?

Experimental Design (or DOE) economically maximizes information

In an experiment, we deliberately change one or more process variables (or factors) in order to observe the effect the changes have on one or more response variables. The (statistical) design of experiments (*DOE*) is an efficient procedure for planning experiments so that the data obtained can be analyzed to yield valid and objective conclusions.

DOE begins with determining the objectives of an experiment and selecting the process factors for the study. An *Experimental Design* is the laying out of a detailed experimental plan in advance of doing the experiment. Well chosen experimental designs maximize the amount of "information" that can be obtained for a given amount of experimental effort.

The statistical theory underlying DOE generally begins with the concept of *process models*.

Process Models for DOE

Black box process model

It is common to begin with a process model of the `black box' type, with several discrete or continuous input factors that can be controlled--that is, varied at will by the experimenter--and one or more measured output responses. The output responses are assumed continuous. Experimental data are used to derive an empirical (approximation) model linking the outputs and inputs. These empirical models generally contain first and second-order terms.

Often the experiment has to account for a number of uncontrolled factors that may be discrete, such as different machines or operators, and/or continuous such as ambient temperature or humidity. Figure 1.1 illustrates this situation.

Schematic for a typical process with controlled inputs, outputs, discrete uncontrolled factors and continuous uncontrolled factors

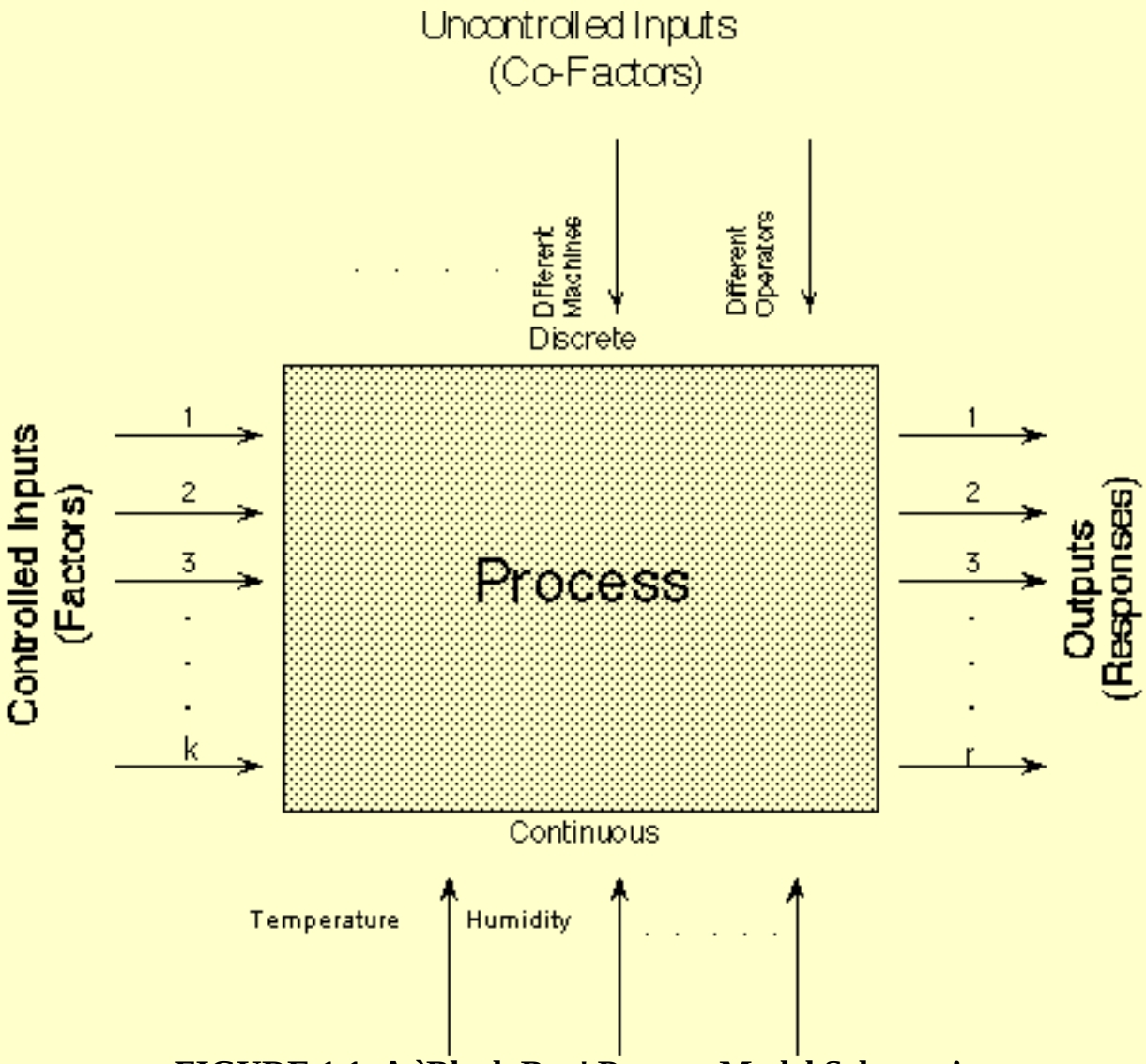


FIGURE 1.1 A `Black Box' Process Model Schematic

Models for DOE's

The most common empirical models fit to the experimental data take either a *linear* form or *quadratic* form.

Linear model

A linear model with two factors, X_1 and X_2, can be written as

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \text{experimental error}$$

Here, Y is the response for given levels of the main effects X_1 and X_2 and the X_1X_2 term is included to account for a possible interaction effect between X_1 and X_2. The constant β_0 is the response of Y when both main effects are 0.

For a more complicated example, a linear model with three factors X_1, X_2, X_3 and one response, Y, would look like (if all possible terms were included in the model)

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{12} X_1 X_2 + \beta_{13} X_1 X_3 + \beta_{23} X_2 X_3 + \beta_{123} X_1 X_2 X_3 + \text{experimental error}$$

The three terms with single "*X*'s" are the *main* *effects* terms. There are $k(k\text{-}1)/2 = 3*2/2 = 3$ *two-way* *interaction* terms and 1 *three-way* interaction term (which is often omitted, for simplicity). When the experimental data are analyzed, all the unknown "β" parameters are estimated and the coefficients of the "*X*" terms are tested to see which ones are significantly different from 0.

Quadratic model

A second-order (quadratic) model (typically used in *response surface* DOE's with suspected curvature) does not include the three-way interaction term but adds three more terms to the linear model, namely

$$\beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{33} X_3^2.$$

Note: Clearly, a full model could include many cross-product (or interaction) terms involving squared X's. However, in general these terms are not needed and most DOE software defaults to leaving them out of the model.

5.1.2. What are the uses of DOE?

DOE is a multipurpose tool that can help in many situations

Below are seven examples illustrating situations in which experimental design can be used effectively:

- Choosing Between Alternatives
- Selecting the Key Factors Affecting a Response
- Response Surface Modeling to:
  - Hit a Target
  - Reduce Variability
  - Maximize or Minimize a Response
  - Make a Process Robust (i.e., the process gets the "right" results even though there are uncontrollable "noise" factors)
  - Seek Multiple Goals
- Regression Modeling

Choosing Between Alternatives (Comparative Experiment)

A common use is planning an experiment to gather data to make a decision between two or more alternatives

Supplier A vs. supplier B? Which new additive is the most effective? Is catalyst `x' an improvement over the existing catalyst? These and countless other choices between alternatives can be presented to us in a never-ending parade. Often we have the choice made for us by outside factors over which we have no control. But in many cases we are also asked to make the choice. It helps if one has valid data to back up one's decision.

The preferred solution is to agree on a measurement by which competing choices can be compared, generate a sample of data from each alternative, and compare average results. The 'best' average outcome will be our preference. *We have performed a comparative experiment!*

Types of comparitive studies

Sometimes this comparison is performed under one common set of conditions. This is a comparative study with a narrow scope - which is suitable for some initial comparisons of possible alternatives. Other comparison studies, intended to validate that one alternative is perferred over a wide range of conditions, will purposely and systematically vary the background conditions under which the primary comparison is made in order to reach a conclusion that will be proven valid over a broad scope. We discuss experimental designs for each of these types of comparisons in Sections 5.3.3.1 and 5.3.3.2.

Selecting the Key Factors Affecting a Response (Screening Experiments)

Selecting the few that matter from the many possible factors

Often there are many possible factors, some of which may be critical and others which may have little or no effect on a response. It may be desirable, as a goal by itself, to reduce the number of factors to a relatively small set (2-5) so that attention can be focussed on controlling those factors with appropriate specifications, control charts, etc.

Screening experiments are an efficient way, with a minimal number of runs, of determining the important factors. They may also be used as a first step when the ultimate goal is to model a response with a response surface. We will discuss experimental designs for screening a large number of factors in Sections 5.3.3.3, 5.3.3.4 and 5.3.3.5.

Response Surface Modeling a Process

Some reasons to model a process

Once one knows the primary variables (factors) that affect the responses of interest, a number of additional objectives may be pursued. These include:

- Hitting a Target
- Maximizing or Minimizing a Response
- Reducing Variation
- Making a Process Robust
- Seeking Multiple Goals

What each of these purposes have in common is that experimentation is used to fit a model that may permit a rough, local approximation to the actual surface. Given that the particular objective can be met with such an approximate model, the experimental effort is kept to a minimum while still achieving the immediate goal.

These response surface modeling objectives will now be briefly expanded upon.

Hitting a Target

Often we want to "fine tune" a process to consistently hit a target

This is a frequently encountered goal for an experiment.

One might try out different settings until the desired target is `hit' consistently. For example, a machine tool that has been recently overhauled may require some setup `tweaking' before it runs on target. Such action is a small and common form of experimentation. However, rather than experimenting in an ad hoc manner until we happen to find a setup that hits the target, one can fit a model estimated from a small experiment and use this model to determine the necessary adjustments to hit the target.

More complex forms of experimentation, such as the determination of the correct chemical mix of a coating that will yield a desired refractive index for the dried coat (and simultaneously achieve specifications for other attributes), may involve many ingredients and be very sensitive to small changes in the percentages in the mix. Fitting suitable models, based on sequentially planned experiments, may be the only way to efficiently achieve this goal of hitting targets for multiple responses simultaneously.

Maximizing or Minimizing a Response

Optimizing a process output is a common goal

Many processes are being run at sub-optimal settings, some of them for years, even though each factor has been optimized individually over time. Finding settings that increase yield or decrease the amount of scrap and rework represent opportunities for substantial financial gain. Often, however, one must experiment with multiple inputs to achieve a better output. Section 5.3.3.6 on second-order designs plus material in Section 5.5.3 will be useful for these applications.

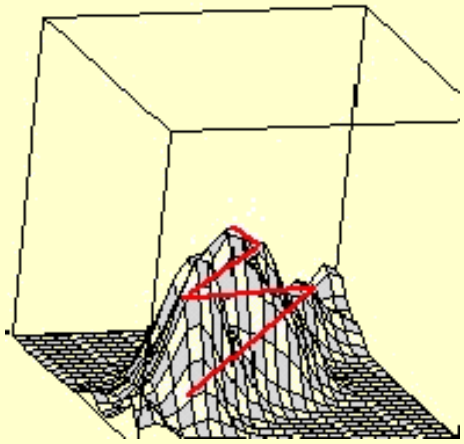


FIGURE 1.1 Pathway up the process response surface to an `optimum'

Reducing Variation

Processes that are on target, on the average, may still have too much variability

A process may be performing with unacceptable consistency, meaning its internal variation is too high.

Excessive variation can result from many causes. Sometimes it is due to the lack of having or following standard operating procedures. At other times, excessive variation is due to certain hard-to-control inputs that affect the critical output characteristics of the process. When this latter situation is the case, one may experiment with these hard-to-control factors, looking for a region where the surface is flatter and the process is easier to manage. To take advantage of such flatness in the surface, one must use designs - such as the second-order designs of Section 5.3.3.6 - that permit identification of these features. Contour or surface plots are useful for elucidating the key features of these fitted models. See also 5.5.3.1.4.

Graph of data before variation reduced

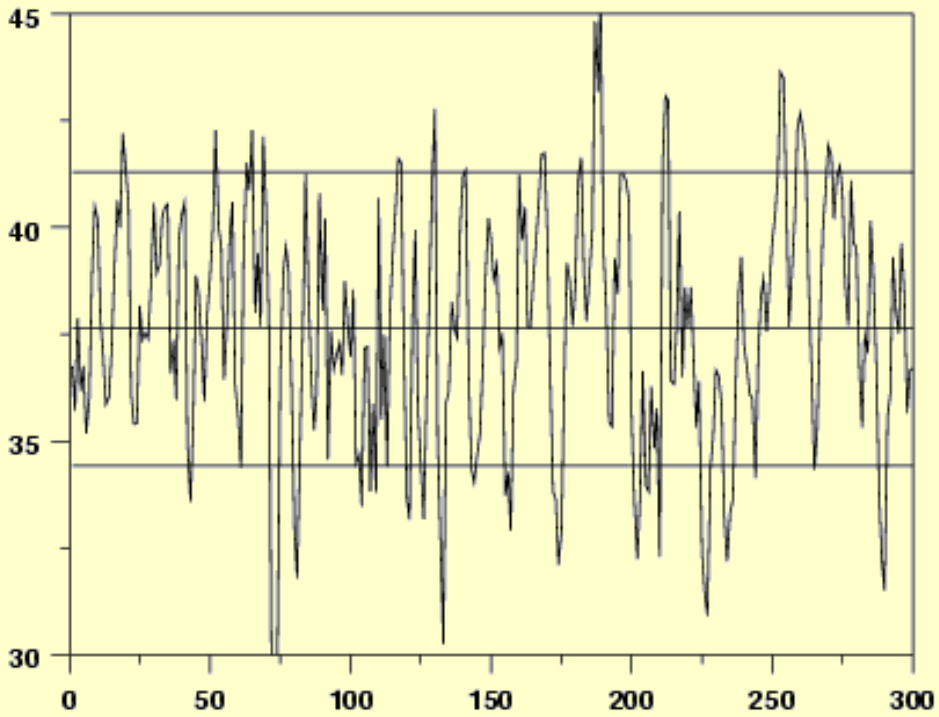


Figure 1.2 Process before variation reduced

It might be possible to reduce the variation by altering the setpoints (recipe) of the process, so that it runs in a more `stable' region.

Graph of data after process variation reduced

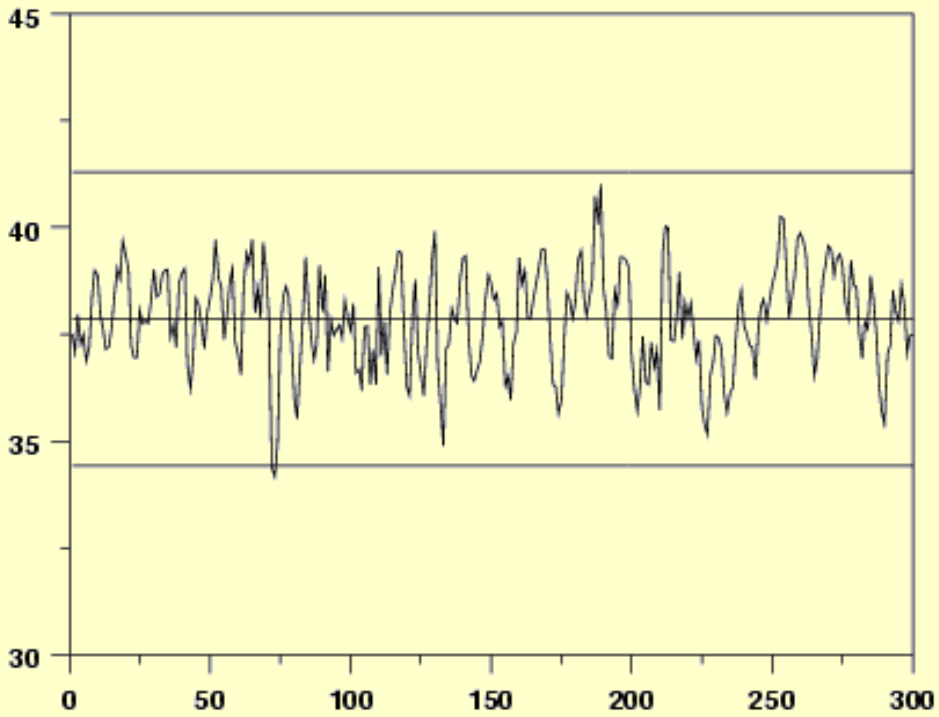


Figure 1.3 Process after variation reduced

Finding this new recipe could be the subject of an experiment, especially if there are many input factors that could conceivably affect the output.

Making a Process Robust

The less a process or product is affected by external conditions, the better it is - this is called "Robustness"

An item designed and made under controlled conditions will be later `field tested' in the hands of the customer and may prove susceptible to failure modes not seen in the lab or thought of by design. An example would be the starter motor of an automobile that is required to operate under extremes of external temperature. A starter that performs under such a wide range is termed `robust' to temperature.

Designing an item so that it is robust calls for a special experimental effort. It is possible to stress the item in the design lab and so determine the critical components affecting its performance. A different gauge of armature wire might be a solution to the starter motor, but so might be many other alternatives. The correct combination of factors can be found only by experimentation.

Seeking Multiple Goals

Sometimes we have multiple outputs and we have to compromise to achieve desirable outcomes - DOE can help here

A product or process seldom has just one desirable output characteristic. There are usually several, and they are often interrelated so that improving one will cause a deterioration of another. For example: rate vs. consistency; strength vs. expense; etc.

Any product is a trade-off between these various desirable final characteristics. Understanding the boundaries of the trade-off allows one to make the correct choices. This is done by either constructing some weighted objective function (`*desirability function*') and optimizing it, or examining contour plots of responses generated by a computer program, as given below.

Sample contour plot of deposition rate and capability

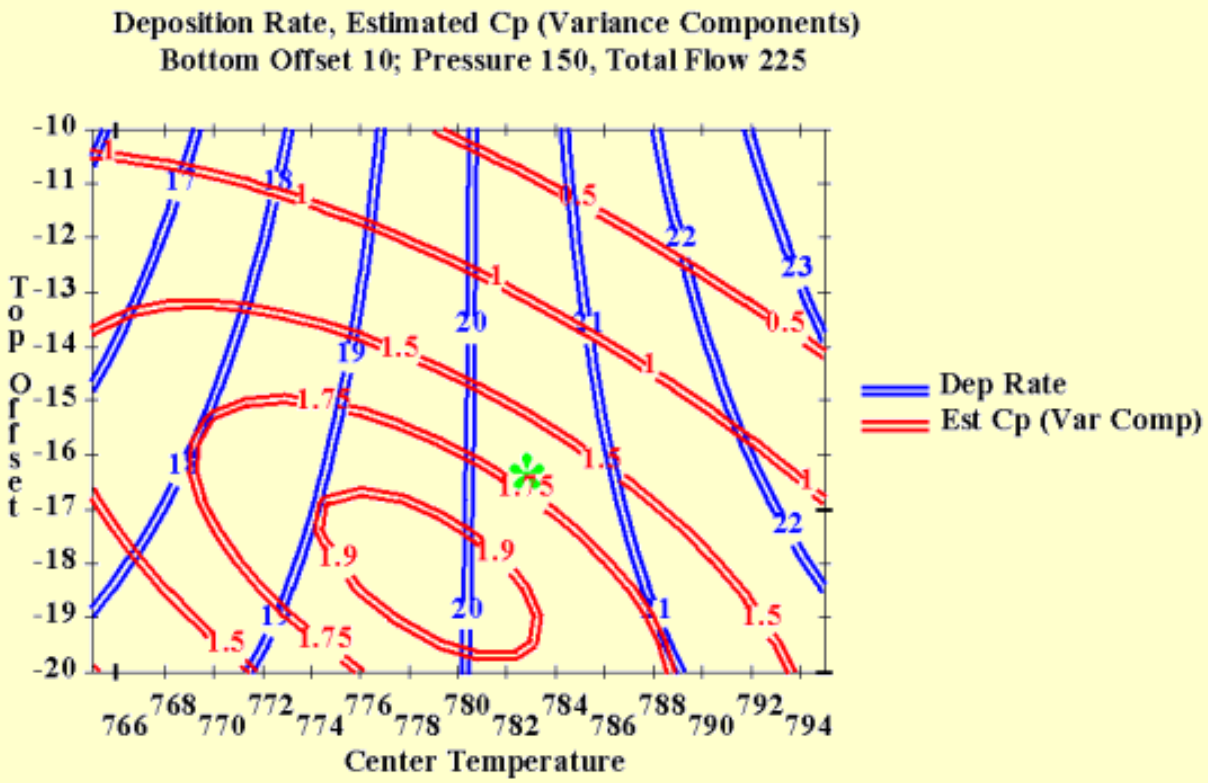


FIGURE 1.4 Overlaid contour plot of Deposition Rate and Capability (Cp)

Regression Modeling

Regression models (Chapter 4) are used to fit more precise models

Sometimes we require more than a rough approximating model over a local region. In such cases, the standard designs presented in this chapter for estimating first- or second-order polynomial models may not suffice. Chapter 4 covers the topic of experimental design and analysis for fitting general models for a single explanatory factor. If one has multiple factors, and either a nonlinear model or some other special model, the computer-aided designs of Section 5.5.2 may be useful.

5.1.3. What are the steps of DOE?

Key steps for DOE

Obtaining good results from a DOE involves these seven steps:

1. Set objectives
2. Select process variables
3. Select an experimental design
4. Execute the design
5. Check that the data are consistent with the experimental assumptions
6. Analyze and interpret the results
7. Use/present the results (may lead to further runs or DOE's).

A checklist of practical considerations

Important practical considerations in planning and running experiments are

- Check performance of gauges/measurement devices first.
- Keep the experiment as simple as possible.
- Check that all planned runs are feasible.
- Watch out for process drifts and shifts during the run.
- Avoid unplanned changes (e.g., swap operators at halfway point).
- Allow some time (and back-up material) for unexpected events.
- Obtain buy-in from all parties involved.
- Maintain effective ownership of each step in the experimental plan.
- Preserve all the raw data--do not keep only summary averages!
- Record everything that happens.
- Reset equipment to its original state after the experiment.

The *Sequential* or *Iterative* Approach to DOE

Planning to do a sequence of small experiments is often better than relying on one big experiment to give you all the answers

It is often a mistake to believe that `one big experiment will give the answer.'

A more useful approach to experimental design is to recognize that while one experiment might provide a useful result, it is more common to perform two or three, or maybe more, experiments before a complete answer is attained. In other words, an iterative approach is best and, in the end, most economical. Putting all one's eggs in one basket is not advisable.

Each stage provides insight for next stage

The reason an iterative approach frequently works best is because it is logical to move through stages of experimentation, each stage providing insight as to how the next experiment should be run.

5. Process Improvement

5.2. Assumptions

We should check the engineering and model-building assumptions that are made in most DOE's

In all model building we make assumptions, and we also require certain conditions to be approximately met for purposes of estimation. This section looks at some of the engineering and mathematical assumptions we typically make. These are:

- Are the measurement systems capable for all of your responses?
- Is your process stable?
- Are your responses likely to be approximated well by simple polynomial models?
- Are the residuals (the difference between the model predictions and the actual observations) well behaved?

5.2.1. Is the measurement system capable?

Metrology capabilities are a key factor in most experiments

It is unhelpful to find, after you have finished all the experimental runs, that the measurement devices you have at your disposal cannot measure the changes you were hoping to see. Plan to check this out before embarking on the experiment itself. Measurement process characterization is covered in Chapter 2.

SPC check of measurement devices

In addition, it is advisable, especially if the experimental material is planned to arrive for measurement over a protracted period, that an SPC (i.e., quality control) check is kept on all measurement devices from the start to the conclusion of the whole experimental project. Strange experimental outcomes can often be traced to `hiccups' in the metrology system.

5.2.2. Is the process stable?

Plan to examine process stability as part of your experiment

Experimental runs should have control runs that are made at the `standard' process setpoints, or at least at some standard operating recipe. The experiment should start and end with such runs. A plot of the outcomes of these control runs will indicate if the underlying process itself has drifted or shifted during the experiment.

It is desirable to experiment on a stable process. However, if this cannot be achieved, then the process instability must be accounted for in the analysis of the experiment. For example, if the mean is shifting with time (or experimental trial run), then it will be necessary to include a trend term in the experimental model (i.e., include a time variable or a run number variable).

5. Process Improvement
5.2. Assumptions

5.2.3. Is there a simple model?

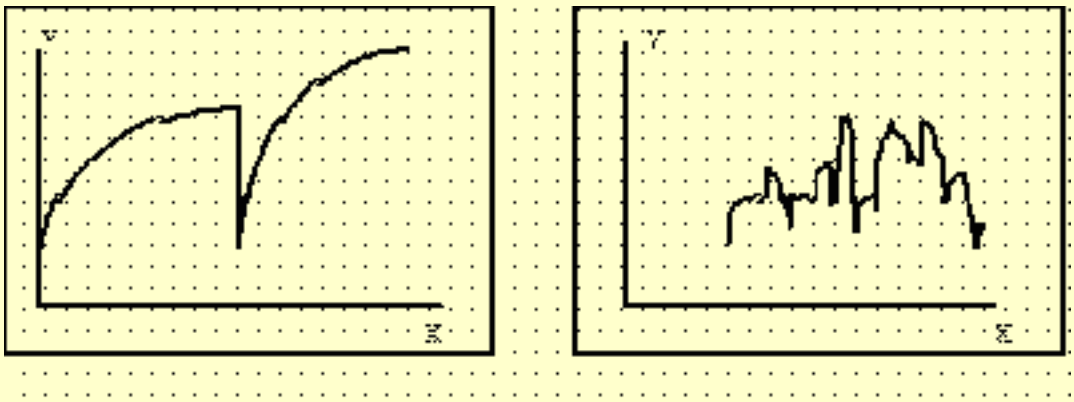
Polynomial approximation models only work for smoothly varying outputs

In this chapter we restrict ourselves to the case for which the response variable(s) are continuous outputs denoted as *Y*. Over the experimental range, the outputs must not only be continuous, but also reasonably smooth. A sharp falloff in *Y* values is likely to be missed by the approximating polynomials that we use because these polynomials assume a smoothly curving underlying response surface.

Piecewise smoothness requires separate experiments

If the surface under investigation is known to be only piecewise smooth, then the experiments will have to be broken up into separate experiments, each investigating the shape of the separate sections. A surface that is known to be very jagged (i.e., non-smooth) will not be successfully approximated by a smooth polynomial.

Examples of piecewise smooth and jagged responses



Piecewise Smooth **Jagged**

FIGURE 2.1 Examples of Piecewise Smooth and Jagged Responses

5.2.4. Are the model residuals well-behaved?

Residuals are the differences between the observed and predicted responses

Residuals are estimates of experimental error obtained by *subtracting the observed responses from the predicted responses.*

The predicted response is calculated from the chosen model, after all the unknown model parameters have been estimated from the experimental data.

Examining residuals is a key part of all statistical modeling, including DOE's. Carefully looking at residuals can tell us whether our assumptions are reasonable and our choice of model is appropriate.

Residuals are elements of variation unexplained by fitted model

Residuals can be thought of as elements of variation unexplained by the fitted model. Since this is a form of error, the same general assumptions apply to the group of residuals that we typically use for errors in general: *one expects them to be (roughly) normal and (approximately) independently distributed with a mean of 0 and some constant variance*.

Assumptions for residuals

These are the assumptions behind ANOVA and classical regression analysis. This means that an analyst should expect a regression model to err in predicting a response in a random fashion; the model should predict values higher than actual and lower than actual with equal probability. In addition, the level of the error should be independent of when the observation occurred in the study, or the size of the observation being predicted, or even the factor settings involved in making the prediction. The overall pattern of the residuals should be similar to the bell-shaped pattern observed when plotting a histogram of normally distributed data.

We emphasize the use of *graphical methods* to examine residuals.

Departures indicate inadequate model

Departures from these assumptions usually mean that the residuals contain *structure* that is not accounted for in the model. Identifying that structure and adding term(s) representing it to the original model leads to a better model.

Tests for Residual Normality

Plots for examining residuals

Any graph suitable for displaying the distribution of a set of data is suitable for judging the normality of the distribution of a group of residuals. The three most common types are:

1. histograms,
2. normal probability plots, and
3. dot plots.

Histogram

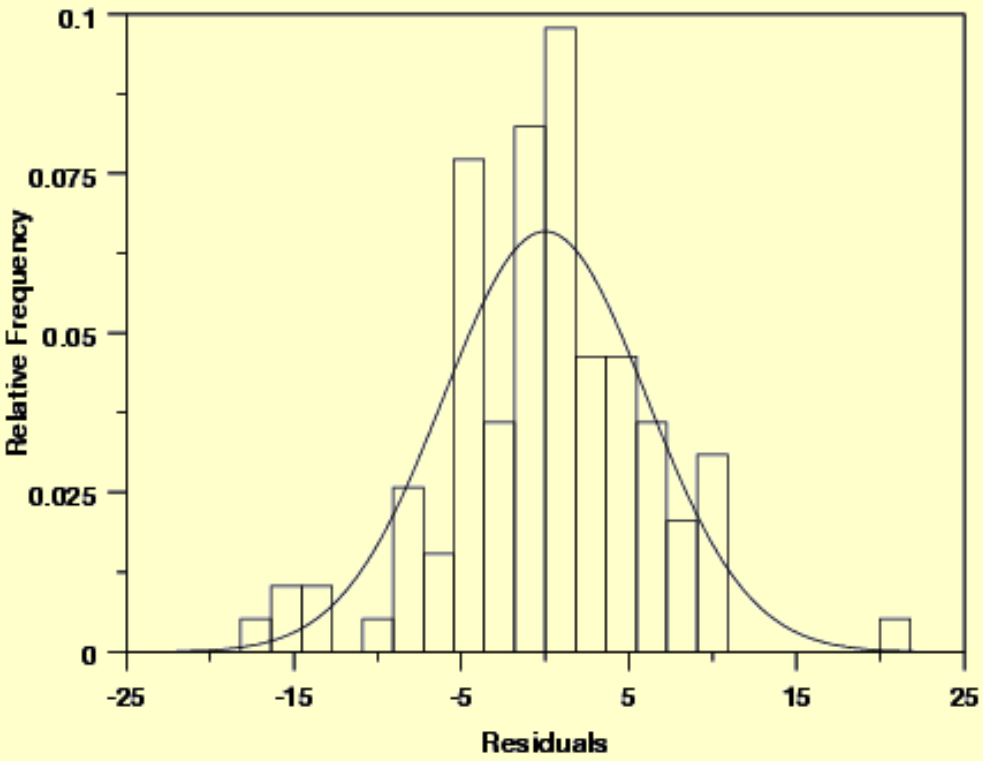


Figure 2.2

The histogram is a frequency plot obtained by placing the data in regularly spaced cells and plotting each cell frequency versus the center of the cell. Figure 2.2 illustrates an approximately normal distribution of residuals produced by a model for a calibration process. We have superimposed a normal density function on the histogram.

Small sample sizes

Sample sizes of residuals are generally small (<50) because experiments have limited treatment combinations, so a histogram is not be the best choice for judging the distribution of residuals. A more sensitive graph is the normal probability plot.

Normal probability plot

The steps in forming a normal probability plot are:

- Sort the residuals into ascending order.
- Calculate the cumulative probability of each residual using the formula:

  $$P(i\text{-th residual}) = i/(N+1)$$

  with P denoting the cumulative probability of a point, i is the order of the value in the list and N is the number of entries in the list.
- Plot the calculated p-values versus the residual value on normal probability paper.

The normal probability plot should produce an approximately straight line if the points come from a normal distribution.

Sample normal probability plot with overlaid dot plot

Figure 2.3 below illustrates the normal probability graph created from the same group of residuals used for Figure 2.2.

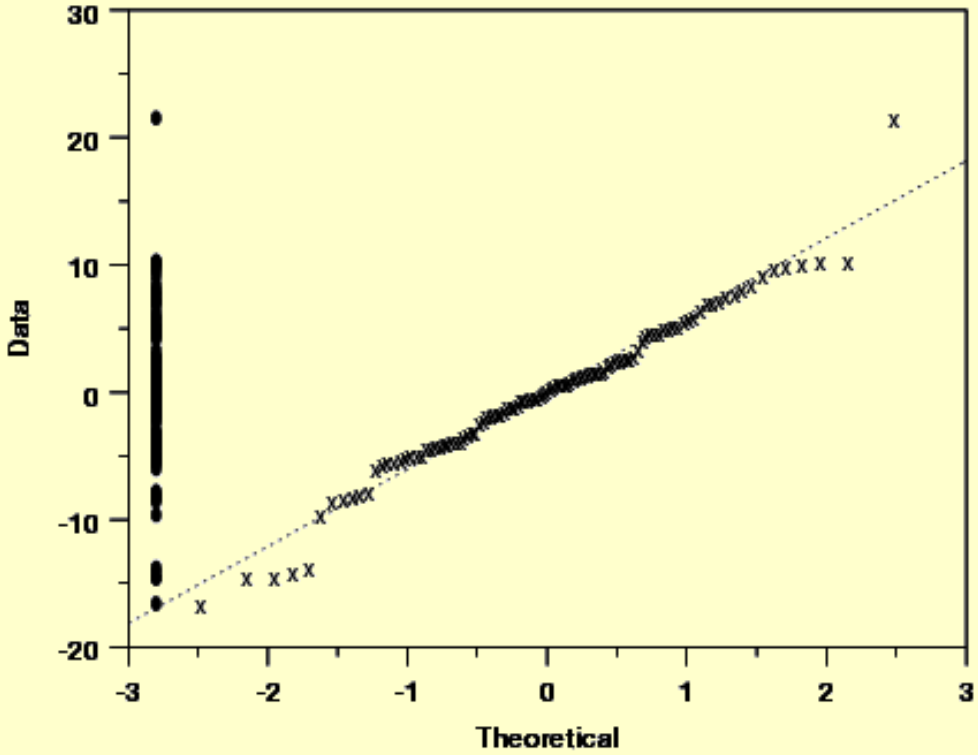


Figure 2.3

This graph includes the addition of a dot plot. The dot plot is the collection of points along the left y-axis. These are the values of the residuals. The purpose of the dot plot is to provide an indication the distribution of the residuals.

"S" shaped curves indicate bimodal distribution

Small departures from the straight line in the normal probability plot are common, but a clearly "S" shaped curve on this graph suggests a bimodal distribution of residuals. Breaks near the middle of this graph are also indications of abnormalities in the residual distribution.

NOTE: Studentized residuals are residuals converted to a scale approximately representing the standard deviation of an individual residual from the center of the residual distribution. The technique used to convert residuals to this form produces a Student's t distribution of values.

Independence of Residuals Over Time

Run sequence plot

If the order of the observations in a data table represents the order of execution of each treatment combination, then a plot of the residuals of those observations versus the case order or time order of the observations will test for any time dependency. These are referred to as run sequence plots.

Sample run sequence plot that exhibits a time trend

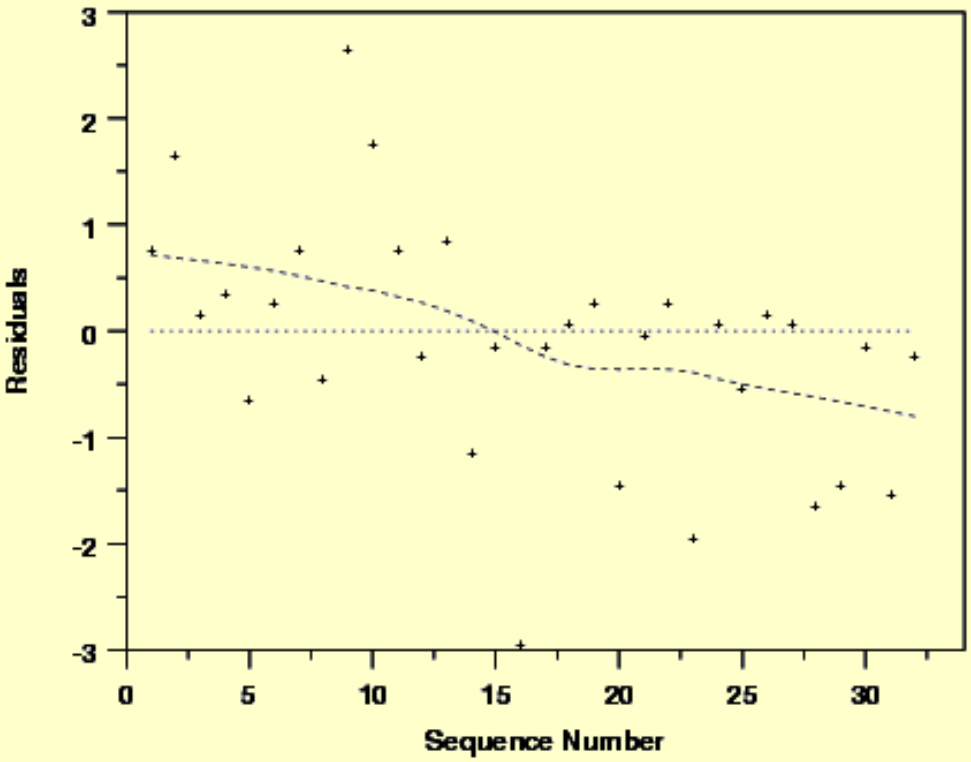


Figure 2.4

Sample run sequence plot that does not exhibit a time trend

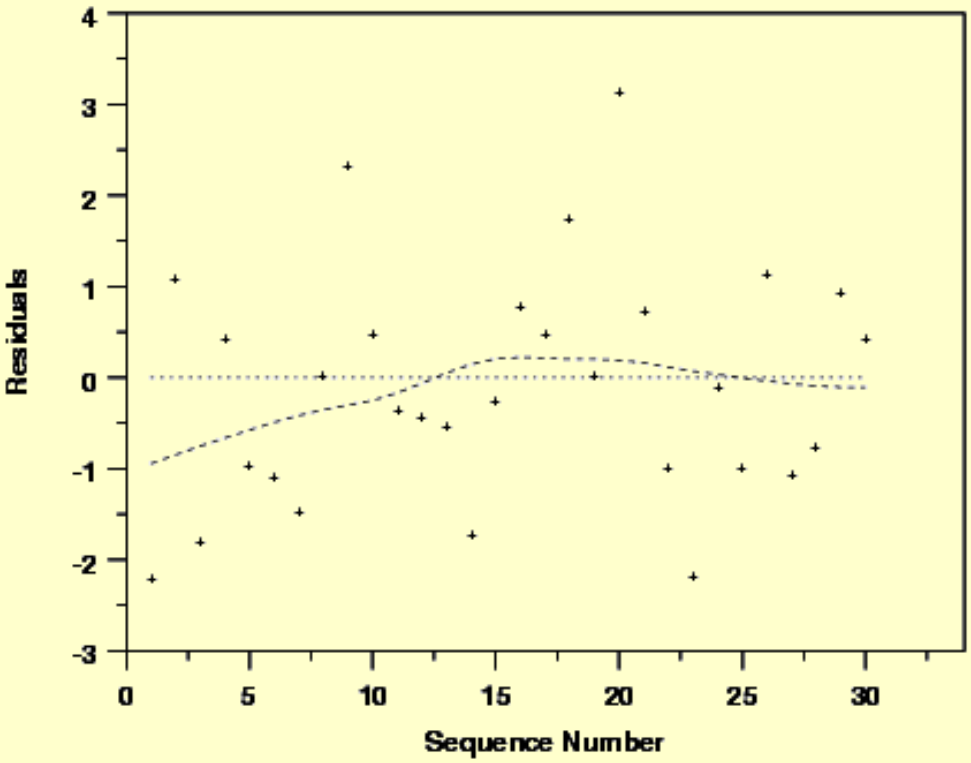


Figure 2.5

Interpretation of the sample run sequence plots

The residuals in Figure 2.4 suggest a time trend, while those in Figure 2.5 do not. Figure 2.4 suggests that the system was drifting slowly to lower values as the investigation continued. In extreme cases a drift of the equipment will produce models with very poor ability to account for the variability in the data (low R^2).

If the investigation includes centerpoints, then plotting them in time order may produce a more clear indication of a time trend if one exists. Plotting the raw responses in time sequence can also sometimes detect trend changes in a process that residual plots might not detect.

Plot of Residuals Versus Corresponding Predicted Values

Check for increasing residuals as size of fitted value increases

Plotting residuals versus the value of a fitted response should produce a distribution of points scattered randomly about 0, regardless of the size of the fitted value. Quite commonly, however, residual values may increase as the size of the fitted value increases. When this happens, the residual cloud becomes "funnel shaped" with the larger end toward larger fitted values; that is, the residuals have larger and larger scatter as the value of the response increases. Plotting the absolute values of the residuals instead of the signed values will produce a "wedge-shaped" distribution; a smoothing function is added to each graph which helps to show the trend.

Sample residuals versus fitted values plot showing increasing residuals

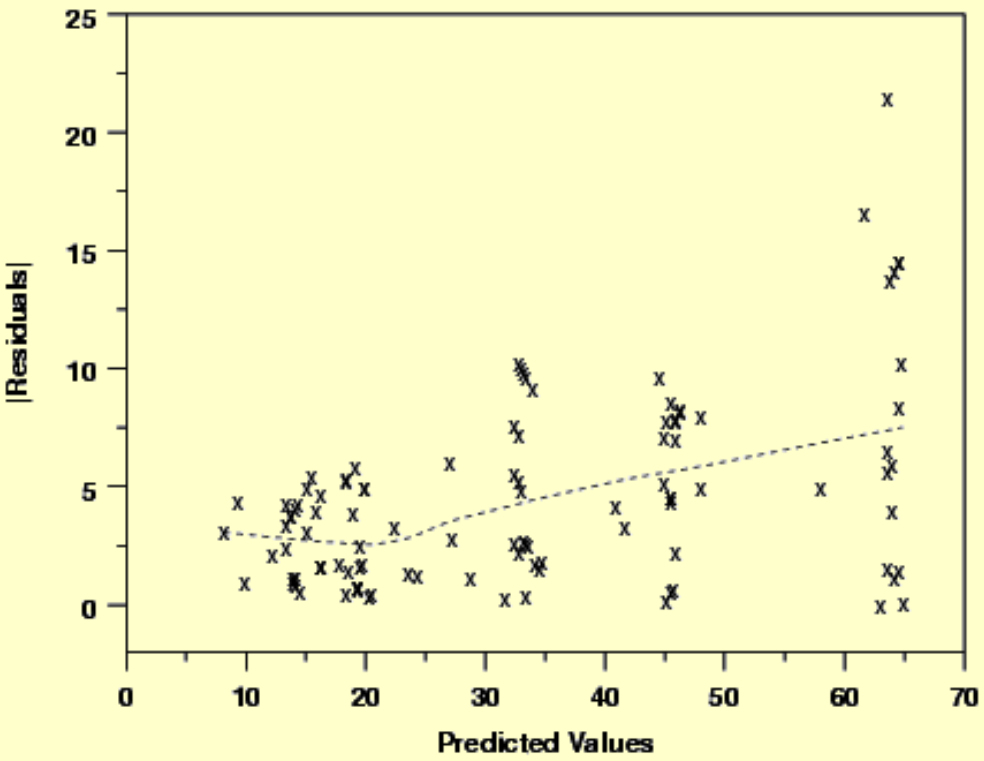


Figure 2.6

Sample residuals versus fitted values plot that does not show increasing residuals

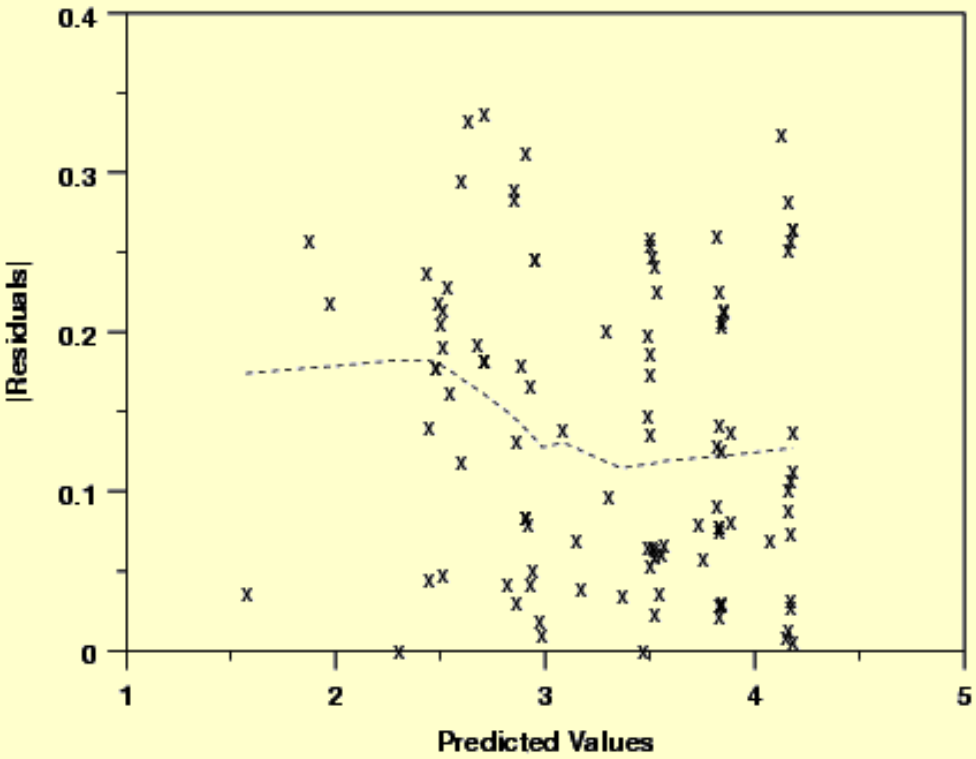


Figure 2.6

Interpretation of the residuals versus fitted values plots

A residual distribution such as that in Figure 2.6 showing a trend to higher absolute residuals as the value of the response increases suggests that one should transform the response, perhaps by modeling its logarithm or square root, etc., (contractive transformations). Transforming a response in this fashion often simplifies its relationship with a predictor variable and leads to simpler models. Later sections discuss transformation in more detail. Figure 2.7 plots the residuals after a transformation on the response variable was used to reduce the scatter. Notice the difference in scales on the vertical axes.

Independence of Residuals from Factor Settings

Sample residuals versus factor setting plot

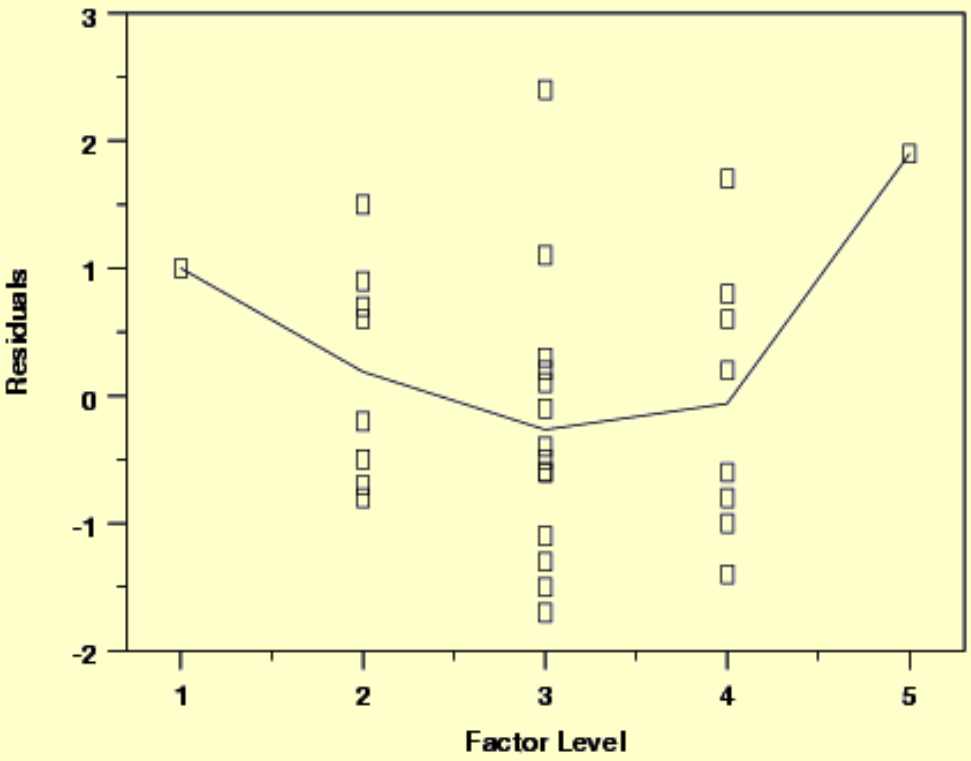


Figure 2.8

Sample residuals versus factor setting plot after adding a quadratic term

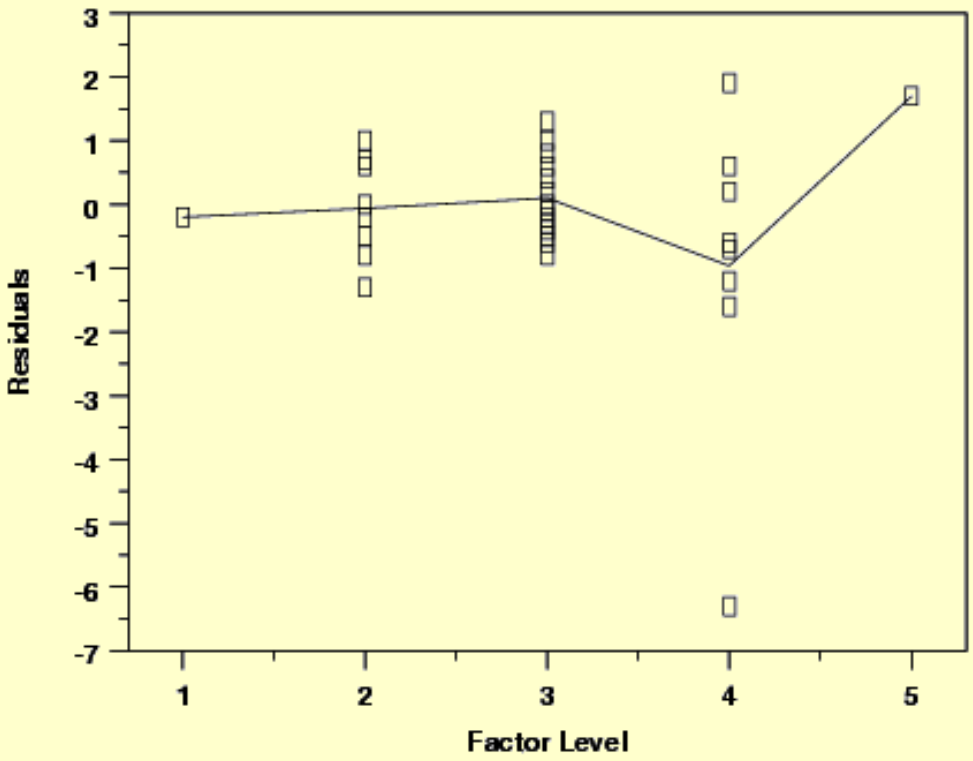


Figure 2.9

Interpreation of residuals versus factor setting plots

Figure 2.8 shows that the size of the residuals changed as a function of a predictor's settings. A graph like this suggests that the model needs a higher-order term in that predictor or that one should transform the predictor using a logarithm or square root, for example. Figure 2.9 shows the residuals for the same response after adding a quadratic term. Notice the single point widely separated from the other residuals in Figure 2.9. This point is an "outlier." That is, its position is well within the range of values used for this predictor in the investigation, but its result was somewhat lower than the model predicted. A signal that curvature is present is a trace resembling a "frown" or a "smile" in these graphs.

Sample residuals versus factor setting plot lacking one or more higher-order terms

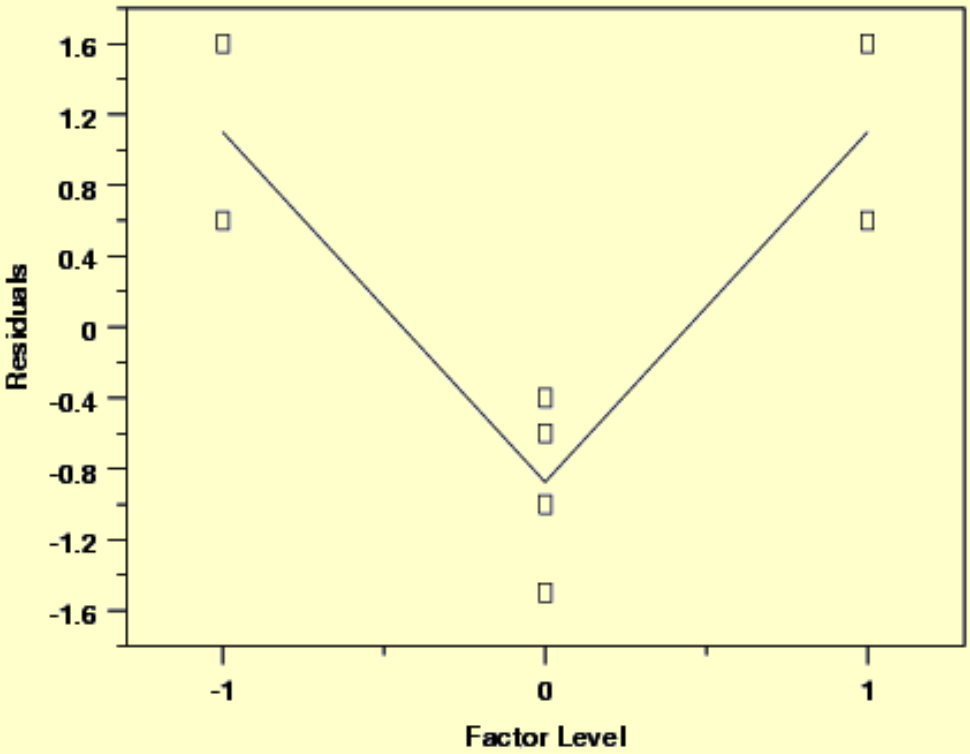


Figure 2.10

Interpretation of plot

The example given in Figures 2.8 and 2.9 obviously involves five levels of the predictor. The experiment utilized a response surface design. For the simple factorial design that includes center points, if the response model being considered lacked one or more higher-order terms, the plot of residuals versus factor settings might appear as in Figure 2.10.

Graph indicates prescence of curvature

While the graph gives a definite signal that curvature is present, identifying the source of that curvature is not possible due to the structure of the design. Graphs generated using the other predictors in that situation would have very similar appearances.

Additional discussion of residual analysis

Note: Residuals are an important subject discussed repeatedly in this Handbook. For example, graphical residual plots using Dataplot are discussed in Chapter 1 and the general examination of residuals as a part of model building is discussed in Chapter 4.

5.3. Choosing an experimental design

Contents of Section 3

This section describes in detail the process of choosing an experimental design to obtain the results you need. The basic designs an engineer needs to know about are described in detail.

Note that this section describes the basic designs used for most engineering and scientific applications

5.3.1. What are the objectives?

Planning an experiment begins with carefully considering what the objectives (or goals) are

The objectives for an experiment are best determined by a team discussion. All of the objectives should be written down, even the "unspoken" ones.

The group should discuss which objectives are the key ones, and which ones are "nice but not really necessary". Prioritization of the objectives helps you decide which direction to go with regard to the selection of the factors, responses and the particular design. Sometimes prioritization will force you to start over from scratch when you realize that the experiment you decided to run does not meet one or more critical objectives.

Types of designs

Examples of goals were given earlier in Section 5.1.2, in which we described four broad categories of experimental designs, with various objectives for each. These were:

- **Comparative designs** to:
  - choose between alternatives, with narrow scope, suitable for an initial comparison (see Section 5.3.3.1)
  - choose between alternatives, with broad scope, suitable for a confirmatory comparison (see Section 5.3.3.2)
- **Screening designs** to identify which factors/effects are important
  - when you have 2 - 4 factors and can perform a full factorial (Section 5.3.3.3)
  - when you have more than 3 factors and want to begin with as small a design as possible (Section 5.3.3.4 and 5.3.3.5)
  - when you have some qualitative factors, or you have some quantitative factors that are known to have a non-monotonic effect (Section 3.3.3.10)

Note that some authors prefer to restrict the term screening design to the case where you are trying to extract the most important factors from a large (say > 5) list of initial factors (usually a fractional factorial design). We include the case with a smaller

number of factors, usually a full factorial design, since the basic purpose and analysis is similar.

- **Response Surface** modeling to achieve one or more of the following objectives:
  - hit a target
  - maximize or minimize a response
  - reduce variation by locating a region where the process is easier to manage
  - make a process robust (note: this objective may often be accomplished with screening designs rather than with response surface designs - see Section 5.5.6)
- **Regression modeling**
  - to estimate a precise model, quantifying the dependence of response variable(s) on process inputs.

Based on objective, where to go next

After identifying the objective listed above that corresponds most closely to your specific goal, you can

- proceed to the next section in which we discuss selecting experimental factors

and then

- select the appropriate design named in section 5.3.3 that suits your objective (and follow the related links).

5.3.2. How do you select and scale the process variables?

Guidelines to assist the engineering judgment process of selecting process variables for a DOE

Process variables include both *inputs* and *outputs* - i.e., *factors* and *responses*. The selection of these variables is best done as a team effort. The team should

- Include all important factors (based on engineering judgment).
- Be bold, but not foolish, in choosing the low and high factor levels.
- Check the factor settings for impractical or impossible combinations - i.e., very low pressure and very high gas flows.
- Include all relevant responses.
- Avoid using only responses that combine two or more measurements of the process. For example, if interested in selectivity (the ratio of two etch rates), measure both rates, not just the ratio.

Be careful when choosing the allowable range for each factor

We have to choose the range of the settings for input factors, and it is wise to give this some thought beforehand rather than just try extreme values. In some cases, extreme values will give runs that are not feasible; in other cases, extreme ranges might move one out of a smooth area of the response surface into some jagged region, or close to an asymptote.

Two-level designs have just a "high" and a "low" setting for each factor

The most popular experimental designs are *two-level designs*. Why only two levels? There are a number of good reasons why two is the most common choice amongst engineers: one reason is that it is ideal for screening designs, simple and economical; it also gives most of the information required to go to a multilevel response surface experiment if one is needed.

Consider adding some center points to your two-level design

The term "two-level design" is something of a misnomer, however, as it is recommended to include some center points during the experiment (center points are located in the middle of the design `box').

Notation for 2-Level Designs

Matrix notation for describing an experiment

The standard layout for a 2-level design uses +1 and -1 notation to denote the "high level" and the "low level" respectively, for each factor. For example, the matrix below

	Factor 1 (X1)	Factor 2 (X2)
Trial 1	-1	-1
Trial 2	+1	-1
Trial 3	-1	+1
Trial 4	+1	+1

describes an experiment in which 4 trials (or runs) were conducted with each factor set to high or low during a run according to whether the matrix had a +1 or -1 set for the factor during that trial. If the experiment had more than 2 factors, there would be an additional column in the matrix for each additional factor.

Note: Some authors shorten the matrix notation for a two-level design by just recording the plus and minus signs, leaving out the "1's".

Coding the data

The use of +1 and -1 for the factor settings is called *coding* the data. This aids in the interpretation of the coefficients fit to any experimental model. *After factor settings are coded, center points have the value "0"*. Coding is described in more detail in the DOE glossary.

The Model or Analysis Matrix

Design matrices

If we add an "I" column and an "X1*X2" column to the matrix of 4 trials for a two-factor experiment described earlier, we obtain what is known as the *model or analysis matrix* for this simple experiment, which is shown below. The model matrix for a three-factor experiment is shown later in this section.

I	X1	X2	X1*X2
+1	-1	-1	+1
+1	+1	-1	-1
+1	-1	+1	-1
+1	+1	+1	+1

Model for the experiment

The model for this experiment is

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \text{experimental error}$$

and the "I" column of the design matrix has all 1's to provide for the β_0 term. The X1*X2 column is formed by multiplying the "X1" and "X2" columns together, row element by row element. This column gives interaction term for each trial.

Model in matrix notation

In matrix notation, we can summarize this experiment by

$$\boldsymbol{Y} = \boldsymbol{X}\beta + \text{experimental error}$$

for which $\boldsymbol{X}$is the 4 by 4 design matrix of 1's and -1's shown above, β is the vector of unknown model coefficients $(\beta_0, \beta_1, \beta_2, \beta_{12})$ and $\boldsymbol{Y}$ is a vector consisting of the four trial response observations.

Orthogonal Property of Scaling in a 2-Factor Experiment

Coding produces orthogonal columns

Coding is sometime called "*orthogonal coding*" since all the columns of a coded 2-factor design matrix (except the "I" column) are typically orthogonal. That is, the dot product for any pair of columns is zero. For example, for X1 and X2: (-1)(-1) + (+1)(-1) + (-1)(+1) + (+1)(+1) = 0.

5.3.3. How do you select an experimental design?

A design is selected based on the experimental objective and the number of factors

The choice of an experimental design depends on the objectives of the experiment and the number of factors to be investigated.

Experimental Design Objectives

Types of designs are listed here according to the experimental objective they meet

Types of designs are listed here according to the experimental objective they meet.

- **Comparative objective**: If you have one or several factors under investigation, but the primary goal of your experiment is to make a conclusion about one a-priori important factor, (in the presence of, and/or in spite of the existence of the other factors), and the question of interest is whether or not that factor is "significant", (i.e., whether or not there is a significant change in the response for different levels of that factor), then you have a *comparative problem* and you need a *comparative design* solution.
- **Screening objective**: The primary purpose of the experiment is to select or *screen out* the few important main effects from the many less important ones. These *screening designs* are also termed main effects designs.
- **Response Surface (method) objective**: The experiment is designed to allow us to estimate interaction and even quadratic effects, and therefore give us an idea of the (local) shape of the response surface we are investigating. For this reason, they are termed *response surface method (RSM) designs*. RSM designs are used to:
    - Find improved or optimal process settings

- Troubleshoot process problems and weak points
- Make a product or process more *robust* against external and non-controllable influences. "Robust" means relatively insensitive to these influences.

- **Optimizing responses when factors are proportions of a mixture objective**: If you have factors that are proportions of a mixture and you want to know what the "best" proportions of the factors are so as to maximize (or minimize) a response, then you need a *mixture design*.
- **Optimal fitting of a regression model objective**: If you want to model a response as a mathematical function (either known or empirical) of a few continuous factors and you desire "good" model parameter estimates (i.e., unbiased and minimum variance), then you need a *regression design.*

Mixture and regression designs

Mixture designs are discussed briefly in section 5 (Advanced Topics) and regression designs for a single factor are discussed in chapter 4. Selection of designs for the remaining 3 objectives is summarized in the following table.

Summary table for choosing an experimental design for comparative, screening, and response surface designs

TABLE 3.1 Design Selection Guideline

Number of Factors	Comparative Objective	Screening Objective	Response Surface Objective
1	1-factor completely randomized design	–	–
2 - 4	Randomized block design	Full or fractional factorial	Central composite or Box-Behnken
5 or more	Randomized block design	Fractional factorial or Plackett-Burman	Screen first to reduce number of factors

Resources and degree of control over wrong decisions

Choice of a design from within these various types depends on the amount of resources available and the degree of control over making wrong decisions (Type I and Type II errors for testing hypotheses) that the experimenter desires.

Save some runs for center points and "redos" that might be needed

It is a good idea to choose a design that requires somewhat fewer runs than the budget permits, so that center point runs can be added to check for curvature in a 2-level screening design and backup resources are available to redo runs that have processing mishaps.

5.3.3.1. Completely randomized designs

These designs are for studying the effects of one primary factor without the need to take other nuisance factors into account

Here we consider completely randomized designs that have one primary factor. The experiment compares the values of a response variable based on the different levels of that primary factor.

For completely randomized designs, the levels of the primary factor are randomly assigned to the experimental units. By randomization, we mean that the run sequence of the experimental units is determined randomly. For example, if there are 3 levels of the primary factor with each level to be run 2 times, then there are 6 factorial possible run sequences (or 6! ways to order the experimental trials). Because of the replication, the number of unique orderings is 90 (since 90 = 6!/(2!*2!*2!)). An example of an unrandomized design would be to always run 2 replications for the first level, then 2 for the second level, and finally 2 for the third level. To randomize the runs, one way would be to put 6 slips of paper in a box with 2 having level 1, 2 having level 2, and 2 having level 3. Before each run, one of the slips would be drawn blindly from the box and the level selected would be used for the next run of the experiment.

Randomization typically performed by computer software

In practice, the randomization is typically performed by a computer program (in Dataplot, see the Generate Random Run Sequence menu under the main DEX menu). However, the randomization can also be generated from random number tables or by some physical mechanism (e.g., drawing the slips of paper).

Three key numbers

All completely randomized designs with one primary factor are defined by 3 numbers:

k = number of factors (= 1 for these designs)
L = number of levels
n = number of replications

and the total sample size (number of runs) is $N = k \; x \; L \; x \; n$.

Balance

Balance dictates that the number of replications be the same at each level of the factor (this will maximize the sensitivity of subsequent statistical *t* (or *F*) tests).

Typical example of a completely randomized design

A typical example of a completely randomized design is the following:

k = 1 factor ($X1$)
L = 4 levels of that single factor (called "1", "2", "3", and "4")
n = 3 replications per level
N = 4 levels * 3 replications per level = 12 runs

A sample randomized sequence of trials

The randomized sequence of trials might look like:

X1
3
1
4
2
2
1
3
4
1
2
4
3

Note that in this example there are 12!/(3!*3!*3!*3!) = 369,600 ways to run the experiment, all equally likely to be picked by a randomization procedure.

Model for a completely randomized design

The model for the response is

$$Y_{i,j} = \mu + T_i + \text{random error}$$

with

$Y_{i,j}$ being any observation for which $X1 = i$
μ (or mu) is the general location parameter
T_i is the effect of having treatment level i

Estimates and Statistical Tests

Estimating and testing model factor levels

Estimate for μ : $\bar{Y}$ = the average of all the data

Estimate for T_i : $\bar{Y}_i - \bar{Y}$

with $\bar{Y}_i$ = average of all Y for which $X1 = i$.

Statistical tests for levels of $X1$ are shown in the section on one-way ANOVA in Chapter 7.

5.3.3.2. Randomized block designs

Blocking to "remove" the effect of nuisance factors

For randomized block designs, there is one factor or variable that is of primary interest. However, there are also several other nuisance factors.

Nuisance factors are those that may affect the measured result, but are not of primary interest. For example, in applying a treatment, nuisance factors might be the specific operator who prepared the treatment, the time of day the experiment was run, and the room temperature. All experiments have nuisance factors. The experimenter will typically need to spend some time deciding which nuisance factors are important enough to keep track of or control, if possible, during the experiment.

Blocking used for nuisance factors that can be controlled

When we can control nuisance factors, an important technique known as blocking can be used to reduce or eliminate the contribution to experimental error contributed by nuisance factors. The basic concept is to create homogeneous blocks in which the nuisance factors are held constant and the factor of interest is allowed to vary. Within blocks, it is possible to assess the effect of different levels of the factor of interest without having to worry about variations due to changes of the block factors, which are accounted for in the analysis.

Definition of blocking factors

A nuisance factor is used as a blocking factor if every level of the primary factor occurs the same number of times with each level of the nuisance factor. The analysis of the experiment will focus on the effect of varying levels of the primary factor within each block of the experiment.

Block for a few of the most important nuisance factors

The general rule is:

"Block what you can, randomize what you cannot."

Blocking is used to remove the effects of a few of the most important nuisance variables. Randomization is then used to reduce the contaminating effects of the remaining nuisance variables.

Table of randomized block designs

One useful way to look at a randomized block experiment is to consider it as a collection of completely randomized experiments, each run within one of the blocks of the total experiment.

Randomized Block Designs (RBD)

Name of Design	Number of Factors k	Number of Runs n
2-factor RBD	2	$L_1 * L_2$
3-factor RBD	3	$L_1 * L_2 * L_3$
4-factor RBD	4	$L_1 * L_2 * L_3 * L_4$
.	.	.
k-factor RBD	k	$L_1 * L_2 * ... * L_k$

with

L_1 = number of levels (settings) of factor 1
L_2 = number of levels (settings) of factor 2
L_3 = number of levels (settings) of factor 3
L_4 = number of levels (settings) of factor 4

.
.
.

L_k = number of levels (settings) of factor k

Example of a Randomized Block Design

Example of a randomized block design

Suppose engineers at a semiconductor manufacturing facility want to test whether different wafer implant material dosages have a significant effect on resistivity measurements after a diffusion process taking place in a furnace. They have four different dosages they want to try and enough experimental wafers from the same lot to run three wafers at each of the dosages.

Furnace run is a nuisance factor

The nuisance factor they are concerned with is "furnace run" since it is known that each furnace run differs from the last and impacts many process parameters.

Ideal would be to eliminate nuisance furnace factor

An ideal way to run this experiment would be to run all the 4x3=12 wafers in the same furnace run. That would eliminate the nuisance furnace factor completely. However, regular production wafers have furnace priority, and only a few experimental wafers are allowed into any furnace run at the same time.

Non-Blocked method

A non-blocked way to run this experiment would be to run each of the twelve experimental wafers, in random order, one per furnace run. That would increase the experimental error of each resistivity measurement by the run-to-run furnace variability and make it more difficult to study the effects of the different dosages. The blocked way to run this experiment, assuming you can convince manufacturing to let you put four experimental wafers in a furnace run, would be to put four wafers with different dosages in each of three furnace runs. The only randomization would be choosing which of the three wafers with dosage 1 would go into furnace run 1, and similarly for the wafers with dosages 2, 3 and 4.

Description of the experiment

Let $X1$ be dosage "level" and $X2$ be the blocking factor furnace run. Then the experiment can be described as follows:

k = 2 factors (1 primary factor $X1$ and 1 blocking factor $X2$)
L_1 = 4 levels of factor $X1$
L_2 = 3 levels of factor $X2$
n = 1 replication per cell
$N = L_1 * L_2 = 4 * 3 = 12$ runs

Design trial before randomization

Before randomization, the design trials look like:

$X1$	$X2$
1	1
1	2
1	3
2	1
2	2
2	3
3	1
3	2
3	3
4	1
4	2
4	3

Matrix representation

An alternate way of summarizing the design trials would be to use a 4x3 matrix whose 4 rows are the levels of the treatment $X1$ and whose columns are the 3 levels of the blocking variable $X2$. The cells in the matrix have indices that match the $X1$, $X2$ combinations above.

By extension, note that the trials for any K-factor randomized block design are simply the cell indices of a K dimensional matrix.

Model for a Randomized Block Design

Model for a randomized block design

The model for a randomized block design with one nuisance variable is

$$Y_{i,j} = \mu + T_i + B_j + \text{random error}$$

where

$Y_{i,j}$ is any observation for which $X1 = i$ and $X2 = j$
$X1$ is the primary factor
$X2$ is the blocking factor
μ is the general location parameter (i.e., the mean)
T_i is the effect for being in treatment i (of factor $X1$)
B_j is the effect for being in block j (of factor $X2$)

Estimates for a Randomized Block Design

Estimating factor effects for a randomized block design

Estimate for μ : $\bar{Y}$ = the average of all the data

Estimate for T_i : $\bar{Y}_i - \bar{Y}$

with $\bar{Y}_i$ = average of all Y for which $X1 = i$.

Estimate for B_j : $\bar{Y}_j - \bar{Y}$

with $\bar{Y}_j$ = average of all Y for which $X2 = j$.

5.3.3.2.1. Latin square and related designs

Latin square (and related) designs are efficient designs to block from 2 to 4 nuisance factors

Latin square designs, and the related Graeco-Latin square and Hyper-Graeco-Latin square designs, are a special type of comparative design.

There is a single factor of primary interest, typically called the treatment factor, and several nuisance factors. For Latin square designs there are 2 nuisance factors, for Graeco-Latin square designs there are 3 nuisance factors, and for Hyper-Graeco-Latin square designs there are 4 nuisance factors.

Nuisance factors used as blocking variables

The nuisance factors are used as blocking variables.

1. For Latin square designs, the 2 nuisance factors are divided into a tabular grid with the property that each row and each column receive each treatment exactly once.
2. As with the Latin square design, a Graeco-Latin square design is a *k*x*k* tabular grid in which *k* is the number of levels of the treatment factor. However, it uses 3 blocking variables instead of the 2 used by the standard Latin square design.
3. A Hyper-Graeco-Latin square design is also a *k*x*k* tabular grid with *k* denoting the number of levels of the treatment factor. However, it uses 4 blocking variables instead of the 2 used by the standard Latin square design.

Advantages and disadvantages of Latin square designs

The advantages of Latin square designs are:

1. They handle the case when we have several nuisance factors and we either cannot combine them into a single factor or we wish to keep them separate.
2. They allow experiments with a relatively small number of runs.

The disadvantages are:

1. The number of levels of each blocking variable must equal the number of levels of the treatment factor.
2. The Latin square model assumes that there are no interactions between the blocking variables or between the treatment variable and the blocking variable.

Note that Latin square designs are equivalent to specific fractional factorial designs (e.g., the 4x4 Latin square design is equivalent to a 4^{3-1} fractional factorial design).

Summary of designs

Several useful designs are described in the table below.

Some Useful Latin Square, Graeco-Latin Square and Hyper-Graeco-Latin Square Designs

Name of Design	Number of Factors *k*	Number of Runs N
3-by-3 Latin Square	3	9
4-by-4 Latin Square	3	16
5-by-5 Latin Square	3	25
3-by-3 Graeco-Latin Square	4	9
4-by-4 Graeco-Latin Square	4	16
5-by-5 Graeco-Latin Square	4	25
4-by-4 Hyper-Graeco-Latin Square	5	16
5-by-5 Hyper-Graeco-Latin Square	5	25

Model for Latin Square and Related Designs

Latin square design model and estimates for effect levels

The model for a response for a latin square design is

$$Y_{ijk} = \mu + R_i + C_j + T_k + \text{random error}$$

with

Y_{ijk} denoting any observation for which

$X1 = i$, $X2 = j$, $X3 = k$
$X1$ and $X2$ are blocking factors
$X3$ is the primary factor

μ denoting the general location parameter
R_i denoting the effect for block i
C_j denoting the effect for block j
T_k denoting the effect for treatment k

Models for Graeco-Latin and Hyper-Graeco-Latin squares are the obvious extensions of the Latin square model, with additional blocking variables added.

Estimates for Latin Square Designs

Estimates

Estimate for μ: $\bar{Y}$ = the average of all the data

Estimate for R_i: $\bar{Y}_i - \bar{Y}$

$\bar{Y}_i$ = average of all Y for which $X1 = i$

Estimate for C_j: $\bar{Y}_j - \bar{Y}$

$\bar{Y}_j$ = average of all Y for which $X2 = j$

Estimate for T_k: $\bar{Y}_k - \bar{Y}$

$\bar{Y}_k$ = average of all Y for which $X3 = k$

Randomize as much as design allows

Designs for Latin squares with 3-, 4-, and 5-level factors are given next. These designs show what the treatment combinations should be for each run. ***When using any of these designs, be sure to randomize the treatment units and trial order, as much as the design allows***.

For example, one recommendation is that a Latin square design be randomly selected from those available, then randomize the run order.

Latin Square Designs for 3-, 4-, and 5-Level Factors

Designs for 3-level factors (and 2 nuisance or blocking factors)

3-Level Factors

*X*1 row blocking factor	*X*2 column blocking factor	*X*3 treatment factor
1	1	1
1	2	2
1	3	3
2	1	3
2	2	1
2	3	2
3	1	2
3	2	3
3	3	1

with

k = 3 factors (2 blocking factors and 1 primary factor)
L_1 = 3 levels of factor $X1$ (block)
L_2 = 3 levels of factor $X2$ (block)
L_3 = 3 levels of factor $X3$ (primary)
$N = L1 * L2 = 9$ runs

This can alternatively be represented as

A	B	C
C	A	B
B	C	A

Designs for 4-level factors (and 2 nuisance or blocking factors)

4-Level Factors

*X*1 row blocking factor	*X*2 column blocking factor	*X*3 treatment factor
1	1	1
1	2	2
1	3	4
1	4	3
2	1	4
2	2	3
2	3	1
2	4	2
3	1	2

3	2	4
3	3	3
3	4	1
4	1	3
4	2	1
4	3	2
4	4	4

with

k = 3 factors (2 blocking factors and 1 primary factor)
L_1 = 4 levels of factor $X1$ (block)
L_2 = 4 levels of factor $X2$ (block)
L_3 = 4 levels of factor $X3$ (primary)
$N = L1 * L2$ = 16 runs

This can alternatively be represented as

A	B	D	C
D	C	A	B
B	D	C	A
C	A	B	D

Designs for 5-level factors (and 2 nuisance or blocking factors)

5-Level Factors

$X1$ row blocking factor	$X2$ column blocking factor	$X3$ treatment factor
1	1	1
1	2	2
1	3	3
1	4	4
1	5	5
2	1	3
2	2	4
2	3	5
2	4	1
2	5	2
3	1	5
3	2	1
3	3	2
3	4	3
3	5	4
4	1	2

4	2	3
4	3	4
4	4	5
4	5	1
5	1	4
5	2	5
5	3	1
5	4	2
5	5	3

with

k = 3 factors (2 blocking factors and 1 primary factor)
L_1 = 5 levels of factor $X1$ (block)
L_2 = 5 levels of factor $X2$ (block)
L_3 = 5 levels of factor $X3$ (primary)
$N = L1 * L2 = 25$ runs

This can alternatively be represented as

A	B	C	D	E
C	D	E	A	B
E	A	B	C	D
B	C	D	E	A
D	E	A	B	C

Further information

More details on Latin square designs can be found in Box, Hunter, and Hunter (1978).

5.3.3.2.2. Graeco-Latin square designs

These designs handle 3 nuisance factors

Graeco-Latin squares, as described on the previous page, are efficient designs to study the effect of one treatment factor in the presence of 3 nuisance factors. They are restricted, however, to the case in which all the factors have the same number of levels.

Randomize as much as design allows

Designs for 3-, 4-, and 5-level factors are given on this page. These designs show what the treatment combinations would be for each run. ***When using any of these designs, be sure to randomize the treatment units and trial order, as much as the design allows***.

For example, one recommendation is that a Graeco-Latin square design be randomly selected from those available, then randomize the run order.

Graeco-Latin Square Designs for 3-, 4-, and 5-Level Factors

Designs for 3-level factors

3-Level Factors

*X*1 row blocking factor	*X*2 column blocking factor	*X*3 blocking factor	*X*4 treatment factor
1	1	1	1
1	2	2	2
1	3	3	3
2	1	2	3
2	2	3	1
2	3	1	2
3	1	3	2
3	2	1	3
3	3	2	1

with

k = 4 factors (3 blocking factors and 1 primary factor)
L_1 = 3 levels of factor $X1$ (block)
L_2 = 3 levels of factor $X2$ (block)
L_3 = 3 levels of factor $X3$ (primary)
L_4 = 3 levels of factor $X4$ (primary)
$N = L1 * L2 = 9$ runs

This can alternatively be represented as (A, B, and C represent the treatment factor and 1, 2, and 3 represent the blocking factor):

A1	B2	C3
C2	A3	B1
B3	C1	A2

Designs for 4-level factors

4-Level Factors

$X1$ row blocking factor	$X2$ column blocking factor	$X3$ blocking factor	$X4$ treatment factor
1	1	1	1
1	2	2	2
1	3	3	3
1	4	4	4
2	1	2	4
2	2	1	3
2	3	4	2
2	4	3	1
3	1	3	2
3	2	4	1
3	3	1	4
3	4	2	3
4	1	4	3
4	2	3	4
4	3	2	1
4	4	1	2

with

k = 4 factors (3 blocking factors and 1 primary factor)
L_1 = 3 levels of factor $X1$ (block)
L_2 = 3 levels of factor $X2$ (block)
L_3 = 3 levels of factor $X3$ (primary)
L_4 = 3 levels of factor $X4$ (primary)

$N = L1 * L2 = 16$ runs

This can alternatively be represented as (A, B, C, and D represent the treatment factor and 1, 2, 3, and 4 represent the blocking factor):

A1	B2	C3	D4
D2	C1	B4	A3
B3	A4	D1	C2
C4	D3	A2	B1

Designs for 5-level factors

5-Level Factors

*X*1 row blocking factor	*X*2 column blocking factor	*X*3 blocking factor	*X*4 treatment factor
1	1	1	1
1	2	2	2
1	3	3	3
1	4	4	4
1	5	5	5
2	1	2	3
2	2	3	4
2	3	4	5
2	4	5	1
2	5	1	2
3	1	3	5
3	2	4	1
3	3	5	2
3	4	1	3
3	5	2	4
4	1	4	2
4	2	5	3
4	3	1	4
4	4	2	5
4	5	3	1
5	1	5	4
5	2	1	5
5	3	2	1
5	4	3	2
5	5	4	3

with

$k = 4$ factors (3 blocking factors and 1 primary factor)
$L_1 = 3$ levels of factor $X1$ (block)

L_2 = 3 levels of factor $X2$ (block)
L_3 = 3 levels of factor $X3$ (primary)
L_4 = 3 levels of factor $X4$ (primary)
$N = L1 * L2 = 25$ runs

This can alternatively be represented as (A, B, C, D, and E represent the treatment factor and 1, 2, 3, 4, and 5 represent the blocking factor):

A1	B2	C3	D4	E5
C2	D3	E4	A5	B1
E3	A4	B5	C1	D2
B4	C5	D1	E2	A3
D5	E1	A2	B3	C4

Further information

More designs are given in Box, Hunter, and Hunter (1978).

5.3.3.2.3. Hyper-Graeco-Latin square designs

These designs handle 4 nuisance factors

Hyper-Graeco-Latin squares, as described earlier, are efficient designs to study the effect of one treatment factor in the presence of 4 nuisance factors. They are restricted, however, to the case in which all the factors have the same number of levels.

Randomize as much as design allows

Designs for 4- and 5-level factors are given on this page. These designs show what the treatment combinations should be for each run. ***When using any of these designs, be sure to randomize the treatment units and trial order, as much as the design allows***.

For example, one recommendation is that a hyper-Graeco-Latin square design be randomly selected from those available, then randomize the run order.

Hyper-Graeco-Latin Square Designs for 4- and 5-Level Factors

Designs for 4-level factors (there are no 3-level factor Hyper-Graeco Latin square designs)

4-Level Factors

*X*1 row blocking factor	*X*2 column blocking factor	*X*3 blocking factor	*X*4 blocking factor	*X*5 treatment factor
1	1	1	1	1
1	2	2	2	2
1	3	3	3	3
1	4	4	4	4
2	1	4	2	3
2	2	3	1	4
2	3	2	4	1
2	4	1	3	2
3	1	2	3	4

3	2	1	4	3
3	3	4	1	2
3	4	3	2	1
4	1	3	4	2
4	2	4	3	1
4	3	1	2	4
4	4	2	1	3

with

k = 5 factors (4 blocking factors and 1 primary factor)
L_1 = 4 levels of factor $X1$ (block)
L_2 = 4 levels of factor $X2$ (block)
L_3 = 4 levels of factor $X3$ (primary)
L_4 = 4 levels of factor $X4$ (primary)
L_5 = 4 levels of factor $X5$ (primary)
$N = L1 * L2 = 16$ runs

This can alternatively be represented as (A, B, C, and D represent the treatment factor and 1, 2, 3, and 4 represent the blocking factors):

A11	B22	C33	D44
C42	D31	A24	B13
D23	C14	B41	A32
B34	A43	D12	C21

Designs for 5-level factors

5-Level Factors

$X1$ row blocking factor	$X2$ column blocking factor	$X3$ blocking factor	$X4$ blocking factor	$X5$ treatment factor
1	1	1	1	1
1	2	2	2	2
1	3	3	3	3
1	4	4	4	4
1	5	5	5	5
2	1	2	3	4
2	2	3	4	5
2	3	4	5	1
2	4	5	1	2
2	5	1	2	3
3	1	3	5	2
3	2	4	1	3
3	3	5	2	4

3	4	1	3	5
3	5	2	4	1
4	1	4	2	5
4	2	5	3	1
4	3	1	4	2
4	4	2	5	3
4	5	3	1	4
5	1	5	4	3
5	2	1	5	4
5	3	2	1	5
5	4	3	2	1
5	5	4	3	2

with

k = 5 factors (4 blocking factors and 1 primary factor)
L_1 = 5 levels of factor $X1$ (block)
L_2 = 5 levels of factor $X2$ (block)
L_3 = 5 levels of factor $X3$ (primary)
L_4 = 5 levels of factor $X4$ (primary)
L_5 = 5 levels of factor $X5$ (primary)
$N = L1 * L2 = 25$ runs

This can alternatively be represented as (A, B, C, D, and E represent the treatment factor and 1, 2, 3, 4, and 5 represent the blocking factors):

A11	B22	C33	D44	E55
D23	E34	A45	B51	C12
B35	C41	D52	E31	A24
E42	A53	B14	C25	D31
C54	D15	E21	A32	B43

Further information

More designs are given in Box, Hunter, and Hunter (1978).

5.3.3.3. Full factorial designs

Full factorial designs in two levels

A design in which every setting of every factor appears with every setting of every other factor is a full factorial design

A common experimental design is one with all input factors set at two levels each. These levels are called `high' and `low' or `+1' and `-1', respectively. A design with all possible high/low combinations of all the input factors is called a full factorial design in two levels.

If there are k factors, each at 2 levels, a full factorial design has 2^k runs.

TABLE 3.2 Number of Runs for a 2^k Full Factorial

Number of Factors	Number of Runs
2	4
3	8
4	16
5	32
6	64
7	128

Full factorial designs not recommended for 5 or more factors

As shown by the above table, when the number of factors is 5 or greater, a full factorial design requires a large number of runs and is not very efficient. As recommended in the Design Guideline Table, a fractional factorial design or a Plackett-Burman design is a better choice for 5 or more factors.

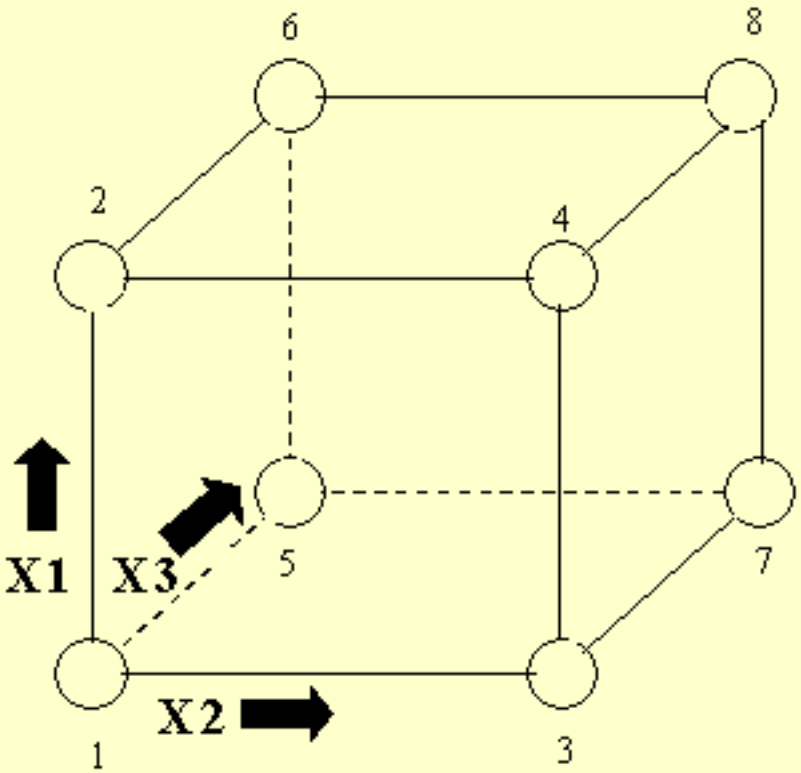
5.3.3.3.1. Two-level full factorial designs

Description

Graphical representation of a two-level design with 3 factors

Consider the two-level, full factorial design for three factors, namely the 2^3 design. This implies eight runs (not counting replications or center point runs). Graphically, we can represent the 2^3 design by the cube shown in Figure 3.1. The arrows show the direction of increase of the factors. The numbers `1' through `8' at the corners of the design box reference the `Standard Order' of runs (see Figure 3.1).

FIGURE 3.1 A 2^3 two-level, full factorial design; factors X1, X2, X3



The design matrix

In tabular form, this design is given by:

TABLE 3.3 A 2^3 two-level, full factorial design table showing runs in `Standard Order'

run	$X1$	$X2$	$X3$
1	-1	-1	-1
2	1	-1	-1
3	-1	1	-1
4	1	1	-1
5	-1	-1	1
6	1	-1	1
7	-1	1	1
8	1	1	1

The left-most column of Table 3.3, numbers 1 through 8, specifies a (non-randomized) run order called the `Standard Order.' These numbers are also shown in Figure 3.1. For example, run 1 is made at the `low' setting of all three factors.

Standard Order for a 2^k Level Factorial Design

Rule for writing a 2^k full factorial in "standard order"

We can readily generalize the 2^3 standard order matrix to a 2-level full factorial with k factors. The first ($X1$) column starts with -1 and alternates in sign for all 2^k runs. The second ($X2$) column starts with -1 repeated twice, then alternates with 2 in a row of the opposite sign until all 2^k places are filled. The third ($X3$) column starts with -1 repeated 4 times, then 4 repeats of +1's and so on. In general, the i-th column (X_i) starts with 2^{i-1} repeats of -1 folowed by 2^{i-1} repeats of +1.

Example of a 2^3 Experiment

Analysis matrix for the 3-factor complete factorial

An engineering experiment called for running three factors; namely, Pressure (*factor X*1), Table speed (*factor X*2) and Down force (*factor X*3), each at a `high' and `low' setting, on a production tool to determine which had the greatest effect on product uniformity. Two replications were run at each setting. A (full factorial) 2^3 design with 2 replications calls for 8*2=16 runs.

TABLE 3.4 Model or Analysis Matrix for a 2^3 Experiment

Model Matrix								Response Variables	
I	***X*1**	***X*2**	***X*1*X*2**	***X*3**	***X*1*X*3**	***X*2*X*3**	***X*1*X*2*X*3**	**Rep 1**	**Rep 2**
+1	-1	-1	+1	-1	+1	+1	-1	-3	-1
+1	+1	-1	-1	-1	-1	+1	+1	0	-1
+1	-1	+1	-1	-1	+1	-1	+1	-1	0
+1	+1	+1	+1	-1	-1	-1	-1	+2	+3
+1	-1	-1	+1	+1	-1	-1	+1	-1	0
+1	+1	-1	-1	+1	+1	-1	-1	+2	+1
+1	-1	+1	-1	+1	-1	+1	-1	+1	+1
+1	+1	+1	+1	+1	+1	+1	+1	+6	+5

The block with the 1's and -1's is called the *Model Matrix* or the *Analysis Matrix.* The table formed by the columns X1, X2 and X3 is called the *Design Table* or *Design Matrix.*

Orthogonality Properties of Analysis Matrices for 2-Factor Experiments

Eliminate correlation between estimates of main effects and interactions

When all factors have been coded so that the high value is "1" and the low value is "-1", the design matrix for any full (or suitably chosen fractional) factorial experiment has columns that are all pairwise orthogonal and all the columns (except the "I" column) sum to 0.

The orthogonality property is important because it eliminates correlation between the estimates of the main effects and interactions.

5.3.3.3.2. Full factorial example

A Full Factorial Design Example

An example of a full factorial design with 3 factors

The following is an example of a full factorial design with 3 factors that also illustrates replication, randomization, and added center points.

Suppose that we wish to improve the yield of a polishing operation. The three inputs (factors) that are considered important to the operation are Speed (**X1**), Feed (**X2**), and Depth **(X3**). We want to ascertain the relative importance of each of these factors on Yield (**Y**).

Speed, Feed and Depth can all be varied continuously along their respective scales, from a low to a high setting. Yield is observed to vary smoothly when progressive changes are made to the inputs. This leads us to believe that the ultimate response surface for **Y** will be smooth.

Table of factor level settings

TABLE 3.5 High (+1), Low (-1), and Standard (0) Settings for a Polishing Operation

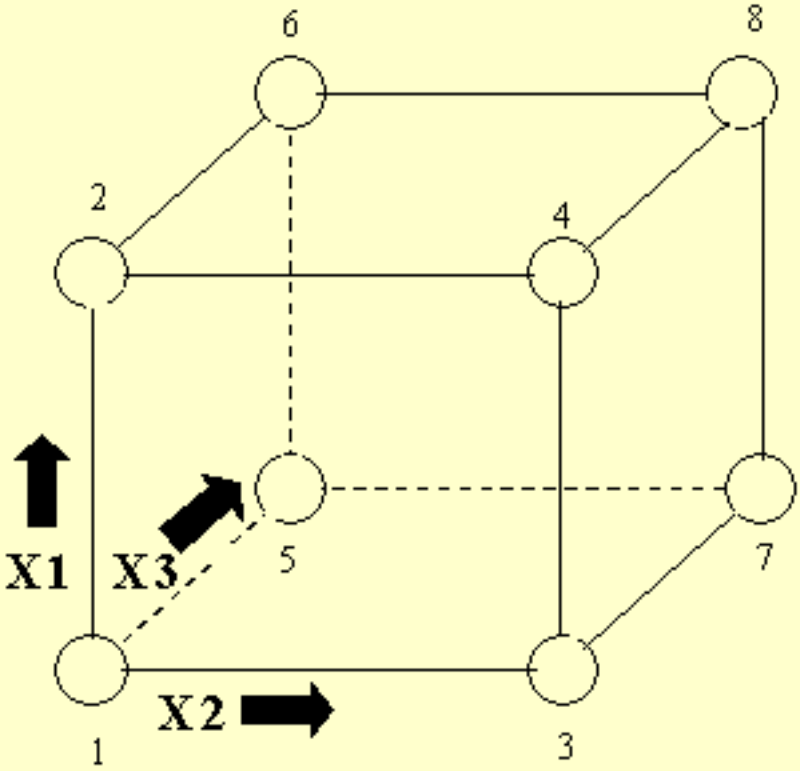
	Low (-1)	Standard (0)	High (+1)	Units
Speed	16	20	24	rpm
Feed	0.001	0.003	0.005	cm/sec
Depth	0.01	0.015	0.02	cm/sec

Factor Combinations

Graphical representation of the factor level settings

We want to try various combinations of these settings so as to establish the best way to run the polisher. There are eight different ways of combining high and low settings of Speed, Feed, and Depth. These eight are shown at the corners of the following diagram.

FIGURE 3.2 A 2^3 Two-level, Full Factorial Design; Factors X1, X2, X3. (The arrows show the direction of increase of the factors.)



2^3 implies 8 runs

Note that if we have k factors, each run at two levels, there will be 2^k different combinations of the levels. In the present case, $k = 3$ and $2^3 = 8$.

Full Model

Running the full complement of all possible factor combinations means that we can estimate all the main and interaction effects. There are three main effects, three two-factor interactions, and a three-factor interaction, all of which appear in the full model as follows:

$$\begin{aligned} Y = {} & \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \\ & \beta_{12} X_1 * X_2 + \beta_{23} * X_2 * X_3 + \beta_{13} X_1 * X_3 + \\ & \beta_{123} X_1 * X_2 * X_3 \end{aligned}$$

A full factorial design allows us to estimate all eight `beta' coefficients $\{\beta_0, \ldots, \beta_{123}\}$.

Standard order

Coded variables in standard order

The numbering of the corners of the box in the last figure refers to a standard way of writing down the settings of an experiment called `standard order'. We see standard order displayed in the following tabular representation of the eight-cornered box. Note that the factor settings have been coded, replacing the low setting by -1 and the high setting by 1.

Factor settings in tabular form

TABLE 3.6 A 2^3 Two-level, Full Factorial Design Table Showing Runs in `Standard Order'

	*X*1	*X*2	*X*3
1	-1	-1	-1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	-1
5	-1	-1	+1
6	+1	-1	+1
7	-1	+1	+1
8	+1	+1	+1

Replication

Replication provides information on variability

Running the entire design more than once makes for easier data analysis because, for each run (i.e., `corner of the design box') we obtain an average value of the response as well as some idea about the dispersion (variability, consistency) of the response at that setting.

Homogeneity of variance

One of the usual analysis assumptions is that the response dispersion is uniform across the experimental space. The technical term is `homogeneity of variance'. Replication allows us to check this assumption and possibly find the setting combinations that give inconsistent yields, allowing us to avoid that area of the factor space.

Factor settings in standard order with replication

We now have constructed a design table for a two-level full factorial in three factors, replicated twice.

TABLE 3.7 The 2^3 Full Factorial Replicated Twice and Presented in Standard Order

	Speed, *X*1	Feed, *X*2	Depth, *X*3
1	16, -1	.001, -1	.01, -1
2	24, +1	.001, -1	.01, -1
3	16, -1	.005, +1	.01, -1
4	24, +1	.005, +1	.01, -1
5	16, -1	.001, -1	.02, +1
6	24, +1	.001, -1	.02, +1
7	16, -1	.005, +1	.02, +1
8	24, +1	.005, +1	.02, +1
9	16, -1	.001, -1	.01, -1
10	24, +1	.001, -1	.01, -1
11	16, -1	.005, +1	.01, -1
12	24, +1	.005, +1	.01, -1
13	16, -1	.001, -1	.02, +1
14	24, +1	.001, -1	.02, +1
15	16, -1	.005, +1	.02, +1
16	24, +1	.005, +1	.02, +1

Randomization

No randomization and no center points

If we now ran the design as is, in the order shown, we would have two deficiencies, namely:

1. no randomization, and
2. no center points.

Randomization provides protection against extraneous factors affecting the results

The more freely one can randomize experimental runs, the more insurance one has against extraneous factors possibly affecting the results, and hence perhaps wasting our experimental time and effort. For example, consider the `Depth' column: the settings of Depth, in standard order, follow a `four low, four high, four low, four high' pattern.

Suppose now that four settings are run in the day and four at night, and that (unknown to the experimenter) ambient temperature in the polishing shop affects Yield. We would run the experiment over two days and two nights and conclude that Depth influenced Yield, when in fact ambient temperature was the significant influence. So the moral is: Randomize experimental runs as much as possible.

Table of factor settings in randomized order

Here's the design matrix again with the rows randomized (using the RAND function of EXCEL). The old standard order column is also shown for comparison and for re-sorting, if desired, after the runs are in.

TABLE 3.8 The 2^3 Full Factorial Replicated Twice with Random Run Order Indicated

Random Order	Standard Order	*X*1	*X*2	*X*3
1	5	-1	-1	+1
2	15	-1	+1	+1
3	9	-1	-1	-1
4	7	-1	+1	+1
5	3	-1	+1	-1
6	12	+1	+1	-1
7	6	+1	-1	+1
8	4	+1	+1	-1
9	2	+1	-1	-1
10	13	-1	-1	+1
11	8	+1	+1	+1
12	16	+1	+1	+1
13	1	-1	-1	-1
14	14	+1	-1	+1
15	11	-1	+1	-1
16	10	+1	-1	-1

Table showing design matrix with randomization and center points

This design would be improved by adding at least 3 centerpoint runs placed at the beginning, middle and end of the experiment. The final design matrix is shown below:

TABLE 3.9 The 2^3 Full Factorial Replicated Twice with Random Run Order Indicated and Center Point Runs Added

Random Order	Standard Order	*X*1	*X*2	*X*3
1		0	0	0
2	5	-1	-1	+1
3	15	-1	+1	+1
4	9	-1	-1	-1
5	7	-1	+1	+1
6	3	-1	+1	-1
7	12	+1	+1	-1
8	6	+1	-1	+1
9		0	0	0
10	4	+1	+1	-1
11	2	+1	-1	-1
12	13	-1	-1	+1
13	8	+1	+1	+1
14	16	+1	+1	+1
15	1	-1	-1	-1
16	14	+1	-1	+1
17	11	-1	+1	-1
18	10	+1	-1	-1
19		0	0	0

5.3.3.3.3. Blocking of full factorial designs

Eliminate the influence of extraneous factors by "blocking"

We often need to eliminate the influence of extraneous factors when running an experiment. We do this by "blocking".

Previously, blocking was introduced when randomized block designs were discussed. There we were concerned with one factor in the presence of one of more nuisance factors. In this section we look at a general approach that enables us to divide 2-level factorial experiments into blocks.

For example, assume we anticipate predictable shifts will occur while an experiment is being run. This might happen when one has to change to a new batch of raw materials halfway through the experiment. The effect of the change in raw materials is well known, and we want to eliminate its influence on the subsequent data analysis.

Blocking in a 2^3 factorial design

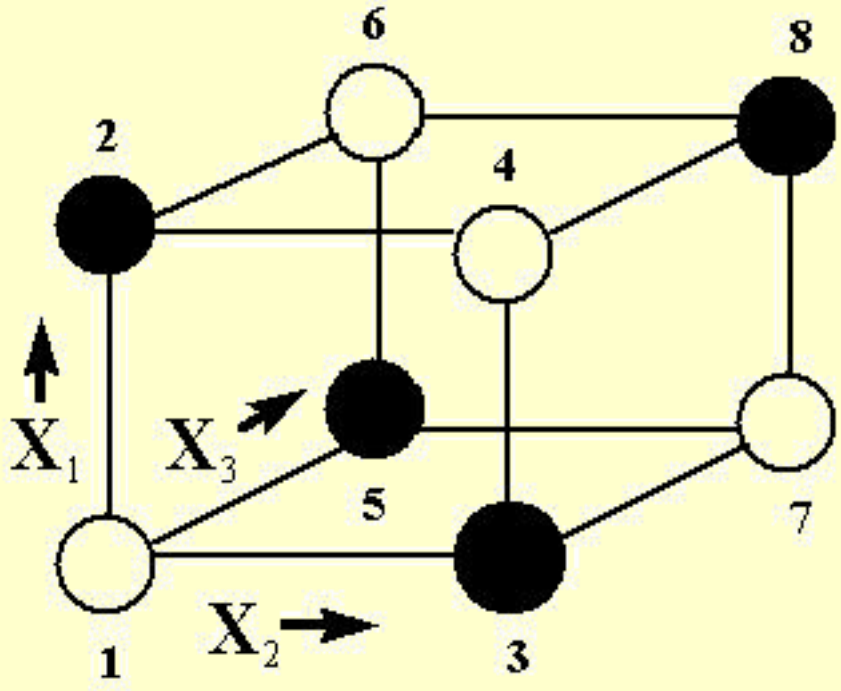
In this case, we need to divide our experiment into two halves (2 *blocks*), one with the first raw material batch and the other with the new batch. The division has to balance out the effect of the materials change in such a way as to eliminate its influence on the analysis, and we do this by *blocking.*

Example

Example: An eight-run 2^3 full factorial has to be blocked into two groups of four runs each. Consider the design `box' for the 2^3 full factorial. Blocking can be achieved by assigning the first block to the dark-shaded corners and the second block to the open circle corners.

Graphical representation of blocking scheme

FIGURE 3.3 Blocking Scheme for a 2^3 Using Alternate Corners



Three-factor interaction confounded with the block effect

This works because we are in fact assigning the `estimation' of the (unwanted) blocking effect to the three-factor interaction, and because of the special property of two-level designs called orthogonality. That is, the three-factor interaction is "confounded" with the block effect as will be seen shortly.

Orthogonality

Orthogonality guarantees that we can always estimate the effect of one factor or interaction clear of any influence due to any other factor or interaction. Orthogonality is a very desirable property in DOE and this is a major reason why two-level factorials are so popular and successful.

Table showing blocking scheme

Formally, consider the 2^3 design table with the three-factor interaction column added.

TABLE 3.10 Two Blocks for a 2^3 Design

SPEED *X1*	FEED *X2*	DEPTH *X3*	*X1*X2*X3*	BLOCK
-1	-1	-1	-1	**I**
+1	-1	-1	+1	**II**
-1	+1	-1	+1	**II**
+1	+1	-1	-1	**I**
-1	-1	+1	+1	**II**
+1	-1	+1	-1	**I**
-1	+1	+1	-1	**I**
+1	+1	+1	+1	**II**

Block by assigning the "Block effect" to a high-order interaction

Rows that have a `-1' in the three-factor interaction column are assigned to `Block I' (rows 1, 4, 6, 7), while the other rows are assigned to `Block II' (rows 2, 3, 5, 8). Note that the Block I rows are the open circle corners of the design `box' above; Block II are dark-shaded corners.

Most DOE software will do blocking for you

The general rule for blocking is: use one or a combination of high-order interaction columns to construct blocks. This gives us a formal way of blocking complex designs. Apart from simple cases in which you can design your own blocks, your statistical/DOE software will do the blocking if asked, but you do need to understand the principle behind it.

Block effects are confounded with higher-order interactions

The price you pay for blocking by using high-order interaction columns is that you can no longer distinguish the high-order interaction(s) from the blocking effect - they have been `confounded,' or `aliased.' In fact, the blocking effect is now the sum of the blocking effect and the high-order interaction effect. This is fine as long as our assumption about negligible high-order interactions holds true, which it usually does.

Center points within a block

Within a block, center point runs are assigned as if the block were a separate experiment - which in a sense it is. Randomization takes place within a block as it would for any non-blocked DOE.

5.3.3.4. Fractional factorial designs

Full factorial experiments can require many runs

The ASQC (1983) Glossary & Tables for Statistical Quality Control defines fractional factorial design in the following way: "*A factorial experiment in which only an adequately chosen fraction of the treatment combinations required for the complete factorial experiment is selected to be run.*"

A carefully chosen fraction of the runs may be all that is necessary

Even if the number of factors, k, in a design is small, the 2^k runs specified for a full factorial can quickly become very large. For example, $2^6 = 64$ runs is for a two-level, full factorial design with six factors. To this design we need to add a good number of centerpoint runs and we can thus quickly run up a very large resource requirement for runs with only a modest number of factors.

Later sections will show how to choose the "right" fraction for 2-level designs - these are both balanced and orthogonal

The solution to this problem is to use only a fraction of the runs specified by the full factorial design. Which runs to make and which to leave out is the subject of interest here. In general, we pick a fraction such as ½, ¼, etc. of the runs called for by the full factorial. We use various strategies that ensure an appropriate choice of runs. The following sections will show you how to choose an appropriate fraction of a full factorial design to suit your purpose at hand. *Properly chosen fractional factorial designs for 2-level experiments have the desirable properties of being both balanced and orthogonal.*

2-Level fractional factorial designs emphasized

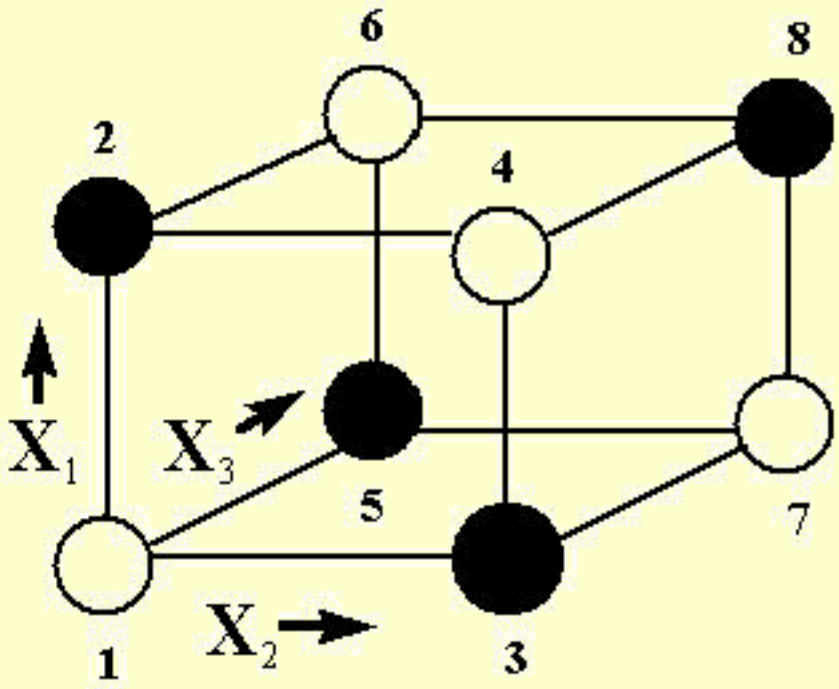
Note: We will be emphasizing fractions of two-level designs only. This is because two-level fractional designs are, in engineering at least, by far the most popular fractional designs. Fractional factorials where some factors have three levels will be covered briefly in Section 5.3.3.10.

5.3.3.4.1. A 2^{3-1} design (half of a 2^3)

We can run a fraction of a full factorial experiment and still be able to estimate main effects

Consider the two-level, full factorial design for three factors, namely the 2^3 design. This implies eight runs (not counting replications or center points). Graphically, as shown earlier, we can represent the 2^3 design by the following cube:

FIGURE 3.4 A 2^3 Full Factorial Design; Factors X_1, X_2, X_3. (The arrows show the direction of increase of the factors. Numbers `1' through `8' at the corners of the design cube reference the `Standard Order' of runs)



Tabular representation of the design

In tabular form, this design (also showing eight observations `y_j' (j = 1,...,8) is given by

TABLE 3.11 A 2^3 Two-level, Full Factorial Design Table Showing Runs in `Standard Order,' Plus Observations (y_j)

	$X1$	$X2$	$X3$	Y
1	-1	-1	-1	$y_1 = 33$
2	+1	-1	-1	$y_2 = 63$
3	-1	+1	-1	$y_3 = 41$
4	+1	+1	-1	$Y_4 = 57$
5	-1	-1	+1	$y_5 = 57$
6	+1	-1	+1	$y_6 = 51$
7	-1	+1	+1	$y_7 = 59$
8	+1	+1	+1	$y_8 = 53$

Responses in standard order

The right-most column of the table lists `$y_1$' through `y_8' to indicate the responses measured for the experimental runs when listed in standard order. For example, `$y_1$' is the response (i.e., output) observed when the three factors were all run at their `low' setting. The numbers entered in the "y" column will be used to illustrate calculations of effects.

Computing X1 main effect

From the entries in the table we are able to compute all `effects' such as main effects, first-order `interaction' effects, etc. For example, to compute the main effect estimate `$c_1$' of factor $X_1$, we compute the average response at all runs with $X_1$ at the `high' setting, namely $(1/4)(y_2 + y_4 + y_6 + y_8)$, minus the average response of all runs with X_1 set at `low,' namely $(1/4)(y_1 + y_3 + y_5 + y_7)$. That is,

$$c_1 = (1/4)\,(y_2 + y_4 + y_6 + y_8) - (1/4)(y_1 + y_3 + y_5 + y_7) \text{ or}$$
$$c_1 = (1/4)(63+57+51+53\,) - (1/4)(33+41+57+59) = 8.5$$

Can we estimate X1 main effect with four runs?

Suppose, however, that we only have enough resources to do four runs. Is it still possible to estimate the main effect for X_1? Or any other main effect? The answer is yes, and there are even different choices of the four runs that will accomplish this.

Example of computing the main effects using only four runs

For example, suppose we select only the four light (unshaded) corners of the design cube. Using these four runs (1, 4, 6 and 7), we can still compute c_1 as follows:

$$c_1 = (1/2)\,(y_4 + y_6) - (1/2)\,(y_1 + y_7) \text{ or}$$
$$c_1 = (1/2)\,(57+51) - (1/2)\,(33+59) = 8.$$

Simarly, we would compute c_2, the effect due to X_2, as

$$c_2 = (1/2)\,(y_4 + y_7) - (1/2)\,(y_1 + y_6) \text{ or}$$
$$c_2 = (1/2)\,(57+59) - (1/2)\,(33+51) = 16.$$

Finally, the computation of c_3 for the effect due to X_3 would be

$$c_3 = (1/2)\,(y_6 + y_7) - (1/2)\,(y_1 + y_4) \text{ or}$$
$$c_3 = (1/2)\,(51+59) - (1/2)\,(33+57) = 10.$$

Alternative runs for computing main effects

We could also have used the four dark (shaded) corners of the design cube for our runs and obtained similiar, but slightly different, estimates for the main effects. In either case, we would have used half the number of runs that the full factorial requires. *The half fraction we used is a new design written as* 2^{3-1}. Note that $2^{3-1} = 2^3/2 = 2^2 = 4$, which is the number of runs in this half-fraction design. In the next section, a general method for choosing fractions that "work" will be discussed.

Example of how fractional factorial experiments often arise in industry

Example: An engineering experiment calls for running three factors, namely Pressure, Table speed, and Down force, each at a `high' and a `low' setting, on a production tool to determine which has the greatest effect on product uniformity. Interaction effects are considered negligible, but uniformity measurement error requires that at least two separate runs (replications) be made at each process setting. In addition, several `standard setting' runs (centerpoint runs) need to be made at regular intervals during the experiment to monitor for process drift. As experimental time and material are limited, no more than 15 runs can be planned.

A full factorial 2^3 design, replicated twice, calls for 8x2 = 16 runs, even without centerpoint runs, so this is not an option. However a 2^{3-1} design replicated twice requires only 4x2 = 8 runs, and then we would have 15-8 = 7 spare runs: 3 to 5 of these spare runs can be used for centerpoint runs and the rest saved for backup in case something goes wrong with any run. As long as we are confident that the interactions are negligbly small (compared to the main effects), and as long as complete replication is required, then the above replicated 2^{3-1} fractional factorial design (with center points) is a very reasonable

choice.

On the other hand, if interactions are potentially large (and if the replication required could be set aside), then the usual 2^3 full factorial design (with center points) would serve as a good design.

5.3.3.4.2. Constructing the 2^{3-1} half-fraction design

Construction of a 2^{3-1} half fraction design by staring with a 2^2 full factorial design

First note that, mathematically, $2^{3-1} = 2^2$. This gives us the first step, which is to start with a regular 2^2 full factorial design. That is, we start with the following design table.

TABLE 3.12 A Standard Order 2^2 Full Factorial Design Table

	*X*1	*X*2
1	-1	-1
2	+1	-1
3	-1	+1
4	+1	+1

Assign the third factor to the interaction column of a 2^2 design

This design has four runs, the right number for a half-fraction of a 2^3, but there is no column for factor *X*3. We need to add a third column to take care of this, and we do it by adding the *X*1**X*2 interaction column. This column is, as you will recall from full factorial designs, constructed by multiplying the row entry for *X*1 with that of *X*2 to obtain the row entry for *X*1**X*2.

TABLE 3.13 A 2^2 Design Table Augmented with the X1*X2 Interaction Column `X1*X2'

	*X*1	*X*2	*X*1**X*2
1	-1	-1	+1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	+1

*Design table with X3 set to X1*X2*

We may now substitute `X3' in place of `X1*X2' in this table.

TABLE 3.15 A 2^{3-1} Design Table with Column X3 set to X1*X2

	*X*1	*X*2	*X*3
1	-1	-1	+1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	+1

*Design table with X3 set to -X1*X2*

Note that the rows of Table 3.14 give the dark-shaded corners of the design in Figure 3.4. If we had set X3 = -X1*X2 as the rule for generating the third column of our 2^{3-1} design, we would have obtained:

TABLE 3.15 A 2^{3-1} Design Table with Column X3 set to - X1*X2

	*X*1	*X*2	*X*3
1	-1	-1	-1
2	+1	-1	+1
3	-1	+1	+1
4	+1	+1	-1

Main effect estimates from fractional factorial not as good as full factorial

This design gives the light-shaded corners of the box of Figure 3.4. Both 2^{3-1} designs that we have generated are equally good, and both save half the number of runs over the original 2^3 full factorial design. If c_1, c_2, and c_3 are our estimates of the main effects for the factors $X1$, $X2$, $X3$ (i.e., the difference in the response due to going from "low" to "high" for an effect), then the precision of the estimates c_1, c_2, and c_3 are not quite as good as for the full 8-run factorial because we only have four observations to construct the averages instead of eight; this is one price we have to pay for using fewer runs.

Example

Example: For the `Pressure (P), Table speed (T), and Down force (D)' design situation of the previous example, here's a replicated $2^{3-1}$ in randomized run order, with five centerpoint runs (`000') interspersed among the runs. This design table was constructed using the technique discussed above, with D = P*T.

Design table for the example

TABLE 3.16 A 2^{3-1} Design Replicated Twice, with Five Centerpoint Runs Added

	Pattern	P	T	D	Center Point
1	000	0	0	0	1
2	+--	+1	-1	-1	0
3	-+-	-1	+1	-1	0
4	000	0	0	0	1
5	+++	+1	+1	+1	0
6	--+	-1	-1	+1	0
7	000	0	0	0	1
8	+--	+1	-1	-1	0
9	--+	-1	-1	+1	0
10	000	0	0	0	1
11	+++	+1	+1	+1	0
12	-+-	-1	+1	-1	0
13	000	0	0	0	1

5.3.3.4.3. Confounding (also called aliasing)

Confounding means we have lost the ability to estimate some effects and/or interactions

One price we pay for using the design table column $X1*X2$ to obtain column $X3$ in Table 3.14 is, clearly, our inability to obtain an estimate of the interaction effect for $X1*X2$ (i.e., c_{12}) that is separate from an estimate of the main effect for $X3$. In other words, we have confounded the main effect estimate for factor $X3$ (i.e., c_3) with the estimate of the interaction effect for $X1$ and $X2$ (i.e., with c_{12}). The whole issue of confounding is fundamental to the construction of fractional factorial designs, and we will spend time discussing it below.

Sparsity of effects assumption

In using the 2^{3-1} design, we also assume that c_{12} is small compared to c_3; this is called a `*sparsity of effects*' assumption. Our computation of c_3 is in fact a computation of $c_3 + c_{12}$. If the desired effects are only confounded with non-significant interactions, then we are OK.

A Notation and Method for Generating Confounding or Aliasing

A short way of writing factor column multiplication

A short way of writing `$X3 = X1*X2$' (understanding that we are talking about multiplying columns of the design table together) is: `3 = 12' (similarly 3 = -12 refers to $X3 = -X1*X2$). Note that `12' refers to column multiplication of the kind we are using to construct the fractional design and any column multiplied by itself gives the identity column of all 1's.

Next we multiply both sides of 3=12 by 3 and obtain 33=123, or I=123 since 33=I (or a column of all 1's). Playing around with this "algebra", we see that 2I=2123, or 2=2123, or 2=1223, or 2=13 (since 2I=2, 22=I, and 1I3=13). Similarly, 1=23.

Definition of "design generator" or "generating relation" and "defining relation"

I=123 is called a *design generator* or a *generating relation* for this 2^{3-1}design (the dark-shaded corners of Figure 3.4). Since there is only one design generator for this design, it is also the *defining relation* for the design. Equally, I=-123 is the design generator (and defining relation) for the light-shaded corners of Figure 3.4. *We call I=123 the defining relation for the 2^{3-1} design because with it we can generate (by "multiplication") the complete confounding pattern for the design*. That is, given I=123, we can generate the set of {1=23, 2=13, 3=12, I=123}, which is the complete set of *aliases*, as they are called, for this 2^{3-1} fractional factorial design. With I=123, we can easily generate all the columns of the half-fraction design 2^{3-1}.

Principal fraction

Note: We can replace any design generator by its negative counterpart and have an equivalent, but different fractional design. The fraction generated by positive design generators is sometimes called the *principal fraction*.

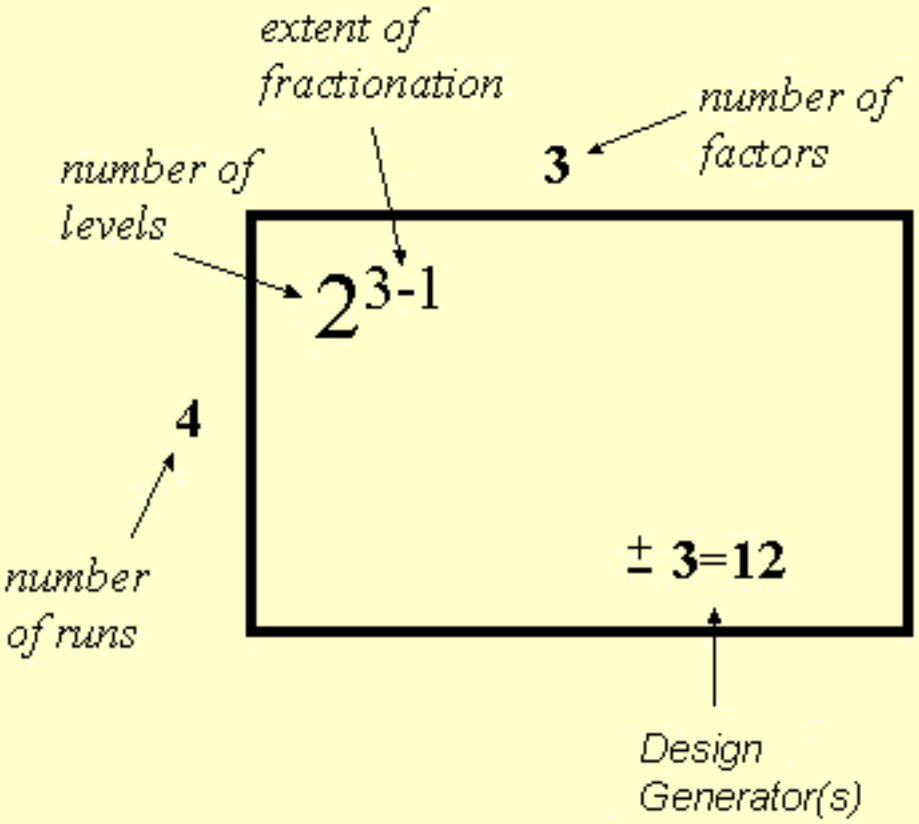
All main effects of 2^{3-1} design confounded with two-factor interactions

The confounding pattern described by 1=23, 2=13, and 3=12 tells us that all the main effects of the 2^{3-1} design are confounded with two-factor interactions. That is the price we pay for using this fractional design. Other fractional designs have different confounding patterns; for example, in the typical quarter-fraction of a 2^6 design, i.e., in a 2^{6-2} design, main effects are confounded with three-factor interactions (e.g., 5=123) and so on. In the case of 5=123, we can also readily see that 15=23 (etc.), which alerts us to the fact that certain two-factor interactions of a 2^{6-2} are confounded with other two-factor interactions.

A useful summary diagram for a fractional factorial design

Summary: A convenient summary diagram of the discussion so far about the 2^{3-1} design is as follows:

FIGURE 3.5 Essential Elements of a 2^{3-1} Design



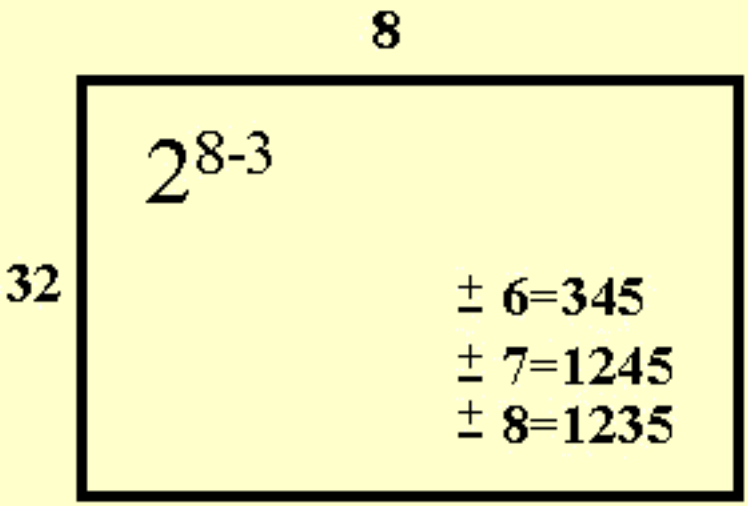
The next section will add one more item to the above box, and then we will be able to select the right two-level fractional factorial design for a wide range of experimental tasks.

5.3.3.4.4. Fractional factorial design specifications and design resolution

Generating relation and diagram for the 2^{8-3} fractional factorial design

We considered the 2^{3-1} design in the previous section and saw that its generator written in "I = ... " form is {I = +123}. Next we look at a one-eighth fraction of a 2^8 design, namely the 2^{8-3} fractional factorial design. Using a diagram similar to Figure 3.5, we have the following:

FIGURE 3.6 Specifications for a 2^{8-3} Design



2^{8-3} design has 32 runs

Figure 3.6 tells us that a 2^{8-3} design has 32 runs, not including centerpoint runs, and eight factors. There are three generators since this is a $1/8 = 2^{-3}$ fraction (in general, a 2^{k-p} fractional factorial needs p generators which define the settings for p additional factor columns to be added to the 2^{k-p} full factorial design columns - see the following detailed description for the 2^{8-3} design).

How to Construct a Fractional Factorial Design From the Specification

Rule for constructing a fractional factorial design

In order to construct the design, we do the following:

1. Write down a full factorial design in standard order for *k-p* factors (8-3 = 5 factors for the example above). In the specification above we start with a 2^5 full factorial design. Such a design has $2^5 = 32$ rows.
2. Add a sixth column to the design table for factor 6, using 6 = 345 (or 6 = -345) to manufacture it (i.e., create the new column by multiplying the indicated old columns together).
3. Do likewise for factor 7 and for factor 8, using the appropriate design generators given in Figure 3.6.
4. The resultant design matrix gives the 32 trial runs for an 8-factor fractional factorial design. (When actually running the experiment, we would of course randomize the run order.

Design generators

We note further that the design generators, written in `I = ...' form, for the principal 2^{8-3} fractional factorial design are:

{ I = + 3456; I = + 12457; I = +12358 }.

These design generators result from multiplying the "6 = 345" generator by "6" to obtain "I = 3456" and so on for the other two generqators.

"Defining relation" for a fractional factorial design

The total collection of design generators for a factorial design, *including all new generators that can be formed as products of these generators*, is called a *defining relation*. There are seven "*words"*, or strings of numbers, in the defining relation for the 2^{8-3} design, starting with the original three generators and adding all the new "words" that can be formed by multiplying together any two or three of these original three words. These seven turn out to be I = 3456 = 12457 = 12358 = 12367 = 12468 = 3478 = 5678. In general, there will be $(2^p - 1)$ words in the defining relation for a 2^{k-p} fractional factorial.

Definition of "Resolution"

The length of the shortest word in the defining relation is called the resolution of the design. Resolution describes the degree to which estimated main effects are aliased (or confounded) with estimated 2-level interactions, 3-level interactions, etc.

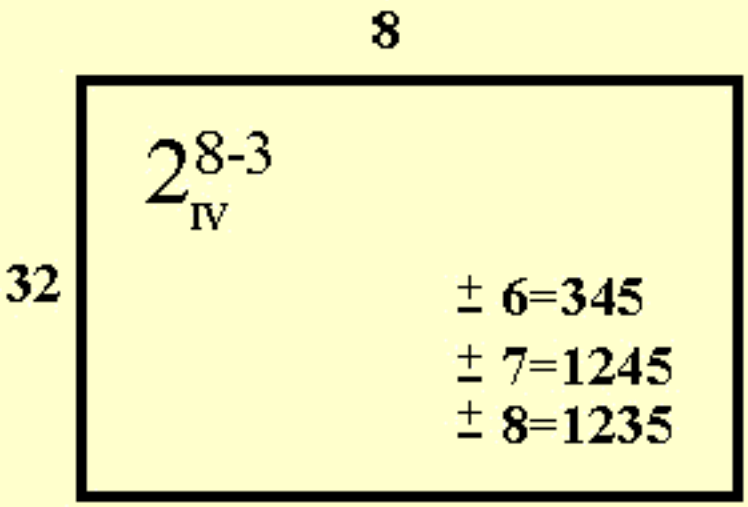
Notation for resolution (Roman numerals)

The length of the shortest word in the defining relation for the 2^{8-3} design is four. This is written in Roman numeral script, and subscripted as 2^{8-3}_{IV}. Note that the 2^{3-1} design has only one word, "I = 123" (or "I = -123"), in its defining relation since there is only one design generator, and so this fractional factorial design has resolution three; that is, we may write 2^{3-1}_{III}.

Diagram for a 2^{8-3} design showing resolution

Now Figure 3.6 may be completed by writing it as:

FIGURE 3.7 Specifications for a 2^{8-3}, Showing Resolution IV



Resolution and confounding

The design resolution tells us how badly the design is confounded. Previously, in the 2^{3-1} design, we saw that the main effects were confounded with two-factor interactions. However, main effects were not confounded with other main effects. So, at worst, we have 3=12, or 2=13, etc., but we do not have 1=2, etc. In fact, a resolution II design would be pretty useless for any purpose whatsoever!

Similarly, in a resolution IV design, main effects are confounded with at worst three-factor interactions. We can see, in Figure 3.7, that 6=345. We also see that 36=45, 34=56, etc. (i.e., some two-factor interactions are confounded with certain other two-factor interactions) etc.; but we never see anything like 2=13, or 5=34, (i.e., main effects confounded with two-factor interactions).

The complete first-order interaction confounding for the given 2^{8-3} design

The complete confounding pattern, for confounding of up to two-factor interactions, arising from the design given in Figure 3.7 is

34 = 56 = 78
35 = 46
36 = 45
37 = 48
38 = 47
57 = 68
58 = 67

All of these relations can be easily verified by multiplying the indicated two-factor interactions by the generators. For example, to verify that 38= 47, multiply both sides of 8=1235 by 3 to get 38=125. Then, multiply 7=1245 by 4 to get 47=125. From that it follows that 38=47.

One or two factors suspected of possibly having significant first-order interactions can be assigned in such a way as to avoid having them aliased

For this 2^{8-3}_{V} fractional factorial design, 15 two-factor interactions are aliased (confounded) in pairs or in a group of three. The remaining 28 - 15 = 13 two-factor interactions are only aliased with higher-order interactions (which are generally assumed to be negligible). This is verified by noting that factors "1" and "2" never appear in a length-4 word in the defining relation. So, all 13 interactions involving "1" and "2" are clear of aliasing with any other two factor interaction.

If one or two factors are suspected of possibly having significant first-order interactions, they can be assigned in such a way as to avoid having them aliased.

Higher resoulution designs have less severe confounding, but require more runs

A resolution IV design is "better" than a resolution III design because we have less-severe confounding pattern in the `IV' than in the `III' situation; higher-order interactions are less likely to be significant than low-order interactions.

A higher-resolution design for the same number of factors will, however, require more runs and so it is `worse' than a lower order design in that sense.

Resolution V designs for 8 factors

Similarly, with a resolution V design, main effects would be confounded with four-factor (and possibly higher-order) interactions, and two-factor interactions would be confounded with certain three-factor interactions. To obtain a resolution V design for 8 factors requires more runs than the 2^{8-3} design. One option, if estimating all main effects and two-factor interactions is a requirement, is a 2_V^{8-3} design. However, a 48-run alternative (John's 3/4 fractional factorial) is also available.

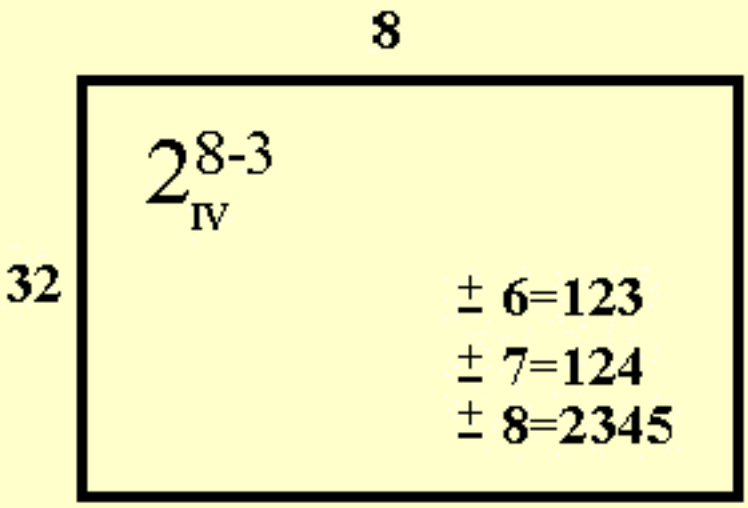
There are many choices of fractional factorial designs - some may have the same number of runs and resolution, but different aliasing patterns.

Note: There are other 2_V^{8-3} fractional designs that can be derived starting with different choices of design generators for the "6", "7" and "8" factor columns. However, they are either equivalent (in terms of the number of words of length of length of four) to the fraction with generators 6 = 345, 7 = 1245, 8 = 1235 (obtained by relabeling the factors), or they are inferior to the fraction given because their defining relation contains more words of length four (and therefore more confounded two-factor interactions). For example, the 2_V^{8-3} design with generators 6 = 12345, 7 = 135, and 8 = 245 has five length-four words in the defining relation (the defining relation is I = 123456 = 1357 = 2458 = 2467 = 1368 = 123478 = 5678). As a result, this design would confound more two factor-interactions (23 out of 28 possible two-factor interactions are confounded, leaving only "12", "14", "23", "27" and "34" as estimable two-factor interactions).

Diagram of an alternative way for generating the 2^{8-3} design

As an example of an equivalent "best" 2_V^{8-3} fractional factorial design, obtained by "relabeling", consider the design specified in Figure 3.8.

FIGURE 3.8 Another Way of Generating the 2^{8-3} Design



This design is equivalent to the design specified in Figure 3.7 after relabeling the factors as follows: 1 becomes 5, 2 becomes 8, 3 becomes 1, 4 becomes 2, 5 becomes 3, 6 remains 6, 7 becomes 4 and 8 becomes 7.

Minimum aberration

A table given later in this chapter gives a collection of useful fractional factorial designs that, for a given *k* and *p*, maximize the possible resolution and minimize the number of short words in the defining relation (which minimizes two-factor aliasing). The term for this is "minimum aberration".

Design Resolution Summary

Commonly used design Resolutions

The meaning of the most prevalent resolution levels is as follows:

Resolution III Designs

Main effects are confounded (aliased) with two-factor interactions.

Resolution IV Designs

No main effects are aliased with two-factor interactions, but two-factor interactions are aliased with each other.

Resolution V Designs

No main effect or two-factor interaction is aliased with any other main effect or two-factor interaction, but two-factor interactions are aliased with three-factor interactions.

5.3.3.4.5. Use of fractional factorial designs

Use low-resolution designs for screening among main effects and use higher-resolution designs when interaction effects and response surfaces need to be investigated

The basic purpose of a fractional factorial design is to economically investigate cause-and-effect relationships of significance in a given experimental setting. This does not differ in essence from the purpose of any experimental design. However, because we are able to choose fractions of a full design, and hence be more economical, we also have to be aware that different factorial designs serve different purposes.

Broadly speaking, with designs of resolution three, and sometimes four, we seek to screen out the few important main effects from the many less important others. For this reason, these designs are often termed main effects designs, or screening designs.

On the other hand, designs of resolution five, and higher, are used for focusing on more than just main effects in an experimental situation. These designs allow us to estimate interaction effects and such designs are easily augmented to complete a second-order design - a design that permits estimation of a full second-order (quadratic) model.

Different purposes for screening/RSM designs

Within the screening/RSM strategy of design, there are a number of functional purposes for which designs are used. For example, an experiment might be designed to determine how to make a product better or a process more robust against the influence of external and non-controllable influences such as the weather. Experiments might be designed to troubleshoot a process, to determine bottlenecks, or to specify which component(s) of a product are most in need of improvement. Experiments might also be designed to optimize yield, or to minimize defect levels, or to move a process away from an unstable operating zone. All these aims and purposes can be achieved using fractional factorial designs and their appropriate design enhancements.

5.3.3.4.6. Screening designs

Screening designs are an efficient way to identify significant main effects

The term `Screening Design' refers to an experimental plan that is intended to find the few significant factors from a list of many potential ones. Alternatively, we refer to a design as a screening design if its primary purpose is to identify significant main effects, rather than interaction effects, the latter being assumed an order of magnitude less important.

Use screening designs when you have many factors to consider

Even when the experimental goal is to eventually fit a response surface model (an RSM analysis), the first experiment should be a screening design when there are many factors to consider.

Screening designs are usually resolution III or IV

Screening designs are typically of resolution III. The reason is that resolution III designs permit one to explore the effects of many factors with an efficient number of runs.

Sometimes designs of resolution IV are also used for screening designs. In these designs, main effects are confounded with, at worst, three-factor interactions. This is better from the confounding viewpoint, but the designs require more runs than a resolution III design.

Plackett-Burman designs

Another common family of screening designs is the Plackett-Burman set of designs, so named after its inventors. These designs are of resolution III and will be described later.

Economical plans for determing significant main effects

In short, screening designs are economical experimental plans that focus on determining the relative significance of many main effects.

5.3.3.4.7. Summary tables of useful fractional factorial designs

Useful fractional factorial designs for up to 10 factors are summarized here

There are very useful summaries of two-level fractional factorial designs for up to 11 factors, originally published in the book *Statistics for Experimenters* by G.E.P. Box, W.G. Hunter, and J.S. Hunter (New York, John Wiley & Sons, 1978). and also given in the book *Design and Analysis of Experiments, 5th edition* by Douglas C. Montgomery (New York, John Wiley & Sons, 2000).

Generator column notation can use either numbers or letters for the factor columns

They differ in the notation for the design generators. Box, Hunter, and Hunter use numbers (as we did in our earlier discussion) and Montgomery uses capital letters according to the following scheme:

1	2	3	4	5	6	7	8	9	10	11
A	B	C	D	E	F	G	H	J	K	L

Notice the absence of the letter I. This is usually reserved for the intercept column that is identically 1. As an example of the letter notation, note that the design generator "6 = 12345" is equivalent to "F = ABCDE".

Details of the design generators, the defining relation, the confounding structure, and the design matrix

TABLE 3.17 catalogs these useful fractional factorial designs using the notation previously described in FIGURE 3.7.

Clicking on the 2^{k-p}_R specification for a given design provides details (courtesy of Dataplot files) of the design generators, the defining relation, the confounding structure (as far as main effects and two-level interactions are concerned), and the design matrix. The notation used follows our previous labeling of factors with numbers, not letters.

Click on the design specification in the table below and a text file with details about the design can be viewed or saved

TABLE 3.17 Summary of Useful Fractional Factorial Designs

Number of Factors, k	Design Specification	Number of Runs N
3	2_{III}^{3-1}	4
4	2_{IV}^{4-1}	8
5	2_{V}^{5-1}	16
5	2_{III}^{5-2}	8
6	2_{VI}^{6-1}	32
6	2_{IV}^{6-2}	16
6	2_{III}^{6-3}	8
7	2_{VII}^{7-1}	64
7	2_{IV}^{7-2}	32
7	2_{IV}^{7-3}	16
7	2_{III}^{7-4}	8
8	2_{VI}^{8-1}	128
8	2_{V}^{8-2}	64
8	2_{IV}^{8-3}	32
8	2_{IV}^{8-4}	16
9	2_{VI}^{9-2}	128
9	2_{IV}^{9-3}	64
9	2_{IV}^{9-4}	32

9	2_{III}^{9-5}	16
10	2_{V}^{10-3}	128
10	2_{IV}^{10-4}	64
10	2_{IV}^{10-5}	32
10	2_{III}^{10-6}	16
11	2_{V}^{11-4}	128
11	2_{IV}^{11-5}	64
11	2_{IV}^{11-6}	32
11	2_{III}^{11-7}	16
15	2_{III}^{15-11}	16
31	2_{III}^{31-26}	32

5.3.3.5. Plackett-Burman designs

Plackett-Burman designs

In 1946, R.L. Plackett and J.P. Burman published their now famous paper "The Design of Optimal Multifactorial Experiments" in *Biometrika* (vol. 33). This paper described the construction of very economical designs with the run number a multiple of four (rather than a power of 2). Plackett-Burman designs are very efficient screening designs when only main effects are of interest.

These designs have run numbers that are a multiple of 4

Plackett-Burman (PB) designs are used for screening experiments because, in a PB design, main effects are, in general, heavily confounded with two-factor interactions. The PB design in 12 runs, for example, may be used for an experiment containing up to 11 factors.

12-Run Plackett-Burnam design

TABLE 3.18 Plackett-Burman Design in 12 Runs for up to 11 Factors

	Pattern	*X*1	*X*2	*X*3	*X*4	*X*5	*X*6	*X*7	*X*8	*X*9	*X*10	*X*11
1	++++++++++	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
2	-+-+++---+-	-1	+1	-1	+1	+1	+1	-1	-1	-1	+1	-1
3	--+-+++---+	-1	-1	+1	-1	+1	+1	+1	-1	-1	-1	+1
4	+--+-+++---	+1	-1	-1	+1	-1	+1	+1	+1	-1	-1	-1
5	-+--+-+++--	-1	+1	-1	-1	+1	-1	+1	+1	+1	-1	-1
6	--+--+-+++-	-1	-1	+1	-1	-1	+1	-1	+1	+1	+1	-1
7	---+--+-+++	-1	-1	-1	+1	-1	-1	+1	-1	+1	+1	+1
8	+---+--+-++	+1	-1	-1	-1	+1	-1	-1	+1	-1	+1	+1
9	++---+--+-+	+1	+1	-1	-1	-1	+1	-1	-1	+1	-1	+1
10	+++---+--+-	+1	+1	+1	-1	-1	-1	+1	-1	-1	+1	-1
11	-+++---+--+	-1	+1	+1	+1	-1	-1	-1	+1	-1	-1	+1
12	+-+++---+--	+1	-1	+1	+1	+1	-1	-1	-1	+1	-1	-1

Saturated Main Effect designs

PB designs also exist for 20-run, 24-run, and 28-run (and higher) designs. With a 20-run design you can run a screening experiment for up to 19 factors, up to 23 factors in a 24-run design, and up to 27 factors in a 28-run design. These Resolution III designs are known as *Saturated Main Effect* designs because all degrees of freedom are utilized to estimate main effects. The designs for 20 and 24 runs are shown below.

20-Run Plackett-Burnam design

TABLE 3.19 A 20-Run Plackett-Burman Design

	*X*1	*X*2	*X*3	*X*4	*X*5	*X*6	*X*7	*X*8	*X*9	*X*10	*X*11	*X*12	*X*13	*X*14	*X*15	*X*16	*X*17	*X*18	*X*19
1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
2	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1
3	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1
4	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1
5	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1
6	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1
7	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1
8	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1
9	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1
10	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1
11	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1	+1
12	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1	-1
13	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1	+1
14	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1	+1
15	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1	+1
16	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1	+1
17	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1	-1
18	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1	-1
19	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1	+1
20	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1

24-Run Plackett-Burnam design

TABLE 3.20 A 24-Run Plackett-Burman Design

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22	X23
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1
3	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1
4	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1
5	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1
6	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1
7	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1
8	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1
9	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1
10	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1
11	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1
12	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1	1
13	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1	-1

14	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1	-1
15	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1	1
16	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1	1
17	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1	-1
18	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1	1
19	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1	-1
20	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1	1
21	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1	1
22	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1	1
23	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1	1
24	1	1	1	1	-1	1	-1	1	1	-1	-1	1	1	-1	-1	1	-1	1	-1	-1	-1	-1	-1

No defining relation

These designs do not have a defining relation since interactions are not identically equal to main effects. With the 2^{k-p}_{III} designs, a main effect column X_i is either orthogonal to X_iX_j or identical to plus or minus X_iX_j. For Plackett-Burman designs, the two-factor interaction column X_iX_j is correlated with every X_k (for k not equal to i or j).

Economical for detecting large main effects

However, these designs are very useful for economically detecting large main effects, assuming all interactions are negligible when compared with the few important main effects.

5.3.3.6. Response surface designs

Response surface models may involve just main effects and interactions or they may also have quadratic and possibly cubic terms to account for curvature

Earlier, we described the response surface method (RSM) objective. Under some circumstances, a model involving only main effects and interactions may be appropriate to describe a response surface when

1. Analysis of the results revealed no evidence of "pure quadratic" curvature in the response of interest (i.e., the response at the center approximately equals the average of the responses at the factorial runs).
2. The design matrix originally used included the limits of the factor settings available to run the process.

Equations for quadratic and cubic models

In other circumstances, a complete description of the process behavior might require a quadratic or cubic model:

Quadratic

$$\hat{y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3 + b_{11}x_1^2 + b_{22}x_2^2 + b_{33}x_3^2$$

Cubic

$$\hat{y} = \text{quadratic model} + b_{123}x_1x_2x_3 + b_{112}x_1^2x_2 + b_{113}x_1^2x_3 + b_{122}x_1x_2^2 + b_{133}x_1x_3^2 + b_{223}x_2^2x_3 + b_{233}x_2x_3^2 + b_{111}x_1^3 + b_{222}x_2^3 + b_{333}x_3^3$$

These are the full models, with all possible terms, rarely would all of the terms be needed in an application.

Quadratic models almost always sufficient for industrial applications

If the experimenter has defined factor limits appropriately and/or taken advantage of all the tools available in multiple regression analysis (transformations of responses and factors, for example), then finding an industrial process that requires a third-order model is highly unusual. Therefore, we will only focus on designs that are useful for fitting quadratic models. As we will see, these designs often provide lack of fit detection that will help determine when a higher-order model is needed.

General quadratic surface types

Figures 3.9 to 3.12 identify the general quadratic surface types that an investigator might encounter

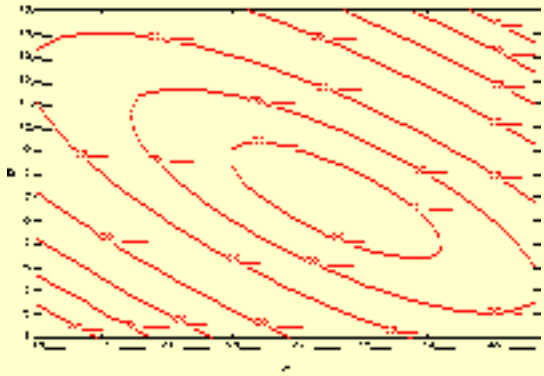


FIGURE 3.9 A Response Surface "Peak"

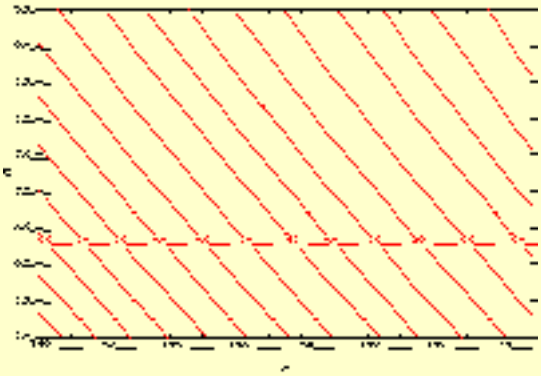


FIGURE 3.10 A Response Surface "Hillside"

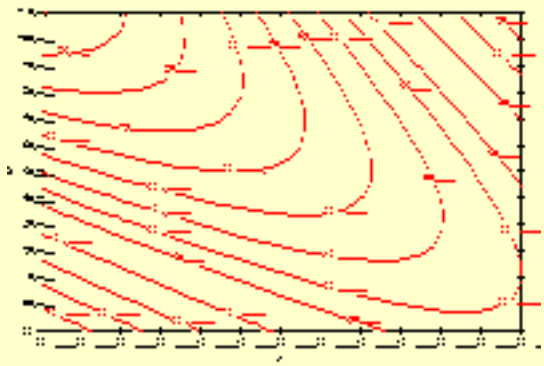


FIGURE 3.11 A Response Surface "Rising Ridge"

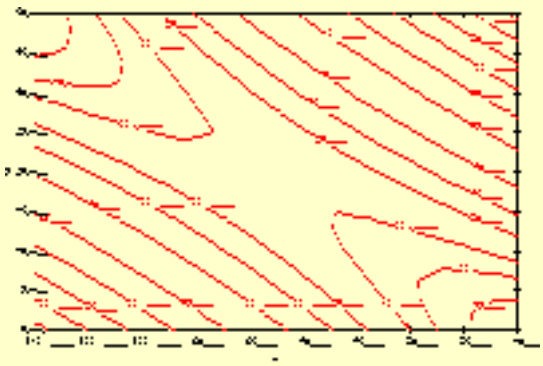


FIGURE 3.12 A Response Surface "Saddle"

Factor Levels for Higher-Order Designs

Possible behaviors of responses as functions of factor settings

Figures 3.13 through 3.15 illustrate possible behaviors of responses as functions of factor settings. In each case, assume the value of the response increases from the bottom of the figure to the top and that the factor settings increase from left to right.

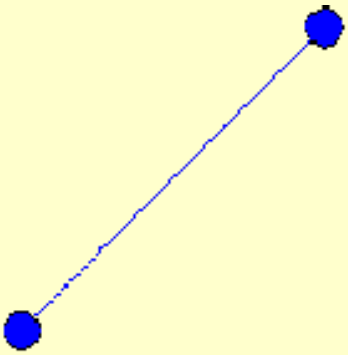


FIGURE 3.13
Linear Function

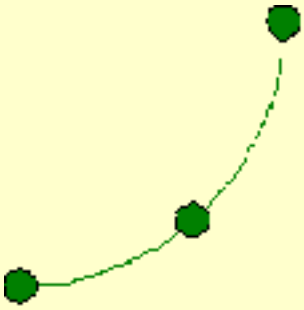


FIGURE 3.14
Quadratic Function

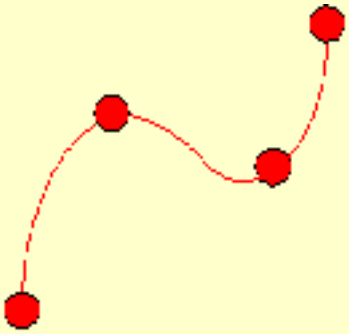


FIGURE 3.15
Cubic Function

A two-level experiment with center points can detect, but not fit, quadratic effects

If a response behaves as in Figure 3.13, the design matrix to quantify that behavior need only contain factors with two levels -- low and high. This model is a basic assumption of simple two-level factorial and fractional factorial designs. If a response behaves as in Figure 3.14, the minimum number of levels required for a factor to quantify that behavior is three. One might logically assume that adding center points to a two-level design would satisfy that requirement, but the arrangement of the treatments in such a matrix confounds all quadratic effects with each other. *While a two-level design with center points cannot estimate individual pure quadratic effects, it can detect them effectively*.

Three-level factorial design

A solution to creating a design matrix that permits the estimation of simple curvature as shown in Figure 3.14 would be to use a three-level factorial design. Table 3.21 explores that possibility.

Four-level factorial design

Finally, in more complex cases such as illustrated in Figure 3.15, the design matrix must contain at least four levels of each factor to characterize the behavior of the response adequately.

3-level factorial designs can fit quadratic models but they require many runs when there are more than 4 factors

TABLE 3.21 Three-level Factorial Designs

Number of Factors	Treatment Combinations 3^k Factorial	Number of Coefficients Quadratic Empirical Model
2	9	6
3	27	10
4	81	15
5	243	21
6	729	28

Fractional factorial designs created to avoid such a large number of runs

Two-level factorial designs quickly become too large for practical application as the number of factors investigated increases. This problem was the motivation for creating `fractional factorial' designs. Table 3.21 shows that the number of runs required for a 3^k factorial becomes unacceptable even more quickly than for 2^k designs. The last column in Table 3.21 shows the number of terms present in a quadratic model for each case.

Number of runs large even for modest number of factors

With only a modest number of factors, the number of runs is very large, even an order of magnitude greater than the number of parameters to be estimated when k isn't small. For example, the absolute minimum number of runs required to estimate all the terms present in a four-factor quadratic model is 15: the intercept term, 4 main effects, 6 two-factor interactions, and 4 quadratic terms.

The corresponding 3^k design for $k = 4$ requires 81 runs.

Complex alias structure and lack of rotatability for 3-level fractional factorial designs

Considering a fractional factorial at three levels is a logical step, given the success of fractional designs when applied to two-level designs. Unfortunately, the alias structure for the three-level fractional factorial designs is considerably more complex and harder to define than in the two-level case.

Additionally, the three-level factorial designs suffer a major flaw in their lack of `*rotatability*.'

Rotatability of Designs

"Rotatability" is a desirable property not present in 3-level factorial designs

In a rotatable design, the variance of the predicted values of *y* is a function of the distance of a point from the center of the design and is not a function of the direction the point lies from the center. Before a study begins, little or no knowledge may exist about the region that contains the optimum response. Therefore, the experimental design matrix should not bias an investigation in any direction.

Contours of variance of predicted values are concentric circles

In a rotatable design, the contours associated with the variance of the predicted values are concentric circles. Figures 3.16 and 3.17 (adapted from Box and Draper, `Empirical Model Building and Response Surfaces,' page 485) illustrate a three-dimensional plot and contour plot, respectively, of the `information function' associated with a 3^2 design.

Information function

The information function is:

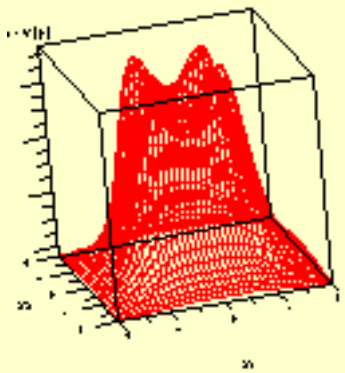
$$\frac{1}{V(\hat{y})}$$

with V denoting the variance (of the predicted value $\hat{y}$).

Each figure clearly shows that the information content of the design is not only a function of the distance from the center of the design space, but also a function of direction.

Graphs of the information function for a rotatable quadratic design

Figures 3.18 and 3.19 are the corresponding graphs of the information function for a rotatable quadratic design. In each of these figures, the value of the information function depends only on the distance of a point from the center of the space.



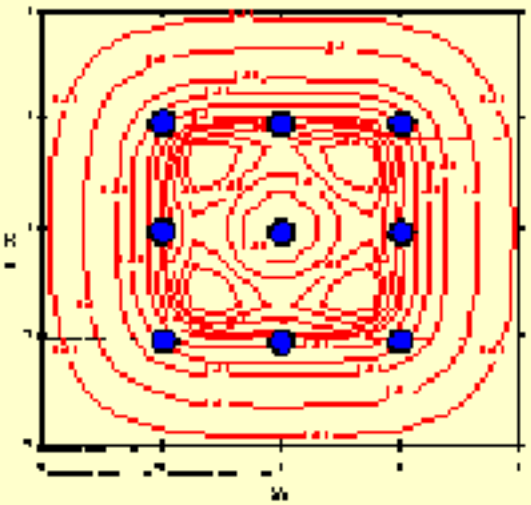


FIGURE 3.16
Three-Dimensional Illustration for the Information Function of a 3^2 Design

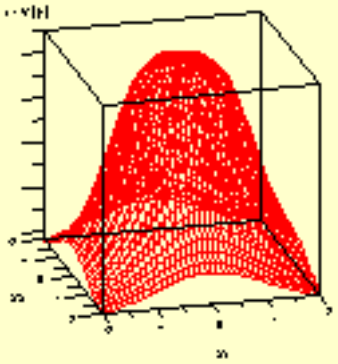


FIGURE 3.17
Contour Map of the Information Function for a 3^2 Design

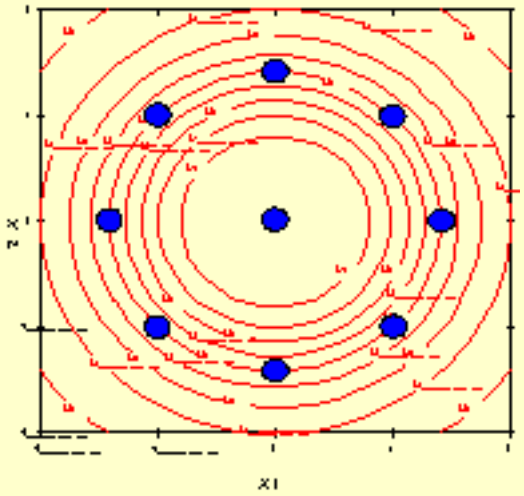


FIGURE 3.18
Three-Dimensional Illustration of the Information Function for a Rotatable Quadratic Design for Two Factors

FIGURE 3.19 Contour Map of the Information Function for a Rotatable Quadratic Design for Two Factors

Classical Quadratic Designs

Central composite and Box-Behnken designs

Introduced during the 1950's, classical quadratic designs fall into two broad categories: Box-Wilson central composite designs and Box-Behnken designs. The next sections describe these design classes and their properties.

5.3.3.6.1. Central Composite Designs (CCD)

Box-Wilson Central Composite Designs

CCD designs start with a factorial or fractional factorial design (with center points) and add "star" points to estimate curvature

A Box-Wilson Central Composite Design, commonly called `a central composite design,' contains an imbedded factorial or fractional factorial design with center points that is augmented with a group of `star points' that allow estimation of curvature. If the distance from the center of the design space to a factorial point is ±1 unit for each factor, the distance from the center of the design space to a star point is $\pm\alpha$ with $|\alpha| > 1$. The precise value of α depends on certain properties desired for the design and on the number of factors involved.

Similarly, the number of centerpoint runs the design is to contain also depends on certain properties required for the design.

Diagram of central composite design generation for two factors

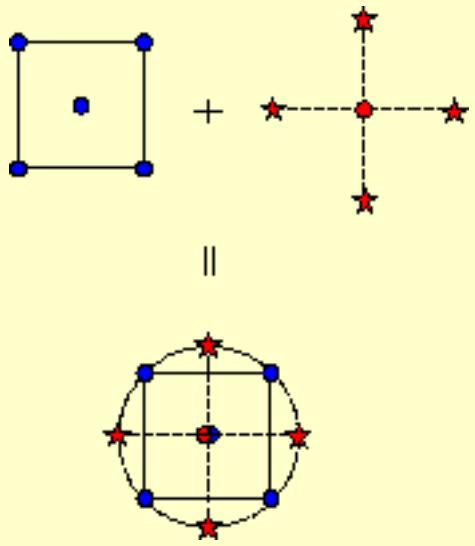


FIGURE 3.20 Generation of a Central Composite Design for Two Factors

A CCD design with k factors has 2k star points

A central composite design always contains twice as many star points as there are factors in the design. The star points represent new extreme values (low and high) for each factor in the design. Table 3.22 summarizes the properties of the three varieties of central composite designs. Figure 3.21 illustrates the relationships among these varieties.

Description of 3 types of CCD designs, which depend on where the star points are placed

TABLE 3.22 Central Composite Designs

Central Composite Design Type	Terminology	Comments
Circumscribed	CCC	CCC designs are the original form of the central composite design. The star points are at some distance α from the center based on the properties desired for the design and the number of factors in the design. The star points establish new extremes for the low and high settings for all factors. Figure 5 illustrates a CCC design. These designs have circular, spherical, or hyperspherical symmetry and require 5 levels for each factor. Augmenting an existing factorial or resolution V fractional factorial design with star points can produce this design.
Inscribed	CCI	For those situations in which the limits specified for factor settings are truly limits, the CCI design uses the factor settings as the star points and creates a factorial or fractional factorial design within those limits (in other words, a CCI design is a scaled down CCC design with each factor level of the CCC design divided by α to generate the CCI design). This design also requires 5 levels of each factor.

Face Centered	CCF	In this design the star points are at the center of each face of the factorial space, so α = ± 1. This variety requires 3 levels of each factor. Augmenting an existing factorial or resolution V design with appropriate star points can also produce this design.

Pictorial representation of where the star points are placed for the 3 types of CCD designs

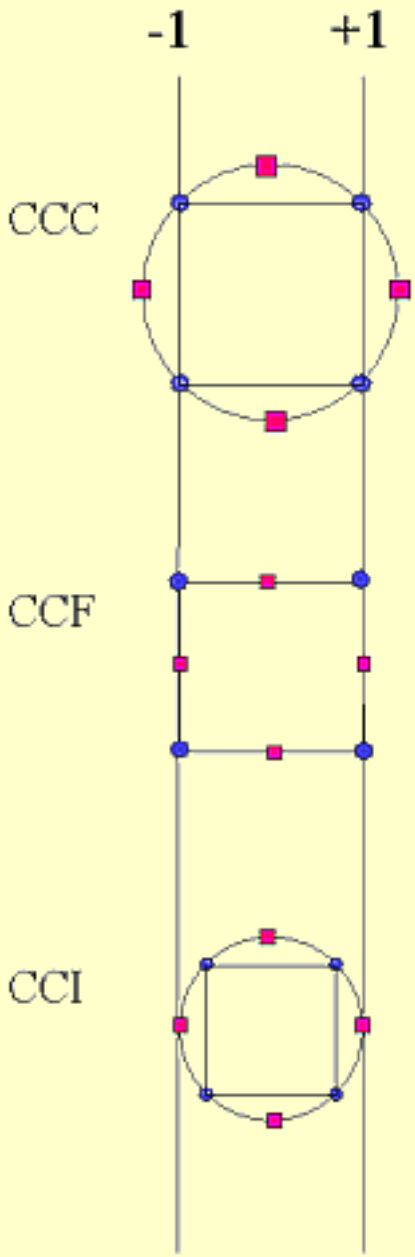


FIGURE 3.21 Comparison of the Three Types of Central Composite Designs

Comparison of the 3 central composite designs

The diagrams in Figure 3.21 illustrate the three types of central composite designs for two factors. Note that the CCC explores the largest process space and the CCI explores the smallest process space. Both the CCC and CCI are rotatable designs, but the CCF is not. In the CCC design, the design points describe a circle *circumscribed* about the factorial square. For three factors, the CCC design points describe a sphere around the factorial cube.

Determining α in Central Composite Designs

The value of α is chosen to maintain rotatability

To maintain rotatability, the value of α depends on the number of experimental runs in the factorial portion of the central composite design:

$$\alpha = [\text{number of factorial runs}]^{1/4}$$

If the factorial is a full factorial, then

$$\alpha = \left[2^k\right]^{1/4}$$

However, the factorial portion can also be a fractional factorial design of resolution V.

Table 3.23 illustrates some typical values of α as a function of the number of factors.

Values of α depending on the number of factors in the factorial part of the design

TABLE 3.23 Determining α for Rotatability

Number of Factors	Factorial Portion	Scaled Value for α Relative to ±1
2	2^2	$2^{2/4} = 1.414$
3	2^3	$2^{3/4} = 1.682$
4	2^4	$2^{4/4} = 2.000$
5	2^5	$2^{5-1} = 2.000$
5	2^5	$2^{5/4} = 2.378$
6	2^{6-1}	$2^{5/4} = 2.378$
6	2^6	$2^{6/4} = 2.828$

Orthogonal blocking

The value of α also depends on whether or not the design is orthogonally blocked. That is, the question is whether or not the design is divided into blocks such that the block effects do not affect the estimates of the coefficients in the 2nd order model.

Example of both rotatability and orthogonal blocking for two factors

Under some circumstances, the value of α allows simultaneous rotatability and orthogonality. One such example for $k = 2$ is shown below:

BLOCK	X1	X2
1	-1	-1
1	1	-1
1	-1	1
1	1	1
1	0	0
1	0	0
2	-1.414	0
2	1.414	0
2	0	-1.414
2	0	1.414
2	0	0
2	0	0

Additional central composite designs

Examples of other central composite designs will be given after Box-Behnken designs are described.

5.3.3.6.2. Box-Behnken designs

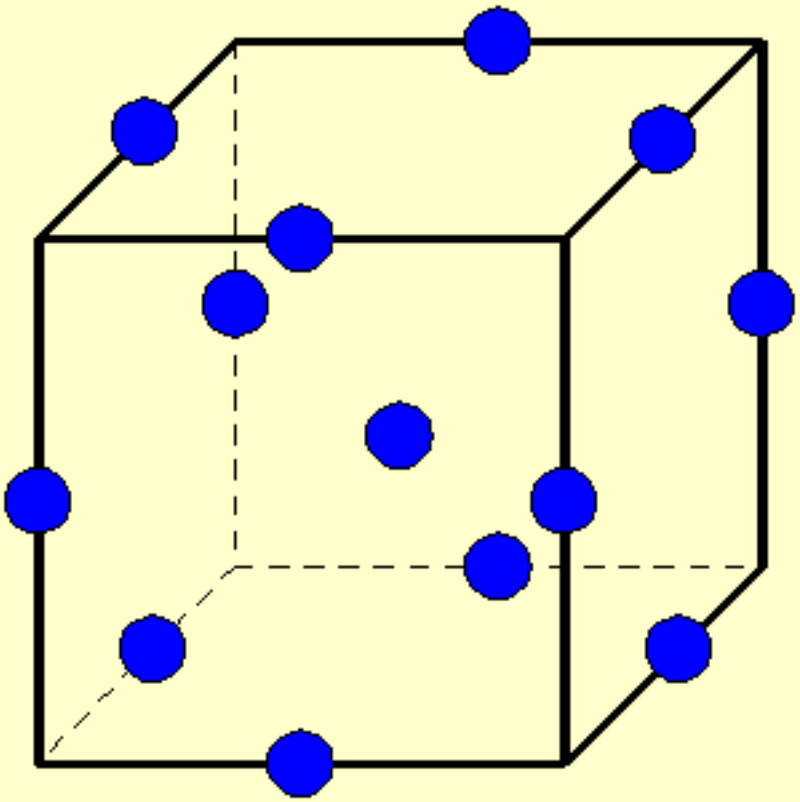
An alternate choice for fitting quadratic models that requires 3 levels of each factor and is rotatable (or "nearly" rotatable)

The Box-Behnken design is an independent quadratic design in that it does not contain an embedded factorial or fractional factorial design. In this design the treatment combinations are at the midpoints of edges of the process space and at the center. These designs are rotatable (or near rotatable) and require 3 levels of each factor. The designs have limited capability for orthogonal blocking compared to the central composite designs.

Figure 3.22 illustrates a Box-Behnken design for three factors.

Box-Behnken design for 3 factors

FIGURE 3.22 A Box-Behnken Design for Three Factors



Geometry of the design

The geometry of this design suggests a sphere within the process space such that the surface of the sphere protrudes through each face with the surface of the sphere tangential to the midpoint of each edge of the space.

Examples of Box-Behnken designs are given on the next page.

5.3.3.6.3. Comparisons of response surface designs

Choosing a Response Surface Design

Various CCD designs and Box-Behnken designs are compared and their properties discussed

Table 3.24 contrasts the structures of four common quadratic designs one might use when investigating three factors. The table combines CCC and CCI designs because they are structurally identical.

For three factors, the Box-Behnken design offers some advantage in requiring a fewer number of runs. For 4 or more factors, this advantage disappears.

Structural comparisons of CCC (CCI), CCF, and Box-Behnken designs for three factors

TABLE 3.24 Structural Comparisons of CCC (CCI), CCF, and Box-Behnken Designs for Three Factors

CCC (CCI)				CCF				Box-Behnken			
Rep	***X*1**	***X*2**	***X*3**	**Rep**	***X*1**	***X*2**	***X*3**	**Rep**	***X*1**	***X*2**	***X*3**
1	-1	-1	-1	1	-1	-1	-1	1	-1	-1	0
1	+1	-1	-1	1	+1	-1	-1	1	+1	-1	0
1	-1	+1	-1	1	-1	+1	-1	1	-1	+1	0
1	+1	+1	-1	1	+1	+1	-1	1	+1	+1	0
1	-1	-1	+1	1	-1	-1	+1	1	-1	0	-1
1	+1	-1	+1	1	+1	-1	+1	1	+1	0	-1
1	-1	+1	+1	1	-1	+1	+1	1	-1	0	+1
1	+1	+1	+1	1	+1	+1	+1	1	+1	0	+1
1	-1.682	0	0	1	-1	0	0	1	0	-1	-1
1	1.682	0	0	1	+1	0	0	1	0	+1	-1
1	0	-1.682	0	1	0	-1	0	1	0	-1	+1
1	0	1.682	0	1	0	+1	0	1	0	+1	+1

1	0	0	-1.682	1	0	0	-1	3	0	0	0
1	0	0	1.682	1	0	0	+1				
6	0	0	0	6	0	0	0				
Total Runs = 20				**Total Runs = 20**				**Total Runs = 15**			

Factor settings for CCC and CCI three factor designs

Table 3.25 illustrates the factor settings required for a central composite circumscribed (CCC) design and for a central composite inscribed (CCI) design (standard order), assuming three factors, each with low and high settings of 10 and 20, respectively. Because the CCC design generates new extremes for all factors, the investigator must inspect any worksheet generated for such a design to make certain that the factor settings called for are reasonable.

In Table 3.25, treatments 1 to 8 in each case are the factorial points in the design; treatments 9 to 14 are the star points; and 15 to 20 are the system-recommended center points. Notice in the CCC design how the low and high values of each factor have been extended to create the star points. In the CCI design, the specified low and high values become the star points, and the system computes appropriate settings for the factorial part of the design inside those boundaries.

TABLE 3.25 Factor Settings for CCC and CCI Designs for Three Factors

Central Composite Circumscribed CCC					**Central Composite Inscribed CCI**			
Sequence Number	***X*1**	***X*2**	***X*3**		**Sequence Number**	***X*1**	***X*2**	***X*3**
1	10	10	10		**1**	12	12	12
2	20	10	10		**2**	18	12	12
3	10	20	10		**3**	12	18	12
4	20	20	10		**4**	18	18	12
5	10	10	20		**5**	12	12	18
6	20	10	20		**6**	18	12	18
7	10	20	20		**7**	12	12	18
8	20	20	20		**8**	18	18	18
9	6.6	15	15	*	**9**	10	15	15
10	23.4	15	15	*	**10**	20	15	15
11	15	6.6	15	*	**11**	15	10	15
12	15	23.4	15	*	**12**	15	20	15
13	15	15	6.6	*	**13**	15	15	10
14	15	15	23.4	*	**14**	15	15	20
15	15	15	15		**15**	15	15	15
16	15	15	15		**16**	15	15	15
17	15	15	15		**17**	15	15	15

18	15	15	15		**18**	15	15	15
19	15	15	15		**19**	15	15	15
20	15	15	15		**20**	15	15	15

* are star points

Factor settings for CCF and Box-Behnken three factor designs

Table 3.26 illustrates the factor settings for the corresponding central composite face-centered (CCF) and Box-Behnken designs. Note that each of these designs provides three levels for each factor and that the Box-Behnken design requires fewer runs in the three-factor case.

TABLE 3.26 Factor Settings for CCF and Box-Behnken Designs for Three Factors

Central Composite Face-Centered CCC					**Box-Behnken**			
Sequence Number	***X*1**	***X*2**	***X*3**		**Sequence Number**	***X*1**	***X*2**	***X*3**
1	10	10	10		**1**	10	10	10
2	20	10	10		**2**	20	10	15
3	10	20	10		**3**	10	20	15
4	20	20	10		**4**	20	20	15
5	10	10	20		**5**	10	15	10
6	20	10	20		**6**	20	15	10
7	10	20	20		**7**	10	15	20
8	20	20	20		**8**	20	15	20
9	10	15	15	*	**9**	15	10	10
10	20	15	15	*	**10**	15	20	10
11	15	10	15	*	**11**	15	10	20
12	15	20	15	*	**12**	15	20	20
13	15	15	10	*	**13**	15	15	15
14	15	15	20	*	**14**	15	15	15
15	15	15	15		**15**	15	15	15
16	15	15	15					
17	15	15	15					
18	15	15	15					
19	15	15	15					
20	15	15	15					

* are star points for the CCC

Properties of classical response surface designs

Table 3.27 summarizes properties of the classical quadratic designs. Use this table for broad guidelines when attempting to choose from among available designs.

TABLE 3.27 Summary of Properties of Classical Response Surface Designs

Design Type	Comment
CCC	CCC designs provide high quality predictions over the entire design space, but require factor settings outside the range of the factors in the factorial part. **Note:** When the possibility of running a CCC design is recognized before starting a factorial experiment, factor spacings can be reduced to ensure that $\pm \alpha$ for each coded factor corresponds to feasible (reasonable) levels. Requires 5 levels for each factor.
CCI	CCI designs use only points within the factor ranges originally specified, but do not provide the same high quality prediction over the entire space compared to the CCC. Requires 5 levels of each factor.
CCF	CCF designs provide relatively high quality predictions over the entire design space and do not require using points outside the original factor range. However, they give poor precision for estimating pure quadratic coefficients. Requires 3 levels for each factor.
Box-Behnken	These designs require fewer treatment combinations than a central composite design in cases involving 3 or 4 factors. The Box-Behnken design is rotatable (or nearly so) but it contains regions of poor prediction quality like the CCI. Its "missing corners" may be useful when the experimenter should avoid combined factor extremes. This property prevents a potential loss of data in those cases. Requires 3 levels for each factor.

Number of runs required by central composite and Box-Behnken designs

Table 3.28 compares the number of runs required for a given number of factors for various Central Composite and Box-Behnken designs.

TABLE 3.28 Number of Runs Required by Central Composite and Box-Behnken Designs

Number of Factors	Central Composite	Box-Behnken
2	13 (5 center points)	-
3	20 (6 centerpoint runs)	15
4	30 (6 centerpoint runs)	27
5	33 (fractional factorial) or 52 (full factorial)	46
6	54 (fractional factorial) or 91 (full factorial)	54

Desirable Features for Response Surface Designs

A summary of desirable properties for response surface designs

G. E. P. Box and N. R. Draper in "Empirical Model Building and Response Surfaces," John Wiley and Sons, New York, 1987, page 477, identify desirable properties for a response surface design:

- Satisfactory distribution of information across the experimental region.
  - *rotatability*
- Fitted values are as close as possible to observed values.
  - *minimize residuals or error of prediction*
- Good lack of fit detection.
- Internal estimate of error.
- Constant variance check.
- Transformations can be estimated.
- Suitability for blocking.
- Sequential construction of higher order designs from simpler designs
- Minimum number of treatment combinations.
- Good graphical analysis through simple data patterns.
- Good behavior when errors in settings of input variables occur.

5.3.3.6.4. Blocking a response surface design

How can we block a response surface design?

When augmenting a resolution V design to a CCC design by adding star points, it may be desirable to block the design

If an investigator has run either a 2^k full factorial or a 2^{k-p} fractional factorial design of at least resolution V, augmentation of that design to a central composite design (either CCC of CCF) is easily accomplished by adding an additional set (block) of star and centerpoint runs. If the factorial experiment indicated (via the *t* test) curvature, this composite augmentation is the best follow-up option (follow-up options for other situations will be discussed later).

An orthogonal blocked response surface design has advantages

An important point to take into account when choosing a response surface design is the possibility of running the design in blocks. Blocked designs are better designs if the design allows the estimation of individual and interaction factor effects independently of the block effects. This condition is called orthogonal blocking. Blocks are assumed to have no impact on the nature and shape of the response surface.

CCF designs cannot be orthogonally blocked

The CCF design does not allow orthogonal blocking and the Box-Behnken designs offer blocking only in limited circumstances, whereas the CCC does permit orthogonal blocking.

Axial and factorial blocks

In general, when two blocks are required there should be an axial block and a factorial block. For three blocks, the factorial block is divided into two blocks and the axial block is not split. The blocking of the factorial design points should result in orthogonality between blocks and individual factors and between blocks and the two factor interactions.

The following Central Composite design in two factors is broken into two blocks.

Table of CCD design with 2 factors and 2 blocks

TABLE 3.29 CCD: 2 Factors, 2 Blocks

Pattern	Block	*X*1	*X*2	Comment
--	+1	-1	-1	Full Factorial
-+	+1	-1	+1	Full Factorial
+-	+1	+1	-1	Full Factorial
++	+1	+1	+1	Full Factorial
00	+1	0	0	Center-Full Factorial
00	+1	0	0	Center-Full Factorial
00	+1	0	0	Center-Full Factorial
-0	+2	-1.414214	0	Axial
+0	+2	+1.414214	0	Axial
0-	+2	0	-1.414214	Axial
0+	+2	0	+1.414214	Axial
00	+2	0	0	Center-Axial
00	+2	0	0	Center-Axial
00	+2	0	0	Center-Axial

Note that the first block includes the full factorial points and three centerpoint replicates. The second block includes the axial points and another three centerpoint replicates. Naturally these two blocks should be run as two separate random sequences.

Table of CCD design with 3 factors and 3 blocks

The following three examples show blocking structure for various designs.

TABLE 3.30 CCD: 3 Factors 3 Blocks, Sorted by Block

Pattern	Block	*X*1	*X*2	*X*3	Comment
---	1	-1	-1	-1	Full Factorial
-++	1	-1	+1	+1	Full Factorial
+-+	1	+1	-1	+1	Full Factorial
++-	1	+1	+1	-1	Full Factorial
000	1	0	0	0	Center-Full Factorial
000	1	0	0	0	Center-Full Factorial
--+	2	-1	-1	+1	Full Factorial

-+-	2	-1	+1	-1	Full Factorial
+--	2	+1	-1	-1	Full Factorial
+++	2	+1	+1	+1	Full Factorial
000	2	0	0	0	Center-Full Factorial
000	2	0	0	0	Center-Full Factorial
-00	3	-1.681793	0	0	Axial
+00	3	+1.681793	0	0	Axial
0-0	3	0	-1.681793	0	Axial
0+0	3	0	+1.681793	0	Axial
00-	3	0	0	-1.681793	Axial
00+	3	0	0	+1.681793	Axial
000	3	0	0	0	Axial
000	3	0	0	0	Axial

Table of CCD design with 4 factors and 3 blocks

TABLE 3.31 CCD: 4 Factors, 3 Blocks

Pattern	Block	*X*1	*X*2	*X*3	*X*4	Comment
---+	1	-1	-1	-1	+1	Full Factorial
--+-	1	-1	-1	+1	-1	Full Factorial
-+--	1	-1	+1	-1	-1	Full Factorial
-+++	1	-1	+1	+1	+1	Full Factorial
+---	1	+1	-1	-1	-1	Full Factorial
+-++	1	+1	-1	+1	+1	Full Factorial
++-+	1	+1	+1	-1	+1	Full Factorial
+++-	1	+1	+1	+1	-1	Full Factorial
0000	1	0	0	0	0	Center-Full Factorial
0000	1	0	0	0	0	Center-Full Factorial
----	2	-1	-1	-1	-1	Full Factorial
--++	2	-1	-1	+1	+1	Full Factorial
-+-+	2	-1	+1	-1	+1	Full Factorial
-++-	2	-1	+1	+1	-1	Full Factorial
+--+	2	+1	-1	-1	+1	Full Factorial
+-+-	2	+1	-1	+1	-1	Full Factorial
++--	2	+1	+1	-1	-1	Full Factorial
++++	2	+1	+1	+1	+1	Full Factorial
0000	2	0	0	0	0	Center-Full Factorial
0000	2	0	0	0	0	Center-Full Factorial
-000	3	-2	0	0	0	Axial
+000	3	+2	0	0	0	Axial
+000	3	+2	0	0	0	Axial
0-00	3	0	-2	0	0	Axial
0+00	3	0	+2	0	0	Axial
00-0	3	0	0	-2	0	Axial

00+0	3	0	0	+2	0	Axial
000-	3	0	0	0	-2	Axial
000+	3	0	0	0	+2	Axial
0000	3	0	0	0	0	Center-Axial

Table of CCD design with 5 factors and 2 blocks

TABLE 3.32 CCD: 5 Factors, 2 Blocks

Pattern	Block	*X*1	*X*2	*X*3	*X*4	*X*5	Comment
----+	1	-1	-1	-1	-1	+1	Fractional Factorial
---+-	1	-1	-1	-1	+1	-1	Fractional Factorial
--+--	1	-1	-1	+1	-1	-1	Fractional Factorial
--+++	1	-1	-1	+1	+1	+1	Fractional Factorial
-+---	1	-1	+1	-1	-1	-1	Fractional Factorial
-+-++	1	-1	+1	-1	+1	+1	Fractional Factorial
-++-+	1	-1	+1	+1	-1	+1	Fractional Factorial
-+++-	1	-1	+1	+1	+1	-1	Fractional Factorial
+----	1	+1	-1	-1	-1	-1	Fractional Factorial
+--++	1	+1	-1	-1	+1	+1	Fractional Factorial
+-+-+	1	+1	-1	+1	-1	+1	Fractional Factorial
+-++-	1	+1	-1	+1	+1	-1	Fractional Factorial
++--+	1	+1	+1	-1	-1	+1	Fractional Factorial
++-+-	1	+1	+1	-1	+1	-1	Fractional Factorial
+++--	1	+1	+1	+1	-1	-1	Fractional Factorial
+++++	1	+1	+1	+1	+1	+1	Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
00000	1	0	0	0	0	0	Center-Fractional Factorial
-0000	2	-2	0	0	0	0	Axial
+0000	2	+2	0	0	0	0	Axial
0-000	2	0	-2	0	0	0	Axial
0+000	2	0	+2	0	0	0	Axial
00-00	2	0	0	-2	0	0	Axial
00+00	2	0	0	+2	0	0	Axial
000-0	2	0	0	0	-2	0	Axial

000+0	2	0	0	0	+2	0	Axial
0000-	2	0	0	0	0	-2	Axial
0000+	2	0	0	0	0	+2	Axial
00000	2	0	0	0	0	0	Center-Axial

5.3.3.7. Adding centerpoints

Center point, or `Control' Runs

Centerpoint runs provide a check for both process stability and possible curvature

As mentioned earlier in this section, we add centerpoint runs interspersed among the experimental setting runs for two purposes:

1. To provide a measure of process stability and inherent variability
2. To check for curvature.

Centerpoint runs are not randomized

Centerpoint runs should begin and end the experiment, and should be dispersed as evenly as possible throughout the design matrix. The centerpoint runs are not randomized! There would be no reason to randomize them as they are there as guardians against process instability and the best way to find instability is to sample the process on a regular basis.

Rough rule of thumb is to add 3 to 5 center point runs to your design

With this in mind, we have to decide on how many centerpoint runs to do. This is a tradeoff between the resources we have, the need for enough runs to see if there is process instability, and the desire to get the experiment over with as quickly as possible. *As a rough guide, you should generally add approximately 3 to 5 centerpoint runs to a full or fractional factorial design.*

Table of randomized, replicated 2^3 full factorial design with centerpoints

In the following Table we have added three centerpoint runs to the otherwise randomized design matrix, making a total of nineteen runs.

TABLE 3.32 Randomized, Replicated 2^3 Full Factorial Design Matrix with Centerpoint Control Runs Added

	Random Order	Standard Order	SPEED	FEED	DEPTH
1	not applicable	not applicable	0	0	0
2	1	5	-1	-1	1
3	2	15	-1	1	1
4	3	9	-1	-1	-1
5	4	7	-1	1	1
6	5	3	-1	1	-1
7	6	12	1	1	-1
8	7	6	1	-1	1
9	8	4	1	1	-1
10	not applicable	not applicable	0	0	0
11	9	2	1	-1	-1
12	10	13	-1	-1	1
13	11	8	1	1	1
14	12	16	1	1	1
15	13	1	-1	-1	-1
16	14	14	1	-1	1
17	15	11	-1	1	-1
18	16	10	1	-1	-1
19	not applicable	not applicable	0	0	0

Preparing a worksheet for operator of experiment

To prepare a worksheet for an operator to use when running the experiment, delete the columns `RandOrd' and `Standard Order.' Add an additional column for the output (Yield) on the right, and change all `-1', `0', and `1' to original factor levels as follows.

Operator worksheet

TABLE 3.33 DOE Worksheet Ready to Run

Sequence Number	Speed	Feed	Depth	Yield
1	20	0.003	0.015	
2	16	0.001	0.02	
3	16	0.005	0.02	
4	16	0.001	0.01	
5	16	0.005	0.02	
6	16	0.005	0.01	
7	24	0.005	0.01	
8	24	0.001	0.02	
9	24	0.005	0.01	
10	20	0.003	0.015	
11	24	0.001	0.01	
12	16	0.001	0.02	
13	24	0.005	0.02	
14	24	0.005	0.02	
15	16	0.001	0.01	
16	24	0.001	0.02	
17	16	0.005	0.01	
18	24	0.001	0.01	
19	20	0.003	0.015	

Note that the control (centerpoint) runs appear at rows 1, 10, and 19.

This worksheet can be given to the person who is going to do the runs/measurements and asked to proceed through it from first row to last in that order, filling in the Yield values as they are obtained.

Pseudo Center points

Center points for discrete factors

One often runs experiments in which some factors are nominal. For example, Catalyst "A" might be the (-1) setting, catalyst "B" might be coded (+1). The choice of which is "high" and which is "low" is arbitrary, but one must have some way of deciding which catalyst setting is the "standard" one.

These standard settings for the discrete input factors together with center points for the continuous input factors, will be regarded as the "center points" for purposes of design.

Center Points in Response Surface Designs

Uniform precision

In an unblocked response surface design, the number of center points controls other properties of the design matrix. The number of center points can make the design orthogonal or have "uniform precision." We will only focus on uniform precision here as classical quadratic designs were set up to have this property.

Variance of prediction

Uniform precision ensures that the variance of prediction is the same at the center of the experimental space as it is at a unit distance away from the center.

Protection against bias

In a response surface context, to contrast the virtue of uniform precision designs over replicated center-point orthogonal designs one should also consider the following guidance from Montgomery ("Design and Analysis of Experiments," Wiley, 1991, page 547), "*A uniform precision design offers more protection against bias in the regression coefficients than does an orthogonal design because of the presence of third-order and higher terms in the true surface*.

Controlling α and the number of center points

Myers, Vining, et al, ["Variance Dispersion of Response Surface Designs," Journal of Quality Technology, 24, pp. 1-11 (1992)] have explored the options regarding the number of center points and the value of α somewhat further: An investigator may control two parameters, α and the number of center points (n_c), given k factors. Either set $\alpha = 2^{(k/4)}$ (for rotatability) or $\sqrt{k}$ -- an axial point on perimeter of design region. Designs are similar in performance with $\sqrt{k}$ preferable as k increases. Findings indicate that the best overall design performance occurs with $\alpha \approx \sqrt{k}$ and $2 \leq n_c \leq 5$.

5.3.3.8. Improving fractional factorial design resolution

Foldover designs increase resolution

Earlier we saw how fractional factorial designs resulted in an alias structure that confounded main effects with certain interactions. Often it is useful to know how to run a few additional treatment combinations to remove alias structures that might be masking significant effects or interactions.

Partial foldover designs break up specific alias patterns

Two methods will be described for selecting these additional treatment combinations:

- Mirror-image foldover designs (to build a resolution IV design from a resolution III design)
- Alternative foldover designs (to break up specific alias patterns).

5.3.3.8.1. Mirror-Image foldover designs

A foldover design is obtained from a fractional factorial design by reversing the signs of all the columns

A mirror-image fold-over (or foldover, without the hyphen) design is used to augment fractional factorial designs to increase the resolution of 2^{3-1}_{III} and Plackett-Burman designs. It is obtained by reversing the signs of all the columns of the original design matrix. The original design runs are combined with the mirror-image fold-over design runs, and this combination can then be used to estimate all main effects clear of any two-factor interaction. This is referred to as: *breaking the alias link between main effects and two-factor interactions.*

Before we illustrate this concept with an example, we briefly review the basic concepts involved.

Review of Fractional 2^{k-p} Designs

A resolution III design, combined with its mirror-image foldover, becomes resolution IV

In general, a design type that uses a specified fraction of the runs from a full factorial and is balanced and orthogonal is called a *fractional factorial*.

A 2-level fractional factorial is constructed as follows: *Let the number of runs be 2^{k-p}. Start by constructing the full factorial for the k-p variables. Next associate the extra factors with higher-order interaction columns. The Table shown previously details how to do this to achieve a minimal amount of confounding.*

For example, consider the 2^{5-2} design (a resolution III design). The full factorial for $k = 5$ requires $2^5 = 32$ runs. The fractional factorial can be achieved in $2^{5-2} = 8$ runs, called a quarter (1/4) fractional design, by setting $X4 = X1*X2$ and $X5 = X1*X3$.

Design matrix for a 2^{5-2} fractional factorial

The design matrix for a 2^{5-2} fractional factorial looks like:

TABLE 3.34 Design Matrix for a 2^{5-2} Fractional Factorial

run	*X*1	*X*2	*X*3	*X*4 = *X*1*X*2	*X*5 = *X*1*X*3
1	-1	-1	-1	+1	+1
2	+1	-1	-1	-1	-1
3	-1	+1	-1	-1	+1
4	+1	+1	-1	+1	-1
5	-1	-1	+1	+1	-1
6	+1	-1	+1	-1	+1
7	-1	+1	+1	-1	-1
8	+1	+1	+1	+1	+1

Design Generators, Defining Relation and the Mirror-Image Foldover

Increase to resolution IV design by augmenting design matrix

In this design the *X*1*X*2 column was used to generate the *X*4 main effect and the *X*1*X*3 column was used to generate the *X*5 main effect. The design generators are: 4 = 12 and 5 = 13 and the defining relation is I = 124 = 135 = 2345. Every main effect is confounded (aliased) with at least one first-order interaction (see the confounding structure for this design).

We can increase the resolution of this design to IV if we augment the 8 original runs, adding on the 8 runs from the mirror-image fold-over design. These runs make up another 1/4 fraction design with design generators 4 = -12 and 5 = -13 and defining relation I = -124 = -135 = 2345. The augmented runs are:

Augmented runs for the design matrix

run	*X*1	*X*2	*X*3	*X*4 = -*X*1*X*2	*X*5 = -*X*1*X*3
9	+1	+1	+1	-1	-1
10	-1	+1	+1	+1	+1
11	+1	-1	+1	+1	-1
12	-1	-1	+1	-1	+1
13	+1	+1	-1	-1	+1
14	-1	+1	-1	+1	-1
15	+1	-1	-1	+1	+1
16	-1	-1	-1	-1	-1

Mirror-image foldover design reverses all signs in original design matrix

A *mirror-image foldover design* is the original design with *all signs reversed*. It breaks the alias chains between *every main factor and two-factor interaction*of a resolution III design. That is, we can estimate *all the main effects clear of any two-factor interaction*.

A 1/16 Design Generator Example

2^{7-3} example

Now we consider a more complex example.

We would like to study the effects of 7 variables. A full 2-level factorial, 2^7, would require 128 runs.

Assume economic reasons restrict us to 8 runs. We will build a $2^{7-4} = 2^3$ full factorial and assign certain products of columns to the *X*4, *X*5, *X*6 and *X*7 variables. This will generate a resolution III design in which all of the main effects are aliased with first-order and higher interaction terms. The design matrix (see the previous Table for a complete description of this fractional factorial design) is:

Design matrix for 2^{7-3} fractional factorial

Design Matrix for a 2^{7-3} Fractional Factorial

run	*X*1	*X*2	*X*3	*X*4 = *X*1*X*2	*X*5 = *X*1*X*3	*X*6 = *X*2*X*3	*X*7 = *X*1*X*2*X*3
1	-1	-1	-1	+1	+1	+1	-1
2	+1	-1	-1	-1	-1	+1	+1
3	-1	+1	-1	-1	+1	-1	+1
4	+1	+1	-1	+1	-1	-1	-1
5	-1	-1	+1	+1	-1	-1	+1
6	+1	-1	+1	-1	+1	-1	-1
7	-1	+1	+1	-1	-1	+1	-1
8	+1	+1	+1	+1	+1	+1	+1

Design generators and defining relation for this example

The design generators for this 1/16 fractional factorial design are:

4 = 12, 5 = 13, 6 = 23 and 7 = 123

From these we obtain, by multiplication, the defining relation:

I = 124 = 135 = 236 = 347 = 257 = 167 = 456 = 1237 = 2345 = 1346 = 1256 = 1457 = 2467 = 3567 = 1234567.

Computing alias structure for complete design

Using this defining relation, we can easily compute the alias structure for the complete design, as shown previously in the link to the fractional design Table given earlier. For example, to figure out which effects are aliased (confounded) with factor $X1$ we multiply the defining relation by 1 to obtain:

$$1 = 24 = 35 = 1236 = 1347 = 1257 = 67 = 1456 = 237 = 12345 = 346 = 256 = 457 = 12467 = 13567 = 234567$$

In order to simplify matters, let us ignore all interactions with 3 or more factors; we then have the following 2-factor alias pattern for $X1$: $1 = 24 = 35 = 67$ or, using the full notation, $X1 = X2*X4 = X3*X5 = X6*X7$.

The same procedure can be used to obtain all the other aliases for each of the main effects, generating the following list:

$$1 = 24 = 35 = 67$$
$$2 = 14 = 36 = 57$$
$$3 = 15 = 26 = 47$$
$$4 = 12 = 37 = 56$$
$$5 = 13 = 27 = 46$$
$$6 = 17 = 23 = 45$$
$$7 = 16 = 25 = 34$$

Signs in every column of original design matrix reversed for mirror-image foldover design

The chosen design used a set of generators with all positive signs. The mirror-image foldover design uses generators with negative signs for terms with an even number of factors or, 4 = -12, 5 = -13, 6 = -23 and 7 = 123. This generates a design matrix that is equal to the original design matrix with every sign in every column reversed.

If we augment the initial 8 runs with the 8 mirror-image foldover design runs (with all column signs reversed), we can de-alias all the main effect estimates from the 2-way interactions. The additional runs are:

Design matrix for mirror-image foldover runs

Design Matrix for the Mirror-Image Foldover Runs of the 2^{7-3} Fractional Factorial

run	*X*1	*X*2	*X*3	*X*4 = *X*1*X*2	*X*5 = *X*1*X*3	*X*6 = *X*2*X*3	*X*7 = *X*1*X*2*X*3
1	+1	+1	+1	-1	-1	-1	+1
2	-1	+1	+1	+1	+1	-1	-1
3	+1	-1	+1	+1	-1	+1	-1
4	-1	-1	+1	-1	+1	+1	+1
5	+1	+1	-1	-1	+1	+1	-1
6	-1	+1	-1	+1	-1	+1	+1
7	+1	-1	-1	+1	+1	-1	+1
8	-1	-1	-1	-1	-1	-1	-1

Alias structure for augmented runs

Following the same steps as before and making the same assumptions about the omission of higher-order interactions in the alias structure, we arrive at:

1 = -24 = -35 = -67
2 = -14 = -36 =- 57
3 = -15 = -26 = -47
4 = -12 = -37 = -56
5 = -13 = -27 = -46
6 = -17 = -23 = -45
7 = -16 = -25 = -34

With both sets of runs, we can now estimate all the main effects free from two factor interactions.

Build a resolution IV design from a resolution III design

Note*: In general, a mirror-image foldover design is a method to build a resolution IV design from a resolution III design. It is never used to follow-up a resolution IV design.*

5.3.3.8.2. Alternative foldover designs

Alternative foldover designs can be an economical way to break up a selected alias pattern

The mirror-image foldover (in which signs in all columns are reversed) is only one of the possible follow-up fractions that can be run to augment a fractional factorial design. It is the most common choice when the original fraction is resolution III. However, alternative foldover designs with fewer runs can often be utilized to break up selected alias patterns. We illustrate this by looking at what happens when the signs of a single factor column are reversed.

Example of de-aliasing a single factor

Previously, we described how we de-alias all the factors of a 2^{7-4} experiment. Suppose that we only want to de-alias the $X4$ factor. This can be accomplished by only changing the sign of $X4 = X1X2$ to $X4 = -X1X2$. The resulting design is:

Table showing design matrix of a reverse X4 foldover design

TABLE 3.36 A "Reverse X4" Foldover Design

run	*X*1	*X*2	*X*3	*X*4 = -*X*1*X*2	*X*5 = -*X*1*X*3	*X*6 = *X*2*X*3	*X*7 = *X*1*X*2*X*3
1	-1	-1	-1	-1	+1	+1	-1
2	+1	-1	-1	+1	-1	+1	+1
3	-1	+1	-1	+1	+1	-1	+1
4	+1	+1	-1	-1	-1	-1	-1
5	-1	-1	+1	-1	-1	-1	+1
6	+1	-1	+1	+1	+1	-1	-1
7	-1	+1	+1	+1	-1	+1	-1
8	+1	+1	+1	-1	+1	+1	+1

Alias patterns and effects that can be estimated in the example design

The two-factor alias patterns for $X4$ are: Original experiment: $X4 = X1X2 = X3X7 = X5X6$; "Reverse $X4$" foldover experiment: $X4 = -X1X2 = -X3X7 = -X5X6$.

The following effects can be estimated by combining the original 2^{7-4}_{III} with the "Reverse $X4$" foldover fraction:

$X1 + X3X5 + X6X7$
$X2 + X3X6 + X5X7$
$X3 + X1X5 + X2X6$
$X4$
$X5 + X1X3 + X2X7$
$X6 + X2X3 + X1X7$
$X7 + X2X5 + X1X6$
$X1X4$
$X2X4$
$X3X4$
$X4X5$
$X4X6$
$X4X7$
$X1X2 + X3X7 + X5X6$

Note: The 16 runs allow estimating the above 14 effects, with one degree of freedom left over for a possible block effect.

Advantage and disadvantage of this example design

The advantage of this follow-up design is that it permits estimation of the $X4$ effect and each of the six two-factor interaction terms involving $X4$.

The disadvantage is that the combined fractions still yield a resolution III design, with all main effects other than $X4$ aliased with two-factor interactions.

Case when purpose is simply to estimate all two-factor interactions of a single factor

Reversing a single factor column to obtain de-aliased two-factor interactions for that one factor works for any resolution III or IV design. When used to follow-up a resolution IV design, there are relatively few new effects to be estimated (as compared to 2^{k-p}_{III} designs). When the original resolution IV fraction provides sufficient precision, and the purpose of the follow-up runs is simply to estimate all two-factor interactions for one factor, the *semifolding* option should be considered.

Semifolding

Number of runs can be reduced for resolution IV designs

For resolution IV fractions, it is possible to economize on the number of runs that are needed to break the alias chains for all two-factor interactions of a single factor. In the above case we needed 8 additional runs, which is the same number of runs that were used in the original experiment. This can be improved upon.

Additional information on John's 3/4 designs

We can repeat only the points that were set at the high levels of the factor of choice and then run them at their low settings in the next experiment. For the given example, this means an additional 4 runs instead 8. We mention this technique only in passing, more details may be found in the references (or see John's 3/4 designs).

5.3.3.9. Three-level full factorial designs

Three-level designs are useful for investigating quadratic effects

The three-level design is written as a 3^k factorial design. It means that k factors are considered, each at 3 levels. These are (usually) referred to as low, intermediate and high levels. These levels are numerically expressed as 0, 1, and 2. One could have considered the digits -1, 0, and +1, but this may be confusing with respect to the 2-level designs since 0 is reserved for center points. Therefore, we will use the 0, 1, 2 scheme. The reason that the three-level designs were proposed is to model possible curvature in the response function and to handle the case of nominal factors at 3 levels. A third level for a continuous factor facilitates investigation of a quadratic relationship between the response and each of the factors.

Three-level design may require prohibitive number of runs

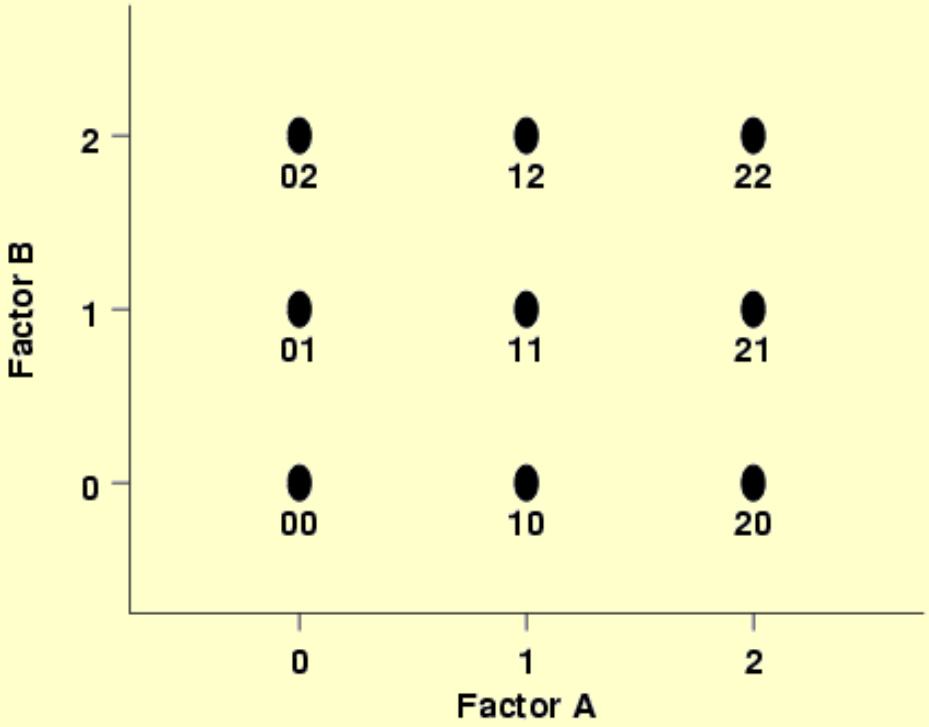
Unfortunately, the three-level design is prohibitive in terms of the number of runs, and thus in terms of cost and effort. For example a two-level design with center points is much less expensive while it still is a very good (and simple) way to establish the presence or absence of curvature.

The 3^2 design

The simplest 3-level design - with only 2 factors

This is the simplest three-level design. It has two factors, each at three levels. The 9 treatment combinations for this type of design can be shown pictorially as follows:

FIGURE 3.23 A 3^2 Design Schematic



A notation such as "20" means that factor A is at its high level (2) and factor B is at its low level (0).

The 3^3 design

The model and treatment runs for a 3 factor, 3-level design

This is a design that consists of three factors, each at three levels. It can be expressed as a 3 x 3 x 3 = 3^3 design. The model for such an experiment is

$$Y_{ijk} = \mu + A_i + B_j + AB_{ij} + C_k + AC_{ik} + BC_{jk} + ABC_{ijk} + \varepsilon_{ijk}$$

where each factor is included as a nominal factor rather than as a continuous variable. In such cases, main effects have 2 degrees of freedom, two-factor interactions have $2^2 = 4$ degrees of freedom and k-factor interactions have 2^k degrees of freedom. The model contains 2 + 2 + 2 + 4 + 4 + 4 + 8 = 26 degrees of freedom. Note that if there is no replication, the fit is exact and there is no error term (the epsilon term) in the model. In this no replication case, if one assumes that there are no three-factor interactions, then one can use these 8 degrees of freedom for error estimation.

In this model we see that i = 1, 2, 3, and similarly for j and k, making 27

treatments.

Table of treatments for the 3^3 design

These treatments may be displayed as follows:

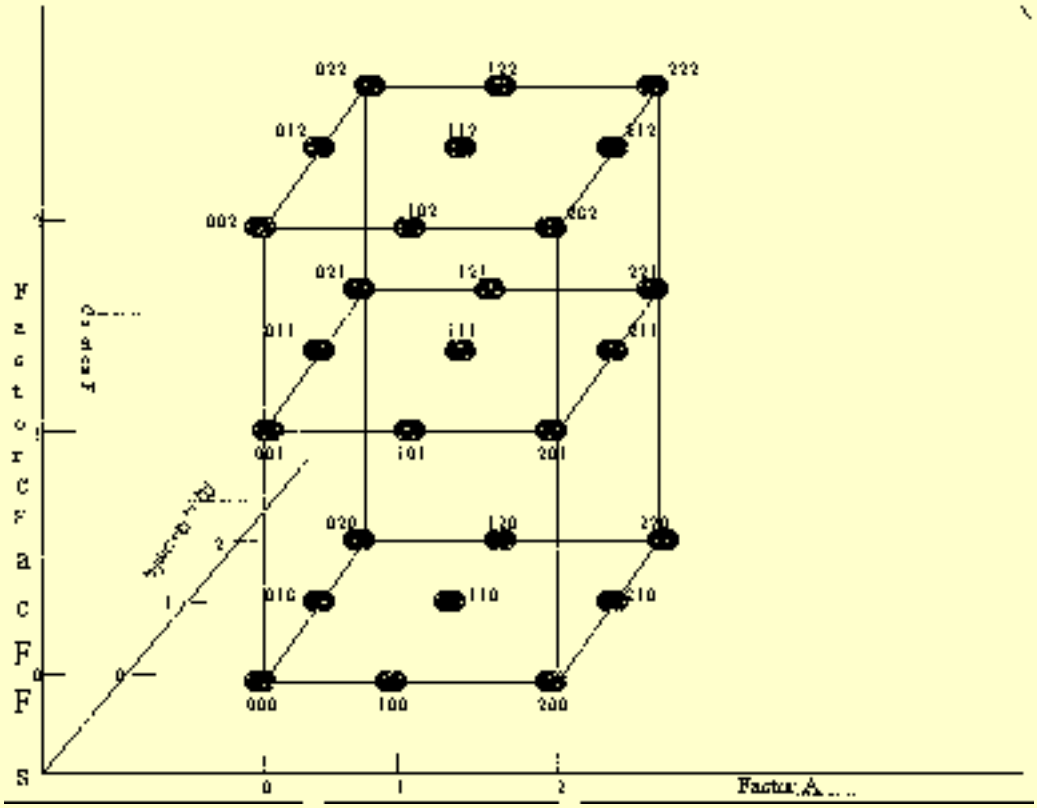
TABLE 3.37 The 3^3 Design

Factor B	Factor C	Factor A 0	Factor A 1	Factor A 2
0	0	000	100	200
0	1	001	101	201
0	2	002	102	202
1	0	010	110	210
1	1	011	111	211
1	2	012	112	212
2	0	020	120	220
2	1	021	121	221
2	2	022	122	222

Pictorial representation of the 3^3 design

The design can be represented pictorially by

FIGURE 3.24 A 3^3 Design Schematic



Two types of 3^k designs

Two types of fractions of 3^k designs are employed:

- Box-Behnken designs whose purpose is to estimate a second-order model for quantitative factors (discussed earlier in section 5.3.3.6.2)
- 3^{k-p} orthogonal arrays.

5.3.3.10. Three-level, mixed-level and fractional factorial designs

Mixed level designs have some factors with, say, 2 levels, and some with 3 levels or 4 levels

The 2^k and 3^k experiments are special cases of factorial designs. In a factorial design, one obtains data at every combination of the levels. The importance of factorial designs, especially 2-level factorial designs, was stated by Montgomery (1991): *It is our belief that the two-level factorial and fractional factorial designs should be the cornerstone of industrial experimentation for product and process development and improvement.* He went on to say: *There are, however, some situations in which it is necessary to include a factor (or a few factors) that have more than two levels.*

This section will look at how to add three-level factors starting with two-level designs, obtaining what is called a *mixed-level* design. We will also look at how to add a four-level factor to a two-level design. The section will conclude with a listing of some useful orthogonal three-level and mixed-level designs (a few of the so-called Taguchi "L" orthogonal array designs), and a brief discussion of their benefits and disadvantages.

Generating a Mixed Three-Level and Two-Level Design

Montgomery scheme for generating a mixed design

Montgomery (1991) suggests how to derive a variable at three levels from a 2^3 design, using a rather ingenious scheme. The objective is to generate a design for one variable, *A*, at 2 levels and another, *X*, at three levels. This will be formed by combining the -1 and 1 patterns for the *B* and *C* factors to form the levels of the three-level factor *X*:

TABLE 3.38 Generating a Mixed Design

Two-Level		Three-Level
B	**C**	**X**
-1	-1	x_1
+1	-1	x_2
-1	+1	x_2
+1	+1	x_3

Similar to the 3^k case, we observe that *X* has 2 degrees of freedom, which can be broken out into a linear and a quadratic component. To illustrate how the 2^3 design leads to the design with one factor at two levels and one factor at three levels, consider the following table, with particular attention focused on the column labels.

Table illustrating the generation of a design with one factor at 2 levels and another at 3 levels from a 2^3 design

	A	X_L	X_L	AX_L	AX_L	X_Q	AX_Q	TRT	MNT
Run	**A**	**B**	**C**	**AB**	**AC**	**BC**	**ABC**	**A**	**X**
1	-1	-1	-1	+1	+1	+1	-1	Low	Low
2	+1	-1	-1	-1	-1	+1	+1	High	Low
3	-1	+1	-1	-1	+1	-1	+1	Low	Medium
4	+1	+1	-1	+1	-1	-1	-1	High	Medium
5	-1	-1	+1	+1	-1	-1	+1	Low	Medium
6	+1	-1	+1	-1	+1	-1	-1	High	Medium
7	-1	+1	+1	-1	-1	+1	-1	Low	High

If quadratic effect negligble, we may include a second two-level factor

If we believe that the quadratic effect is negligible, we may include a second two-level factor, D, with D = ABC. In fact, we can convert the design to exclusively a main effect (resolution III) situation consisting of four two-level factors and one three-level factor. This is accomplished by equating the second two-level factor to AB, the third to AC and the fourth to ABC. Column BC cannot be used in this manner because it contains the quadratic effect of the three-level factor X.

More than one three-level factor

3-Level factors from 2^4 and 2^5 designs

We have seen that in order to create one three-level factor, the starting design can be a 2^3 factorial. Without proof we state that a 2^4 can split off 1, 2 or 3 three-level factors; a 2^5 is able to generate 3 three-level factors and still maintain a full factorial structure. For more on this, see Montgomery (1991).

Generating a Two- and Four-Level Mixed Design

Constructing a design with one 4-level factor and two 2-level factors

We may use the same principles as for the three-level factor example in creating a four-level factor. We will assume that the goal is to construct a design with one four-level and two two-level factors.

Initially we wish to estimate all main effects and interactions. It has been shown (see Montgomery, 1991) that this can be accomplished via a 2^4 (16 runs) design, with columns A and B used to create the four level factor *X*.

Table showing design with 4-level, two 2-level factors in 16 runs

TABLE 3.39 A Single Four-level Factor and Two Two-level Factors in 16 runs

Run	(A	B)	= X	C	D
1	-1	-1	x_1	-1	-1
2	+1	-1	x_2	-1	-1
3	-1	+1	x_3	-1	-1
4	+1	+1	x_4	-1	-1
5	-1	-1	x_1	+1	-1
6	+1	-1	x_2	+1	-1
7	-1	+1	x_3	+1	-1
8	+1	+1	x_4	+1	-1

9	-1	-1	x_1	-1	+1
10	+1	-1	x_2	-1	+1
11	-1	+1	x_3	-1	+1
12	+1	+1	x_4	-1	+1
13	-1	-1	x_1	+1	+1
14	+1	-1	x_2	+1	+1
15	-1	+1	x_3	+1	+1
16	+1	+1	x_4	+1	+1

Some Useful (Taguchi) Orthogonal "L" Array Designs

L_9 design

L_9 - A 3^{4-2} Fractional Factorial Design 4 Factors at Three Levels (9 runs)

Run	X1	X2	X3	X4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

L_{18} design

L_{18} - A 2 x 3^{7-5} Fractional Factorial (Mixed-Level) Design 1 Factor at Two Levels and Seven Factors at 3 Levels (18 Runs)

Run	X1	X2	X3	X4	X5	X6	X7	X8
1	1	1	1	1	1	1	1	1
2	1	1	2	2	2	2	2	2
3	1	1	3	3	3	3	3	3
4	1	2	1	1	2	2	3	3
5	1	2	2	2	3	3	1	1
6	1	2	3	3	1	1	2	2
7	1	3	1	2	1	3	2	3
8	1	3	2	3	2	1	3	1
9	1	3	3	1	3	2	1	2
10	2	1	1	3	3	2	2	1
11	2	1	2	1	1	3	3	2
12	2	1	3	2	2	1	1	3
13	2	2	1	2	3	1	3	2
14	2	2	2	3	1	2	1	3
15	2	2	3	1	2	3	2	1
16	2	3	1	3	2	3	1	2
17	2	3	2	1	3	1	2	3
18	2	3	3	2	1	2	3	1

L_{27} design

L_{27} - A 3^{13-10} Fractional Factorial Design
Thirteen Factors at Three Levels (27 Runs)

Run	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	2	2	2	2	2	2	2	2	2
3	1	1	1	1	3	3	3	3	3	3	3	3	3
4	1	2	2	2	1	1	1	2	2	2	3	3	3
5	1	2	2	2	2	2	2	3	3	3	1	1	1
6	1	2	2	2	3	3	3	1	1	1	2	2	2
7	1	3	3	3	1	1	1	3	3	3	2	2	2
8	1	3	3	3	2	2	2	1	1	1	3	3	3
9	1	3	3	3	3	3	3	2	2	2	1	1	1
10	2	1	2	3	1	2	3	1	2	3	1	2	3
11	2	1	2	3	2	3	1	2	3	1	2	3	1
12	2	1	2	3	3	1	2	3	1	2	3	1	2
13	2	2	3	1	1	2	3	2	3	1	3	1	2
14	2	2	3	1	2	3	1	3	1	2	1	2	3
15	2	2	3	1	3	1	2	1	2	3	2	3	1
16	2	3	1	2	1	2	3	3	1	2	2	3	1
17	2	3	1	2	2	3	1	1	2	3	3	1	2
18	2	3	1	2	3	1	2	2	3	1	1	2	3
19	3	1	3	2	1	3	2	1	3	2	1	3	2
20	3	1	3	2	2	1	3	2	1	3	2	1	3
21	3	1	3	2	3	2	1	3	2	1	3	2	1
22	3	2	1	3	1	3	2	2	1	3	3	2	1
23	3	2	1	3	2	1	3	3	2	1	1	3	2
24	3	2	1	3	3	2	1	1	3	2	2	1	3
25	3	3	2	1	1	3	2	3	2	1	2	1	3
26	3	3	2	1	2	1	3	1	3	2	3	2	1
27	3	3	2	1	3	2	1	2	1	3	1	3	2

L_{36} design

L36 - A Fractional Factorial (Mixed-Level) Design Eleven Factors at Two Levels and Twelve Factors at 3 Levels (36 Runs)

Run	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22	X23
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2
3	1	1	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3	3	3	3	3
4	1	1	1	1	1	2	2	2	2	2	2	1	1	1	1	2	2	2	2	3	3	3	3
5	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	3	3	3	3	1	1	1	1
6	1	1	1	1	1	2	2	2	2	2	2	3	3	3	3	1	1	1	1	2	2	2	2
7	1	1	2	2	2	1	1	1	2	2	2	1	1	2	3	1	2	3	3	1	2	2	3
8	1	1	2	2	2	1	1	1	2	2	2	2	2	3	1	2	3	1	1	2	3	3	1
9	1	1	2	2	2	1	1	1	2	2	2	3	3	1	2	3	1	2	2	3	1	1	2
10	1	2	1	2	2	1	2	2	1	1	2	1	1	3	2	1	3	2	3	2	1	3	2
11	1	2	1	2	2	1	2	2	1	1	2	2	2	1	3	2	1	3	1	3	2	1	3
12	1	2	1	2	2	1	2	2	1	1	2	3	3	2	1	3	2	1	2	1	3	2	1

13	1	2	2	1	2	2	1	2	1	2	1	1	2	3	1	3	2	1	3	3	2	1	2
14	1	2	2	1	2	2	1	2	1	2	1	2	3	1	2	1	3	2	1	1	3	2	3
15	1	2	2	1	2	2	1	2	1	2	1	3	1	2	3	2	1	3	2	2	1	3	1
16	1	2	2	2	1	2	2	1	2	1	1	1	2	3	2	1	1	3	2	3	3	2	1
17	1	2	2	2	1	2	2	1	2	1	1	2	3	1	3	2	2	1	3	1	1	3	2
18	1	2	2	2	1	2	2	1	2	1	1	3	1	2	1	3	3	2	1	2	2	1	3
19	2	1	2	2	1	1	2	2	1	2	1	1	2	1	3	3	3	1	2	2	1	2	3
20	2	1	2	2	1	1	2	2	1	2	1	2	3	2	1	1	1	2	3	3	2	3	1
21	2	1	2	2	1	1	2	2	1	2	1	3	1	3	2	2	2	3	1	1	3	1	2
22	2	1	2	1	2	2	2	1	1	1	2	1	2	2	3	3	1	2	1	1	3	3	2
23	2	1	2	1	2	2	2	1	1	1	2	2	3	3	1	1	2	3	2	2	1	1	3
24	2	1	2	1	2	2	2	1	1	1	2	3	1	1	2	2	3	1	3	3	2	2	1
25	2	1	1	2	2	2	1	2	2	1	1	1	3	2	1	2	3	3	1	3	1	2	2
26	2	1	1	2	2	2	1	2	2	1	1	2	1	3	2	3	1	1	2	1	2	3	3
27	2	1	1	2	2	2	1	2	2	1	1	3	2	1	3	1	2	2	3	2	3	1	1
28	2	2	2	1	1	1	1	2	2	1	2	1	3	2	2	2	1	1	3	2	3	1	3
29	2	2	2	1	1	1	1	2	2	1	2	2	1	3	3	3	2	2	1	3	1	2	1
30	2	2	2	1	1	1	1	2	2	1	2	3	2	1	1	1	3	3	2	1	2	3	2
31	2	2	1	2	1	2	1	1	1	2	2	1	3	3	3	2	3	2	2	1	2	1	1
32	2	2	1	2	1	2	1	1	1	2	2	2	1	1	1	3	1	3	3	2	3	2	2
33	2	2	1	2	1	2	1	1	1	2	2	3	2	2	1	2	1	1	3	1	1	3	3
34	2	2	1	1	2	1	2	1	2	2	1	1	3	1	2	3	2	3	1	2	2	3	1
35	2	2	1	1	2	1	2	1	2	2	1	2	1	2	3	1	3	1	2	3	3	1	2
36	2	2	1	1	2	1	2	1	2	2	1	3	2	3	1	2	1	2	3	1	1	2	3

Advantages and Disadvantages of Three-Level and Mixed-Level "L" Designs

Advantages and disadvantages of three-level mixed-level designs

The good features of these designs are:

- They are orthogonal arrays. Some analysts believe this simplifies the analysis and interpretation of results while other analysts believe it does not.
- They obtain a lot of information about the main effects in a relatively few number of runs.
- You can test whether non-linear terms are needed in the model, at least as far as the three-level factors are concerned.

On the other hand, there are several undesirable features of these designs to consider:

- They provide limited information about interactions.
- They require more runs than a comparable 2^{k-p} design, and a two-level design will often suffice when the factors are continuous and monotonic (many three-level designs are used when two-level designs would have been adequate).

5.4. Analysis of DOE data

Contents of this section

Assuming you have a starting model that you want to fit to your experimental data and the experiment was designed correctly for your objective, most DOE software packages will analyze your DOE data. This section will illustrate how to analyze DOE's by first going over the generic basic steps and then showing software examples. The contents of the section are:

Prerequisite statistical tools and concepts needed for DOE analyses

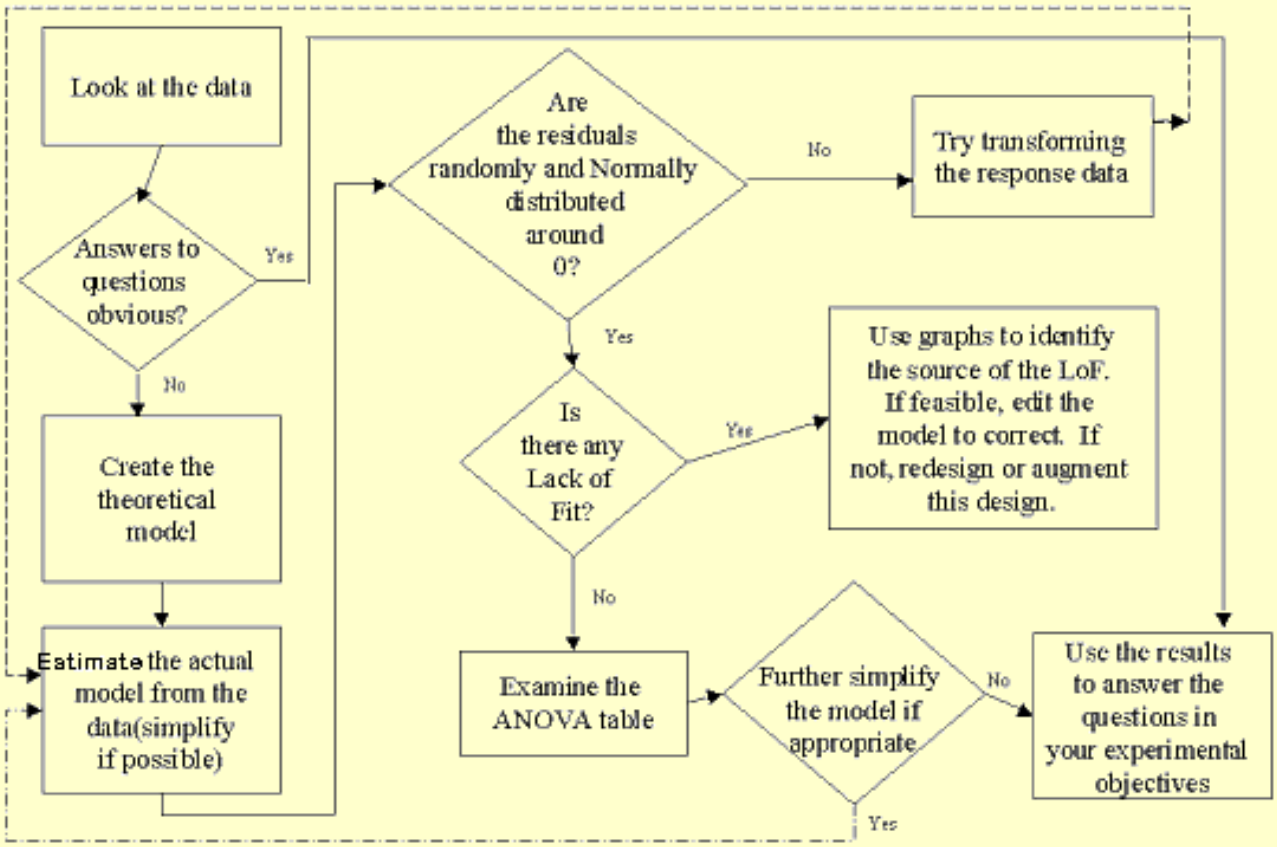
The examples in this section assume the reader is familiar with the concepts of

- ANOVA tables (see Chapter 3 or Chapter 7)
- p-values
- Residual analysis
- Model Lack of Fit tests
- Data transformations for normality and linearity

5.4.1. What are the steps in a DOE analysis?

General flowchart for analyzing DOE data

Flowchart of DOE Analysis Steps



DOE Analysis Steps

Analysis steps: graphics, theoretical model, actual model, validate model, use model

The following are the basic steps in a DOE analysis.

1. Look at the data. Examine it for outliers, typos and obvious problems. Construct as many graphs as you can to get the big picture.
   - Response distributions (histograms, box plots, etc.)
   - Responses versus time order scatter plot (a check for possible time effects)
   - Responses versus factor levels (first look at magnitude of factor effects)
   - Typical DOE plots (which assume standard models for effects and errors)
     - Main effects mean plots
     - Block plots
     - Normal or half-normal plots of the effects

        - Interaction plots
    - Sometimes the right graphs and plots of the data lead to obvious answers for your experimental objective questions and you can skip to step 5. In most cases, however, you will want to continue by fitting and validating a model that can be used to answer your questions.
2. Create the theoretical model (the experiment should have been designed with this model in mind!).
3. Create a model from the data. Simplify the model, if possible, using stepwise regression methods and/or parameter p-value significance information.
4. Test the model assumptions using residual graphs.
    - If none of the model assumptions were violated, examine the ANOVA.
        - Simplify the model further, if appropriate. If reduction is appropriate, then return to step 3 with a new model.
    - If model assumptions were violated, try to find a cause.
        - Are necessary terms missing from the model?
        - Will a transformation of the response help? If a transformation is used, return to step 3 with a new model.
5. Use the results to answer the questions in your experimental objectives -- finding important factors, finding optimum settings, etc.

Flowchart is a guideline, not a hard-and -fast rule

Note: The above flowchart and sequence of steps should not be regarded as a "hard-and-fast rule" for analyzing all DOE's. Different analysts may prefer a different sequence of steps and not all types of experiments can be analyzed with one set procedure. There still remains some *art* in both the design and the analysis of experiments, which can only be learned from experience. In addition, the role of engineering judgment should not be underestimated.

5.4.2. How to "look" at DOE data

The importance of looking at the data with a wide array of plots or visual displays cannot be over-stressed

The right graphs, plots or visual displays of a dataset can uncover anomalies or provide insights that go beyond what most quantitative techniques are capable of discovering. Indeed, in many cases quantitative techniques and models are tools used to confirm and extend the conclusions an analyst has already formulated after carefully "looking" at the data.

Most software packages have a selection of different kinds of plots for displaying DOE data. Dataplot, in particular, has a wide range of options for visualizing DOE (i.e., DEX) data. Some of these useful ways of looking at data are mentioned below, with links to detailed explanations in Chapter 1 (Exploratory Data Analysis or EDA) or to other places where they are illustrated and explained. In addition, examples and detailed explanations of visual (EDA) DOE techniques can be found in section 5.5.9.

Plots for viewing the response data

First "Look" at the Data

- Histogram of responses
- Run-sequence plot (pay special attention to results at center points)
- Scatter plot (for pairs of response variables)
- Lag plot
- Normal probability plot
- Autocorrelation plot

Plots for viewing main effects and 2-factor interactions, explanation of normal or half-normal plots to detect possible important effects

Subsequent Plots: Main Effects, Comparisons and 2-Way Interactions

- Quantile-quantile (q-q) plot
- Block plot
- Box plot
- Bi-histogram
- DEX scatter plot
- DEX mean plot
- DEX standard deviation plot
- DEX interaction plots
- Normal or half-normal probability plots for effects. **Note**: these links show how to generate plots to test for normal (or half-normal) data with points lining up along a straight line, approximately, if the plotted points were from the assumed normal (or half-normal) distribution. For two-level full factorial and fractional factorial experiments, the points plotted are the estimates of all the model effects, including possible interactions. Those effects that are really negligible should have estimates that resemble normally distributed noise, with mean zero and a constant variance. Significant effects can be picked out as the ones that do not line up along the straight line. Normal effect plots use the effect estimates directly, while half-normal plots use the absolute values of the effect estimates.
- Youden plots

Plots for testing and validating models

Model testing and Validation

- Response vs predictions
- Residuals vs predictions
- Residuals vs independent variables
- Residuals lag plot
- Residuals histogram
- Normal probability plot of residuals

Plots for model prediction

Model Predictions

- Contour plots

5.4.3. How to model DOE data

DOE models should be consistent with the goal of the experiment

In general, the trial model that will be fit to DOE data should be consistent with the goal of the experiment and has been predetermined by the goal of the experiment and the experimental design and data collection methodology.

Comparative designs

Models were given earlier for comparative designs (completely randomized designs, randomized block designs and Latin square designs).

Full factorial designs

For full factorial designs with k factors (2^k runs, not counting any center points or replication runs), the full model contains all the main effects and all orders of interaction terms. Usually, higher-order (three or more factors) interaction terms are included initially to construct the normal (or half-normal) plot of effects, but later dropped when a simpler, adequate model is fit. Depending on the software available or the analyst's preferences, various techniques such as normal or half-normal plots, Youden plots, p-value comparisons and stepwise regression routines are used to reduce the model to the minimum number of needed terms. A JMP example of model selection is shown later in this section and a Dataplot example is given as a case study.

Fractional factorial designs

For fractional factorial screening designs, it is necessary to know the alias structure in order to write an appropriate starting model containing only the interaction terms the experiment was designed to estimate (assuming all terms confounded with these selected terms are insignificant). This is illustrated by the JMP fractional factorial example later in this section. The starting model is then possibly reduced by the same techniques described above for full factorial models.

Response surface designs	Response surface initial models include quadratic terms and may occasionally also include cubic terms. These models were described in section 3.
Model validation	Of course, as in all cases of model fitting, residual analysis and other tests of model fit are used to confirm or adjust models, as needed.

5.4.4. How to test and revise DOE models

Tools for testing, revising, and selecting models

All the tools and procedures for testing, revising and selecting final DOE models are covered in various sections of the Handbook. The outline below gives many of the most common and useful techniques and has links to detailed explanations.

Outline of Model Testing and Revising: Tools and Procedures

An outline (with links) covers most of the useful tools and procedures for testing and revising DOE models

- Graphical Indicators for testing models (using residuals)
  - Response vs predictions
  - Residuals vs predictions
  - Residuals vs independent variables
  - Residuals lag plot
  - Residuals histogram
  - Normal probability plot of residuals
- Overall numerical indicators for testing models and model terms
  - R Squared and R Squared adjusted
  - Model Lack of Fit tests
  - ANOVA tables (see Chapter 3 or Chapter 7)
  - *p*-values
- Model selection tools or procedures
  - ANOVA tables (see Chapter 3 or Chapter 7)
  - *p*-values
  - Residual analysis
  - Model Lack of Fit tests
  - Data transformations for normality and linearity
  - Stepwise regression procedures

- Normal or half-normal plots of effects (primarily for two-level full and fractional factorial experiments)
- Youden plots
- Other methods

5.4.5. How to interpret DOE results

Final model used to make conclusions and decisions

Assume that you have a final model that has passed all the relevant tests (visual and quantitative) and you are ready to make conclusions and decisions. These should be responses to the questions or outputs dictated by the original experimental goals.

Checklist relating DOE conclusions or outputs to experimental goals or experimental purpose:

A checklist of how to compare DOE results to the experimental goals

- Do the responses differ significantly over the factor levels? (comparative experiment goal)
- Which are the significant effects or terms in the final model? (screening experiment goal)
- What is the model for estimating responses?
  - Full factorial case (main effects plus significant interactions)
  - Fractional factorial case (main effects plus significant interactions that are not confounded with other possibly real effects)
  - RSM case (allowing for quadratic or possibly cubic models, if needed)
- What responses are predicted and how can responses be optimized? (RSM goal)
  - Contour plots
  - JMP prediction profiler (or other software aids)
  - Settings for confirmation runs and prediction intervals for results

5.4.6. How to confirm DOE results (confirmatory runs)

Definition of confirmation runs

When the analysis of the experiment is complete, one must verify that the predictions are good. These are called confirmation runs.

The interpretation and conclusions from an experiment may include a "best" setting to use to meet the goals of the experiment. Even if this "best" setting were included in the design, you should run it again as part of the confirmation runs to make sure nothing has changed and that the response values are close to their predicted values. would get.

At least 3 confirmation runs should be planned

In an industrial setting, it is very desirable to have a stable process. Therefore, one should run more than one test at the "best" settings. A minimum of 3 runs should be conducted (allowing an estimate of variability at that setting).

If the time between actually running the experiment and conducting the confirmation runs is more than a few hours, the experimenter must be careful to ensure that nothing else has changed since the original data collection.

Carefully duplicate the original environment

The confirmation runs should be conducted in an environment as similar as possible to the original experiment. For example, if the experiment were conducted in the afternoon and the equipment has a warm-up effect, the confirmation runs should be conducted in the afternoon after the equipment has warmed up. Other extraneous factors that may change or affect the results of the confirmation runs are: person/operator on the equipment, temperature, humidity, machine parameters, raw materials, etc.

Checks for when confirmation runs give surprises

What do you do if you don't obtain the results you expected? If the confirmation runs don't produce the results you expected:

1. check to see that nothing has changed since the original data collection
2. verify that you have the correct settings for the confirmation runs
3. revisit the model to verify the "best" settings from the analysis
4. verify that you had the correct predicted value for the confirmation runs.

If you don't find the answer after checking the above 4 items, the model may not predict very well in the region you decided was "best". You still learned from the experiment and you should use the information gained from this experiment to design another follow-up experiment.

Even when the experimental goals are not met, something was learned that can be used in a follow-up experiment

Every well-designed experiment is a success in that you learn something from it. However, every experiment will not necessarily meet the goals established before experimentation. That is why it makes sense to plan to experiment sequentially in order to meet the goals.

5. Process Improvement
5.4. Analysis of DOE data

5.4.7. Examples of DOE's

Software packages do the calculations and plot the graphs for a DOE analysis: the analyst has to know what to request and how to interpret the results

Most DOE analyses of industrial experiments will be performed by statistical software packages. Good statistical software enables the analyst to view graphical displays and to build models and test assumptions. Occasionally the goals of the experiment can be achieved by simply examining appropriate graphical displays of the experimental responses. In other cases, a satisfactory model has to be fit in order to determine the most significant factors or the optimal contours of the response surface. In any case, the software will perform the appropriate calculations as long as the analyst knows what to request and how to interpret the program outputs.

Three detailed DOE analyses will be given using JMP software

Perhaps one of the best ways to learn how to use DOE analysis software to analyze the results of an experiment is to go through several detailed examples, explaining each step in the analysis. This section will illustrate the use of JMP 3.2.6 software to analyze three real experiments. Analysis using other software packages would generally proceed along similar paths.

The examples cover three basic types of DOE's:

1. A full factorial experiment
2. A fractional factorial experiment
3. A response surface experiment

5.4.7.1. Full factorial example

Data Source

This example uses data from a NIST high performance ceramics experiment

This data set was taken from an experiment that was performed a few years ago at NIST (by Said Jahanmir of the Ceramics Division in the Material Science and Engineering Laboratory). The original analysis was performed primarily by Lisa Gill of the Statistical Engineering Division. The example shown here is an independent analysis of a modified portion of the original data set.

The original data set was part of a high performance ceramics experiment with the goal of characterizing the effect of grinding parameters on sintered reaction-bonded silicon nitride, reaction bonded silicone nitride, and sintered silicon nitride.

Only modified data from the first of the 3 ceramic types (sintered reaction-bonded silicon nitride) will be discussed in this illustrative example of a full factorial data analysis.

The reader may want to download the data as a text file and try using other software packages to analyze the data.

Description of Experiment: Response and Factors

Response and factor variables used in the experiment

Purpose: To determine the effect of machining factors on ceramic strength
Response variable = mean (over 15 repetitions) of the ceramic strength
Number of observations = 32 (a complete 2^5 factorial design)

Response Variable Y = Mean (over 15 reps) of Ceramic Strength
Factor 1 = Table Speed (2 levels: slow (.025 m/s) and fast (.125 m/s))
Factor 2 = Down Feed Rate (2 levels: slow (.05 mm) and fast (.125 mm))
Factor 3 = Wheel Grit (2 levels: 140/170 and 80/100)
Factor 4 = Direction (2 levels: longitudinal and transverse)
Factor 5 = Batch (2 levels: 1 and 2)

Since two factors were qualitative (direction and batch) and it was reasonable to expect monotone effects from the quantitative factors, no centerpoint runs were included.

JMP spreadsheet of the data

The design matrix, with measured ceramic strength responses, appears below. The actual randomized run order is given in the last column. (The interested reader may download the data as a text file or as a JMP file.)

JMP - [fullfac]

File Edit Tables Rows Cols Analyze Graph Tools Window Help

32 Rows \ 7 Cols	X1: Table Speed	X2: Feed Rate	X3: Wheel Grit	X4: Direction	X5: Batch	Y: Strength	Actual Run Order
1	-1	-1	-1	-1	-1	680.45	17
2	1	-1	-1	-1	-1	722.48	30
3	-1	1	-1	-1	-1	702.14	14
4	1	1	-1	-1	-1	666.93	8
5	-1	-1	1	-1	-1	703.67	32
6	1	-1	1	-1	-1	642.14	20
7	-1	1	1	-1	-1	692.98	26
8	1	1	1	-1	-1	669.26	24
9	-1	-1	-1	1	-1	491.58	10
10	1	-1	-1	1	-1	475.52	16
11	-1	1	-1	1	-1	478.76	27
12	1	1	-1	1	-1	568.23	18
13	-1	-1	1	1	-1	444.72	3
14	1	-1	1	1	-1	410.37	19
15	-1	1	1	1	-1	428.51	31
16	1	1	1	1	-1	491.47	15
17	-1	-1	-1	-1	1	607.34	12
18	1	-1	-1	-1	1	620.8	1
19	-1	1	-1	-1	1	610.55	4
20	1	1	-1	-1	1	638.04	23
21	-1	-1	1	-1	1	585.19	2
22	1	-1	1	-1	1	586.17	28
23	-1	1	1	-1	1	601.67	11
24	1	1	1	-1	1	608.31	9
25	-1	-1	-1	1	1	442.9	25
26	1	-1	-1	1	1	434.41	21
27	-1	1	-1	1	1	417.66	6
28	1	1	-1	1	1	510.84	7
29	-1	-1	1	1	1	392.11	5
30	1	-1	1	1	1	343.22	13
31	-1	1	1	1	1	385.52	22
32	1	1	1	1	1	446.73	29

Analysis of the Experiment

Analysis follows 5 basic steps

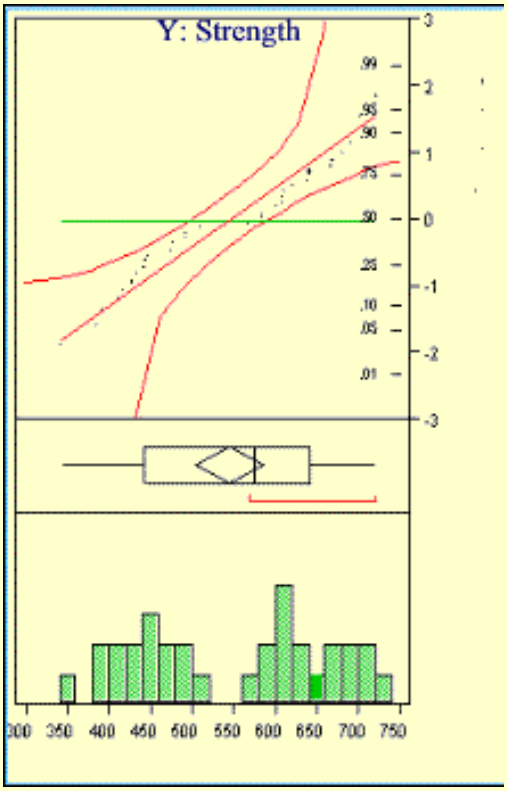
The experimental data will be analyzed following the previously described 5 basic steps using SAS JMP 3.2.6 software.

Step 1: Look at the data

Plot the response variable

We start by plotting the response data several ways to see if any trends or anomalies appear that would not be accounted for by the standard linear response models.

First we look at the distribution of all the responses irrespective of factor levels.



The following plots were generared:

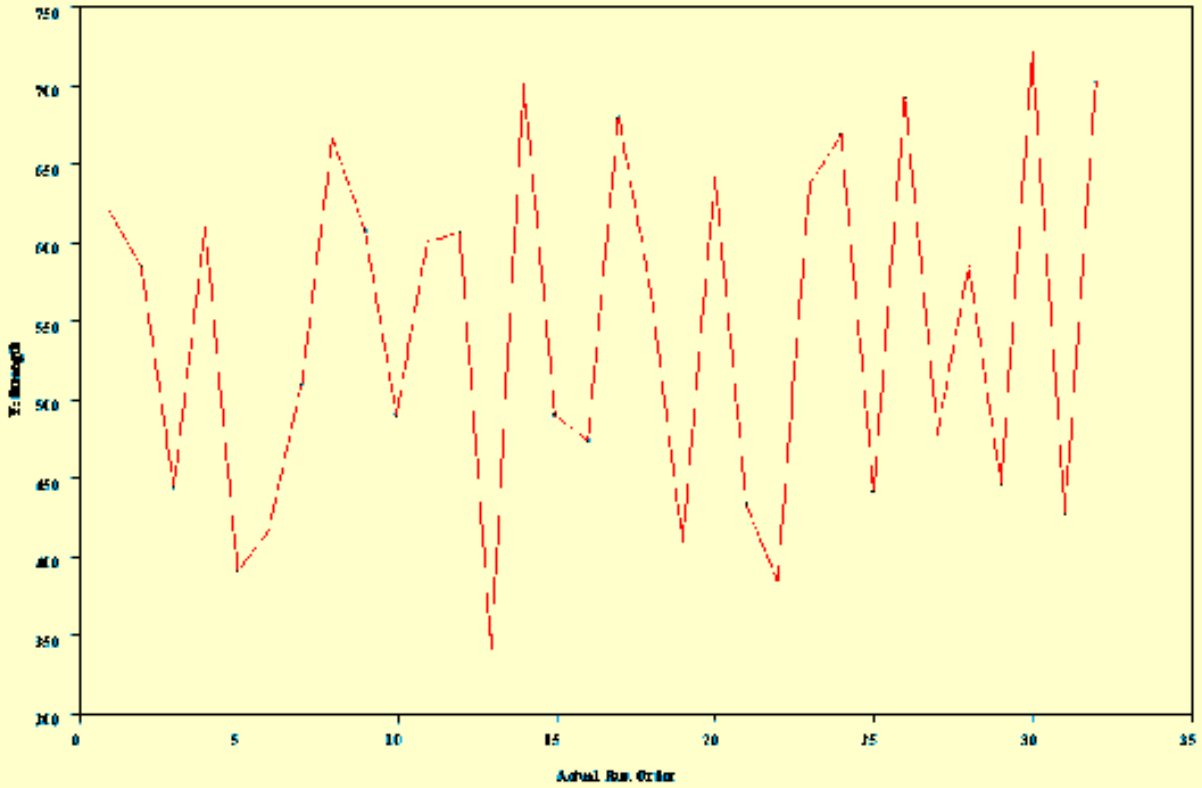
1. The first plot is a normal probability plot of the response variable.
2. The second plot is a box plot of the response variable.
3. The third plot is a histogram of the response variable.

Clearly there is "structure" that we hope to account for when we fit a response model. For example, note the separation of the response into two roughly equal-sized clumps in the histogram. The first clump is centered approximately around the value 450 while the second clump is centered approximately around the value 650.

Plot of response versus run order

Next we look at the responses plotted versus run order to check whether there might be a time sequence component affecting the response levels.

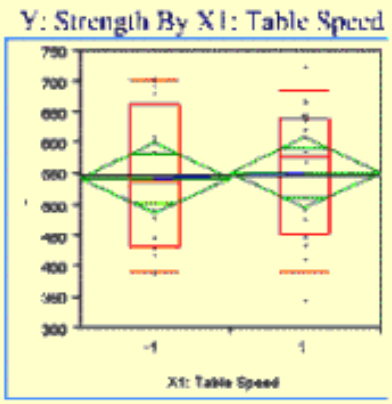
Plot of Response Vs. Run Order

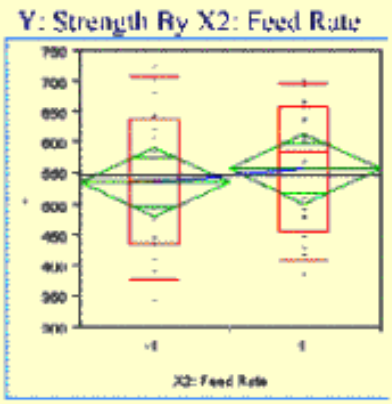


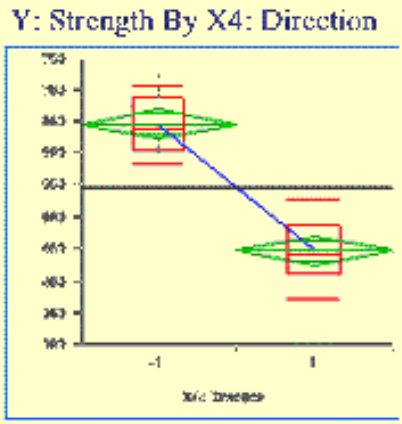
As hoped for, this plot does not indicate that time order had much to do with the response levels.

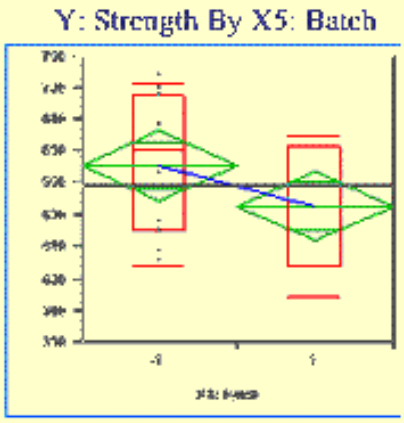
Box plots of response by factor variables

Next, we look at plots of the responses sorted by factor columns.









Several factors, most notably "Direction" followed by "Batch" and possibly "Wheel Grit", appear to change the average response level.

Step 2: Create the theoretical model

Theoretical model: assume all third-order and higher interaction terms are not significant

With a 2^5 full factorial experiment we can fit a model containing a mean term, all 5 main effect terms, all 10 first-order interaction terms, all 10 second-order interaction terms, all 5 third-order interaction terms and the fourth-order interaction term (32 parameters). However, we start by assuming all third-order and higher interaction terms are non-existent (it's very rare for such high-order interactions to be significant, and they are very difficult to interpret from an engineering viewpoint). That allows us to accumulate the sums of squares for these terms and use them to estimate an error term. So we start out with a theoretical model with 26 unknown constants, hoping the data will clarify which of these are the significant main effects and interactions we need for a final model.

Step 3: Create the actual model from the data

Output from fitting up to third-order interaction terms

After fitting the 26 parameter model, the following analysis table is displayed:

Output after Fitting Third Order Model to Response Data

Response: Y: Strength

Summary of Fit

RSquare	0.995127
RSquare Adj	0.974821
Root Mean Square Error	17.81632
Mean of Response	546.8959
Observations	32

Effect Test

Source	DF	Sum of Squares	F Ratio	Prob>F
X1: Table Speed	1	894.33	2.8175	0.1442
X2: Feed Rate	1	3497.20	11.0175	0.0160
X1: Table Speed* X2: Feed Rate	1	4872.57	15.3505	0.0078
X3: Wheel Grit	1	12663.96	39.8964	0.0007
X1: Table Speed* X3: Wheel Grit	1	1838.76	5.7928	0.0528
X2: Feed Rate* X3: Wheel Grit	1	307.46	0.9686	0.3630
X1:Table Speed* X2: Feed Rate* X3: Wheel Grit	1	357.05	1.1248	0.3297
X4: Direction	1	315132.65	992.7901	<.0001
X1: Table Speed* X4: Direction	1	1637.21	5.1578	0.0636
X2: Feed Rate* X4: Direction	1	1972.71	6.2148	0.0470
X1: Table Speed X2: Feed Rate* X4: Direction	1	5895.62	18.5735	0.0050
X3: Wheel Grit*	1	3158.34	9.9500	0.0197

```
    X4: Direction
X1: Table Speed*        1         2.12     0.0067     0.9376
    X3: Wheel Grit*
    X4: Direction
X2: Feed Rate*          1        44.49     0.1401     0.7210
    X3: Wheel Grit*
    X4: Direction
X5: Batch               1     33653.91   106.0229     <.0001
X1: Table Speed*        1       465.05     1.4651     0.2716
    X5: Batch
X2: Feed Rate*          1       199.15     0.6274     0.4585
    X5: Batch
X1: Table Speed*        1       144.71     0.4559     0.5247
    X2: Feed Rate*
    X5: Batch
X3: Wheel Grit*         1        29.36     0.0925     0.7713
    X5: Batch
X1: Table Speed*        1        30.36     0.0957     0.7676
    X3: Wheel Grit*
    X5: Batch
X2: Feed Rate*          1        25.58     0.0806     0.7860
    X3: Wheel Grit*
    X5: Batch
X4: Direction *         1      1328.83     4.1863     0.0867
    X5: Batch
X1: Table Speed*        1       544.58     1.7156     0.2382
    X4: Directio*
    X5: Batch
X2: Feed Rate*          1       167.31     0.5271     0.4952
    X4: Direction*
    X5: Batch
X3: Wheel Grit*         1        32.46     0.1023     0.7600
    X4: Direction*
    X5: Batch
```

This fit has a high R^2 and adjusted R^2, but the large number of high (>0.10) p-values (in the "Prob>F" column) make it clear that the model has many unnecessary terms.

JMP stepwise regression

Starting with these 26 terms, we next use the JMP Stepwise Regression option to eliminate unnecessary terms. By a combination of stepwise regression and the removal of remaining terms with a *p*-value higher than 0.05, we quickly arrive at a model with an intercept and 12 significant effect terms.

Output from fitting the 12-term model

Output after Fitting the 12-Term Model to Response Data

Response: Y: Strength

Summary of Fit

RSquare	0.989114
RSquare Adj	0.982239
Root Mean Square Error	14.96346
Mean of Response	546.8959
Observations (or Sum Wgts)	32

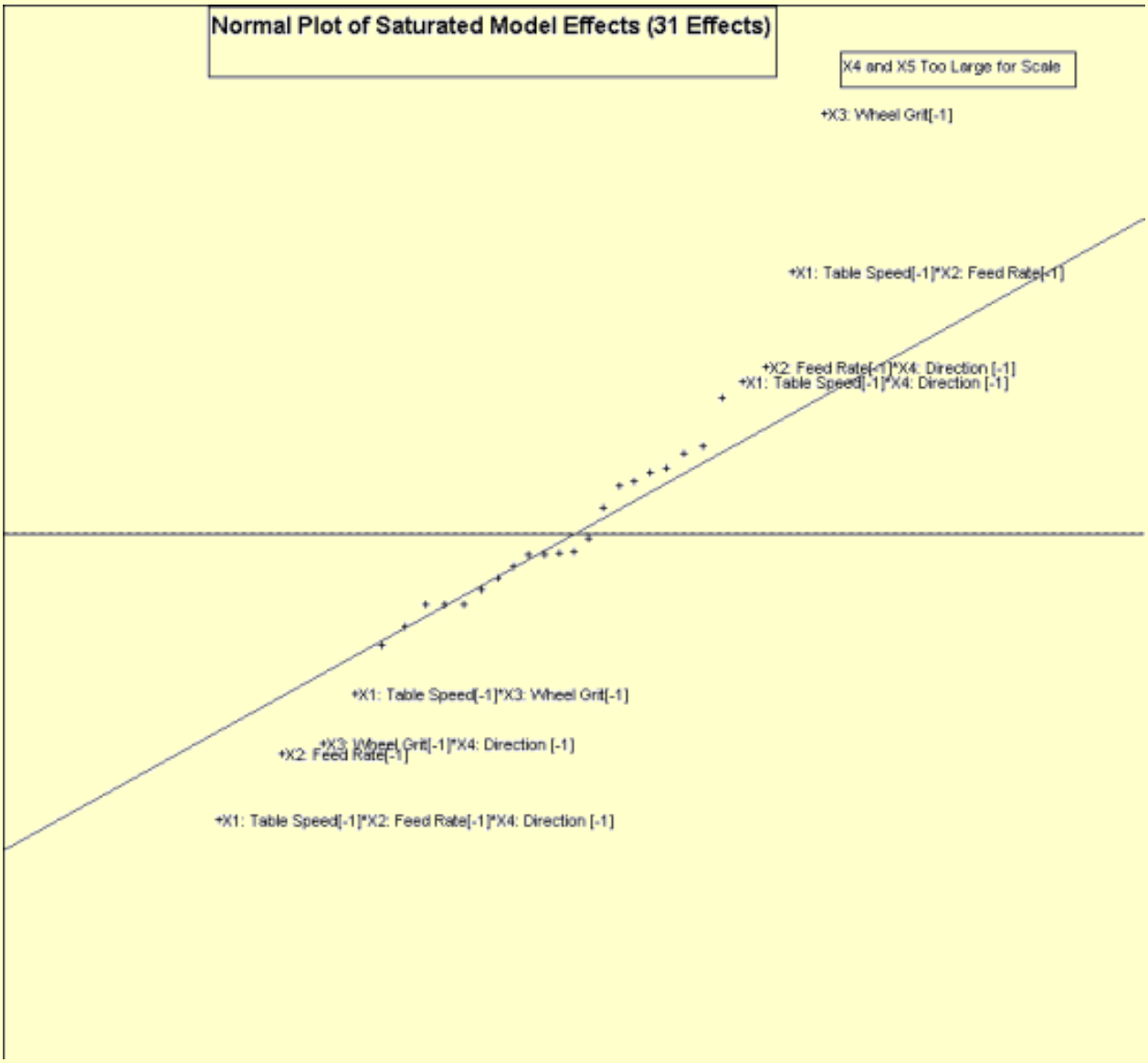
Effect Test

Source	DF	Sum of Squares	F Ratio	Prob>F
X1: Table Speed	1	894.33	3.9942	0.0602
X2: Feed Rate	1	3497.20	15.6191	0.0009
X1: Table Speed* X2: Feed Rate	1	4872.57	21.7618	0.0002
X3: Wheel Grit	1	12663.96	56.5595	<.0001
X1: Table Speed* X3: Wheel Grit	1	1838.76	8.2122	0.0099
X4: Direction	1	315132.65	1407.4390	<.0001
X1: Table Speed* X4: Direction	1	1637.21	7.3121	0.0141
X2: Feed Rate* X4: Direction	1	1972.71	8.8105	0.0079
X1: Table Speed* X2: Feed Rate* X4:Direction	1	5895.62	26.3309	<.0001
X3: Wheel Grit* X4: Direction	1	3158.34	14.1057	0.0013
X5: Batch	1	33653.91	150.3044	<.0001
X4: Direction* X5: Batch	1	1328.83	5.9348	0.0249

Normal plot of the effects

Non-significant effects should effectively follow an approximately normal distribution with the same location and scale. Significant effects will vary from this normal distribution. Therefore, another method of determining significant effects is to generate a normal plot of all 31 effects. Those effects that are substantially away from the straight line fitted to the normal plot are considered significant. Although this is a somewhat subjective criteria, it tends to work well in practice. It is helpful to use both the numerical output from the fit and graphical techniques such as the normal plot in deciding which terms to keep in the model.

The normal plot of the effects is shown below. We have labeled those effects that we consider to be significant. In this case, we have arrived at the exact same 12 terms by looking at the normal plot as we did from the stepwise regression.



Most of the effects cluster close to the center (zero) line and follow the fitted normal model straight line. The effects that appear to be above or below the line by more than a small amount are the same effects identified using the stepwise routine, with the exception of *X*1. Some analysts prefer to include a main effect term when it has several significant interactions even if the main effect term itself does not appear to be significant.

Model appears to account for most of the variability

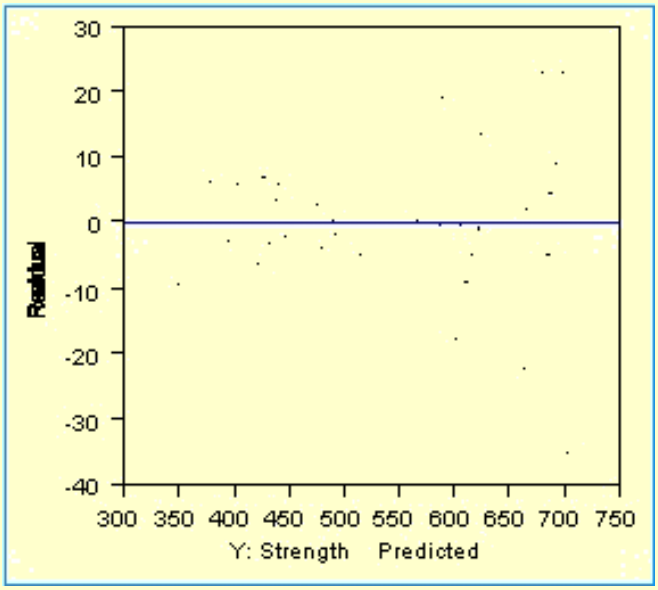
At this stage, this model appears to account for most of the variability in the response, achieving an adjusted R^2 of 0.982. All the main effects are significant, as are 6 first-order interactions and 1 second-order interaction. The only interaction that makes little physical sense is the " *X*4: Direction**X*5: Batch" interaction - why would the response using one batch of material react differently when the batch is cut in a different direction as compared to another batch of the same formulation?

However, before accepting any model, residuals need to be examined.

Step 4: Test the model assumptions using residual graphs (adjust and simplify as needed)

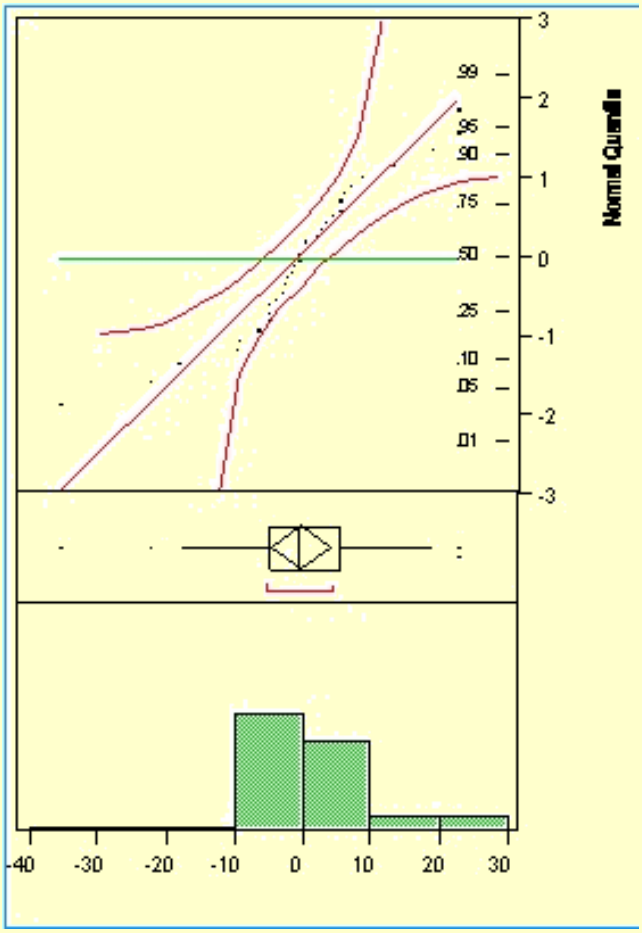
Plot of residuals versus predicted responses

First we look at the residuals plotted versus the predicted responses.



The residuals appear to spread out more with larger values of predicted strength, which should not happen when there is a common variance.

Next we examine the normality of the residuals with a normal quantile plot, a box plot and a histogram.



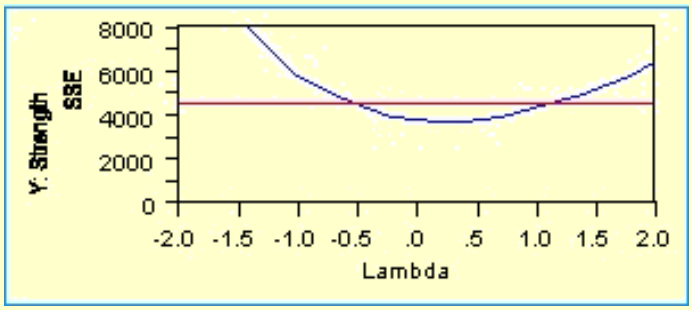
None of these plots appear to show typical normal residuals and 4 of the 32 data points appear as outliers in the box plot.

Step 4 continued: Transform the data and fit the model again

Box-Cox Transformation

We next look at whether we can model a transformation of the response variable and obtain residuals with the assumed properties. JMP calculates an optimum Box-Cox transformation by finding the value of λ that minimizes the model SSE. Note: the Box-Cox transformation used in JMP is different from the transformation used in Dataplot, but roughly equivalent.

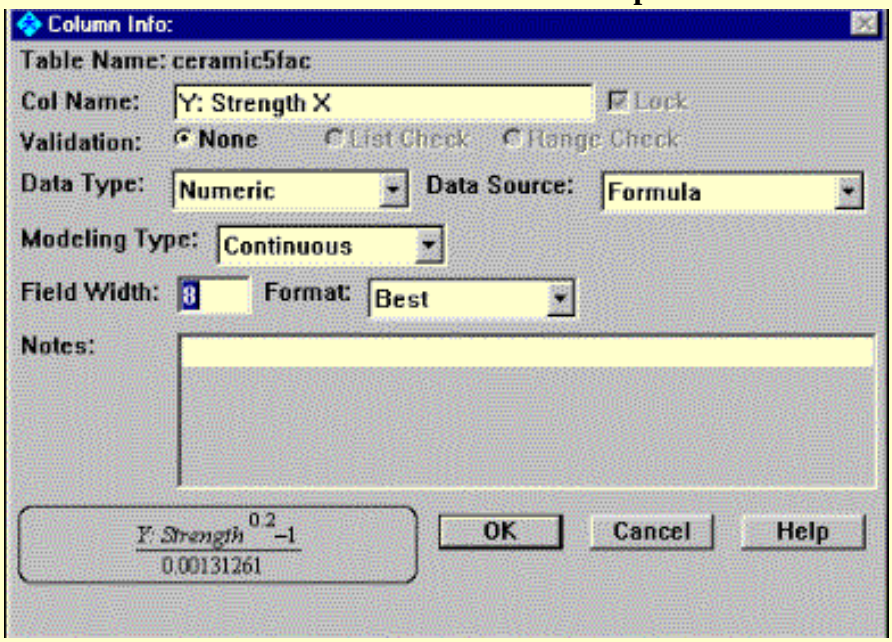
Box-Cox Transformation Graph



The optimum is found at λ = 0.2. A new column Y: Strength X is calculated and added to the JMP data spreadsheet. The properties of this column, showing the transformation equation, are shown below.

JMP data transformation menu

Data Transformation Column Properties



Fit model to transformed data

When the 12-effect model is fit to the transformed data, the "X4: Direction*X5: Batch" interaction term is no longer significant. The 11-effect model fit is shown below, with parameter estimates and *p*-values.

JMP output for fitted model after applying Box-Cox transformation

Output after Fitting the 11-Effect Model to Tranformed Response Data

Response: Y: Strength X

Summary of Fit

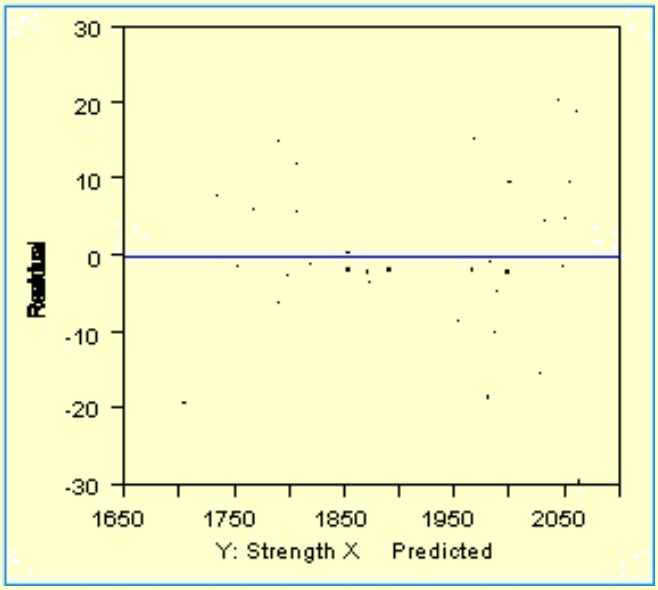
RSquare	0.99041
RSquare Adj	0.985135
Root Mean Square Error	13.81065
Mean of Response	1917.115
Observations (or Sum Wgts)	32

Effect	Parameter Estimate	p-value
Intercept	1917.115	<.0001
X1: Table Speed	5.777	0.0282
X2: Feed Rate	11.691	0.0001
X1: Table Speed* X2: Feed Rate	-14.467	<.0001
X3: Wheel Grit	-21.649	<.0001
X1: Table Speed* X3: Wheel Grit	7.339	0.007
X4: Direction	-99.272	<.0001
X1: Table Speed* X4: Direction	-7.188	0.0080
X2: Feed Rate* X4: Direction	-9.160	0.0013
X1: Table Speed* X2: Feed Rate* X4:Direction	15.325	<.0001
X3: Wheel Grit* X4: Direction	12.965	<.0001
X5: Batch	-31.871	<.0001

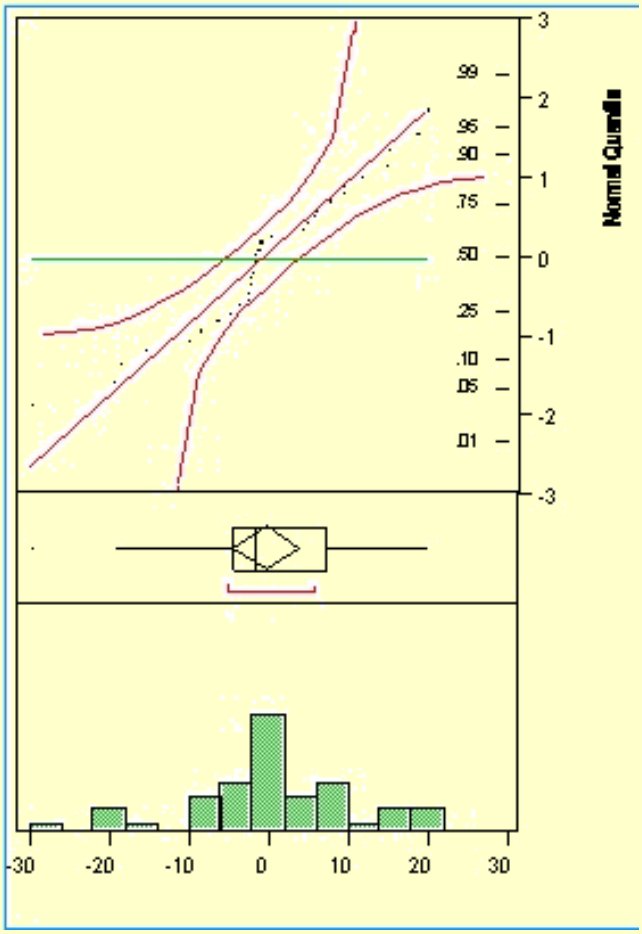
Model has high R^2

This model has a very high R^2 and adjusted R^2. The residual plots (shown below) are quite a bit better behaved than before, and pass the Wilk-Shapiro test for normality.

Residual plots from model with transformed response



The run sequence plot of the residuals does not indicate any time dependent patterns.



The normal probability plot, box plot, and the histogram of the residuals do not indicate any serious violations of the model assumptions.

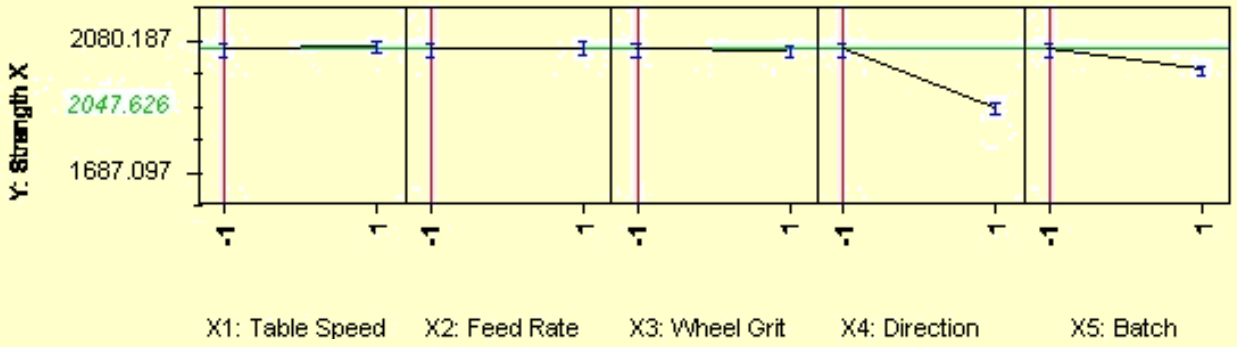
Step 5. Answer the questions in your experimental objectives

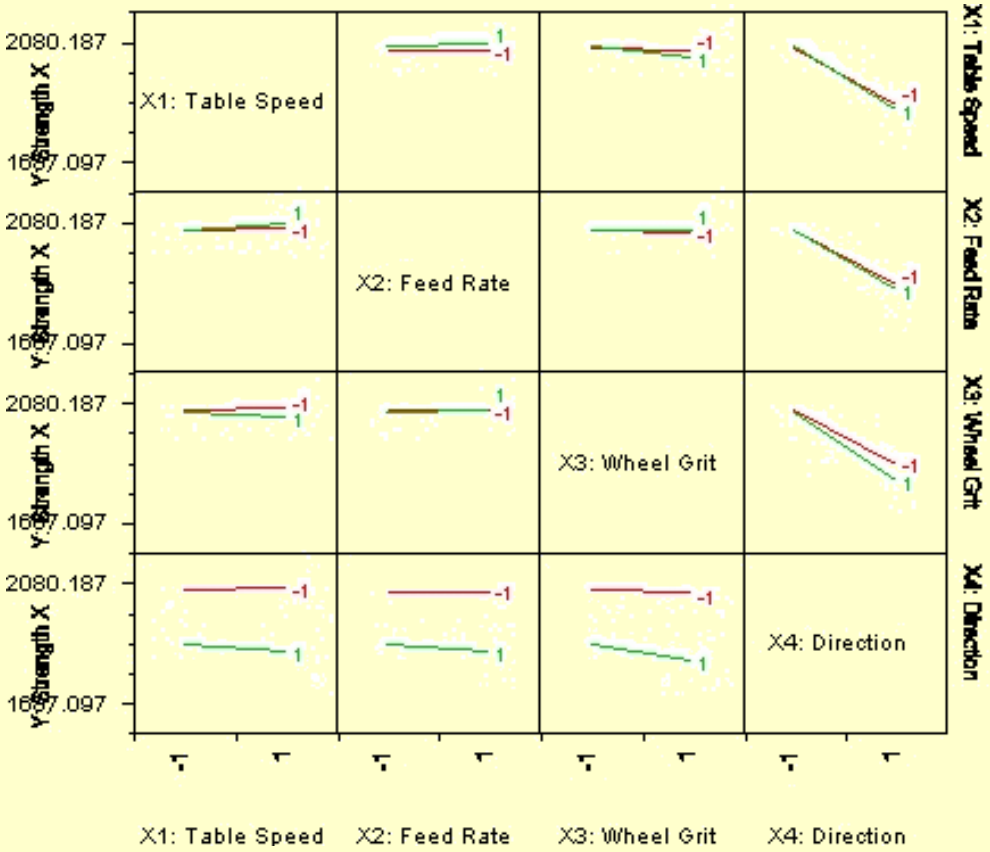
Important main effects and interaction effects

The magnitudes of the effect estimates show that "Direction" is by far the most important factor. "Batch" plays the next most critical role, followed by "Wheel Grit". Then, there are several important interactions followed by "Feed Rate". "Table Speed" plays a role in almost every significant interaction term, but is the least important main effect on its own. Note that large interactions can obscure main effects.

Plots of the main effects and significant 2-way interactions

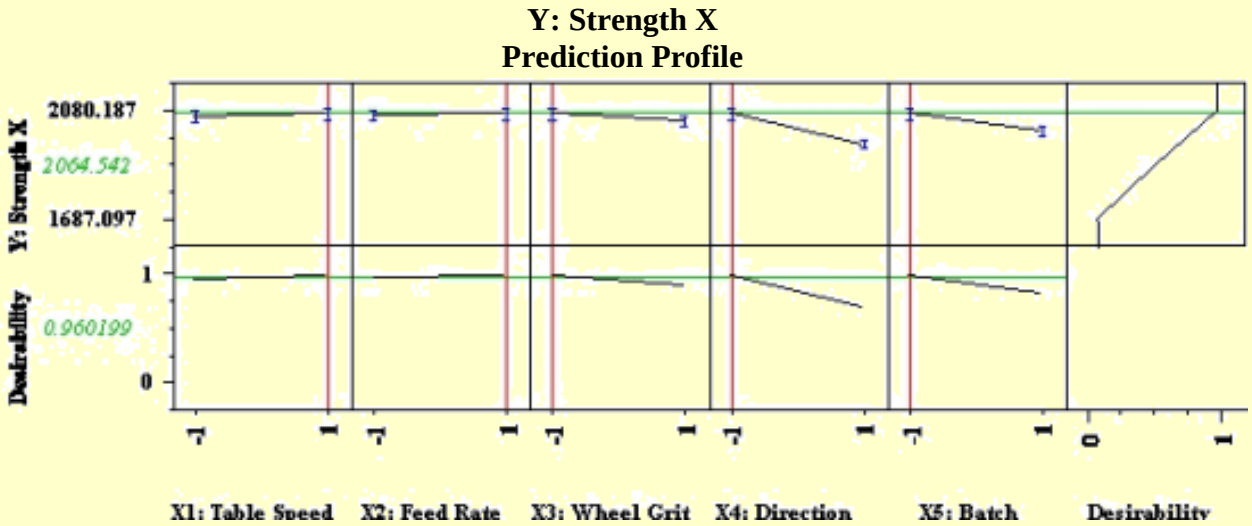
Plots of the main effects and the significant 2-way interactions are shown below.





Prediction profile

To determine the best setting to use for maximum ceramic strength, JMP has the "Prediction Profile" option shown below.



The vertical lines indicate the optimal factor settings to maximize the (transformed) strength response. Translating from -1 and +1 back to the actual factor settings, we have: Table speed at "1" or .125m/s; Down Feed Rate at "1" or .125 mm; Wheel Grit at "-1" or 140/170 and Direction at "-1" or longitudinal.

Unfortunately, "Batch" is also a very significant factor, with the first batch giving higher strengths than the second. Unless it is possible to learn what worked well with this batch, and how to repeat it, not much can be done about this factor.

Comments

Analyses with value of Direction fixed indicates complex model is needed only for transverse cut

1. One might ask what an analysis of just the 2^4 factorial with "Direction" kept at -1 (i.e., longitudinal) would yield. This analysis turns out to have a very simple model; only "Wheel Grit" and "Batch" are significant main effects and no interactions are significant.

   If, on the other hand, we do an analysis of the 2^4 factorial with "Direction" kept at +1 (i.e., transverse), then we obtain a 7-parameter model with all the main effects and interactions we saw in the 2^5 analysis, except, of course, any terms involving "Direction".

   So it appears that the complex model of the full analysis came from the physical properties of a transverse cut, and these complexities are not present for longitudinal cuts.

Half fraction design

2. If we had assumed that three-factor and higher interactions were negligible before experimenting, a 2_V^{5-1} half fraction design might have been chosen. In hindsight, we would have obtained valid estimates for all main effects and two-factor interactions except for $X3$ and $X5$, which would have been aliased with $X1*X2*X4$ in that half fraction.

Natural log transformation

3. Finally, we note that many analysts might prefer to adopt a natural logarithm transformation (i.e., use ln Y) as the response instead of using a Box-Cox transformation with an exponent of 0.2. The natural logarithm transformation corresponds to an exponent of $\lambda = 0$ in the Box-Cox graph.

5.4.7.2. Fractional factorial example

A step-by-step analysis of a fractional factorial "catapult" experiment

A "Catapult" Fractional Factorial Experiment

This experiment was conducted by a team of students on a *catapult* – a table-top wooden device used to teach design of experiments and statistical process control. The catapult has several controllable factors and a response easily measured in a classroom setting. It has been used for over 10 years in hundreds of classes. Below is a small picture of a catapult that can be opened to view a larger version.

Catapult

Description of Experiment: Response and Factors

The experiment has five factors that might affect the distance the golf ball travels

Purpose: To determine the significant factors that affect the distance the ball is thrown by the catapult, and to determine the settings required to reach 3 different distances (30, 60 and 90 inches).

Response Variable: The distance in inches from the front of the catapult to the spot where the ball lands. The ball is a plastic golf ball.

Number of observations: 20 (a 2^{5-1} resolution V design with 4 center points).

Variables:

1. Response Variable Y = distance
2. Factor 1 = band height (height of the pivot point for the rubber bands – levels were 2.25 and 4.75 inches with a centerpoint level of 3.5)
3. Factor 2 = start angle (location of the arm when the operator releases– starts the forward motion of the arm – levels were 0 and 20 degrees with a centerpoint level of 10 degrees)
4. Factor 3 = rubber bands (number of rubber bands used on the catapult– levels were 1 and 2 bands)
5. Factor 4 = arm length (distance the arm is extended – levels were 0 and 4 inches with a centerpoint level of 2 inches)
6. Factor 5 = stop angle (location of the arm where the forward motion of the arm is stopped and the ball starts flying – levels were 45 and 80 degrees with a centerpoint level of 62 degrees)

Design matrix and responses (in run order)

The design matrix appears below in (randomized) run order.

catapult results

18 Cols / 20 Rows

	Pattern	band height	start angle	rbands	arm length	stop angle	Cent...	Y: distance	run order	bh
1	----+	3.25	0	1	0	80	0	28	1	
2	00000	4	10	2	2	62	1	99	2	
3	+++++	4.75	20	2	4	80	0	126.5	3	
4	+-++-	4.75	0	2	4	45	0	126.5	4	
5	-+++-	3.25	20	2	4	45	0	45	5	
6	+---	4.75	0	1	0	45	0	35	6	
7	00000	4	10	1	2	62	1	45	7	
8	++-+	4.75	20	1	0	80	0	28.25	8	
9	+-++	4.75	0	1	4	80	0	85	9	
10	-+--	3.25	20	1	0	45	0	8	10	
11	++-+-	4.75	20	1	4	45	0	36.5	11	
12	---+-	3.25	0	1	4	45	0	33	12	
13	00000	4	10	2	2	62	1	84.5	13	
14	+++--	4.75	20	2	0	45	0	28.5	14	
15	--+--	3.25	0	2	0	45	0	33.5	15	
16	-++-+	3.25	20	2	0	80	0	36	16	
17	+-+-+	4.75	0	2	0	80	0	84	17	
18	-+-++	3.25	20	1	4	80	0	45	18	
19	00000	4	10	1	2	62	1	37.5	19	
20	--+++	3.25	0	2	4	80	0	106	20	

0 \ 0 Selected

You can download the data in a spreadsheet

Readers who want to analyze this experiment may download an Excel spreadsheet catapult.xls or a JMP spreadsheet capapult.jmp.

One discrete factor

Note that 4 of the factors are *continuous*, and one – number of rubber bands – is *discrete*. Due to the prescence of this discrete factor, we actually have two different centerpoints, each with two runs. Runs 7 and 19 are with one rubber band, and the center of the other factors, while runs 2 and 13 are with two rubber bands and the center of the other factors.

5 confirmatory runs

After analyzing the 20 runs and determining factor settings needed to achieve predicted distances of 30, 60 and 90 inches, the team was asked to conduct 5 confirmatory runs at each of the derived settings.

Analysis of the Experiment

Analyze with JMP software

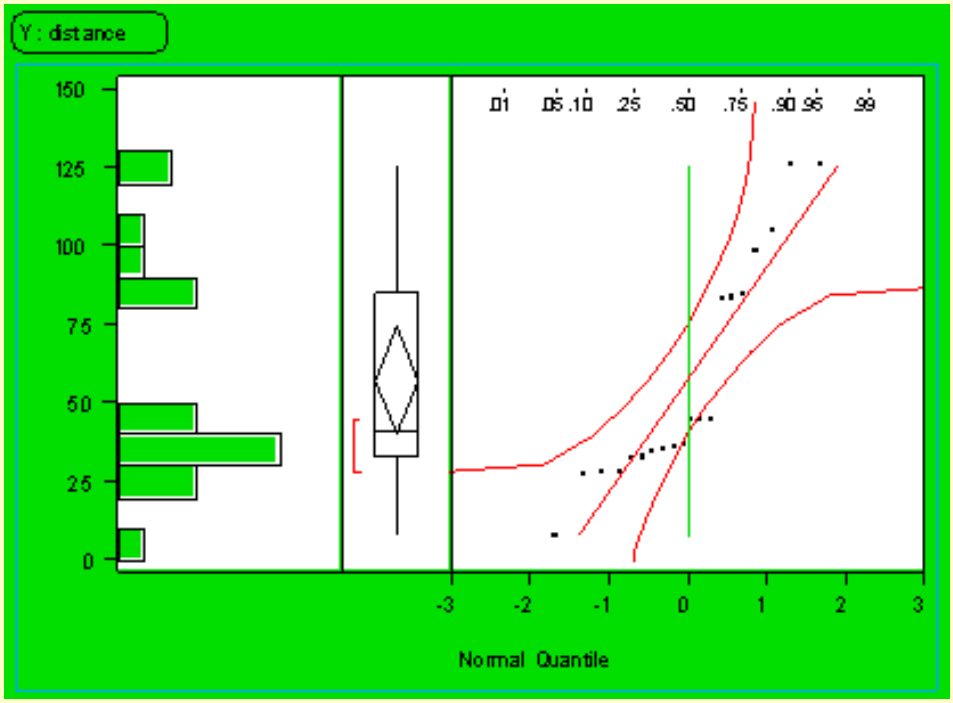
The experimental data will be analyzed using SAS JMP 3.2.6 software.

Step 1: Look at the data

Histogram, box plot, and normal probability plot of the response

We start by plotting the data several ways to see if any trends or anomalies appear that would not be accounted for by the models.

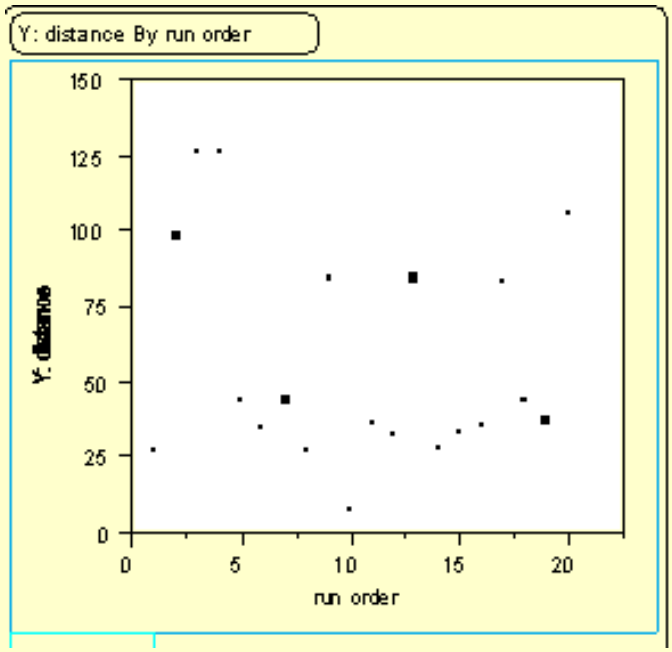
The distribution of the response is given below:



We can see the large spread of the data and a pattern to the data that should be explained by the analysis.

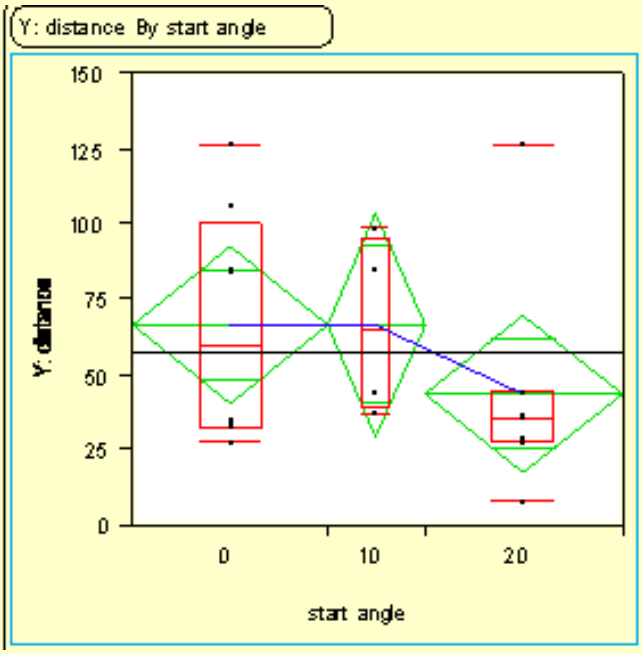
Plot of response versus run order

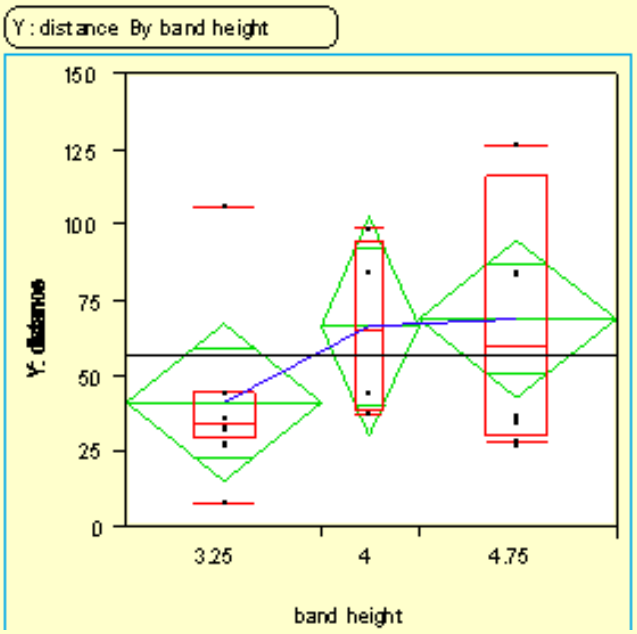
Next we look at the responses versus the run order to see if there might be a time sequence component. The four highlighted points are the center points in the design. Recall that runs 2 and 13 had 2 rubber bands and runs 7 and 19 had 1 rubber band. There may be a slight aging of the rubber bands in that the second center point resulted in a distance that was a little shorter than the first for each pair.

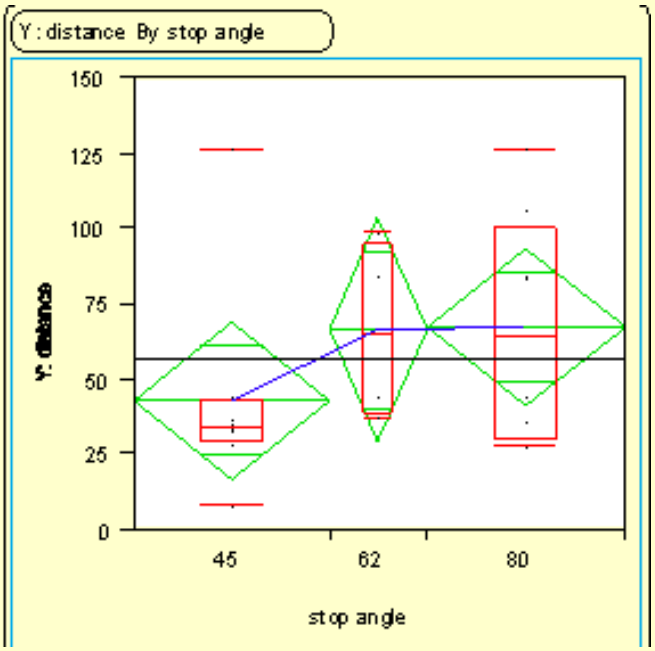


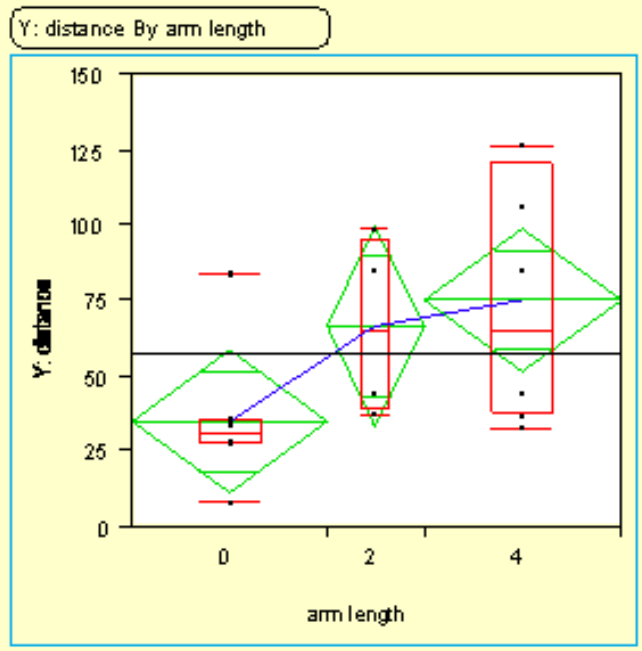
Plots of responses versus factor columns

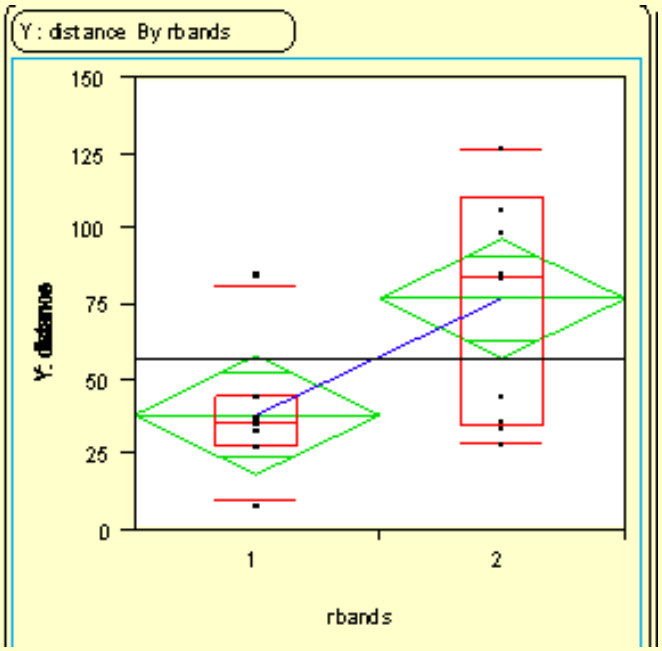
Next look at the plots of responses sorted by factor columns.






Y: distance By stop angle
150
125
100
75
50
25
0
Y: distance
45
62
80
stop angle


Y: distance By arm length
150
125
100
75
50
25
0
Y: distance
0
2
4
arm length



Several factors appear to change the average response level and most have a large spread at each of the levels.

Step 2: Create the theoretical model

The resolution V design can estimate main effects and all 2-factor interactions

With a resolution V design we are able to estimate all the main effects and all two-factor interactions cleanly – without worrying about confounding. Therefore, the initial model will have 16 terms – the intercept term, the 5 main effects, and the 10 two-factor interactions.

Step 3: Create the actual model from the data

Variable coding

Note we have used the orthogonally coded columns for the analysis, and have abbreviated the factor names as follows:

Bheight = band height
Start = start angle
Bands = number of rubber bands
Stop = stop angle
Arm = arm length.

JMP output after fitting the trial model (all main factors and 2-factor interactions)

The following is the JMP output after fitting the trial model (all main factors and 2-factor interactions).

Summary of Fit

RSquare	0.970924
R Square Adj	0.861887
Root Mean Square Error	13.2784
Mean of Response	57.5375
Observations (or Sum Wgts)	20

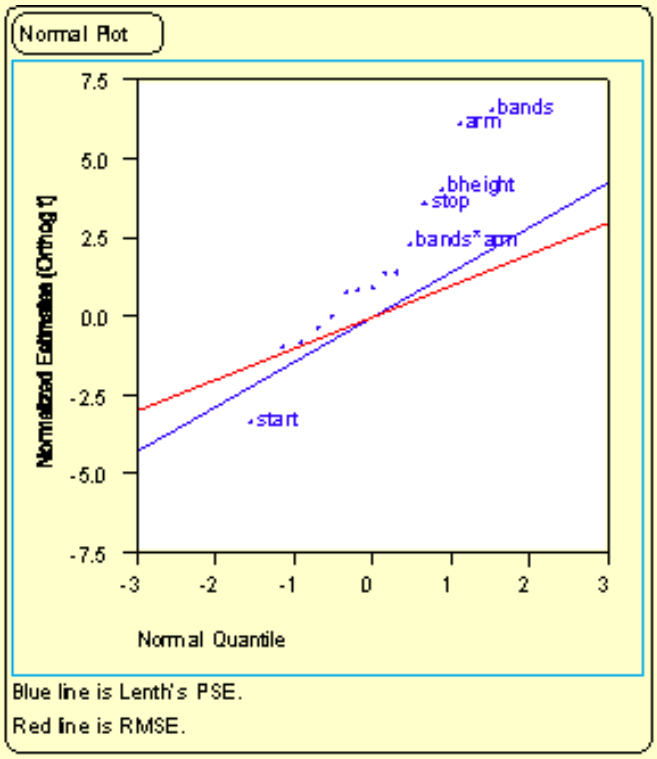
Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	57.5375	2.969141	19.38	<.0001
bheight	13.484375	3.319601	4.06	0.0153
start	-11.07813	3.319601	-3.34	0.0289
bheight*start	-2.765625	3.319601	-0.83	0.4516
bands	19.4125	2.969141	6.54	0.0028
bheight*bands	4.640625	3.319601	1.40	0.2347
start*bands	-3.171875	3.319601	-0.96	0.3934
arm	20.140625	3.319601	6.07	0.0037
bheight*arm	4.703125	3.319601	1.42	0.2295
start*arm	-1.109375	3.319601	-0.33	0.7550
bands*arm	7.609375	3.319601	2.29	0.0836
stop	12.046875	3.319601	3.63	0.0222
bheight*stop	0.109375	3.319601	0.03	0.9753
start*stop	2.671875	3.319601	0.80	0.4660
bands*stop	2.828125	3.319601	0.85	0.4422
arm*stop	3.140625	3.319601	0.95	0.3977

Use p-values to help select significant effects, and also use a normal plot

The model has a good R^2 value, but the fact that R^2 adjusted is considerably smaller indicates that we undoubtedly have some terms in our model that are not significant. Scanning the column of *p*-values (labeled Prob>|t| in the JMP output) for small values shows 5 significant effects at the 0.05 level and another one at the 0.10 level.

The normal plot of effects is a useful graphical tool to determine significant effects. The graph below shows that there are 9 terms in the model that can be assumed to be noise. That would leave 6 terms to be included in the model. Whereas the output above shows a *p*-value of 0.0836 for the interaction of bands and arm, the normal plot suggests we treat this interaction as significant.



A refit using just the effects that appear to matter

Remove the non-significant terms from the model and refit to produce the following output:

Y : distance

Summary of Fit

RSquare	0.913134
RSquare Adj	0.873042
Root Mean Square Error	12.7309
Mean of Response	57.5375
Observations (or Sum Wgts)	20

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	6	22148.548	3691.42	22.7759
Error	13	2106.987	162.08	Prob>F
C Total	19	24255.534		<.0001

Lack of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack of Fit	11	1973.7367	179.431	2.6931
Pure Error	2	133.2500	66.625	Prob>F
Total Error	13	2106.9867		0.3018
				Max RSq
				0.9945

R^2 is OK and there is no significant model "lack of fit"

The R^2 and R^2 adjusted values are acceptable. The ANOVA table shows us that the model is significant, and the Lack of Fit table shows that there is no significant lack of fit.

The Parameter estimates table is below.

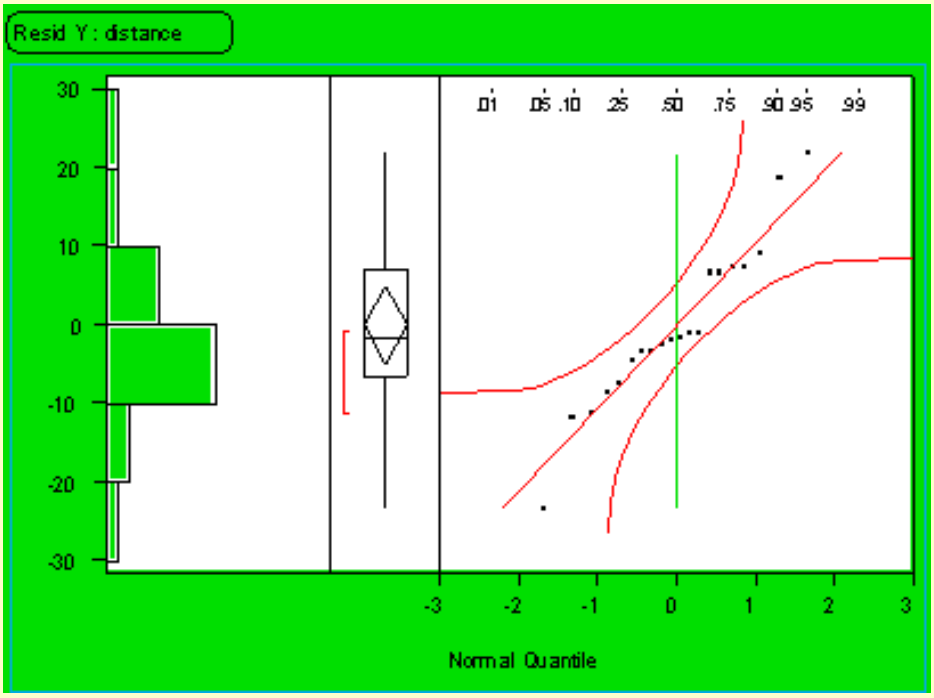
Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	57.5375	2.846717	20.21	<.0001
bheight	13.484375	3.182726	4.24	0.0010
start	-11.07813	3.182726	-3.48	0.0041
bands	19.4125	2.846717	6.82	<.0001
arm	20.140625	3.182726	6.33	<.0001
bands*arm	7.609375	3.182726	2.39	0.0326
stop	12.046875	3.182726	3.79	0.0023

Step 4: Test the model assumptions using residual graphs (adjust and simplify as needed)

Histogram of the residuals to test the model assumptions

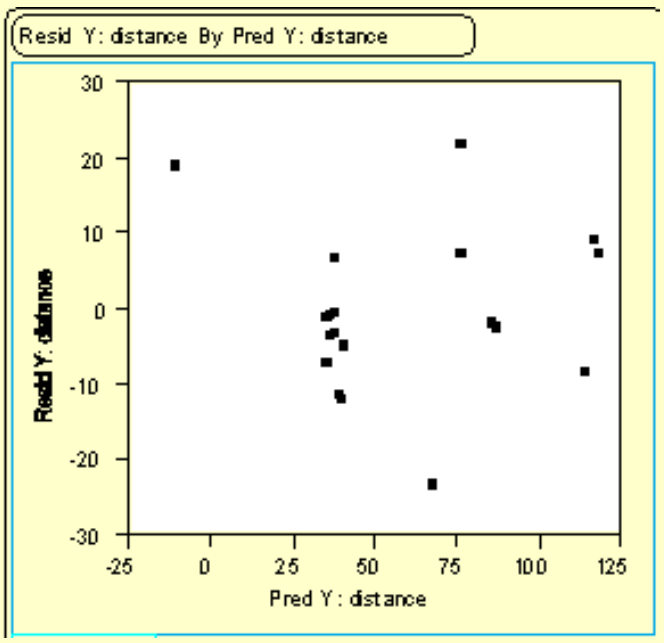
We should test that the residuals are approximately normally distributed, are independent, and have equal variances. First we create a histogram of the residual values.



The residuals do appear to have, at least approximately, a normal distributed.

Plot of residuals versus predicted values

Next we plot the residuals versus the predicted values.

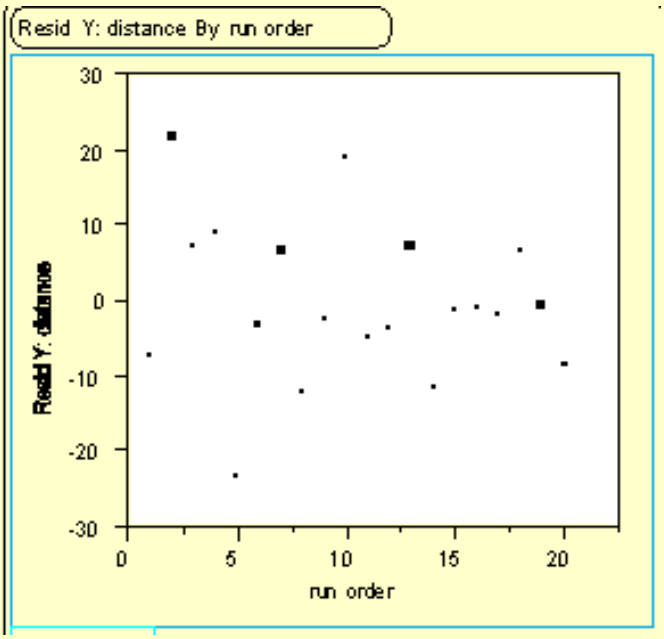


There does not appear to be a pattern to the residuals. One observation about the graph, from a single point, is that the model performs poorly in predicting a short distance. In fact, run number 10 had a measured distance of 8 inches, but the model predicts -11 inches, giving a residual of 19. The fact that the model predicts an impossible negative distance is an obvious shortcoming of the model. We may not be successful at predicting the catapult settings required to hit a distance less

than 25 inches. This is not surprising since there is only one data value less than 28 inches. Recall that the objective is for distances of 30, 60, and 90 inches.

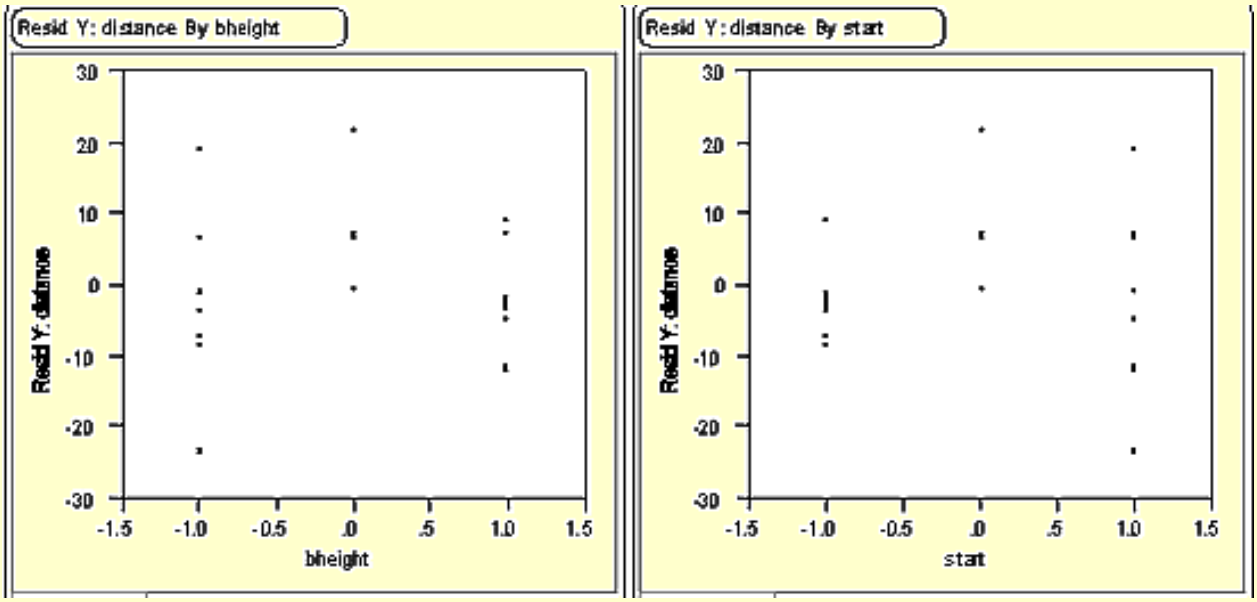
Plot of residuals versus run order

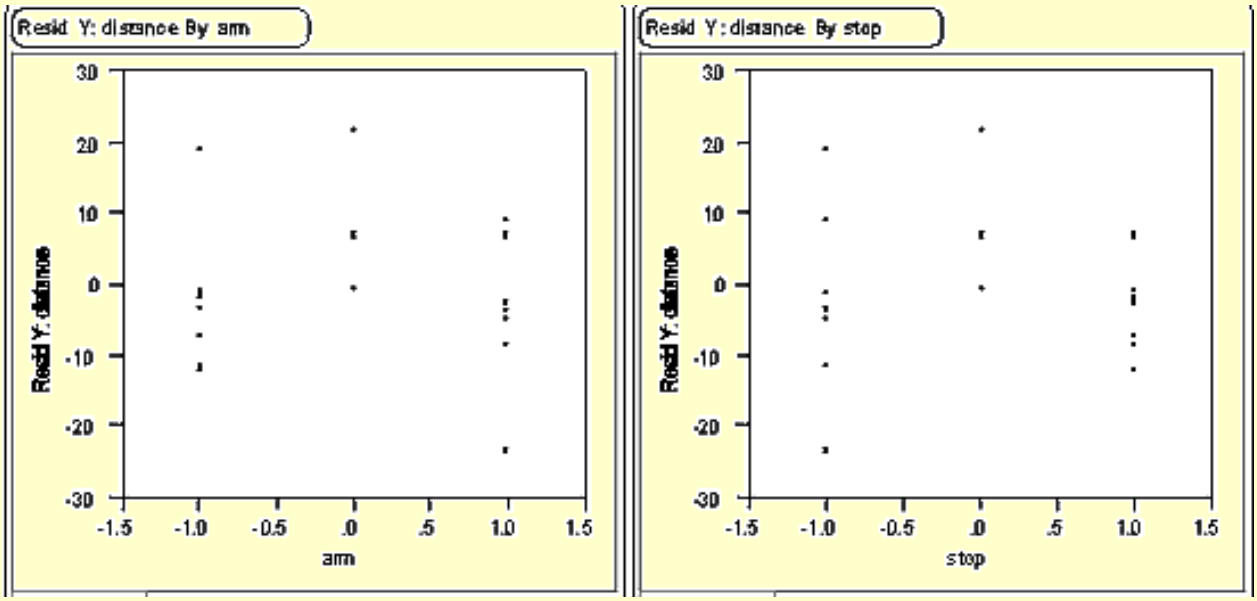
Next we plot the residual values versus the run order of the design. The highlighted points are the centerpoint values. Recall that run numbers 2 and 13 had two rubber bands while run numbers 7 and 19 had only one rubber band.

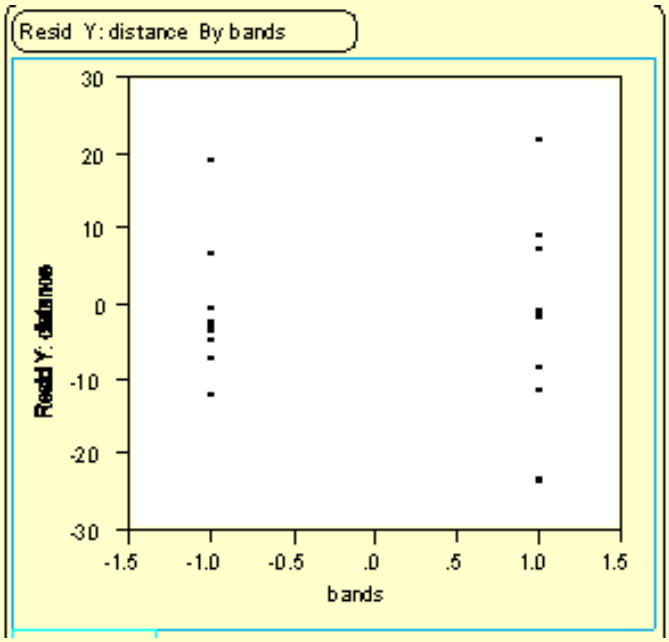


Plots of residuals versus the factor variables

Next we look at the residual values versus each of the factors.







The residual graphs are not ideal, although the model passes "lack of fit" quantitative tests

Most of the residual graphs versus the factors appear to have a slight "frown" on the graph (higher residuals in the center). This may indicate a lack of fit, or sign of curvature at the centerpoint values. The Lack of Fit table, however, indicates that the lack of fit is not significant.

Consider a transformation of the response variable to see if we can obtain a better model

At this point, since there are several unsatisfactory features of the model we have fit and the resultant residuals, we should consider whether a simple transformation of the response variable (*Y* = "Distance") might improve the situation.

There are at least two good reasons to suspect that using the logarithm of distance as the response might lead to a better model.

1. A linear model fit to LN *Y* will always predict a positive distance when converted back to the original scale for any possible combination of *X* factor values.
2. Physical considerations suggest that a realistic model for distance might require quadratic terms since gravity plays a key role - taking logarithms often reduces the impact of non-linear terms.

To see whether using LN *Y* as the response leads to a more satisfactory model, we return to step 3.

Step 3a: Fit the full model using LN Y as the response

First a main effects and 2-factor interaction model is fit to the log distance responses

Proceeding as before, using the coded columns of the matrix for the factor levels and *Y* = the natural logarithm of distance as the response, we initially obtain:

Response: Log Y

Summary of Fit

RSquare	0.956365
RSquare Adj	0.792734
Root Mean Square Error	0.307
Mean of Response	3.857023
Observations (or Sum Wgts)	20

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	3.8570234	0.068647	56.19	<.0001
bheight	0.2573518	0.07675	3.35	0.0285
start	-0.241745	0.07675	-3.15	0.0345
bheight*start	-0.025818	0.07675	-0.34	0.7535
bands	0.3488009	0.068647	5.08	0.0071
bheight*bands	-0.020347	0.07675	-0.27	0.8040
start*bands	0.0085304	0.07675	0.11	0.9169
arm	0.3943654	0.07675	5.14	0.0068
bheight*arm	-0.013955	0.07675	-0.18	0.8646
start*arm	0.0677498	0.07675	0.88	0.4272
bands*arm	0.0149847	0.07675	0.20	0.8547
stop	0.2627291	0.07675	3.42	0.0267
bheight*stop	-0.048729	0.07675	-0.63	0.5600
start*stop	0.0795455	0.07675	1.04	0.3585
bands*stop	-0.011521	0.07675	-0.15	0.8879
arm*stop	-0.011195	0.07675	-0.15	0.8911

A simpler model with just main effects has a satisfactory fit

Examining the *p*-values of the 16 model coefficients, only the intercept and the 5 main effect terms appear significant. Refitting the model with just these terms yields the following results.

Response: Log Y

Summary of Fit

RSquare	0.928359
RSquare Adj	0.902773
Root Mean Square Error	0.210265
Mean of Response	3.857023
Observations (or Sum Wgts)	20

Lack of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack of Fit	12	0.58979890	0.049150	33.710
Pure Error	2	0.02916084	0.014580	Prob>F
Total Error	14	0.61895974		0.2514
				Max RSq
				0.9966

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	3.8570234	0.047017	82.04	<.0001
bheight	0.2573518	0.052566	4.90	0.0002
start	-0.241745	0.052566	-4.60	0.0004
hands	0.3488009	0.047017	7.42	<.0001
arm	0.3943654	0.052566	7.50	<.0001
stop	0.2627291	0.052566	5.00	0.0002

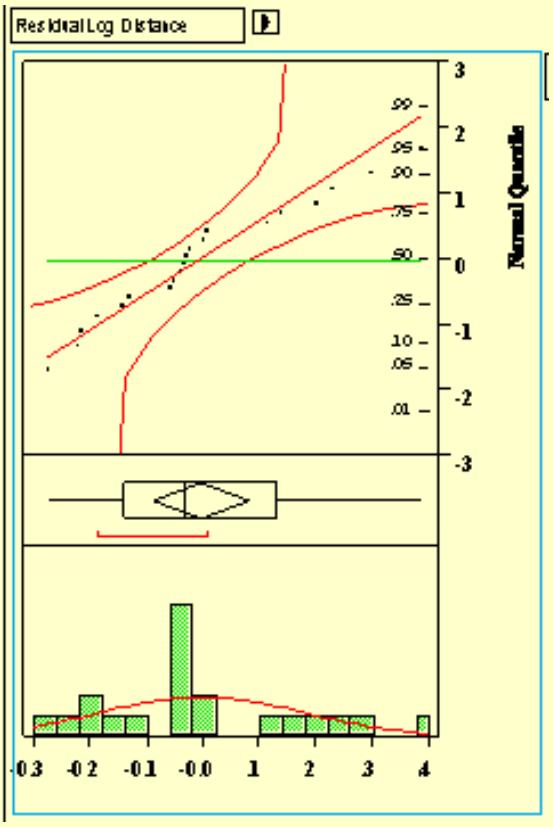
This is a simpler model than previously obtained in Step 3 (no interaction term). All the terms are highly significant and there is no quantitative indication of "lack of fit".

We next look at the residuals for this new model fit.

Step 4a: Test the (new) model assumptions using residual graphs (adjust and simplify as needed)

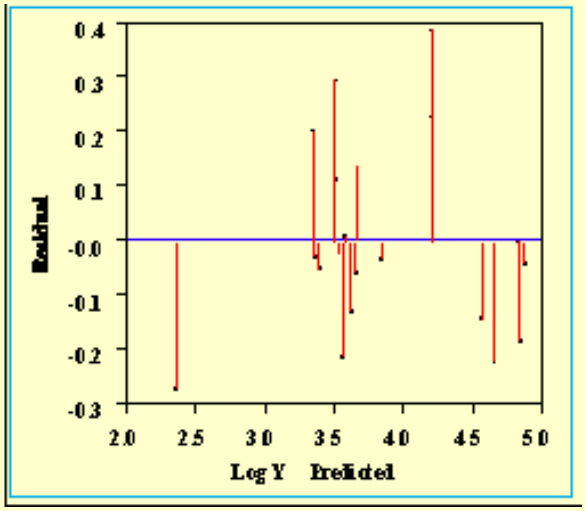
Normal probability plot, box plot, and histogram of the residuals

The following normal plot, box plot, and histogram of the residuals shows no problems.



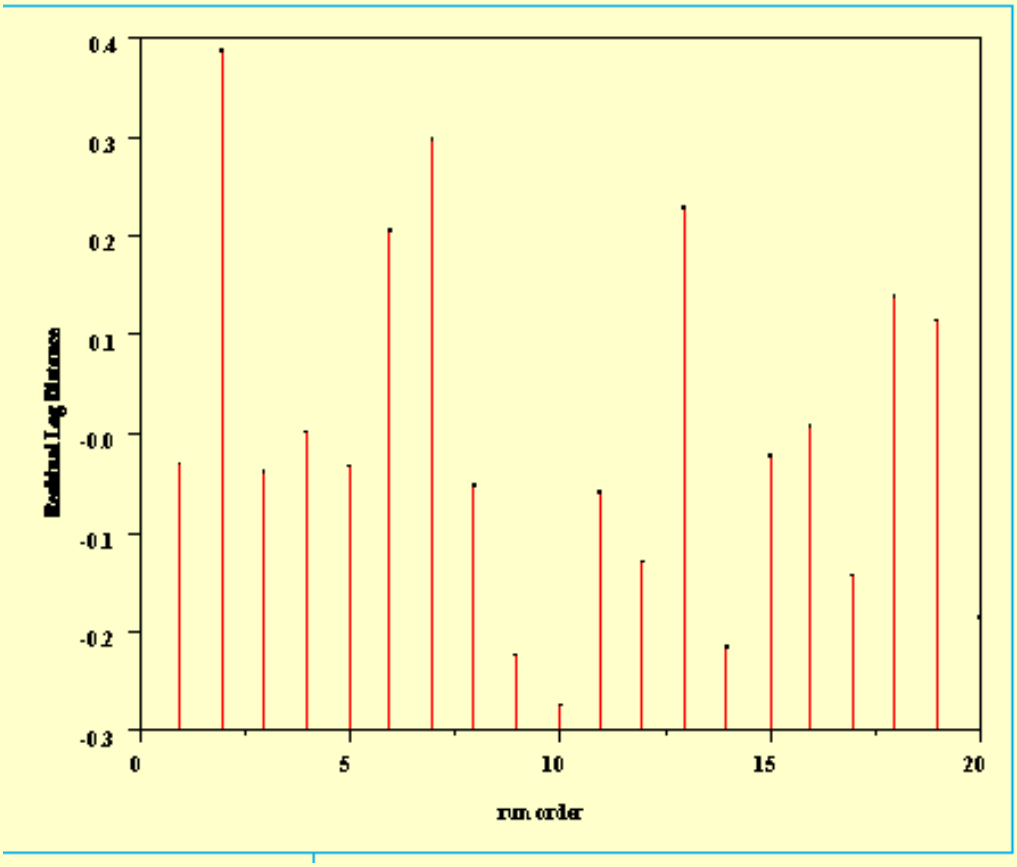
Plot of residuals versus predicted LN Y values

A plot of the residuals versus the predicted LN *Y* values looks reasonable, although there might be a tendency for the model to overestimate slightly for high predicted values.



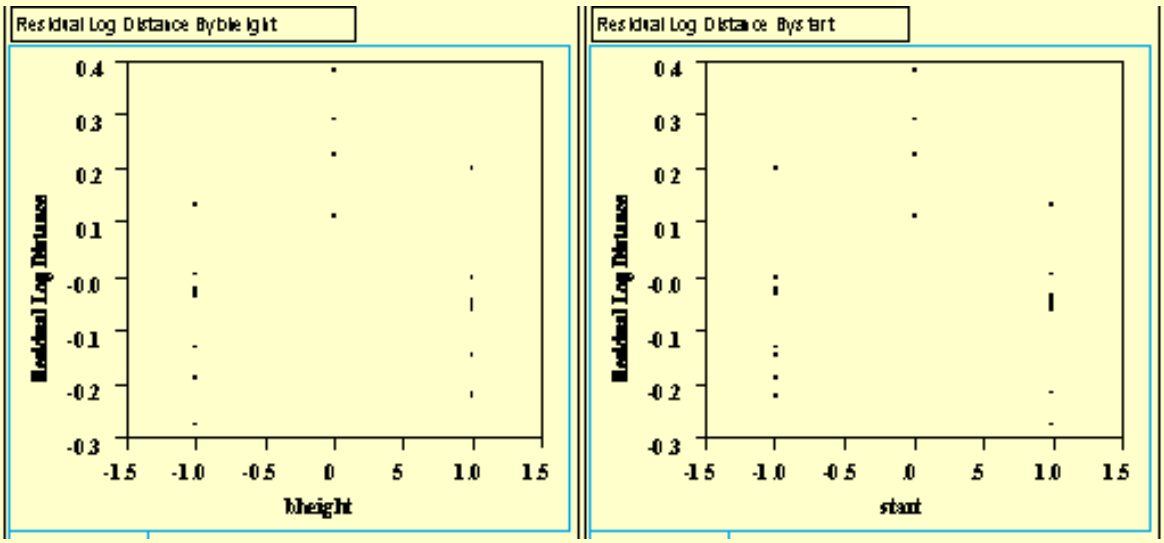
Plot of residuals versus run order

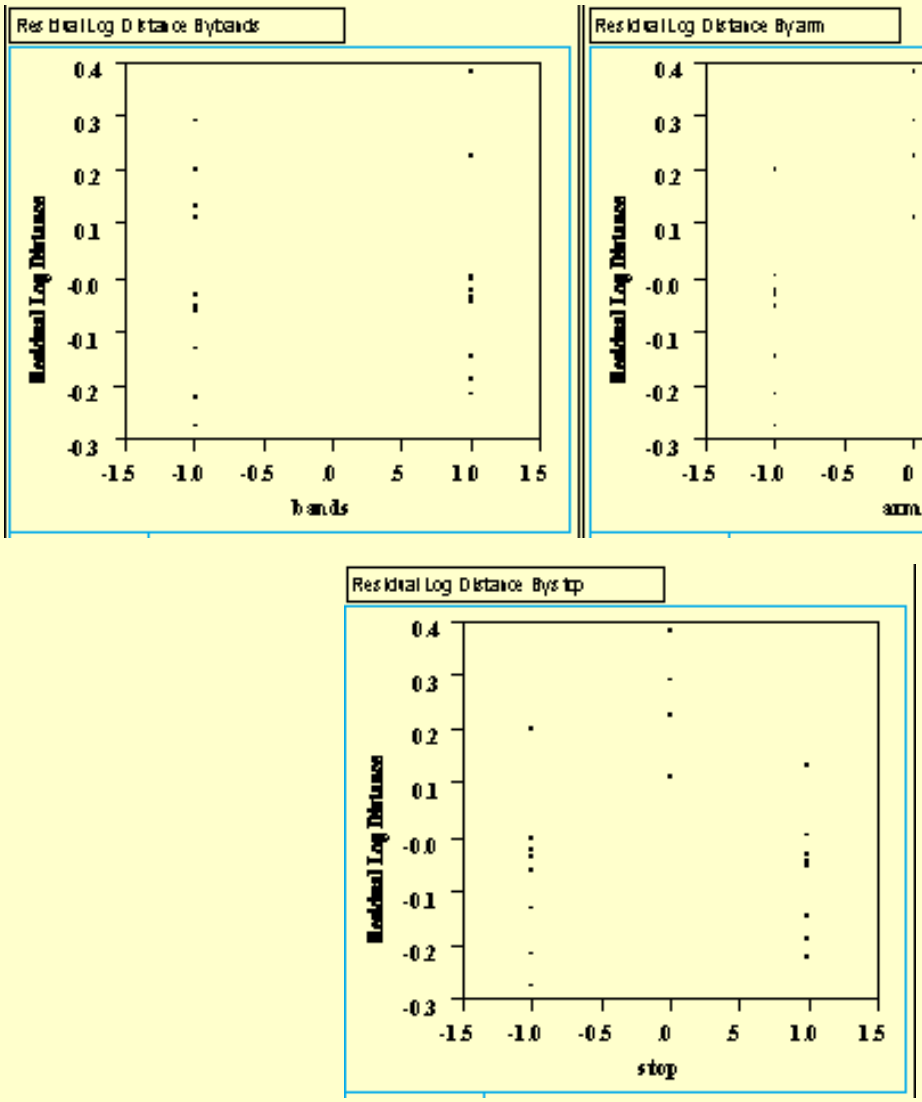
Residuals plotted versus run order again show a possible slight decreasing trend (rubber band fatigue?).



Plot of residuals versus the factor variables

Next we look at the residual values versus each of the factors.





The residuals for the main effects model (fit to natural log distance) are reasonably well behaved

These plots still appear to have a slight "frown" on the graph (higher residuals in the center). However, the model is generally an improvement over the previous model and will be accepted as possibly the best that can be done without conducting a new experiment designed to fit a quadratic model.

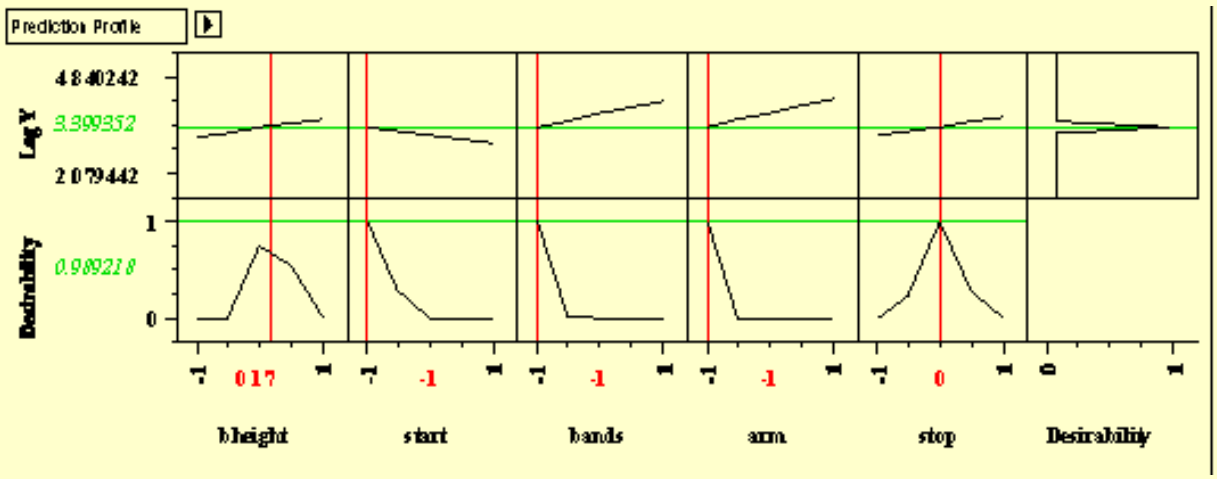
Step 5: Use the results to answer the questions in your experimental objectives

Final step: quantify the influence of all the significant effects and predict what settings should be used to obtain desired distances

The software used for this analysis (JMP 3.2.6) has an option called the "Prediction Profiler" that can be used to derive settings that will yield a desired predicted natural log distance value. The top graph in the figure below shows the direction and strength of each of the main effects in the model. Using natural log 30 = 3.401 as the target value, the Profiler allows us to set up a "Desirability" function that gives 3.401 a maximum desirability value of 1 and values above or below 3.401 have desirabilities that rapidly decrease to 0. This is shown by the desirability graph on the right (see the figure below).

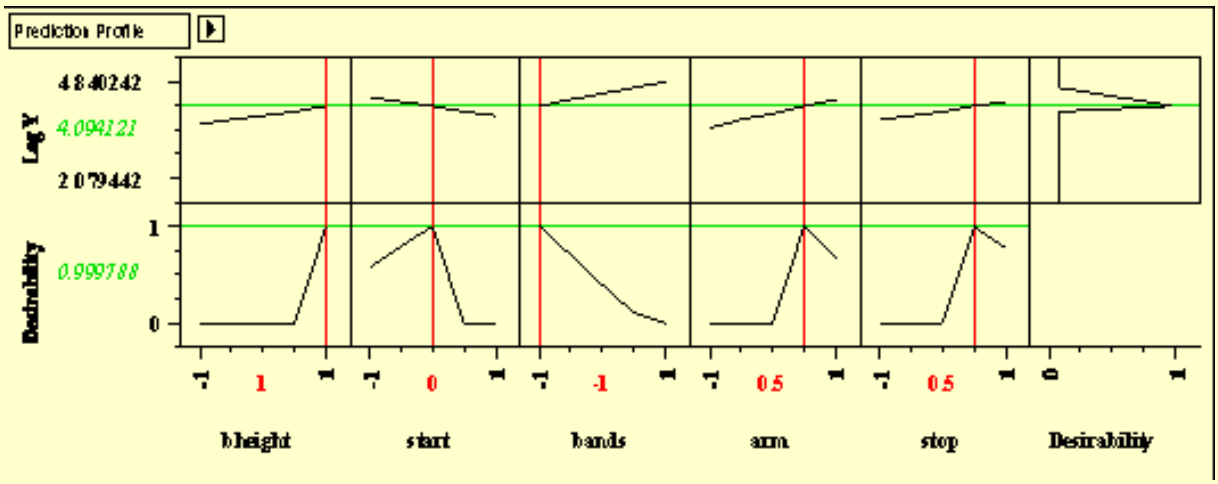
The next step is to set "bands" to either -1 or +1 (this is a discrete factor) and move the values of the other factors interactively until a desirability as close as possible to 1 is obtained. In the figure below, a desirability of .989218 was obtained, yielding a predicted natural log *Y* of 3.399351 (or a distance of 29.94). The corresponding (coded) factor settings are: bheight = 0.17, start = -1, bands = -1, arm = -1 and stop = 0.

Prediction profile plots for Y = 30



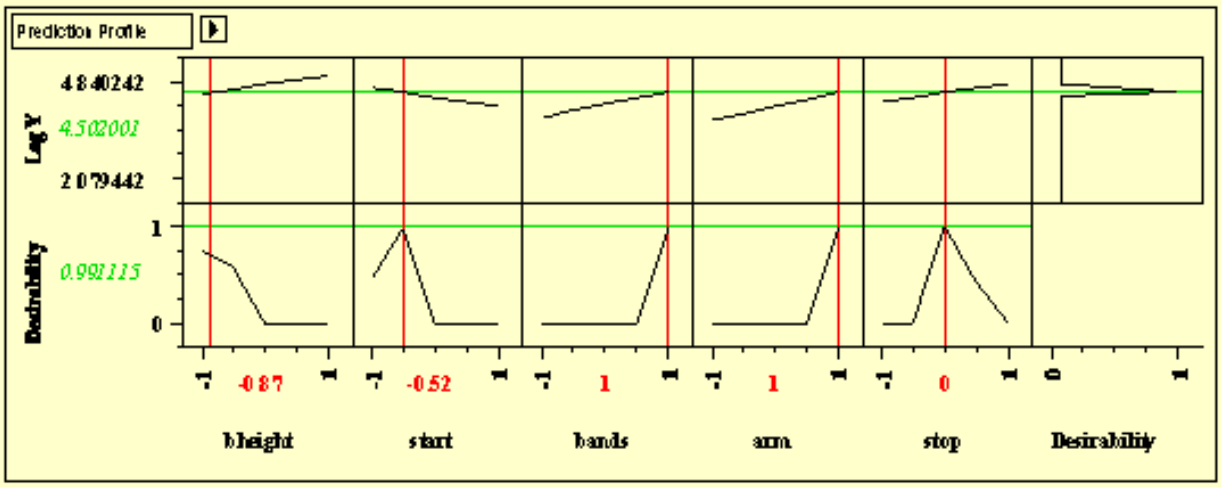
Prediction profile plots for Y = 60

Repeating the profiler search for a *Y* value of 60 (or LN *Y* = 4.094) yielded the figure below for which a natural log distance value of 4.094121 is predicted (a distance of 59.99) for coded factor settings of bheight = 1, start = 0, bands = -1, arm = .5 and stop = .5.



Prediction profile plots for Y = 90

Finally, we set LN *Y* = LN 90 = 4.4998 and obtain (see the figure below) a predicted log distance of 90.20 when bheight = -0.87, start = -0.52, bands = 1, arm = 1, and stop = 0.



"Confirmation" runs were successful

In the confirmatory runs that followed the experiment, the team was successful at hitting all 3 targets, but did not hit them all 5 times.

NOTE: The model discovery and fitting process, as illustrated in this analysis, is often an iterative process.

5.4.7.3. Response surface model example

Data Source

A CCD DOE with two responses

This example uses experimental data published in Czitrom and Spagon, (1997), *Statistical Case Studies for Industrial Process Improvement.* This material is copyrighted by the American Statistical Association and the Society for Industrial and Applied Mathematics, and used with their permission. Specifically, Chapter 15, titled "Elimination of TiN Peeling During Exposure to CVD Tungsten Deposition Process Using Designed Experiments", describes a semiconductor wafer processing experiment (labeled *Experiment 2*).

Goal, response variables, and factor variables

The goal of this experiment was to fit response surface models to the two responses, *deposition layer Uniformity* and *deposition layer Stress*, as a function of two particular controllable factors of the chemical vapor deposition (CVD) reactor process. These factors were *Pressure* (measured in torr) and the ratio of the gaseous reactants H_2 and WF_6 (called H_2/WF_6). The experiment also included an important third (categorical) response - the presence or absence of titanium nitride (TiN) peeling. That part of the experiment has been omitted in this example, in order to focus on the response surface model aspects.

To summarize, the goal is to obtain a response surface model for each response where the responses are: "Uniformity" and "Stress". The factors are: "Pressure" and "H_2/WF_6".

Experiment Description

The design is a 13-run CCI design with 3 centerpoint runs

The maximum and minimum values chosen for pressure were 4 torr and 80 torr. The lower and upper H_2/WF_6 ratios were chosen to be 2 and 10. Since response curvature, especially for Uniformity, was a distinct possibility, an experimental design that allowed estimating a second order (quadratic) model was needed. The experimenters decided to use a central composite inscribed (CCI) design. For two factors, this design is typically recommended to have 13 runs with 5 centerpoint runs. However, the experimenters, perhaps to conserve a limited supply of wafer resources, chose to include only 3 centerpoint runs. The design is still rotatable, but the uniform precision property has been sacrificed.

Table containing the CCI design and experimental responses

The table below shows the CCI design and experimental responses, in the order in which they were run (presumably randomized). The last two columns show coded values of the factors.

Run	Pressure	H_2/WF_6	Uniformity	Stress	Coded Pressure	Coded H_2/WF_6
1	80	6	4.6	8.04	1	0
2	42	6	6.2	7.78	0	0
3	68.87	3.17	3.4	7.58	0.71	-0.71
4	15.13	8.83	6.9	7.27	-0.71	0.71
5	4	6	7.3	6.49	-1	0
6	42	6	6.4	7.69	0	0
7	15.13	3.17	8.6	6.66	-0.71	-0.71
8	42	2	6.3	7.16	0	-1
9	68.87	8.83	5.1	8.33	0.71	0.71
10	42	10	5.4	8.19	0	1
11	42	6	5.0	7.90	0	0

Low values of both responses are better than high

Note: "Uniformity" is calculated from four-point probe sheet resistance measurements made at 49 different locations across a wafer. The value used in the table is the standard deviation of the 49 measurements divided by their mean, expressed as a percentage. So a smaller value of "Uniformity" indicates a more uniform layer - hence, lower values are desirable. The "Stress" calculation is based on an optical measurement of wafer bow, and again lower values are more desirable.

Analysis of DOE Data Using JMP 4.02

Steps for fitting a response surface model using JMP 4.02 (other software packages generally have similar procedures)

The steps for fitting a response surface (second-order or quadratic) model using the JMP 4.02 software for this example are as follows:

1. Specify the model in the "Fit Model" screen by inputting a response variable and the model effects (factors) and using the macro labeled "Response Surface".
2. Choose the "Stepwise" analysis option and select "Run Model".
3. The stepwise regression procedure allows you to select probabilities (*p*-values) for adding or deleting model terms. You can also choose to build up from the simplest models by adding and testing higher-order terms (the "forward" direction), or starting with the full second-order model and eliminating terms until the most parsimonious, adequate model is obtained (the "backward" direction). In combining the two approaches, JMP tests for both addition and deletion, stopping when no further changes to the model can be made. A choice of *p*-values set at 0.10 generally works well, although sometimes the user has to experiment here. Start the stepwise selection process by selecting "go".
4. "Stepwise" will generate a screen with recommended model terms checked and *p*-values shown (these are called "Prob>F" in the output). Sometimes, based on *p*-values, you might choose to drop, or uncheck, some of these terms. However, follow the *hierarchy principle* and keep all main effects that are part of significant higher-order terms or interactions, even if the main effect *p*-value is higher than you would like (note that not all analysts agree with this principle).
5. Choose "make model" and "run model" to obtain the full range of JMP graphic and analytical outputs for the selected model.
6. Examine the fitted model plot, normal plot of effects, interaction plots, residual plots, and ANOVA statistics (R^2, R^2 adjusted, lack of fit test, etc.). By saving the residuals onto your JMP worksheet you can generate residual distribution plots (histograms, box plots, normal plots, etc.). Use all these plots and statistics to determine whether the model fit is satisfactory.

7. Use the JMP contour profiler to generate response surface contours and explore the effect of changing factor levels on the response.
8. Repeat all the above steps for the second response variable.
9. Save prediction equations for each response onto your JMP worksheet (there is an option that does this for you). After satisfactory models have been fit to both responses, you can use "Graph" and "Profiler" to obtain overlaid surface contours for both responses.
10. "Profiler" also allows you to (graphically) input a desirability function and let JMP find optimal factor settings.

The displays below are copies of JMP output screens based on following the above 10 steps for the "Uniformity" and "Stress" responses. Brief margin comments accompany the screen shots.

Fitting a Model to the "Uniformity" Response, Simplifying the Model and Checking Residuals

Model specification screen and stepwise regression (starting from a full second-order model) output

We start with the model specification screen in which we input factors and responses and choose the model we want to fit. We start with a full second-order model and select a "Stepwise Fit". We set "prob" to 0.10 and direction to "Mixed" and then "Go".

Model Specification

Select Columns: Pressure, H2/WF6, Uniformity, Stress, npressure, nH2/WF6

Pick Role Variables: Uniformity

Personality: Stepwise

Construct Model Effects: Degree 2; Attributes: No Intercept

Pressure &RS
H2/WF6 &RS
Pressure*Pressure
H2/WF6*Pressure
H2/WF6*H2/WF6

Stepwise Fit

Response: Uniformity

Stepwise Regression Control

Prob to Enter 0.100
Prob to Leave 0.100
Direction: Mixed
Rules: Combine

Current Estimates

SSE	DFE	MSE	RSquare	RSquare Adj	Cp	AIC
2.6597809	7	0.379969	0.8695	0.8136	2.047747	-7.61617

Lock	Entered	Parameter	Estimate	nDF	SS	"F Ratio"	"Prob>F"
☑	☑	Intercept	5.92727273	1	0	0.000	1.0000
☐	☑	Pressure(4,80)	-1.9124383	2	17.51965	23.054	0.0008
☐	☑	H2/WF6(2,10)	-0.2248749	2	3.092387	4.069	0.0672
☐	☐	Pressure(4,80)*Pressure(4,80)	.	1	0.023583	0.054	0.8245
☐	☑	H2/WF6(2,10)*Pressure(4,80)	1.69905881	1	2.89	7.606	0.0282
☐	☐	H2/WF6(2,10)*H2/WF6(2,10)	.	1	0.000052	0.000	0.9917

Step History

Step History

Step	Parameter	Action	"Sig Prob"	Seq SS	RSquare	Cp	p
1	Pressure(4,80)	Entered	0.0010	14.62965	0.7178	3.9165	2
2	H2/WF6(2,10)*Pressure(4,80)	Entered	0.0672	3.092387	0.8695	2.0477	4

The stepwise routine finds the intercept and three other terms (the main effects and the interaction term) to be significant.

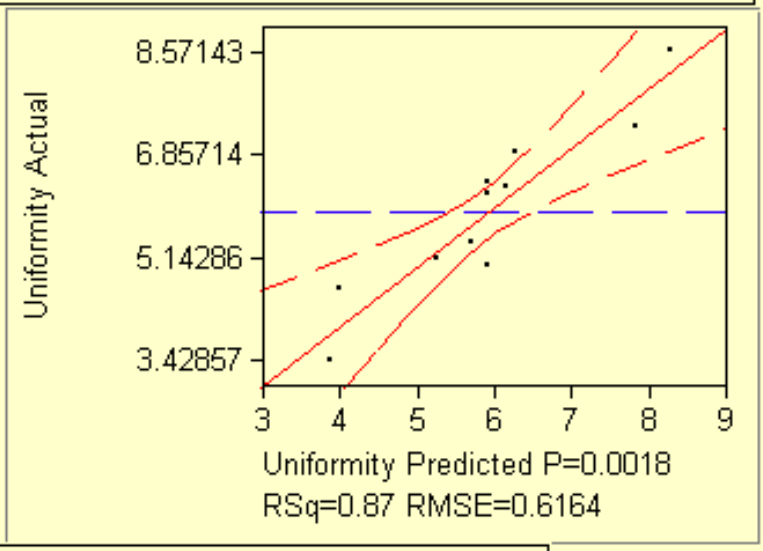
JMP output for analyzing the model selected by the stepwise regression for the Uniformity response

The following is the JMP analysis using the model selected by the stepwise regression in the previous step. The model is fit using coded factors, since the factor columns were given the property "coded".

Response Uniformity

Whole Model

Actual by Predicted Plot



Summary of Fit

RSquare	0.869502
RSquare Adj	0.813575
Root Mean Square Error	0.616416
Mean of Response	5.927273
Observations (or Sum Wgts)	11

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	3	17.722037	5.90735	15.5469
Error	7	2.659781	0.37997	Prob > F
C. Total	10	20.381818		0.0018

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	5	1.5131142	0.302623	0.5278
Pure Error	2	1.1466667	0.573333	Prob > F
Total Error	7	2.6597809		0.7559
				Max RSq
				0.9437

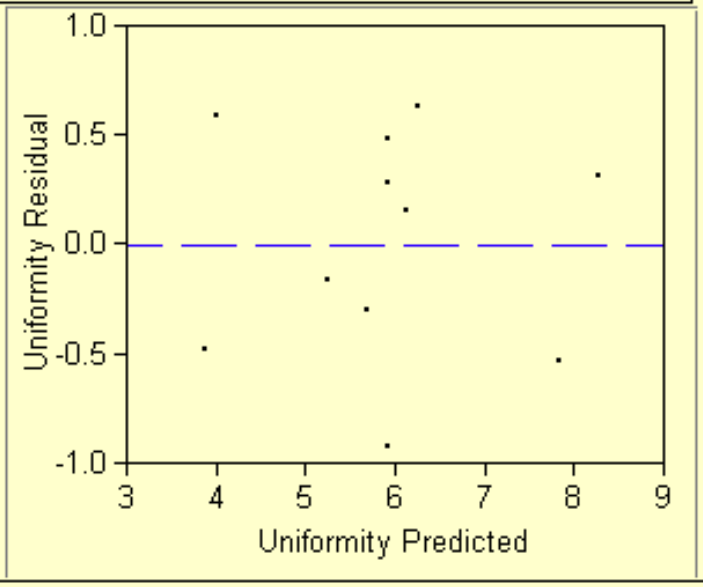
Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	5.9272727	0.185856	31.89	<.0001
Pressure(4,80)	-1.912438	0.308208	-6.21	0.0004
H2/WF6(2,10)	-0.224875	0.308122	-0.73	0.4892
H2/WF6(2,10)*Pressure(4,80)	1.6990588	0.616075	2.76	0.0282

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
Pressure(4,80)	1	1	14.629650	38.5022	0.0004
H2/WF6(2,10)	1	1	0.202387	0.5326	0.4892
H2/WF6(2,10)*Pressure(4,80)	1	1	2.890000	7.6059	0.0282

Residual by Predicted Plot



Effect Screening

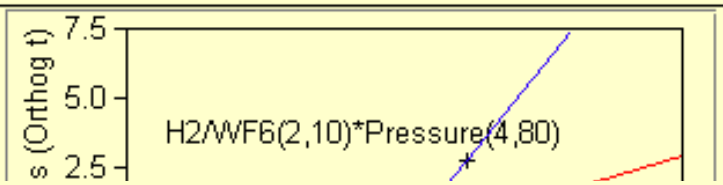
The parameter estimates are not correlated.
The parameter estimates below were transformed to have equal variances.

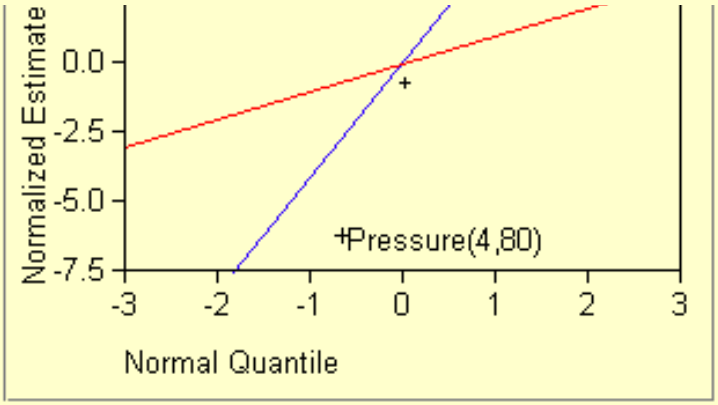
	Lenth PSE
t-Test Scale	4.1368166
Coded Scale	0.7688539

Parameter Estimate Population

Term	Original	Orthog Coded	Orthog t-Test	Prob>|t|
Intercept	5.927273	5.927273	31.8917	<.0001
Pressure(4,80)	-1.912438	-1.153242	-6.2050	0.0004
H2/WF6(2,10)	-0.224875	-0.135642	-0.7298	0.4892
H2/WF6(2,10)*Pressure(4,80)	1.699059	0.512569	2.7579	0.0282

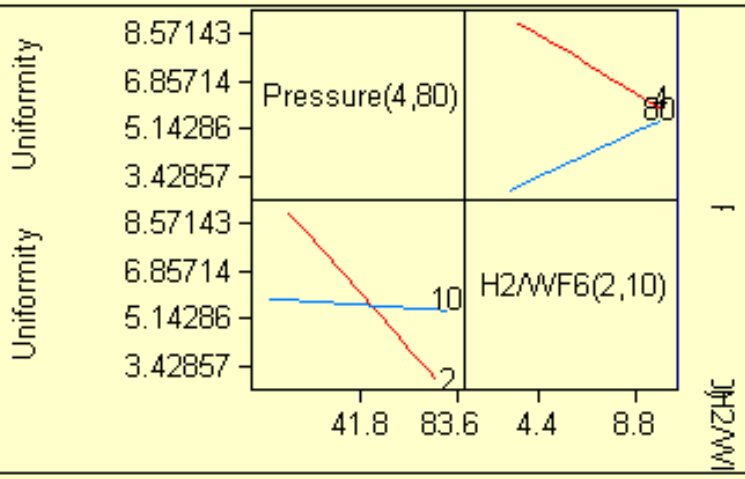
Normal Plot





Blue line is Lenth's PSE, from the estimates population.
Red line is RMSE, Root Mean Squared Error from the residual.

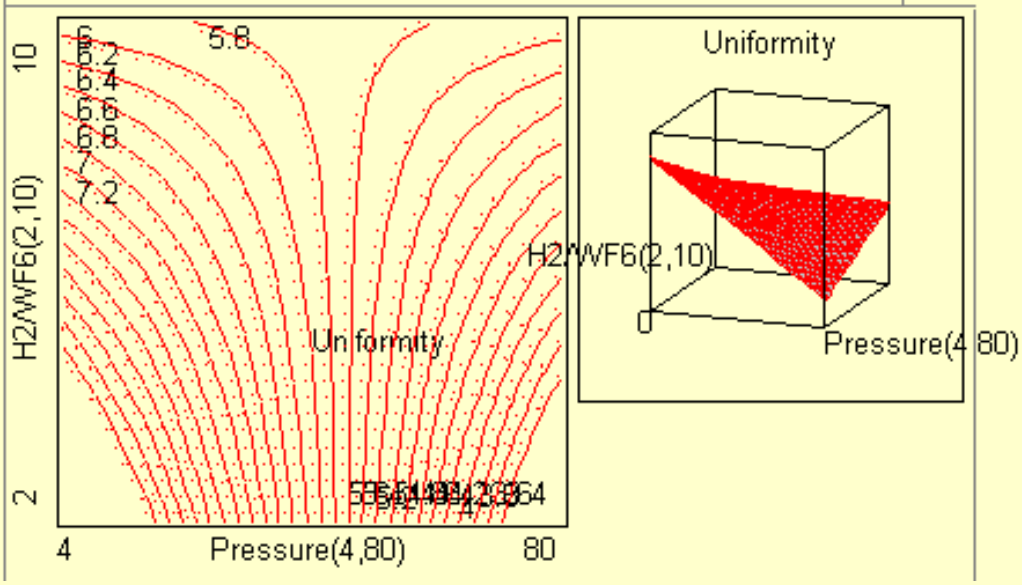
Interaction Profiles



Contour Profiler

Horiz	Vert	Factor	Current X
●	○	Pressure(4,80)	42
○	●	H2/WF6(2,10)	6

Response	Contour	Current Y	Lo Limit	Hi Limit
Uniformity	6	5.9272727	.	.



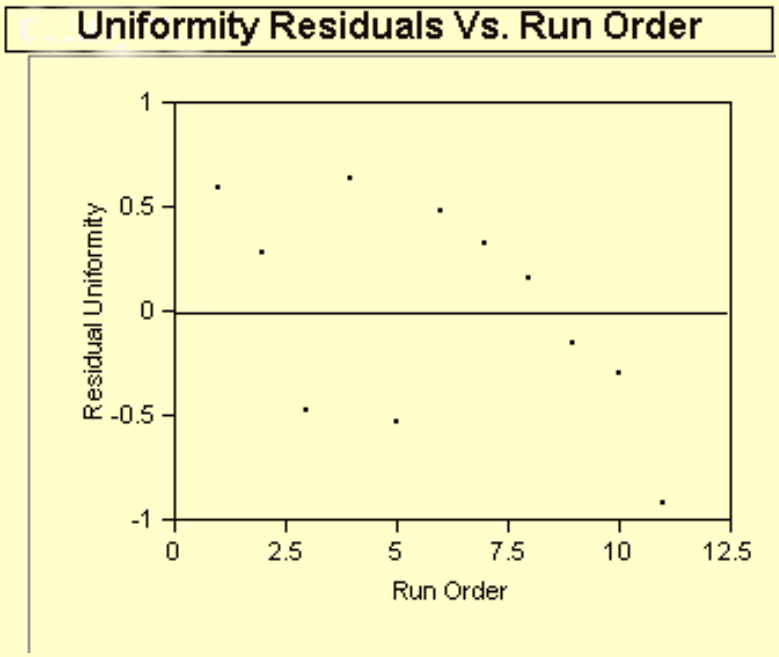
Conclusions from the JMP output

From the above output, we make the following conclusions.

- The R^2 is reasonable for fitting "Uniformity" (well known to be a hard response to model).
- The lack of fit test does not have a problem with the model (very small "Prob > F " would question the model).
- The residual plot does not reveal any major violations of the underlying assumptions.
- The normal plot of main effects and interaction effects provides a visual confirmation of the significant model terms.
- The interaction plot shows why an interaction term is needed (parallel lines would suggest no interaction).

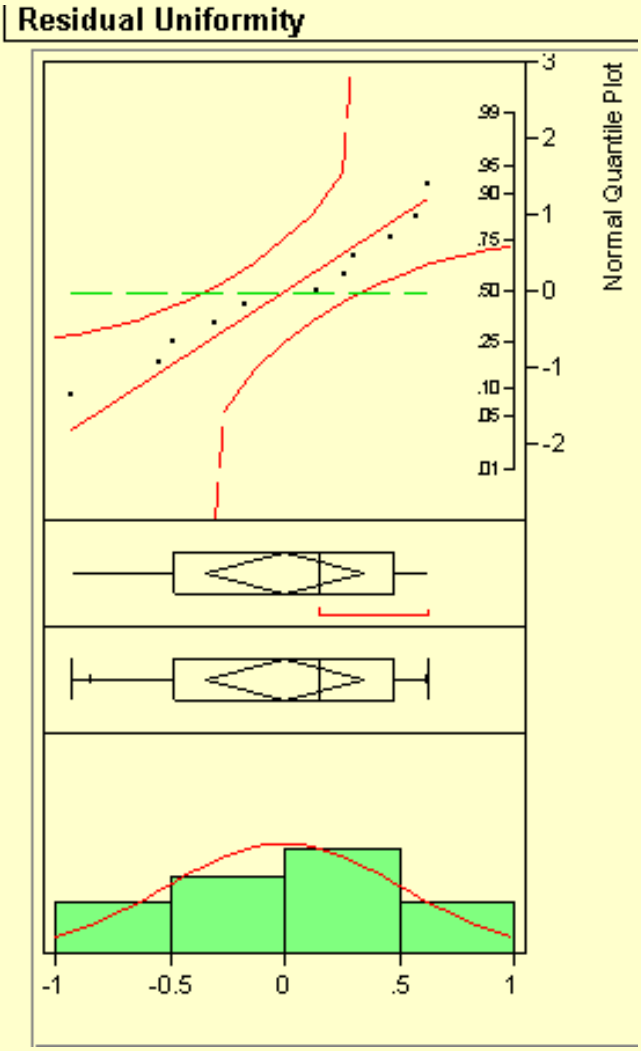
Plot of the residuals versus run order

We next perform a residuals analysis to validate the model. We first generate a plot of the residuals versus run order.



Normal plot, box plot, and histogram of the residuals

Next we generate a normal plot, a box plot, and a histogram of the residuals.



Viewing the above plots of the residuals does not show any reason to question the model.

Fitting a Model to the "Stress" Response, Simplifying the Model and Checking Residuals

Model specification screen and stepwise regression (starting from a full second-order model) output

We start with the model specification screen in which we input factors and responses and choose the model we want to fit. This time the "Stress" response will be modeled. We start with a full second-order model and select a "Stepwise Fit". We set "prob" to 0.10 and direction to "Mixed" and then "Go".

Model Specification

Select Columns: Pressure, H2/WF6, Uniformity, Stress, npressure, nH2/WF6

Pick Role Variables: Stress

Personality: Stepwise

Construct Model Effects: Degree 2; Attributes: No Intercept

Pressure &RS
H2/WF6 &RS
Pressure*Pressure
H2/WF6*Pressure
H2/WF6*H2/WF6

Stepwise Fit

Response: Stress

Stepwise Regression Control

Prob to Enter 0.100
Prob to Leave 0.100
Direction: Mixed
Rules: Combine

Current Estimates

SSE	DFE	MSE	RSquare	RSquare Adj	Cp	AIC
0.0544354	7	0.007776	0.9849	0.9784	6.320118	-50.395

Lock	Entered	Parameter	Estimate	nDF	SS	"F Ratio"	"Prob>F"
☑	☑	Intercept	7.73352911	1	0	0.000	1.0000
☐	☑	Pressure(4,80)	0.73751869	2	2.553956	164.210	0.0000
☐	☑	H2/WF6(2,10)	0.49777311	1	0.991664	127.521	0.0000
☐	☑	Pressure(4,80)*Pressure(4,80)	-0.4947061	1	0.378225	48.637	0.0002
☐	☐	H2/WF6(2,10)*Pressure(4,80)	.	1	0.0049	0.594	0.4703
☐	☐	H2/WF6(2,10)*H2/WF6(2,10)	.	1	0.020332	3.577	0.1075

Step History

Step	Parameter	Action	"Sig Prob"	Seq SS	RSquare	Cp	p
1	H2/WF6(2,10)*Pressure(4,80)	Entered	0.0013	3.172294	0.8812	70.239	4
2	H2/WF6(2,10)*Pressure(4,80)	Removed	0.7852	0.0049	0.8798	69.078	3
3	Pressure(4,80)*Pressure(4,80)	Entered	0.0002	0.378225	0.9849	6.3201	4

The stepwise routine finds the intercept, the main effects, and Pressure squared to be signficant terms.

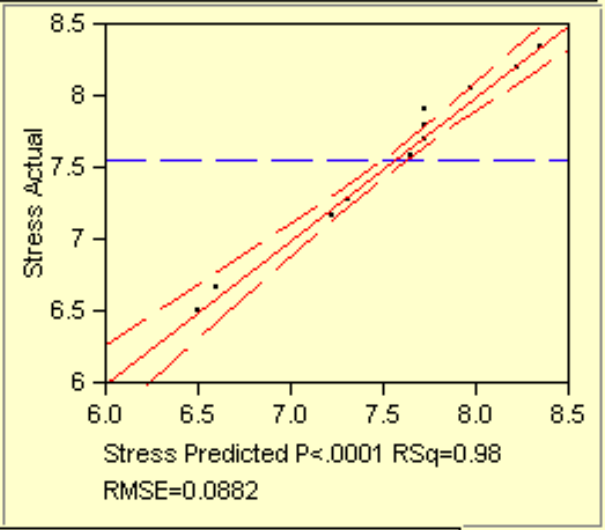
JMP output for analyzing the model selected by the stepwise regression for the Stress response

The following is the JMP analysis using the model selected by the stepwise regression, which contains four significant terms, in the previous step. The model is fit using coded factors, since the factor columns were given the property "coded".

Response Stress

Whole Model

Actual by Predicted Plot



Summary of Fit

RSquare	0.984879
RSquare Adj	0.978399
Root Mean Square Error	0.088184
Mean of Response	7.553636
Observations (or Sum Wgts)	11

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	3	3.5456191	1.18187	151.9802
Error	7	0.0544354	0.00778	Prob > F
C. Total	10	3.6000545		<.0001

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	5	0.03223544	0.006447	0.5808
Pure Error	2	0.02220000	0.011100	Prob > F
Total Error	7	0.05443544		0.7301
				Max RSq
				0.9938

Parameter Estimates

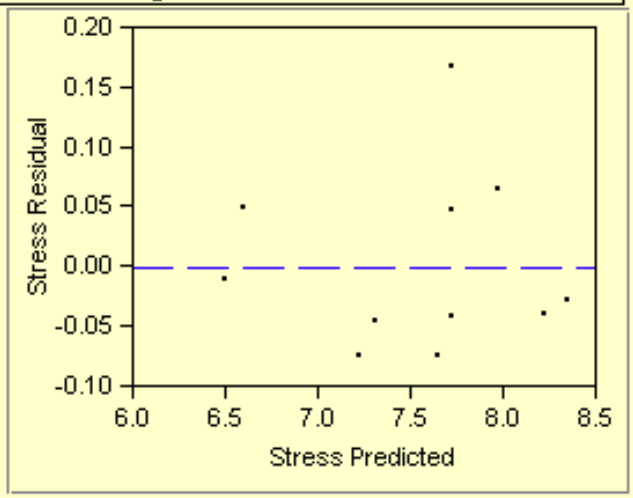
Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	7.7335291	0.037045	208.76	<.0001
Pressure(4,80)&RS	0.7375187	0.044092	16.73	<.0001
H2/WF6(2,10)&RS	0.4977731	0.04408	11.29	<.0001
Pressure(4,80)*Pressure(4,80)	-0.494706	0.070936	-6.97	0.0002

Effect Tests

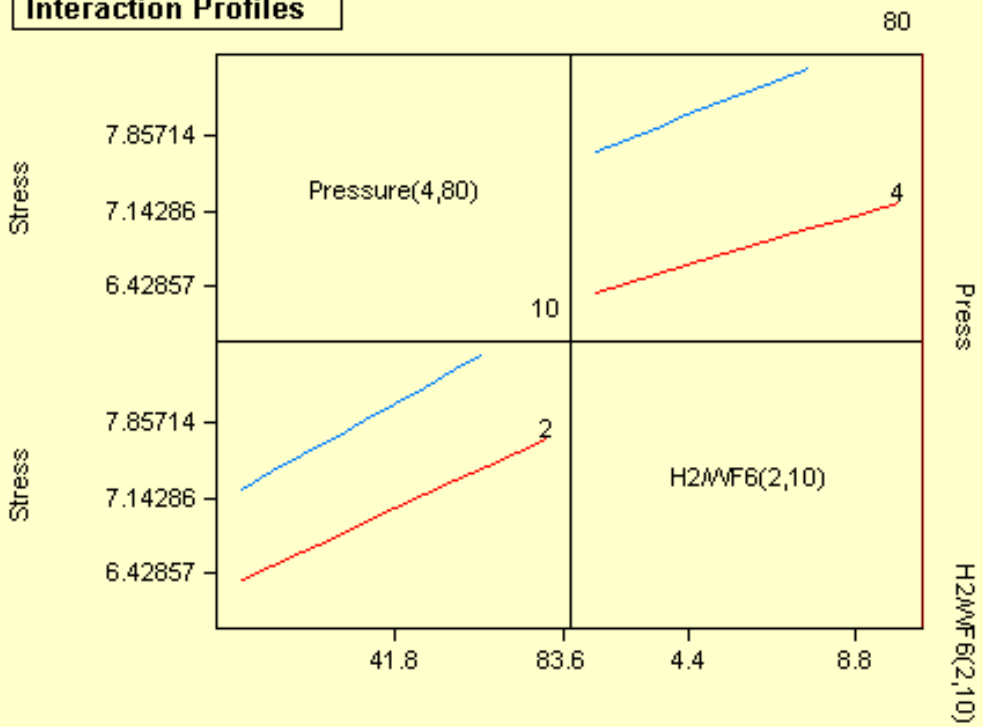
Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
Pressure(4,80)&RS	1	1	2.1757306	279.7830	<.0001
H2/WF6(2,10)&RS	1	1	0.9916636	127.5207	<.0001
Pressure(4,80)*Pressure(4,80)	1	1	0.3782249	48.6370	0.0002

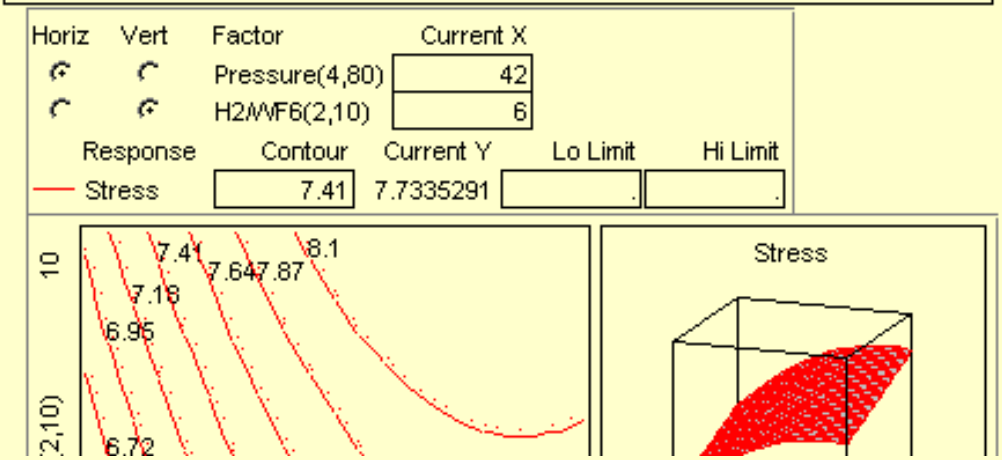
Residual by Predicted Plot

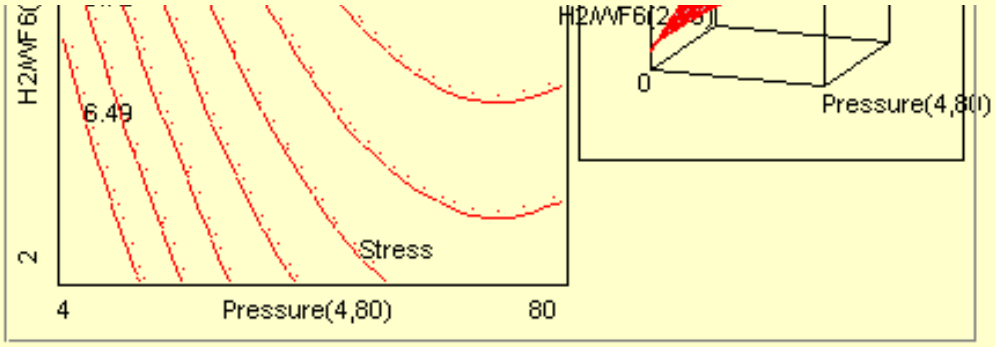


Interaction Profiles



Contour Profiler





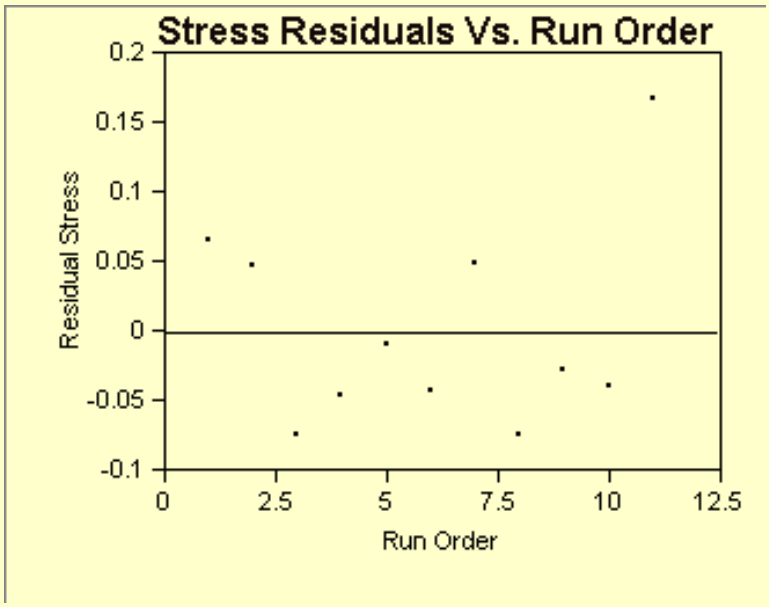
Conclusions from the JMP output

From the above output, we make the following conclusions.

- The R^2 is very good for fitting "Stress".
- The lack of fit test does not have a problem with the model (very small "Prob > F " would question the model).
- The residual plot does not reveal any major violations of the underlying assumptions.
- The interaction plot shows why an interaction term is needed (parallel lines would suggest no interaction).

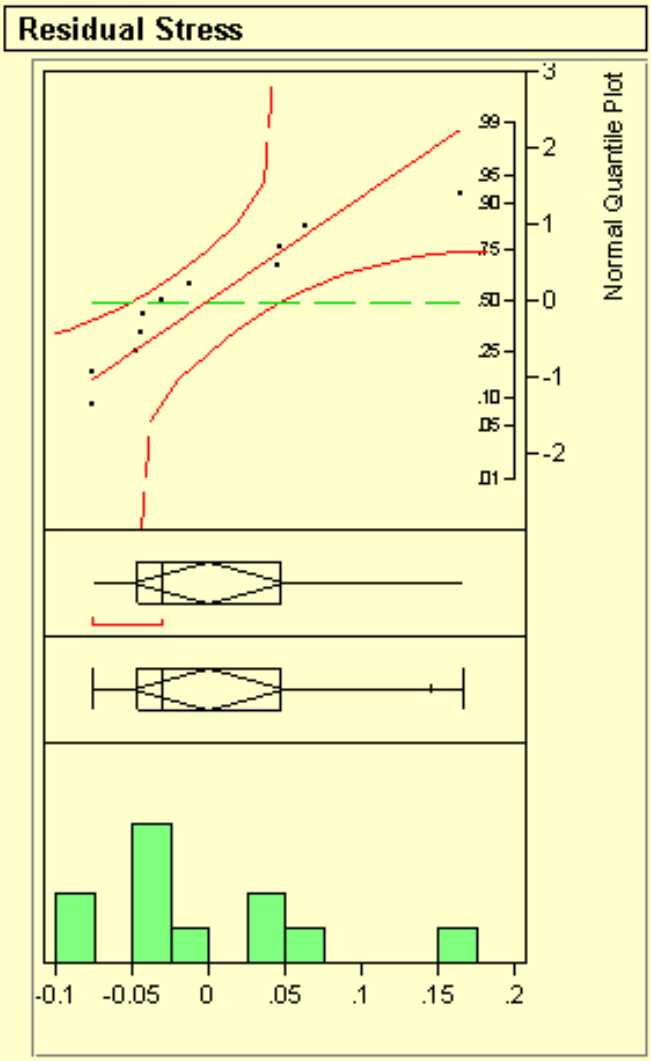
Plot of the residuals versus run order

We next perform a residuals analysis to validate the model. We first generate a plot of the residuals versus run order.



Normal plot, box plot, and histogram of the residuals

Next we generate a normal plot, a box plot, and a histogram of the residuals.

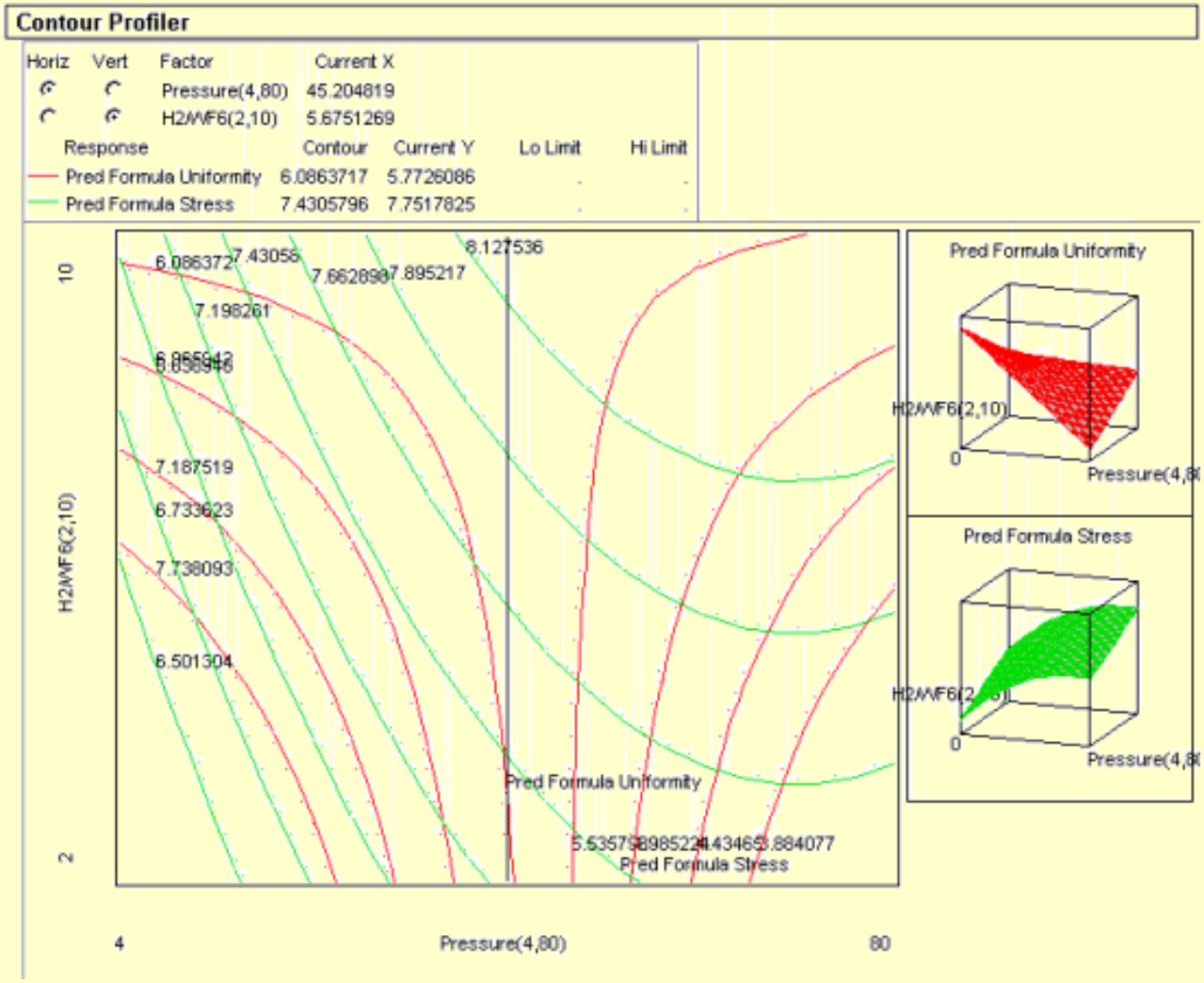


Viewing the above plots of the residuals does not show any reason to question the model.

Response Surface Contours for Both Responses

"Contour Profiler" and "Prediction Profiler"

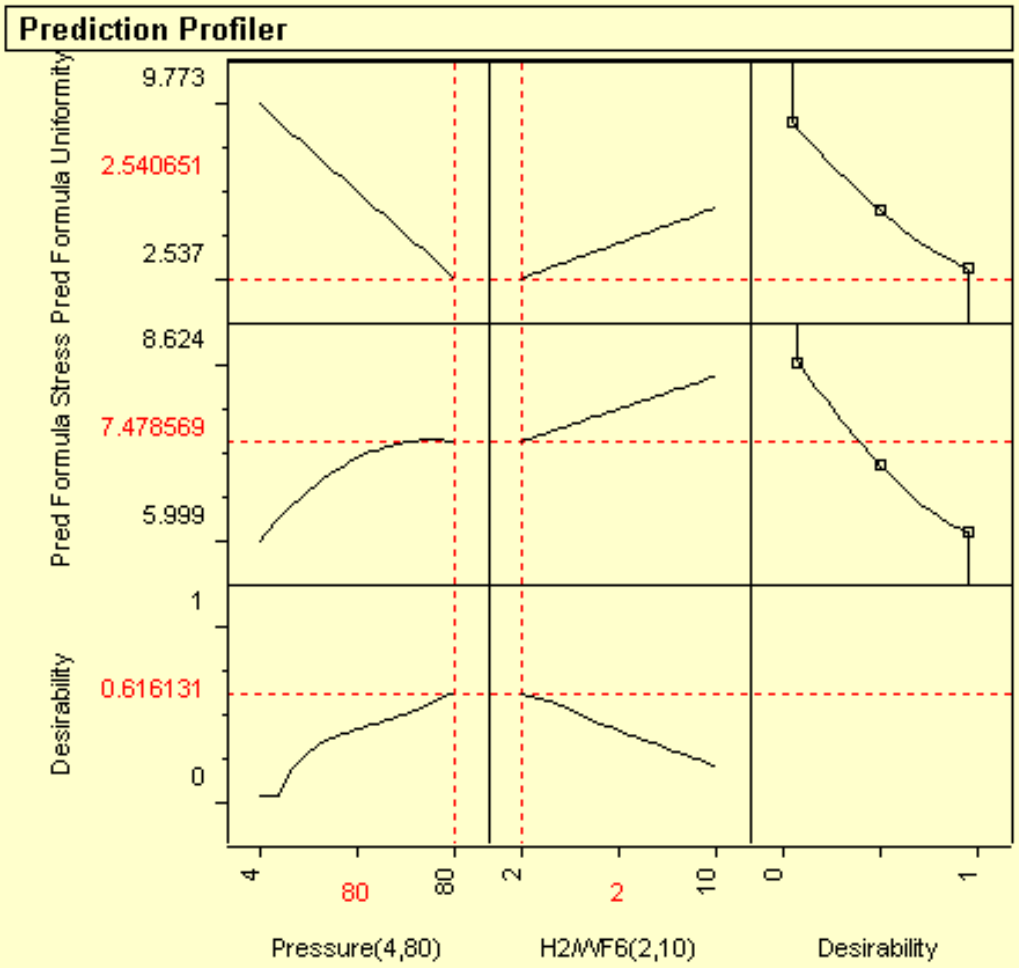
JMP has a "Contour Profiler" and "Prediction Profiler" that visually and interactively show how the responses vary as a function of the input factors. These plots are shown here for both the Uniformity and the Stress response.



Prediction Profiles Desirability Functions for Both Responses

Desirability function: Pressure should be as high as possible and H_2/WF_6 as low as possible

You can graphically construct a desirability function and let JMP find the factor settings that maximize it - here it suggests that Pressure should be as high as possible and H_2/WF_6 as low as possible.



Summary

Final response surface models

The response surface models fit to (coded) "Uniformity" and "Stress" were:

Uniformity = 5.93 - 1.91*Pressure - 0.22*H_2/WF_6 + 1.70*Pressure*H_2/WF_6

Stress = 7.73 + 0.74*Pressure + 0.50*H_2/WF_6 - 0.49*$Pressure^2$

Trade-offs are often needed for multiple responses

These models and the corresponding profiler plots show that trade-offs have to be made when trying to achieve low values for both "Uniformity" and "Stress" since a high value of "Pressure" is good for "Uniformity" while a low value of "Pressure" is good for "Stress". While low values of H_2/WF_6 are good for both responses, the situation is further complicated by the fact that the "Peeling" response (not considered in this analysis) was unacceptable for values of H_2/WF_6 below approximately 5.

"Uniformity" was chosen as more important

In this case, the experimenters chose to focus on optimizing "Uniformity" while keeping H_2/WF_6 at 5. That meant setting "Pressure" at 80 torr.

Confirmation runs validated the model projections	A set of 16 verification runs at the chosen conditions confirmed that all goals, except those for the "Stress" response, were met by this set of process settings.

5. Process Improvement

5.5. Advanced topics

Contents of "Advanced Topics" section

This section builds on the basics of DOE described in the preceding sections by adding brief or survey descriptions of a selection of useful techniques. Subjects covered are:

5. Nested variation
6. Taguchi designs
7. John's 3/4 fractional factorial designs
8. Small composite designs
9. An EDA approach to experimental design
    1. Ordered data plot
    2. Dex scatter plot
    3. Dex mean plot
    4. Interaction effects matrix plot
    5. Block plot
    6. DEX Youden plot
    7. |Effects| plot
    8. Half-normal probability plot
    9. Cumulative residual standard deviation plot
    10. DEX contour plot

5.5.1. What if classical designs don't work?

Reasons designs don't work.

Most experimental situations call for standard designs that can be constructed in many statistical software packages. Standard designs have assured degrees of precision, orthogonality, and other optimal properties that are important for the exploratory nature of most experiments. In some situations, however, standard designs are not appropriate or are impractical. These may include situations where

1. The required blocking structure or blocking size of the experimental situation does not fit into a standard blocked design
2. Not all combinations of the factor settings are feasible, or for some other reason the region of experimentation is constrained or irregularly shaped.
3. A classical design needs to be 'repaired'. This can happen due to improper planning where the original design treatment combinations contained forbidden or unreachable combinations that were not considered before the design was generated.
4. There is a nonlinear model.
5. A quadratic or response surface design is required in the presence of qualitative factors.
6. The factors in the experiment include both components of a mixture and other process variables.
7. There are multiple sources of variation leading to nested or hierarchical data structures and restrictions on what can be randomized.
8. A standard fractional factorial design requires too many treatment combinations for the given amount of time and / or resources,

Computer aided designs

When situations such as the above exist, computer aided designs are a useful option. In some situations, computer aided designs are the only option an experimenter has.

5.5.2. What is a computer-aided design?

Computer-aided designs are generated by a computer algorithm and designed to be optimal for certain models according to one of many types of optimality criteria

Designs generated from a computer algorithm are referred to as computer aided designs. Computer aided designs are experimental designs that are generated based on a particular optimality criterion and are generally 'optimal' only for a specified model. As a result, they are sometimes referred to as optimal designs and generally do not satisfy the desirable properties such as independence among the estimators that standard classical designs do. The design treatment runs that are generated by the algorithms are chosen from an overall candidate set of possible treatment combinations. The candidate set consists of all the possible treatment combinations that one wishes to consider in an experiment.

D-Optimality
A-Optimality
G-Optimality
V-Optimality

These are all optimality criteria that have been suggested

There are various forms of optimality criterion that are used to select the points for a design. One popular criterion is *D-optimality*, which seeks the maximize |X'X|, the determinant of the *information matrix* X'X of the design. This criterion results in minimizing the generalized variance of the parameter estimates based on a pre specified model.

Another criterion is *A-optimality*, which seeks to minimize the trace of the inverse of the information matrix. This criterion results in minimizing the average variance of the parameter estimates based on a pre-specified model.

A third criterion is *G-optimality*, which seeks to minimize the maximum prediction variance, i.e., minimize max. $[d=x'(X'X)^{-1}x]$, over a specified set of design points.

A fourth criterion is *V-optimality*, which seeks to minimize the average prediction variance over a specified set of design points.

Since the optimality criterion of most computer aided designs are based on some function of the information matrix, the 'optimality' of a given design is model dependent. That is, the experimenter must specify a model for the design and the final number of design

points desired before the 'optimal' design' can be generated. The design generated by the computer algorithm is 'optimal' only for that model.

5.5.2.1. D-Optimal designs

D-optimal designs are often used when classical designs do not apply or work

These designs are always an option regardless of model or resolution desired

You start with a candidate set of runs and the algorithm chooses a D-optimal set of design runs

D-optimal designs are one form of design provided by a computer algorithm. These types of computer aided designs are particularly useful when classical designs do not apply.

Unlike standard classical designs such as factorials and fractional factorials, D-optimal design matrices are usually not orthogonal and effect estimates are correlated.

These types of designs are always an option regardless of the type of model the experimenter wishes to fit (for example, first order, first order plus some interactions, full quadratic, cubic, etc.) or the objective specified for the experiment (for example, screening, response surface, etc.). D-optimal designs are straight optimizations based on a chosen optimality criteria and the model that will be fit. The optimality criterion used in generating D-optimal designs is one of maximizing |X'X|, the determinant of the information matrix X'X.

This optimality criterion results in minimizing the generalized variance of the parameter estimates for a pre-specified model. As a result, the'optimality' of a given D-optimal design is model dependent. That is, the experimenter must specify a model for the design before a computer can generate the specific treatment combinations for the design. Given the total number of treatment runs for an experiment and a specified model, the computer algorithm chooses the optimal set of design runs from a *candidate set* of possible design treatment runs. This candidate set of treatment runs usually consists of all possible combinations of various factor levels that one wishes to use in the experiment.

In other words, the candidate set is a collection of treatment combinations from which the D-optimal algorithm chooses the actual treatment combinations to include in the design. The computer algorithm generally uses a stepping and exchanging process to determine which treatment runs the design will consist of.

Note: There is no guarantee that the resulting design the computer generates is actually optimal.

D-optimal designs are particularly useful when resources are limited or there are constraints on factor settings

The reasons for using D-optimal designs instead of standard classical designs generally fall into one of two categories:

1. standard factorial or fractional factorial designs require too many runsfor the amount of resources or time allowed for the experiment
2. the design space is constrained (the process space contains factor settings that are not feasible or are impossible to run).

Industry examples of these two situations are given below and the process flow of how to generateand analyze these types of designs is given. The software package used to demonstrate this is JMP version 3.2. The flow presented below in generating the design is the flow that is specified in the JMP Help screens under its D-optimal platform.

Examples of D-optimal designs

Suppose there are 3 design variables (k = 3), and engineering judgment specifies the following model as appropriate for the process under investigation

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{11} x_1^2$$

The levels being considered by the researcher are (coded)

X1: 5 levels (-1, -0.5, 0, 0.5, 1)
X2: 2 levels (-1, 1)
X3: 2 levels (-1, 1)

One design objective, due to resource limitations, is to use n = 12 design points.

Given the above experimental specifications, the first thing to do towards generating the design isto create the candidate set. The candidate set is a data table with a row for each point (run) you want considered for your design. This is often a full-factorial. You can create a candidate set in JMP by using the Full Factorial design given by the Design Experiment command in the Tables menu.The candidate set for this example is shown below. Since the candidate set is a full factorial in all factors, the candidate set contains (5)*(2)*(2)= 20 possible design runs.

TABLE 5.1 Candidate Set for Variables X1, X2, X3

X1	X2	X3	response
-1	-1	-1	?
-1	-1	1	?
-1	1	-1	?
-1	1	1	?
-0.5	-1	-1	?
-0.5	-1	1	?
-0.5	1	-1	?
-0.5	1	1	?
0	-1	-1	?
0	-1	1	?
0	1	-1	?
0	1	1	?
0.5	-1	-1	?
0.5	-1	1	?
0.5	1	-1	?
0.5	1	1	?
1	-1	-1	?
1	-1	1	?
1	1	-1	?
1	1	1	?

Once the candidate set has been created, specify the model you want in the Fit Model dialog. Do not give a response term for the model! Select D-Optimal as the fitting personality in the pop-up menu at the bottom of the dialog. Click Run Model and use the control panel that appears. Enter the number of runs you want in your design (N=12 in this example). You can also edit other options available in the control panel. This control panel and the editable options are shown in the table below. These other options refer to the number of points chosen at random at the start of an excursion or trip (N Random), the number of worst points at each K-exchange step or iteration (K-value), and the number of times to repeat the search (Trips). Click Go.

For this example, the table below shows what these options were set at and the reported *efficiency values* correspond to the best design found so far.

D-Optimal Control Panel
Optimal Design Controls

N Desired	12
N Random	3
K Value	2
Trips	3

Best Design

D-efficiency	68.2558
A-efficiency	45.4545
G-efficiency	100
AvgPredSE	0.6233
N	12.0000

The algorithm computes efficiency numbers to zero in on a D-optimal design

The four line efficiency report given after each search shows the best design over all the excursions (trips).D-efficiency is the objective. It is a volume criterion on the generalized variance of the estimates. The efficiency of the standard fractional factorial is 100%, but this is not possible when pure quadratic terms such as $X1^2$are included in the model.

The efficiency values are a function of the number of points in the design, the number of independent variables in the model, and the maximum standard error for prediction over the design points. The best design is the one with the highest D-efficiency. The A-efficiencies and G-efficiencies help choose an optimal design when multiple excursions produce alternatives with similar D-efficiency.

The search for a D-optimal design should be made using several excursions or trips. In each trip, JMP 3.2 chooses a different set of random seed points, which can possibly lead to different designs. The Save button saves the best design found. The standard error of prediction is also saved under the variable OptStdPred in the table.

The D-optimal design using 12 runs that JMP 3.2 created is listed below in standard order. The design runs should be randomized before the treatment combinations are executed.

TABLE 5.2 Final D-optimal Design

X1	X2	X3	y	OptStdPred
-1	-1	-1	?	0.645497
-1	-1	1	?	0.645497
-1	1	-1	?	0.645497
-1	1	1	?	0.645497
0	-1	-1	?	0.645497
0	-1	1	?	0.645497
0	1	-1	?	0.645497
0	1	1	?	0.645497
1	-1	-1	?	0.645497
1	-1	1	?	0.645497
1	1	-1	?	0.645497
1	1	1	?	0.645497

Parameter estimates are usually correlated

To see the correlations of the parameter estimates for the best design found, you can click on the Correlations button in the D-optimal Search Control Panel. In most D-optimal designs, the correlations among the estimates are non-zero. However, in this particular example, the correlations are zero.

Note: Other software packages (or even other releases of JMP) may have different procedures for generating D-optimal designs - the above example is a highly software dependent illustration of how to generate a D-optimal design.

5.5.2.2. Repairing a design

Sometimes a classical design needs to be "repaired" because certain runs were lost or impossible to achieve. Computer designs are useful for this.

Computer aided designs are helpful in either repairing or augmenting a current experimental design. They can be used to repair a 'broken' standard classical design.

There may occur situations where, due to improper planning or other issues, the original design matrix contains forbidden or unreachable combinations of the factor settings. A computer aided design (for example a D-optimal design) can be used to 'replace' those runs from the original design that were unattainable. The runs from the original design that are attainable are labeled as 'inclusion' runs and will be included in the final computer aided design.

Given a pre-specified model then the computer aided design can generate the additional attainable runs that are necessary, on top of the inclusion runs, in order to estimate the model of interest. As a result, the computer aided design is just 'replacing' those runs in the original design that were unattainable with a new set of runs that are attainable and still allows the experimenter to get information regarding the factors from the experiment.

The properties of this final design will probably not be as optimal as the original design and there may exist some correlation among the estimates. However, instead of not being able to use any of the data to analyze, generating the replacement runs from a computer aided design, a D-optimal design for example, allows one to have the option of being able to analyze the data. In this same way, computer aided designs can be used to augment a classical design with treatment combinations that will break alias chains among the terms in the model or permit the estimation of curvilinear effects.

5.5.3. How do you optimize a process?

Often the primary DOE goal is to find the operating conditions that maximize (or minimize) the system responses

How do you determine the optimal region to run a process?

The optimal region to run a process is usually determined after a sequence of experiments is conducted and a series of empirical models are obtained. In many engineering and science applications, experiments are conducted and empirical models are developed with the objective of improving the responses of interest. From a mathematical point of view, the objective is to find the operating conditions (or factor levels) X_1, X_2, ...,X_k that maximize or minimize the r system response variables Y_1, Y_2, ...,Y_r. In experimental optimization, different optimization techniques are applied to the *fitted* response equations $\hat{Y}_1, \hat{Y}_2, \ldots, \hat{Y}_r$. Provided that the fitted equations approximate adequately the true (unknown) system responses, the optimal operating conditions of the model will be "close" to the optimal operating conditions of the true system.

The DOE approach to optimization is to find approximate models and iteratively search for (near) optimal operating conditions

The experimental optimization of response surface models differs from classical optimization techniques in at least three ways:

1. Experimental optimization is an iterative process, that is, experiments conducted in one set of experiments result in fitted models that indicate where to search for improved operating conditions in the next set of experiments. Thus the coefficients of the fitted equations (or the form of the fitted equations) may change during the optimization process. This is in contrast to classical optimization where the functions to optimize are supposed to be fixed and given.

2. The response models are fitted from experimental data that usually contains random variability due to uncontrollable or unknown causes. This implies that an experiment, if repeated, will result in a different fitted response surface model that might lead to different optimal operating conditions. Therefore, sampling

Randomness (sampling variability)

effects the final answers and should be taken into account

variability should be considered in experimental optimization.

In contrast, in classical optimization techniques the functions are deterministic and given.

3. The fitted responses are local approximations, implying that the optimization process requires the input of the experimenter (a person familiar with the process). This is in contrast with classical optimization which is always automated in the form of some computer algorithm.

5.5.3.1. Single response case

The optimizing of a single response usually starts with line searches in the direction of maximum improvement. if there is lack of fit for linear models, quadratic models are next tried

The experimental optimization of a single response is usually conducted in two phases or steps, following the advise of Box and Wilson. The first phase consists of a sequence of line searches in the direction of maximum improvement. Each search in the sequence is continued until there is evidence that the direction chosen does not result in further improvements. The sequence of line searches is performed as long as there is no evidence of lack of fit for a simple first order model of the form

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + ... + b_k X_k$$

The second phase is performed when there is lack of linear fit in Phase I, and instead, a second order or quadratic polynomial regression model of the form

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + ... + b_k X_k + b_{11} X_1^2 + b_{22} X_2^2 + ... + b_{kk} X_{kk}^2 + b_{12} X_1 X_2 + b_{13} X_1 X_3 + ... + b_{1k} X_1 X_k + b_{23} X_2 X_3 + ... + b_{2k} X_2 X_k + ... + b_{k-1,k} X_{k-1} X_k$$

exhibits adequate fit. Not all responses will exhibit quadratic fit, and in such case Phase I is stopped when the response of interest can not be improved any further. Each phase is explained and illustrated in the next few sections.

"Flowchart" for two phases of experimental optimization

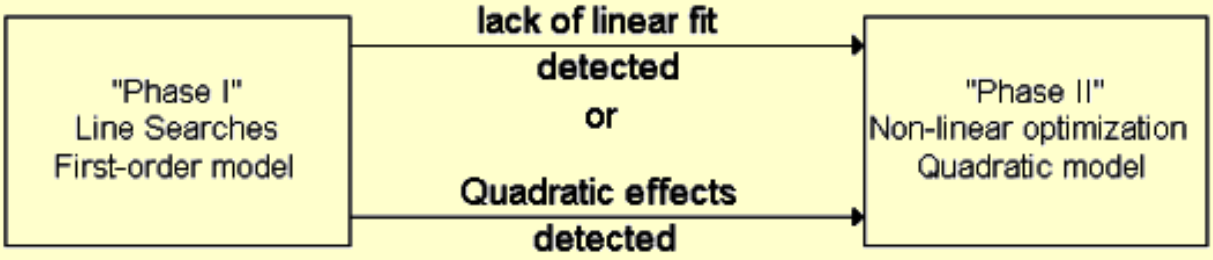


FIGURE 5.1: The Two Phases of Experimental Optimization

5.5.3.1.1. Single response: Path of steepest ascent

Starting at the current operating conditions, fit a linear model, determine the directions of steepest ascent and continue experimenting until no further improvement occurs - then iterate the process

If experimentation is initially performed in a new, poorly understood production process, chances are that the initial operating conditions $X_1, X_2, ...,X_k$ are located far from the region where the factors achieve a maximum or minimum for the response of interest **Y**. A first order model will serve as a good local approximation in a small region close to the initial operating conditions and far from where the process exhibits curvature. Therefore, it makes sense to fit a simple first order (or linear polynomial) model of the form:

$$\hat{Y} = b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k$$

Experimental strategies for fitting this type of models were discussed earlier. Usually, a 2^{k-p} fractional factorial experiment is conducted with repeated runs at the current operating conditions (which serve as the origin of coordinates in orthogonally coded factors).

The idea behind "Phase I" is to keep experimenting along the direction of steepest ascent (or descent, as required) until there is no further improvement in the response. At that point, a new fractional factorial experiment with center runs is conducted to determine a new search direction. This process is repeated until at some point significant curvature in $\hat{Y}$ is detected. This implies that the operating conditions $X_1, X_2, ...,X_k$ are close to where the maximum (or minimum, as required) of **Y** occurs. When significant curvature, or lack of fit, is detected, the experimenter should proceed with "Phase II". Figure 5.2 illustrates a sequence of line searches when seeking a region where curvature exists in a problem with 2 factors (i.e., **k=2**).

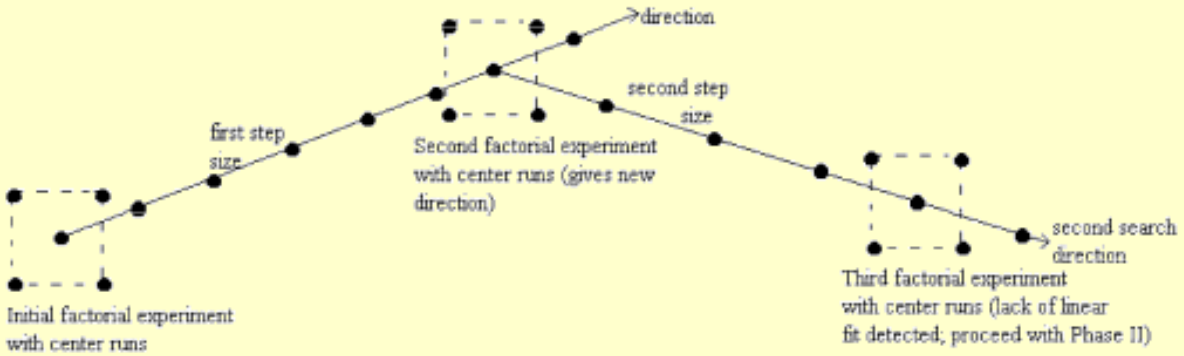


FIGURE 5.2: A Sequence of Line Searches for a 2 Factor Optimization Problem

There are two main decisions an engineer must make in Phase I:

1. determine the search direction;

2. determine the length of the step to move from the current operating conditions.

Figure 5.3 shows a flow diagram of the different iterative tasks required in Phase I. This diagram is intended as a guideline and should not be automated in such a way that the experimenter has no input in the optimization process.

Flow chart of iterative search process

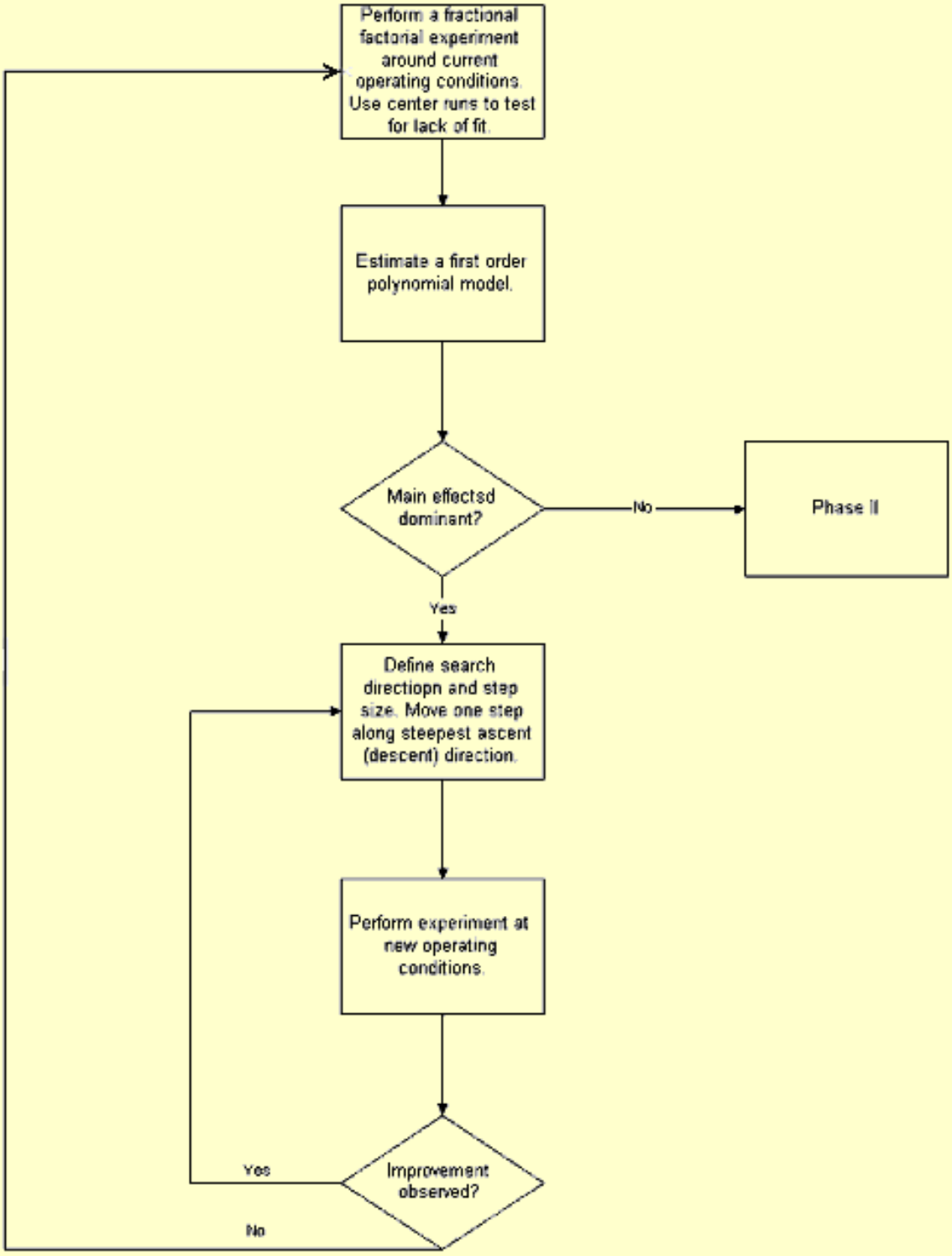


FIGURE 5.3: Flow Chart for the First Phase of the Experimental Optimization Procedure

The direction of steepest ascent is determined by the gradient of the fitted model and depends on the scaling convention - equal variance scaling is recommended

Procedure for Finding the Direction of Maximum Improvement.

Suppose a first order model like above has been fitted and provides a useful approximation. As long as lack of fit (due to pure quadratic curvature and interactions) is small compared to the main effects, steepest ascent can be attempted. To determine the direction of maximum improvement we use

1. the estimated direction of steepest ascent, given by the gradient of $\hat{Y}$, if the objective is to *maximize* **Y**;
2. the estimated direction of steepest descent, given by the negative of the gradient of $\hat{Y}$, if the objective is to *minimize* **Y**.

The direction of the gradient, g, is given by the values of the parameter estimates, that is, $g' = (b_1, b_2, \ldots, b_k)$. Since the parameter estimates $b_1, b_2, \ldots, b_k$ depend on the scaling convention for the factors, the steepest ascent (descent) direction is also scale dependent. That is, two experimenters using different scaling conventions will follow different paths for process improvement. This does not diminish the general validity of the method, since the region of the search, as given by the signs of the parameter estimates, does not change with scale. An equal variance scaling convention, however, is recommended. The coded factors x_i in terms of the factors in the original units of measurement, X_i, are obtained from the relation

$$x_i = \frac{X_i - (X_{low} + X_{high})/2}{(X_{high} - X_{low})/2} \quad i = 1, 2, \ldots, k$$

This coding convention is recommended since it provides better parameter estimates, and therefore, a more reliable search direction. The coordinates of the factor settings on the direction of steepest ascent separated a distance ρ from the origin are given by:

$$\text{maximize} \quad b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_k x_k$$

$$\text{subject to:} \quad \sum_{i=1}^{k} x_i^2 \leq \rho^2$$

This problem can be solved with the aid of an optimization solver (e.g. like the solver option of a spreadsheet). However, in this case this is not really needed, as the solution is a simple equation which yields the coordinates

$$x_i^* = \rho \frac{b_i}{\sqrt{\sum_{j=1}^{k} b_j^2}} \quad i = 1, 2, \ldots, k.$$

An engineer can compute this equation for different increasing values of ρ and get different factor settings all on the steepest ascent direction.

To see the details of why this equation is true, see Technical Appendix 5A.

Optimization by search example

Example. Optimization of a Chemical Process.

It has been concluded (perhaps after a factor screening experiment) that the yield (**Y**, in %) of a chemical process is mainly affected by the temperature (X_1, in °C) and by the reaction time (X_2, in minutes). Due to safety reasons, the region of operation is limited to

$$50 \le X_1 \le 250$$
$$150 \le X_2 \le 500$$

The process is currently run at a temperature of 200 °C and a reaction time of 200 minutes. A process engineer decides to run a 2^2 full factorial experiment with factor levels at

factor	*low*	*center*	*high*
X_1	170	200	230
X_2	150	200	250

Five repeated runs at the center levels are conducted to assess lack of fit. The orthogonally coded factors are

$$x_1 = \frac{X_1 - 200}{30} \quad \text{and} \quad x_2 = \frac{X_2 - 200}{50}.$$

The experimental results were:

x_1	x_2	X_1	X_2	Y(=yield)
-1	-1	170	150	32.79
1	-1	230	150	24.07
-1	1	170	250	48.94
1	1	230	250	52.49
0	0	200	200	38.89
0	0	200	200	48.29
0	0	200	200	29.68
0	0	200	200	46.50
0	0	200	200	44.15

The corresponding ANOVA table for a first order polynomial model, obtained using the DESIGN EASE statistical software, is

SOURCE	SUM OF SQUARES	DF	MEAN SQUARE	F VALUE	PROB>F
MODEL	503.3035	2	251.6517	4.810	0.0684
CURVATURE	8.1536	1	8.1536	0.1558	0.7093
RESIDUAL	261.5935	5	52.3187		
LACK OF FIT	37.6382	1	37.6382	0.6722	0.4583

```
  PURE ERROR    223.9553    4    55.9888

COR TOTAL       773.0506    8
```

It can be seen from the ANOVA table that there is no significant lack of linear fit due to an interaction term, and there is no evidence of curvature. Furthermore, there is evidence that the first order model is significant. Using the DESIGN EXPERT statistical software we obtain the resulting model (in the coded variables) as

$$\hat{Y} = 39.57 - 1.2925x_1 + 11.14x_2$$

Usual diagnostic checks show conformance to the regression assumptions, although the R^2 value is not very high: $R^2 = 0.6580$.

To *maximize* $\hat{Y}$, we use the direction of steepest ascent. The engineer selects $\rho = 1$, since a point on the steepest ascent direction distanced one unit (in the coded units) from the origin is desired. Then from the equation above for the predicted *Y* response, the coordinates of the factor levels for the next run are given by:

$$x_1^* = \frac{\rho b_1}{\sqrt{\sum_{j=1}^{2} b_j^2}} = \frac{(1)(-1.2925)}{\sqrt{(-1.2925)^2 + (11.14)^2}} = -0.1152$$

and

$$x_2^* = \frac{\rho b_2}{\sqrt{\sum_{j=1}^{2} b_j^2}} = \frac{(1)(11.14)}{\sqrt{(-1.2925)^2 + (11.14)^2}} = 0.9933.$$

This means that to improve the process, for every $(-0.1152)(30) = -3.456$°C that temperature is varied (decreased), the reaction time should be varied by $(0.9933)(50) = 49.66$ minutes.

==========

Details of how to determine the path of steepest ascent

Technical Appendix 5A: finding the factor settings on the steepest ascent direction a distance from origin

The problem of finding the factor settings on the steepest ascent/descent direction that are located at a distance ρ from origin is given by the optimization problem,

$$\text{maximize} \quad b_0 + \sum_{k} b_1 x_1 + b_2 x_2 + \ldots + b_k x_k$$

$$\text{subject to:} \quad \sum_{i=1} x_i^2 \leq \rho^2$$

To solve it, use a Lagrange multiplier. First, add a penalty λ for solutions not satisfying the constraint (since we want a direction of steepest ascent, we maximize, and therefore the penalty is negative. For steepest descent we minimize and the penalty term is added instead):

$$\text{maximize} \quad L = b'x - \lambda(x'x - \rho^2)$$

Compute the partials and equate them to zero

$$\frac{\partial L}{\partial x} = b - 2\lambda x = 0$$

$$\frac{\partial L}{\partial \lambda} = -(x'x - \rho^2) = 0$$

These two equations have two unknowns (the vector x and the scalar λ) and thus can be solved yielding the desired solution:

$$x^* = \rho \frac{b}{||b||}$$

or, in non-vector notation:

$$x_i^* = \rho \frac{b_i}{\sqrt{\sum_{j=1}^{k} b_j^2}} \quad i = 1, 2, \ldots, k.$$

From this equation, we can see that any multiple ρ of the direction of the gradient (given by $b/||b||$) will lead to points on the steepest ascent direction. For steepest descent, use instead $-b_i$ in the numerator of the equation above.

5.5.3.1.2. Single response: Confidence region for search path

"Randomness" means that the steepest ascent direction is just an estimate and it is possible to construct a confidence "cone' around this direction estimate

The direction given by the gradient $g' = (b_0, b_1, \ldots, b_k)$ constitutes only a single (point) estimate computed based on a sample of **N** runs. If a different set of **N** runs is conducted, these will provide different parameter estimates, which in turn will give a different gradient. To account for this sampling variability, Box and Draper give a formula for constructing a "cone" around the direction of steepest ascent that with certain probability contains the true (unknown) system gradient given by $(\beta_1, \beta_2, \ldots, \beta_k)$. The width of the confidence cone is useful to assess how reliable an estimated search direction is.

Figure 5.4 shows such a cone for the steepest ascent direction in an experiment with two factors. If the cone is so wide that almost every possible direction is inside the cone, an experimenter should be very careful in moving too far from the current operating conditions along the path of steepest ascent or descent. Usually this will happen when the linear fit is quite poor (i.e., when the R^2 value is low). Thus plotting the confidence cone is not so important as computing its width.

If you are interested in the details on how to compute such a cone (and its width), see Technical Appendix 5B.

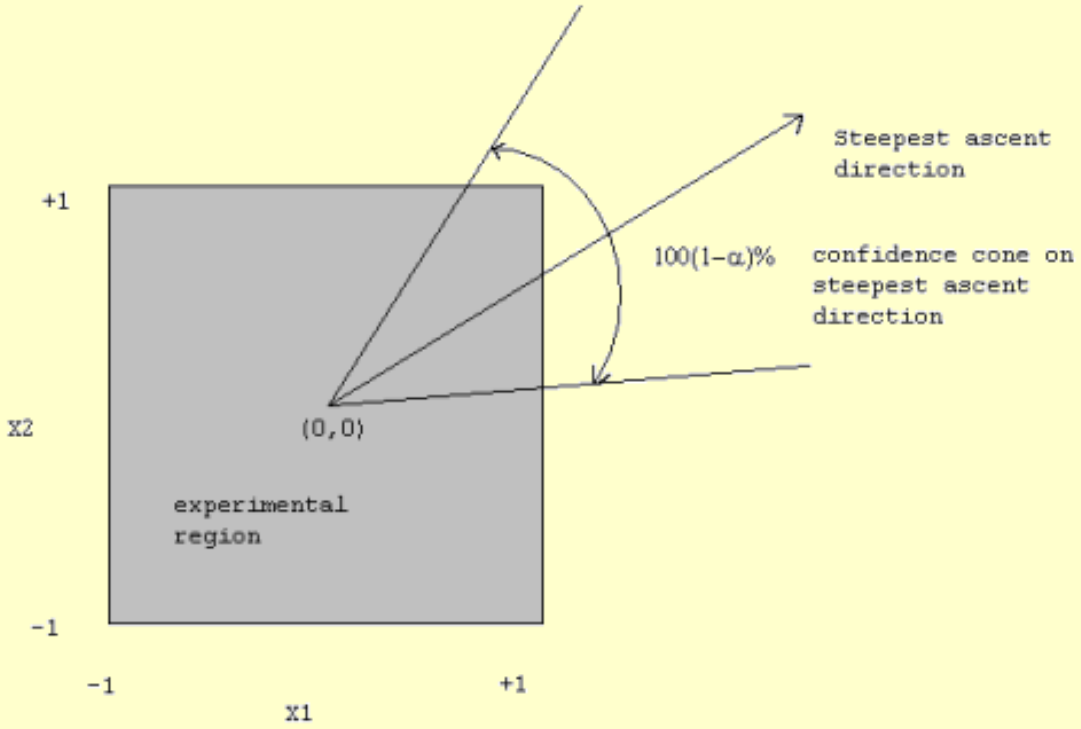


FIGURE 5.4: A Confidence Cone for the Steepest Ascent Direction in an Experiment with 2 Factors

==========

Details of how to construct a confidence cone for the direction of steepest ascent

Technical Appendix 5B: Computing a Confidence Cone on the Direction of Steepest Ascent.

Suppose the response of interest is adequately described by a first order polynomial model. Consider the inequality

$$\sum_{i=1}^{k} b_i^2 - \frac{\left(\sum_{i=1}^{k} b_i x_i\right)^2}{\sum_{i=1}^{k} x_i^2} \leq (k-1) s_b^2 F_{\alpha,k-1,n-p}$$

where $s_b^2 = \text{SS}_{\text{error}} C_{jj}/(n-p)$, C_{jj} is the j-th diagonal element of the matrix $(\boldsymbol{X}'\boldsymbol{X})^{-1}$ (for $j = 1, \ldots, k$ these values are all equal if the experimental design is a 2^{k-p} factorial of at least Resolution III), and $\boldsymbol{X}$ is the design matrix of the experiment (including columns for the intercept and 2nd order terms, if any). Any operating conditions with coordinates $\boldsymbol{x}' = (x_1, x_2, \ldots, x_k)$ that satisfy this inequality generates a direction that lies within the $100(1-\alpha)$% confidence cone of steepest *ascent* if

$$\sum_{i=1}^{k} b_i x_i > 0$$

or inside the $100(1-\alpha)$% confidence cone of steepest *descent* if

$$\sum_{i=1}^{k} b_i x_i < 0.$$

The inequality defines a cone with the apex at the origin and center line located along the gradient of $\hat{Y}$.

A measure of "goodness" of a search direction is given by the fraction of directions *excluded* by the $100(1-\alpha)$% confidence cone around the steepest ascent/descent direction (see Box and Draper, 1987) which is given by:

$$\theta_\alpha = 1 - \phi_\alpha = 1 - T_{k-1}\left(\frac{\sum_{i=1}^{k} b_i^2}{s_b^2 F_{\alpha,k-1,n-p}} - (k-1)\right)^{1/2}$$

where $T_{k-1}()$ denotes the complement of the Student t distribution function with ***k*-1** degrees of freedom (that is, $T_{k-1}(x) = P(t_{k-1} \geq x)$) and $F_{\alpha,k-1,n-p}$ denotes an α percentage point of the F distribution with **k-1** and **n-p** degrees of freedom, where **n-p** is the error degrees of freedom. The value of ϕ_α represents the fraction of directions included by the confidence cone. The smaller θ_α is, the wider the cone is, with $0 \leq \theta_\alpha \leq 1$. Note that the inequality equation and the "goodness measure" equation are valid when operating conditions are given in coded units.

Example: Computing θ_α.

From the ANOVA table in the chemical experiment discussed earlier

$$s_b^2 = (52.3187)(1/4) = 13.0796$$

since $C_{jj} = 1/4$ (**j=2,3**) for a 2^2 factorial. The fraction of directions excluded by a 95 % confidence cone in the direction of steepest ascent is:

$$\theta_{0.05} = 1 - T_1\left(\frac{(-1.2925)^2 + (11.1425)^2}{(13.0796)(5.99)} - 1\right)^{0.5}$$
$$= 1 - 0.2894 = 0.7105$$

since $F_{0.05,1,6} = 5.99$. Thus 71.05 % of the possible directions from the current operating point are excluded with 95% confidence. This is useful information that can be used to select a step length. The smaller θ_α is, the shorter the step should be, as the steepest ascent direction is less reliable. In this example, with high confidence, the true steepest ascent direction is within this cone of 29% of possible directions. For k=2, 29% of 360° = 104.4°, so we are 95% confident that our estimated steepest ascent path is within plus or minus 52.2° of the true steepest path. In this case, we should not use a large step along the estimated steepest ascent path.

5.5.3.1.3. Single response: Choosing the step length

A procedure for choosing how far along the direction of steepest ascent to go for the next trial run

Once the search direction is determined, the second decision needed in Phase I relates to how far in that direction the process should be "moved". The most common procedure for selecting a step length is based on choosing a step size in one factor and then compute step lengths in other factors proportional to their parameter estimates. This provides a point on the direction of maximum improvement. The procedure is given below. A similar approach is obtained by choosing increasing values of ρ in $x_i^* = \rho \frac{b_i}{\sqrt{\sum_{j=1}^{k} b_j^2}} \quad i = 1, 2, ..., k.$

However, the procedure below considers the original units of measurement which are easier to deal with than the coded "distance" ρ.

Procedure: selection of step length.

1. Choose a step length ΔX_j (in natural units of measurement) for some factor **j**. Usually, factor **j** is chosen to be the one engineers feel more comfortable varying, or the one with the largest $|b_j|$. The value of ΔX_j can be based on the width of the confidence cone around the steepest ascent/descent direction . Very wide cones indicate that the estimated steepest ascent/descent direction is not reliable, and thus ΔX_j should be small. This usually occurs when the R^2 value is low. In such case, prior to moving from the current experimental region, additional experiments can be conducted to obtain a better model fit and a better search direction.

2. Transform to coded units: $\Delta x_j = \frac{\Delta X_j}{s_j}$ where s_j is the scale factor used for factor **j** (e.g. $s_j = \text{range}_j/2$).
3. Set $\Delta x_i = \frac{b_i}{b_j}\Delta x_j$ for all other factors **i**.
4. Transform all the Δx_i's to natural units: $\Delta X_i = (\Delta x_i)(s_i)$.

An example

Example: Step Length Selection.

- For the chemical process experiment described previously, the process engineer selected $\Delta X_2 = 50$ minutes. This was based on process engineering considerations. It was also felt that $\Delta X_2 = 50$ does not move the process too far away from the current region of experimentation. This was desired since the R^2 value of 0.6580 for the fitted model is quite low, providing a not very reliable steepest ascent direction (and a wide confidence cone, see Technical Appendix 5B).
- $\Delta x_2 = \frac{50}{50} = 1.0$.
- $\Delta x_1 = \frac{-1.2925}{11.14} = -0.1160$.
- $\Delta X_2 = (-.1160)(30) = -3.48$ °C .

Thus the step size is $\Delta X' = (-3.48$ °C , **50 minutes**).

Procedure: Conducting Experiments Along the Direction of Maximum Improvement.

1. Given current operating conditions $X_0' = (X_1, X_2, \ldots, X_k)$ and a step size $\Delta X' = (\Delta X_1, \Delta X_2, \ldots, \Delta X_k)$, perform experiments at factor levels $X_0 + \Delta X, X_0 + 2\Delta X, X_0 + 3\Delta X, \ldots$ as long as improvement in the response **Y** (decrease or increase, as desired) is observed.
2. Once a point has been reached where there is no further improvement, a new first order experiment (e.g. a 2^{k-p} fractional factorial) should be performed with repeated center runs to assess lack of fit. If there is no significant evidence of lack of fit, the new first order model will provide a new search

direction, and another iteration is performed as indicated in Figure 5.3. Otherwise (there is evidence of lack of fit) the experimental design is augmented and a 2nd order model should be fitted. That is, the experimenter should proceed to "Phase II".

Example: Experimenting Along the Direction of Maximum Improvement.

Step 1:

Given $X_0 = (200^{\circ}\text{C}$, **200** minutes) and $\Delta X = (-3.48^{\circ}\text{C}$, **50** minutes),

next experiments were performed as follows (the step size in temperature was rounded to -3.5 $^{\circ}$C for practical reasons):

	X_1	X_2	x_1	x_2	Y(=yield)
X_0	200	200	0	0	
$X_0 + \Delta X$	196.5	250	-0.1160	1	56.2
$X_0 + 2\Delta X$	193.0	300	-0.2333	2	71.49
$X_0 + 3\Delta X$	189.5	350	-0.3500	3	75.63
$X_0 + 4\Delta X$	186.0	400	-0.4666	4	72.31
$X_0 + 5\Delta X$	182.5	450	-0.5826	5	72.10

Since the goal is to maximize **Y**, the point of maximum observed response is $X_1 = 189.5^{\circ}$C , $X_2 = 350$ minutes. Notice that the search was stopped after 2 consecutive drops in response, to assure we have passed by the "peak" of the "hill".

Step 2:

A new 2^2 factorial experiment is performed with $X' = (189.5, 350)$ as the origin. Using the same scaling factors as before, the new scaled controllable factors are:

$$x_1 = \frac{X_1 - 189.5}{30} \quad \text{and} \quad x_2 = \frac{X_2 - 350}{50}$$

Five center runs (at $X_1 = 189.5, X_2 = 350$) were repeated to assess lack of fit. The experimental results were:

x_1	x_2	X_1	X_2	Y(=yield)
-1	-1	159.5	300	64.33
1	-1	219.5	300	51.78
-1	1	158.5	400	77.30
1	1	219.5	400	45.37
0	0	189.5	350	62.08
0	0	189.5	350	79.36
0	0	189.5	350	75.29
0	0	189.5	350	73.81
0	0	189.5	350	69.45

The corresponding ANOVA table for a linear model, obtained using the DESIGN EASE statistical software, is

SOURCE	SUM OF SQUARES	DF	MEAN SQUARE	F VALUE	PROB > F
MODEL	505.300	2	252.650	4.731	0.0703
CURVATURE	336.309	1	336.309	6.297	0.0539
RESIDUAL	267.036	5	53.407		
LACK OF FIT	93.857	1	93.857	2.168	0.2149
PURE ERROR	173.179	4	43.295		
COR TOTAL	1108.646	8			

From the table, the linear effects (model) are significant and there is no evidence of lack of fit. However, there is a significant curvature effect (at the 5.4% significance level), which implies that the optimization should proceed with Phase II, that is, the fit and optimization of a second order model.

5.5.3.1.4. Single response: Optimization when there is adequate quadratic fit

Regions where quadratic models or even cubic models are needed occur in many cases in industry

After a few steepest ascent (or descent) searches a first order model will eventually lead to no further improvement or it will exhibit lack of fit. The latter case typically occurs when operating conditions have been changed to a region where there are quadratic (second order) effects present in the response. A second order polynomial can be used as a local approximation of the response in a small region where, hopefully, optimal operating conditions exist. However, while a quadratic fit is appropriate in most of the cases in industry, there will be a few times when a quadratic fit will not be sufficiently flexible to explain a given response. In such cases the analyst generally does one of the following:

1. Uses a transformation of Y or the X_i's to improve the fit.
2. Limits use of the model to a smaller region in which the model does fit.
3. Adds other terms to the model.

Procedure: obtaining the estimated optimal operating conditions

Once a linear model exhibits lack of fit or when significant curvature is detected, the experimental design used in Phase I (recall that a 2^{k-p} factorial experiment was recommended) should be augmented with axial runs on each factor to form what is called a central composite design. This experimental design allows estimation of a second order polynomial of the form

Steps to find optimal operating conditions

$$\hat{Y} = b_0 + \sum_{i=1}^{k} b_i x_i + \sum_{i=1}^{k} b_{ii} x_{ii}^2 + \sum_{i<j}^{k} \sum_{j=1}^{k} b_{ij} x_i x_j$$

If the corresponding analysis of variance table indicates no lack of fit for this model, the engineer can proceed to determine the estimated optimal operating conditions.

1. Using some graphics software, get a contour plot of the fitted response. If the number of factors (**k**) is greater than 2, then plot contours in all planes corresponding to all the possible pairs of factors. For **k** greater than, say, 5, this could be too cumbersome (unless the graphic software plots all pairs automatically). In such case a "canonical analysis" of the surface is recommended (see Technical Appendix 5 D).
2. Use an optimization solver to maximize or minimize (as desired) the estimated response $\hat{Y}$.
3. Perform a confirmation experiment at the estimated optimal operating conditions given by the solver in step 2.

Example

We illustrate these steps with the DESIGN EXPERT software and our chemical experiment discussed before. For a technical description of a formula that provides the coordinates of the stationary point of the surface, see Technical Appendix 5C.

Example: Second Phase Optimization of Chemical Process.

Recall that in the chemical experiment, the ANOVA table, obtained from using an experiment run around the coordinates $X_1 = 189.5, X_2 = 350$, indicated significant curvature effects.

Augmenting the 2^2 factorial experiment with axial runs at $\pm\alpha = \pm\sqrt{2}$ to achieve a rotatable central composite experimental design, the following experimental results were obtained:

x_1	x_2	X_1	X_2	Y(=yield)
-1.414	0	147.08	350	72.58
1.414	0	231.92	350	37.42
0	-1.414	189.5	279.3	54.63
0	1.414	189.5	420.7	54.18

The corresponding ANOVA table for the different effects, based on the sequential sum of squares procedure of the DESIGN EXPERT software, gives

SOURCE	SUM OF SQUARES	DF	MEAN SQUARE	F VALUE	PROB > F
MEAN	51418.2	1	51418.2		
Linear	1113.7	2	556.8	5.56	0.024
Quadratic	768.1	3	256.0	7.69	0.013
Cubic	9.9	2	5.0	0.11	0.897
RESIDUAL	223.1	5	44.6		
TOTAL	53533.0	13			

From the table, the linear and quadratic effects are significant. The lack of fit tests and auxiliary diagnostic statistics are:

MODEL	SUM OF SQUARES	DF	MEAN SQUARE	F VALUE	PROB > F
Linear	827.9	6	138.0	3.19	0.141
Quadratic	59.9	3	20.0	0.46	0.725
Cubic	49.9	1	49.9	1.15	0.343
PURE ERROR	173.2	4	43.3		

SOURCE	ROOT MSE	R-SQR	ADJ R-SQR	PRED R-SQR	PRESS
Linear	10.01	0.5266	0.4319	0.2425	1602.02
Quadratic	5.77	0.8898	0.8111	0.6708	696.25
Cubic	6.68	0.8945	0.7468	-0.6393	3466.71

The quadratic model has a larger *p*-value for the lack of fit test, higher adjusted R^2, and a lower

PRESS statistic, thus it should provide a reliable model. The fitted quadratic equation, in coded units, is

$$\hat{Y} = 72.0 - 11.78x_1 + 0.74x_2 - 7.25x_1^2 - 7.55x_2^2 - 4.85x_1x_2$$

Step 1:

A contour plot of this function (Figure 5.5) shows that it appears to have a single optimum point in the region of the experiment (this optimum is calculated below to be (-.9285,.3472), in coded x_1, x_2 units, with a response value of 77.59).

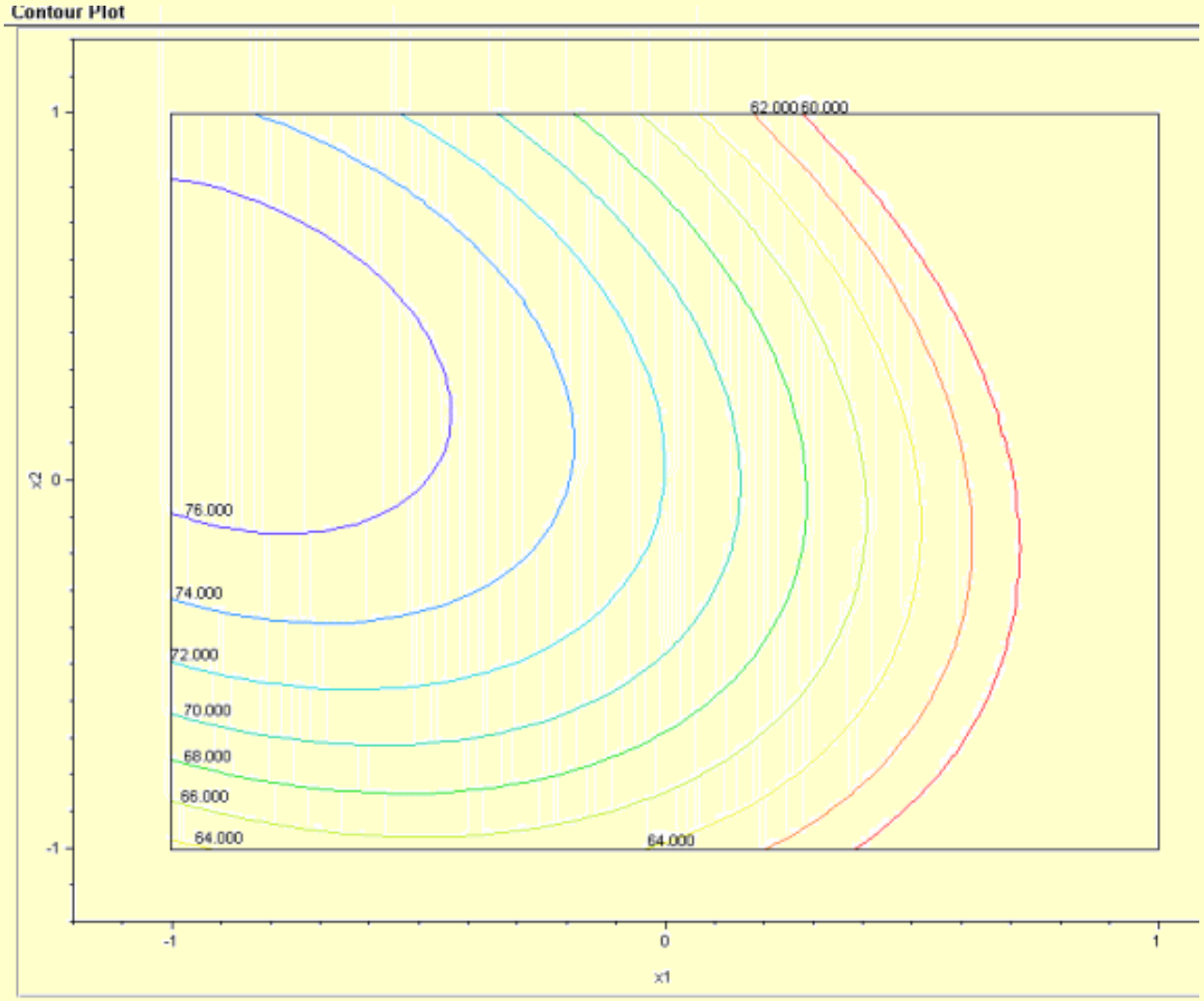


FIGURE 5.5: Contour Plot of the Fitted Response in the Example

Since there are only two factors in this example, we can also get a 3D plot of the fitted response against the two factors (Figure 5.6).

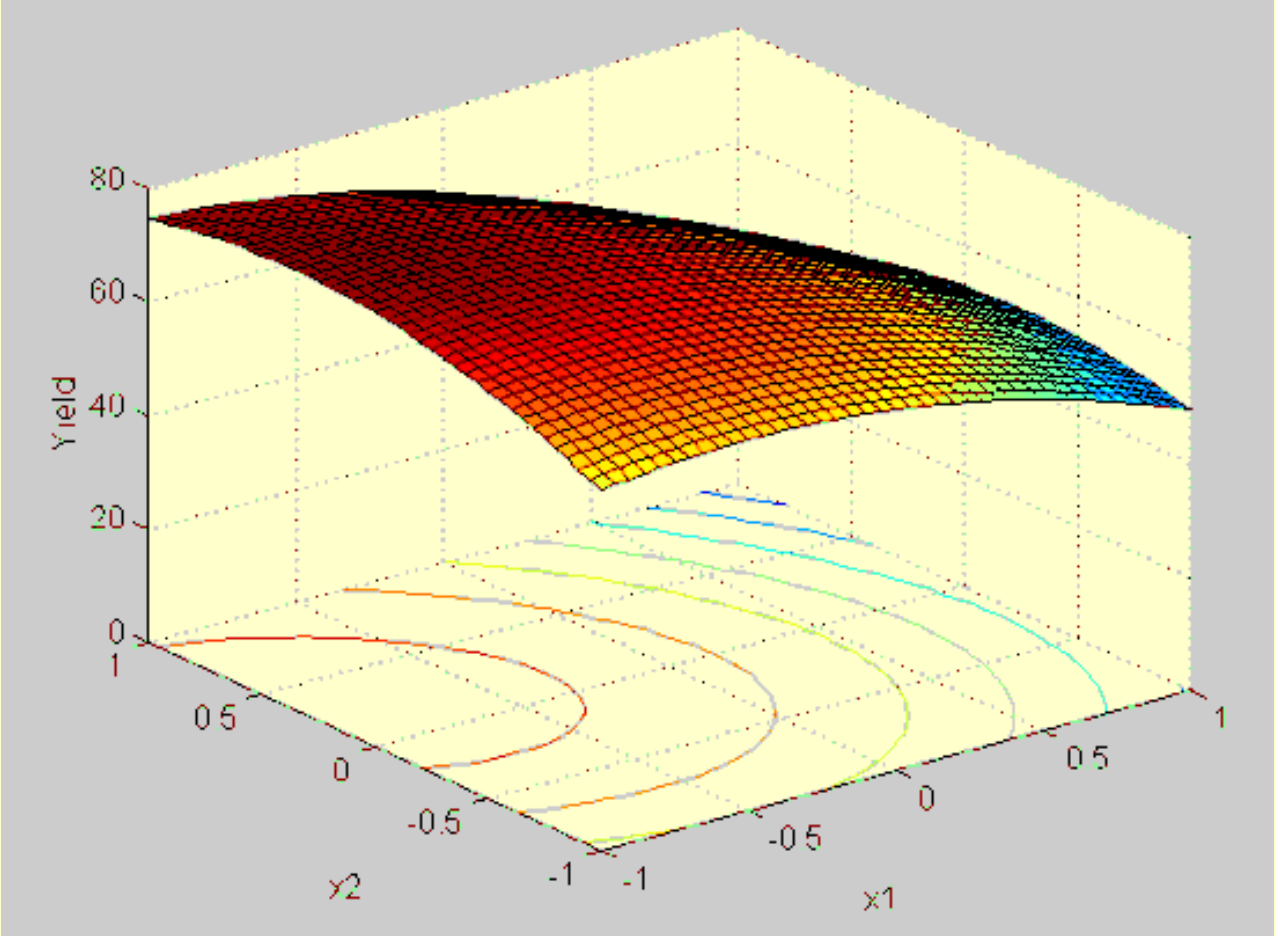


FIGURE 5.6: 3D Plot of the Fitted Response in the Example

Step 2:

The optimization routine in DESIGN EXPERT was invoked for maximizing $\hat{Y}$. The results are $X_1^* = 161.64$°C , $X_2^* = 367.32$ minutes. The estimated yield at the optimal point is $\hat{Y}(X^*) = 77.59$ %.

Step 3:

A confirmation experiment was conducted by the process engineer at settings $X_1 = 161.64$, $X_2 = 367.32$. The observed response was $Y(X^*) = 76.5$ % which is satisfactorily close to the estimated optimum.

==========

Details of how to find the maximum or minimum point for a quadratic response

Technical Appendix 5C: Finding the Factor Settings for the Stationary Point of a Quadratic Response.

1. Rewrite the fitted equation using matrix notation as

$$\hat{Y}(x) = b_o + b'x + x'Bx$$

where $b' = (b_1, b_2, \ldots, b_k)$ is a vector of first order parameter estimates,

$$B = \begin{pmatrix} b_{11} & b_{12}/2 & \cdots & b_{ik}/2 \\ & b_{22} & & \\ & & \ddots & \vdots \\ \text{symmetric} & & & b_{kk} \end{pmatrix}$$

is a matrix of second-order parameter estimates and $x' = (x_1, x_2, \ldots, x_k)$ is the vector of controllable factors. Notice that the off diagonal elements of B are equal to half the two-factor interaction coefficients.

2. Equating the partial derivatives of $\hat{Y}$ with respect to x to zeroes and solving the resulting system of equations, the coordinates of the stationary point of the response are given by

$$x^* = -\frac{1}{2}B^{-1}b$$

The nature of the stationary point (whether it is a point of maximum response, minimum response, or a saddle point) is determined by the matrix B. The two-factor interactions do not, in general, let us "see" what type of point x^* is. One thing that can be said is that if the diagonal elements of B (the b_{ii}) have mixed signs, x^* *is* a saddle point. Otherwise, it is necessary to look at the characteristic roots or eigenvalues of B to see whether B is "positive definite" (so x^* is a point of minimum response) or "negative definite" (the case where x^* is a point of maximum response). This task is easier if the two-factor interactions are "eliminated" from the fitted equation as is described in Technical Appendix 5D..

Example: computing the stationary point, Chemical Process experiment.

The fitted quadratic equation in the chemical experiment discussed in Section 5.5.3.1.1, in coded units, is

$$\hat{Y} = 72.0 - 11.78x_1 + 0.74x_2 - 7.25x_1^2 - 7.55x_2^2 - 4.85x_1x_2$$

from which we obtain $b' = (-11.78, 0.74)$,

$$\boldsymbol{B} = \begin{pmatrix} -7.25 & -2.425 \\ -2.425 & -7.55 \end{pmatrix} \quad ; \quad \boldsymbol{B}^{-1} = \begin{pmatrix} -0.1545 & 0.0496 \\ 0.0496 & -0.1483 \end{pmatrix}$$

and

$$\boldsymbol{x}^* = -\frac{1}{2}\begin{pmatrix} -0.1545 & 0.0496 \\ 0.0496 & -0.1483 \end{pmatrix}\begin{pmatrix} -11.78 \\ 0.74 \end{pmatrix} = \begin{pmatrix} -0.9285 \\ 0.3472 \end{pmatrix}$$

Transforming back to the original units of measurement, the coordinates of the stationary point are

$$\boldsymbol{X}^* = \begin{pmatrix} 161.64\ ^\circ\mathrm{C} \\ 367.36\ \mathrm{min.} \end{pmatrix}.$$

Notice this is the same solution as obtained by using the optimization routine of DESIGN EXPERT (see section 5.5.3.1.1). The predicted response at the stationary point is $\hat{Y}(\boldsymbol{X}^*) = 77.59$ %.

Technical Appendix 5D: "Canonical Analysis" of Quadratic Responses.

Whether the stationary point $\boldsymbol{X}^*$ represents a point of maximum or minimum response, or just a saddle point, is determined by the matrix of second order coefficients, $\boldsymbol{B}$. In the simpler case of just a single controllable factor (**k=1**), $\boldsymbol{B}$ is a scalar proportional to the second derivative of $\hat{Y}(x)$ with respect to **x**. If $d^2\hat{Y}/dx^2$ is positive, recall that from calculus that the function $\hat{Y}(x)$ is convex ("bowl shaped") and then $\boldsymbol{x}^*$ is a point of minimum response.

Unfortunately, the multiple factor case (**k>1**) is not so easy since the two factor interactions (the off diagonal elements of $\boldsymbol{B}$) obscure the picture of what is going on. A recommended procedure for analyzing whether $\boldsymbol{B}$ is "positive definite" (we have a min.) or "negative definite" (we have a max.) is to rotate the axes $x_1, x_2, \ldots, x_k$ so that the two factor interactions disappear. It is also customary (Box and Draper, 1987; Khuri and Cornell, 1987; Myers and Montgomery, 1995) to translate the origin of coordinates to the stationary point, so that the intercept terms are eliminated from the equation of $\hat{Y}(\boldsymbol{x})$. This procedure is called the canonical analysis of $\hat{Y}(\boldsymbol{x})$.

Procedure: Canonical Analysis

1. Define new axis $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{x}^*$ (translation step). The fitted equation becomes
$$\hat{Y}(\boldsymbol{z}) = \hat{Y}(\boldsymbol{x}^*) + \boldsymbol{z}'\boldsymbol{B}\boldsymbol{z}.$$
2. Define new axis $\boldsymbol{w} = \boldsymbol{E}'\boldsymbol{z}$, where $\boldsymbol{E}'\boldsymbol{B}\boldsymbol{E} = \boldsymbol{D}$ and $\boldsymbol{D}$ is a diagonal matrix to be defined (rotation step). The fitted equation becomes $\hat{Y}(\boldsymbol{w}) = \hat{Y}(\boldsymbol{x}^*) + \boldsymbol{w}'\boldsymbol{D}\boldsymbol{w}$. This

is the so-called canonical form of the model. The elements on the diagonal of $\boldsymbol{D}$, $\lambda_i (i = 1, 2, \ldots, k)$ are the eigenvalues of $\boldsymbol{B}$. The columns of $\boldsymbol{E}'$, $\boldsymbol{e}_i$, are the *orthonormal eigenvectors* of $\boldsymbol{B}$, which means that the $\boldsymbol{e}_i$ satisfy $(\boldsymbol{B} - \lambda_i \boldsymbol{I})\boldsymbol{e}_i = 0$, $\boldsymbol{e}_i'\boldsymbol{e}_j = 0$ for $i \neq j$, and $\boldsymbol{e}_i'\boldsymbol{e}_i = 1.0$.

3. If all the λ_i are negative, $\boldsymbol{x}^*$ corresponds to a point of maximum response. If all λ_i are positive, $\boldsymbol{x}^*$ corresponds to a point of minimum response. Finally, if the λ_i are of mixed signs, the response is a saddle function and $\boldsymbol{x}^*$ is the saddle point.

If some $\lambda_1 \approx 0$, the fitted ellipsoid $\hat{Y}(\boldsymbol{w}) = \hat{Y}(\boldsymbol{x}^*) + \sum_{i=1}^{k} \lambda_i w_i^2$ is elongated (i.e., it is flat) along the direction of the w_i axis. Points along the w_i axis will have an estimated response close to optimal, thus the process engineer has flexibility in choosing "good" operating conditions. If two eigenvalues (say λ_i and λ_j) are close to zero, then a plane in the (w_i, w_j) coordinates will have close to optimal operating conditions, etc.

It is nice to know that the JMP or SAS software (PROC RSREG) computes the eigenvalues λ_i and the orthonormal eigenvectors $\boldsymbol{e}_i$, thus there is no need to do a canonical analysis by hand.

Example: Canonical Analysis of Yield Response in Chemical Experiment using SAS.

Let us return to the chemical experiment example. This will serve us for illustration of the method, but keep in mind that when the number of factors is small (e.g., **k=2** as in this example) canonical analysis is not recommended in practice since simple contour plotting will provide sufficient information. The fitted equation of the model yields

$$\boldsymbol{B} = \begin{pmatrix} -7.25 & -2.425 \\ -2.425 & -7.55 \end{pmatrix}$$

To compute the eigenvalues λ_i, we have to find all roots of the expression that results from equating the determinant of $\boldsymbol{B} - \lambda \boldsymbol{I}$ to zero. Since $\boldsymbol{B}$ is symmetric and has real coefficients, there will be **k** real roots $\lambda_i, i = 1, 2, \ldots, k$. To find the orthonormal eigenvectors, solve the simultaneous equations $(\boldsymbol{B} - \lambda_i \boldsymbol{I})\boldsymbol{e}_i = 0$ and $\boldsymbol{e}_i'\boldsymbol{e}_i = 1$. This is the hard way, of course. These computations are easily done using the SAS software PROC RSREG. The SAS program applied to our example is:

```
data;
input x1 x2 y;
cards;
-1    -1     64.33
 1    -1     51.78
-1     1     77.30
```

```
 1     1     45.37
 0     0     62.08
 0     0     79.36
 0     0     75.29
 0     0     73.81
 0     0     69.45
-1.414 0     72.58
 1.414 0     37.42
 0    -1.414 54.63
 0     1.414 54.18
;
proc rsreg;
model y=x1 x2 /nocode/lackfit;
run;
```

The "nocode" option was entered since the factors were already input in coded form. The corresponding output from SAS canonical analysis is as follows:

Canonical Analysis of Response Surface

Factor	Critical Value
X1	-0.922
X2	0.346800

Predicted value at stationary point 77.589146

Eigenvalues	Eigenvectors X1	Eigenvectors X2
-4.973187	0.728460	-0.685089
-9.827317	0.685089	0.728460

Stationary point is a maximum.

Notice that the eigenvalues are the two roots of $\det(\boldsymbol{B} - \lambda \boldsymbol{I}) = (-7.25\lambda)(-7.55 - \lambda) - (-2.425(-2.245)) = 0$. As mentioned previously, the stationary point is $\boldsymbol{x}^{*'} = (-0.9278, 0.3468)$ which corresponds to $\boldsymbol{X}^{*} = (161.64, 367.36)$. Since both eigenvalues are negative, $\boldsymbol{x}^{*}$ is a point of maximum response. To get the directions of the axis of the fitted ellipsoid, compute

$$w_1 = 0.7284(x_1 + 0.9278) - 0.6850(x_2 - 0.3468) = 0.9143 + 0.7286x_1 - 0.6850x_2$$

and

$$w_2 = 0.6850(x_1 + 0.9278) - 0.7284(x_2 - 0.3468) = 0.3530 + 0.6849x_1 + 0.7254x_2$$

Since $|\lambda_1| < |\lambda_2|$, there is somewhat more elongation in the w_1 direction. However, since both eigenvalues are quite far from zero, there is not much flexibility in choosing operating conditions. It

can be seen from Figure 5.5 that the fitted ellipses do not have a great elongation in the w_1 direction, the direction of the major axis. It is important to emphasize that confirmation experiments at x^* should be performed to check the validity of the estimated optimal solution.

5.5.3.1.5. Single response: Effect of sampling error on optimal solution

Experimental error means all derived optimal operating conditions are just estimates - confidence regions that are likely to contain the optimal points can be derived and confirmation runs are very important

Process engineers should be aware that the estimated optimal operating conditions $\mathbf{x}^*$ represent a single estimate of the true (unknown) system optimal point. That is, due to sampling (experimental) error, if the experiment is repeated, a different quadratic function will be fitted which will yield a different stationary point $\mathbf{x}^*$. Some authors (Box and Hunter, 1954; Myers and Montgomery, 1995) provide a procedure that allows to compute a region in the factor space that, with certain probability, contains the system stationary point. This region is useful information for a process engineer in that it provides a measure of how "good" the point estimate $\mathbf{x}^*$ is. In general, the larger this region is, the less reliable the point estimate $\mathbf{x}^*$ is. When the number of factors, **k**, is greater than 3, these confidence regions are difficult to visualize.

Awareness of experimental error should make a process engineer realize the importance of performing confirmation runs at $\mathbf{x}^*$, the estimated optimal operating conditions.

5.5.3.1.6. Single response: Optimization subject to experimental region constraints

Ridge analysis is a method of finding optimal factor settings that satisfy certain constraints

Sometimes the optimal operating conditions $\mathbf{x}^*$ simply fall outside the region where the experiment was conducted. In these cases constrained optimization techniques can be used to find the solution $\mathbf{x}^*$ that optimizes $\hat{Y}(\mathbf{x})$ without leaving the region in the factor space where the experiment took place.

"Ridge Analysis", as developed by Hoerl (1959), Hoerl (1964) and Draper (1963), is an optimization technique that finds factor settings $\mathbf{x}^*$ such that they

$$\begin{aligned} &\text{optimize} \quad \hat{Y}(\mathbf{x}) = b_o + \mathbf{b}'\mathbf{x} + \mathbf{x}'\mathbf{B}\mathbf{x} \\ &\text{subject to:} \quad \mathbf{x}'\mathbf{x} = \rho^2 \end{aligned}$$

The solution $\mathbf{x}^*$ to this problem provides operating conditions which yield an estimated absolute maximum or minimum response on a sphere of radius ρ. Different solutions can be obtained by trying different values of ρ.

The original formulation of Ridge Analysis was based on the eigenvalues of a stationarity matrix. With the wide availability of non-linear programming codes, Ridge Analysis problems can be solved without recourse to eigenvalue analysis. Ridge Analysis is a particular case of dual response systems, which we analyze later.

5.5.3.2. Multiple response case

When there are multiple responses, it is often impossible to simultaneously optimize each one - trade-offs must be made

In the multiple response case, finding process operating conditions that simultaneously maximize (or minimize, as desired) all the responses is quite difficult, and often, impossible. Almost inevitably, the process engineer must make some trade-offs in order to find process operating conditions that are satisfactory for most (and hopefully all) the responses. In this sub-section, we examine some effective ways to make these trade-offs.

- Path of steepest ascent
- The desirability function approach
- The mathematical programming approach
  - Dual response systems
  - More than 2 responses

5.5.3.2.1. Multiple response: Path of steepest ascent

A weighted priority strategy is described using the path of steepest ascent for each response

When the responses exhibit adequate linear fit (i.e., the response models are all linear) the question is to find a direction or path that simultaneously considers the individual paths of maximum improvement and balances them in some way. This case is addressed next.

When there is a mix of linear and higher order responses, or when all empirical response models are of higher order, see sections 5.5.3.2.2 and 5.5.3.2.3. The desirability method (section 5.5.3.2.2) can also be used when all response models are linear.

Procedure: Path of Steepest Ascent, Multiple Responses.

1. Compute the gradients g_i (i = 1, 2, . . ., k) of all responses as explained in section 5.5.3.1.1. If one of the responses is clearly of primary interest compared to all the others, use only the gradient of this response and follow the procedure of section 5.5.3.1.1. Otherwise, continue with step 2.
2. Determine relative priorities π_i for each of the **k** responses. Then, the weighted gradient for the search direction is given by

$$g = \frac{\pi_1 g_1 + \pi_2 g_2 + \ldots + \pi_k g_k}{\sum_{i=1}^{k} \pi_i}$$

and the weighted direction is

$$d = \frac{g}{||g||}$$

The confidence cone for the direction of maximum improvement explained in section 5.6.1.1 can be used to weight down "poor" response models that provide very wide cones and unreliable directions. Since the width of the cone is proportional to $(1 - R^2)$, we can use

$$\pi_j = \frac{R_j^2}{\sum_{i=1}^{k} R_i^2} \qquad j = 1, 2, \ldots, k$$

Given a weighted direction of maximum improvement we can follow the single response steepest ascent procedure as in section 5.5.3.1.1. by selecting points with coordinates

$x_i^* = \rho d_i, i = 1, 2, \ldots, k$. These and related issues are explained more fully in Del Castillo (1996).

An example using the weighted priority method

Example: Path of Steepest Ascent, Multiple Response Case

Suppose the response model:

$$\hat{y}_1 = 711.0 + 50.9x_1 + 154.8x_2$$

with $R_1^2 = 0.8968$, represents the average yield of a production process obtained from a replicated factorial experiment in the two controllable factors (in coded units). From the same experiment, a second response model for the process standard deviation of the yield is obtained and given by

$$\hat{y}_2 = 19.26 + 6.31x_1 + 6.28x_2$$

with $R_2^2 = 0.5977$. We wish to maximize the mean yield while minimizing the standard deviation of the yield.

Step 1: compute the gradients:

$$g_1' = \left(\frac{50.9}{\sqrt{50.9^2 + 154.8^2}}, \frac{154.8}{\sqrt{50.9^2 + 154.8^2}} \right) = (0.3124, 0.9500)$$

$$g_2' = \left(\frac{-6.31}{\sqrt{6.31^2 + 6.28^2}}, \frac{-6.28}{\sqrt{6.31^2 + 6.28^2}} \right) = (-0.7088, -0.7054)$$

(recall we wish to *minimize* y_2).

Step 2: find relative priorities.

Since there are no clear priorities, we use the quality of fit as priority:

$$\pi_1 = \frac{0.8968}{0.8968 + 0.5977} = 0.6$$

$$\pi_2 = \frac{0.5977}{0.8968 + 0.5977} = 0.4$$

Then, the weighted gradient is

$$g' = (0.6(0.3124) + 0.4(-0.7088), 0.6(0.95) + 0.4(-0.7054)) = (-0.096, 0.2878)$$

which, after normalizing it (by dividing each coordinate by $\sqrt{0.096^2 + 0.2878^2}$) gives the weighted direction d' = **(-.03164, 0.9486).**

Therefore, if we want to move $\rho = 1$ coded units along the path of maximum improvement, we will set

$x_1 = (1)(-0.3164) = -0.3164$, $x_2 = (1)(0.9486) = 0.9486$ in the next run or experiment.

5.5.3.2.2. Multiple response: The desirability approach

The desirability approach is a popular method that assigns a "score" to a set of responses and chooses factor settings that maximize that score

The desirability function approach is one of the most widely used methods in industry for dealing with the optimization of multiple response processes. It is based on the idea that the "quality" of a product or process that has multiple quality characteristics, with one of them outside of some "desired" limits, is completely unacceptable. The method finds operating conditions $\mathbf{x}$ that provide the "most desirable" response values.

For each response $Y_i(\mathbf{x})$, a desirability function $d_i(Y_i)$ assigns numbers between 0 and 1 to the possible values of Y_i, with $d_i(Y_i) = 0$ representing a completely undesirable value of Y_i and $d_i(Y_i) = 1$ representing a completely desirable or ideal response value. The individual desirabilities are then combined using the geometric mean, which gives the *overall desirability* **D**:

$$D = (d_1(Y_1) \times d_2(Y_2) \times \ldots \times d_k(Y_k))^{1/k}$$

where **k** denotes the number of responses. Notice that if any response $\mathbf{Y_i}$ is completely undesirable $(d_i(Y_i) = 0)$ then the overall desirability is zero. In practice, fitted response

models $\hat{Y}_i$ are used in the method.

Depending on whether a particular response Y_i is to be maximized, minimized, or assigned to a target value, different desirability functions $d_i(Y_i)$ can be used. A useful class of desirability functions was proposed by Derringer and Suich (1980). Let L_i, U_i and T_i be the lower, upper, and target values desired for response $\mathbf{Y_i}$, where $L_i \leq T_i \leq U_i$. If a response is of the "target is best" kind, then its individual desirability function is

Desirability approach steps

$$d_i(\hat{Y}_i) = \begin{cases} 0 & \text{if } \hat{Y}_i(\mathbf{x}) < L_i \\ \left(\frac{\hat{Y}_i(\mathbf{x}) - L_i}{T_i - L_i}\right)^s & \text{if } L_i \leq \hat{Y}_i(\mathbf{x}) \leq T_i \\ \left(\frac{\hat{Y}_i(\mathbf{x}) - U_i}{T_i - U_i}\right)^t & \text{if } T_i \leq \hat{Y}_i(\mathbf{x}) \leq U_i \\ 0 & \text{if } \hat{Y}_i(\mathbf{x}) > U_i \end{cases}$$

where the exponents **s** and **t** determine how strictly the target value is desired. For **s = t =1**, the desirability function increases linearly towards T_i, for **s<1, t<1**, the function is convex, and for **s>1, t>1**, the function is concave (see the example below for an illustration).

If a response is to be maximized instead, the individual desirability is instead defined as

An example using the desirability approach

$$d_i(\hat{Y}_i) = \begin{cases} 0 & \text{if } \hat{Y}_i(\mathbf{x}) < L_i \\ \left(\frac{\hat{Y}_i(\mathbf{x}) - L_i}{T_i - L_i}\right)^s & \text{if } L_i \leq \hat{Y}_i(\mathbf{x}) \leq T_i \\ 1.0 & \text{if } \hat{Y}_i(\mathbf{x}) > T_i \end{cases}$$

where in this case T_i is interpreted as a large enough value for the response. Finally, if we want to minimize a response, we could use

$$d_i(\hat{Y}_i) = \begin{cases} 1.0 & \text{if } \hat{Y}_i(\mathbf{x}) < T_i \\ \left(\frac{\hat{Y}_i(\mathbf{x}) - U_i}{T_i - U_i}\right)^s & \text{if } T_i \leq \hat{Y}_i(\mathbf{x}) \leq U_i \\ 0 & \text{if } \hat{Y}_i(\mathbf{x}) > U_i \end{cases}$$

where T_i represents a small enough value for the response.

The desirability approach consists of the following steps:

1. Conduct experiments and fit response models for all **k** responses;
2. Define individual desirability functions for each response;
3. *Maximize* the overall desirability **D** with respect to the controllable factors.

Example:

Derringer and Suich (1980) present the following multiple response experiment arising in the development of a tire tread compound. The controllable factors are : x_1, hydrated silica level, x_2, silane coupling agent level, and x_3, sulfur level. The four responses to be optimized and their desired ranges are:

PICO Abrasion index, Y_1	$120 < Y_1$
200% modulus, Y_2	$1000 < Y_2$
Elongation at break, Y_3	$400 < Y_3 < 600$
Hardness, Y_4	$60 < Y_4 < 75$

The first two responses are to be maximized, and the value **s=1** was chosen for their desirability functions. The last two responses are "target is best" with $T_3 = 500$ and $T_4 = 67.5$. The values **s = t =1** were chosen in both cases. The following experiments were conducted according to a central composite design.

Run no.	x_1	x_2	x_3	Y_1	Y_2	Y_3	Y_4
1	-1.00	-1.00	-1.00	102	900	470	67.5
2	1.00	-1.00	-1.00	120	860	410	65
3	-1.00	1.00	-1.00	117	800	570	77.5
4	1.00	1.00	-1.00	198	2294	240	74.5
5	-1.00	-1.00	1.00	103	490	640	62.5
6	1.00	-1.00	1.00	132	1289	270	67
7	-1.00	1.00	1.00	132	1270	410	78
8	1.00	1.00	1.00	139	1090	380	70
9	-1.63	0.00	0.00	102	770	590	76
10	1.63	0.00	0.00	154	1690	260	70
11	0.00	-1.63	0.00	96	700	520	63
12	0.00	1.63	0.00	163	1540	380	75
13	0.00	0.00	-1.63	116	2184	520	65
14	0.00	0.00	1.63	153	1784	290	71
15	0.00	0.00	0.00	133	1300	380	70
16	0.00	0.00	0.00	133	1300	380	68.5
17	0.00	0.00	0.00	140	1145	430	68
18	0.00	0.00	0.00	142	1090	430	68
19	0.00	0.00	0.00	145	1260	390	69
20	0.00	0.00	0.00	142	1344	390	70

Using ordinary least squares and standard diagnostics, the fitted responses were:

$$\hat{Y}_1 = 139.12 + 16.49x_1 + 17.88x_2 + 2.21x_3 - 4.01x_1^2 - 3.45x_2^2 - 1.57x_3^2 + 5.12x_1x_2 - 7.88x_1x_3 - 7.13x_2x_3$$

(adj. $R^2 = 0.6903$);

$$\hat{Y}_2 = 1261.13 + 268.15x_1 + 246.5x_2 - 102.6x_3 - 83.57x_1^2 - 124.82x_2^2 + 199.2x_3^2 + 69.37x_1x_2 - 104.38x_1x_3 - 94.13x_2x_3$$

(adj. $R^2 = 0.4562$);

$$\hat{Y}_3 = 417.5 - 99.67x_1 - 31.4x_2 - 27.42x_3$$

(adj $R^2 = 0.6224$);

$$\hat{Y}_4 = 68.91 - 1.41x_1 + 4.32x_2 + 0.21x_3 + 1.56x_1^2 + 0.058x_2^2 - 0.32x_3^2 - 1.62x_1x_2 + 0.25x_1x_3 - 0.12x_2x_3$$

(adj. $R^2 = 0.7466$).

Note that no interactions were significant for response 3, and that the fit for response 2 is quite poor.

Optimization of **D** with respect to $\mathbf{x}$ was carried out using the Design Expert software. Figure 5.7 shows the individual desirability functions $d_i(\hat{Y}_i)$ for each of the four responses. The functions are linear since the values of **s** and **t** were selected equal to one. A dot indicates the best solution found by the Design Expert solver.

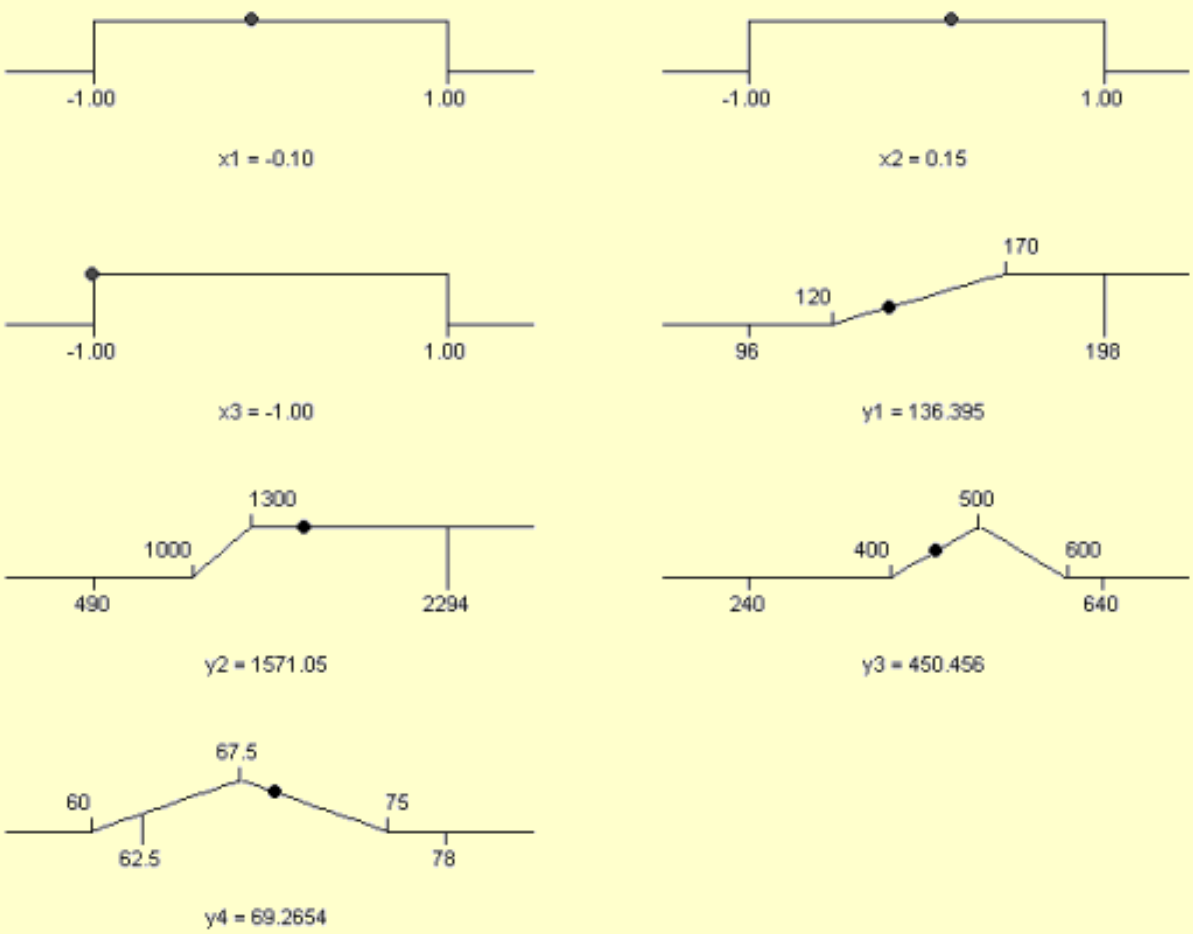


FIGURE 5.7 Desirability Functions and Optimal Solution for Example Problem

The best solution is $x^{*'} = (-0.10, 0.15, -1.0)$ and results in

$$d_1(\hat{Y}_1) = 0.34 \ \ (\hat{Y}_1(x^*) = 136.4), d_2(\hat{Y}_2) = 1.0 \ \ (\hat{Y}_2(x^*) = 1571.1),$$

$$d_3(\hat{Y}_3) = 0.49 \ \ (\hat{Y}_3(x^*) = 450.46)$$

and $d_4(\hat{Y}_4) = 0.76 \ \ (\hat{Y}_4(x^*) = 69.26)$. The overall desirability for this solution is 0.596. All responses are predicted to be within the desired limits.

Figure 5.8 shows a 3D plot of the overall desirability function $D(x)$ for the $x_2 - x_3$ plane when x_1 is fixed at -0.10. The function $D(x)$ is quite "flat" in the vicinity of the optimal solution, indicating that small variations around x^* are not predicted to change the overall desirability drastically. However, it should be emphasized the importance of performing confirmatory runs at the estimated optimal operating conditions. This is particularly true in this

example given the poor fit of the response models (e.g. $\hat{Y}_2$).

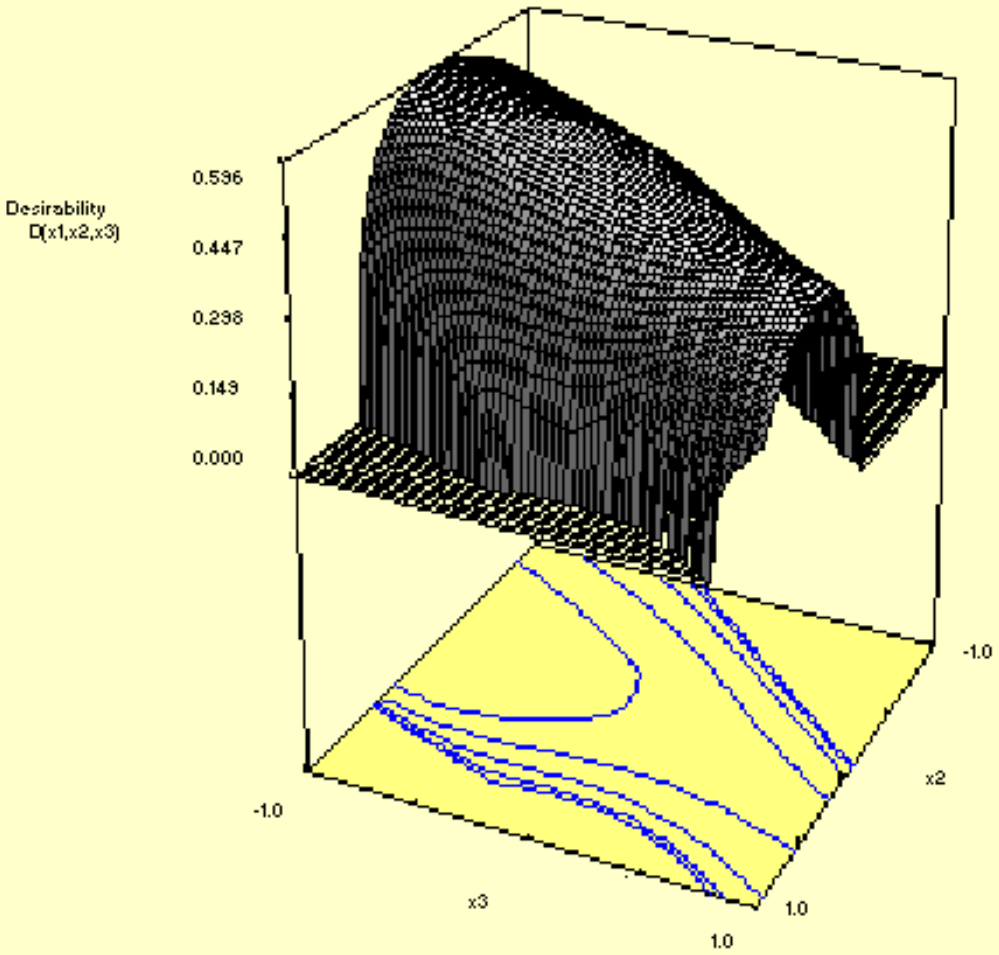


FIGURE 5.8 Overall Desirability Function for Example Problem

5.5.3.2.3. Multiple response: The mathematical programming approach

The mathematical programming approach maximizes or minimizes a primary response, subject to appropriate constraints on all other responses

The analysis of multiple response systems usually involves some type of optimization problem. When one response can be chosen as the "primary", or most important response, and bounds or targets can be defined on all other responses, a mathematical programming approach can be taken. If this is not possible, the desirability approach should be used instead.

In the mathematical programming approach the primary response is maximized or minimized, as desired, subject to appropriate constraints on all other responses. The case of two responses ("dual" responses) has been studied in more detail by some authors and is presented first. Then, the case of more than 2 responses is illustrated.

- Dual response systems
- More than 2 responses

Dual response systems

The optimization of dual response systems (DRS) consists of finding operating conditions $\mathbf{x}$ that

$$\begin{aligned} \text{optimize} \quad & \hat{Y}_p(\mathbf{x}) \\ \text{subject to:} \quad & \hat{Y}_s(\mathbf{x}) = T \\ & \mathbf{x}'\mathbf{x} \leq \rho^2 \end{aligned}$$

where **T** is the target value for the secondary response and ρ is the radius of a spherical constraint that limits the region in the controllable factor space where the search should be undertaken. The value of ρ should be chosen with the purpose of avoiding solutions that extrapolate too far outside of the region where the experimental data were obtained. For example, if the experimental design is a central composite design, choosing $\rho = \alpha$ (axial distance) is a logical choice. Bounds of the form $L \leq x_i \leq U$ can be used instead if a cubical experimental region was used (e.g., when using a factorial experiment). Note that a Ridge Analysis problem is nothing but a DRS problem where the secondary constraint is absent. Thus, any algorithm or solver for DRS's will also work for the Ridge Analysis of single response systems.

In a DRS, the response models $\hat{Y}_p$ and $\hat{Y}_s$ can be linear, quadratic or even cubic polynomials. A nonlinear programming algorithm has to be used for the optimization of a DRS. For the particular case of quadratic responses, an equality constraint for the secondary response, and a spherical region of experimentation, specialized optimization algorithms exist that guarantee global optimal solutions. In such case, the algorithm DRSALG can be used (download from

http://www.nist.gov/cgi-bin/exit_nist.cgi?url=http://www.stat.cmu.edu/jqt/29-3) but a Fortran compiler is necessary.

In the more general case of inequality constraints or a cubical region of experimentation, a general purpose nonlinear solver must be used and several starting points should be tried to avoid local optima. This is illustrated in the next section.

More than 2 responses

Example:

3 components (x_1, x_2, x_3) of a propellant need to be selected to maximize a primary response, burning rate (Y_1) subject to satisfactory levels of two secondary reponses, namely, the variance of the burning rate (Y_2) and the cost (Y_3). The three components must add up to 100% of the mixture. The fitted models were:

$$\hat{Y}_1 = 35.4x_1 + 42.77x_2 + 70.36x_3 + 16.02x_1x_2 + 36.33x_1x_3 + 136.8x_2x_3 + 854.9x_1x_2x_3$$

$$\hat{Y}_2 = 3.88x_1 + 9.03x_2 + 13.63x_3 - 0.1904x_1x_2 - 16.61x_1x_3 - 27.67x_2x_3$$

$$\hat{Y}_3 = 23.13x_1 + 19.73x_2 + 14.73x_3.$$

The optimization problem is therefore:

$$\begin{aligned} \text{maximize} \quad & \hat{Y}_1(\mathbf{x}) \\ \text{subject to} \quad & \hat{Y}_2(\mathbf{x}) \leq 4.5 \\ & \hat{Y}_3(\mathbf{x}) \leq 20 \\ & x_1 + x_2 + x_3 = 1.0 \\ & 0 \leq x_1 \leq 1 \\ & 0 \leq x_2 \leq 1 \\ & 0 \leq x_3 \leq 1 \end{aligned}$$

We can use Microsoft Excel's "solver" to solve this problem. The table below shows an Excel spreadsheet that has been setup with the problem above. Cells B2:B4 contain the decision variables (cells to be changed), cell E2 is to be maximized, and all the constraints need to be entered appropriately. The figure shows the spreadsheet after the solver completes the optimization. The solution is $\mathbf{x}^{*'} = (0.212, 0.343, 0.443)$ which provides $\hat{Y}_1 = 106.62$, $\hat{Y}_2 = 4.17$, and $\hat{Y}_3 = 18.23$. Therefore, both secondary responses meet the desired bounds. The solver should be run from a variety of starting points (i.e., try different initial values in cells B1:B3 prior to start the solver) to avoid local optima. Once again, confirmatory experiments should be conducted at the estimated optimal operating conditions.

	A	B	C	D	E
1	Factors			Responses	
2	x1	0.21233		Y1(x)	106.6217
3	x2	0.343725		Y2(x)	4.176743
4	x3	0.443946		Y3(x)	18.23221
5	Additional	constraint			
6	x1+x2+x3	1.000001			

5.5.4. What is a mixture design?

When the factors are proportions of a blend, you need to use a mixture design

In a mixture experiment, the independent factors are proportions of different components of a blend. For example, if you want to optimize the tensile strength of stainless steel, then the factors of interest might be the proportions of iron, copper, nickel, and chromium in the alloy. The fact that the proportions of the different factors must sum to 100% complicate the design as well as the analysis of mixture experiments.

When the mixture components are subject to the constraint that they must sum to one, there are standard mixture designs for fitting standard models, such as *Simplex-Lattice* designs and *Simple-Centroid* designs. When mixture components are subject to additional constraints, such as a maximum and/or minimum value for each component, designs other than the standard mixture designs, referred to as constrained mixture designs or *Extreme-Vertices* designs, are appropriate.

In mixture experiments, the measured response is assumed to depend only on the relative proportions of the ingredients or components in the mixture and not on the amount of the mixture. The amount of the mixture could also be studied as an additional factor in the experiment, however, this would be an example of where mixture and process variables are treated together.

The main distinction between mixture experiments and independent variable experiments is that with the former, the input variables or components are non negative proportionate amounts of the mixture, and if expressed as fractions of the mixture, they must sum to one. If the sum of the component proportions are less than one, then the variable proportions can be rewritten as scaled fractions so that the scaled fractions sum to one.

Purpose of a mixture design

In mixture problems, the purpose of the experiment is to model the blending surface with some form of mathematical equation so that:

1. Predictions of the response for any mixture or combination of the ingredients can be made empirically, or
2. Some measure of the influence on the response of each component singly and in combination with other components can be obtained.

Assumptions for mixture experiments

The usual assumptions made for factorial experiments are also assumed for mixture experiments. In particular, it is assumed that the errors are assumed to be independent and identically distributed with zero mean and common variance. Another assumption that is made, similar to that made for factorial designs, is that the true underlying response surface is continuous over the region being studied.

Steps in planning a mixture experiment

Planning a mixture experiment typically involves the following steps (Cornell, Piepel, 1994):

1. Define the objectives of the experiment
2. Select the mixture components and any other factors to be studied. Other factors may include process variables or the total amount of the mixture.
3. Identify any constraints on the mixture components or other factors in order to specify the experimental region.
4. Identify the response variables to be measured.
5. Propose an appropriate model form for modeling the response data as functions of the mixture components and other factors selected for the experiment.
6. Select an experimental design that is sufficient not only to fit the proposed model form but allows a test of model adequacy as well.

5.5.4.1. Mixture screening designs

Screening experiments can be used to identify the important mixture factors

In some areas of mixture experiments, for example, certain chemical industries, there is often present a large number, *q*, of potentially important components that can be considered candidates in an experiment. The objectives of these types of experiments is to screen these components to identify the ones that are most important. In this type of situation, the experimenter should consider a screening experiment to reduce the number of possible components.

The construction of screening designs and their corresponding models often begins with the first order or first degree mixture model

A first order mixture model

$$E(Y) = \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_q x_q$$

where the beta coefficients are non negative and sum to one.

Choices of types of screening designs depend on constraints

If the experimental region is a simplex, it is generally a good idea to make the ranges of the components as similar as possible. Then the relative effects of the components can be assessed by ranking the ratios of the parameter estimates *i*, where *i = 1,...,q*, relative to their standard errors. Designs, called simplex screening designs are recommended for those cases where it is possible to experiment over the total simplex region. Constrained mixture designs are suggested when the proportions of some or all of the components are restricted by upper and lower bounds. If these designs are not feasible in this situation, then D-optimal designs for a linear model are always an option.

5.5.4.2. Simplex-lattice designs

Definition of simplex-lattice points

A {*q*, *m*} simplex lattice design for *q* components consists of points defined by the following coordinate settings: the proportions assumed by each component take the *m*+1 equally spaced values from 0 to 1,

$x_i = 0, 1/m, 2/m, \ldots, 1$ for $i = 1, 2, \ldots, q$

and all possible combinations (mixtures) of the proportions from this equation are used.

Except for the center, all design points are on the simplex boundaries

Note that the standard Simplex-Lattice and the Simplex-Centroid designs (described later) are boundary point designs; that is, with the exception of the overall centroid, all the design points are on the boundaries of the simplex. When one is interested in prediction in the interior, it is highly desirable to augment the simplex type designs with interior design points.

Consider a three component mixture where the number of equally spaced levels for each component is four (i.e. $x_i = 0, 0.333, 0.667, 1$). In this example then $q = 3$ and $m = 3$. If one considers all possible blends of the three components with these proportions, then the {3, 3} simplex-lattice contains the 10 blending coordinates listed in the table below. The experimental region and the distribution of design runs over the simplex region is shown in the figure below. There are a total of 10 design runs for the {3, 3} simplex lattice design.

Example of a 3-component simplex lattice design

TABLE 5.3 Simplex Lattice Design

X1	X2	X3	Y
0	0	1	?
0	.333	.667	?
0	.667	.333	?
0	1	0	?
.333	0	.667	?
.333	.333	.333	?
.333	.667	0	?
.667	0	.333	?
.667	.333	0	?
1	0	0	?

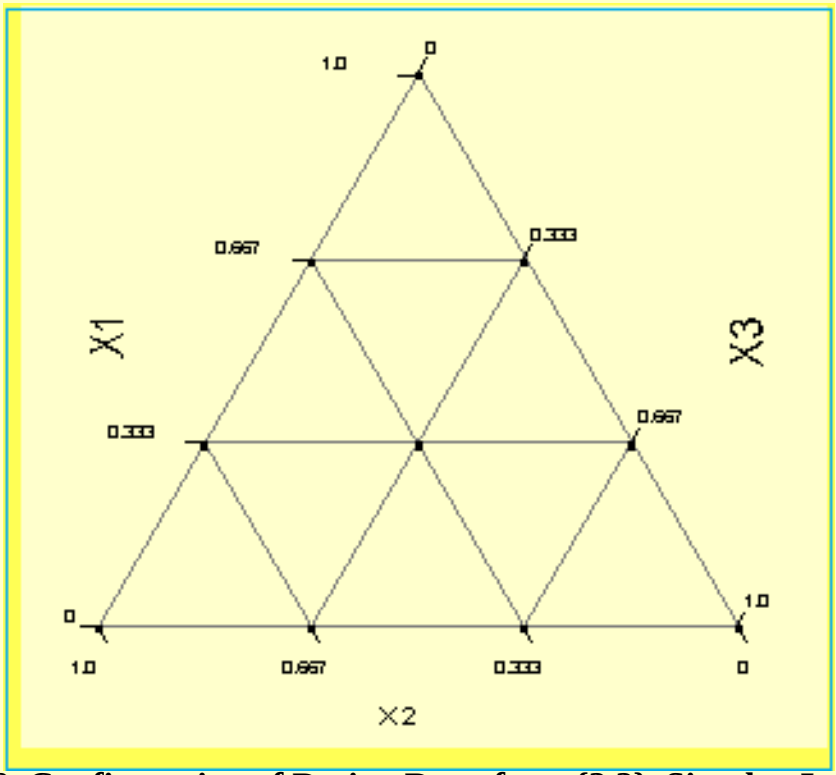


FIGURE 5.9 Configuration of Design Runs for a {3,3} Simplex-Lattice Design

The number of design points in the simplex-lattice is $(q+m-1)!/m!(q-1)!$.

Definition of canonical polynomial model used in mixture experiments

Now consider the form of the polynomial model that one might fit to the data from a mixture experiment. Due to the restriction $x_1 + x_2 + ... + x_q = 1$, the form of the regression function that is fit to the data from mixture experiments is somewhat different than the traditional polynomial fit and is often referred to as the canonical polynomial. Its form is derived using the general form of the regression function that can be fit to data collected at the points of a {*q*, *m*} simplex-lattice design and substituting into this the dependence relationship among the x_i terms. The number of terms in the {*q*, *m*} polynomial is (*q*+*m*-1)!/*m*!(*q*-1)!. This number is equal to the number of points that make up the associated {*q*, *m*} simplex-lattice design.

For example, the equation that can be fit to the points from a {*q*, *m*=1} simplex-lattice design is

$$E(Y) = \beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q$$

Multiplying the β_0 by $\left(x_1 + x_2 + \cdots x_q = 1\right)$, the resulting equation is

$$E(Y) = \beta_1^{\bullet} x_1 + \cdots + \beta_q^{\bullet} x_q$$

where $\beta_i^{\bullet} = \beta_0 + \beta_i$ for all $i = 1, \cdots, q$

This is called the canonical form of the first order mixture model. In general, the canonical forms of the mixture models (with the asterisks removed from the parameters) are as follows:

Summary of canonical mixture models

Linear $$E(Y) = \sum_{i=1}^{q} \beta_i x_i$$

Quadratic $$E(Y) = \sum_{i=1}^{q} \beta_i x_i + \sum \sum_{i<j}^{q} \beta_{ij} x_i x_j$$

Cubic $$E(Y) = \sum_{i=1}^{q} \beta_i x_i + \sum \sum_{i<j}^{q} \beta_{ij} x_i x_j + \sum \sum_{i<j}^{q} \delta_{ij} x_i x_j (x_i - x_j) + \sum \sum \sum_{i<j<k} \beta_{ijk} x_i x_j x_k$$

Special Cubic $$E(Y) = \sum_{i=1}^{q} \beta_i x_i + \sum \sum_{i<j}^{q} \beta_{ij} x_i x_j + \sum \sum \sum_{i<j<k} \beta_{ijk} x_i x_j x_k$$

The terms in the canonical mixture polynomials have simple interpretations. Geometrically, the parameter β_i in the above equations represent the expected response to the pure mixture $x_i=1$, $x_j=0$, ij, and is the height of the mixture surface at the vertex $x_i=1$. The portion of each of the above polynomials given by

$$\sum_{i=1}^{q} \beta_i x_i$$

is called the linear blending portion. When blending is strictly additive, then the linear model form above is an appropriate model.

Three component mixture example

The following example is from Cornell (1990) and consists of a three component mixture problem. The three components are Polyethylene (X1), polystyrene (X2), and polypropylene (X3) which are blended together to form fiber that will be spun into yarn. The product developers are only interested in the pure and binary blends of these three materials. The response variable of interest is yarn elongation in kilograms of force applied. A {3,2} simplex-lattice design is used to study the blending process. The simplex region and the six design runs are shown in the figure below. The figure was generated in JMP version 3.2. The design and the observed responses are listed in the table below. There were two replicate observations run at each of the pure blends. There were three replicate observations run at the binary blends. There are a total of 15 observations with six unique design runs.

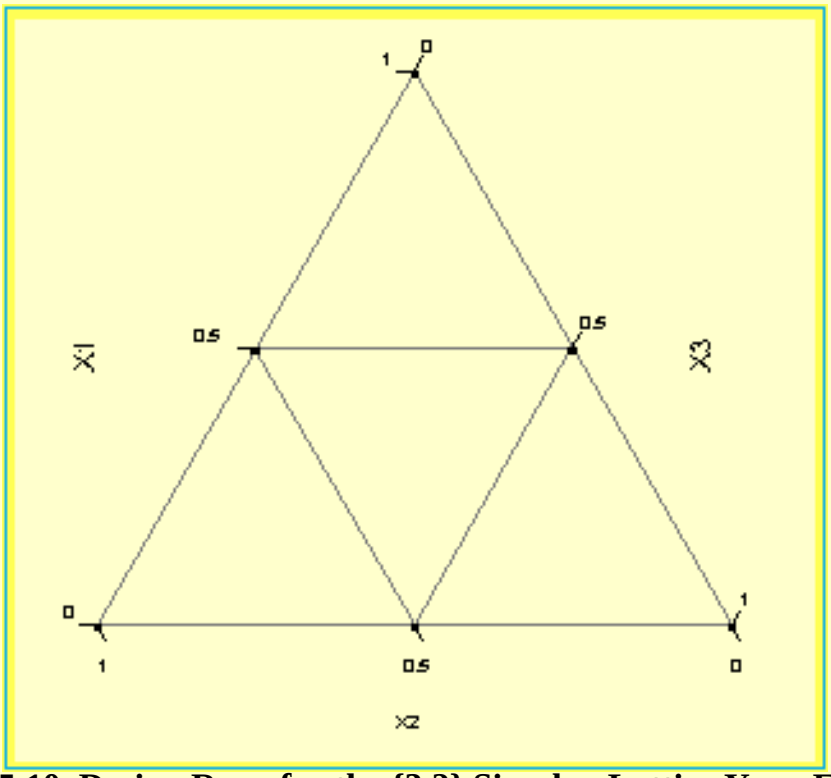


FIGURE 5.10 Design Runs for the {3,2} Simplex-Lattice Yarn Elongation Problem

---

TABLE 5.4 Simplex-Lattice Design for Yarn Elongation Problem

X1	X2	X3	Observed Elongation Values
0	0	1	16.8, 16.0
0	.5	.5	10.0, 9.7, 11.8
0	1	0	8.8, 10.0
.5	0	.5	17.7, 16.4, 16.6
.5	.5	0	15.0, 14.8, 16.1
1	0	0	11.0, 12.4

The design runs listed in the above table are in standard order. The actual order of the 15 treatment runs was completely randomized. JMP 3.2 will be used to analyze the results. Since there are three levels of each of the three mixture components, a quadratic mixture model can be fit to the data. The output from the model fit is shown below. Note that there was no intercept fit in the model. To analyze the data in JMP, create a new table with one column corresponding to the observed elongation values. Select Fit Model and create the quadratic mixture model (this will look like the 'traditional' interactions regression model obtained from standard classical designs). Check the No Intercept box on the Fit Model screen. Click on Run Model. The output is shown below.

JMP analysis for the mixture model example

JMP Output for {3,2} Simplex-Lattice Design

Screening Fit

Summary of Fit

RSquare	0.951356
RSquare Adj	0.924331
Root Mean Square Error	0.85375
Mean of Response	13.54
Observations (or Sum Wgts)	15

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	5	128.29600	25.6592	35.2032
Error	9	6.56000	0.7289	
C Total	14	134.85600		

Prob > F < .0001

Tested against reduced model: Y=mean

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
X1	11.7	0.603692	19.38	<.0001
X2	9.4	0.603692	15.57	<.0001
X3	16.4	0.603692	27.17	<.0001
X2*X1	19	2.608249	7.28	<.0001
X3*X1	11.4	2.608249	4.37	0.0018
X3*X2	-9.6	2.608249	-3.68	0.0051

Under the parameter estimates section of the output are the individual t-tests for each of the parameters in the model. The three cross product terms are significant (X1*X2, X3*X1, X3*X2), indicating a significant quadratic fit.

The fitted quadratic mixture model is

$$\hat{y} = 11.7x_1 + 9.4x_2 + 16.4x_3 + 19.0x_1x_2 + 11.4x_1x_3 - 9.6x_2x_3$$

Since $b_3 > b_1 > b_2$, one can conclude that component 3 (polypropylene) produces yarn with the highest elongation. Additionally, since b12 and b13 are positive, blending components 1 and 2 or components 1 and 3 produce higher elongation values than would be expected just by averaging the elongations of the pure blends. This is an example of 'synergistic' blending effects. Components 2 and 3 have antagonistic blending effects because b23 is negative.

The figure below is the contour plot of the elongation values From the plot it can be seen that if maximum elongation is desired, a blend of components 1 and 3 should be chosen consisting of about 75% - 80% component 3 and 20% - 25% component 1.

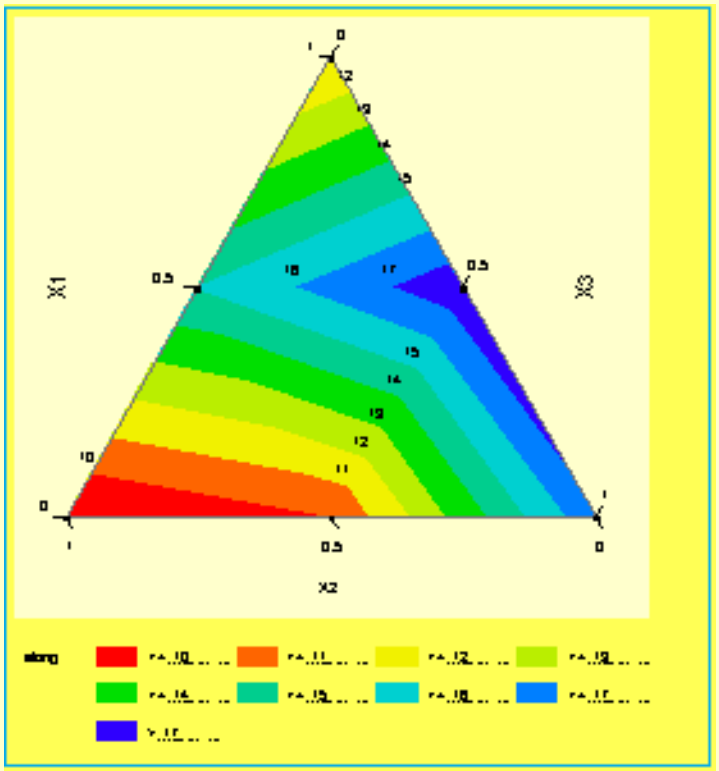


FIGURE 5.11 Contour Plot of Predicted Elongation Values from {3,2} Simplex-Lattice Design

5.5.4.3. Simplex-centroid designs

Definition of simplex-centroid designs

A second type of mixture design is the simplex-centroid design. In the q-component simplex-centroid design, the number of distinct points is 2^q-1. These points correspond to q permutations of (1, 0, 0,…,0) or q single component blends, the (q choose 2) permutations of (.5, .5, 0, …,0) or all binary mixtures, the (q choose 3) permutations of (1/3, 1/3, 1/3, 0,…,0), …, and so on, with finally the overall centroid point (1/q, 1/q, …,1/q) or q-nary mixture.

The design points in the Simplex-Centroid design will support the polynomial

Model supported by simplex-centroid designs

$$E(Y) = \sum_{i=1}^{q} \beta_i x_i + \sum \sum_{i<j=1}^{q} \beta_{ij} x_i x_j + \sum \sum \sum_{i<j<k=1}^{q} \beta_{ijk} x_i x_j x_k + \cdots + \beta_{12\ldots q} x_i x_j \cdots x_q$$

which is the q-th-order mixture polynomial. For q = 2, this is the quadratic model. For q = 3, this is the special cubic model.

For example, the fifteen runs for a four component (q = 4) simplex-centroid design are: (1,0,0,0), (0,1,0, 0), (0,0,1,0), (0,0,0,1), (.5,.5, 0,0), (.5,0,.5, 0) ..., (0,0,.5,.5), (1/3,1/3,1/3,0), ...,(0,1/3,1/3,1/3), (1/4,1/4,1/4,1/4). The runs for a three component simplex-centroid design of degree 2 are (1,0,0), (0,1,0), (0,0,1), (.5, .5, 0), (.5, 0, .5), (0, .5, .5), (1/3, 1/3, 1/3). However, in order to fit a first order model, only the five terms with a "1" or all "1/4's" would be needed. To fit a second order model, add the six terms with a ".5" (this also fits a saturated third order model, with no degrees of freedom left for error).

5.5.4.4. Constrained mixture designs

Upper and/or lower bound constraints may be present

In mixture designs where there are constraints on the component proportions, these are often upper and/or lower bound constraints of the form L_i, x_i, U_i, i = 1, 2,..., q, where L_i is the lower bound for the i-th component and U_i is the upper bound for the i-th component. The general form of the constrained mixture problem is

Typical additional constraints

$$x_1 + x_2 + \cdots x_q = 1$$

$$L_i \le x_i \le U_i, \quad \text{for } i = 1, 2, \cdots, q$$

$$\text{where } L_i \ge 0 \text{ and } U_i \le 1$$

.

Consider the following case where only the lower bounds in the above equation are imposed, so that the constrained mixture problem becomes

Example

$$x_1 + x_2 + \cdots x_q = 1$$

$$L_i \le x_i \le 1, \quad \text{for } i = 1, 2, \cdots, q$$

Assume we have a three component mixture problem with constraints

$$0.3 \le x_1 \qquad 0.4 \le x_2 \qquad 0.1 \le x_3$$

The feasible mixture space is shown in the figure below. Note that the existence of lower bounds does not affect the shape of the mixture region, it is still a simplex region. In general, this will always be the case if only lower bounds are imposed on any of the component proportions.

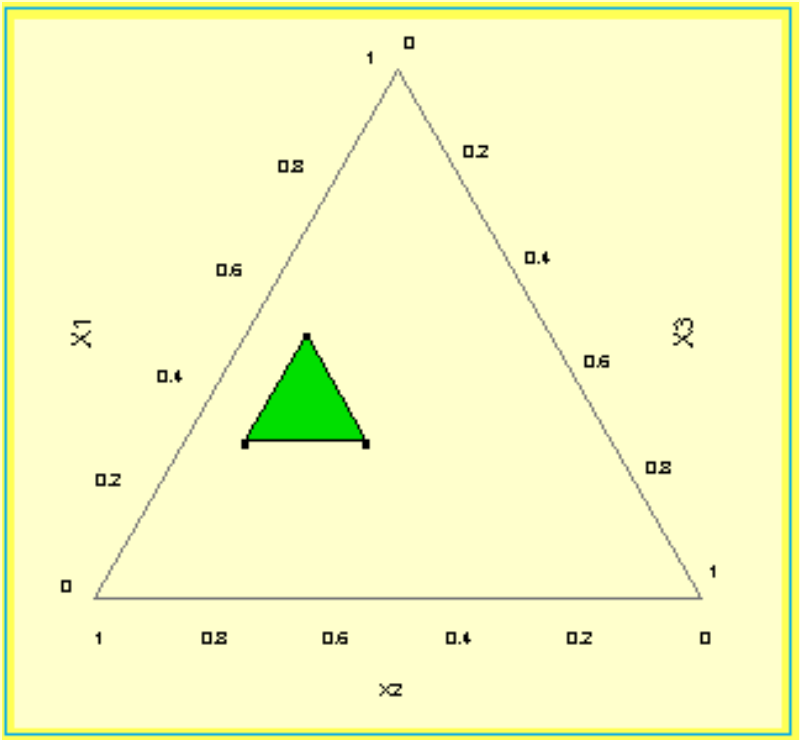


FIGURE 5.12 The Feasible Mixture Space (Shaded Region) for Three Components with Lower Bounds

A simple transformation helps in design construction and analysis

Since the new region of the experiment is still a simplex, it is possible to define a new set of components that take on the values from 0 to 1 over the feasible region. This will make the design construction and the model fitting easier over the constrained region of interest. These new components (x^*_i) are called pseudo components and are defined using the following formula

$$x_i^* = \frac{x_i - L_i}{1 - L}$$

where $L = \sum_{i=1}^{q} L_i < 1$ is the sum of all the lower bounds.

In the three component example above, the pseudo components are

$$x_1^* = \frac{x_1 - 0.3}{0.2} \qquad x_2^* = \frac{x_2 - 0.4}{0.2} \qquad x_3^* = \frac{x_3 - 0.1}{0.2}$$

Constructing a design in the pseudo components is done by specifying the design points in terms of the x^*_i and then converting them to the original component settings using

$$x_i = L_i + (1 - L)\, x^*_i$$

In terms of the pseudo components, the experimenter has the choice of selecting a Simplex-Lattice or a Simplex-Centroid design, depending on the objectives of the experiment.

Simplex-centroid design example (after transformation)

Suppose, we decided to use a Simplex-centroid design for the three component experiment. The table below shows the design points in the pseudo components, along with the corresponding setting for the original components.

TABLE 5.5 Pseudo Component Settings and Original Component Settings, Three Component Simplex-Centroid Design

Pseudocomponents			Original Components		
X_1	X_2	X_3	X_1^*	X_2^*	X_3^*
1	0	0	.5	.4	.1
0	1	0	.3	.6	.1
0	0	1	.3	.4	.3
.5	.5	0	.4	.5	.1
.5	0	.5	.4	.4	.2
0	.5	.5	.3	.5	.2
.3333	.333	.3333	.3667	.4667	.1666

Use of pseudo components (after transformation) is recommended

It is recommended that the pseudo components be used to fit the mixture model. This is due to the fact that the constrained design space will usually have relatively high levels of multicollinearity among the predictors. Once the final predictive model for the pseudo components has been determined, then the equation in terms of the original components can be determined by substituting in the relationship between x_i and x^*_i.

D-optimal designs can also be used

Computer aided designs (D-optimal for example) can be used to select points for a mixture design in a constrained region. See Myers and Montgomery (1995) for more details on using D-optimal designs in mixture experiments.

Extreme vertice desgns anre another option

Note: There are other mixture designs that cover only a sub portion or smaller space within the simplex. These types of mixture designs (not covered here) are referred to as *extreme vertices designs*. (see chapters 4, 3 of Myers and Montgomery (1995), respectively).

5.5.4.5. Treating mixture and process variables together

Options for setting up experiments for processes that have both standard process variables and mixture variables

Consider a mixture experiment consisting of *q* mixture components and *k* process variables. First consider the case where each of the process variables to be studied has only two levels. Orthogonally scaled factor settings for the process variables will be used (i.e. -1 is the low level, 1 is the high level, 0 is the center point). Also assume that each of the components xi can range from 0 to 1. The region of interest then for the process variables is an *k*-dimensional hyper cube.

The region of interest for the mixture components is the (*q*-1)-dimensional simplex. The combined region of interest for both the process variables and the mixture components is of dimensionality *q* - 1 + *k*.

For example, consider three mixture components (x_1, x_2, x_3) with three process variables (z_1, z_2, z_3). The dimensionality of the region is 5. The combined region of interest for three mixture components and three process variables is shown in the two figures below. The complete space of the design can be viewed in either of two ways. The first diagram shows the idea of a full factorial at each vertex of the three component simplex region. The second diagram show the idea of three component simplex region at each point in the full factorial. In either case, the same overall process space is being investigated.

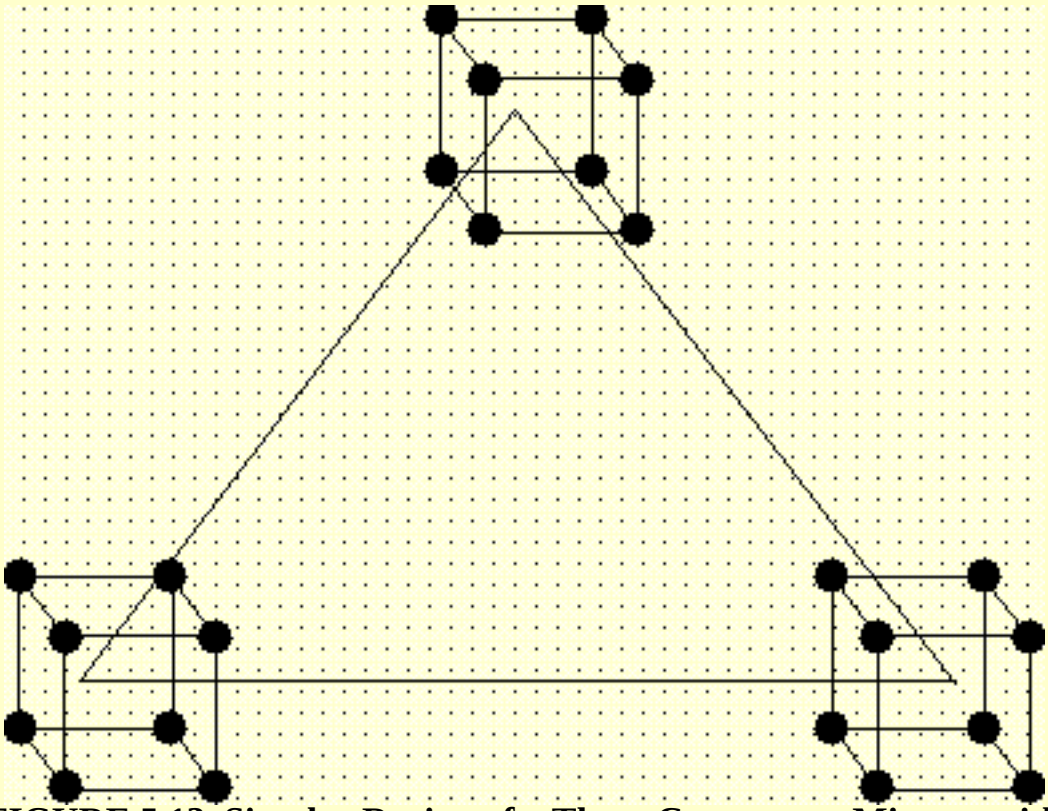


FIGURE 5.13 Simplex Region of a Three Component Mixture with a 2^3 Full Factorial at Each Pure Mixture Run

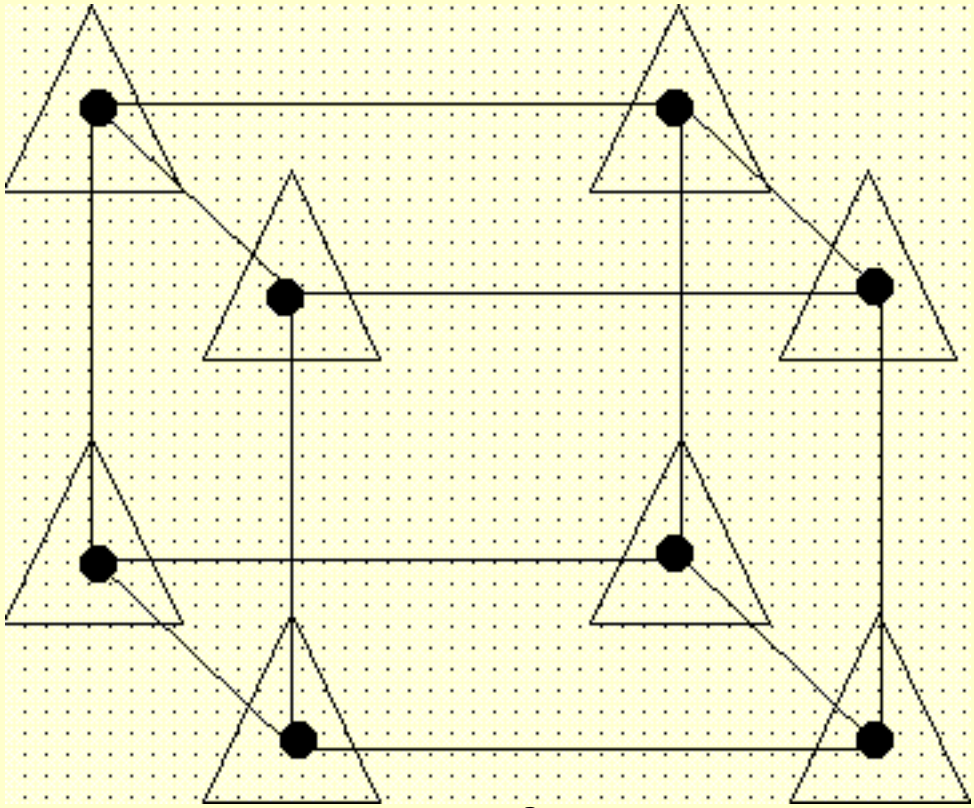


FIGURE 5.14 Process Space of a 2^3 Full Factorial with the Three Component Simplex Region at Each Point of the Full Factorial

As can be seen from the above diagrams, setting up the design configurations in the process variables and mixture components involves setting up a mixture design at each point of a configuration in the process variables, or similarly, creating a factorial arrangement in the process variables at each point of composition in the mixture components. For the example depicted in the above two diagrams, this is not the only design available for this number of mixture components with the specified number of process variables. Another option might be to run a fractional factorial design at each vertex or point of the mixture design, where the same fraction is run at each mixture design point. Still another option might be to run a fractional factorial design at each vertex or point of the mixture design, where a different fraction is run at each mixture design point.

5.5.5. How can I account for nested variation (restricted randomization)?

Nested data structures are common and lead to many sources of variability

Many processes have more than one source of variation in them. In order to reduce variation in processes these multiple sources must be understood, and that often leads to the concept of nested or hierarchical data structures. For example, in the semiconductor industry, a batch process may operate on several wafers at a time (wafers are said to be *nested* within batch). Understanding the input variables that control variation among those wafers, as well as understanding the variation across each wafer in a run, is an important part of the strategy for minimizing the total variation in the system.

Example of nested data

Figure 5.15 below represents a batch process uses 7 monitor wafers in each run. The plan further calls for measuring some response on each wafer at each of 9 sites. The organization of the sampling plan has a hierarchical or nested structure: the batch run is the topmost level, the second level is an individual wafer, the third level is the site on the wafer.

The total amount of data generated per batch run will be 7*9 = 63 data points. One approach to analyzing this data would be to compute the mean of all these points as well as their standard deviation and use those results as responses for each run.

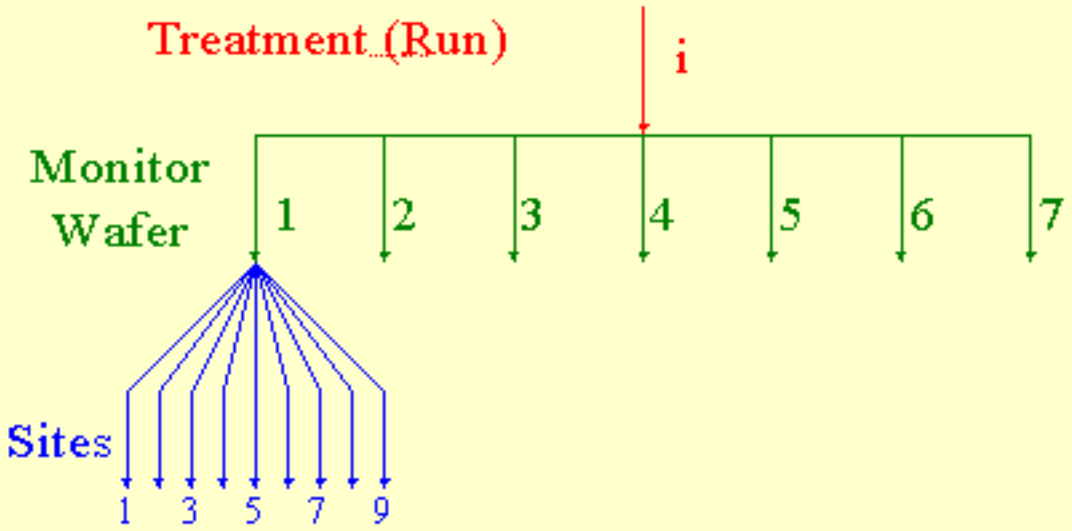


FIGURE 5.15 Hierarchical Data Structure Example

Analyzing the data as suggested above is not absolutely incorrect, but doing so loses information that one might otherwise obtain. For example, site 1 on wafer 1 is physically different from site 1 on wafer 2 or on any other wafer. The same is true for any of the sites on any of the wafers. Similarly, wafer 1 in run 1 is physically different from wafer 1 in run 2, and so on. To describe this situation one says that *sites are nested within wafers* while *wafers are nested within runs*.

As consequence of this nesting situation, there are restrictions on the randomization that can occur in the experiment. This kind of restricted randomization always produces nested sources of variation.. Examples of nested variation or restricted randomization discussed on this page are *split-plot* and *strip-plot* designs.

Wafer to wafer and site to site variations are often "noise factors" in an experiment

The objective of an experiment with the type of sampling plan described in Figure 5.15 generally involves reducing the variability due to sites on the wafers and wafers within runs (or batches) in the process. The sites on the wafers and the wafers within a batch become sources of unwanted variation and an investigator seeks to make the system robust to those sources -- in other words, one could treat wafers and sites as noise factors in such an experiment.

Treating wafers and sites as random effects

Because the wafers and the sites represent an unwanted source of variation and because one of the objectives is to reduce the process sensitivity to that source of variation, treating wafers and sites as random effects in the analysis of the data is a reasonable approach. In other words, nested variation is often another way of saying nested random effects or nested sources of noise. If the factors wafers and sites, are treated as random effects, then it is possible to estimate a variance

allows calculation of variance estimates

component due to that particular source of variation through analysis of variance techniques. Once estimates of the variance components have been obtained, then an investigator is able to determine the largest source of variation in the process under experimentation and the magnitudes of the other sources of variation in relation to the largest source.

Split-plot designs often arise when some factors are "hard to vary" or when batch processes are run

If an experiment or process has nested variation then the experiment or process has multiple sources of random error or random effects that affect its output. Having nested random effects in a model is the same thing as having nested variation in a model.

Split-Plot Designs

Split-plot designs result when a particular type of restricted randomization has occurred during the experiment. A simple factorial experiment can result in a split-plot type of design because of the way the experiment was actually executed.

A split-plot experiment example

In many industrial experiments three situations often occur:

1. some of the factors of interest may be 'hard to vary' while the remaining factors are easy to vary. As a result, the order in which the treatment combinations for the experiment are run are 'ordered' by these 'hard to vary' factors
2. experimental units are processed together as a batch for one or more of the factors in a particular treatment combination
3. experimental units are processed individually one right after the other for the same treatment combination without resetting the factor settings for that treatment combination.

An experiment run under one of the above three situations usually results in a split-plot type of design. Consider an experiment to examine electroplating of aluminum (non-aqueous) on copper strips. The three

factors of interest are: current (A); solution temperature (T); and the solution concentration of the plating agent (S). Plating rate is the measured response. There are a total of 16 copper strips available for the experiment. The treatment combinations to be run (orthogonally scaled) are listed below in standard order (i.e. they have not been randomized):

TABLE 5.6 Orthogonally Scaled Treatment Combinations from a 23 Full Factorial**

Current	Temperature	Concentration
-1	-1	-1
-1	-1	1
-1	1	-1
-1	1	1
1	-1	-1
1	-1	1
1	1	-1
1	1	1

Consider running the experiment under the first condition listed above, where the factor solution concentration of the plating agent (S) is hard to vary. Since this factor is hard to vary the experimenter would like to randomize the treatment combinations so that the solution concentration factor has a minimal number of changes. In other words, the randomization of the treatment runs is restricted somewhat by the level of the solution concentration factor.

As a result, the treatment combinations might be randomized such that those treatment runs corresponding to one level of the concentration (-1) are run first. Each copper strip is individually plated, meaning only one strip at a time gets placed in the solution for a given treatment combination. Once the four runs at the low level of solution concentration have been completed, the solution is changed to the high level of concentration (1), and the remaining four runs of the experiment are performed (where again, each strip is individually plated).

Definition of experimental units and whole plot and subplot factors for

Once one complete replicate of the experiment has been completed, a second replicate is performed in exactly the same way (with a set of four copper strips processed for a given level of solution concentration before changing the concentration and processing the remaining four

this experiment

strips).

Running the experiment in this way results in a split plot design. Solution concentration is known as the *whole plot factor* and the *subplot factors* are the current and the solution temperature.

Each size of experimental unit leads to an error term in the model for the experiment

A split plot design has more than one size experimental unit. In this experiment, one size experimental unit is an individual copper strip. The treatments or factors that were applied to the individual strips are solution temperature and current (these factors were changed each time a new strip was placed in the solution). The other or larger size experimental unit is a set of four copper strips. The treatment or factor that was applied to a set of four strips is solution concentration (this factor was changed after four strips were processed). The smaller size experimental unit is referred to as the subplot experimental unit, while the larger experimental unit is referred to as the whole plot unit..

There are a total of 16 subplot experimental units for this experiment. Solution temperature and current are the subplot factors in this experiment. There are a total of four whole plot experimental units in this experiment. Solution concentration is the whole plot factor in this experiment. Since there are two sizes of experimental units in the experiment, there are two error terms in the model, one that corresponds to the whole plot error or whole plot experimental unit and one that corresponds to the subplot error or subplot experimental unit.

The ANOVA table for this experiment would look as follows:

Source	DF
Replication	1
Concentration	1
Error (Whole plot) = Rep*Conc	1
Temperature	1
Rep*Temp	1
Current	1
Rep*Current	1
Temp*Conc	1
Rep*Temp*Conc	1
Temp*Current	1
Rep*Temp*Current	1
Current*Conc	1
Rep*Current*Conc	1
Temp*Current*Conc	1
Error (Subplot) =Rep*Temp*Current*Conc	1

A batch process leads to a

The first three sources are from the whole plot level, while the next 12

different experiment - also a strip-plot

are from the subplot portion. A normal plot of the 12 subplot term estimates could be used to look for significant terms.

Consider running the experiment under the second condition listed above (i.e. a batch process) where four copper strips are placed in the solution at one time. A specified level of current can be applied to an individual strip within the solution. The same 16 treatment combinations (a replicated 2**3 factorial) are run as were run under the first scenario, however, the way in which the experiment is performed would be different. There are four treatment combinations of solution temperature and solution concentration: (-1, -1), (-1, 1), (1, -1), (1, 1). The experimenter randomly choose one of these four treatments to set up first. Four copper strips are placed in the solution. Two of the four strips are randomly assigned to the low current level. The remaining two strips are assigned to the high current level. The plating is performed and the response is measured. A second treatment combination of temperature and concentration is chosen and the same procedure is followed. This is done for all four temperature / concentration combinations.

Running the experiment in this way also results in a split plot design where the whole plot factors are now solution concentration and solution temperature and the subplot factor is current.

In this experiment, one size experimental unit is again an individual copper strip. The treatment or factor that was applied to the individual strips is current (this factor was changed each time for a different strip within the solution). The other or larger size experimental unit is again a set of four copper strips. The treatments or factors that were applied to a set of four strips are solution concentration and solution temperature (these factors were changed after four strips were processed).

The smaller size experimental unit is again referred to as the subplot experimental unit. There are a total of 16 subplot experimental units for this experiment. Current is the subplot factor in this experiment.

The larger size experimental unit is the whole plot experimental unit. There are a total of four whole plot experimental units in this experiment. Solution concentration and solution temperature are the whole plot factors in this experiment.

Since there are two sizes of experimental units in the experiment, there are two error terms in the model, one that corresponds to the whole plot error or whole plot experimental unit and one that corresponds to the sub plot error or subplot experimental unit.

The ANOVA for this experiment looks as follows:

Source	DF
Concentration	1
Temperature	1
Error (Whole plot) = Conc*Temp	1
Current	1
Conc*Current	1
Temp*Current	1
Conc*Temp*Current	1
Error (Subplot)	8

The first three sources come from the whole plot level and the next 5 come from the subplot level. Since there are 8 degrees of freedom for the subplot error term, this MSE can be used to test each effect that involves current.

Consider running the experiment under the third scenario listed above There is only one copper strip in the solution at one time, however, two strips, one at the low current and one at the high current, are processed one right after the other under the same temperature and concentration setting. Once two strips have been processed, the concentration is changed and the temperature is reset to another combination. Two strips are again processed one after the other under this temperature and concentration setting. This process is continued until all 16 copper strips have been processed.

Running the experiment in this way also results in a split plot design where the whole plot factors are again solution concentration and solution temperature and the subplot factor is current. In this experiment, one size experimental unit is an individual copper strip. The treatment or factor that was applied to the individual strips is current (this factor was changed each time for a different strip within the solution). The other or larger size experimental unit is a set of two copper strips. The treatments or factors that were applied to a pair of two strips are solution concentration and solution temperature (these factors were changed after two strips were processed). The smaller size experimental unit is referred to as the subplot experimental unit.

There are a total of 16 subplot experimental units for this experiment. Current is the subplot factor in this experiment. There are a total of eight whole plot experimental units in this experiment. Solution concentration and solution temperature are the whole plot factors in this experiment. There are two error terms in the model, one that corresponds to the whole plot error or whole plot experimental unit and one that corresponds to the subplot error or subplot experimental unit.

The ANOVA for this (third) approach is as follows:

Source	DF
Concentration	1
Temperature	1
Conc*Temp	1
Error (Whole plot)	4
Current	1
Conc*Current	1
Temp*Current	1
Conc*Temp*Current	1
Error (Subplot)	4

The first four terms come from the whole plot analysis and the next 5 terms come from the subplot analysis. Note that we have separate error terms for both the whole plot and the subplot effects, each based on 4 degrees of freedom.

As can be seen from the three scenarios above, one of the major differences in split plot designs versus simple factorial designs are the number of different sizes of experimental units in the experiment. Split plot designs have more than one size experimental unit, i.e. more than one error term. Since these designs involve different sizes of experimental units and different variances, the standard errors of the various mean comparisons involve one or more of the variances. Specifying the appropriate model for a split-plot design involves being able to identify each size of experimental unit. Each experimental unit has its own design structure (for example, a completely randomized design versus a randomized complete block design) and its own treatment structure (for example, a full 2**3 factorial, a resolution V half fraction, a two-way treatment structure with a control group). As a result of having greater than one size experimental unit, the appropriate model used to analyze split plot designs are mixed models.

If the data from the experiment are analyzed where there is only one error term used in the model, misleading and invalid conclusions can be drawn from the results. For a more detailed discussion of these designs and the appropriate analysis procedures see Milliken, Analysis of Messy Data, Vol. 1.

Strip Plot Designs

Similar to a split plot design, a strip-plot design can result when some type of restricted randomization has occurred during the experiment. A simple factorial design can result in a strip-plot design depending on how the experiment was conducted. Strip-plot designs often result from experiments that are conducted over two or more process steps where each process step is a batch process, i.e. to complete each treatment

combination of the experiment requires more than one processing step where at each process step experimental units are processed together. As in the split-plot design, strip-plot designs result when the randomization in the experiment has been restricted in some way. As a result of the restricted randomization that occurs in strip-plot designs, there are multiple sizes of experimental units and therefore, are different error terms or different error variances that are used to test the factors of interest in the design. A traditional strip-plot design has three sizes of experimental units

Consider the following example from the semiconductor industry. An experiment requires an implant step and an anneal step. Both the anneal and the implant steps contain three factors to test. The implant process accommodates 12 wafers in a batch, and implanting a single wafer under a specified set of conditions is not practical nor does doing so represent economical use of the implanter. The anneal furnace can handle up to 100 wafers.

The figure below shows the design structure for how the experiment was run. In the figure below, the rectangles at the top of the diagram represent the settings for a two-level factorial design for the three factors in the implant step (A, B, C). Similarly, the rectangles at the lower left of the diagram represent a two-level factorial design for the three factors in the anneal step (D, E, F).

The arrows connecting each set of rectangles to the grid in the center of the diagram represent a randomization of trials in the experiment. The horizontal elements in the grid represent the experimental units for the anneal factors. The vertical elements in the grid represent the experimental units for the implant factors. The intersection of the vertical and horizontal elements represent the experimental units for the interaction effects between the implant factors and the anneal factors. Therefore, this experiment contains three sizes of experimental units, each one of which will have a unique error term for estimating the significance of effects.

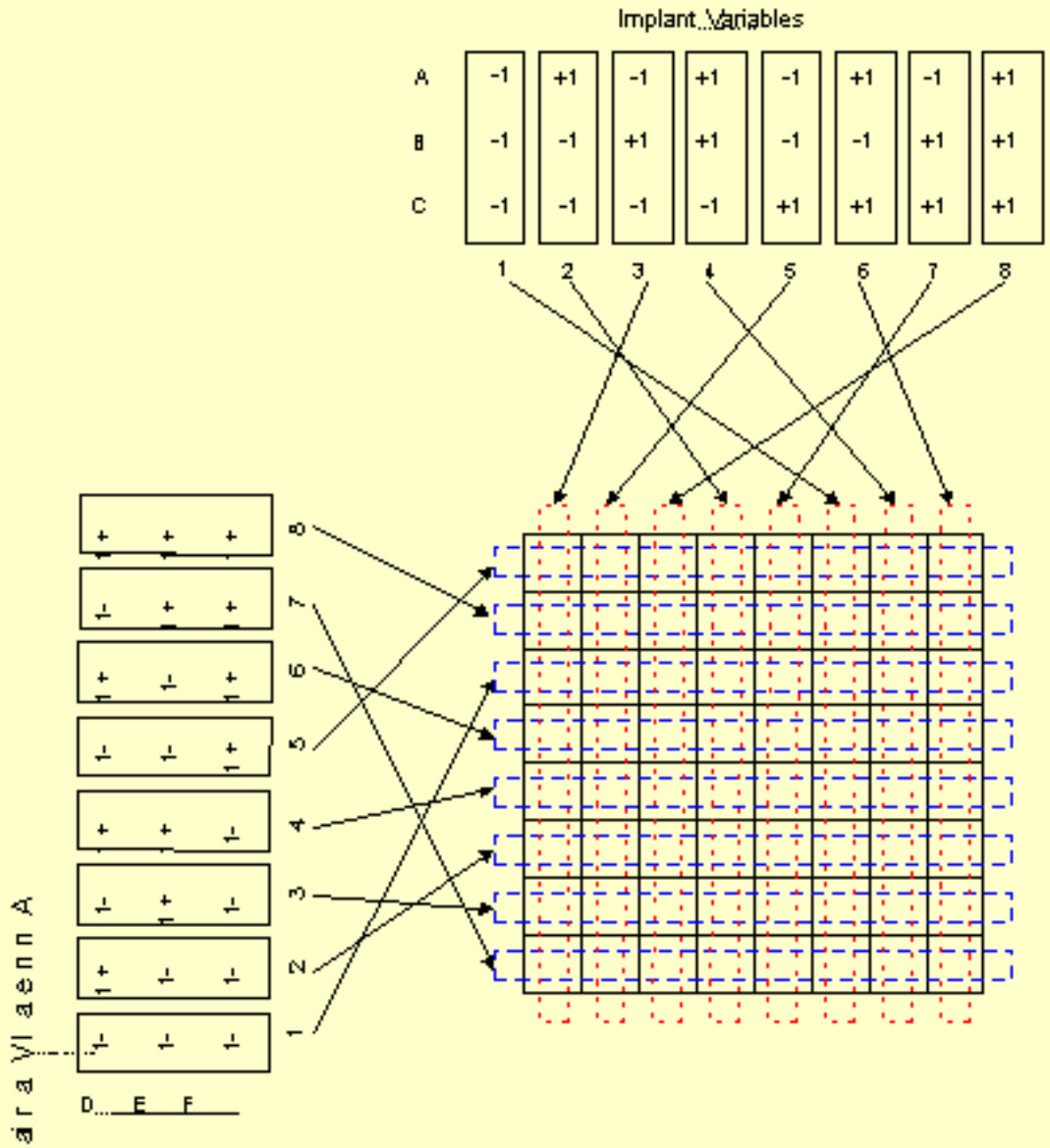


FIGURE 5.16 Diagram of a strip-plot design involving two process steps with three factors in each step

To put actual physical meaning to each of the experimental units in the above example, consider each cell in the grid as an individual wafer. A batch of eight wafers go through the implant step first. According to the figure, treatment combination 3 in factors A, B, and C is the first implant treatment run. This implant treatment is applied to all eight wafers at once. Once the first implant treatment is finished, another set of eight wafers gets implanted with treatment combination 5 of factors A, B, and C. This continues until the last batch of eight wafers gets implanted with treatment combination 6 of factors A, B, and C. Once all of the eight treatment combination of the implant factors have been run then the anneal step starts. The first anneal treatment combination to be run is treatment combination 5 of factors D, E, and F. This anneal treatment combination gets applied to a set of eight wafers, where each of these eight wafers comes from one of each of the eight implant treatment combinations. After this first batch of wafers have been

annealed, then the second anneal treatment is applied to a second batch of eight wafers, where again, these eight wafers come from one each of the eight implant treatment combinations. This is continued until the last batch of eight wafers have been implanted with a particular combination of factors D, E, and F.

Running the experiment in this way results in a strip-plot design with three sizes of experimental units. A set of eight wafers that get implanted together is the experimental unit for the implant factors A, B, and C and all of their interactions. There are a total of eight experimental units for the implant factors. A different set of eight wafers are annealed together. This different set of eight wafers is the second size experimental unit and is the experimental unit for the anneal factors D, E, and F and all of their interactions. The third size experimental unit is a single wafer. This is the experimental unit for all of the interaction effects between the implant factors and the anneal factors.

Actually, the above figure of the strip-plot design represents one block or one replicate of this experiment. If the experiment contains no replication and the model for the implant contains only the main effects and two-factor interactions, then the three-factor interaction term A*B*C (1 degree of freedom) provides the error term for the estimation of effects within the implant experimental unit. Invoking a similar model for the anneal experimental unit produces the three-factor interaction term D*E*F for the error term (1 degree of freedom) for effects within the anneal experimental unit.

For more details about strip-plot designs, see Milliken and Johnson (1987) or Miller (1997).

5.5.6. What are Taguchi designs?

Taguchi designs are a subset of classical factorial and fractional factorial designs - many of which are large screening designs

Genichi Taguchi, a Japanese engineer, proposed several approaches to experimental designs which are sometimes called 'Taguchi Methods.' These methods utilize two, three, and mixed level fractional factorial designs. Large screening designs seem to be particularly favored by Taguchi adherents.

Taguchi refers to experimental design as 'off-line quality control' because it is a method of ensuring good performance in the design stage of product or process. Some experimental designs, however, such as evolutionary operation, can be done on-line while the process is running. He has also published a booklet of design nomograms ('Orthogonal Arrays and Linear Graphs,' 1987, American Supplier Institute) which may be used as a design guide, similar to the table of fractional factorial designs given previously in Section 5.3. Some of well known Taguchi orthogonal arrays (L9, L18, L27 and L36) were given earlier when three-level, mixed level and fractional factorial designs were discussed.

If this were the only aspects of 'Taguchi Designs,' there would be little additional reason to consider them over and above our previous discussion on factorials. 'Taguchi' designs are largely our familiar factorial designs. However, Taguchi has introduced several noteworthy new ways of conceptualizing an experiment that are very valuable, especially in development and industrial engineering, and we will look at two of his main ideas, namely Parameter Design, and Tolerance Design

Taguchi advocated using inner and outer array designs to take into account noise factors (outer) and design factors (inner)

Parameter Design.

The aim here is to make a product or process less variable (more robust) in the face of variation over which we have little or no control. A simple fictitious example might be that of the starter motor of an automobile that has to perform reliably in the face of variation in ambient temperature, and varying states of battery weakness. The engineer here has control over, say, number of armature turns, gauge of armature wire, and ferric content of magnet alloy.

Conventionally, one can view this as an experiment in five factors. Taguchi has pointed out the usefulness of viewing it as a set up of three inner array factors (turns, gauge, ferric %) over which we have design control, plus an outer array of factors over which we have control only in the laboratory (temperature, battery voltage).

Pictorially, we can think about this design as being a conventional design in the inner array factors (compare Figure 3.1) with the addition of a 'small' outer array factorial design at each corner of the 'inner array' box.

Set I1 = 'turns,' I2 = 'gauge,' I3 = 'ferric %,' E1 = 'temperature, and E2 = 'voltage.' Then

we construct a 2^3 design 'box' for the I's, and at each of the eight corners so constructed, we place a 2^2 design 'box' for the E's, as is shown in Figure 5.17.

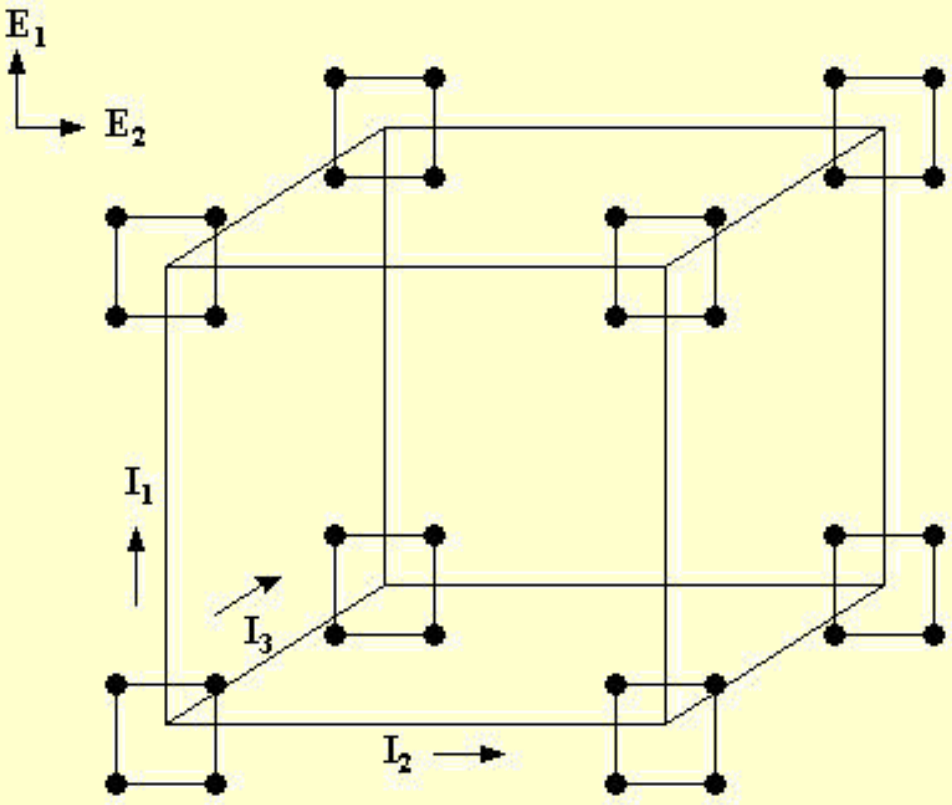


FIGURE 5.17 Inner 2^3 and outer 2^2 arrays for robust design with 'I' the inner array, 'E' the outer array.

We now have a total of 8x4 = 32 experimental settings, or runs. These are set out in Table 5.7, where the 2^3 design in the I's is given in standard order on the left of the table, and the 2^2 design in the E's is written out sideways along the top. Note that the experiment would not be run in the standard order but should, as always, have its runs randomized. The output measured is the percent of (theoretical) maximum torque.

An example of an inner and outer array designed experiment

TABLE 5.7 Design table, in standard order(s) for the parameter design of Figure 5.9

Run #	I1	I2	I3		1	2	3	4	Output	Output
				E1	-1	1	-1	1		
				E2	-1	-1	1	1	MEAN	STD.DEV
1	-1	-1	-1		75	86	67	98	**81.5**	**13.5**
2	1	-1	-1		87	78	56	91	**78.0**	**15.6**
3	-1	1	-1		77	89	78	8	**63.0**	**37.1**
4	1	1	-1		95	65	77	95	**83.0**	**14.7**
5	-1	-1	1		78	78	59	94	**77.3**	**14.3**
6	1	-1	1		56	79	67	94	**74.0**	**16.3**
7	-1	1	1		79	80	66	85	**77.5**	**8.1**
8	1	1	1		71	80	73	95	**79.8**	**10.9**

Note that there are four outputs measured on each row. These correspond to the four 'outer array' design points at each corner of the 'outer array' box. As there are eight corners of the outer array box, there are eight rows in all.

Each row yields a mean and standard deviation % of maximum torque. Ideally there would be one row that had both the highest average torque and the lowest standard deviation (variability). Row 4 has the highest torque, and row 7 has the lowest variability, so we are forced to compromise. We can't simply 'pick the winner.'

Taguchi also advocated tolerance studies to determine, based on a loss or cost function, which parameters have critical tolerances that need to be tightened

One might also observe that all the outcomes we have occur at the corners of the design 'box' which means that we cannot see 'inside' the box. An optimum point might occur within the box, and we can search for such a point using contour plots. Contour plots were illustrated in the example of response surface design analysis given in Section 4..

Note that we could have used fractional factorials for one or both the inner and outer array designs.

Tolerance Design

This section deals with the problem of how, and when, to specify tightened tolerances for a product or a process so that quality and performance/productivity are enhanced. Every product or process has a number—perhaps a large number—of components. We explain here how to identify which are the critical components to target when tolerances have to be tightened.

It is a natural impulse to believe that the quality and performance of any item can easily be improved by merely tightening up on some or all of its tolerance requirements. By this we mean that if the old version of the item specified, say, machining to ± 1 micron, we naturally believe that we can get better performance by specifying machining to ± ½ micron.

This can get expensive and, besides, is often not a guarantee of much better performance. One has merely to witness the high initial and maintenance costs of such tight-tolerance-level items as space vehicles, expensive automobiles, etc. to realize that tolerance design—the selection of critical tolerances and the re-specification of those critical tolerances—is not a task to be undertaken without careful thought. In fact, it is

recommended that *only after extensive parameter design studies have been completed* should tolerance design be done *as a last resort* to improve quality and productivity.

Example

Customers for an electronic component complained to its supplier that the measurement reported by the supplier on the as-delivered items appeared to be imprecise. The supplier undertook to investigate the matter.

The supplier's engineers reported that the measurement in question was made up of two components, namely x and y, and the final measurement M was reported according to the standard formula

$$\mathbf{M = K\ x/y}$$

where 'K' was a known physical constant. Components x and y were measured separately in the laboratory using two different techniques, and the results combined by software to produce M. Buying new measurement devices for both components would be prohibitively expensive, and it was not even know by how much the x or y component tolerances should be improved to give the desired improvement in the precision of M.

Assume that in a measurement of a standard item the 'true' value of x is x_o, and for y it is y_o. Let f(x, y) =M; then the Taylor Series expansion for f(x, y) is

$$f(x,y) = f(x_0,y_0) + (x-x_0)\frac{df}{dx} + (y-y_0)\frac{df}{dy} + (x-x_0)^2\frac{d^2f}{dx^2} + (y-y_0)^2\frac{d^2f}{dy^2} + (x-x_0)(y-y_0)\frac{d^2f}{dxdy} + (\text{higher order terms})$$

and where all the partial derivatives 'df/dx' etc. are evaluated at (x_o, y_o).

Applying this formula to M(x, y) = Kx/y, we obtain

$$M(x,y) = K\frac{x_0}{y_0} + (x-x_0)\frac{K}{y_0} - (y-y_0)\frac{K}{y_0^2} - 2(y-y_0)^2\frac{K}{y_0^3} - (x-x_0)(y-y_0)\frac{K}{y_0^2} + (\text{higher order terms})$$

It is assumed known from experience that the measurements of x show a distribution with average value x_o, and with standard deviation σ_x = 0.003 x-units.

In addition, we assume that the distribution of x is normal. Since 99.74% of a normal distribution's range is covered by 6σ, we take $3\sigma_x$ = 0.009 x-units to be the existing

tolerance T_x for measurements on x. That is, $T_x = \pm 0.009$ x-units is the 'play' around x_o that we expect from the existing measurement system.

It is also assumed known that the y measurements show a normal distribution around y_o, with standard deviation $\sigma_y = 0.004$ y-units. Thus $T_y = \pm 3\sigma_y = \pm 0.012$.

Now $\pm T_x$ and $\pm T_y$ may be thought of as 'worst case' values for $(x-x_o)$ and $(y-y_o)$. Substituting T_x for $(x-x_o)$ and T_y for $(y-y_o)$ in the expanded formula for M(x, y) we have

$$M_T = K\frac{x_0}{y_0} + T_x\frac{K}{y_0} - T_y\frac{K}{y_0^2} - 2T_y^2\frac{K}{y_0^3} - T_xT_y\frac{K}{y_0^2} + (\text{higher order terms})$$

$T_y{}^2$ and T_xT_y terms, and all terms of higher order, are going to be at least an order of magnitude smaller than terms in T_x and in T_y, and for this reason we drop them, so that

$$M_T \approx K\frac{x_0}{y_0} + T_x\frac{K}{y_0} - T_y\frac{K}{y_0^2}$$

Thus a 'worst case' Euclidean distance Δ of M(x, y) from its ideal value Kx_o/y_o is (approximately)

$$\Delta = \sqrt{\left(T_x\frac{K}{y_0}\right)^2 + \left(T_y\frac{K}{y_0^2}\right)^2} = \sqrt{\left(0.009\frac{K}{y_0}\right)^2 + \left(0.012\frac{K}{y_0^2}\right)^2}$$

$$= \sqrt{(0.00008 + 0.000144)\left(\frac{K}{y_0}\right)}$$

This shows the components of the variation in the measurement: The *x* component contributes 0.000081, the *y* component contributes $0.000144/y_o{}^2$.

As y_o is a known quantity, and reduction in T_x and in T_y each carries its own price tag, it becomes an economic decision whether one should spend resources to reduce T_x or T_y , or both.

In this example, we have used a Taylor series approximation to obtain a simple expression that highlights the benefit of T_x and T_y. Alternatively, one might simulate values of **M = K**

x/y, given a specified (T_x,T_y) and (x_0,y_0), and then summarize the results with a model for the variability of **M** as a function of (T_x,T_y).

In other applications, no functional form is available, and one must use experimentation to empirically determine the optimal tolerance design. See Bisgaard and Steinberg (1997).

5.5.7. What are John's 3/4 fractional factorial designs?

John's designs require only 3/4 of the number of runs a full 2^n factorial would require

Three-quarter (¾) designs are two-level factorial designs that require only three-quarters of the number of runs of the 'original' design. For example, instead of running all the sixteen runs required for a 2^4 fractional factorial design, we need only run 12 of them. Such designs were invented by Professor Peter John of the University of Texas, and are sometimes called 'John's ¾ designs.'

¾ fractional factorial designs can be used to save on resources in two different contexts. In the first place, we may wish to do more runs after having completed a fractional factorial, so as to de-alias certain specific interaction patterns; in the second place, we may wish to do a ¾ design to begin with and so save on 25% of the run requirement of a regular design.

Semifolding Example

We have four experimental factors to investigate, namely X1, X2, X3, and X4, and we have designed and run a 2^{4-1} fractional factorial design. Such a design has eight runs, or rows, if we don't count center point runs (or replications).

The 2^{4-1} design is of resolution IV which means that main effects are confounded with, at worst, three-factor interactions, but that two factor interactions are confounded with other two factor interactions.

The design layout matrix, in standard order, is shown in Table 5.8 along with all the two-factor interaction columns. Note that the column for X4 is constructed by multiplying columns for X1, X2, and X3 together (i.e. 4=123).

Table 5.8 The 2^{4-1} design plus 2-factor interaction columns shown in standard order. Note that 4=123.

					TWO-FACTOR INTERACTION COLUMNS					
	X1	X2	X3	X4	X1X2	X1X3	X1X4	X2X3	X2X4	X3X4
1	-1	-1	-1	-1	1	1	1	1	1	1
2	1	-1	-1	1	-1	-1	1	1	-1	-1
3	-1	1	-1	1	-1	1	-1	-1	1	-1
4	1	1	-1	-1	1	-1	-1	-1	-1	1
5	-1	-1	1	1	1	-1	-1	-1	-1	1
6	1	-1	1	-1	-1	1	-1	-1	1	-1
7	-1	1	1	-1	-1	-1	1	1	-1	-1
8	1	1	1	1	1	1	1	1	1	1

Note also that 12=34, 13=24, and 14=23. These follow from the generating relationship 4=123, and tells us that we cannot estimate any two-factor interaction that is free of some other two-factor alias.

Suppose that we became interested in estimating some or all of the two-factor interactions that involved factor X1; that is, we want to estimate one or more of the relationships 12, 13, and 14 free of two-factor confounding.

One way of doing this is to run the 'other half' of the design—an additional eight rows formed from the relationship 4=-123. Putting these two 'halves' together—the original one and the new one, we'd get a 2^4 design in sixteen runs. Eight of these runs would already have been run, so all we'd need to do is run the remaining half.

There is a way to get what we want while only adding four more runs. These runs are selected in the following manner: take the four rows of Table 5.8 that have '-1' in the 'X1' column, and switch the '-' sign under X1 to '+' to get the four-row table of Table 5.9. This is called a foldover on X1, choosing the subset of runs with X1 = -1. Note that this choice of 4 runs is not unique, and that if the initial design suggested that X1 = -1 was a desirable level, we would have chosen to experiment at the other four treatment combinations that were omitted from the initial design.

TABLE 5.9 Foldover on 'X1' of the 2^{4-1} design of Table 5.5

9	1	-1	-1	-1
10	1	1	-1	1
11	1	-1	1	1
12	1	1	1	-1

Add this new block of rows to the bottom of Table 5.8 to give a design in twelve rows. We show this done in Table 5.10, and also add in the two-factor interactions as well for illustration (not needed when we do the runs).

TABLE 5.10 A twelve-run design based on the 2^{4-1} also showing all two-factor interaction columns

					TWO-FACTOR INTERACTION COLUMNS					
	X1	X2	X3	X4	X1X2	X1X3	X1X4	X2X3	X2X4	X3X4
1	-1	-1	-1	-1	1	1	1	1	1	1
2	1	-1	-1	1	-1	-1	1	1	-1	-1
3	-1	1	-1	1	-1	1	-1	-1	1	-1
4	1	1	-1	-1	1	-1	-1	-1	-1	1
5	-1	-1	1	1	1	-1	-1	-1	-1	1
6	1	-1	1	-1	-1	1	-1	-1	1	-1
7	-1	1	1	-1	-1	-1	1	1	-1	-1
8	1	1	1	1	1	1	1	1	1	1
9	1	-1	-1	-1	-1	-1	-1	1	1	1
10	1	1	-1	1	1	-1	1	-1	1	-1
11	1	-1	1	1	-1	1	1	-1	-1	1
12	1	1	1	-1	1	1	-1	1	-1	-1

Examine the two-factor interaction columns and convince yourself that no two are alike. This means that no two-factor interaction involving X1 is aliased with any other two-factor interaction, which means that the design we now have is at least of resolution V when considering factor X1. In fact, it is of resolution V for all factors—which is not always the case when constructing these types of ¾ foldover designs.

What we now possess is a design that has 12 runs, and with which we can estimate all the two-factor interaction of X1 free of aliasing with any other two-factor interaction. It is called a ¾ design because it has ¾ the number of rows than the next regular factorial design (a 2^4).

If one fits a model with an intercept, a block effect, the four main effects and the six two factor interactions then each coefficient has a standard error of $\sigma/8^{1/2}$ - instead of $\sigma/12^{1/2}$ - because the design is not orthogonal and each estimates is correlated with two other estimates. Note that no degrees of freedom exist for estimating σ. Instead, one should plot the 10 effect estimates using a normal (or half-normal) effects plot to judge which effects to keep.

For more details on ¾ fractions obtained by adding a follow-up design half the size of the original design, see Mee and Peralta (2000).

Next we consider an example where a ¾ fraction arises when the (¾) 2^{k-p} design is planned from the start because it is an efficient design that allows estimation of a sufficient number of effects.

A 48 Run 3/4 Design Example

Suppose we wish to run an experiment for k=8 factors, for which we want to estimate all main effects and two factor interactions. We could use the 2_V^{8-2} desgn described in the summary table of fractional factorial designs, but this would require a 64 run experiment to estimate the 1 + 8 + 28 = 37 desired coefficients. In this context, and for larger resolution V designs, ¾ of the resolution design will generally suffice.

The 48 run design is constructed as follows: start by creating the full 2_V^{8-2} design using the generators 7 = 1234 and 8 = 1256. The defining relation is I = 12347 = 12568 = 345678 (see the summary table details for this design).

Next arrange these 64 treatment combination into four blocks of size 16, blocking on the interactions 135 and 246 (i.e. block 1 has 135 = 246 = -1 runs, block 2 has 135 = -1, 246 = +1, block 3 has 135 = +1, 246 = -1 and block 4 has 135 = 246 = +1). If we exclude the first block where 135 = 246 = -1, we have the desired ¾ design reproduced below (the reader can verify that these are the runs described in the summary table, excluding the runs numbered 1, 6, 11, 16, 18, 21, 28, 31, 35, 40, 41,46, 52, 55, 58 and 61).

X1	X2	X3	X4	X5	X6	X7	X8
+1	-1	-1	-1	-1	-1	-1	-1
-1	+1	-1	-1	-1	-1	-1	-1
+1	+1	-1	-1	-1	-1	+1	+1
-1	-1	+1	-1	-1	-1	-1	+1
-1	+1	+1	-1	-1	-1	+1	-1
+1	+1	+1	-1	-1	-1	-1	+1
-1	-1	-1	+1	-1	-1	-1	+1
+1	-1	-1	+1	-1	-1	+1	-1
+1	+1	-1	+1	-1	-1	-1	+1
-1	-1	+1	+1	-1	-1	+1	+1
+1	-1	+1	+1	-1	-1	-1	-1
-1	+1	+1	+1	-1	-1	-1	-1
-1	-1	-1	-1	+1	-1	+1	-1
-1	+1	-1	-1	+1	-1	-1	+1
+1	+1	-1	-1	+1	-1	+1	-1
+1	-1	+1	-1	+1	-1	+1	+1
-1	+1	+1	-1	+1	-1	+1	+1
+1	+1	+1	-1	+1	-1	-1	-1
-1	-1	-1	+1	+1	-1	-1	-1
+1	-1	-1	+1	+1	-1	+1	+1

-1	+1	-1	+1	+1	-1	+1	+1
-1	-1	+1	+1	+1	-1	+1	-1
+1	-1	+1	+1	+1	-1	-1	+1
+1	+1	+1	+1	+1	-1	+1	-1
-1	-1	-1	-1	-1	+1	+1	-1
+1	-1	-1	-1	-1	+1	-1	+1
+1	+1	-1	-1	-1	+1	+1	-1
-1	-1	+1	-1	-1	+1	-1	-1
+1	-1	+1	-1	-1	+1	+1	+1
-1	+1	+1	-1	-1	+1	+1	+1
+1	-1	-1	+1	-1	+1	+1	+1
-1	+1	-1	+1	-1	+1	+1	+1
+1	+1	-1	+1	-1	+1	-1	-1
-1	-1	+1	+1	-1	+1	+1	-1
-1	+1	+1	+1	-1	+1	-1	+1
+1	+1	+1	+1	-1	+1	+1	-1
-1	-1	-1	-1	+1	+1	+1	+1
+1	-1	-1	-1	+1	+1	-1	-1
-1	+1	-1	-1	+1	+1	-1	-1
-1	-1	+1	-1	+1	+1	-1	+1
+1	-1	+1	-1	+1	+1	+1	-1
+1	+1	+1	-1	+1	+1	-1	+1
-1	-1	-1	+1	+1	+1	-1	+1
-1	+1	-1	+1	+1	+1	+1	-1
+1	+1	-1	+1	+1	+1	-1	+1
+1	-1	+1	+1	+1	+1	-1	-1
-1	+1	+1	+1	+1	+1	-1	-1
+1	+1	+1	+1	+1	+1	+1	+1

This design provides 11 degrees of freedom for error and also provides good precision for coefficient estimates (some of the coefficients have a standard error of $\sigma/32^{.5}$ and some have a standard error of $\sigma/42.55^{.5}$).

More about John's ¾ designs can be found in John (1971) or Diamond (1989).

5.5.8. What are small composite designs?

Small composite designs save runs compared to Resolution V response surface designs by adding star points to a Resolution III design

Response surface designs (RSD) were described earlier. A typical RSD requires about 13 runs for 2 factors, 20 runs for 3 factors, 31 runs for 4 factors, and 32 runs for 5 factors. It is obvious that, once you have four or more factors you wish to include in a RSD, you will need more than one lot of experimental units for your basic design. This is what most statistical software today will give you, including RS/1, JMP, and SAS. However, there is a way to cut down on the number of runs, as suggested by H.O Hartley in his paper '*Smallest Composite Designs for Quadratic Response Surfaces*' published in Biometrics, December 1959

This method addresses the theory that using a Resolution V design as the smallest fractional design to create an RSD is unnecessary. This method adds star points to designs of Resolution III, and then uses the star points toclear the main effects of aliasing with the two-factor interactions. This method will not estimate three-factor interactions or higher. It provides poor interaction coefficient estimates, and should not be used unless the error variability is negligible compared to the systematic effects of the factors.

This could be particularly useful when you have a design containing four or five factors and you wish to only use the experimental units from one lot.

Design for 4 factors

The following is a design for four factors. You would want to randomize these runs before implementing them. -1 and +1 represent the low and high settings of each factor.

TABLE 5.11 Four factors: Factorial design section is based on a generator of I = X1*X2*X3, Resolution III. -α and +α are the star points, calculated beyond the factorial range. 0 represents the midpoint of the factor range.

Row	X1	X2	X3	X4
1	1	-1	-1	-1
2	-1	1	-1	-1
3	-1	-1	1	-1
4	1	1	1	-1
5	1	-1	-1	1
6	-1	1	-1	1
7	-1	-1	1	1
8	1	1	1	1
9	$-\alpha$	0	0	0
10	α	0	0	0
11	0	$-\alpha$	0	0
12	0	α	0	0
13	0	0	$-\alpha$	0
14	0	0	α	0
15	0	0	0	$-\alpha$
16	0	0	0	α
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0

Determining α in Small Composite Designs

To maintain rotatability for usual CCD's, the value of α was determined by the number of treatment combinations in the factorial portion of the central composite design:

$$\alpha = \left[number\ of\ factorial\ runs \right]^{1/4}$$

However, small composite designs are not rotatable, regardless of the choice of α. For small composite designs, α should not be smaller than [number of factorial runs]$^{1/4}$ nor larger than $k^{1/2}$.

5.5.9. An EDA approach to experimental design

Introduction

This section presents an exploratory data analysis (EDA) approach to analyzing the data from a designed experiment. This material is meant to complement, not replace, the more model-based approach for analyzing experiment designs given in section 4 of this chapter.

Choosing an appropriate design is discussed in detail in section 3 of this chapter.

Starting point

Problem category

The problem category we will address is the screening problem. Two characteristics of screening problems are:

1. There are many factors to consider.
2. Each of these factors may be either continuous or discrete.

Desired output

The desired output from the analysis of a screening problem is:

- A ranked list (by order of importance) of factors.
- The best settings for each of the factors.
- A good model.
- Insight.

Problem essentials

The essentials of the screening problem are:

- There are k factors with n observations.
- The generic model is:

$$Y = f(X_1, X_2, ..., X_k)$$

Design type

In particular, the EDA approach is applied to 2^k full factorial and 2^{k-p} fractional factorial designs.

An EDA approach is particularly applicable to screening designs because we are in the preliminary stages of understanding our process.

EDA philosophy

EDA is not a single technique. It is an approach to analyzing data.

- EDA is data-driven. That is, we do not assume an initial model. Rather, we attempt to let the data speak for themselves.
- EDA is question-based. That is, we select a technique to answer one or more questions.
- EDA utilizes multiple techniques rather than depending on a single technique. Different plots have a different basis, focus, and sensitivities, and therefore may bring out different aspects of the data. When multiple techniques give us a redundancy of conclusions, this increases our confidence that our conclusions are valid. When they give conflicting conclusions, this may be giving us a clue as to the nature of our data.
- EDA tools are often graphical. The primary objective is to provide insight into the data, which graphical techniques often provide more readily than quantitative techniques.

10-Step process

The following is a 10-step EDA process for analyzing the data from 2^k full factorial and 2^{k-p} fractional factorial designs.

1. Ordered data plot
2. Dex scatter plot
3. Dex mean plot
4. Interaction effects matrix plot
5. Block plot
6. DEX Youden plot
7. |Effects| plot
8. Half-normal probability plot
9. Cumulative residual standard deviation plot
10. DEX contour plot

Each of these plots will be presented with the following format:

- Purpose of the plot
- Output of the plot
- Definition of the plot

- Motivation for the plot
- An example of the plot using the defective springs data
- A discussion of how to interpret the plot
- Conclusions we can draw from the plot for the defective springs data

Data set

Defective springs data

The plots presented in this section are demonstrated with a data set from Box and Bisgaard (1987).

These data are from a 2^3 full factorial data set that contains the following variables:

1. Response variable Y = percentage of springs without cracks
2. Factor 1 = oven temperature (2 levels: 1450 and 1600 F)
3. Factor 2 = carbon concentration (2 levels: .5% and .7%)
4. Factor 3 = quench temperature (2 levels: 70 and 120 F)

Y Percent Acceptable	X1 Oven Temperature	X2 Carbon Concentration	X3 Quench Temperature
67	-1	-1	-1
79	+1	-1	-1
61	-1	+1	-1
75	+1	+1	-1
59	-1	-1	+1
90	+1	-1	+1
52	-1	+1	+1
87	+1	+1	+1

You can read this file into Dataplot with the following commands:

```
SKIP 25
READ BOXSPRIN.DAT Y X1 X2 X3
```

5.5.9.1. Ordered data plot

Purpose

The ordered data plot answers the following two questions:

1. What is the best setting (based on the data) for each of the *k* factors?
2. What is the most important factor?

In the above two questions, the terms "best" and "important" need more precise definitions.

Settings may be declared as "best" in three different ways:

1. "best" with respect to the data;
2. "best" on average;
3. "best" with respect to predicted values from an adequate model.

In the worst case, each of the above three criteria may yield different "best settings". If that occurs, then the three answers must be consolidated at the end of the 10-step process.

The ordered data plot will yield best settings based on the first criteria (data). That is, this technique yields those settings that correspond to the best response value, with the best value dependent upon the project goals:

1. maximization of the response;
2. minimization of the response;
3. hitting a target for the response.

This, in turn, trivially yields the best response value:

1. maximization: the observed maximum data point;
2. minimization: the observed minimum data point;
3. target: the observed data value closest to the specified target.

With respect to the most "important" factor, this by default refers to the single factor which causes the greatest change in the value of the response variable as we proceed from the "-" setting to the "+" setting of the factor. In practice, if a factor has one setting for the best and near-best response values and the opposite setting for the worst and near-worst response values, then that factor is usually the most important factor.

Output

The output from the ordered data plot is:

1. Primary: Best setting for each of the *k* factors.
2. Secondary: The name of the most important factor.

Definition

An ordered data plot is formed by:

- Vertical Axis: The ordered (smallest to largest) raw response value for each of the *n* runs in the experiment.
- Horizontal Axis: The corresponding dummy run index (1 to *n*) with (at each run) a designation of the corresponding settings (- or +) for each of the *k* factors.

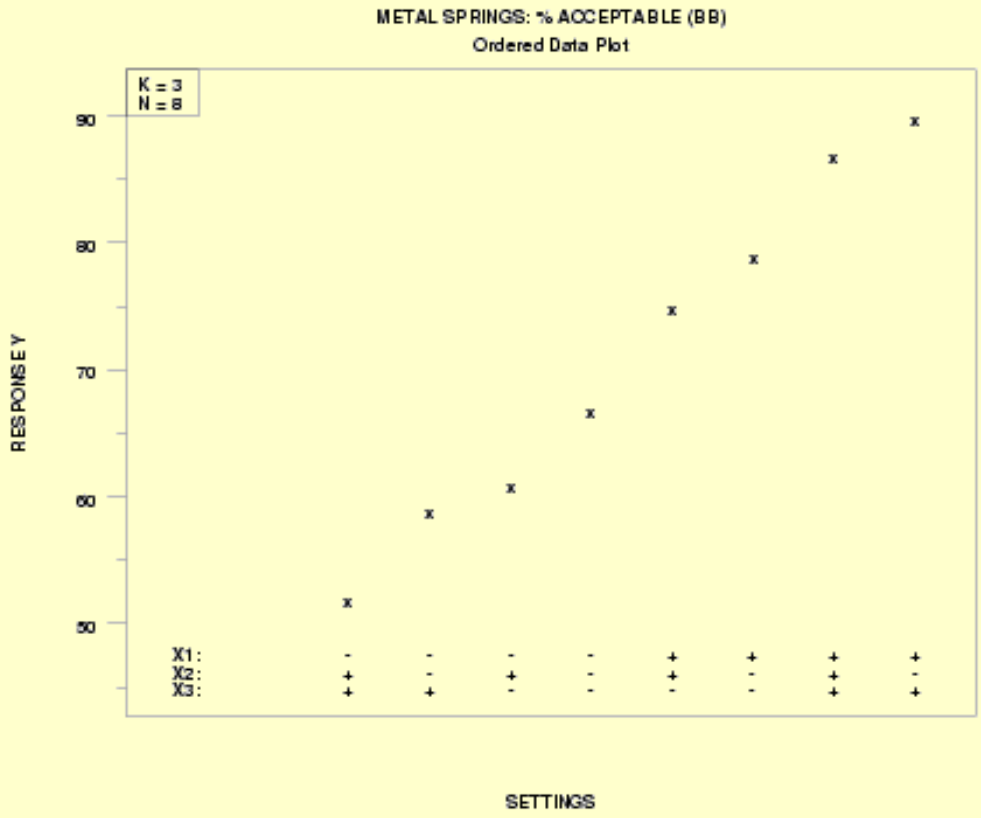
In essence, the ordered data plot may be viewed as a scatter plot of the ordered data versus a single *n*-treatment consolidation factor.

Motivation

To determine the best setting, an obvious place to start is the best response value. What constitutes "best"? Are we trying to maximize the response, minimize the response, or hit a specific target value? This non-statistical question must be addressed and answered by the analyst. For example, if the project goal is ultimately to achieve a large response, then the desired experimental goal is maximization. In such a case, the analyst would note from the plot the largest response value and the corresponding combination of the *k*-factor settings that yielded that best response.

Plot for defective springs data

Applying the ordered response plot for the defective springs data set yields the following plot.



How to interpret

From the ordered data plot, we look for the following:

1. best settings;
2. most important factor.

Best Settings (Based on the Data):

At the best (highest or lowest or target) response value, what are the corresponding settings for each of the *k* factors? This defines the best setting based on the raw data.

Most Important Factor:

For the best response point and for the nearby neighborhood of near-best response points, which (if any) of the *k* factors has consistent settings? That is, for the subset of response values that is best or near-best, do all of these values emanate from an identical level of some factor?

Alternatively, for the best half of the data, does this half happen to result from some factor with a common setting? If yes, then the factor that displays such consistency is an excellent candidate for being declared the "most important factor". For a balanced experimental design, when all of the best/near-best response values come from one setting, it follows that all of the worst/near-worst response values will come from the other setting of that factor. Hence that factor becomes "most important".

At the bottom of the plot, step though each of the *k* factors and determine which factor, if any, exhibits such behavior. This defines the "most important" factor.

Conclusions for the defective springs data

The application of the ordered data plot to the defective springs data set results in the following conclusions:

1. Best Settings (Based on the Data):

   ($X1$,$X2$,$X3$) = (+,-,+) = (+1,-1,+1) is the best setting since
   1. the project goal is maximization of the percent acceptable springs;
   2. Y = 90 is the largest observed response value; and
   3. ($X1$,$X2$,$X3$) = (+,-,+) at Y = 90.
2. Most important factor:

   $X1$ is the most important factor since the four largest response values (90, 87, 79, and 75) have factor $X1$ at +1, and the four smallest response values (52, 59, 61, and 67) have factor $X1$ at -1.

5.5.9.2. Dex scatter plot

Purpose

The dex (design of experiments) scatter plot answers the following three questions:

1. What are the most important factors?
2. What is the best setting for each of these important factors?
3. What data points are outliers?

In the above questions, the terms "important", "best", and "outliers" need clarification and specificity:

Important

A factor can be "important" if it leads to a significant shift in either the location or the variation of the response variable as we go from the "-" setting to the "+" setting of the factor. Both definitions are relevant and acceptable. The default definition of "important" in engineering/scientific applications is a shift in location. Unless specified otherwise, when a factor is claimed to be important, the implication is that the factor caused a large location shift in the response.

Best

A factor setting is "best" if it results in a typical response that is closest, in location, to the desired project goal (maximization, minimization, target). This desired project goal is an engineering, not a statistical, question, and so the desired optimization goal must be specified by the engineer.

Outlier

A data point is an "outlier" if it comes from a different probability distribution or from a different deterministic model than the remainder of the data. A single outlier in a data set can affect all effect estimates and so in turn can potentially invalidate the factor rankings in terms of importance.

Given the above definitions, the dex scatter plot is a useful early-step tool for determining the important factors, best settings, and outliers. An alternate name for the dex scatter plot is "main effects plot".

Output

The output for the dex scatter plot is:

1. Primary: Identification of the important factors.
2. Secondary: Best setting for these factors and identification of outliers.

Definition

The dex scatter plot is formed by

- Vertical Axis: The response (= the raw data) for a given setting (- or +) of a factor for each of the *k* factors.
- Horizontal Axis: The *k* factors, and the two settings (- and +) within each factor.

Motivation

The scatter plot is the primary data analysis tool for determining if and how a response relates to another factor. Determining if such a relationship exists is a necessary first step in converting statistical association to possible engineering cause-and-effect. Looking at how the raw data change as a function of the different levels of a factor is a fundamental step which, it may be argued, should never be skipped in any data analysis.

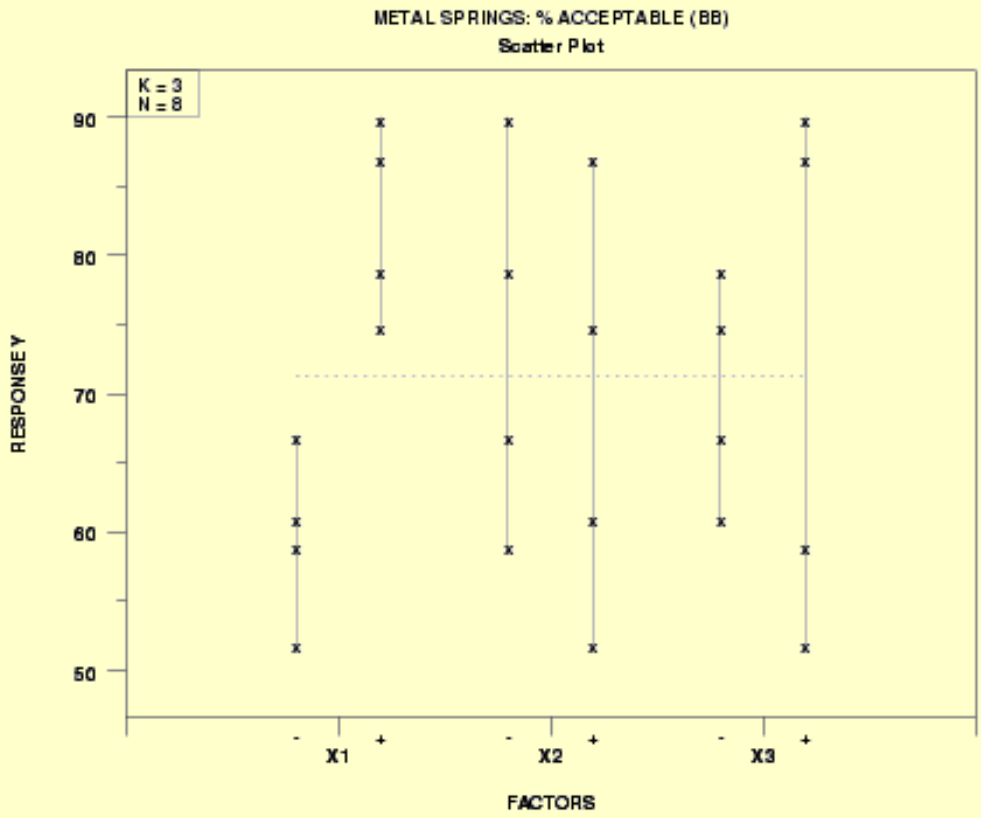
From such a foundational plot, the analyst invariably extracts information dealing with location shifts, variation shifts, and outliers. Such information may easily be washed out by other "more advanced" quantitative or graphical procedures (even computing and plotting means!). Hence there is motivation for the dex scatter plot.

If we were interested in assessing the importance of a single factor, and since "important" by default means shift in location, then the simple scatter plot is an ideal tool. A large shift (with little data overlap) in the body of the data from the "-" setting to the "+" setting of a given factor would imply that the factor is important. A small shift (with much overlap) would imply the factor is not important.

The dex scatter plot is actually a sequence of *k* such scatter plots with one scatter plot for each factor.

Plot for defective springs data

The dex scatter plot for the defective springs data set is as follows.



How to interpret

As discussed previously, the dex scatter plot is used to look for the following:

1. Most Important Factors;
2. Best Settings of the Most Important Factors;
3. Outliers.

Each of these will be discussed in turn.

Most Important Factors:

For each of the *k* factors, as we go from the "-" setting to the "+" setting within the factor, is there a location shift in the body of the data? If yes, then

1. Which factor has the biggest such data location shift (that is, has least data overlap)? This defines the "most important factor".
2. Which factor has the next biggest shift (that is, has next least data overlap)? This defines the "second most important factor".
3. Continue for the remaining factors.

In practice, the dex scatter plot will typically only be able to discriminate the most important factor (largest shift) and perhaps the second most important factor (next largest shift). The degree of overlap in remaining factors is frequently too large to ascertain with certainty the ranking for other factors.

Best Settings for the Most Important Factors:

For each of the most important factors, which setting ("-" or "+") yields the "best" response?

In order to answer this question, the engineer must first define "best". This is done with respect to the overall project goal in conjunction with the specific response variable under study. For some experiments (e.g., maximizing the speed of a chip), "best" means we are trying to maximize the response (speed). For other experiments (e.g., semiconductor chip scrap), "best" means we are trying to minimize the response (scrap). For yet other experiments (e.g., designing a resistor) "best" means we are trying to hit a specific target (the specified resistance). Thus the definition of "best" is an engineering precursor to the determination of best settings.

Suppose the analyst is attempting to maximize the response. In such a case, the analyst would proceed as follows:

1. For factor 1, for what setting (- or +) is the body of the data higher?
2. For factor 2, for what setting (- or +) is the body of the data higher?
3. Continue for the remaining factors.

The resulting *k*-vector of best settings:

(x1best, x2best, ..., xkbest)

is thus theoretically obtained by looking at each factor individually in the dex scatter plot and choosing the setting (- or +) that has the body of data closest to the desired optimal (maximal, minimal, target) response.

As indicated earlier, the dex scatter plot will typically be able to estimate best settings for only the first few important factors. Again, the degree of data overlap precludes ascertaining best settings for the remaining factors. Other tools, such as the dex mean plot, will do a better job of determining such settings.

Outliers:

Do any data points stand apart from the bulk of the data? If so, then such values are candidates for

further investigation as outliers. For multiple outliers, it is of interest to note if all such anomalous data cluster at the same setting for any of the various factors. If so, then such settings become candidates for avoidance or inclusion, depending on the nature (bad or good), of the outliers.

Conclusions for the defective springs data

The application of the dex scatter plot to the defective springs data set results in the following conclusions:

1. Most Important Factors:
    1. $X1$ (most important);
    2. $X2$ (of lesser importance);
    3. $X3$ (of least importance).

    that is,
    - factor 1 definitely looks important;
    - factor 2 is a distant second;
    - factor 3 has too much overlap to be important with respect to location, but is flagged for further investigation due to potential differences in variation.
2. Best Settings:

    $(X1,X2,X3) = (+,-,- = (+1,-1,-1)$
3. Outliers: None detected.

5.5.9.3. Dex mean plot

Purpose

The dex (design of experiments) mean plot answers the following two questions:

1. What is the ranked list of factors (not including the interactions)? The ranking is from the most important factor to least important factor.
2. What is the best setting for each of the *k* factors?

In the above two questions, the terms "important" and "best" need clarification and specificity.

A factor can be important if it leads to a significant shift in the location of the response variable as we go from the "-" setting of the factor to the "+" setting of the factor. Alternatively, a factor can be important if it leads to a significant change in variation (spread) as we go from the "-" to the "+" settings. Both definitions are relevant and acceptable. The default definition of "important" in engineering/scientific applications is the former (shift in location). Unless specified to the contrary, when a factor is claimed to be important, the implication is that the factor caused a large location shift in the response.

In this context, a factor setting is best if it results in a typical response that is closest (in location) to the desired project goal (that is, a maximization, minimization, or hitting a target). This desired project goal is an engineering, not a statistical, question, and so the desired optimization goal must be overtly specified by the engineer.

Given the above two definitions of important and best, the dex mean plot is a useful tool for determining the important factors and for determining the best settings.

An alternate name for the dex mean plot is the "main effects plot".

Output

The output from the dex mean plot is:

1. Primary: A ranked list of the factors (not including interactions) from most important to least important.
2. Secondary: The best setting for each of the *k* factors.

Definition

The dex mean plot is formed by:

- Vertical Axis: The mean response for a given setting ("-" or "+") of a factor, for each of the *k* factors.
- Horizontal Axis: The *k* factors and the two settings ("-" and "+") within each factor.

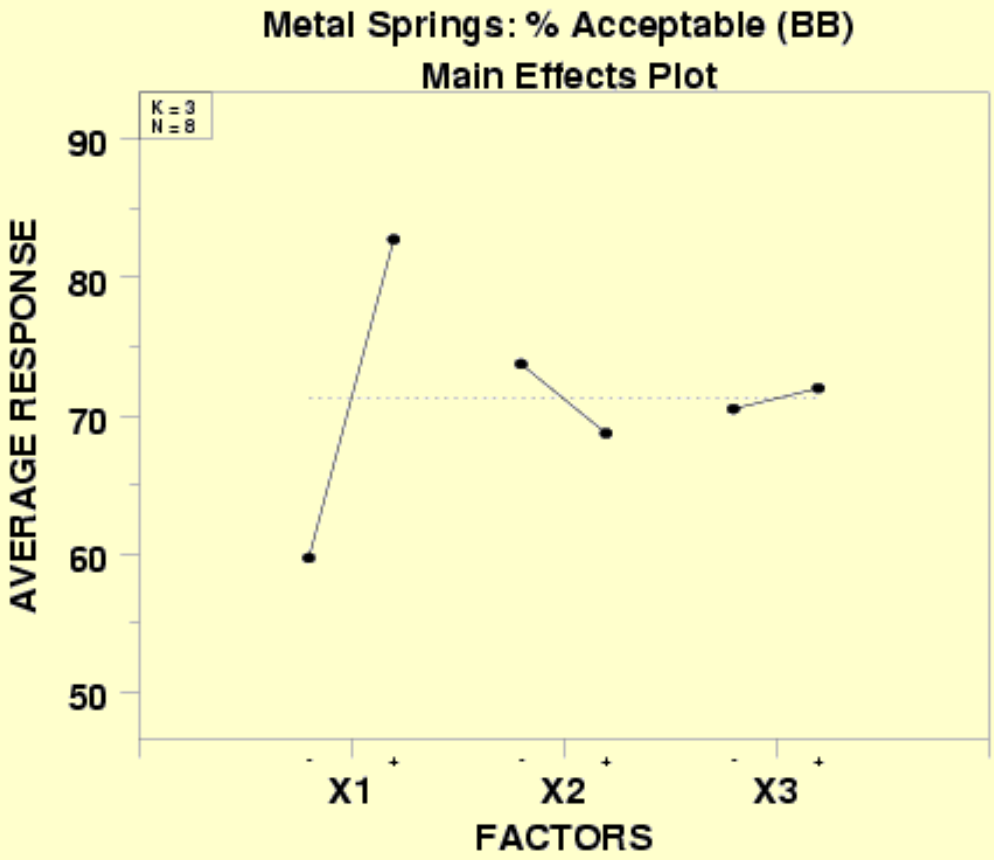
Motivation

If we were interested in assessing the importance of a single factor, and since important, by default, means shift in location, and the average is the simplest location estimator, a reasonable graphics tool to assess a single factor's importance would be a simple mean plot. The vertical axis of such a plot would be the mean response for each setting of the factor and the horizontal axis is the two settings of the factor: "-" and "+" (-1 and +1). A large difference in the two means would imply the factor is important while a small difference would imply the factor is not important.

The dex mean plot is actually a sequence of *k* such plots, with one mean plot for each factor. To assist in comparability and relative importance, all of the mean plots are on the same scale.

Plot for defective springs data

Applying the dex mean plot to the defective springs data yields the following plot.



How to interpret

From the dex mean plot, we look for the following:

1. A ranked list of factors from most important to least important.
2. The best settings for each factor (on average).

Ranked List of Factors--Most Important to Least Important:

For each of the *k* factors, as we go from the "-" setting to the "+" setting for the factor, is there a shift in location of the average response?

If yes, we would like to identify the factor with the biggest shift (the "most important factor"), the next biggest shift (the "second most important factor"), and so on until all factors are accounted for.

Since we are only plotting the means and each factor has identical (-,+) = (-1,+1) coded factor settings, the above simplifies to

1. What factor has the steepest line? This is the most important factor.
2. The next steepest line? This is the second most important factor.
3. Continue for the remaining factors.

This ranking of factors based on local means is the most important step in building the definitive ranked list of factors as required in screening experiments.

Best Settings (on Average):

For each of the *k* factors, which setting (- or +) yields the "best" response?

In order to answer this, the engineer must first define "best". This is done with respect to the overall project goal in conjunction with the specific response variable under study. For some experiments, "best" means we are trying to maximize the response (e.g., maximizing the speed of a chip). For other experiments, "best" means we are trying to minimize the response (e.g., semiconductor chip scrap). For yet other experiments, "best" means we are trying to hit a specific target (e.g., designing a resistor to match a specified resistance). Thus the definition of "best" is a precursor to the determination of best settings.

For example, suppose the analyst is attempting to maximize the response. In that case, the analyst would proceed as follows:

1. For factor 1, what setting (- or +) has the largest average response?
2. For factor 2, what setting (- or +) has the largest average response?
3. Continue for the remaining factors.

The resulting *k*-vector of best settings:

(x1best, x2best, ..., xkbest)

is in general obtained by looking at each factor individually in the dex mean plot and choosing that setting (- or +) that has an average response closest to the desired optimal (maximal, minimal, target) response.

This candidate for best settings is based on the averages. This *k*-vector of best settings should be similar to that obtained from the dex scatter plot, though the dex mean plot is easier to interpret.

Conclusions for the defective springs data

The application of the dex mean plot to the defective springs data set results in the following conclusions:

1. Ranked list of factors (excluding interactions):
    1. $X1$ (most important). Qualitatively, this factor looks definitely important.
    2. $X2$ (of lesser importantance). Qualitatively, this factor is a distant second to $X1$.
    3. $X3$ (unimportant). Qualitatively, this factor appears to be unimportant.
2. Best settings (on average):

    $(X1,X2,X3) = (+,-,+) = (+1,-1,+1)$

5.5.9.4. Interaction effects matrix plot

Purpose

The interaction effects matrix plot is an extension of the dex mean plot to include both main effects and 2-factor interactions (the dex mean plot focuses on main effects only). The interaction effects matrix plot answers the following two questions:

1. What is the ranked list of factors (including 2-factor interactions), ranked from most important to least important; and
2. What is the best setting for each of the *k* factors?

For a *k*-factor experiment, the effect on the response could be due to main effects and various interactions all the way up to *k*-term interactions. As the number of factors, *k*, increases, the total number of interactions increases exponentially. The total number of possible interactions of all orders = 2^k - 1 - *k*. Thus for *k* = 3, the total number of possible interactions = 4, but for *k* = 7 the total number of possible interactions = 120.

In practice, the most important interactions are likely to be 2-factor interactions. The total number of possible 2-factor interactions is

$$\binom{k}{2} = \frac{k!}{2!(k-2)!} = \frac{k(k-1)}{2}$$

Thus for *k* = 3, the number of 2-factor interactions = 3, while for *k* = 7, the number of 2-factor interactions = 21.

It is important to distinguish between the number of interactions that are active in a given experiment versus the number of interactions that the analyst is capable of making definitive conclusions about. The former depends only on the physics and engineering of the problem. The latter depends on the number of factors, *k*, the choice of the *k* factors, the constraints on the number of runs, *n*, and ultimately on the experimental design that the analyst chooses to use. In short, the number of possible interactions is **not** necessarily identical to the number of interactions that we can detect.

Note that

1. with full factorial designs, we can uniquely estimate interactions of all orders;
2. with fractional factorial designs, we can uniquely estimate only some (or at times no) interactions; the more fractionated the design, the fewer interactions that we can estimate.

Output

The output for the interaction effects matrix plot is

1. Primary: Ranked list of the factors (including 2-factor interactions) with the factors are ranked from important to unimportant.
2. Secondary: Best setting for each of the *k* factors.

Definition

The interaction effects matrix plot is an upper right-triangular matrix of mean plots consisting of k main effects plots on the diagonal and $k*(k-1)/2$ 2-factor interaction effects plots on the off-diagonal.

In general, interactions are **not** the same as the usual (multiplicative) cross-products. However, for the special case of 2-level designs coded as (-,+) = (-1 +1), the interactions **are** identical to cross-products. By way of contrast, if the 2-level designs are coded otherwise (e.g., the (1,2) notation espoused by Taguchi and others), then this equivalance is **not** true. Mathematically,

{-1,+1} x {-1,+1} => {-1,+1}

but

{1,2} x {1,2} => {1,2,4}

Thus, coding does make a difference. We recommend the use of the (-,+) coding.

It is remarkable that with the - and + coding, the 2-factor interactions are dealt with, interpreted, and compared in the same way that the k main effects are handled. It is thus natural to include both 2-factor interactions and main effects within the same matrix plot for ease of comparison.

For the off-diagonal terms, the first construction step is to form the horizontal axis values, which will be the derived values (also - and +) of the cross-product. For example, the settings for the $X1*X2$ interaction are derived by simple multiplication from the data as shown below.

$X1$	$X2$	$X1*X2$
-	-	+
+	-	-
-	+	-
+	+	+

Thus $X1$, $X2$, and $X1*X2$ all form a closed (-, +) system. The advantage of the closed system is that graphically interactions can be interpreted in the exact same fashion as the k main effects.

After the entire $X1*X2$ vector of settings has been formed in this way, the vertical axis of the $X1*X2$ interaction plot is formed:

1. the plot point above $X1*X2$ = "-" is simply the mean of all response values for which $X1*X2$ = "-"
2. the plot point above $X1*X2$ = "+" is simply the mean of all response values for which $X1*X2$ = "+".

We form the plots for the remaining 2-factor interactions in a similar fashion.

All the mean plots, for both main effects and 2-factor interactions, have a common scale to facilitate comparisons. Each mean plot has

1. Vertical Axis: The mean response for a given setting (- or +) of a given factor or a given 2-factor interaction.
2. Horizontal Axis: The 2 settings (- and +) within each factor, or within each 2-factor interaction.
3. Legend:
    1. A tag (1, 2, ..., k, 12, 13, etc.), with 1 = $X1$, 2 = $X2$, ..., $k = X_k$, 12 = $X1*X2$, 13 = $X1*X3$, 35 = $X3*X5$, 123 = $X1*X2*X3$, etc.) which identifies the particular mean plot; and
    2. The least squares estimate of the factor (or 2-factor interaction) effect. These effect estimates are large in magnitude for important factors and near-zero in magnitude for unimportant factors.

In a later section, we discuss in detail the models associated with full and fraction factorial 2-level designs. One such model representation is

$$Y = \mu + (1/2)(\beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \ldots)$$

Written in this form (with the leading 0.5), it turns out that the β's are identically the effect due to factor *X*. Further, the least squares estimate turns out to be, due to orthogonality, the simple difference of means at the + setting and the - setting. This is true for the *k* main factors. It is also true for all 2-factor and multi-factor interactions.

Thus, visually, the difference in the mean values on the plot is identically the least squares estimate for the effect. Large differences (steep lines) imply important factors while small differences (flat lines) imply unimportant factors.

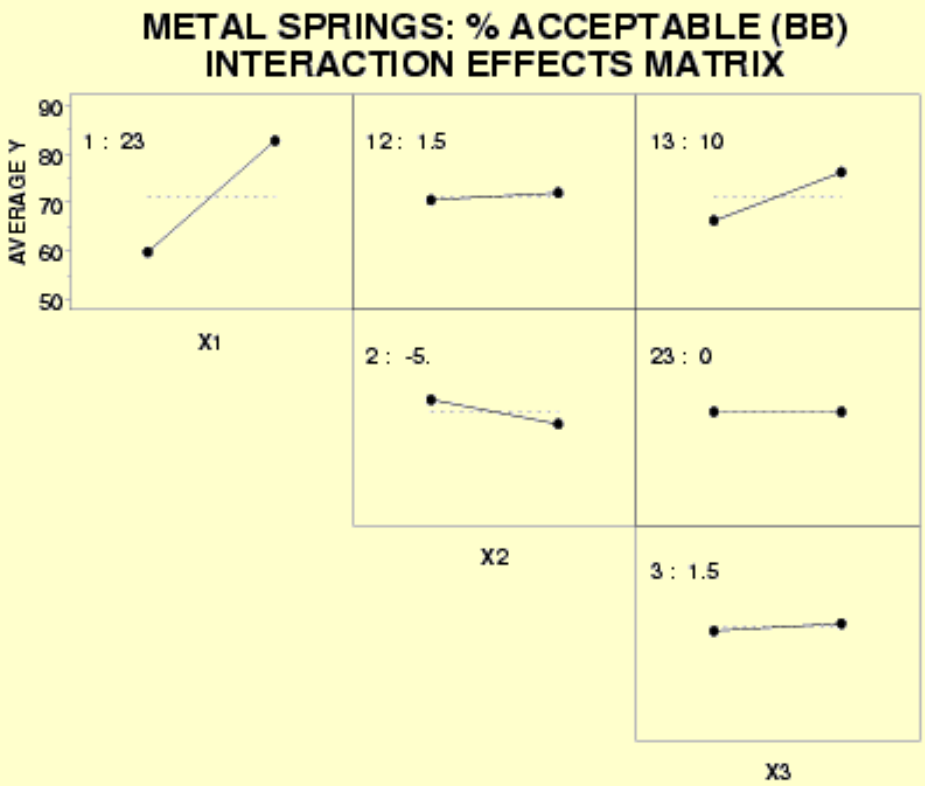
In earlier sections, a somewhat different form of the model is used (without the leading 0.5). In this case, the plotted effects are not necessarily equivalent to the least squares estimates. When using a given software program, you need to be aware what convention for the model the software uses. In either case, the effects matrix plot is still useful. However, the estimates of the coefficients in the model are equal to the effect estimates only if the above convention for the model is used.

Motivation

As discussed in detail above, the next logical step beyond main effects is displaying 2-factor interactions, and this plot matrix provides a convenient graphical tool for examining the relative importance of main effects and 2-factor interactions in concert. To do so, we make use of the striking aspect that in the context of 2-level designs, the 2-factor interactions are identical to cross-products and the 2-factor interaction effects can be interpreted and compared the same way as main effects.

Plot for defective springs data

Constructing the interaction effects matrix plot for the defective springs data set yields the following plot.



How to interpret

From the interaction effects matrix, we can draw three important conclusions:

1. Important Factors (including 2-factor interactions);
2. Best Settings;
3. Confounding Structure (for fractional factorial designs).

We discuss each of these in turn.

1. **Important factors (including 2-factor interactions):**

   Jointly compare the *k* main factors and the *k**(*k*-1)/2 2-factor interactions. For each of these subplots, as we go from the "-" setting to the "+" setting within a subplot, is there a shift in location of the average data (yes/no)? Since all subplots have a common (-1, +1) horizontal axis, questions involving shifts in location translate into questions involving steepness of the mean lines (large shifts imply steep mean lines while no shifts imply flat mean lines).

   1. Identify the factor or 2-factor interaction that has the largest shift (based on averages). This defines the "most important factor". The largest shift is determined by the steepest line.
   2. Identify the factor or 2-factor interaction that has the next largest shift (based on averages). This defines the "second most important factor". This shift is determined by the next steepest line.
   3. Continue for the remaining factors.

   This ranking of factors and 2-factor interactions based on local means is a major step in building the definitive list of ranked factors as required for screening experiments.

2. **Best settings:**

For each factor (of the *k* main factors along the diagonal), which setting (- or +) yields the "best" (highest/lowest) average response?

Note that the experimenter has the ability to change settings for only the *k* main factors, not for any 2-factor interactions. Although a setting of some 2-factor interaction may yield a better average response than the alternative setting for that same 2-factor interaction, the experimenter is unable to set a 2-factor interaction setting in practice. That is to say, there is no "knob" on the machine that controls 2-factor interactions; the "knobs" only control the settings of the *k* main factors.

How then does this matrix of subplots serve as an improvement over the *k* best settings that one would obtain from the dex mean plot? There are two common possibilities:

1. Steep Line:

   For those main factors along the diagonal that have steep lines (that is, are important), choose the best setting directly from the subplot. This will be the same as the best setting derived from the dex mean plot.

2. Flat line:

   For those main factors along the diagonal that have flat lines (that is, are unimportant), the naive conclusion to use either setting, perhaps giving preference to the cheaper setting or the easier-to-implement setting, may be unwittingly incorrect. In such a case, the use of the off-diagonal 2-factor interaction information from the interaction effects matrix is critical for deducing the better setting for this nominally "unimportant" factor.

   To illustrate this, consider the following example:

   - Suppose the factor *X*1 subplot is steep (important) with the best setting for *X*1 at "+".
   - Suppose the factor *X*2 subplot is flat (unimportant) with both settings yielding about the same mean response.

   Then what setting should be used for *X*2? To answer this, consider the following two cases:

   1. Case 1. If the *X*1**X*2 interaction plot happens also to be flat (unimportant), then choose either setting for *X*2 based on cost or ease.
   2. Case 2. On the other hand, if the *X*1**X*2 interaction plot is steep (important), then this dictates a prefered setting for *X*2 **not** based on cost or ease.

   To be specific for case 2, if *X*1**X*2 is important, with *X*1**X*2 = "+" being the better setting, and if *X*1 is important, with *X*1 = "+" being the better setting, then this implies that the best setting for *X*2 must be "+" (to assure that *X*1**X*2 (= +*+) will also be "+"). The reason for this is that since we are already locked into *X*1 = "+", and since *X*1**X*2 = "+" is better, then the only way we can obtain *X*1**X*2 = "+" with *X*1 = "+" is for *X*2 to be "+" (if *X*2 were "-", then *X*1**X*2 with *X*1 = "+" would yield *X*1**X*2 = "-").

   In general, if *X*1 is important, *X*1**X*2 is important, and *X*2 is not important, then there are 4 distinct cases for deciding what the best setting is for *X*2:

*X*1	*X*1**X*2	=> *X*2
+	+	+
+	-	-
-	+	-

- - +

By similar reasoning, examining each factor and pair of factors, we thus arrive at a resulting vector of the *k* best settings:

(x1best, x2best, ..., xkbest)

This average-based *k*-vector should be compared with best settings *k*-vectors obtained from previous steps (in particular, from step 1 in which the best settings were drawn from the best data value).

When the average-based best settings and the data-based best settings agree, we benefit from the increased confidence given our conclusions.

When the average-based best settings and the data-based best settings disagree, then what settings should the analyst finally choose? Note that in general the average-based settings and the data-based settings will invariably be identical for all "important" factors. Factors that do differ are virtually always "unimportant". Given such disagreement, the analyst has three options:

1. Use the average-based settings for minor factors. This has the advantage of a broader (average) base of support.
2. Use the data-based settings for minor factors. This has the advantage of demonstrated local optimality.
3. Use the cheaper or more convenient settings for the local factor. This has the advantage of practicality.

Thus the interaction effects matrix yields important information not only about the ranked list of factors, but also about the best settings for each of the *k* main factors. This matrix of subplots is one of the most important tools for the experimenter in the analysis of 2-level screening designs.

3. **Confounding Structure (for Fractional Factorial Designs)**

When the interaction effects matrix is used to analyze 2-level fractional (as opposed to full) factorial designs, important additional information can be extracted from the matrix regarding confounding structure.

It is well-known that all fractional factorial designs have confounding, a property whereby every estimated main effect is confounded/contaminated/biased by some high-order interactions. The practical effect of this is that the analyst is unsure of how much of the estimated main effect is due to the main factor itself and how much is due to some confounding interaction. Such contamination is the price that is paid by examining *k* factors with a sample size *n* that is less than a full factorial $n = 2^k$ runs.

It is a "fundamental theorem" of the discipline of experimental design that for a given number of factors *k* and a given number of runs *n*, some fractional factorial designs are better than others. "Better" in this case means that the intrinsic confounding that **must** exist in all fractional factorial designs has been minimized by the choice of design. This minimization is done by constructing the design so that the main effect confounding is pushed to as high an order interaction as possible.

The rationale behind this is that in physical science and engineering systems it has been found that the "likelihood" of high-order interactions being significant is small (compared to the likelihood of main effects and 2-factor interactions being significant). Given this, we would prefer that such inescapable main effect confounding be with the highest order interaction possible, and hence the bias to the estimated main effect be as small as possible.

The worst designs are those in which the main effect confounding is with 2-factor interactions. This may be dangerous because in physical/engineering systems, it is quite common for Nature to have some real (and large) 2-factor interactions. In such a case, the 2-factor interaction effect will be inseparably entangled with some estimated main effect, and so the experiment will be flawed in that

1. ambiguous estimated main effects and
2. an ambiguous list of ranked factors

will result.

If the number of factors, k, is large and the number of runs, n, is constrained to be small, then confounding of main effects with 2-factor interactions is unavoidable. For example, if we have $k = 7$ factors and can afford only $n = 8$ runs, then the corresponding 2-level fractional factorial design is a 2^{7-4} which necessarily will have main effects confounded with (3) 2-factor interactions. This cannot be avoided.

On the other hand, situations arise in which 2-factor interaction confounding with main effects results not from constraints on k or n, but on poor design construction. For example, if we have $k = 7$ factors and can afford $n = 16$ runs, a poorly constructed design might have main effects counfounded with 2-factor interactions, but a well-constructed design with the same $k = 7$, $n = 16$ would have main effects confounded with 3-factor interactions but **no** 2-factor interactions. Clearly, this latter design is preferable in terms of minimizing main effect confounding/contamination/bias.

For those cases in which we do have main effects confounded with 2-factor interactions, an important question arises:

> For a particular main effect of interest, how do we know which 2-factor interaction(s) confound/contaminate that main effect?

The usual answer to this question is by means of generator theory, confounding tables, or alias charts. An alternate complementary approach is given by the interaction effects matrix. In particular, if we are examining a 2-level fractional factorial design and

1. if we are not sure that the design has main effects confounded with 2-factor interactions, or
2. if we are sure that we have such 2-factor interaction confounding but are not sure what effects are confounded,

then how can the interaction effects matrix be of assistance? The answer to this question is that the confounding structure can be read **directly** from the interaction effects matrix.

For example, for a 7-factor experiment, if, say, the factor $X3$ is confounded with the 2-factor interaction $X2*X5$, then

1. the appearance of the factor $X3$ subplot and the appearance of the 2-factor interaction $X2*X5$ subplot will necessarily be identical, and
2. the value of the estimated main effect for $X3$ (as given in the legend of the main effect subplot) and the value of the estimated 2-factor interaction effect for $X2*X5$ (as given in the legend of the 2-factor interaction subplot) will also necessarily be identical.

The above conditions are necessary, but not sufficient for the effects to be confounded.

Hence, in the abscence of tabular descriptions (from your statistical software program) of the confounding structure, the interaction effect matrix offers the following graphical alternative for deducing confounding structure in fractional factorial designs:

1. scan the main factors along the diagonal subplots and choose the subset of factors that are "important".
2. For each of the "important" factors, scan all of the 2-factor interactions and compare the main factor subplot and estimated effect with each 2-factor interaction subplot and estimated effect.
3. If there is no match, this implies that the main effect is **not** confounded with any 2-factor interaction.
4. If there is a match, this implies that the main effect **may** be confounded with that 2-factor interaction.
5. If none of the main effects are confounded with any 2-factor interactions, we can have high confidence in the integrity (non-contamination) of our estimated main effects.
6. In practice, for highly-fractionated designs, each main effect may be confounded with several 2-factor interactions. For example, for a 2^{7-4} fractional factorial design, each main effect will be confounded with three 2-factor interactions. These 1 + 3 = 4 identical subplots will be blatantly obvious in the interaction effects matrix.

Finally, what happens in the case in which the design the main effects are **not** confounded with 2-factor interactions (**no** diagonal subplot matches any off-diagonal subplot). In such a case, does the interaction effects matrix offer any useful further insight and information?

The answer to this question is yes because even though such designs have main effects unconfounded with 2-factor interactions, it is fairly common for such designs to have 2-factor interactions confounded with one another, and on occasion it may be of interest to the analyst to understand that confounding. A specific example of such a design is a 2^{4-1} design formed with $X4$ settings = $X1*X2*X3$. In this case, the 2-factor-interaction confounding structure may be deduced by comparing all of the 2-factor interaction subplots (and effect estimates) with one another. Identical subplots and effect estimates hint strongly that the two 2-factor interactions are confounded. As before, such comparisons provide necessary (but not sufficient) conditions for confounding. Most statistical software for analyzing fractional factorial experiments will explicitly list the confounding structure.

Conclusions for the defective springs data

The application of the interaction effects matrix plot to the defective springs data set results in the following conclusions:

1. Ranked list of factors (including 2-factor interactions):
    1. $X1$ (estimated effect = 23.0)
    2. $X1*X3$ (estimated effect = 10.0)
    3. $X2$ (estimated effect = -5.0)
    4. $X3$ (estimated effect = 1.5)
    5. $X1*X2$ (estimated effect = 1.5)
    6. $X2*X3$ (estimated effect = 0.0)

    Factor 1 definitely looks important. The $X1*X3$ interaction looks important. Factor 2 is of lesser importance. All other factors and 2-factor interactions appear to be unimportant.
2. Best Settings (on the average):

    $(X1,X2,X3) = (+,-,+) = (+1,-1,+1)$

5.5.9.5. Block plot

Purpose

The block plot answers the following two general questions:

1. What are the important factors (including interactions)?
2. What are the best settings for these important factors?

The basic (single) block plot is a multifactor EDA technique to determine if a factor is important and to ascertain if that importance is unconditional (robust) over all settings of all other factors in the system. In an experimental design context, the block plot is actually a sequence of block plots with one plot for each of the k factors.

Due to the ability of the block plot to determine whether a factor is important over all settings of all other factors, the block plot is also referred to as a dex robustness plot.

Output

The block plot provides specific information on

1. Important factors (of the k factors and the $\binom{k}{2}$ 2-factor interactions); and
2. Best settings of the important factors.

Definition

The block plot is a series of k basic block plots with each basic block plot for a main effect. Each basic block plot asks the question as to whether that particular factor is important:

1. The first block plot asks the question: "Is factor $X1$ important?
2. The second block plot asks the question: "Is factor $X2$ important?
3. Continue for the remaining factors.

The i-th basic block plot, which targets factor i and asks whether factor X_i is important, is formed by:

- Vertical Axis: Response
- Horizontal Axis: All 2^{k-1} possible combinations of the (k-1) non-target factors (that is, "robustness" factors). For example, for the block plot focusing on factor $X1$ from a 2^3 full factorial experiment, the horizontal axis will consist of all $2^{3-1} = 4$ distinct combinations of factors $X2$ and $X3$. We create this robustness factors axis because we are interested in determining if $X1$ is important robustly. That is, we are interested in whether $X1$ is important not only in a general/summary kind of way, but also whether the importance of X is universally and consistently valid over each of the $2^{3-1} = 4$ combinations of factors $X2$ and $X3$. These 4 combinations are ($X2$,$X3$) = (+,+), (+,-), (-,+), and (-,-). The robustness factors on the horizontal axis change from one block plot to the next. For example, for the k = 3 factor case:
    1. the block plot targeting $X1$ will have robustness factors $X2$ and $X3$;
    2. the block plot targeting $X2$ will have robustness factors $X1$ and $X3$;

3. the block plot targeting $X3$ will have robustness factors $X1$ and $X2$.

- Plot Character: The setting (- or +) for the target factor X_i. Each point in a block plot has an associated setting for the target factor X_i. If X_i = "-", the corresponding plot point will be "-"; if X_i = "+", the corresponding plot point will be "+".

For a particular combination of robustness factor settings (horizontally), there will be two points plotted above it (vertically):

1. one plot point for X_i = "-"; and
2. the other plot point for X_i = "+".

In a block plot, these two plot points are surrounded by a box (a block) to focus the eye on the internal within-block differences as opposed to the distraction of the external block-to-block differences. Internal block differences reflect on the importance of the target factor (as desired). External block-to-block differences reflect on the importance of various robustness factors, which is not of primary interest.

Large within-block differences (that is, tall blocks) indicate a large local effect on the response which, since all robustness factors are fixed for a given block, can only be attributed to the target factor. This identifies an "important" target factor. Small within-block differences (small blocks) indicate that the target factor X_i is unimportant.

For a given block plot, the specific question of interest is thus

> Is the target factor X_i important? That is, as we move within a block from the target factor setting of "-" to the target factor setting of "+", does the response variable value change by a large amount?

The height of the block reflects the "local" (that is, for that particular combination of robustness factor settings) effect on the response due to a change in the target factor settings. The "localized" estimate for the target factor effect for X_i is in fact identical to the difference in the response between the target factor X_i at the "+" setting and at the "-" setting. Each block height of a robustness plot is thus a localized estimate of the target factor effect.

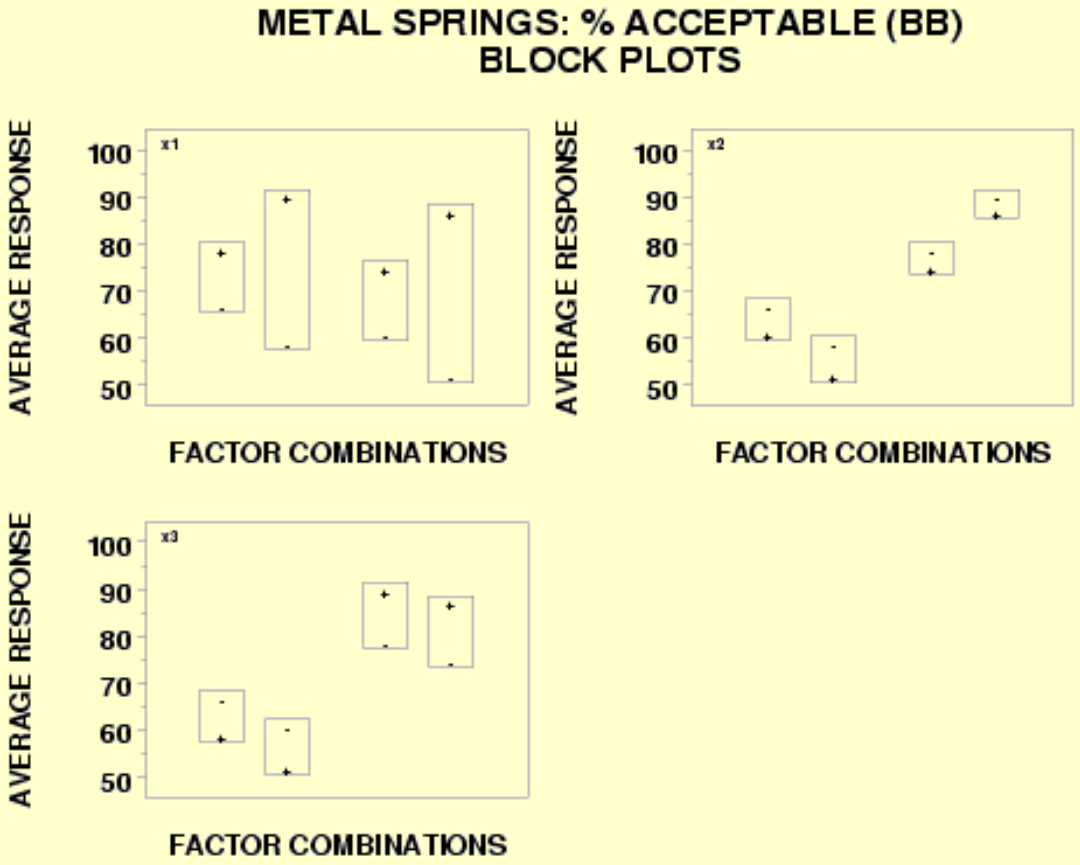
In summary, important factors will have both

1. consistently large block heights; and
2. consistent +/- sign arrangements

where the "consistency" is over all settings of robustness factors. Less important factors will have only one of these two properties. Unimportant factors will have neither property.

Plot for defective springs data

Applying the ordered response plot to the defective springs data set yields the following plot.



How to interpret

From the block plot, we are looking for the following:

1. Important factors (including 2-factor interactions);
2. Best settings for these factors.

We will discuss each of these in turn.

Important factors (including 2-factor interactions):

Look at each of the *k* block plots. Within a given block plot,

> Are the corresponding block heights consistently large as we scan across the within-plot robustness factor settings--yes/no; and are the within-block sign patterns (+ above -, or - above +) consistent across all robustness factors settings--yes/no?

To facilitate intercomparisons, all block plots have the same vertical axis scale. Across such block plots,

1. Which plot has the consistently largest block heights, along with consistent arrangement of within-block +'s and -'s? This defines the "most important factor".
2. Which plot has the consistently next-largest block heights, along with consistent arrangement of within-block +'s and -'s? This defines the "second most important factor".
3. Continue for the remaining factors.

This scanning and comparing of the *k* block plots easily leads to the identification of the most important factors. This identification has the additional virtue over previous steps in that it is robust. For a given important factor, the consistency of block heights and sign arrangement across robustness factors gives additional credence to the robust importance of that factor. The factor is important (the change in the response will be large) irrespective of what settings the robustness

factors have. Having such information is both important and comforting.

Important Special Case; Large but Inconsistent:

What happens if the block heights are large but **not** consistent? Suppose, for example, a 2^3 factorial experiment is being analyzed and the block plot focusing on factor $X1$ is being examined and interpreted so as to address the usual question of whether factor $X1$ is important.

Let us consider in some detail how such a block plot might appear. This $X1$ block plot will have $2^{3-1} = 4$ combinations of the robustness factors $X2$ and $X3$ along the horizontal axis in the following order:

($X2$,$X3$) = (+,+); ($X2$,$X3$) = (+,-); ($X2$,$X3$) = (-,+); ($X2$,$X3$) = (-,-).

If the block heights are consistently large (with "+" above "-" in each block) over the 4 combinations of settings for $X2$ and $X3$, as in

($X2$,$X3$)	block height (= local $X1$ effect)
(+,+)	30
(+,-)	29
(-,+)	29
(-,-)	31

then from binomial considerations there is one chance in $2^{4-1} = 1/8 \approx 12.5\%$ of the the 4 local $X1$ effects having the same sign (i.e., all positive or all negative). The usual statistical cutoff of 5% has not been achieved here, but the 12.5% is suggestive. Further, the consistency of the 4 $X1$ effects (all near 30) is evidence of a robustness of the X effect over the settings of the other two factors. In summary, the above suggests:

1. Factor 1 is probably important (the issue of how large the effect has to be in order to be considered important will be discussed in more detail in a later section); and
2. The estimated factor 1 effect is about 30 units.

On the other hand, suppose the 4 block heights for factor 1 vary in the following cyclic way:

($X2$,$X3$)	block height (= local $X1$ effect)
(+,+)	30
(+,-)	20
(-,+)	30
(-,-)	20

then how is this to be interpreted?

The key here to such interpretation is that the block plot is telling us that the estimated $X1$ effect is in fact at least 20 units, but **not** consistent. The effect is changing, but it is changing in a structured way. The "trick" is to scan the $X2$ and $X3$ settings and deduce what that substructure is. Doing so from the above table, we see that the estimated $X1$ effect is 30

- for point 1 ($X2$,$X3$) = (+,+) and
- for point 3 ($X2$,$X3$) = (-,+)

and then the estimated $X1$ effect drops 10 units to 20

- for point 2 ($X2$,$X3$) = (+,-) and
- for point 4 ($X2$,$X3$) = (-,-)

We thus deduce that the estimated $X1$ effect is

1. 30 whenever $X3$ = "+"

2. 20 whenever $X3$ = "-"

When the factor $X1$ effect is not consistent, but in fact changes depending on the setting of factor $X3$, then definitionally that is said to be an "$X1*X3$ interaction". That is precisely the case here, and so our conclusions would be:

1. factor $X1$ is probably important;
2. the estimated factor $X1$ effect is 25 (= average of 30,20,30,and 20);
3. the $X1*X3$ interaction is probably important;
4. the estimated $X1*X3$ interaction is about 10 (= the change in the factor $X1$ effect as $X3$ changes = 30 - 20 = 10);
5. hence the $X1*X3$ interaction is less important than the $X1$ effect.

Note that we are using the term important in a qualitative sense here. More precise determinations of importance in terms of statistical or engineering significance are discussed in later sections.

The block plot gives us the structure and the detail to allow such conclusions to be drawn and to be understood. It is a valuable adjunct to the previous analysis steps.

Best settings:

After identifying important factors, it is also of use to determine the best settings for these factors. As usual, best settings are determined for main effects only (since main effects are all that the engineer can control). Best settings for interactions are not done because the engineer has no direct way of controlling them.

In the block plot context, this determination of best factor settings is done simply by noting which factor setting (+ or -) within each block is closest to that which the engineer is ultimately trying to achieve. In the defective springs case, since the response variable is % acceptable springs, we are clearly trying to maximize (as opposed to minimize, or hit a target) the response and the ideal optimum point is 100%. Given this, we would look at the block plot of a given important factor and note within each block which factor setting (+ or -) yields a data value closest to 100% and then select that setting as the best for that factor.

From the defective springs block plots, we would thus conclude that

1. the best setting for factor 1 is +;
2. the best setting for factor 2 is -;
3. the best setting for factor 3 cannot be easily determined.

Conclusions for the defective springs data

In summary, applying the block plot to the defective springs data set results in the following conclusions:

1. Unranked list of important factors (including interactions):
   - $X1$ is important;
   - $X2$ is important;
   - $X1*X3$ is important.
2. Best Settings:

   ($X1$,$X2$,$X3$) = (+,-,?) = (+1,-1,?)

5.5.9.6. Dex Youden plot

Purpose

The dex (design of experiments) Youden plot answers the following question:

What are the important factors (including interactions)?

In its original interlab rendition, the Youden plot was a graphical technique developed in the 1960's by Jack Youden of NIST for assessing between-lab biases and within-lab variation problems in the context of interlab experimentation. In particular, it was appropriate for the analysis of round-robin data when exactly two materials, batches, etc. were used in the design.

In a design of experiments context, we borrow this duality emphasis and apply it to 2-level designs. The 2-component emphasis of the Youden plot makes it a natural to be applied to such designs.

Output

The dex Youden plot provides specific information on

1. Ranked list of factors (including interactions); and
2. Separation of factors into two categories: important and unimportant.

The primary output from a dex Youden plot is the ranked list of factors (out of the k factors and interactions). For full factorial designs, interactions include the full complement of interactions at all orders; for fractional factorial designs, interactions include only some, and occasionally none, of the actual interactions. Further, the dex Youden plot yields information identifying which factors/interactions are important and which are unimportant.

Definition

The dex Youden plot consists of the following:

- Vertical Axis: Mean response at the "+" setting for each factor and each interaction. For a given factor or interaction, $n/2$ response values will go into computing the "+" mean.
- Horizontal Axis: Mean response at the "-" setting for each factor and each interaction. For a given factor or interaction, $n/2$ response values will go into computing the "-" mean.
- Plot Character: Factor/interaction identification for which

  1 indicates factor $X1$;
  2 indicates factor $X2$;
  ...
  12 indicates the 2-factor $X1*X2$ interaction
  123 indicates the 3-factor $X1*X2*X3$ interaction
  etc.

In essence, the dex Youden plot is a scatter plot of the "+" average responses versus the "-" average responses. The plot will consist of n - 1 points with one point for each factor and one point for each (available) interaction. Each point on the plot is annotated to identify which factor or interaction is being represented.

Motivation

Definitionally, if a factor is **unimportant, the "+" average will be approximately the same as the "-" average, and if a factor is important, the "+" average will be considerably different from the "-" average. Hence a plot that compares the "+" averages with the "-" averages directly seems potentially informative.**

From the definition above, the dex Youden plot is a scatter plot with the "+" averages on the vertical axis and the "-" averages on the horizontal axis. Thus, unimportant factors will tend to cluster in the middle of the plot and important factors will tend to be far removed from the middle.

Because of an arithmetic identity which requires that the average of any corresponding "+" and "-" means must equal the grand mean, all points on a dex Youden plot will lie on a -45 degree diagonal line. Or to put it another way, for each factor

average (+) + average (-) = constant (with constant = grand mean)

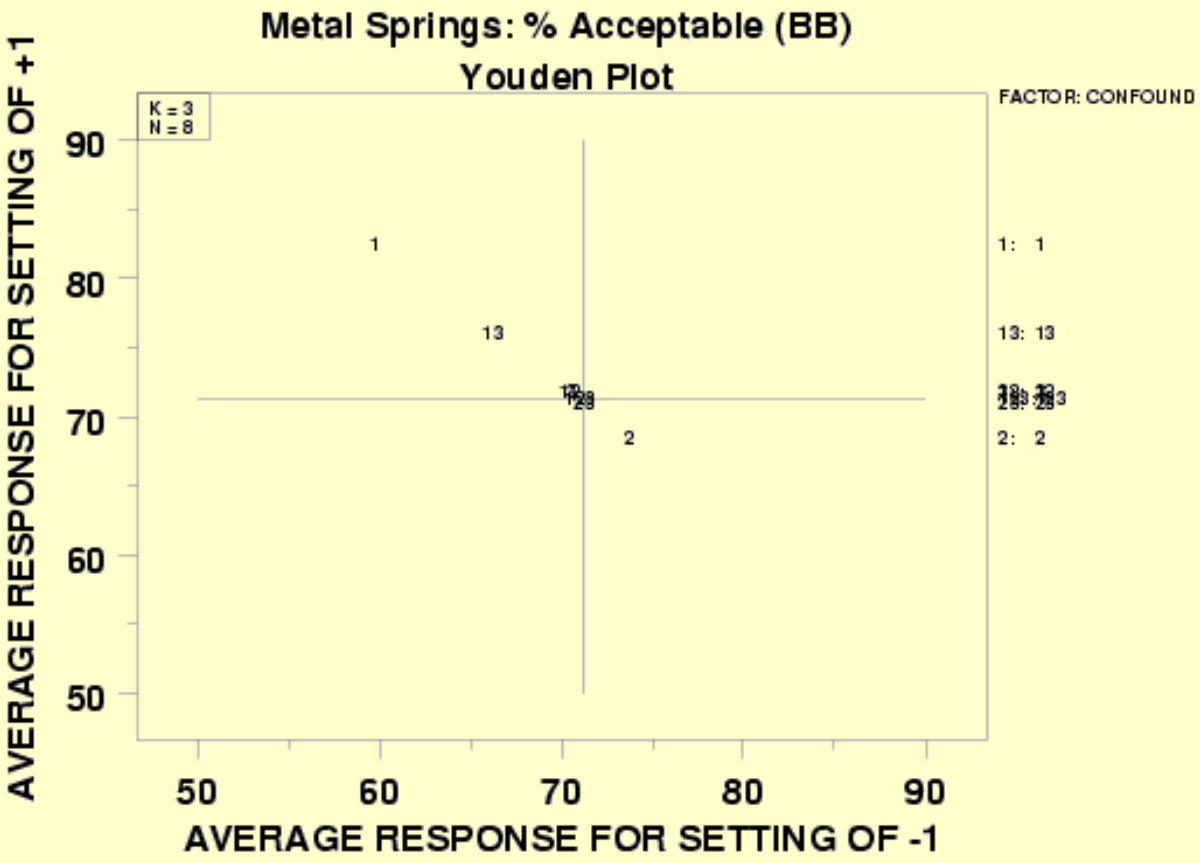
So

average (+) = constant - average (-)

Therefore, the slope of the line is -1 and all points lie on the line. Important factors will plot well-removed from the center because average (+) = average (-) at the center.

Plot for defective springs data

Applying the dex Youden plot for the defective springs data set yields the following plot.



How to interpret

In the dex Youden plot, we look for the following:

1. A ranked list of factors (including interactions). The intersecting dotted lines at the center of the plot are the value of the grand mean on both the vertical and horizontal axes. Scan the points along the negative-slope diagonal line and note as to whether such points are clustered around the grand mean or are displaced up or down the diagonal line.
    1. Which point is farthest away from the center? This defines the "most important" factor.
    2. Which point is next farthest away from the center? This defines the "second most important" factor.
    3. Continue in a similar manner for the remaining points. The points closest to the center define the "least important" factors.
2. Separation of factors into important/unimportant categories. Interpretationally, if a factor is **unimportant**, the "+" average will be about the same as the "-" average, so the plot of "+" vertically and "-" horizontally will be near the grand mean of all n - 1 data points.

    Conversely, if a factor is important, the "+" average will differ greatly from the "-" average, and so the plot of "+" vertically and "-" horizontally will be considerably displaced up into the top left quadrant or down into the bottom right quadrant.

    The separation of factors into important/unimportant categories is thus done by answering the question:

    > Which points visually form a cluster around the center? (these define the "unimportant factors"--all remaining factors are "important").

This ranked list of important factors derived from the dex Youden plot is to be compared with the ranked lists obtained from previous steps. Invariably, there will be a large degree of consistency exhibited across all/most of the techniques.

Conclusions for the defective springs data

The application of the dex Youden plot to the defective springs data set results in the following conclusions:

1. Ranked list of factors (including interactions):
    1. $X1$ (most important)
    2. $X1*X3$ (next most important)
    3. $X2$
    4. other factors are of lesser importance
2. Separation of factors into important/unimportant categories:
    - "Important": $X1$, $X1*X3$, and $X2$
    - "Unimportant": the remainder

5.5.9.7. |Effects| plot

Purpose

The |effects| plot answers the question:

What are the important factors (including interactions)?

Quantitatively, the question as to what is the estimated effect of a given factor or interaction and what is its rank relative to other factors and interactions is answered via the least squares estimation criterion (that is, forming effect estimates that minimize the sum of the squared differences between the raw data and the fitted values from such estimates). Based on such an estimation criterion, one could then construct a tabular list of the factors and interactions ordered by the effect magnitude.

The |effects| plot provides a graphical representation of these ordered estimates, Pareto-style from largest to smallest.

The |effects| plot, as presented here, yields both of the above: the plot itself, and the ranked list table. Further, the plot also presents auxiliary confounding information, which is necessary in forming valid conclusions for fractional factorial designs.

Output

The output of the |effects| plot is:

1. Primary: A ranked list of important effects (and interactions). For full factorial designs, interactions include the full complement of interactions at all orders; for fractional factorial designs, interactions include only some, and occasionally none, of the actual interactions.
2. Secondary: Grouping of factors (and interactions) into two categories: important and unimportant.

Definition

The |effects| plot is formed by:

- Vertical Axis: Ordered (largest to smallest) absolute value of the estimated effects for the main factors and for (available) interactions. For n data points (no replication), typically (n-1) effects will be estimated and the (n-1) |effects| will be plotted.
- Horizontal Axis : Factor/interaction identification:

  1 indicates factor $X1$;
  2 indicates factor $X2$;
  ...
  12 indicates the 2-factor $X1*X2$ interaction
  123 indicates the 3-factor $X1*X2*X3$ interaction,
  etc.
- Far right margin : Factor/interaction identification (built-in redundancy):

  1 indicates factor $X1$;
  2 indicates factor $X2$;
  ...
  12 indicates the 2-factor $X1*X2$ interaction

123 indicates the 3-factor $X1*X2*X3$ interaction,
etc.

If the design is a fractional factorial,the confounding structure is provided for main factors and 2-factor interactions.

- Upper right table: Ranked (largest to smallest by magnitude) list of the least squares estimates for the main effects and for (available) interactions.

  As before, if the design is a fractional factorial, the confounding structure is provided for main factors and 2-factor interactions.

The estimated effects that form the basis for the vertical axis are optimal in the least squares sense. No other estimators exist that will yield a smaller sum of squared deviations between the raw data and the fitted values based on these estimates.

For both the 2^k full factorial designs and 2^{k-p} fractional factorial designs, the form for the least squares estimate of the factor *i* effect, the 2-factor interaction effect, and the multi-factor interaction effect has the following simple form:

factor *i* effect = $\bar{Y}$(+) - $\bar{Y}$(-)
2-factor interaction effect = $\bar{Y}$(+) - $\bar{Y}$(-)
multi-factor interaction effect = $\bar{Y}$(+) - $\bar{Y}$(-)

with $\bar{Y}$(+) denoting the average of all response values for which factor *i* (or the 2-factor or multi-factor interaction) takes on a "+" value, and $\bar{Y}$(-) denoting the average of all response values for which factor *i* (or the 2-factor or multi-factor interaction) takes on a "-" value.

The essence of the above simplification is that the 2-level full and fractional factorial designs are all orthogonal in nature, and so all off-diagonal terms in the least squares $X'X$ matrix vanish.

Motivation

Because of the difference-of-means definition of the least squares estimates, and because of the fact that all factors (and interactions) are standardized by taking on values of -1 and +1 (simplified to - and +), the resulting estimates are all on the same scale. Therefore, comparing and ranking the estimates based on magnitude makes eminently good sense.

Moreover, since the sign of each estimate is completely arbitrary and will reverse depending on how the initial assignments were made (e.g., we could assign "-" to treatment A and "+" to treatment B or just as easily assign "+" to treatment A and "-" to treatment B), forming a ranking based on magnitudes (as opposed to signed effects) is preferred.

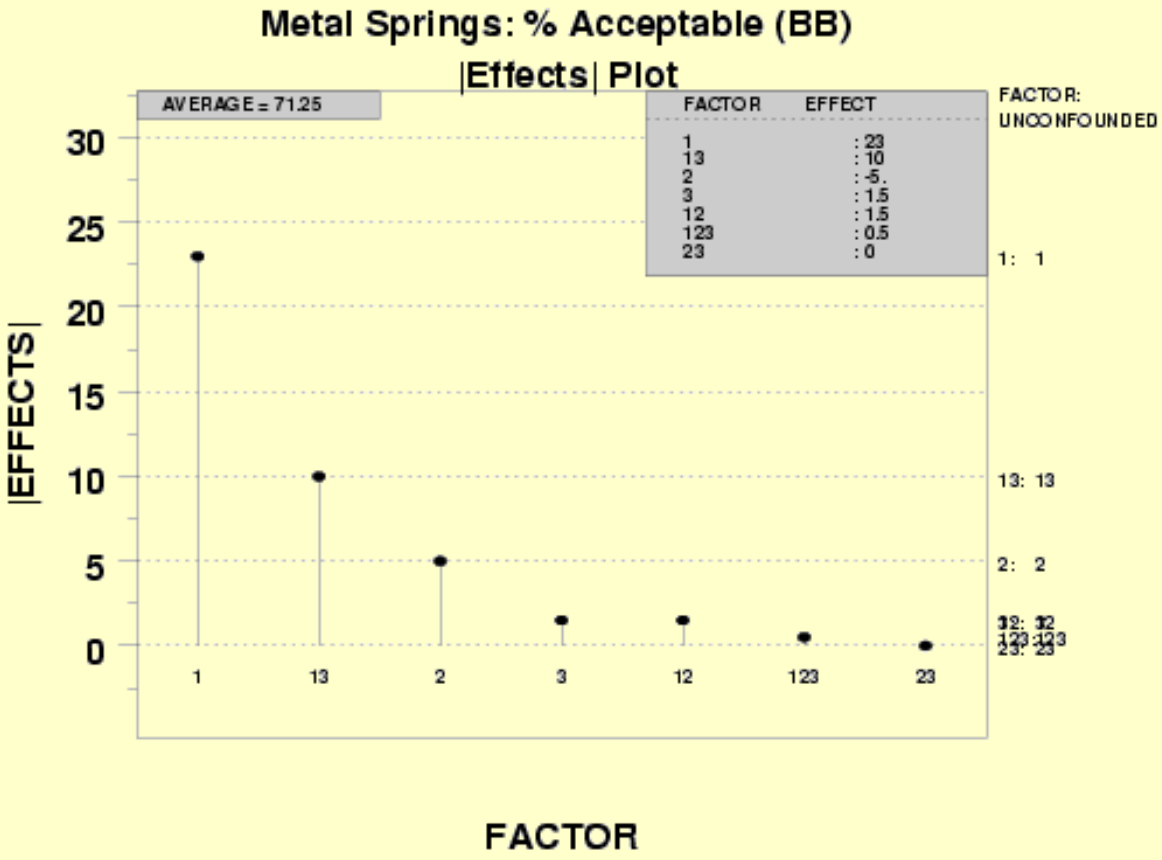
Given that, the ultimate and definitive ranking of factor and interaction effects will be made based on the ranked (magnitude) list of such least squares estimates. Such rankings are given graphically, Pareto-style, within the plot; the rankings are given quantitatively by the tableau in the upper right region of the plot. For the case when we have fractional (versus full) factorial designs, the upper right tableau also gives the confounding structure for whatever design was used.

If a factor is important, the "+" average will be considerably different from the "-" average, and so the absolute value of the difference will be large. Conversely, unimportant factors have small differences in the averages, and so the absolute value will be small.

We choose to form a Pareto chart of such |effects|. In the Pareto chart, the largest effects (= most important factors) will be presented first (to the left) and then progress down to the smallest effects (= least important) factors) to the right.

Plot for defective springs data

Applying the |effects| plot to the defective springs data yields the following plot.



How to interpret

From the |effects| plot, we look for the following:

1. The ranked list of factors (including interactions) is given by the left-to-right order of the spikes. These spikes should be of decreasing height as we move from left to right. Note the factor identifier associated with each of these bars.
2. Identify the important factors. Forming the ranked list of factors is important, but is only half of the analysis. The second part of the analysis is to take the ranking and "draw the (horizontal) line" in the list and on the graph so that factors above the line are deemed "important while factors below the line are deemed unimportant.

   Since factor effects are frequently a continuum ranging from the very large through the moderate and down to the very small, the separation of all such factors into two groups (important and unimportant) may seem arbitrary and severe. However, in practice, from both a research funding and a modeling point of view, such a bifurcation is both common and necessary.

   From an engineering research-funding point of view, one must frequently focus on a subset of factors for future research, attention, and money, and thereby necessarily set aside other factors from any further consideration. From a model-building point of view, a final model either has a term in it or it does not--there is no middle ground. Parsimonious models require in-or-out decisions. It goes without saying that as soon as we have identified the important factors, these are the factors that will comprise our (parsimonious) good model, and those that are declared as unimportant will not be in the model.

Given that, where does such a bifurcation line go?

There are four ways, each discussed in turn, to draw such a line:

1. Statistical significance;
2. Engineering significance;
3. Numerical significance; and
4. Pattern significance.

The ranked list and segregation of factors derived from the |effects| plot are to be compared with the ranked list of factors obtained in previous steps. Invariably, there will be a considerable degree of consistency exhibited across all of the techniques.

Conclusions for the defective springs data

The application of the |effects| plot to the defective springs data set results in the following conclusions:

1. Ranked list of factors (including interactions):
    1. $X1$ (most important)
    2. $X1*X3$ (next most important)
    3. $X2$
    4. other factors are of lesser importance
2. Separation of factors into important/unimportant categories:
    - Important: $X1$, $X1*X3$, and $X2$
    - Unimportant: the remainder

5.5.9.7.1. Statistical significance

Formal statistical methods

Formal statistical methods to answer the question of statistical significance commonly involve the use of

- ANOVA (analysis of variance); and
- *t*-based confidence intervals for the effects.

ANOVA

The virtue of ANOVA is that it is a powerful, flexible tool with many applications. The drawback of ANOVA is that

- it is heavily quantitative and non-intuitive;
- it must have an assumed underlying model; and
- its validity depends on assumptions of a constant error variance and normality of the errors.

t confidence intervals

T confidence intervals for the effects, using the *t*-distribution, are also heavily used for determining factor significance. As part of the *t* approach, one first needs to determine *sd*(*effect*), the standard deviation of an effect. For 2-level full and fractional factorial designs, such a standard deviation is related to σ, the standard deviation of an observation under fixed conditions, via the formula:

$$sd(effect) = \frac{2\sigma}{\sqrt{n}}$$

which in turn leads to forming 95% confidence intervals for an effect via

c * *sd*(*effect*)

for an appropriate multiple *c* (from the *t* distribution). Thus in the context of the |effects| plot, "drawing the line" at *c* * *sd*(*effect*) would serve to separate, as desired, the list of effects into 2 domains:

- significant (that is, important); and
- not significant (that is, unimportant).

Estimating sd(effect)

The key in the above approach is to determine an estimate for *sd*(*effect*). Three statistical approaches are common:

1. Prior knowledge about σ:

   If σ is known, we can compute *sd*(*effect*) from the above expression and make use of a conservative (normal-based) 95% confidence interval by drawing the line at

   $$2sd(effect) = 2(\frac{2\sigma}{\sqrt{n}})$$

   This method is rarely used in practice because σ is rarely known.

2. Replication in the experimental design:

   Replication will allow σ to be estimated from the data without depending on the correctness of a deterministic model. This is a real benefit. On the other hand, the downside of such replication is that it increases the number of runs, time, and expense of the experiment. If replication can be afforded, this method should be used. In such a case, the analyst separates important from unimportant terms by drawing the line at

   $$t * sd(effect) = t * (\frac{2\hat{\sigma}}{\sqrt{n}})$$

   with *t* denoting the 97.5 percent point from the appropriate Student's-*t* distribution.

3. Assume 3-factor interactions and higher are zero:

   This approach "assumes away" all 3-factor interactions and higher and uses the data pertaining to these interactions to estimate σ. Specifically,

   $$\hat{\sigma} = \sqrt{\frac{SSQ}{h}}$$

   with *h* denoting the number of 3-factor interactions and higher, and SSQ is the sum of squares for these higher-order effects. The analyst separates important from unimportant effects by drawing the line at

   $$t * sd(effect) = t * (\frac{2\hat{\sigma}}{\sqrt{n}})$$

   with *t* denoting the 97.5 percent point from the appropriate

(with *h* degrees of freedom) Student's-*t* distribution.

This method warrants caution:

- it involves an untestable assumption (that such interactions = 0);
- it can result in an estimate for *sd*(*effect*) based on few terms (even a single term); and
- it is virtually unusable for highly-fractionated designs (since high-order interactions are not directly estimable).

Non-statistical considerations

The above statistical methods can and should be used. Additionally, the non-statistical considerations discussed in the next few sections are frequently insightful in practice and have their place in the EDA approach as advocated here.

5.5.9.7.2. Engineering significance

Engineering cutoff

Draw the horizontal line on the chart at that value which you as an engineer have declared beforehand as the engineering cutoff. Any effect larger than this cutoff will be considered as significant from an engineering point of view.

Specifying a cutoff value requires non-statistical thinking, but is frequently useful

This approach requires preliminary, data-free thinking on the part of the analyst as to how big (= what number?) an effect (any effect) must be before the analyst would "care" as an engineer/scientist? In other words, in the units of the response variable, how much would the response variable have to change consistently before the analyst would say "that's a big enough change for me from an engineering point of view"? An engineering number, a cutoff value, is needed here. This value is non-statistical; thie value must emanate from the engineer's head.

If upon reflection the analyst does **not** have such a value in mind, this "engineering significance" approach would be set aside. From experience, it has been found that the engineering soul-searching that goes into evoking such a cutoff value is frequently useful and should be part of the decision process, independent of statistical considerations, of separating the effects into important/unimportant categories.

A rough engineering cutoff

In the absence of a known engineering cutoff, a rough cutoff value is commonly 5% or 10% of the **average** (or current) production response for the system. Thus, if a chemical reaction production process is yielding a reaction rate of about 70, then 5% of 70 = 3. The engineer may declare any future effect that causes an average change of 3 or more units in the response (that is, any estimated effect whose magnitude exceeds 3) to be "engineering significant". In the context of the |effects| plot, the engineer would draw the line at a height of 3 on the plot, and all effects that are above the line are delared as significant and all below the line are declared not significant.

5.5.9.7.3. Numerical significance

10% of the largest effect	Note the height of the largest bar (= the magnitude of the largest effect). Declare as "significant" any effect that exceeds 10% of the **largest** effect. The 10% is arbitrary and has no statistical (or engineering) basis, but it does have a "numeric" basis in that it results in keeping the largest effect and any effects that are within 90% of the largest effect.
Apply with caution	As with any rule-of-thumb, some caution should be used in applying this critierion. Specifically, if the largest effect is in fact not very large, this rule-of-thumb may not be useful.

5.5.9.7.4. Pattern significance

Look for L-shaped pattern

The |effects| plot has a characteristic horizontally-elongated *L*-shaped pattern. The vertical arm of the *L* consists of important factors. The horizontal arm is comprised of unimportant factors. If a factor is important, the bar height will be large and succeeding bar heights may drop off considerably (perhaps by 50%)--such factors make up the left arm of the *L*. On the other hand, if a factor is not important, its bar height will tend to be small and near-zero--such factors make up the bottom arm of the *L*. It is of interest to note where the kink is in the *L*. Factors to the left of that kink are arguably declared important while factors at the kink point and to the right of it are declared unimportant.

Factor labels

As a consequence of this "kinking", note the labels on the far right margin of the plot. Factors to the left and above the kink point tend to have far-right labels distinct and isolated. Factors at, to the right, and below the kink point tend to have far right labels that are overstruck and hard to read. A (rough) rule-of-thumb would then be to declare as important those factors/interactions whose far-right labels are easy to distinguish, and to declare as unimportant those factors/interactions whose far-right labels are overwritten and hard to distinguish.

5.5.9.8. Half-normal probability plot

Purpose

The half-normal probability plot answers the question:

What are the important factors (including interactions)?

Quantitatively, the estimated effect of a given main effect or interaction and its rank relative to other main effects and interactions is given via least squares estimation (that is, forming effect estimates that minimize the sum of the squared differences between raw data and the fitted values from such estimates). Having such estimates in hand, one could then construct a list of the main effects and interactions ordered by the effect magnitude.

The half-normal probability plot is a graphical tool that uses these ordered estimated effects to help assess which factors are important and which are unimportant.

A half-normal distribution is the distribution of the $|X|$ with X having a normal distribution.

Output

The outputs from the half-normal probablity plot are

1. Primary: Grouping of factors and interactions into two categories: important and unimportant. For full factorial designs, interactions include the full complement of interactions of all orders; for fractional factorial designs, interactions include only some, and occasionally none, of the actual interactions (when they aren't estimable).
2. Secondary: Ranked list of factors and interactions from most important down to least important.

Definition

A half-normal probability plot is formed by

- Vertical Axis: Ordered (largest to smallest) absolute value of the estimated effects for the main factors and available interactions. If n data points (no replication) have been collected, then typically (n-1) effects will be estimated and the (n-1) |effects| will be plotted.
- Horizontal Axis: (n-1) theoretical order statistic medians from a half-normal distribution. These (n-1) values are not data-dependent. They depend only on the half-normal distribution and the number of items plotted (= n-1). The theoretical medians represent an "ideal" typical ordered data set that would have been obtained from a random drawing of (n-1) samples from a half-normal distribution.
- Far right margin : Factor/interaction identification:

  1 indicates factor $X1$;
  2 indicates factor $X2$;
  ...
  12 indicates the 2-factor $X1*X2$ interaction
  123 indicates the 3-factor $X1*X2*X3$ interaction,
  etc.

If the design is a fractional factorial, the confounding structure is provided for main effects and 2-factor interactions.

Motivation

To provide a rationale for the half-normal probability plot, we first dicuss the motivation for the normal probability plot (which also finds frequent use in these 2-level designs).

The basis for the normal probability plot is the mathematical form for each (and all) of the estimated effects. As discussed for the |effects| plot, the estimated effects are the optimal least squares estimates. Because of the orthogonality of the 2^k full factorial and the 2^{k-p} fractional factorial designs, all least squares estimators for main effects and interactions simplify to the form:

estimated effect = $\bar{Y}(+) - \bar{Y}(-)$

with $\bar{Y}(+)$ the average of all response values for which the factor or interaction takes on a "+" value, and where $\bar{Y}(-)$ is the average of all response values for which the factor or interaction takes on a "-" value.

Under rather general conditions, the Central Limit Thereom allows that the difference-of-sums form for the estimated effects tends to follow a normal distribution (for a large enough sample size *n*) a normal distribution.

The question arises as to what normal distribution; that is, a normal distribution with what mean and what standard deviation? Since all estimators have an identical form (a difference of averages), the standard deviations, though unknown, will in fact be the same under the assumption of constant σ. This is good in that it simplifies the normality analysis.

As for the means, however, there will be differences from one effect to the next, and these differences depend on whether a factor is unimportant or important. **Unimportant** factors are those that have near-zero effects and **important** factors are those whose effects are considerably removed from zero. Thus, **unimportant effects tend to have a normal distribution centered near zero while important effects tend to have a normal distribution centered at their respective true large (but unknown) effect values.**

In the simplest experimental case, if the experiment were such that no factors were important (that is, all effects were near zero), the (*n*-1) estimated effects would behave like random drawings from a normal distribution centered at zero. We can test for such normality (and hence test for a null-effect experiment) by using the normal probability plot. Normal probability plots are easy to interpret. In simplest terms:

if linear, then normal

If the normal probability plot of the (*n*-1) estimated effects is linear, this implies that all of the true (unknown) effects are zero or near-zero. That is, no factor is important.

On the other hand, if the truth behind the experiment is that there is exactly one factor that was important (that is, significantly non-zero), and all remaining factors are unimportant (that is, near-zero), then the normal probability plot of all (*n*-1) effects is near-linear for the (*n*-2) unimportant factors and the remaining single important factor would stand well off the line.

Similarly, if the experiment were such that some subset of factors were important and all remaining factors were unimportant, then the normal probability plot of all (*n*-1) effects would be near-linear for all unimportant factors with the remaining important factors all well off the line.

In real life, with the number of important factors unknown, this suggests that one could

form a normal probability plot of the (*n*-1) estimated effects and draw a line through those (unimportant) effects in the vicinity of zero. This identifies and extracts all remaining effects off the line and declares them as important.

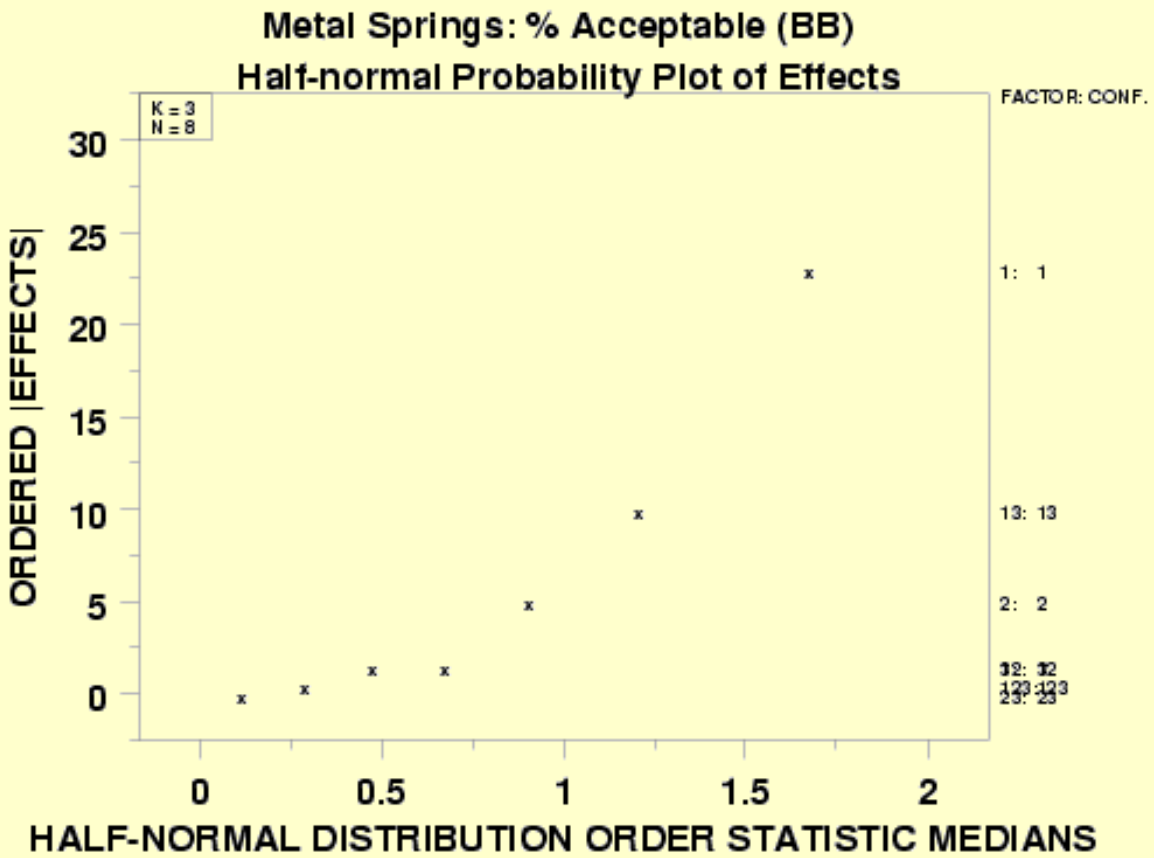
The above rationale and methodology works well in practice, with the net effect that the normal probability plot of the effects is an important, commonly used and successfully employed tool for identifying important factors in 2-level full and factorial experiments. Following the lead of Cuthbert Daniel (1976), we augment the methodology and arrive at a further improvement. Specifically, the sign of each estimate is completely arbitrary and will reverse depending on how the initial assignments were made (e.g., we could assign "-" to treatment A and "+" to treatment B or just as easily assign "+" to treatment A and "-" to treatment B).

This arbitrariness is addressed by dealing with the effect magnitudes rather than the signed effects. If the signed effects follow a normal distribution, the absolute values of the effects follow a half-normal distribution.

In this new context, one tests for important versus unimportant factors by generating a half-normal probability plot of the absolute value of the effects. As before, linearity implies half-normality, which in turn implies all factors are unimportant. More typically, however, the half-normal probability plot will be only partially linear. Unimportant (that is, near-zero) effects manifest themselves as being near zero and on a line while important (that is, large) effects manifest themselves by being off the line and well-displaced from zero.

Plot for defective springs data

The half-normal probability plot of the effects for the defectice springs data set is as follows.



How to interpret

From the half-normal probability plot, we look for the following:

1. Identifying Important Factors:

   Determining the subset of important factors is the most important task of the half-normal probability plot of |effects|. As discussed above, the estimated |effect| of an **un**important factor will typically be on or close to a near-zero line, while the estimated |effect| of an important factor will typically be displaced well off the line.

   The separation of factors into important/unimportant categories is thus done by answering the question:

   > Which points on the half-normal probability plot of |effects| are large and well-off the linear collection of points drawn in the vicinity of the origin?

   This line of unimportant factors typically encompasses the majority of the points on the plot. The procedure consists, therefore, of the following:

   1. identifying this line of near-zero (unimportant) factors; then
   2. declaring the remaining off-line factors as important.

   Note that the half-normal probability plot of |effects| and the |effects| plot have the same vertical axis; namely, the ordered |effects|, so the following discussion about right-margin factor identifiers is relevant to both plots. As a consequence of the natural on-line/off-line segregation of the |effects| in half-normal probability plots, factors off-line tend to have far-right labels that are distinct and isolated while factors near the line tend to have far-right labels that are overstruck and hard to read. The rough rule-of-thumb would then be to declare as important those factors/interactions whose far-right labels are easy to distinguish and to declare as unimportant those factors/interactions whose far-right labels are overwritten and hard to distinguish.

2. Ranked List of Factors (including interactions):

   This is a minor objective of the half-normal probability plot (it is better done via the |effects| plot). To determine the ranked list of factors from a half-normal probability plot, simply scan the vertical axis |effects|

   1. Which |effect| is largest? Note the factor identifier associated with this largest |effect| (this is the "most important factor").
   2. Which |effect| is next in size? Note the factor identifier associated with this next largest |effect| (this is the "second most important factor").
   3. Continue for the remaining factors. In practice, the bottom end of the ranked list (the unimportant factors) will be hard to extract because of overstriking, but the top end of the ranked list (the important factors) will be easy to determine.

In summary, it should be noted that since the signs of the estimated effects are arbitrary, we recommend the use of the half-normal probability plot of |effects| technique over the normal probability plot of the |effects|. These probability plots are among the most commonly-employed EDA procedure for identification of important factors in 2-level full and factorial designs. The half-normal probability plot enjoys widespread usage across both "classical" and Taguchi camps. It deservedly plays an important role in our recommended 10-step graphical procedure for the analysis of 2-level designed experiments.

Conclusions for the defective springs data

The application of the half-normal probability plot to the defective springs data set results in the following conclusions:

1. Ranked list of factors (including interactions):
    1. $X1$ (most important)
    2. $X1*X3$ (next most important)
    3. $X2$
    4. other factors are of lesser importance
2. Separation of factors into important/unimportant categories:

    Important: $X1$, $X1*X3$, and $X2$
    Unimportant: the remainder

5.5.9.9. Cumulative residual standard deviation plot

Purpose

The cumulative residual sd (standard deviation) plot answers the question:

What is a good model for the data?

The prior 8 steps in this analysis sequence addressed the two important goals:

1. Factors: determining the most important factors that affect the response, and
2. Settings: determining the best settings for these factors.

In addition to the above, a third goal is of interest:

3. Model: determining a model (that is, a prediction equation) that functionally relates the observed response Y with the various main effects and interactions.

Such a function makes particular sense when all of the individual factors are continuous and ordinal (such as temperature, pressure, humidity, concentration, etc.) as opposed to any of the factors being discrete and non-ordinal (such as plant, operator, catalyst, supplier).

In the continuous-factor case, the analyst could use such a function for the following purposes.

1. Reproduction/Smoothing: predict the response at the observed design points.
2. Interpolation: predict what the response would be at (unobserved) regions between the design points.
3. Extrapolation: predict what the response would be at (unobserved) regions beyond the design points.

For the discrete-factor case, the methods developed below to arrive at such a function still apply, and so the resulting model may be used for reproduction. However, the interpolation and extrapolation aspects do not apply.

In modeling, we seek a function f in the k factors $X_1, X_2, ..., X_k$ such that the predicted values

$$\hat{Y} = f(X_1, X_2, ..., X_k)$$

are "close" to the observed raw data values Y. To this end, two tasks exist:

1. Determine a good functional form f;
2. Determine good estimates for the coefficients in that function f.

For example, if we had two factors X_1 and X_2, our goal would be to

1. determine some function $Y = f(X_1,X_2)$; and
2. estimate the parameters in f

such that the resulting model would yield predicted values $\hat{Y}$ that are as close as possible to the observed response values Y. If the form f has been wisely chosen, a good model will result and that model will have the characteristic that the differences ("residuals" = $Y - \hat{Y}$) will be uniformly near zero. On the other hand, a poor model (from a poor choice of the form f) will have the characteristic that some or all of the residuals will be "large".

For a given model, a statistic that summarizes the quality of the fit via the typical size of the *n* residuals is the residual standard deviation:

$$s_{res} = \sqrt{\frac{\sum_{i=1}^{n} r_i^2}{n-p}}$$

with *p* denoting the number of terms in the model (including the constant term) and *r* denoting the *i*th residual. We are also assuming that the mean of the residuals is zero, which will be the case for models with a constant term that are fit using least squares.

If we have a good-fitting model, s_{res} will be small. If we have a poor-fitting model, s_{res} will be large.

For a given data set, each proposed model has its own quality of fit, and hence its own residual standard deviation. Clearly, the residual standard deviation is more of a model-descriptor than a data-descriptor. Whereas "nature" creates the data, the analyst creates the models. Theoretically, for the same data set, it is possible for the analyst to propose an indefinitely large number of models.

In practice, however, an analyst usually forwards only a small, finite number of plausible models for consideration. Each model will have its own residual standard deviation. The cumulative residual standard deviation plot is simply a graphical representation of this collection of residual standard deviations for various models. The plot is beneficial in that

1. good models are distinguished from bad models;
2. simple good models are distinguished from complicated good models.

In summary, then, the cumulative residual standard deviation plot is a graphical tool to help assess

1. which models are poor (least desirable); and
2. which models are good but complex (more desirable); and
3. which models are good and simple (most desirable).

Output

The outputs from the cumulative residual standard deviation plot are

1. Primary: A good-fitting prediction equation consisting of an additive constant plus the most important main effects and interactions.
2. Secondary: The residual standard deviation for this good-fitting model.

Definition

A cumulative residual sd plot is formed by

1. Vertical Axis: Ordered (largest to smallest) residual standard deviations of a sequence of progressively more complicated fitted models.
2. Horizontal Axis: Factor/interaction identification of the last term included into the linear model:

   1 indicates factor *X*1;
   2 indicates factor *X*2;
   ...
   12 indicates the 2-factor *X*1**X*2 interaction
   123 indicates the 3-factor *X*1**X*2**X*3 interaction
   etc.
3. Far right margin: Factor/interaction identification (built-in redundancy):

   1 indicates factor *X*1;

2 indicates factor *X*2;
...
12 indicates the 2-factor *X*1**X*2 interaction
123 indicates the 3-factor *X*1**X*2**X*3 interaction
etc.

If the design is a fractional factorial, the confounding structure is provided for main effects and 2-factor interactions.

The cumulative residual standard deviations plot is thus a Pareto-style, largest to smallest, graphical summary of residual standard deviations for a selected series of progressively more complicated linear models.

The plot shows, from left to right, a model with only a constant and the model then augmented by including, one at a time, remaining factors and interactions. Each factor and interaction is incorporated into the model in an additive (rather than in a multiplicative or logarithmic or power, etc. fashion). At any stage, the ordering of the next term to be added to the model is such that it will result in the maximal decrease in the resulting residual standard deviation.

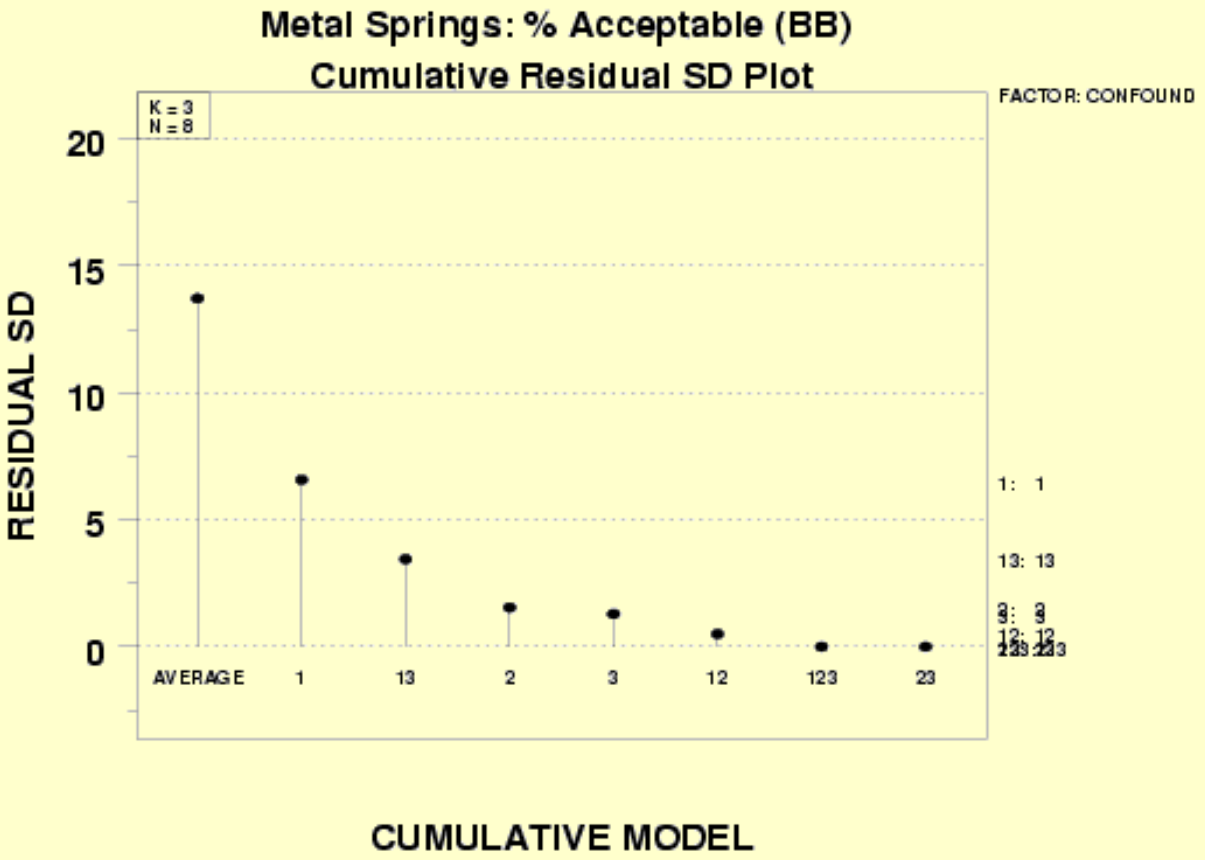
Motivation

This section addresses the following questions:

1. What is a model?
2. How do we select a goodness-of-fit metric for a model?
3. How do we construct a good model?
4. How do we know when to stop adding terms?
5. What is the final form for the model?
6. Why is the 1/2 in the model?
7. What are the advantages of the linear model?
8. How do we use the model to generate predicted values?
9. How do we use the model beyond the data domain?
10. What is the best confirmation point for interpolation?
11. How do we use the model for interpolation?
12. How do we use the model for extrapolation?

Plot for defective springs data

Applying the cumulative residual standard deviation plot to the defective springs data set yields the following plot.



How to interpret

As discussed in detail under question 4 in the Motivation section, the cumulative residual standard deviation "curve" will characteristically decrease left to right as we add more terms to the model. The incremental improvement (decrease) tends to be large at the beginning when important factors are being added, but then the decrease tends to be marginal at the end as unimportant factors are being added.

Including all terms would yield a perfect fit (residual standard deviation = 0) but would also result in an unwieldy model. Including only the first term (the average) would yield a simple model (only one term!) but typically will fit poorly. Although a formal quantitative stopping rule can be developed based on statistical theory, a less-rigorous (but good) alternative stopping rule that is graphical, easy to use, and highly effective in practice is as follows:

> Keep adding terms to the model until the curve's "elbow" is encountered. The "elbow point" is that value in which there is a consistent, noticeably shallower slope (decrease) in the curve. Include all terms up to (and including) the elbow point (after all, each of these included terms decreased the residual standard deviation by a large amount). Exclude any terms **after** the elbow point since all such successive terms decreased the residual standard deviation so slowly that the terms were "not worth the complication of keeping".

From the residual standard deviation plot for the defective springs data, we note the following:

1. The residual standard deviation (rsd) for the "baseline" model

$$\hat{Y} = \bar{Y} = 71.25$$

is $s_{res} = 13.7$.

2. As we add the next term, *X*1, the rsd drops nearly 7 units (from 13.7 to 6.6).
3. If we add the term *X*1**X*3, the rsd drops another 3 units (from 6.6 to 3.4).
4. If we add the term *X*2, the rsd drops another 2 units (from 3.4 to 1.5).
5. When the term *X*3 is added, the reduction in the rsd (from about 1.5 to 1.3) is negligible.
6. Thereafter to the end, the total reduction in the rsd is from only 1.3 to 0.

In step 5, note that when we have effects of equal magnitude (the *X*3 effect is equal to the *X*1**X*2 interaction effect), we prefer including a main effect before an interaction effect and a lower-order interaction effect before a higher-order interaction effect.

In this case, the "kink" in the residual standard deviation curve is at the *X*2 term. Prior to that, all added terms (including *X*2) reduced the rsd by a large amount (7, then 3, then 2). After the addition of *X*2, the reduction in the rsd was small (all less than 1): .2, then .8, then .5, then 0.

The final recommended model in this case thus involves $p = 4$ terms:

1. the average (= 71.25)
2. factor *X*1
3. the *X*1**X*3
4. factor *X*2

The fitted model thus takes on the form

$$\hat{Y} = average + 0.5 * (B_1 * X1 + B_{13} * X1 * X3 - B_2 * X2)$$

The motivation for using the 0.5 term was given in an earlier section.

The least squares estimates for the coefficients in this model are

average = 71.25
B_1 = 23
B_{13} = 10
B_2 = -5

The B_1 = 23, B_{13} = 10, and B_2 = -5 least squares values are, of course, identical to the estimated effects E_1 = 23, E_{13} = 10, and E_2 = -5 (= $\hat{Y}$(+1) - $\hat{Y}$(-1)) values as previously derived in step 7 of this recommended 10-step DEX analysis procedure.

The final fitted model is thus

$$\hat{Y} = 71.25 + 0.5 * (23 * X1 + 10 * (X1 * X3) - 5 * X2)$$

Applying this prediction equation to the 8 design points yields: predicted values $\hat{Y}$ that are close to the data *Y*, and residuals (*Res* = *Y* - $\hat{Y}$) that are close to zero:

X1	X2	X3	Y	$\hat{Y}$	Res
-	-	-	67	67.25	-0.25
+	-	-	79	80.25	-1.25
-	+	-	61	62.25	-1.25
+	+	-	75	75.25	-0.25
-	-	+	59	57.25	+1.75
+	-	+	90	90.25	-0.25
-	+	+	52	52.25	-0.25
+	+	+	87	85.25	+1.75

Computing the residual standard deviation:

$$s_{res} = \sqrt{\frac{\sum_{i=1}^{n} r_i^2}{n-p}}$$

with n = number of data points = 8, and p = 4 = number of estimated coefficients (including the average) yields

s_{res} = 1.54 (= 1.5 if rounded to 1 decimal place)

This detailed *res* = 1.54 calculation brings us full circle for 1.54 is the value given above the *X*3 term on the cumulative residual standard deviation plot.

Conclusions for the defective springs data

The application of the Cumulative Residual Standard Deviation Plot to the defective springs data set results in the following conclusions:

1. Good-fitting Parsimonious (constant + 3 terms) Model:

$$\hat{Y} = 71.25 + 0.5 * (23 * X1 + 10 * (X1 * X3) - 5 * X2)$$

2. Residual Standard Deviation for this Model (as a measure of the goodness-of-fit for the model):

s_{res} = 1.54

5.5.9.9.1. Motivation: What is a Model?

Mathematical models: functional form and coefficients

A model is a mathematical function that relates the response Y to the factors X_1 to X_k. A model has a

1. functional form; and
2. coefficients.

An excellent and easy-to-use functional form that we find particularly useful is a linear combination of the main effects and the interactions (the selected model is a subset of the full model and almost always a proper subset). The coefficients in this linear model are easy to obtain via application of the least squares estimation criterion (regression). A given functional form with estimated coefficients is referred to as a "fitted model" or a "prediction equation".

Predicted values and residuals

For given settings of the factors X_1 to X_k, a fitted model will yield predicted values. For each (and every) setting of the X_i's, a "perfect-fit" model is one in which the predicted values are identical to the observed responses Y at these X_i's. In other words, a perfect-fit model would yield a vector of predicted values identical to the observed vector of response values. For these same X_i's, a "good-fitting" model is one that yields predicted values "acceptably near", but not necessarily identical to, the observed responses Y.

The residuals (= deviations = error) of a model are the vector of differences ($Y - \hat{Y}$) between the responses and the predicted values from the model. For a perfect-fit model, the vector of residuals would be all zeros. For a good-fitting model, the vector of residuals will be acceptably (from an engineering point of view) close to zero.

5.5.9.9.2. Motivation: How do we Construct a Goodness-of-fit Metric for a Model?

Motivation

This question deals with the issue of how to construct a metric, a statistic, that may be used to ascertain the quality of the fitted model. The statistic should be such that for one range of values, the implication is that the model is good, whereas for another range of values, the implication is that the model gives a poor fit.

Sum of absolute residuals

Since a model's adequacy is inversely related to the size of its residuals, one obvious statistic is the sum of the absolute residuals.

$$AR = \sum_{i=1}^{n} |r_i|$$

Clearly, for a fixed *n*,the smaller this sum is, the smaller are the residuals, which implies the closer the predicted values are to the raw data *Y*, and hence the better the fitted model. The primary disadvantage of this statistic is that it may grow larger simply as the sample size *n* grows larger.

Average absolute residual

A better metric that does not change (much) with increasing sample size is the average absolute residual:

$$AAR = \frac{\sum_{i=1}^{n} |r_i|}{n}$$

with *n* denoting the number of response values. Again, small values for this statistic imply better-fitting models.

Square root of the average squared residual

An alternative, but similar, metric that has better statistical properties is the square root of the average squared residual.

$$\sqrt{\frac{\sum_{i=1}^{n} r_i^2}{n}}$$

As with the previous statistic, the smaller this statistic, the better the model.

Residual standard deviation

Our final metric, which is used directly in inferential statistics, is the residual standard deviation

$$s_{res} = \sqrt{\frac{\sum_{i=1}^{n} r_i^2}{n-p}}$$

with p denoting the number of fitted coefficients in the model. This statistic is the standard deviation of the residuals from a given model. The smaller is this residual standard deviation, the better fitting is the model. We shall use the residual standard deviation as our metric of choice for evaluating and comparing various proposed models.

5.5.9.9.3. Motivation: How do we Construct a Good Model?

Models for 2^k and 2^{k-p} designs

Given that we have a statistic to measure the quality of a model, any model, we move to the question of how to construct reasonable models for fitting data from 2^k and 2^{k-p} designs.

Initial simple model

The simplest such proposed model is

$$Y = c + \epsilon$$

that is, the response Y = a constant + random error. This trivial model says that all of the factors (and interactions) are in fact worthless for prediction and so the best-fit model is one that consists of a simple horizontal straight line through the body of the data. The least squares estimate for this constant c in the above model is the sample mean $\bar{Y}$. The prediction equation for this model is thus

$$\hat{Y} = \bar{Y}$$

The predicted values $\hat{Y}$ for this fitted trivial model are thus given by a vector consisting of the same value (namely $\bar{Y}$) throughout. The residual vector for this model will thus simplify to simple deviations from the mean:

$$Y - \bar{Y}$$

Since the number of fitted coefficients in this model is 1 (namely the constant c), the residual standard deviation is the following:

$$s_{res} = \sqrt{\frac{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}{n-1}}$$

which is of course the familiar, commonly employed sample standard deviation. If the residual standard deviation for this trivial model were "small enough", then we could terminate the model-building process right there with no further inclusion of terms. In practice, however, this trivial model does **not** yield a residual standard deviation that is small

enough (because the common value $\bar{Y}$ will not be close enough to some of the raw responses *Y*) and so the model must be augmented--but how?

Next-step model

The logical next-step proposed model will consist of the above additive constant plus some term that will improve the predicted values the most. This will equivalently reduce the residuals the most and thus reduce the residual standard deviation the most.

Using the most important effects

As it turns out, it is a mathematical fact that the factor or interaction that has the largest estimated effect

$$\hat{E} = \bar{Y}(+) - \bar{Y}(-)$$

will necessarily, after being included in the model, yield the "biggest bang for the buck" in terms of improving the predicted values toward the response values *Y*. Hence at this point the model-building process and the effect estimation process merge.

In the previous steps in our analysis, we developed a ranked list of factors and interactions. We thus have a ready-made ordering of the terms that could be added, one at a time, to the model. This ranked list of effects is precisely what we need to cumulatively build more complicated, but better fitting, models.

Step through the ranked list of factors

Our procedure will thus be to step through, one by one, the ranked list of effects, cumulatively augmenting our current model by the next term in the list, and then compute (for all *n* design points) the predicted values, residuals, and residual standard deviation. We continue this one-term-at-a-time augmentation until the predicted values are acceptably close to the observed responses *Y* (and hence the residuals and residual standard deviation become acceptably close to zero).

Starting with the simple average, each cumulative model in this iteration process will have its own associated residual standard deviation. In practice, the iteration continues until the residual standard deviations become sufficiently small.

Cumulative residual standard deviation plot

The cumulative residual standard deviation plot is a graphical summary of the above model-building process. On the horizontal axis is a series of terms (starting with the average, and continuing on with various main effects and interactions). After the average, the ordering of terms on the horizontal axis is identical to the ordering of terms based on the half-normal probability plot ranking based on effect magnitude.

On the vertical axis is the corresponding residual standard deviation that results when the cumulative model has its coefficients fitted via least squares, and then has its predicted values, residuals, and residual standard deviations computed. The first residual standard deviation (on the far left of the cumulative residual standard deviation plot) is that which results from the model consisting of

1. the average.

The second residual standard deviation plotted is from the model consisting of

1. the average, plus
2. the term with the largest |effect|.

The third residual standard deviation plotted is from the model consisting of

1. the average, plus
2. the term with the largest |effect|, plus
3. the term with the second largest |effect|.

and so forth.

5.5.9.9.4. Motivation: How do we Know When to Stop Adding Terms?

Cumulative residual standard deviation plot typically has a hockey stick appearance

Proceeding left to right, as we add more terms to the model, the cumulative residual standard deviation "curve" will typically decrease. At the beginning (on the left), as we add large-effect terms, the decrease from one residual standard deviation to the next residual standard deviation will be large. The incremental improvement (decrease) then tends to drop off slightly. At some point the incremental improvement will typically slacken off considerably. Appearance-wise, it is thus very typical for such a curve to have a "hockey stick" appearance:

1. starting with a series of large decrements between successive residual standard deviations; then
2. hitting an elbow; then
3. having a series of gradual decrements thereafter.

Stopping rule

The cumulative residual standard deviation plot provides a visual answer to the question:

What is a good model?

by answering the related question:

When do we stop adding terms to the cumulative model?

Graphically, the most common stopping rule for adding terms is to cease immediately upon encountering the "elbow". We include all terms up to and including the elbow point since each of these terms decreased the residual standard deviation by a large amount. However, we exclude any terms afterward since these terms do not decrease the residual standard deviation fast enough to warrant inclusion in the model.

5.5.9.9.5. Motivation: What is the Form of the Model?

Models for various values of k

From the above discussion, we thus note and recommend a form of the model that consists of an additive constant plus a linear combination of main effects and interactions. What then is the specific form for the linear combination?

The following are the full models for various values of *k*. The selected final model will be a subset of the full model.

- For the trivial $k = 1$ factor case:

$$Y = f(X_1) = c + (1/2) * (B_1 * X_1) + \epsilon$$

- For the $k = 2$ factor case:

$$\begin{aligned} Y &= f(X_1, X_2) \\ &= c + (1/2) * (B_1 * X_1 + B_2 * X_2 + B_{12} * (X_1 * X_2)) + \epsilon \end{aligned}$$

- For the $k = 3$ factor case:

$$\begin{aligned} Y &= f(X_1, X_2, X_3) \\ &= c + (1/2) * (B_1 * X_1 + B_2 * X_2 + B_3 * X_3 + \\ &\quad B_{12} * (X_1 * X_2) + B_{13} * (X_1 * X_3) + B_{23} * (X_2 * X_3) + \\ &\quad B_{123} * (X_1 * X_2 * X_3)) + \epsilon \end{aligned}$$

- and for the general *k* case:

$Y = f(X_1, X_2, ..., X_k) =$
c + (1/2)*(linear combination of all main effects and all interactions of all orders) + ϵ

Ordered linear combination

The listing above has the terms ordered with the main effects, then the 2-factor interactions, then the 3-factor interactions, etc. In practice, it is recommended that the terms be ordered by importance (whether they be main effects or interactions). Aside from providing a functional representation of the response, models should help reinforce what is driving the response, which such a re-ordering does. Thus for $k = 2$, if factor 2 is most important, the 2-factor interaction is next in importance, and factor 1 is least important, then it is recommended that the above ordering of

$$\begin{aligned} Y &= f(X_1, X_2) \\ &= c + (1/2) * (B_1 * X_1 + B_2 * X_2 + B_{12} * (X_1 * X_2)) + \epsilon \end{aligned}$$

be rewritten as

$$\begin{aligned} Y &= f(X_1, X_2) \\ &= c + (1/2) * (B_2 * X_2 + B_{12} * (X_1 * X_2) + B_1 * X_1) + \epsilon \end{aligned}$$

5.5.9.9.6. Motivation: Why is the 1/2 in the Model?

Presence of 1/2 term does not affect predictive quality of model

The leading 1/2 is a multiplicative constant that we have chosen to include in our expression of the linear model. Some authors and software prefer to "simplify" the model by omitting this leading 1/2. It is our preference to include the 1/2. This follows the convention given on page 334 of Box, Hunter, and Hunter (1978). The presence or absence of the arbitrary 1/2 term does **not** affect the predictive quality of the model after least squares fitting. Clearly, if we choose to exclude the 1/2, then the least squares fitting process will simply yield estimated values of the coefficients that are twice the size of the coefficients that would result if we included the 1/2.

Included so least squares coefficient estimate equal to estimated effect

We recommend the inclusion of the 1/2 because of an additional property that we would like to impose on the model; namely, we desire that:

> the value of the least squares estimated coefficient B for a given factor (or interaction) be visually identical to the estimated effect E for that factor (or interaction).

For a given factor, say X2, the estimated least squares coefficient B2 and the estimated effect E2 are **not** in general identical in either value or concept.

Effect

For factor X2, the effect E2 is defined as the change in the mean response as we proceed from the "-" setting of the factor to the "+" setting of the factor. Symbolically:

$$\hat{E} = \bar{Y}(+) - \bar{Y}(-)$$

Note that the estimated effect E2 value does **not** involve a model per se, and is definitionally invariant to any other factors and interactions that may effect the response. We examined and derived the factor effects E in the previous steps of the general DEX analysis procedure.

On the other hand, the estimated coefficient B2 in a model is defined as the value that results when we place the model into the least squares fitting process (regression). The value that returns for B2 depends, in general, on the form of the model, on what other terms are included in the model, and on the experimental design that was run. The least squares estimate for B2 is mildly complicated since it involves a behind-the-scenes design matrix multiplication and inversion. The coefficient values B that result are generally obscured by the mathematics to make the coefficients have the collective property that the fitted model as a whole yield a minimum sum of squared deviations ("least squares").

Orthogonality

Rather remarkably, these two concepts and values:

1. factor and interaction effect estimates E, and
2. least squares coefficient estimates B

merge for the class of experimental designs for which this 10-step procedure was developed, namely, 2-level full and fractional designs that are orthogonal. Orthogonality has been promoted and chosen because of its desirable design properties. That is, every factor is balanced (every level of a factor occurs an equal number of times) and every 2-factor cross-product is balanced. But to boot, orthogonality has 2 extraordinary properties on the data analysis side:

1. For the above linear models, the usual matrix solution for the least squares estimates for the coefficients B reduce to a computationally trivial and familiar form, namely,

$$B = E = \bar{Y}(+) - \bar{Y}(-)$$

2. The usual general modeling property that the least squares estimate for a factor coefficient changes depending on what other factors have been included in or excluded from the model is now moot. With orthogonal designs, the coefficient estimates are invariant in the sense that the estimate (e.g., B2) for a given factor (e.g., X2) will **not** change as other factors and interactions are included in or excluded from the model. That is, the estimate

of the factor 2 effect (B2) remains the same regardless of what other factors are included in the model.

The net effect of the above two properties is that a factor effect can be computed once, and that value will hold for any linear model involving that term regardless of how simple or complicated the model is, provided that the design is orthogonal. This process greatly simplifies the model-building process because the need to recalculate all of the model coefficients for each new model is eliminated.

Why is 1/2 the appropriate multiplicative term in these orthogonal models?

Given the computational simplicity of orthogonal designs, why then is 1/2 the appropriate multiplicative constant? Why not 1/3, 1/4, etc.? To answer this, we revisit our specified desire that

> when we view the final fitted model and look at the coefficient associated with X2, say, we want the value of the coefficient B2 to reflect identically the expected total change ΔY in the response Y as we proceed from the "-" setting of X2 to the "+" setting of X2 (that is, we would like the estimated coefficient B2 to be identical to the estimated effect E2 for factor X2).

Thus in glancing at the final model with this form, the coefficients B of the model will immediately reflect not only the relative importance of the coefficients, but will also reflect (absolutely) the effect of the associated term (main effect or interaction) on the response.

In general, the least squares estimate of a coefficient in a linear model will yield a coefficient that is essentially a slope:

$\Delta Y/\Delta X$ = (change in response)/(change in factor levels)

associated with a given factor X. Thus in order to achieve the desired interpretation of the coefficients B as being the raw change in the Y (ΔY), we must account for and remove the change in X (ΔX).

What is the ΔX? In our design descriptions, we have chosen the notation of Box, Hunter and Hunter (1978) and set each (coded) factor to levels of "-" and "+". This "-" and "+" is a shorthand notation for -1 and +1. The advantage of this notation is that 2-factor interactions (and any higher-order interaction) also uniformly take on the closed values of -1 and +1, since

```
-1*-1 = +1
-1*+1 = -1
+1*-1 = -1
+1*+1 = +1
```

and hence the set of values that the 2-factor interactions (and all interactions) take on are in the closed set {-1,+1}. This -1 and +1 notation is superior in its consistency to the (1,2) notation of Taguchi

in which the interaction, say X1*X2, would take on the values

```
1*1 = 1
1*2 = 2
2*1 = 2
2*2 = 4
```

which yields the set {1,2,4}. To circumvent this, we would need to replace multiplication with modular multiplication (see page 440 of Ryan (2000)). Hence, with the -1,+1 values for the main factors, we also have -1,+1 values for all interactions which in turn yields (for all terms) a consistent ΔX of

$$\Delta X = (+1) - (-1) = +2$$

In summary then,

$$\begin{aligned} B &= (\Delta Y / \Delta X) \\ &= (\Delta Y) \ / \ 2 \\ &= (1/2) \ * \ (\Delta Y) \end{aligned}$$

and so to achieve our goal of having the final coefficients reflect ΔY only, we simply gather up all of the 2's in the denominator and create a leading multiplicative constant of 1 with denominator 2, that is, 1/2.

Example for k = 1 case

For example, for the trivial $k = 1$ case, the obvious model

$$Y = \text{intercept} + \text{slope}*X1$$
$$Y = c + (\Delta Y / \Delta X)*X1$$

becomes

$$Y = c + (1/\Delta X) * (\Delta Y)*X1$$

or simply

$$Y = c + (1/2) * (\Delta Y)*X1$$
$$Y = c + (1/2)*(\text{factor 1 effect})*X1$$
$$Y = c + (1/2)*(B^*)*X1, \text{ with } B^* = 2B = E$$

This $k = 1$ factor result is easily seen to extend to the general k-factor case.

5.5.9.9.7. Motivation: What are the Advantages of the Linear Combinatoric Model?

Advantages: perfect fit and comparable coefficients

The linear model consisting of main effects and all interactions has two advantages:

1. Perfect Fit: If we choose to include in the model all of the main effects and all interactions (of all orders), then the resulting least squares fitted model will have the property that the predicted values will be **identical** to the raw response values *Y*. We will illustrate this in the next section.
2. Comparable Coefficients: Since the model fit has been carried out in the coded factor (-1,+1) units rather than the units of the original factor (temperature, time, pressure, catalyst concentration, etc.), the factor coefficients immediately become comparable to one another, which serves as an immediate mechanism for the scale-free ranking of the relative importance of the factors.

Example

To illustrate in detail the above latter point, suppose the (-1,+1) factor *X*1 is really a coding of temperature T with the original temperature ranging from 300 to 350 degrees and the (-1,+1) factor *X*2 is really a coding of time *t* with the original time ranging from 20 to 30 minutes. Given that, a linear model in the original temperature T and time *t* would yield coefficients whose magnitude depends on the magnitude of T (300 to 350) and *t* (20 to 30), and whose value would change if we decided to change the units of T (e.g., from Fahrenheit degrees to centigrade degrees) and *t* (e.g., from minutes to seconds). All of this is avoided by carrying out the fit not in the original units for T (300,350) and *t* (20,30), but in the coded units of X1 (-1,+1) and X2 (-1,+1). The resulting coefficients are unit-invariant, and thus the coefficient magnitudes reflect the true contribution of the factors and interactions without regard to the unit of measurement.

Coding does not lead to loss of generality

Such coding leads to no loss of generality since the coded factor may be expressed as a simple linear relation of the original factor ($X1$ to T, $X2$ to t). The unit-invariant coded coefficients may be easily transformed to unit-sensitive original coefficients if so desired.

5.5.9.9.8. Motivation: How do we use the Model to Generate Predicted Values?

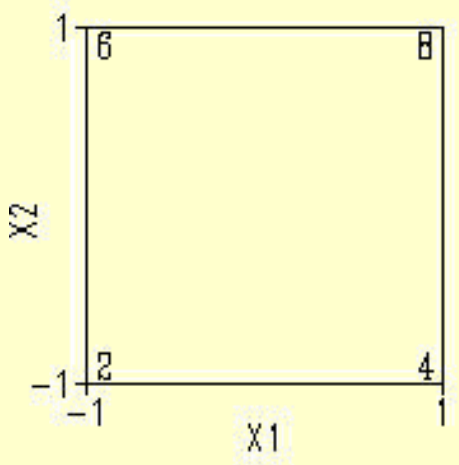
Design matrix with response for 2 factors

To illustrate the details as to how a model may be used for prediction, let us consider a simple case and generalize from it. Consider the simple Yates-order 2^2 full factorial design in $X1$ and $X2$, augmented with a response vector Y:

X1	X2	Y
-	-	2
+	-	4
-	+	6
+	+	8

Geometric representation

This can be represented geometrically



Determining the prediction equation

For this case, we might consider the model

$$Y = c + (1/2) * (B_1 * X_1 + B_2 * X_2 + B_{12} * (X_1 * X_2)) + \epsilon$$

From the above diagram, we may deduce that the estimated factor effects are:

c = the average response = $\bar{Y}$
= (2 + 4 + 6 + 8) / 4 = 5

B_1 = average change in Y as X>1 goes from -1 to +1
= ((4-2) + (8-6)) / 2 = (2 + 2) / 2 = 2

Note: the (4-2) is the change in Y (due to $X1$) on the lower axis; the (8-6) is the change in Y (due to $X1$) on the upper axis.

B_2 = average change in Y as $X2$ goes from -1 to +1
= ((6-2) + (8-4)) / 2 = (4 + 4) / 2 = 4

B_{12} = interaction = (the less obvious) average change in Y as $X1$*$X2$ goes from -1 to +1
= ((2-4) + (8-6)) / 2 = (-2 + 2) / 2 = 0

and so the fitted model (that is, the prediction equation) is

$$\hat{Y} = 5 + (1/2) * (2 * X_1 + 4 * X_2 + 0 * (X_1 * X_2))$$

or with the terms rearranged in descending order of importance

$$\hat{Y} = 5 + (1/2) * (4 * X_2 + 2 * X_1) + \epsilon$$

Table of fitted values

Substituting the values for the four design points into this equation yields the following fitted values

X1	X2	Y	$\hat{Y}$
-	-	2	2
+	-	4	4
-	+	6	6
+	+	8	8

Perfect fit

This is a perfect-fit model. Such perfect-fit models will result anytime (in this orthogonal 2-level design family) we include all main effects and all interactions. Remarkably, this is true not only for k = 2 factors, but for general k.

Residuals

For a given model (any model), the difference between the response value Y and the predicted value $\hat{Y}$ is referred to as the "residual":

residual = Y - $\hat{Y}$

The perfect-fit full-blown (all main factors and all interactions of all orders) models will have all residuals identically zero.

The perfect fit is a mathematical property that comes if we choose to use the linear model with all possible terms.

Price for perfect fit

What price is paid for this perfect fit? One price is that the variance of $\hat{Y}$ is increased unnecessarily. In addition, we have a non-parsimonious model. We must compute and carry the average and the coefficients of all main effects and all interactions. Including the average, there will in general be 2^k coefficients to fully describe the fitting of the $n = 2^k$ points. This is very much akin to the $Y = f(X)$ polynomial fitting of n distinct points. It is well known that this may be done "perfectly" by fitting a polynomial of degree n-1. It is comforting to know that such perfection is mathematically attainable, but in practice do we want to do this all the time or even anytime? The answer is generally "no" for two reasons:

1. Noise: It is very common that the response data Y has noise (= error) in it. Do we want to go out of our way to fit such noise? Or do we want our model to filter out the noise and just fit the "signal"? For the latter, fewer coefficients may be in order, in the same spirit that we may forego a perfect-fitting (but jagged) 11-th degree polynomial to 12 data points, and opt out instead for an imperfect (but smoother) 3rd degree polynomial fit to the 12 points.
2. Parsimony: For full factorial designs, to fit the $n = 2^k$ points we would need to compute 2^k coefficients. We gain information by noting the magnitude and sign of such coefficients, but numerically we have n data values Y as input and n coefficients B as output, and so no numerical reduction has been achieved. We have simply used one set of n numbers (the data) to obtain another set of n numbers (the coefficients). Not all of these coefficients will be equally important. At times that importance becomes clouded by the sheer volume of the $n = 2^k$ coefficients. Parsimony suggests that our result should be simpler and more focused than our n starting points. Hence fewer retained coefficients are called for.

The net result is that in practice we almost always give up the perfect, but unwieldy, model for an imperfect, but parsimonious, model.

Imperfect fit

The above calculations illustrated the computation of predicted values for the full model. On the other hand, as discussed above, it will generally be convenient for signal or parsimony purposes to deliberately omit some unimportant factors. When the analyst chooses such a model, we note that the methodology for computing predicted values $\hat{Y}$ is precisely the same. In such a case, however, the resulting predicted values will in general **not** be identical to the original response values Y; that is, we no longer obtain a perfect fit. Thus, linear models that omit some terms will have virtually all non-zero residuals.

5.5.9.9.9. Motivation: How do we Use the Model Beyond the Data Domain?

Interpolation and extrapolation

The previous section illustrated how to compute predicted values at the points included in the design. One of the virtues of modeling is that the resulting prediction equation is **not** restricted to the design data points. From the prediction equation, predicted values can be computed elsewhere and anywhere:

1. within the domain of the data (interpolation);
2. outside of the domain of the data (extrapolation).

In the hands of an expert scientist/engineer/analyst, the ability to predict elsewhere is extremely valuable. Based on the fitted model, we have the ability to compute predicted values for the response at a large number of internal and external points. Thus the analyst can go beyond the handful of factor combinations at hand and can get a feel (typically via subsequent contour plotting) as to what the nature of the entire response surface is.

This added insight into the nature of the response is "free" and is an incredibly important benefit of the entire model-building exercise.

Predict with caution

Can we be fooled and misled by such a mathematical and computational exercise? After all, is not the only thing that is "real" the data, and everything else artificial? The answer is "yes", and so such interpolation/extrapolation is a double-edged sword that must be wielded with care. The best attitude, and especially for extrapolation, is that the derived conclusions must be viewed with extra caution.

By construction, the recommended fitted models should be good at the design points. If the full-blown model were used, the fit will be perfect. If the full-blown model is reduced just a bit, then the fit will still typically be quite good. By continuity, one would expect perfection/goodness at the design points would lead to goodness in the immediate vicinity of the design points. However, such local goodness

does **not** guarantee that the derived model will be good at some distance from the design points.

Do confirmation runs

Modeling and prediction allow us to go beyond the data to gain additional insights, but they must be done with great caution. Interpolation is generally safer than extrapolation, but mis-prediction, error, and misinterpretation are liable to occur in either case.

The analyst should definitely perform the model-building process and enjoy the ability to predict elsewhere, but the analyst must always be prepared to validate the interpolated and extrapolated predictions by collection of additional real, confirmatory data. The general empirical model that we recommend knows "nothing" about the engineering, physics, or chemistry surrounding your particular measurement problem, and although the model is the best generic model available, it must nonetheless be confirmed by additional data. Such additional data can be obtained pre-experimentally or post-experimentally. If done pre-experimentally, a recommended procedure for checking the validity of the fitted model is to augment the usual 2^k or 2^{k-p} designs with additional points at the center of the design. This is discussed in the next section.

Applies only for continuous factors

Of course, all such discussion of interpolation and extrapolation makes sense only in the context of continuous ordinal factors such as temperature, time, pressure, size, etc. Interpolation and extrapolation make no sense for discrete non-ordinal factors such as supplier, operators, design types, etc.

5.5.9.9.10. Motivation: What is the Best Confirmation Point for Interpolation?

Augment via center point

For the usual continuous factor case, the best (most efficient and highest leverage) additional model-validation point that may be added to a 2^k or 2^{k-p} design is at the center point. This center point augmentation "costs" the experimentalist only one additional run.

Example

For example, for the $k = 2$ factor (Temperature (300 to 350), and time (20 to 30)) experiment discussed in the previous sections, the usual 4-run 2^2 full factorial design may be replaced by the following 5-run 2^2 full factorial design with a center point.

X1	X2	Y
-	-	2
+	-	4
-	+	6
+	+	8
0	0	

Predicted value for the center point

Since "-" stands for -1 and "+" stands for +1, it is natural to code the center point as (0,0). Using the recommended model

$$\hat{Y} = 5 + (1/2) * (4 * X_2 + 2 * X_1) + \epsilon$$

we can substitute 0 for X1 and X2 to generate the predicted value of 5 for the confirmatory run.

Importance of the confirmatory run

The importance of the confirmatory run cannot be overstated. If the confirmatory run at the center point yields a data value of, say, $Y = 5.1$, since the predicted value at the center is 5 and we know the model is perfect at the corner points, that would give the analyst a greater confidence that the quality of the fitted model may extend over the entire interior (interpolation) domain. On the other hand, if the confirmatory run yielded a center point data value quite different (e.g., $Y = 7.5$) from the center point predicted value of 5, then that would prompt the analyst to **not** trust the fitted model even for interpolation purposes. Hence when our factors are continuous, a single confirmatory run at the center point helps immensely in assessing the range of trust for our model.

Replicated center points

In practice, this center point value frequently has two, or even three or more, replications. This not only provides a reference point for assessing the interpolative power of the model at the center, but it also allows us to compute model-free estimates of the natural error in the data. This in turn allows us a more rigorous method for computing the uncertainty for individual coefficients in the model and for rigorously carrying out a lack-of-fit test for assessing general model adequacy.

5.5.9.9.11. Motivation: How do we Use the Model for Interpolation?

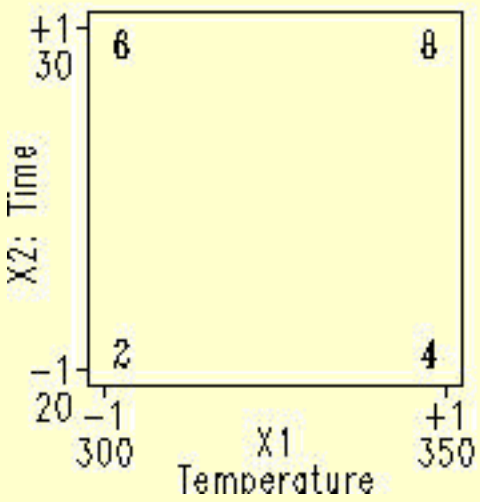
Design table in original data units

As for the mechanics of interpolation itself, consider a continuation of the prior $k = 2$ factor experiment. Suppose temperature T ranges from 300 to 350 and time t ranges from 20 to 30, and the analyst can afford $n = 4$ runs. A 2^2 full factorial design is run. Forming the coded temperature as $X1$ and the coded time as $X2$, we have the usual:

Temperature	Time	X1	X2	Y
300	20	-	-	2
350	20	+	-	4
300	30	-	+	6
350	30	+	+	8

Graphical representation

Graphically the design and data are as follows:



Typical interpolation question

As before, from the data, the "perfect-fit" prediction equation is

$$\hat{Y} = 5 + 0.5 * (4 * X_2 + 2 * X_1)$$

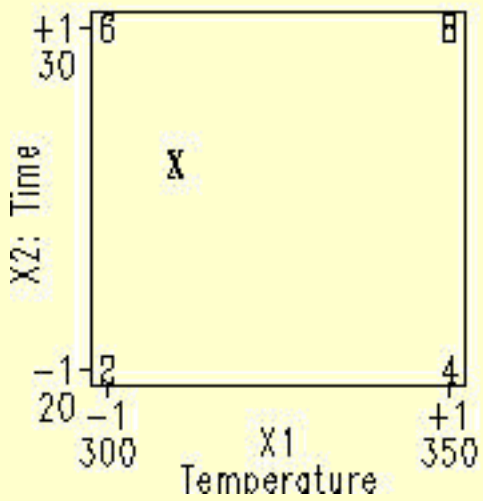
We now pose the following typical interpolation question:

> From the model, what is the predicted response at, say, temperature = 310 and time = 26?

In short:

> $\hat{Y}(T = 310, t = 26) = ?$

To solve this problem, we first view the $k = 2$ design and data graphically, and note (via an "X") as to where the desired ($T = 310$, $t = 26$) interpolation point is:



Predicting the response for the interpolated point

The important next step is to convert the raw (in units of the original factors T and t) interpolation point into a coded (in units of $X1$ and $X2$) interpolation point. From the graph or otherwise, we note that a linear translation between T and $X1$, and between t and $X2$ yields

> $T = 300 \Rightarrow X1 = -1$
> $T = 350 \Rightarrow X1 = +1$

thus

> $X1 = 0$ is at $T = 325$

```
      |-------------|-------------|
     -1      ?      0             +1
     300    310     325           350
```

which in turn implies that

> $T = 310 \Rightarrow X1 = -0.6$

Similarly,

$t = 20 \Rightarrow X2 = -1$

$t = 30 \Rightarrow X2 = +1$

therefore,

$X2 = 0$ is at $t = 25$

```
  |-------------|-------------|
-1              0   ?         +1
20              25  26        30
```

thus

t = 26 => X2 = +0.2

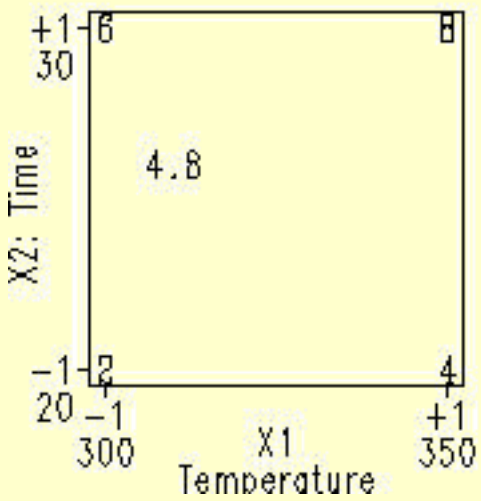
Substituting $X1 = -0.6$ and $X2 = +0.2$ into the prediction equation

$$\hat{Y} = 5 + 0.5 * (4 * X_2 + 2 * X_1)$$

yields a predicted value of 4.8.

Graphical representation of response value for interpolated data point

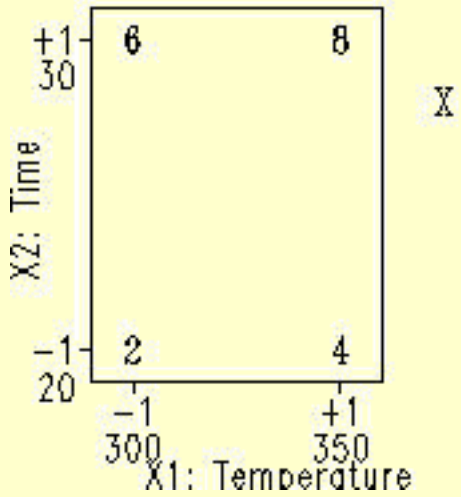
Thus



5.5.9.9.12. Motivation: How do we Use the Model for Extrapolation?

Graphical representation of extrapolation

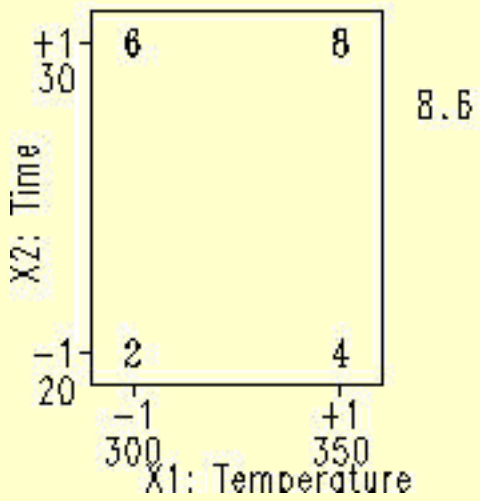
Extrapolation is performed similarly to interpolation. For example, the predicted value at temperature T = 375 and time t = 28 is indicated by the "X":



and is computed by substituting the values $X1$ = +2.0 (T=375) and $X2$ = +0.8 (t=28) into the prediction equation

$$\hat{Y} = 5 + 0.5 * (4 * X_2 + 2 * X_1)$$

yielding a predicted value of 8.6. Thus we have



Pseudo-data

The predicted value from the modeling effort may be viewed as pseudo-data, data obtained without the experimental effort. Such "free" data can add tremendously to the insight via the application of graphical techniques (in particular, the contour plots and can add significant insight and understanding as to the nature of the response surface relating *Y* to the *X*'s.

But, again, a final word of caution: the "pseudo data" that results from the modeling process is exactly that, pseudo-data. It is **not** real data, and so the model and the model's predicted values must be validated by additional confirmatory (real) data points. A more balanced approach is that:

> Models may be trusted as "real" [that is, to generate predicted values and contour curves], but must always be verified [that is, by the addition of confirmatory data points].

The rule of thumb is thus to take advantage of the available and recommended model-building mechanics for these 2-level designs, but do treat the resulting derived model with an equal dose of both optimism and caution.

Summary

In summary, the motivation for model building is that it gives us insight into the nature of the response surface along with the ability to do interpolation and extrapolation; further, the motivation for the use of the cumulative residual standard deviation plot is that it serves as an easy-to-interpret tool for determining a good and parsimonious model.

5.5.9.10. DEX contour plot

Purpose

The dex contour plot answers the question:

> Where else could we have run the experiment to optimize the response?

Prior steps in this analysis have suggested the best setting for each of the *k* factors. These best settings may have been derived from

1. Data: which of the *n* design points yielded the best response, and what were the settings of that design point, or from
2. Averages: what setting of each factor yielded the best response "on the average".

This 10th (and last) step in the analysis sequence goes beyond the limitations of the *n* data points already chosen in the design and replaces the data-limited question

> "From among the *n* data points, what was the best setting?"

to a region-related question:

> "In general, what should the settings have been to optimize the response?"

Output

The outputs from the dex contour plot are

1. Primary: Best setting (X_{10}, X_{20}, ..., X_{k0}) for each of the *k* factors. This derived setting should yield an optimal response.
2. Secondary: Insight into the nature of the response surface and the importance/unimportance of interactions.

Definition

A dex contour plot is formed by

- Vertical Axis: The second most important factor in the experiment.
- Horizontal Axis: The most important factor in the experiment.

More specifically, the dex contour plot is constructed and utilized via the following 7 steps:

1. Axes
2. Contour Curves
3. Optimal Response Value
4. Best Corner
5. Steepest Ascent/Descent
6. Optimal Curve
7. Optimal Setting

with

1. Axes: Choose the two most important factors in the experiment as the two axes on the plot.
2. Contour Curves: Based on the fitted model and the best data settings for all of the remaining factors, draw contour curves involving the two dominant factors. This yields a

graphical representation of the response surface. The details for constructing linear contour curves are given in a later section.

3. Optimal Value: Identify the theoretical value of the response that constitutes "best." In particular, what value would we like to have seen for the response?
4. Best "Corner": The contour plot will have four "corners" for the two most important factors X_i and X_j: (X_i,X_j) = (-,-), (-,+), (+,-), and (+,+). From the data, identify which of these four corners yields the highest average response $\bar{Y}$.
5. Steepest Ascent/Descent: From this optimum corner point, and based on the nature of the contour lines near that corner, step out in the direction of steepest ascent (if maximizing) or steepest descent (if minimizing).
6. Optimal Curve: Identify the curve on the contour plot that corresponds to the ideal optimal value.
7. Optimal Setting: Determine where the steepest ascent/descent line intersects the optimum contour curve. This point represents our "best guess" as to where we could have run our experiment so as to obtain the desired optimal response.

Motivation

In addition to increasing insight, most experiments have a goal of optimizing the response. That is, of determining a setting (X_{10}, X_{20}, ..., X_{k0}) for which the response is optimized.

The tool of choice to address this goal is the dex contour plot. For a pair of factors X_i and X_j, the dex contour plot is a 2-dimensional representation of the 3-dimensional $Y = f(X_i,X_j)$ response surface. The position and spacing of the isocurves on the dex contour plot are an easily interpreted reflection of the nature of the surface.

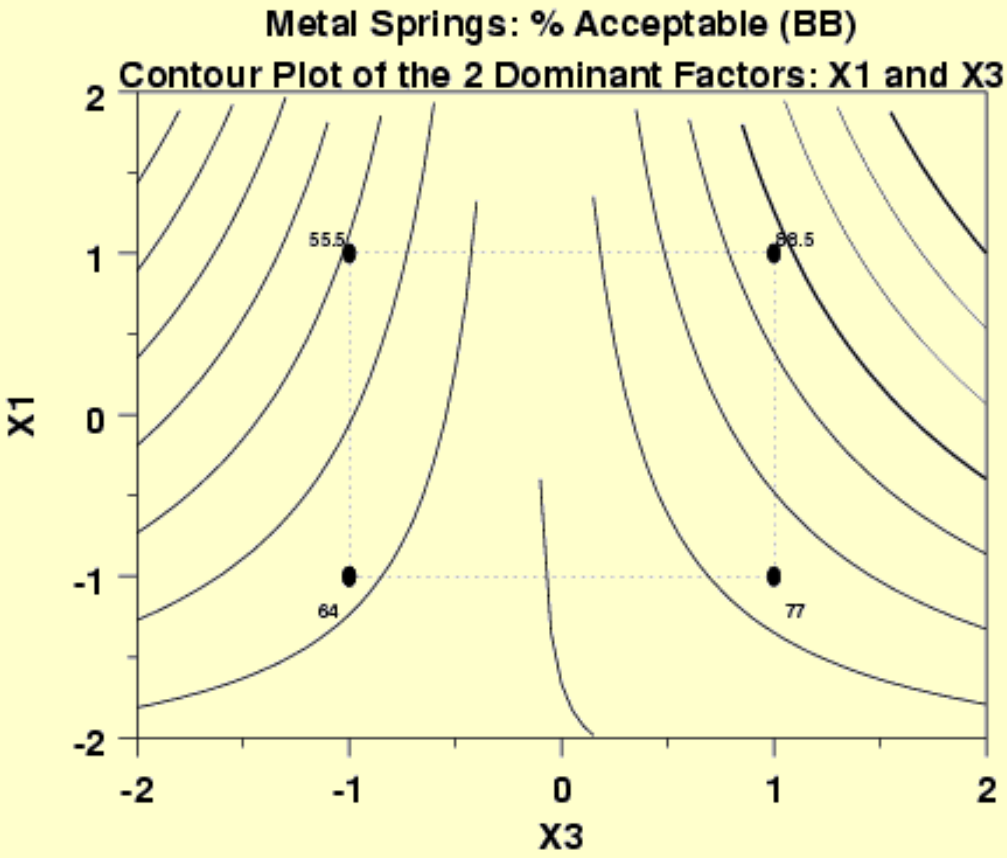
In terms of the construction of the dex contour plot, there are three aspects of note:

1. Pairs of Factors: A dex contour plot necessarily has two axes (only); hence only two out of the k factors can be represented on this plot. All other factors must be set at a fixed value (their optimum settings as determined by the ordered data plot, the dex mean plot, and the interaction effects matrix plot).
2. Most Important Factor Pair: Many dex contour plots are possible. For an experiment with k factors, there are $\binom{k}{2} = \frac{k!}{2!(k-2)!} = \frac{k(k-1)}{2}$ possible contour plots. For example, for k = 4 factors there are 6 possible contour plots: X_1 and X_2, X_1 and X_3, X_1 and X_4, X_2 and X_3, X_2 and X_4, and X_3 and X_4. In practice, we usually generate only one contour plot involving the two most important factors.
3. Main Effects Only: The contour plot axes involve main effects only, not interactions. The rationale for this is that the "deliverable" for this step is k settings, a best setting for each of the k factors. These k factors are real and can be controlled, and so optimal settings can be used in production. Interactions are of a different nature as there is no "knob on the machine" by which an interaction may be set to -, or to +. Hence the candidates for the axes on contour plots are main effects only--no interactions.

In summary, the motivation for the dex contour plot is that it is an easy-to-use graphic that provides insight as to the nature of the response surface, and provides a specific answer to the question "Where (else) should we have collected the data so to have optimized the response?".

Plot for defective springs data

Applying the dex contour plot for the defective springs data set yields the following plot.



How to interpret

From the dex contour plot for the defective springs data, we note the following regarding the 7 framework issues:

- Axes
- Contour curves
- Optimal response value
- Optimal response curve
- Best corner
- Steepest Ascent/Descent
- Optimal setting

Conclusions for the defective springs data

The application of the dex contour plot to the defective springs data set results in the following conclusions:

1. Optimal settings for the "next" run:

   Coded : ($X1$,$X2$,$X3$) = (+1.5,+1.0,+1.3)
   Uncoded: (OT,CC,QT) = (1637.5,0.7,127.5)

2. Nature of the response surface:

   The $X1$*$X3$ interaction is important, hence the effect of factor $X1$ will change depending on the setting of factor $X3$.

NIST SEMATECH HOME TOOLS & AIDS SEARCH BACK NEXT

5.5.9.10.1. How to Interpret: Axes

What factors go on the 2 axes?

For this first item, we choose the two most important factors in the experiment as the plot axes.

These are determined from the ranked list of important factors as discussed in the previous steps. In particular, the |effects| plot includes a ranked factor table. For the defective springs data, that ranked list consists of

Factor/Interaction	Effect Estimate
*X*1	23
*X*1**X*3	10
*X*2	-5
*X*3	1.5
*X*1**X*2	1.5
*X*1**X*2**X*3	0.5
*X*2**X*3	0

Possible choices

In general, the two axes of the contour plot could consist of

- *X*1 and *X*2,
- *X*1 and *X*3, or
- *X*2 and *X*3.

In this case, since *X*1 is the top item in the ranked list, with an estimated effect of 23, *X*1 is the most important factor and so will occupy the horizontal axis of the contour plot. The admissible list thus reduces to

- *X*1 and *X*2, or
- *X*1 and *X*3.

To decide between these two pairs, we look to the second item in the ranked list. This is the interaction term *X*1**X*3, with an estimated effect of 10. Since interactions are **not** allowed as contour plot axes, *X*1**X*3 must be set aside. On the other hand, the components of this interaction

(*X*1 and *X*3) are not to be set aside. Since *X*1 has already been identified as one axis in the contour plot, this suggests that the other component (*X*3) be used as the second axis. We do so. Note that *X*3 itself does **not** need to be important (in fact, it is noted that *X*3 is ranked fourth in the listed table with a value of 1.5).

In summary then, for this example the contour plot axes are:

Horizontal Axis: *X*1
Vertical Axis: *X*3

Four cases for recommended choice of axes

Other cases can be more complicated. In general, the recommended rule for selecting the two plot axes is that they be drawn from the first two items in the ranked list of factors. The following four cases cover most situations in practice:

- Case 1:
  1. Item 1 is a main effect (e.g., *X*3)
  2. Item 2 is another main effect (e.g., *X*5)

  Recommended choice:
  1. Horizontal axis: item 1 (e.g., *X*3);
  2. Vertical axis: item 2 (e.g., *X*5).
- Case 2:
  1. Item 1 is a main effect (e.g., *X*3)
  2. Item 2 is a (common-element) interaction (e.g., X3*X4)

  Recommended choice:
  1. Horizontal axis: item 1 (e.g., *X*3);
  2. Vertical axis: the remaining component in item 2 (e.g., *X*4).
- Case 3:
  1. Item 1 is a main effect (e.g., *X*3)
  2. Item 2 is a (non-common-element) interaction (e.g., *X*2**X*4)

  Recommended choice:
  1. Horizontal axis: item 1 (e.g., *X*3);
  2. Vertical axis: either component in item 2 (e.g., *X*2, or *X*4), but preferably the one with the largest individual effect (thus scan the rest of the ranked factors and if the *X*2 |effect| > *X*4 |effect|, choose *X*2; otherwise choose *X*4).
- Case 4:
  1. Item 1 is a (2-factor) interaction (e.g., *X*2**X*4)

2. Item 2 is anything

Recommended choice:

1. Horizontal axis: component 1 from the item 1 interaction e.g., *X*2);
2. Horizontal axis: component 2 from the item 1 interaction (e.g., *X*4).

5.5.9.10.2. How to Interpret: Contour Curves

Non-linear appearance of contour curves implies strong interaction

Based on the fitted model (cumulative residual standard deviation plot) and the best data settings for all of the remaining factors, we draw contour curves involving the two dominant factors. This yields a graphical representation of the response surface.

Before delving into the details as to how the contour lines were generated, let us first note as to what insight can be gained regarding the general nature of the response surface. For the defective springs data, the dominant characteristic of the contour plot is the non-linear (fan-shaped, in this case) appearance. Such non-linearity implies a strong *X*1**X*3 interaction effect. If the *X*1**X*3 interaction were small, the contour plot would consist of a series of near-parallel lines. Such is decidedly not the case here.

Constructing the contour curves

As for the details of the construction of the contour plot, we draw on the model-fitting results that were achieved in the cumulative residual standard deviation plot. In that step, we derived the following good-fitting prediction equation:

$$\hat{Y} = 71.25 + 0.5 * (23 * X1 + 10 * (X1 * X3) - 5 * X2)$$

The contour plot has axes of *X*1 and *X*3. *X*2 is not included and so a fixed value of *X*2 must be assigned. The response variable is the percentage of acceptable springs, so we are attempting to maximize the response. From the ordered data plot, the main effects plot, and the interaction effects matrix plot of the general analysis sequence, we saw that the best setting for factor *X*2 was "-". The best observed response data value (*Y* = 90) was achieved with the run (*X*1,*X*2,*X*3) = (+,-,+), which has *X*2 = "-". Also, the average response for *X*2 = "-" was 73 while the average response for *X*2 = "+" was 68. We thus set *X*2 = -1 in the prediction equation to obtain

$$\hat{Y} = 71.25 + 0.5 * (23 * X1 + 10 * (X1 * X3) - 5 * (-1))$$
$$\hat{Y} = 76.25 + 0.5 * (23 * X1 + 10 * (X1 * X3))$$

This equation involves only *X*1 and *X*3 and is immediately usable for the *X*1 and *X*3 contour plot. The raw response values in the data ranged from 52 to 90. The

response implies that the theoretical worst is $Y = 0$ and the theoretical best is $Y = 100$.

To generate the contour curve for, say, $Y = 70$, we solve

$$70 = 76.25 + 0.5 * (23 * X1 + 10 * (X1 * X3))$$

by rearranging the equation in $X3$ (the vertical axis) as a function of $X1$ (the horizontal axis). By substituting various values of $X1$ into the rearranged equation, the above equation generates the desired response curve for $Y = 70$. We do so similarly for contour curves for any desired response value Y.

Values for X1

For these $X3 = g(X1)$ equations, what values should be used for $X1$? Since $X1$ is coded in the range -1 to +1, we recommend expanding the horizontal axis to -2 to +2 to allow extrapolation. In practice, for the dex contour plot generated previously, we chose to generate X1 values from -2, at increments of .05, up to +2. For most data sets, this gives a smooth enough curve for proper interpretation.

Values for Y

What values should be used for Y? Since the total theoretical range for the response Y (= percent acceptable springs) is 0% to 100%, we chose to generate contour curves starting with 0, at increments of 5, and ending with 100. We thus generated 21 contour curves. Many of these curves did not appear since they were beyond the -2 to +2 plot range for the X1 and X3 factors.

Summary

In summary, the contour plot curves are generated by making use of the (rearranged) previously derived prediction equation. For the defective springs data, the appearance of the contour plot implied a strong X1*X3 interaction.

5.5.9.10.3. How to Interpret: Optimal Response Value

Need to define "best"

We need to identify the theoretical value of the response that would constitute "best". What value would we like to have seen for the response?

For example, if the response variable in a chemical experiment is percent reacted, then the ideal theoretical optimum would be 100%. If the response variable in a manufacturing experiment is amount of waste, then the ideal theoretical optimum would be zero. If the response variable in a flow experiment is the fuel flow rate in an engine, then the ideal theoretical optimum (as dictated by engine specifications) may be a specific value (e.g., 175 cc/sec). In any event, for the experiment at hand, select a number that represents the ideal response value.

Optimal value for this example

For the defective springs data, the response (percentage of acceptable springs) ranged from $Y = 52$ to 90. The theoretically worst value would be 0 (= no springs are acceptable), and the theoretically best value would be 100 (= 100% of the springs are acceptable). Since we are trying to maximize the response, the selected optimal value is 100.

5.5.9.10.4. How to Interpret: Best Corner

Four corners representing 2 levels for 2 factors

The contour plot will have four "corners" (two factors times two settings per factor) for the two most important factors X_i and X_j: (X_i,X_j) = (-,-), (-,+), (+,-), or (+,+). Which of these four corners yields the highest average response $\bar{Y}$? That is, what is the "best corner"?

Use the raw data

This is done by using the raw data, extracting out the two "axes factors", computing the average response at each of the four corners, then choosing the corner with the best average.

For the defective springs data, the raw data were

X1	X2	X3	Y
-	-	-	67
+	-	-	79
-	+	-	61
+	+	-	75
-	-	+	59
+	-	+	90
-	+	+	52
+	+	+	87

The two plot axes are $X1$ and $X3$ and so the relevant raw data collapses to

X1	X3	Y
-	-	67
+	-	79
-	-	61
+	-	75
-	+	59
+	+	90
-	+	52
+	+	87

Averages which yields averages

X1	X3	Y
-	-	(67 + 61)/2 = 64
+	-	(79 + 75)/2 = 77
-	+	(59 + 52)/2 = 55.5
+	+	(90 + 87)/2 = 88.5

These four average values for the corners are annotated on the plot. The best (highest) of these values is 88.5. This comes from the (+,+) upper right corner. We conclude that for the defective springs data the best corner is (+,+).

5.5.9.10.5. How to Interpret: Steepest Ascent/Descent

Start at optimum corner point

From the optimum corner point, based on the nature of the contour surface at that corner, step out in the direction of steepest ascent (if maximizing) or steepest descent (if minimizing).

Defective springs example

Since our goal for the defective springs problem is to maximize the response, we seek the path of steepest ascent. Our starting point is the best corner (the upper right corner (+,+)), which has an average response value of 88.5. The contour lines for this plot have increments of 5 units. As we move from left to right across the contour plot, the contour lines go from low to high response values. In the plot, we have drawn the maximum contour level, 105, as a thick line. For easier identification, we have also drawn the contour level of 90 as thick line. This contour level of 90 is immediately to the right of the best corner

Conclusions on steepest ascent for defective springs example

The nature of the contour curves in the vicinity of (+,+) suggests a path of steepest ascent

1. in the "northeast" direction
2. about 30 degrees above the horizontal.

5.5.9.10.6. How to Interpret: Optimal Curve

Corresponds to ideal optimum value

The optimal curve is the curve on the contour plot that corresponds to the ideal optimum value.

Defective springs example

For the defective springs data, we search for the Y = 100 contour curve. As determined in the steepest ascent/descent section, the Y = 90 curve is immediately outside the (+,+) point. The next curve to the right is the Y = 95 curve, and the next curve beyond that is the Y = 100 curve. This is the optimal response curve.

5.5.9.10.7. How to Interpret: Optimal Setting

Optimal setting

The "near-point" optimality setting is the intersection of the steepest-ascent line with the optimal setting curve.

Theoretically, any ($X1$,$X3$) setting along the optimal curve would generate the desired response of $Y = 100$. In practice, however, this is true only if our estimated contour surface is identical to "nature's" response surface. In reality, the plotted contour curves are truth estimates based on the available (and "noisy") $n = 8$ data values. We are confident of the contour curves in the vicinity of the data points (the four corner points on the chart), but as we move away from the corner points, our confidence in the contour curves decreases. Thus the point on the $Y = 100$ optimal response curve that is "most likely" to be valid is the one that is closest to a corner point. Our objective then is to locate that "near-point".

Defective springs example

In terms of the defective springs contour plot, we draw a line from the best corner, (+,+), outward and perpendicular to the $Y = 90$, $Y = 95$, and $Y = 100$ contour curves. The $Y = 100$ intersection yields the "nearest point" on the optimal response curve.

Having done so, it is of interest to note the coordinates of that optimal setting. In this case, from the graph, that setting is (in coded units) approximately at

($X1 = 1.5$, $X3 = 1.3$)

Table of coded and uncoded factors

With the determination of this setting, we have thus, in theory, formally completed our original task. In practice, however, more needs to be done. We need to know "What is this optimal setting, not just in the coded units, but also in the original (uncoded) units"? That is, what does (*X*1=1.5, *X*3=1.3) correspond to in the units of the original data?

To deduce his, we need to refer back to the original (uncoded) factors in this problem. They were:

Coded Factor	**Uncoded Factor**
*X*1	OT: Oven Temperature
*X*2	CC: Carbon Concentration
*X*3	QT: Quench Temperature

Uncoded and coded factor settings

These factors had settings-- what were the settings of the coded and uncoded factors? From the original description of the problem, the uncoded factor settings were:

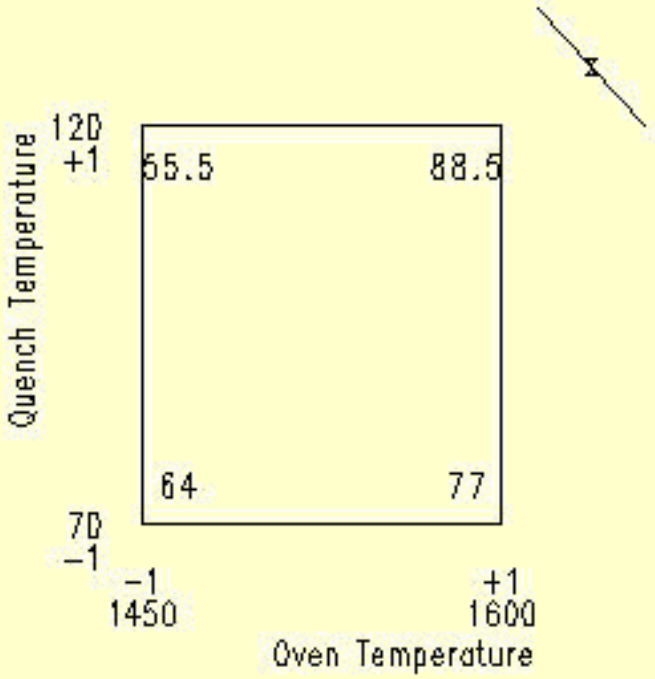
1. Oven Temperature (1450 and 1600 degrees)
2. Carbon Concentration (.5% and .7%)
3. Quench Temperature (70 and 120 degrees)

with the usual settings for the corresponding coded factors:

1. *X*1 (-1,+1)
2. *X*2 (-1,+1)
3. *X*3 (-1,+1)

Diagram

To determine the corresponding setting for (X1=1.5, X3=1.3), we thus refer to the following diagram, which mimics a scatter plot of response averages--oven temperature (OT) on the horizontal axis and quench temperature (QT) on the vertical axis:



The "X" on the chart represents the "near point" setting on the optimal curve.

Optimal setting for X1 (oven temperature)

To determine what "X" is in uncoded units, we note (from the graph) that a linear transformation between OT and *X*1 as defined by

OT = 1450 => *X*1 = -1
OT = 1600 => *X*1 = +1

yields

*X*1 = 0 being at OT = (1450 + 1600) / 2 = 1525

thus

```
            |-------------|-------------|
X1:        -1             0             +1
OT:      1450          1525          1600
```

and so X1 = +2, say, would be at oven temperature OT = 1675:

```
            |-------------|-------------|-------------|
X1:        -1             0             +1            +2
OT:      1450          1525          1600          1675
```

and hence the optimal *X*1 setting of 1.5 must be at

OT = 1600 + 0.5*(1675-1600) = 1637.5

Optimal setting for X3 (quench temperature)

Similarly, from the graph we note that a linear transformation between quench temperature QT and coded factor *X*3 as specified by

QT = 70 => *X*3 = -1
QT = 120 => *X*3 = +1

yields

*X*3 = 0 being at QT = (70 + 120) / 2 = 95

as in

```
         |-------------|-------------|
X3:     -1             0             +1
QT:     70            95            120
```

and so *X*3 = +2, say, would be quench temperature = 145:

```
         |-------------|-------------|-------------|
X3:     -1             0             +1            +2
QT:     70            95            120           145
```

Hence, the optimal *X*3 setting of 1.3 must be at

QT = 120 + .3*(145-120)
QT = 127.5

Summary of optimal settings

In summary, the optimal setting is

coded : (X1 = +1.5, X3 = +1.3)
uncoded: (OT = 1637.5 degrees, QT = 127.5 degrees)

and finally, including the best setting of the fixed *X*2 factor (carbon concentration CC) of *X*2 = -1 (CC = .5%), we thus have the final, complete recommended optimal settings for all three factors:

coded : (*X*1 = +1.5, *X*2 = -1.0, *X*3 = +1.3)
uncoded: (OT = 1637.5, CC = .7%, QT = 127.5)

If we were to run another experiment, this is the point (based on the data) that we would set oven temperature, carbon concentration, and quench temperature with the hope/goal of achieving 100% acceptable springs.

Options for next step

In practice, we could either

1. collect a single data point (if money and time are an issue) at this recommended setting and see how close to 100% we achieve, or
2. collect two, or preferably three, (if money and time are less of an issue) replicates at the center point (recommended setting).
3. if money and time are not an issue, run a 2^2 full factorial design with center point. The design is centered on the optimal setting ($X1 = +1{,}5$, $X3 = +1.3$) with one overlapping new corner point at ($X1 = +1$, $X3 = +1$) and with new corner points at ($X1$,$X3$) = (+1,+1), (+2,+1), (+1,+1.6), (+2,+1.6). Of these four new corner points, the point (+1,+1) has the advantage that it overlaps with a corner point of the original design.

5. Process Improvement

5.6. Case Studies

Contents

The purpose of this section is to illustrate the analysis of designed experiments with data collected from experiments run at the National Institute of Standards and Technology and SEMATECH. A secondary goal is to give the reader an opportunity to run the analyses in real-time using the Dataplot software package.

5.6.1. Eddy Current Probe Sensitivity Case Study

Analysis of a 2^3 Full Factorial Design

This case study demonstrates the analysis of a 2^3 full factorial design.

The analysis for this case study is based on the EDA approach discussed in an earlier section.

Contents

The case study is divided into the following sections:

5.6.1.1. Background and Data

Background

The data for this case study is a subset of a study performed by Capobianco, Splett, and Iyer. Capobianco was a member of the NIST Electromagnetics Division and Splett and Iyer were members of the NIST Statistical Engineering Division at the time of this study.

The goal of this project is to develop a nondestructive portable device for detecting cracks and fractures in metals. A primary application would be the detection of defects in airplane wings. The internal mechanism of the detector would be for sensing crack-induced changes in the detector's electromagnetic field, which would in turn result in changes in the detector's electromagnetic field, which would in turn result in changes in the impedance level of the detector. This change of impedance is termed "sensitivity" and it is a sub-goal of this experiment to maximize such sensitivity as the detector is moved from an unflawed region to a flawed region on the metal.

Statistical Goals

The case study illustrates the analysis of a 2^3 full factorial experimental design. The specific statistical goals of the experiment are:

1. Determine the important factors that affect sensitivity.
2. Determine the settings that maximize sensitivity.
3. Determine a predicition equation that functionally relates sensitivity to various factors.

Data Used in the Analysis

There were three detector wiring component factors under consideration:

1. $X1$ = Number of wire turns
2. $X2$ = Wire winding distance
3. $X3$ = Wire guage

Since the maximum number of runs that could be afforded timewise and costwise in this experiment was $n = 10$, a 2^3 full factoral experiment (involving $n = 8$ runs) was chosen. With an eye to the usual monotonicity assumption for 2-level factorial designs, the selected settings for the three factors were as follows:

1. $X1$ = Number of wire turns : -1 = 90, +1 = 180
2. $X2$ = Wire winding distance: -1 = 0.38, +1 = 1.14
3. $X3$ = Wire guage : -1 = 40, +1 = 48

The experiment was run with the 8 settings executed in random order. The following data resulted.

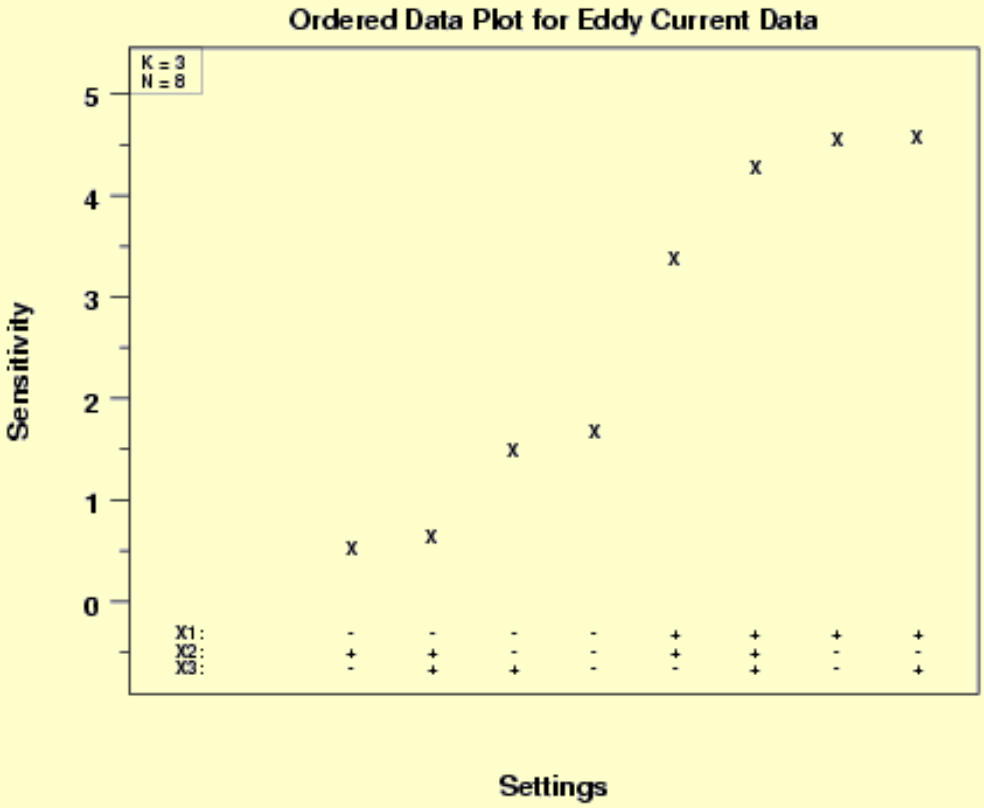
Y Probe Impedance	X1 Number of Turns	X2 Winding Distance	X3 Wire Guage	Run Sequence
1.70	-1	-1	-1	2
4.57	+1	-1	-1	8
0.55	-1	+1	-1	3
3.39	+1	+1	-1	6
1.51	-1	-1	+1	7
4.59	+1	-1	+1	1
0.67	-1	+1	+1	4
4.29	+1	+1	+1	5

Note that the independent variables are coded as +1 and -1. These represent the low and high settings for the levels of each variable. Factorial designs often have 2 levels for each factor (independent variable) with the levels being coded as -1 and +1. This is a scaling of the data that can simplify the analysis. If desired, these scaled values can be converted back to the original units of the data for presentation.

5.6.1.2. Initial Plots/Main Effects

Plot the Data: Ordered Data Plot

The first step in the analysis is to generate an ordered data plot.



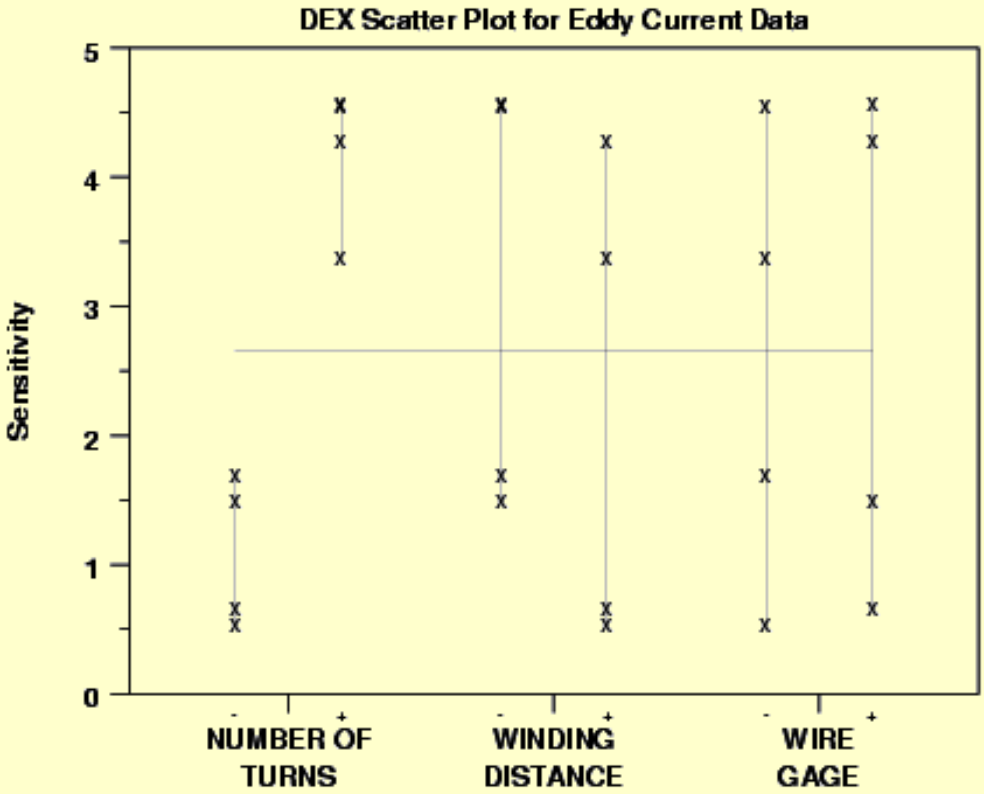
Conclusions from the Ordered Data Plot

We can make the following conclusions based on the ordered data plot.

1. Important Factors: The 4 highest response values have $X1$ = + while the 4 lowest response values have $X1$ = -. This implies factor 1 is the most important factor. When $X1$ = -, the - values of $X2$ are higher than the + values of $X2$. Similarly, when $X1$ = +, the - values of $X2$ are higher than the + values of $X2$. This implies $X2$ is important, but less so than $X1$. There is no clear pattern for $X3$.
2. Best Settings: In this experiment, we are using the device as a detector, and so high sensitivities are desirable. Given this, our first pass at best settings yields ($X1$ = +1, $X2$ = -1, $X3$ = either).

Plot the Data: Dex Scatter Plot

The next step in the analysis is to generate a dex scatter plot.



Conclusions from the DEX Scatter Plot

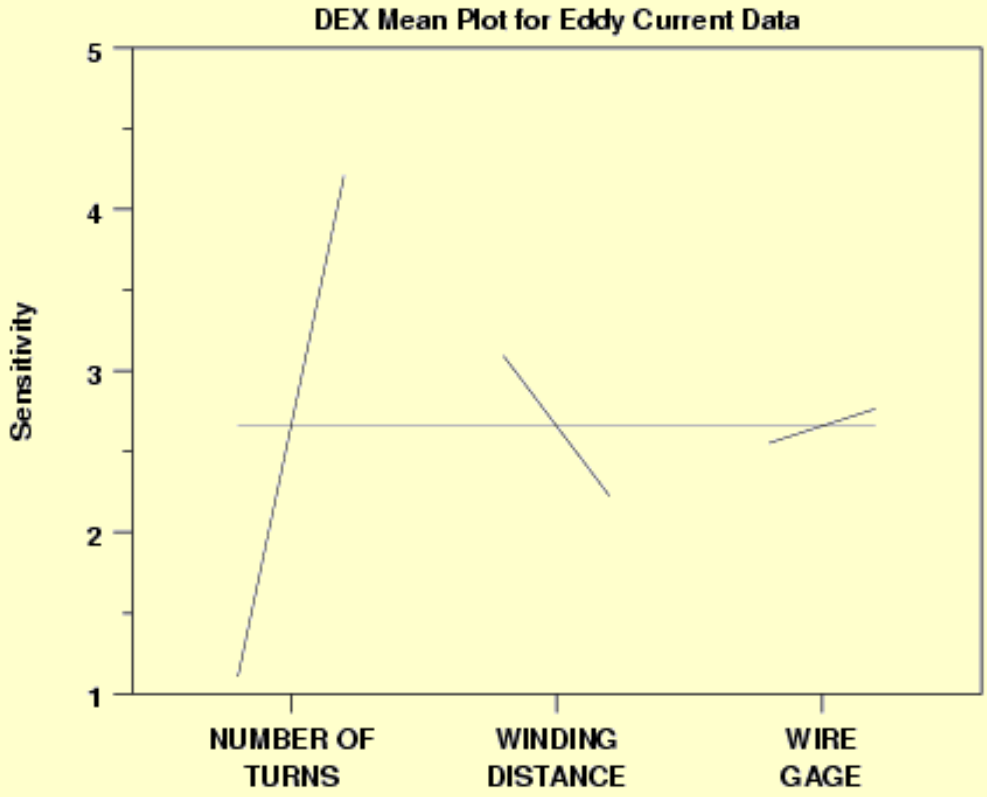
We can make the following conclusions based on the dex scatter plot.

1. Important Factors: Factor 1 (Number of Turns) is clearly important. When $X1 = -1$, all 4 senstivities are low, and when $X1 = +1$, all 4 sensitivities are high. Factor 2 (Winding Distance) is less important. The 4 sensitivities for $X2 = -1$ are slightly higher, as a group, than the 4 sensitivities for $X2 = +1$. Factor 3 (Wire Gage) does not appear to be important at all. The sensitivity is about the same (on the average) regardless of the settings for $X3$.
2. Best Settings: In this experiment, we are using the device as a detector, so high sensitivities are desirable. Given this, our first pass at best settings yields ($X1 = +1$, $X2 = -1$, $X3 =$ either).
3. There does not appear to be any significant outliers.

Check for Main Effects: Dex Mean Plot

One of the primary questions is: what are the most important factors? The ordered data plot and the dex scatter plot provide useful summary plots of the data. Both of these plots indicated that factor *X*1 is clearly important, *X*2 is somewhat important, and *X*3 is probably not important.

The dex mean plot shows the main effects. This provides probably the easiest to interpert indication of the important factors.



Conclusions from the DEX Mean Plot

The dex mean plot (or main effects plot) reaffirms the ordering of the dex scatter plot, but additional information is gleaned because the eyeball distance between the mean values gives an approximation to the least squares estimate of the factor effects.

We can make the following conclusions from the dex mean plot.

1. Important Factors:
   *X*1 (effect = large: about 3 ohms)
   *X*2 (effect = moderate: about -1 ohm)
   *X*3 (effect = small: about 1/4 ohm)
2. Best Settings: As before, choose the factor settings that (on the average) maximize the sensitivity:
   (*X*1,*X*2,*X*3) = (+,-,+)

Comparison of Plots

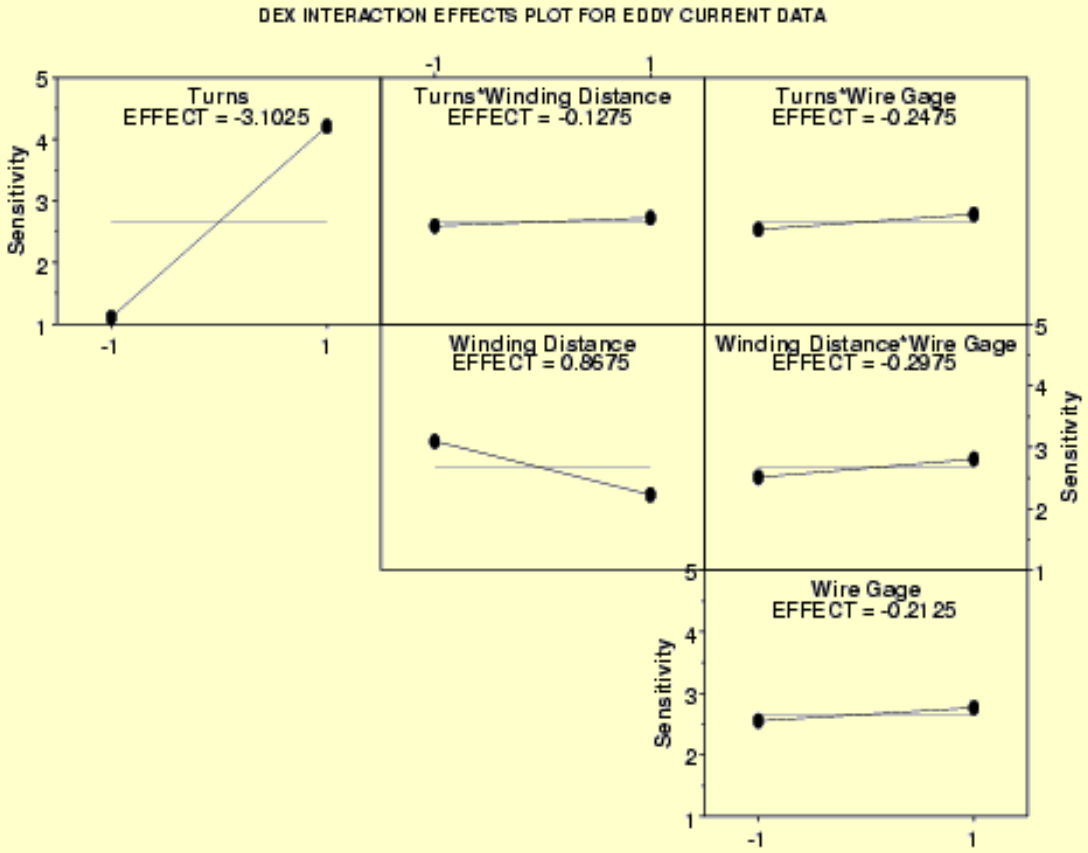
All of these plots are used primarily to detect the most important factors. Because it plots a summary statistic rather than the raw data, the dex mean plot shows the main effects most clearly. However, it is still recommended to generate either the ordered data plot or the dex scatter plot (or both). Since these plot the raw data, they can sometimes reveal features of the data that might be masked by the dex mean plot.

NIST SEMATECH HOME TOOLS & AIDS SEARCH BACK NEXT

5.6.1.3. Interaction Effects

Check for Interaction Effects: Dex Interaction Plot

In addition to the main effects, it is also important to check for interaction effects, especially 2-factor interaction effects. The dex interaction effects plot is an effective tool for this.



Conclusions from the DEX Interaction Effects Plot

We can make the following conclusions from the dex interaction effects plot.

1. Important Factors: Looking for the plots that have the steepest lines (that is, largest effects), we note that:
   - $X1$ (number of turns) is the most important effect: estimated effect = -3.1025;
   - $X2$ (winding distance) is next most important: estimated effect = -.8675;
   - $X3$ (wire gage) is relatively unimportant;
   - All three 2-factor interactions are relatively unimporant.
2. Best Settings: As with the main effects plot, the best settings to maximize the sensitivity are

   $$(X1,X2,X3) = (+1,-1,+1)$$

   but with the $X3$ setting of +1 mattering little.

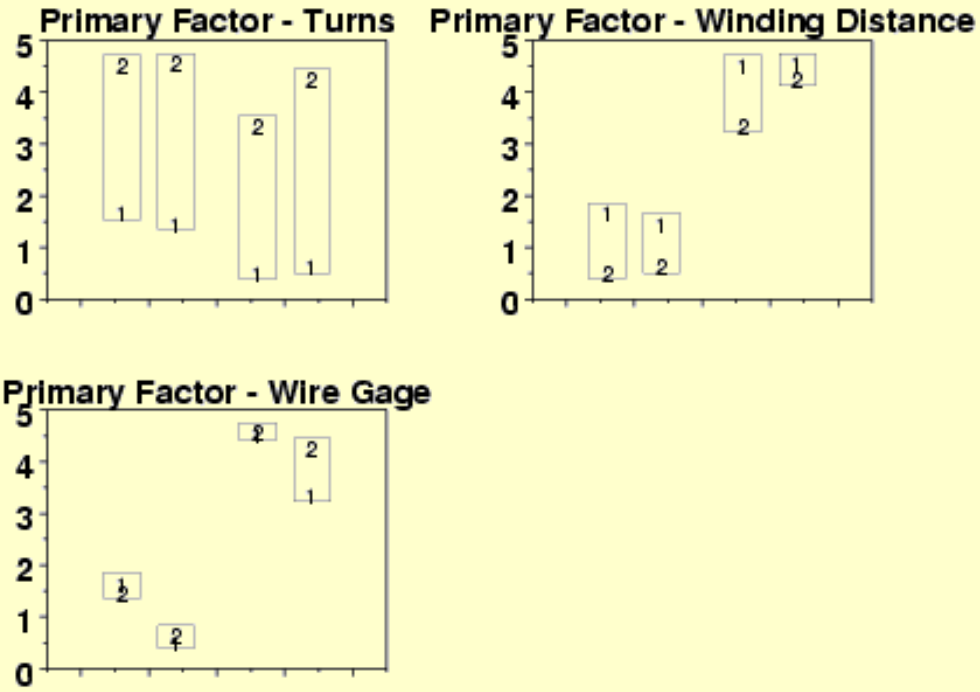
5.6.1.4. Main and Interaction Effects: Block Plots

Block Plots

Block plots are a useful adjunct to the dex mean plot and the dex interaction effects plot to confirm the importance of factors, to establish the robustness of main effect conclusions, and to determine the existence of interactions. Specifically,

1. The first plot below answers the question: Is factor 1 important? If factor 1 is important, is this importance robust over all 4 settings of $X2$ and $X3$?
2. The second plot below answers the question: Is factor 2 important? If factor 2 is important, is this importance robust over all 4 settings of $X1$ and $X3$?
3. The third plot below answers the question: Is factor 3 important? If factor 3 is important, is this importance robust over all 4 settings of $X1$ and $X2$?

For block plots, it is the height of the bars that is important, not the relative positioning of each bar. Hence we focus on the size and internals of the blocks, not "where" the blocks are one relative to another.



Conclusions from the Block Plots

It is recalled that the block plot will access factor importance by the degree of consistency (robustness) of the factor effect over a variety of conditions. In this light, we can make the following conclusions from the block plots.

1. Relative Importance of Factors: All of the bar heights in plot 1 (turns) are greater than the bar heights in plots 2 and 3. Hence, factor 1 is more important than factors 2 and 3.
2. Statistical Significance: In plot 1, looking at the levels within each bar, we note that the response for level 2 is higher than level 1 in each of the 4 bars. By chance, this happens with probability $1/(2^4) = 1/16 = 6.25\%$. Hence, factor 1 is near-statistically significant at the 5% level. Similarly, for plot 2, level 1 is greater than level 2 for all 4 bars. Hence, factor 2 is near-statistically significant. For factor 3, there is not consistent ordering within all 4 bars and hence factor 3 is not statistically significant. Rigorously speaking then, factors 1 and 2 are not statistically significant (since 6.25% is not < 5%); on the other hand such near-significance is suggestive to the analyst that such factors may in fact be important, and hence warrant further attention.

   Note that the usual method for determining statistical significance is to perform an analysis of variance (ANOVA). ANOVA is based on normality assumptions. If these normality assumptions are in fact valid, then ANOVA methods are the most powerful method for determining statistical signficance. The advantage of the block plot method is that it is based on less rigorous assumptions than ANOVA. At an exploratory stage, it is useful to know that our conclusions regarding important factors are valid under a wide range of assumptions.
3. Interactions: For factor 1, the 4 bars do not change height in any systematic way and hence there is no evidence of $X1$ interacting with either $X2$ or $X3$. Similarly, there is no evidence of interactions for factor 2.

5.6.1.5. Estimate Main and Interaction Effects

Effects Estimation

Although the effect estimates were given on the dex interaction plot on a previous page, they can also be estimated quantitatively.

The full model for the 2^3 factorial design is

$$Y = \mu + 0.5 * (\beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{12} X_1 X_2 + \beta_{13} X_1 X_3 + \beta_{23} X_2 X_3 + \beta_{123} X_1 X_2 X_3) + \epsilon$$

Data from factorial designs with two levels can be analyzed using the Yates technique, which is described in Box, Hunter, and Hunter. The Yates technique utilizes the special structure of these designs to simplify the computation and presentation of the fit.

Dataplot Output

Dataplot generated the following output for the Yates analysis.

```
(NOTE--DATA MUST BE IN STANDARD ORDER)
NUMBER OF OBSERVATIONS              =        8
NUMBER OF FACTORS                   =        3
NO REPLICATION CASE

PSEUDO-REPLICATION STAND. DEV.      =    0.20152531564E+00
PSEUDO-DEGREES OF FREEDOM           =        1
(THE PSEUDO-REP. STAND. DEV. ASSUMES ALL
3, 4, 5, ...-TERM INTERACTIONS ARE NOT REAL,
BUT MANIFESTATIONS OF RANDOM ERROR)

STANDARD DEVIATION OF A COEF.       =    0.14249992371E+00
(BASED ON PSEUDO-REP. ST. DEV.)

GRAND MEAN                          =    0.26587500572E+01
GRAND STANDARD DEVIATION            =    0.17410624027E+01

99% CONFIDENCE LIMITS (+-)          =    0.90710897446E+01
95% CONFIDENCE LIMITS (+-)          =    0.18106349707E+01
```

```
99.5% POINT OF T DISTRIBUTION    =    0.63656803131E+02
97.5% POINT OF T DISTRIBUTION    =    0.12706216812E+02

IDENTIFIER    EFFECT        T VALUE      RESSD:     RESSD:
                                         MEAN +     MEAN +
                                         TERM    CUM TERMS
-------------------------------------------------------------
   MEAN       2.65875                    1.74106    1.74106
      1       3.10250         21.8*      0.57272    0.57272
      2      -0.86750         -6.1       1.81264    0.30429
     23       0.29750          2.1       1.87270    0.26737
     13       0.24750          1.7       1.87513    0.23341
      3       0.21250          1.5       1.87656    0.19121
    123       0.14250          1.0       1.87876    0.18031
     12       0.12750          0.9       1.87912    0.00000
```

Description of Yates Output

In fitting 2-level factorial designs, Dataplot takes advantage of the special structure of these designs in computing the fit and printing the results. Specifically, the main effects and interaction effects are printed in sorted order from most significant to least significant. It also prints the *t*-value for the term and the residual standard deviation obtained by fitting the model with that term and the mean (the column labeled RESSD MEAN + TERM), and for the model with that term, the mean, and all other terms that are more statistically significant (the column labeled RESSD MEAN + CUM TERMS).

Of the five columns of output, the most important are the first (which is the identifier), the second (the least squares estimated effect = the difference of means), and the last (the residuals standard deviation for the cumulative model, which will be discussed in more detail in the next section).

Conclusions

In summary, the Yates analysis provides us with the following ranked list of important factors.

1. *X*1 (Number of Turns): effect estimate = 3.1025 ohms
2. *X*2 (Winding Distance): effect estimate = -0.8675 ohms
3. *X*2**X*3 (Winding Distance with Wire Guage): effect estimate = 0.2975 ohms
4. *X*1**X*3 (Number of Turns with Wire Guage): effect estimate = 0.2475 ohms
5. *X*3 (Wire Guage): effect estimate = 0.2125 ohms
6. *X*1**X*2**X*3 (Number of Turns with Winding Distance with Wire Guage): effect estimate = 0.1425 ohms
7. *X*1**X*2 (Number of Turns with Winding Distance): effect estimate = 0.1275 ohms

5.6.1.6. Modeling and Prediction Equations

Parameter Estimates Don't Change as Additional Terms Added

In most cases of least squares fitting, the model coefficient estimates for previously added terms change depending on what was successively added. For example, the estimate for the $X1$ coefficient might change depending on whether or not an $X2$ term was included in the model. This is **not** the case when the design is orthogonal, as is this 2^3 full factorial design. In such a case, the estimates for the previously included terms do not change as additional terms are added. This means the ranked list of effect estimates in the Yates table simultaneously serves as the least squares coefficient estimates for progressively more complicated models.

The last column of the Yates table gave the residual standard deviation for 8 possible models, each one progressively more complicated.

Default Model: Grand Mean

At the top of the Yates table, if none of the factors are important, the prediction equation defaults to the mean of all the response values (the overall or grand mean). That is,

$$\hat{Y} = 2.65875$$

From the last column of the Yates table, it can be seen that this simplest of all models has a residual standard deviation (a measure of goodness of fit) of 1.74106 ohms. Finding a good-fitting model was **not** one of the stated goals of this experiment, but the determination of a good-fitting model is "free" along with the rest of the analysis, and so it is included.

Conclusions

From the last column of the Yates table, we can summarize the following prediction equations:

- $\hat{Y} = 2.65875$

  has a residual standard deviation of 1.74106 ohms.
- $\hat{Y} = 2.65875 + 0.5(3.1025X1)$

  has a residual standard deviation of 0.57272 ohms.
- $\hat{Y} = 2.65875 + 0.5(3.1025X1 - 0.8675X2)$

  has a residual standard deviation of 0.30429 ohms.
- $\hat{Y} = 2.65875 + 0.5(3.1025X1 - 0.8675X2 + 0.2975X2 * X3)$

has a residual standard deviation of 0.29750 ohms.

- The remaining models can be listed in a similar fashion. Note that the full model provides a perfect fit to the data.

5.6.1.7. Intermediate Conclusions

Important Factors

Taking stock from all of the graphical and quantitative analyses of the previous sections, we conclude that $X1$ (= number of turns) is the most important engineering factor affecting sensitivity, followed by $X2$ (= wire distance) as next in importance, followed then by some less important interactions and $X3$ (= wire guage).

Best Settings

Also, from the various analyses, we conclude that the best design settings (on the average) for a high-sensitivity detector are

$$(X1,X2,X3) = (+,-,+)$$

that is

number of turns = 180,
winding distance = 0.38, and
wire guage = 48.

Can We Extract More From the Data?

Thus, in a very real sense, the analysis is complete. We have achieved the two most important stated goals of the experiment:

1. gaining insight into the most important factors, and
2. ascertaining the optimal production settings.

On the other hand, more information can be squeezed from the data, and that is what this section and the remaining sections address.

1. First of all, we focus on the problem of taking the ranked list of factors and objectively ascertaining which factors are "important" versus "unimportant".
2. In a parallel fashion, we use the subset of important factors derived above to form a "final" prediction equation that is good (that is, having a sufficiently small residual standard deviation) while being parsimonious (having a small number of terms), compared to the full model, which is perfect (having a residual standard deviation = 0, that is, the predicted values = the raw data), but is unduly complicated (consisting of a constant + 7 terms).

5.6.1.8. Important Factors and Parsimonious Prediction

Identify Important Factors

The two problems discussed in the previous section (important factors and a parsimonious model) will be handled in parallel since determination of one yields the other. In regard to the "important factors", our immediate goal is to take the full subset of 7 main effects and interactions and extract a subset that we will declare as "important", with the complementary subset being "unimportant". Seven criteria are discussed in detail under the Yates analysis in the EDA Chapter (Chapter 1). The relevant criteria will be applied here. These criteria are not all equally important, nor will they yield identical subsets, in which case a consensus subset or a weighted consensus subset must be extracted.

Criteria for Including Terms in the Model

The criteria that we can use in determining whether to keep a factor in the model can be summarized as follows.

1. Effects: Engineering Significance
2. Effects: 90% Numerical Significance
3. Effects: Statistical Significance
4. Effects: Half-normal Probability Plots
5. Averages: Youden Plot

The first four criteria focus on effect estimates with three numerical criteria and one graphical criterion. The fifth criterion focuses on averages. We discuss each of these criteria in detail in the following sections.

The last section summarizes the conclusions based on all of the criteria.

Effects: Engineering Significance

The minimum engineering significant difference is defined as

$$|\hat{\beta}_i| > \Delta$$

where $|\hat{\beta}_i|$ is the absolute value of the parameter estimate (i.e., the effect) and Δ is the minimum engineering significant difference. That is, declare a factor as "important" if the effect is greater than some a priori declared engineering difference. We use a rough rule-of-thumb of keeping only those factors whose effect is greater than 10% of the current production average. In this case, let's say that the average detector has a sensitivity of 2.5 ohms. This suggests that we would declare all factors whose effect is greater than 10% of 2.5 ohms = 0.25 ohm to be significant from an engineering point of view.

Based on this minimum engineering-significant-difference criterion, we conclude to keep two terms: $X1$ (3.10250) and $X2$ (-.86750).

Effects: 90% Numerical Significance

The 90% numerical significance criterion is defined as

$$|\hat{\beta}_i| > (max|\hat{\beta}_i|)/10$$

That is, declare a factor as important if it exceeds 10% of the largest effect. For the current case study, the largest effect is from factor 1 (3.10250 ohms), and so 10% of that is 0.31 ohms. This suggests keeping all factors whose effects exceed 0.31 ohms.

Based on the 90% numerical criterion, we thus conclude to keep two terms: *X*1 (3.10250) and *X*2 (-.86750). The *X*2**X*3 term, (0.29750), is just under the cutoff.

Effects: Statistical Significance

Statistical significance is defined as

$$|\hat{\beta}_i| > 2sd(\hat{\beta}_i) = 2(\frac{2\sigma}{\sqrt{n}})$$

That is, declare a factor as "important" if its effect is more than 2 standard deviations away from 0 (0, by definition, meaning "no effect"). The difficulty with this is that in order to invoke this we need the σ = the standard deviation of an observation.

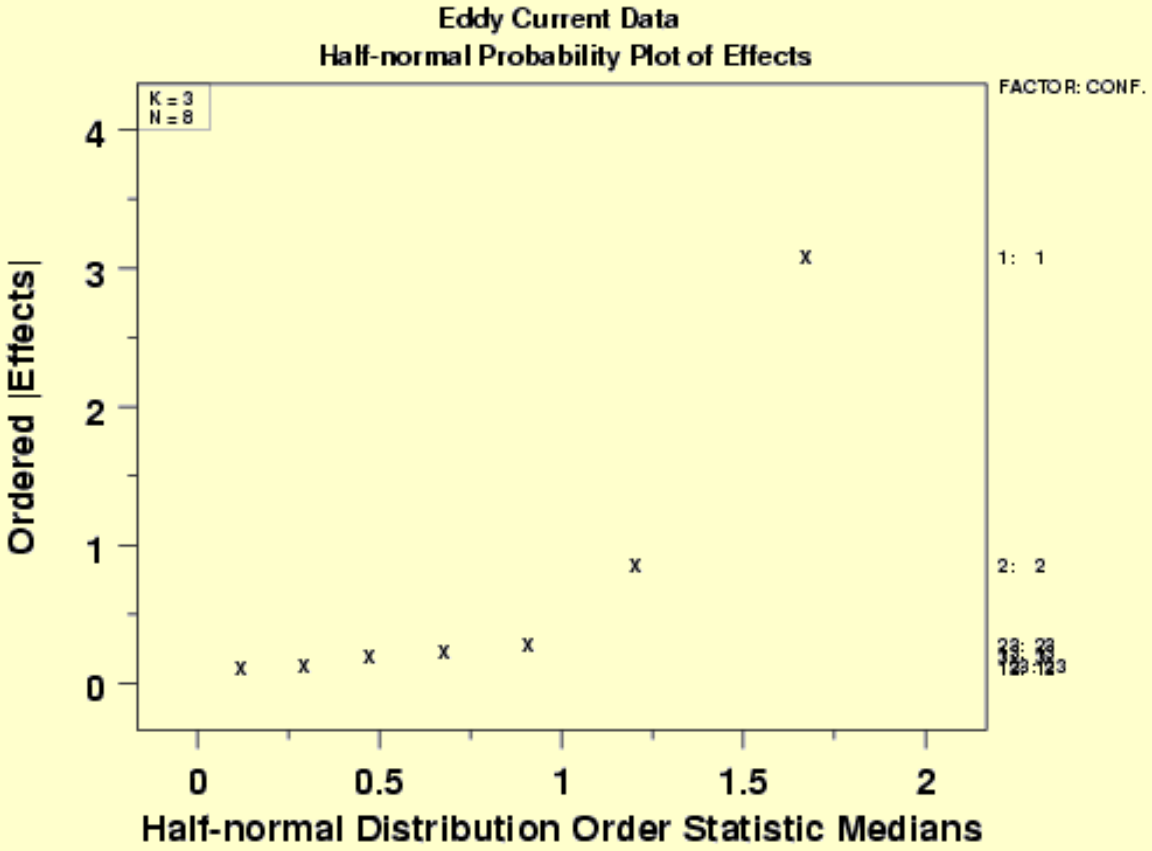
For the current case study, ignoring 3-factor interactions and higher-order interactions leads to an estimate of σ based on omitting only a single term: the *X*1***X*2***X*3 interaction.

Thus for this current case study, if one assumes that the 3-factor interaction is nil and hence represents a single drawing from a population centered at zero, an estimate of the standard deviation of an effect is simply the estimate of the interaction effect (0.1425). Two such effect standard deviations is 0.2850. This rule becomes to keep all $|\hat{\beta}_i| > 0.2850$. This results in keeping three terms: *X*1 (3.10250), *X*2 (-.86750), and *X*1***X*2 (.29750).

Effects: Probability Plots

The half-normal probability plot can be used to identify important factors.

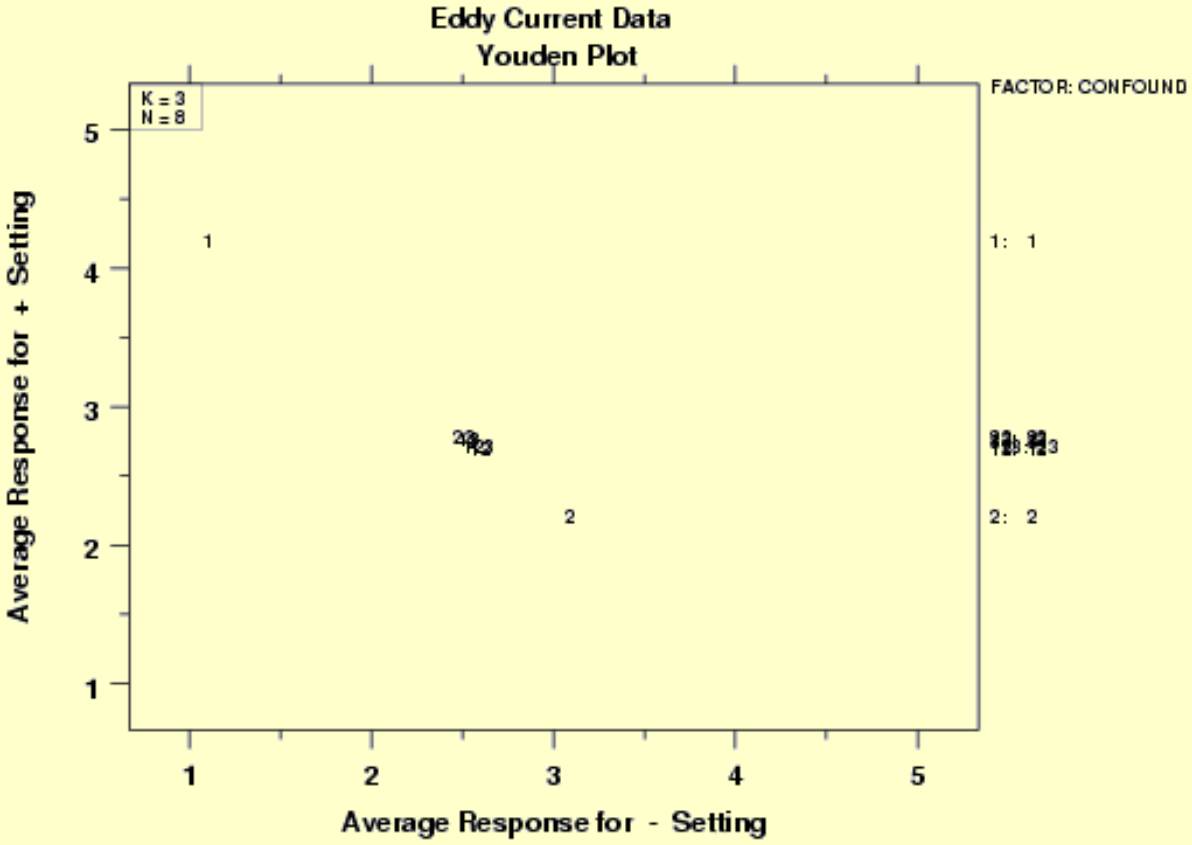
The following plot shows the half-normal probability plot of the absolute value of the effects.



The half-normal probablity plot clearly shows two factors displaced off the line, and we see that those two factors are factor 1 and factor 2. In conclusion, keep two factors: *X*1 (3.10250) and *X*2 (-.86750).

Effects:
Youden Plot

A dex Youden plot can be used in the following way. Keep a factor as "important" if it is displaced away from the central-tendency bunch in a Youden plot of high and low averages.



For the case study at hand, the Youden plot clearly shows a cluster of points near the grand average (2.65875) with two displaced points above (factor 1) and below (factor 2). Based on the Youden plot, we thus conclude to keep two factors: *X*1 (3.10250) and *X*2 (-.86750).

Conclusions

In summary, the criterion for specifying "important" factors yielded the following:

1. Effects, Engineering Significant: *X*1 *X*2
2. Effects, Numerically Significant: *X*1 *X*2 (*X*2**X*3 is borderline)
3. Effects, Statistically Significant: *X*1 *X*2 *X*2**X*3
4. Effects, Half-Normal Probability Plot: *X*1 *X*2
5. Averages, Youden Plot: *X*1 *X*2

All the criteria select *X*1 and *X*2. One also includes the *X*2**X*3 interaction term (and it is borderline for another criteria).

We thus declare the following consensus:

1. Important Factors: *X*1 and *X*2
2. Parsimonious Prediction Equation:

$$\hat{Y} = 2.65875 + 0.5(3.1025X1 - 0.8675X2)$$

(with a residual standard deviation of .30429 ohms)

5.6.1.9. Validate the Fitted Model

Model Validation

In the Important Factors and Parsimonious Prediction section, we came to the following model

$$Y = 2.65875 + 0.5[3.10250X_1 - 0.86750X_2] + \epsilon$$

The residual standard deviation for this model is 0.30429.

The next step is to validate the model. The primary method of model validation is graphical residual analysis; that is, through an assortment of plots of the differences between the observed data Y and the predicted value $\hat{Y}$ from the model. For example, the design point (-1,-1,-1) has an observed data point (from the Background and data section) of Y = 1.70, while the predicted value from the above fitted model for this design point is

$$\hat{Y} = 2.65875 + 0.5(3.1025(-1) - 0.8675(-1)) = 1.54125$$

which leads to the residual 0.15875.

Table of Residuals

If the model fits well, $\hat{Y}$ should be near *Y* for all 8 design points. Hence the 8 residuals should all be near zero. The 8 predicted values and residuals for the model with these data are:

X1	X2	X3	Observed	Predicted	Residual
-1	-1	-1	1.70	1.54125	0.15875
+1	-1	-1	4.57	4.64375	-0.07375
-1	+1	-1	0.55	0.67375	-0.12375
+1	+1	-1	3.39	3.77625	-0.38625
-1	-1	+1	1.51	1.54125	-0.03125
+1	-1	+1	4.59	4.64375	-0.05375
-1	+1	+1	0.67	0.67375	-0.00375
+1	+1	+1	4.29	3.77625	0.51375

Residual Standard Deviation

What is the magnitude of the typical residual? There are several ways to compute this, but the statistically optimal measure is the residual standard deviation:

$$s_{res} = \sqrt{\frac{\sum_{i=1}^{N} r_i^2}{N - P}}$$

with r_i denoting the *i*th residual, $N = 8$ is the number of observations, and $P = 3$ is the number of fitted parameters. From the Yates table, the residual standard deviation is 0.30429.

How Should Residuals Behave?

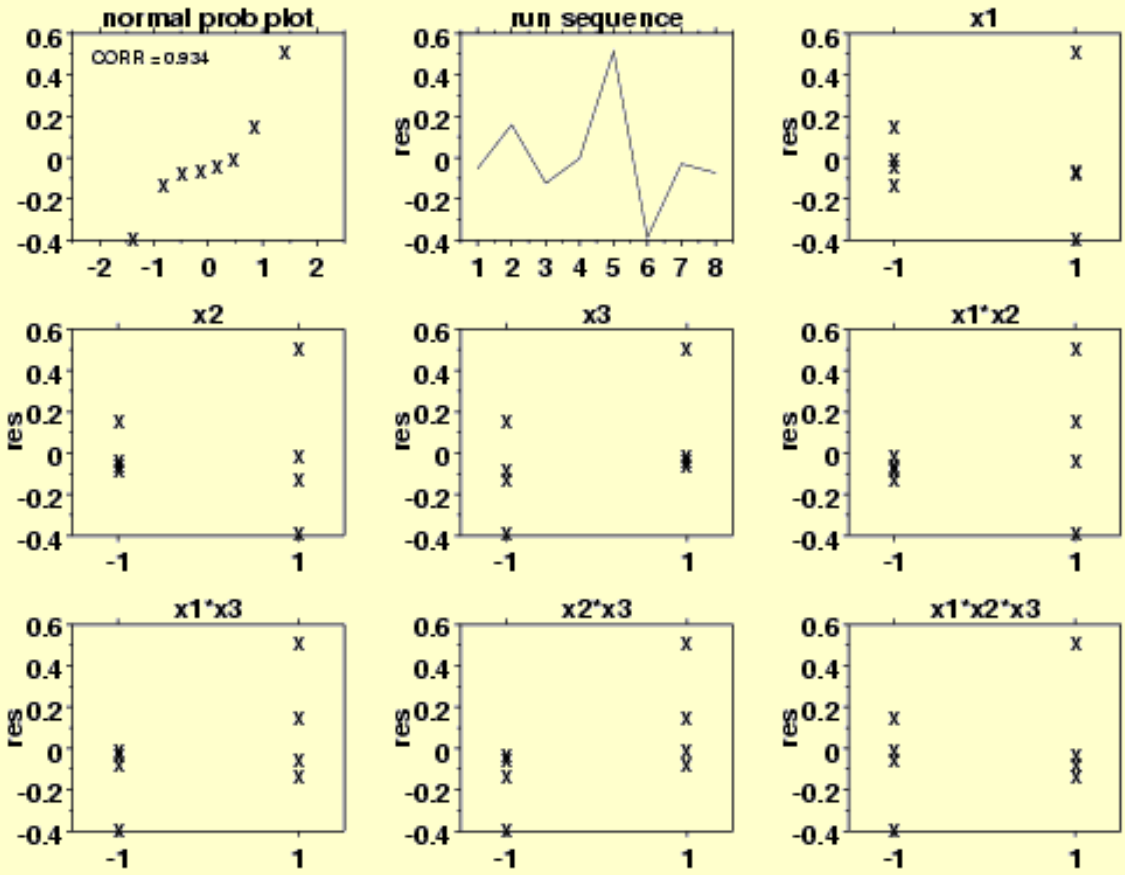
If the prediction equation is adequate, the residuals from that equation should behave like random drawings (typically from an approximately normal distribution), and should, since presumably random, have no structural relationship with any factor. This includes any and all potential terms (*X*1, *X*2, *X*3, *X*1**X*2, *X*1**X*3, *X*2**X*3, *X*1**X*2**X*3).

Further, if the model is adequate and complete, the residuals should have no structural relationship with **any** other variables that may have been recorded. In particular, this includes the run sequence (time), which is really serving as a surrogate for any physical or environmental variable correlated with time. Ideally, all such residual scatter plots should appear structureless. Any scatter plot that exhibits structure suggests that the factor should have been formally included as part of the prediction equation.

Validating the prediction equation thus means that we do a final check as to whether any other variables may have been inadvertently left out of the prediction equation, including variables drifting with time.

The graphical residual analysis thus consists of scatter plots of the residuals versus all 3 factors and 4 interactions (all such plots should be structureless), a scatter plot of the residuals versus run sequence (which also should be structureless), and a normal probability plot of the residuals (which should be near linear). We present such plots below.

Residual Plots



The first plot is a normal probability plot of the residuals. The second plot is a run sequence plot of the residuals. The remaining plots are plots of the residuals against each of the factors and each of the interaction terms.

Conclusions We make the following conclusions based on the above plots.

1. Main Effects and Interactions: The *X*1 and *X*2 scatter plots are "flat" (as they must be since *X*1 and *X*2 were explicitly included in the model). The *X*3 plot shows some structure as does the *X*1**X*3, the *X*2**X*3, and the *X*1**X*2**X*3 plots. The *X*1**X*2 plot shows little structure. The net effect is that the relative ordering of these scatter plots is very much in agreement (again, as it must be) with the relative ordering of the "unimportant" factors given on lines 3-7 of the Yates table. From the Yates table and the *X*2**X*3 plot, it is seen that the next most influential term to be added to the model would be *X*2**X*3. In effect, these plots offer a higher-resolution confirmation of the ordering that was in the Yates table. On the other hand, none of these other factors "passed" the criteria given in the previous section, and so these factors, suggestively influential as they might be, are still not influential enough to be added to the model.
2. Time Drift: The run sequence scatter plot is random. Hence there does not appear to be a drift either from time, or from any factor (e.g., temperature, humidity, pressure, etc.) possibly correlated with time.
3. Normality: The normal probability plot of the 8 residuals has some curvature, which suggests that additional terms might be added. On the other hand, the correlation coefficient of the 8 ordered residuals and the 8 theoretical normal N(0,1) order statistic medians (which define the two axes of the plot) has the value 0.934, which is well within acceptable (5%) limits of the normal probability plot correlation coefficient test for normality. Thus, the plot is not so non-linear as to reject normality.

In summary, therefore, we accept the model

$$Y = 2.65875 + 0.5\left[3.10250X_1 - 0.86750X_2\right] + \epsilon$$

as a parsimonious, but good, representation of the sensitivity phenomenon under study.

5.6.1.10. Using the Fitted Model

Model Provides Additional Insight

Although deriving the fitted model was not the primary purpose of the study, it does have two benefits in terms of additional insight:

1. Global prediction
2. Global determination of best settings

Global Prediction

How does one predict the response at points other than those used in the experiment? The prediction equation yields good results at the 8 combinations of coded -1 and +1 values for the three factors:

1. $X1$ = Number of turns = 90 and 180
2. $X2$ = Winding distance = .38 and 1.14
3. $X3$ = Wire gauge = 40 and 48

What, however, would one expect the detector to yield at target settings of, say,

1. Number of turns = 150
2. Winding distance = .50
3. Wire guage = 46

Based on the fitted equation, we first translate the target values into coded target values as follows:

coded target = -1 + 2*(target-low)/(high-low)

Hence the coded target values are

1. $X1$ = -1 + 2*(150-90)/(180-90) = 0.333333
2. $X2$ = -1 + 2*(.50-.38)/(1.14-.38) = -0.684211
3. $X3$ = -1 + 2*(46-40)/(48-40) = 0.5000

Thus the raw data

(Number of turns,Winding distance,Wire guage) = (150,0.50,46)

translates into the coded

($X1$,$X2$,$X3$) = (0.333333,-0.684211,0.50000)

on the -1 to +1 scale.

Inserting these coded values into the fitted equation yields, as desired, a predicted value of

$\hat{Y}$ = 2.65875 + 0.5(3.10250*(.333333) - 0.86750*(-.684211)) = 3.47261

The above procedure can be carried out for any values of turns, distance, and gauge. This is subject to the usual cautions that equations that are good near the data point vertices may not necessarily be good everywhere in the factor space. Interpolation is a bit safer than extrapolation, but it is not guaranteed to provide good results, of course. One would feel more comfortable about interpolation (as in our example) if additional data had been collected at the center point and the center point data turned out to be in good agreement with predicted values at the center

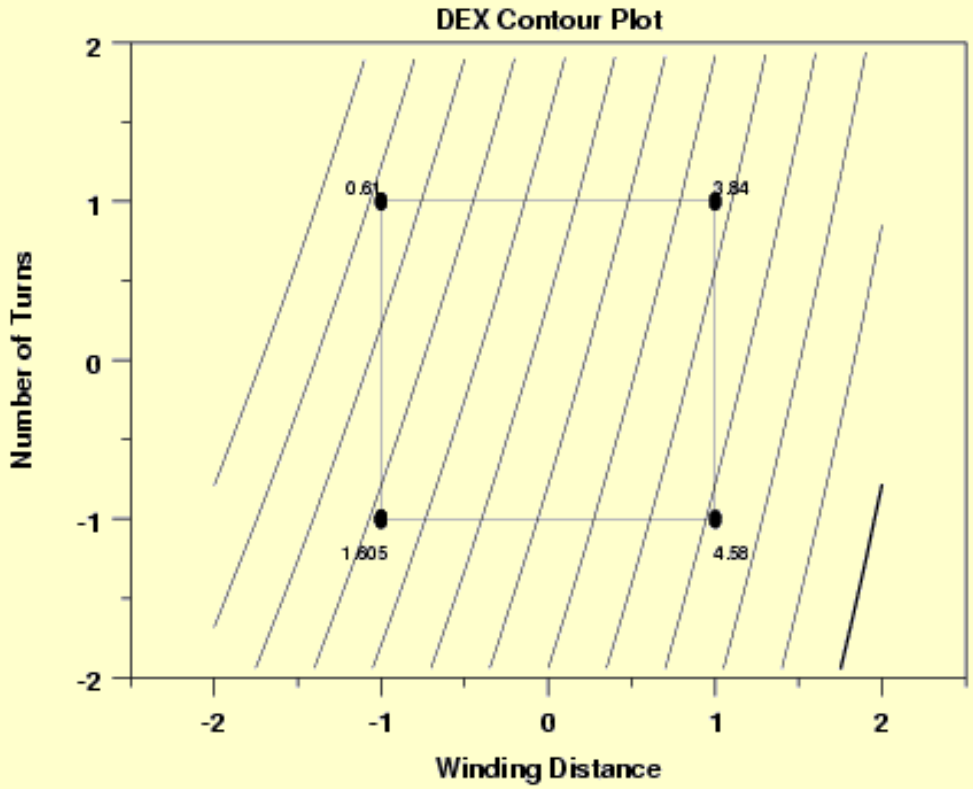
point based on the fitted model. In our case, we had no such data and so the sobering truth is that the user of the equation is assuming something in which the data set as given is not capable of suggesting one way or the other. Given that assumption, we have demonstrated how one may cautiously but insightfully generate predicted values that go well beyond our limited original data set of 8 points.

Global Determination of Best Settings

In order to determine the best settings for the factors, we can use a dex contour plot. The dex contour plot is generated for the two most significant factors and shows the value of the response variable at the vertices (i.e, the -1 and +1 settings for the factor variables) and indicates the direction that maximizes (or minimizes) the response variable. If you have more than two significant factors, you can generate a series of dex contour plots with each one using two of the important factors.

DEX Contour Plot

The following is the dex contour plot of the number of turns and the winding distance.



The maximum value of the response variable (eddy current) corresponds to $X1$ (number of turns) equal to -1 and $X2$ (winding distance) equal to +1. The thickened line in the contour plot corresponds to the direction that maximizes the response variable. This information can be used in planning the next phase of the experiment.

5.6.1.11. Conclusions and Next Step

Conclusions

The goals of this case study were:

1. Determine the most important factors.
2. Determine the best settings for the factors.
3. Determine a good prediction equation for the data.

The various plots and Yates analysis showed that the number of turns (*X*1) and the winding distance (*X*2) were the most important factors and a good prediction equation for the data is:

$$Y = 2.65875 + 0.5[3.10250X_1 - 0.86750X_2]$$

The dex contour plot gave us the best settings for the factors (*X*1 = -1 and *X*2 = 1).

Next Step

Full and fractional designs are typically used to identify the most important factors. In some applications, this is sufficient and no further experimentation is performed. In other applications, it is desired to maximize (or minimize) the response variable. This typically involves the use of response surface designs. The dex contour plot can provide guidance on the settings to use for the factor variables in this next phase of the experiment.

This is a common sequence for designed experiments in engineering and scientific applications. Note the iterative nature of this approach. That is, you typically do not design one large experiment to answer all your questions. Rather, you run a series of smaller experiments. The initial experiment or experiments are used to identify the important factors. Once these factors are identified, follow-up experiments can be run to fine tune the optimal settings (in terms of maximizing/minimizing the response variable) for these most important factors.

For this particular case study, a response surface design was not used.

5.6.1.12. Work This Example Yourself

View Dataplot Macro for this Case Study

This page allows you to repeat the analysis outlined in the case study description on the previous page using Dataplot. It is required that you have already downloaded and installed Dataplot and configured your browser to run Dataplot. Output from each analysis step below will be displayed in one or more of the Dataplot windows. The four main windows are the Output window, the Graphics window, the Command History window, and the Data Sheet window. Across the top of the main windows are menus for executing Dataplot commands. Across the bottom is a command entry window where commands can be typed in.

Data Analysis Steps	Results and Conclusions
Click on the links below to start Dataplot and run this case study yourself. Each step may use results from previous steps, so please be patient. Wait until the software verifies that the current step is complete before clicking on the next step.	*The links in this column will connect you with more detailed information about each analysis step from the case study description.*
1. Get set up and started. 1. Read in the data.	1. You have read 4 columns of numbers into Dataplot: variables Y, X1, X2, and X3.
2. Plot the main effects. 1. Ordered data plot. 2. Dex scatter plot. 3. Dex mean plot.	1. Ordered data plot shows factor 1 clearly important, factor 2 somewhat important. 2. Dex scatter plot shows significant differences for factors 1 and 2. 3. Dex mean plot shows significant differences in means for factors 1 and 2.

3. Plots for interaction effects 1. Generate a dex interaction effects matrix plot.	1. The dex interaction effects matrix plot does not show any major interaction effects.
4. Block plots for main and interaction effects 1. Generate block plots.	1. The block plots show that the factor 1 and factor 2 effects are consistent over all combinations of the other factors.
5. Estimate main and interaction effects 1. Perform a Yates fit to estimate the main effects and interaction effects.	1. The Yates analysis shows that the factor 1 and factor 2 main effects are significant, and the interaction for factors 2 and 3 is at the boundary of statistical significance.
6. Model selection 1. Generate half-normal probability plots of the effects. 2. Generate a Youden plot of the effects.	1. The probability plot indicates that the model should include main effects for factors 1 and 2. 2. The Youden plot indicates that the model should include main effects for factors 1 and 2.
7. Model validation 1. Compute residuals and predicted values from the partial model suggested by the Yates analysis. 2. Generate residual plots to validate the model.	1. Check the link for the values of the residual and predicted values. 2. The residual plots do not indicate any major problems with the model using main effects for factors 1 and 2.

8. Dex contour plot 1. Generate a dex contour plot using factors 1 and 2.	1. The dex contour plot shows X1 = -1 and X2 = +1 to be the best settings.

5.6.2. Sonoluminescent Light Intensity Case Study

Analysis of a 2^{7-3} Fractional Factorial Design

This case study demonstrates the analysis of a 2^{7-3} fractional factorial design.

This case study is a Dataplot analysis of the optimization of sonoluminescent light intensity.

The case study is based on the EDA approach to experimental design discussed in an earlier section.

Contents

The case study is divided into the following sections:

5.6.2.1. Background and Data

Background and Motivation

Sonoluminescence is the process of turning sound energy into light. An ultrasonic horn is used to resonate a bubble of air in a medium, usually water. The bubble is ultrasonically compressed and then collapses to light-emitting plasma.

In the general physics community, sonoluminescence studies are being carried out to characterize it, to understand it, and to uncover its practical uses. An unanswered question in the community is whether sonoluminescence may be used for cold fusion.

NIST's motive for sonoluminescent investigations is to assess its suitability for the dissolution of physical samples, which is needed in the production of homogeneous Standard Reference Materials (SRMs). It is believed that maximal dissolution coincides with maximal energy and maximal light intensity. The ultimate motivation for striving for maximal dissolution is that this allows improved determination of alpha-and beta-emitting radionuclides in such samples.

The objectives of the NIST experiment were to determine the important factors that affect sonoluminescent light intensity and to ascertain optimal settings of such factors that will predictably achieve high intensities. An original list of 49 factors was reduced, based on physics reasons, to the following seven factors: molarity (amount of solute), solute type, pH, gas type in the water, water depth, horn depth, and flask clamping.

Time restrictions caused the experiment to be about one month, which in turn translated into an upper limit of roughly 20 runs. A 7-factor, 2-level fractional factorial design (Resolution IV) was constructed and run. The factor level settings are given below.

Eva Wilcox and Ken Inn of the NIST Physics Laboratory conducted this experiment during 1999. Jim Filliben of the NIST Statistical Engineering Division performed the analysis of the experimental data.

Response Variable, Factor Variables, and Factor-Level Settings

This experiment utilizes the following response and factor variables.

1. Response Variable (Y) = The sonoluminescent light intensity.
2. Factor 1 ($X1$) = Molarity (amount of Solute). The coding is -1 for 0.10 mol and +1 for 0.33 mol.
3. Factor 2 ($X2$) = Solute type. The coding is -1 for sugar and +1 for glycerol.
4. Factor 3 ($X3$) = pH. The coding is -1 for 3 and +1 for 11.
5. Factor 4 ($X4$) = Gas type in water. The coding is -1 for helium and +1 for air.
6. Factor 5 ($X5$) = Water depth. The coding is -1 for half and +1 for full.
7. Factor 6 ($X6$) = Horn depth. The coding is -1 for 5 mm and +1 for 10 mm.
8. Factor 7 ($X7$) = Flask clamping. The coding is -1 for unclamped and +1 for clamped.

This data set has 16 observations. It is a 2^{7-3} design with no center points.

Goal of the Experiment

This case study demonstrates the analysis of a 2^{7-3} fractional factorial experimental design. The goals of this case study are:

1. Determine the important factors that affect the sonoluminescent light intensity. Specifically, we are trying to maximize this intensity.
2. Determine the best settings of the seven factors so as to maximize the sonoluminescent light intensity.

Data Used in the Analysis

The following are the data used for this analysis. This data set is given in Yates order.

Y Light Intensity	X1 Molarity	X2 Solute type	X3 pH	X4 Gas Type	X5 Water Depth	X6 Horn Depth	X7 Flask Clamping
80.6	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
66.1	1.0	-1.0	-1.0	-1.0	-1.0	1.0	1.0
59.1	-1.0	1.0	-1.0	-1.0	1.0	-1.0	1.0
68.9	1.0	1.0	-1.0	-1.0	1.0	1.0	-1.0
75.1	-1.0	-1.0	1.0	-1.0	1.0	1.0	1.0
373.8	1.0	-1.0	1.0	-1.0	1.0	-1.0	-1.0
66.8	-1.0	1.0	1.0	-1.0	-1.0	1.0	-1.0
79.6	1.0	1.0	1.0	-1.0	-1.0	-1.0	1.0
114.3	-1.0	-1.0	-1.0	1.0	1.0	1.0	-1.0
84.1	1.0	-1.0	-1.0	1.0	1.0	-1.0	1.0
68.4	-1.0	1.0	-1.0	1.0	-1.0	1.0	1.0
88.1	1.0	1.0	-1.0	1.0	-1.0	-1.0	-1.0
78.1	-1.0	-1.0	1.0	1.0	-1.0	-1.0	1.0

327.2	1.0	-1.0	1.0	1.0	-1.0	1.0	-1.0
77.6	-1.0	1.0	1.0	1.0	1.0	-1.0	-1.0
61.9	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Reading Data into Dataplot

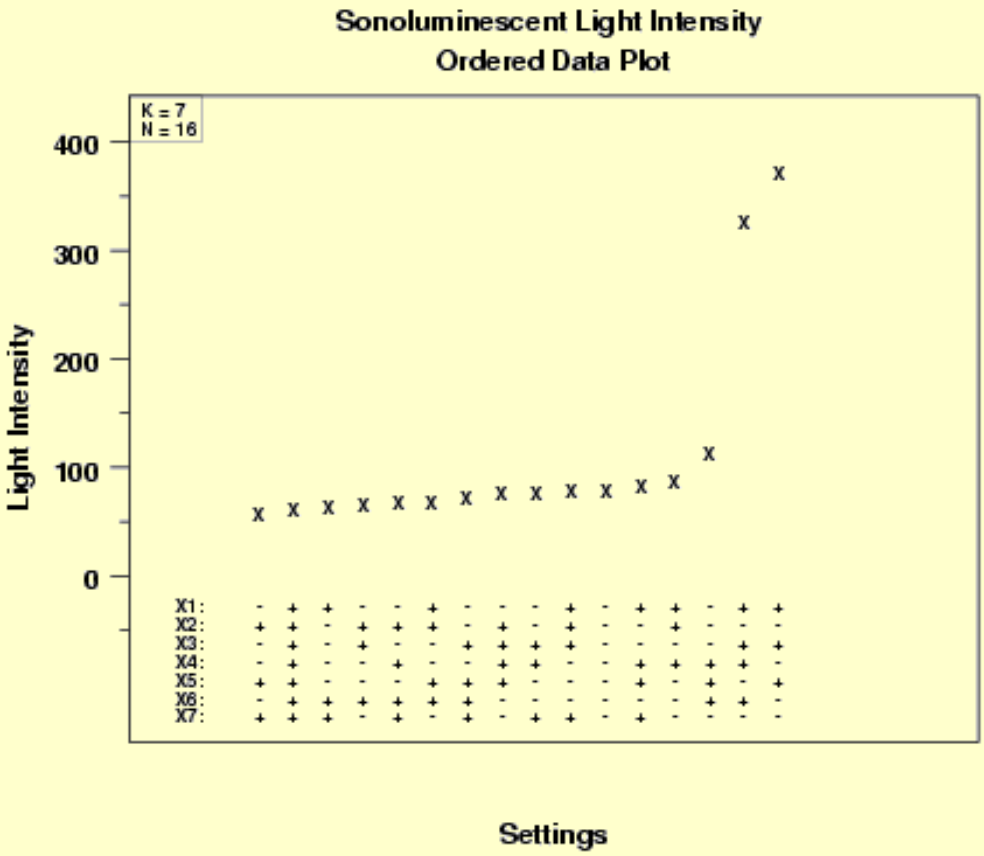
These data can be read into Dataplot with the following commands

```
SKIP 25
READ INN.DAT Y X1 TO X7
```

5.6.2.2. Initial Plots/Main Effects

Plot the Data: Ordered Data Plot

The first step in the analysis is to generate an ordered data plot.



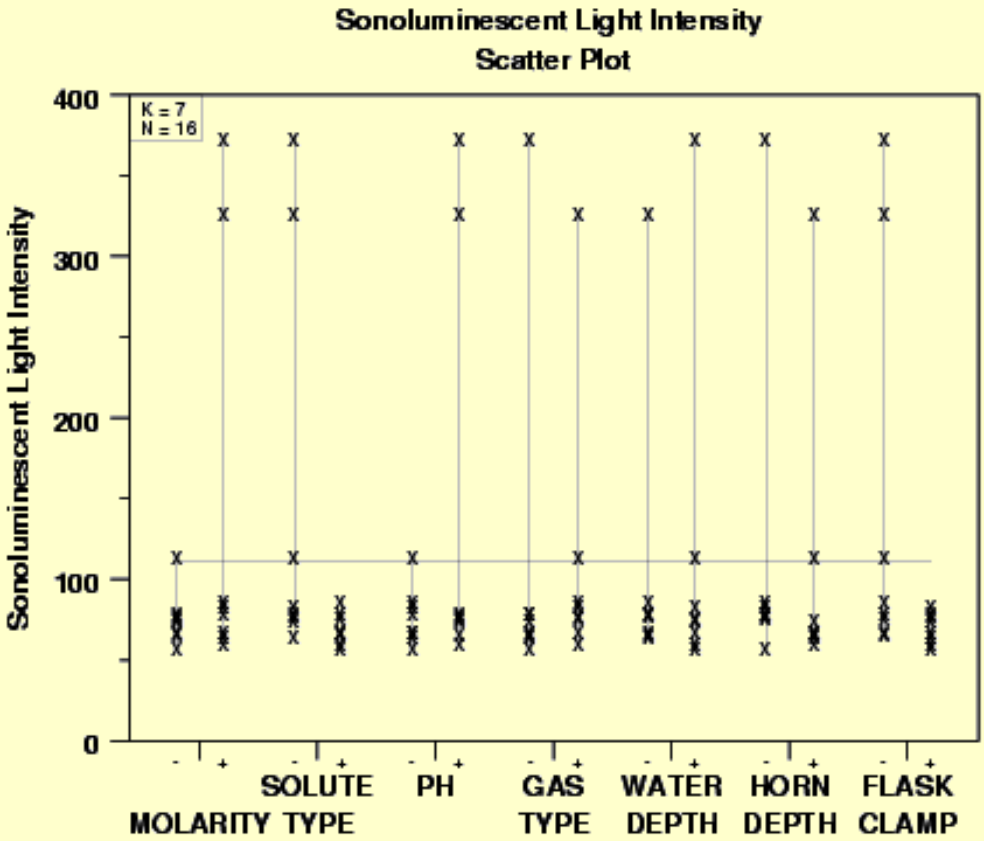
Conclusions from the Ordered Data Plot

We can make the following conclusions based on the ordered data plot.

1. Two points clearly stand out. The first 13 points lie in the 50 to 100 range, the next point is greater than 100, and the last two points are greater than 300.
2. Important Factors: For these two highest points, factors *X*1, *X*2, *X*3, and *X*7 have the same value (namely, +, -, +, -, respectively) while *X*4, *X*5, and *X*6 have differing values. We conclude that *X*1, *X*2, *X*3, and *X*7 are potentially important factors, while *X*4, *X*5, and *X*6 are not.
3. Best Settings: Our first pass makes use of the settings at the observed maximum (*Y* = 373.8). The settings for this maximum are (+, -, +, -, +, -, -).

Plot the Data: Dex Scatter Plot

The next step in the analysis is to generate a dex scatter plot.



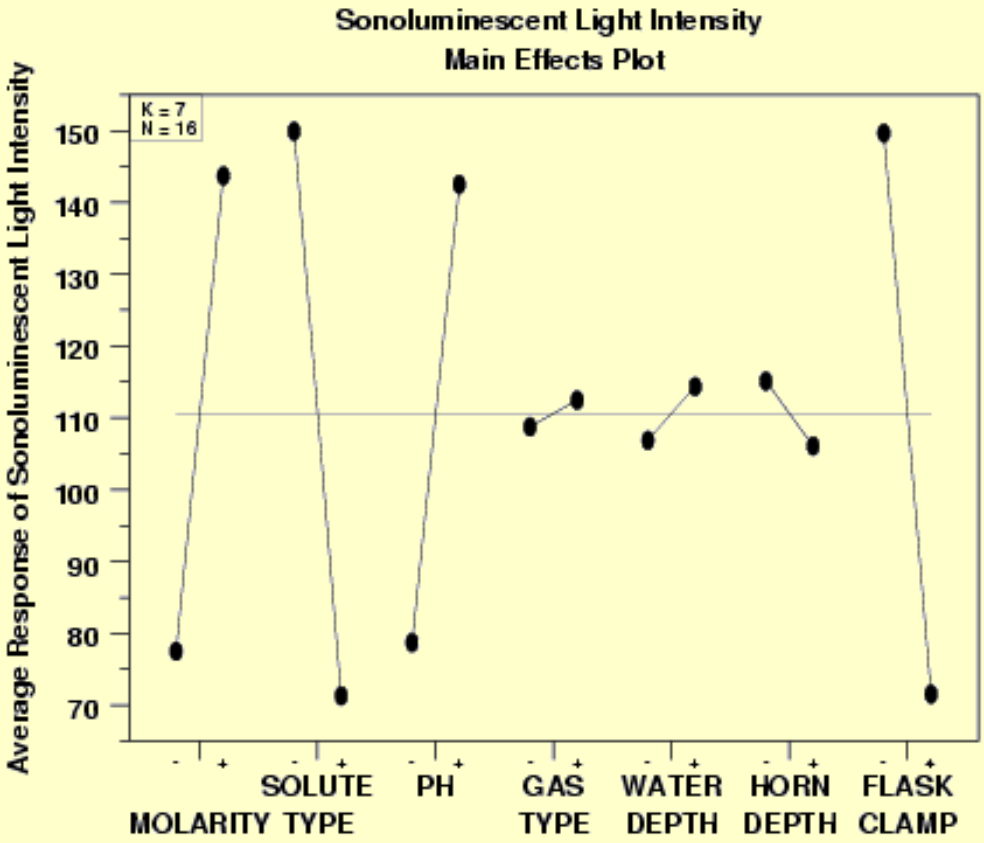
Conclusions from the DEX Scatter Plot

We can make the following conclusions based on the dex scatter plot.

1. Important Factors: Again, two points dominate the plot. For $X1$, $X2$, $X3$, and $X7$, these two points emanate from the same setting, (+, -, +, -), while for $X4$, $X5$, and $X6$ they emanate from different settings. We conclude that $X1$, $X2$, $X3$, and $X7$ are potentially important, while $X4$, $X5$, and $X6$ are probably not important.
2. Best Settings: Our first pass at best settings yields ($X1$ = +, $X2$ = -, $X3$ = +, $X4$ = either, $X5$ = either, $X6$ = either, $X7$ = -).

Check for Main Effects: Dex Mean Plot

The dex mean plot is generated to more clearly show the main effects:



Conclusions from the DEX Mean Plot

We can make the following conclusions from the dex mean plot.

1. Important Factors:
   *X*2 (effect = large: about -80)
   *X*7 (effect = large: about -80)
   *X*1 (effect = large: about 70)
   *X*3 (effect = large: about 65)
   *X*6 (effect = small: about -10)
   *X*5 (effect = small: between 5 and 10)
   *X*4 (effect = small: less than 5)
2. Best Settings: Here we step through each factor, one by one, and choose the setting that yields the highest average for the sonoluminescent light intensity:

   $(X1,X2,X3,X4,X5,X6,X7) = (+,-,+,+,+,-,-)$

Comparison of Plots

All of the above three plots are used primarily to determine the most important factors. Because it plots a summary statistic rather than the raw data, the dex mean plot shows the ordering of the main effects most clearly. However, it is still recommended to generate either the ordered data plot or the dex scatter plot (or both). Since these plot the raw data, they can sometimes reveal features of the data that might be masked by the dex mean plot.

In this case, the ordered data plot and the dex scatter plot clearly show two dominant points. This feature would not be obvious if we had generated only the dex mean plot.

Interpretation-wise, the most important factor *X*2 (solute) will, on the average, change the light intensity by about 80 units regardless of the settings of the other factors. The other factors are interpreted similarly.

In terms of the best settings, note that the ordered data plot, based on the maximum response value, yielded

+, -, +, -, +, -, -

Note that a consensus best value, with "." indicating a setting for which the three plots disagree, would be

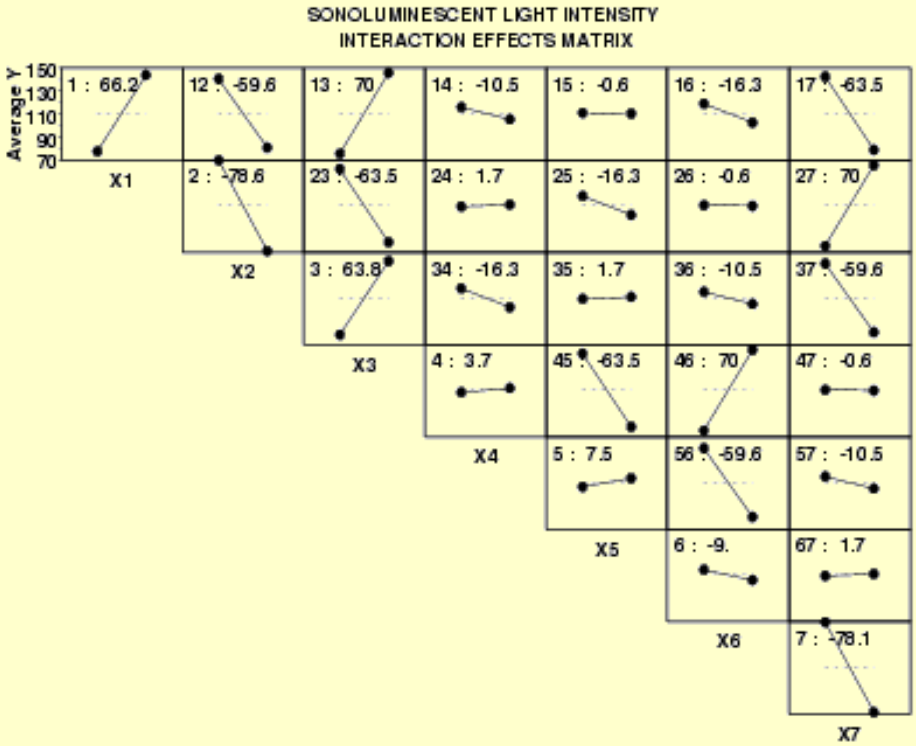
+, -, +, ., +, -, -

Note that the factor for which the settings disagree, *X*4, invariably defines itself as an "unimportant" factor.

5.6.2.3. Interaction Effects

Check for Interaction Effects: Dex Interaction Plot

In addition to the main effects, it is also important to check for interaction effects, especially 2-factor interaction effects. The dex interaction effects plot is an effective tool for this.



Conclusions from the DEX Interaction Effects Plot

We make the following conclusions from the dex interaction effects plot.

1. Important Factors: Looking for the plots that have the steepest lines (that is, the largest effects), and noting that the legends on each subplot give the estimated effect, we have that
   - The diagonal plots are the main effects. The important factors are: *X*2, *X*7, *X*1, and *X*3. These four factors have |effect| > 60. The remaining three factors have |effect| < 10.
   - The off-diagonal plots are the 2-factor interaction effects. Of the 21 2-factor interactions, 9 are nominally important, but they fall into three groups of three:
     - *X*1**X*3, *X*4**X*6, *X*2**X*7 (effect = 70)
     - *X*2**X*3, *X*4**X*5, *X*1**X*7 (effect approximately 63.5)
     - *X*1**X*2, *X*5**X*6, *X*3**X*7 (effect = -59.6)

   All remaining 2-factor interactions are small having an |effect| < 20. A virtue of the interaction effects matrix plot is that the confounding structure of this Resolution IV design can be read off the plot. In this case, the fact that *X*1**X*3, *X*4**X*6, and *X*2**X*7 all have effect estimates identical to 70 is not a mathematical coincidence. It is a reflection of the fact that for this design, the three 2-factor interactions are confounded. This is also true for the other two sets of three (*X*2**X*3, *X*4**X*5, *X*1**X*7, and *X*1**X*2, *X*5**X*6, *X*3**X*7).
2. Best Settings: Reading down the diagonal plots, we select, as before, the best settings "on the average":

   (*X*1,*X*2,*X*3,*X*4,*X*5,*X*6,*X*7) = (+,-,+,+,+,-,-)

   For the more important factors (*X*1, *X*2, *X*3, *X*7), we note that the best settings (+, -, +, -) are consistent with the best settings for the 2-factor interactions (cross-products):

   *X*1: +, *X*2: - with *X*1**X*2: -
   *X*1: +, *X*3: + with *X*1**X*3: +
   *X*1: +, *X*7: - with *X*1**X*7: -
   *X*2: -, *X*3: + with *X*2**X*3: -
   *X*2: -, *X*7: - with *X*2**X*7: +
   *X*3: +, *X*7: - with *X*3**X*7: -

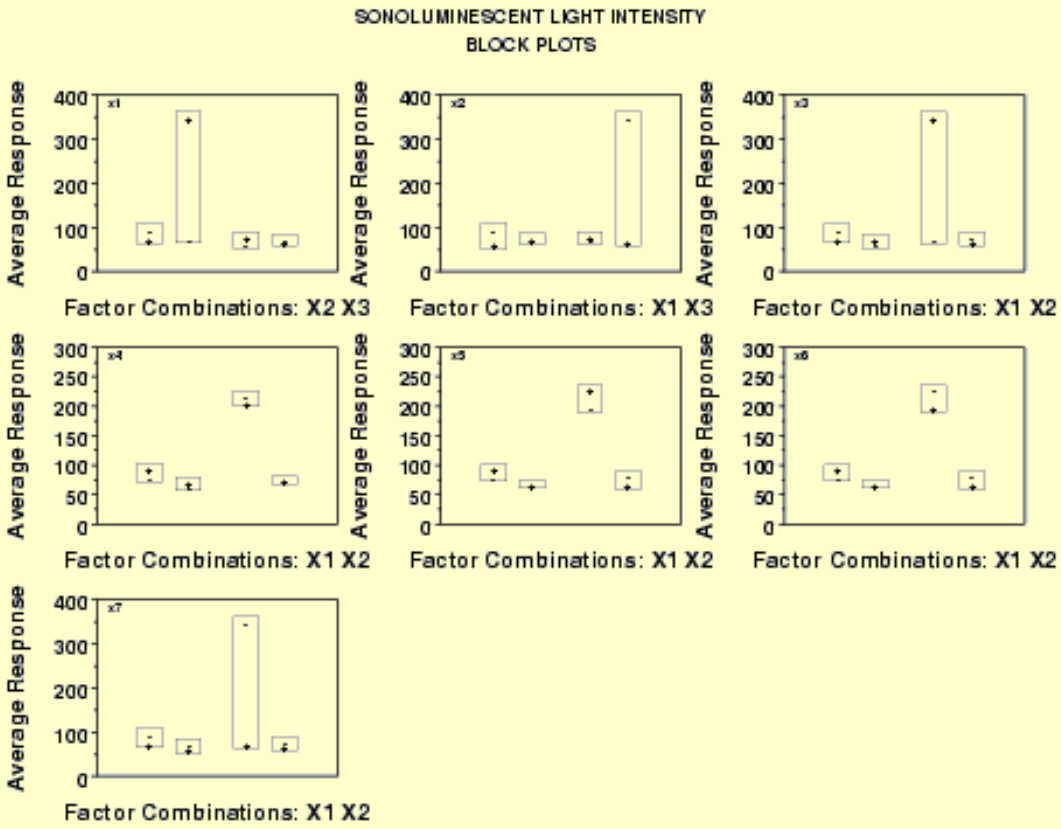
5.6.2.4. Main and Interaction Effects: Block Plots

Block Plots

Block plots are a useful adjunct to the dex mean plot and the dex interaction effects plot to confirm the importance of factors, to establish the robustness of main effect conclusions, and to determine the existence of interactions.

For block plots, it is the height of the bars that is important, not the relative positioning of each bar. Hence we focus on the size and internal signs of the blocks, not "where" the blocks are relative to each other.

We note in passing that for a fractional factorial design, we cannot display all combinations of the six remaining factors. We have arbitrarily chosen two robustness factors, which yields four blocks for comparison.



Conclusions from the Block Plots

We can make the following conclusions from the block plots.

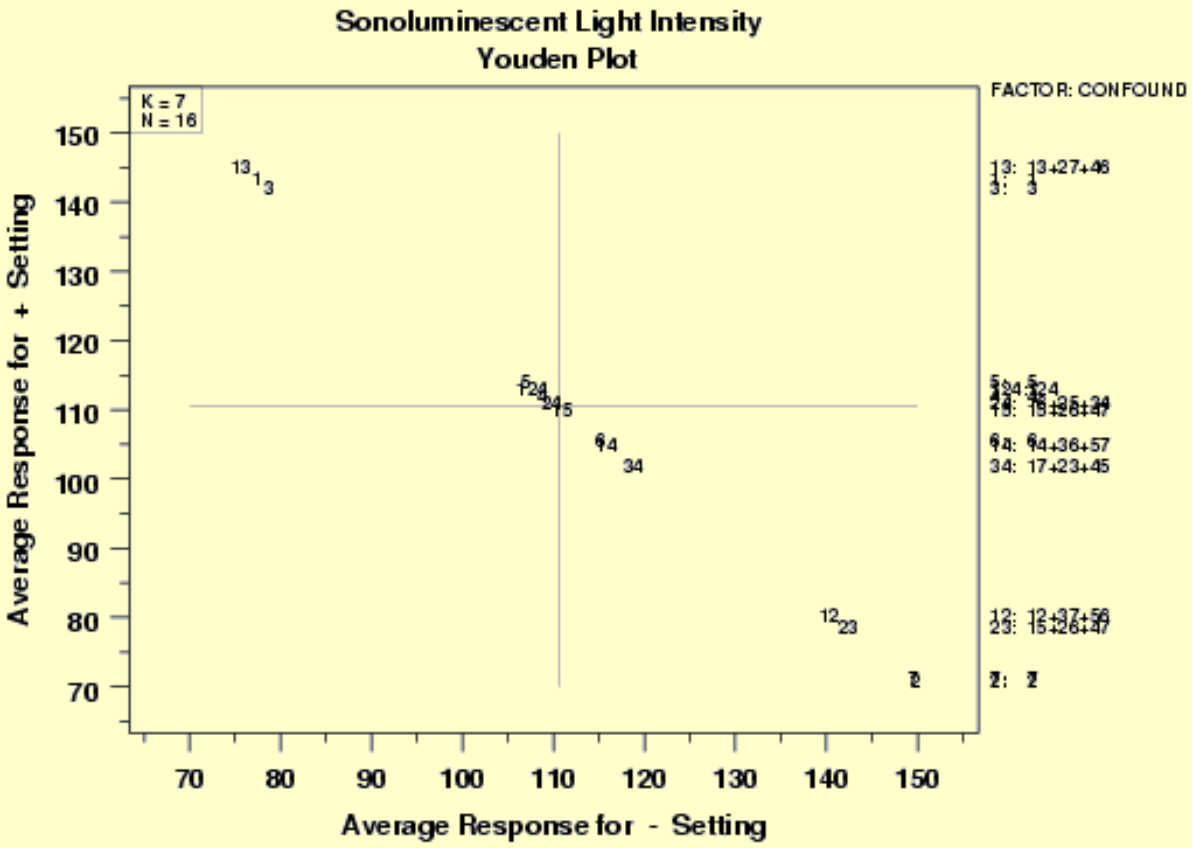
1. Relative Importance of Factors: Because of the expanded vertical axis, due to the two "outliers", the block plot is not particularly revealing. Block plots based on alternatively scaled data (e.g., LOG(Y)) would be more informative.

NIST SEMATECH HOME TOOLS & AIDS SEARCH BACK NEXT

5.6.2.5. Important Factors: Youden Plot

Purpose The dex Youden plot is used to distinguish between important and unimportant factors.

Sample Youden Plot



Conclusions from the Youden plot

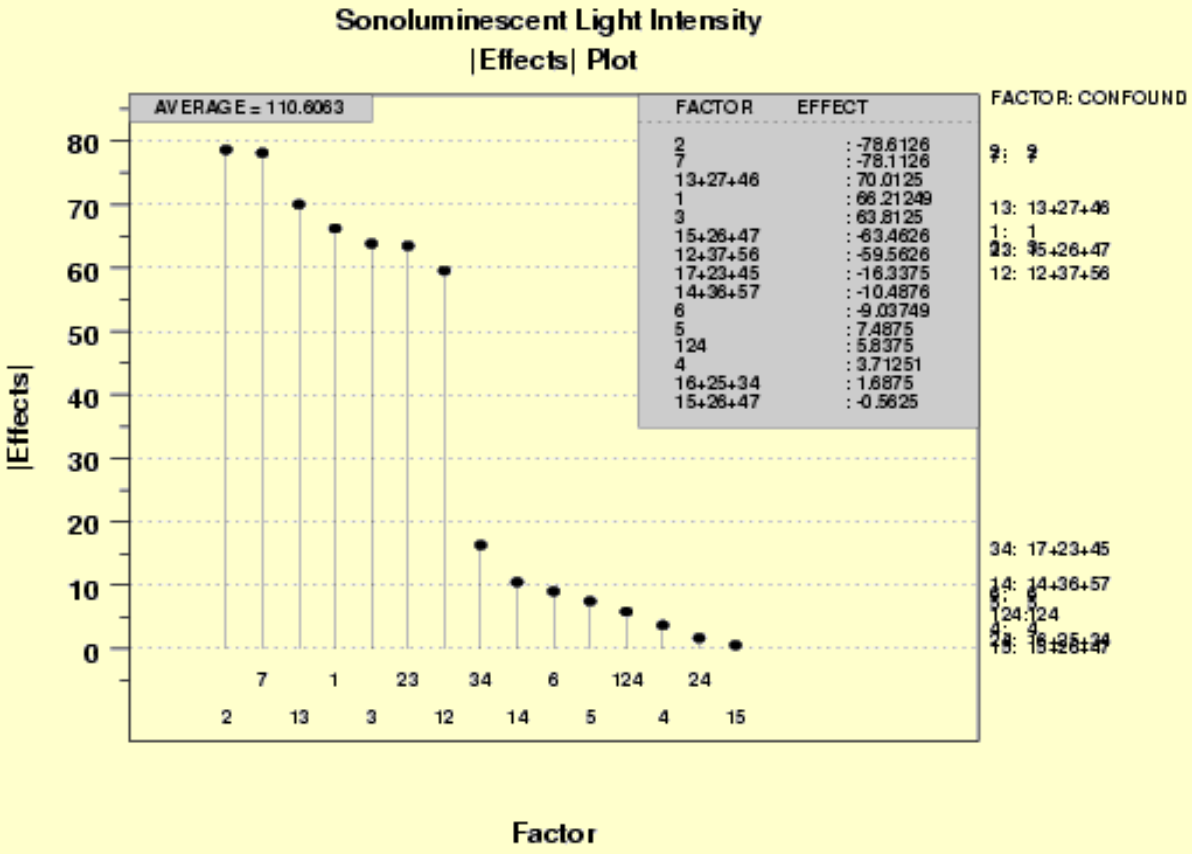
We can make the following conclusions from the Youden plot.

1. In the upper left corner are the interaction term $X1*X3$ and the main effects $X1$ and $X3$.
2. In the lower right corner are the main effects $X2$ and $X7$ and the interaction terms $X2*X3$ and $X1*X2$.
3. The remaining terms are clustered in the center, which indicates that such effects have averages that are similar (and hence the effects are near zero), and so such effects are relatively unimportant.
4. On the far right of the plot, the confounding structure is given (e.g., 13: 13+27+46), which suggests that the information on $X1*X3$ (on the plot) must be tempered with the fact that $X1*X3$ is confounded with $X2*X7$ and $X4*X6$.

5.6.2.6. Important Factors: |Effects| Plot

Purpose The |effects| plot displays the results of a Yates analysis in both a tabular and a graphical format. It is used to distinguish between important and unimportant effects.

Sample |Effects| Plot



Conclusions from the |effects| plot

We can make the following conclusions from the |effects| plot.

1. A ranked list of main effects and interaction terms is:

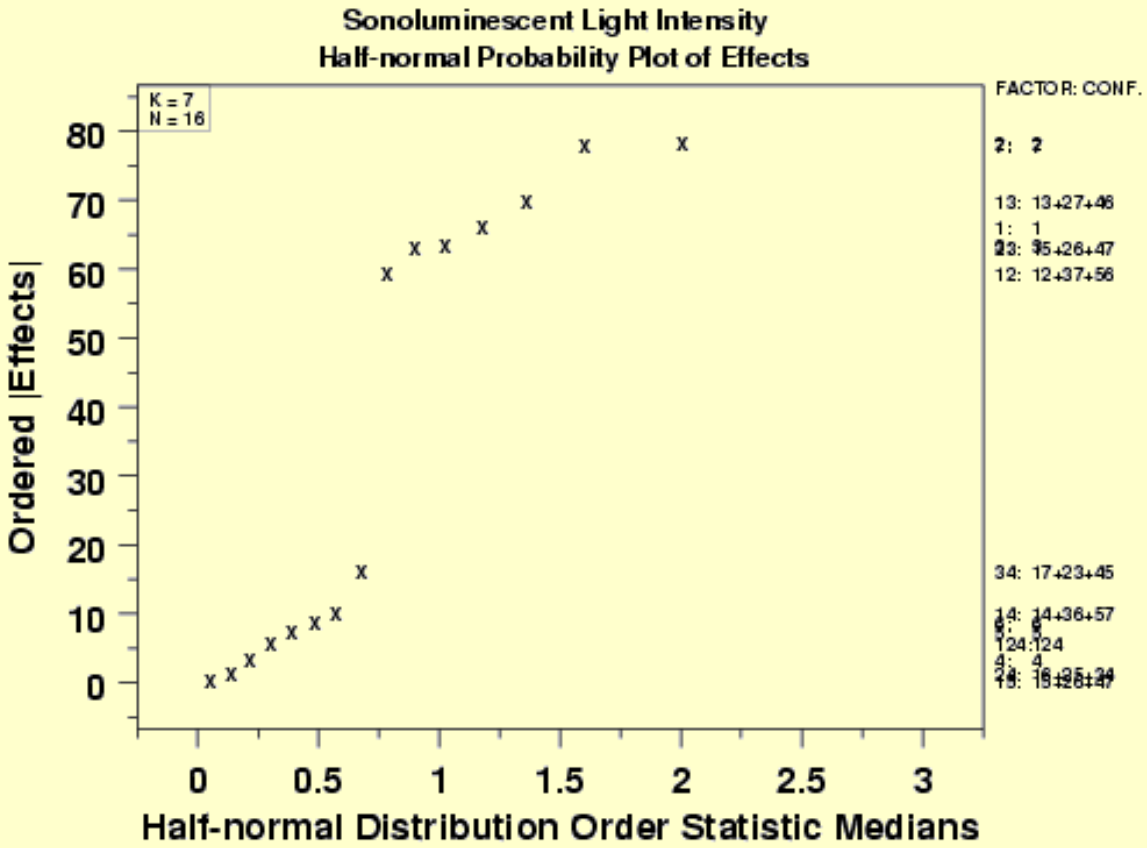
   *X*2
   *X*7
   *X*1**X*3 (confounded with *X*2**X*7 and *X*4**X*6)
   *X*1
   *X*3
   *X*2**X*3 (confounded with *X*4**X*5 and *X*1**X*7)
   *X*1**X*2 (confounded with *X*3**X*7 and *X*5**X*6)
   *X*3**X*4 (confounded with *X*1**X*6 and *X*2**X*5)
   *X*1**X*4 (confounded with *X*3**X*6 and *X*5**X*7)
   *X*6
   *X*5
   *X*1**X*2**X*4 (confounded with other 3-factor interactions)
   *X*4
   *X*2**X*4 (confounded with *X*3**X*5 and *X*6**X*7)
   *X*1**X*5 (confounded with *X*2**X*6 and *X*4**X*7)

2. From the graph, there is a clear dividing line between the first seven effects (all |effect| > 50) and the last eight effects (all |effect| < 20). This suggests we retain the first seven terms as "important" and discard the remaining as "unimportant".
3. Again, the confounding structure on the right reminds us that, for example, the nominal effect size of 70.0125 for *X*1**X*3 (molarity*pH) can come from an *X*1**X*3 interaction, an *X*2**X*7 (solute*clamping) interaction, an *X*4**X*6 (gas*horn depth) interaction, or any mixture of the three interactions.

5.6.2.7. Important Factors: Half-Normal Probability Plot

Purpose The half-normal probability plot is used to distinguish between important and unimportant effects.

Sample Half-Normal Probability Plot



Conclusions from the Half-Normal Probability Plot

We can make the following conclusions from the half-normal probability plot.

1. The points in the plot divide into two clear clusters:
   - An upper cluster (|effect| > 60).
   - A lower cluster (|effect| < 20).
2. The upper cluster contains the effects:

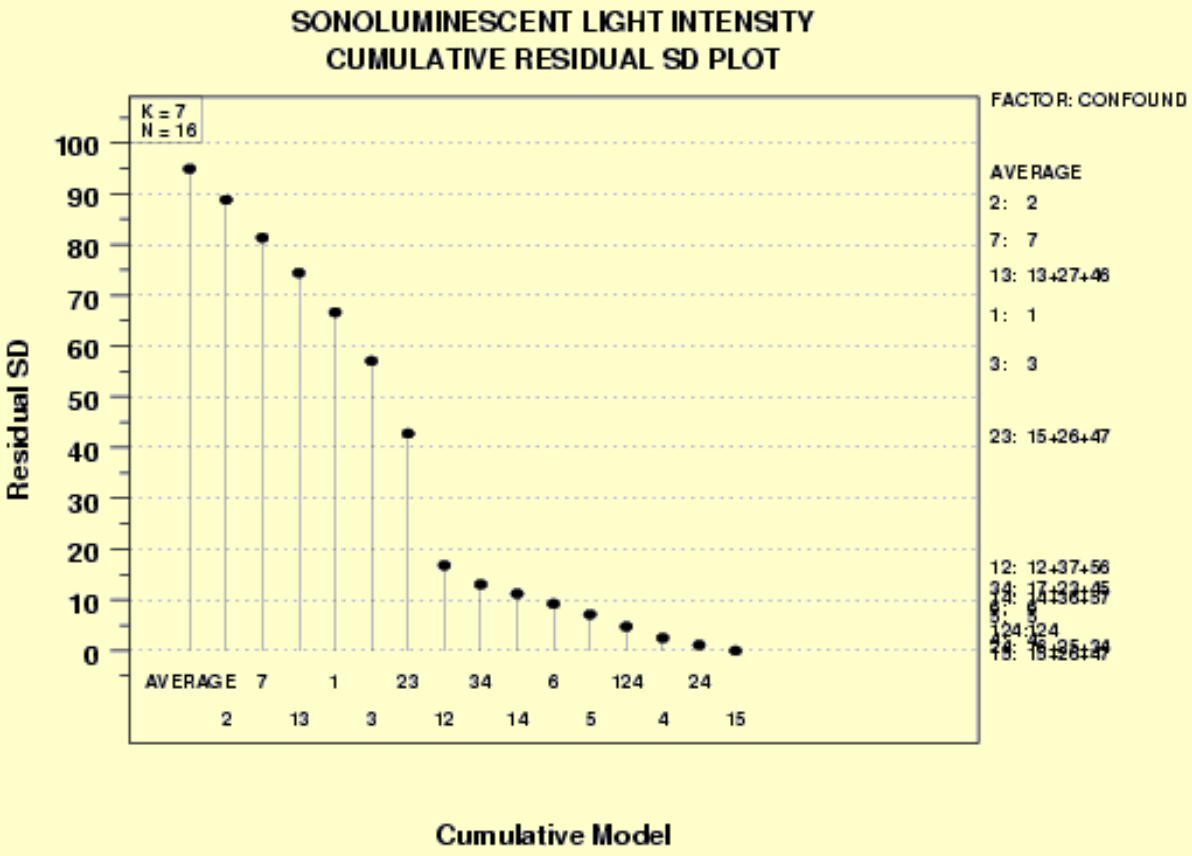
   *X*2, *X*7, *X*1**X*3 (and confounding), *X*1, *X*3, *X*2**X*3 (and confounding), *X*1**X*2 (and confounding)

   These effects should definitely be considered important.
3. The remaining effects lie on a line and form a lower cluster. These effects are declared relatively unimportant.
4. The effect id's and the confounding structure are given on the far right (e.g., 13:13+27+46).

5.6.2.8. Cumulative Residual Standard Deviation Plot

Purpose The cumulative residual standard deviation plot is used to identify the best (parsimonious) model.

Sample Cumulative Residual Standard Deviation Plot



Conclusions from the Cumulative Residual SD Plot

We can make the following conclusions from the cumulative residual standard deviation plot.

1. The baseline model consisting only of the average ($\hat{Y}$) = 110.6063) has a high residual standard deviation (95).
2. The cumulative residual standard deviation shows a significant and steady decrease as the following terms are added to the average: *X*2, *X*7, *X*1**X*3, *X*1, *X*3, *X*2**X*3, and *X*1**X*2. Including these terms reduces the cumulative residual standard deviation from approximately 95 to approximately 17.
3. Exclude from the model any term after *X*1**X*2 as the decrease in the residual standard deviation becomes relatively small.
4. From the |effects| plot, we see that the average is 110.6063, the estimated *X*2 effect is -78.6126, and so on. We use this to from the following prediction equation:

$$\begin{aligned}\hat{Y} = {} & 110.6063 + 0.5(-78.6126X_2 - 78.1126X_7 + \\ & 70.0125(X_1 * X_3) + 66.21249X_1 + 63.8125X_3 \\ & -63.4626(X_1 * X_5) - 59.562(X_1 * X_2))\end{aligned}$$

   Note that *X*1**X*3 is confounded with *X*2**X*7 and *X*4**X*6, *X*1**X*5 is confounded with *X*2**X6 and *X*4**X*7, and *X*1**X*2 is confounded with *X*3**X*7 and *X*5**X*6.

   From the above graph, we see that the residual standard deviation for this model is approximately 17.

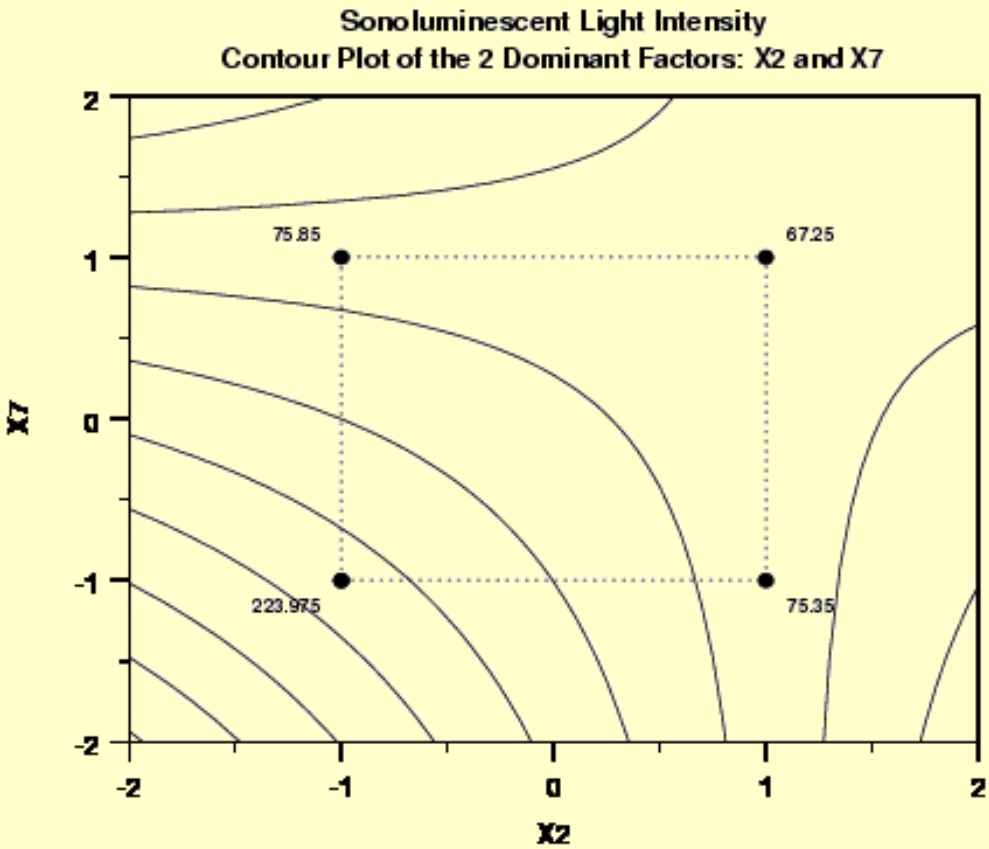
5.6.2.9. Next Step: Dex Contour Plot

Purpose

The dex contour plot is used to determine the best factor settings for the two most important factors in the next iteration of the experiment.

From the previous plots, we identified *X*2 (solute) and *X*7 (horn depth) as the two most important factors.

Sample Dex Contour Plot



Conclusions from the Dex Contour Plot

We can make the following conclusions from the dex contour plot.

1. The best (high light intensity) setting for *X*2 is "-" and the best setting for *X*7 is "-". This combination yields an average response of approximately 224. The next highest average response from any other combination of these factors is only 76.
2. The non-linear nature of the contour lines implies that the *X*2**X*7 interaction is important.
3. On the left side of the plot from top to bottom, the contour lines start at 0, increment by 50 and stop at 400. On the bottom of the plot from right to left, the contour lines start at 0, increment by 50 and stop at 400.

   To achieve a light intensity of, say 400, this suggests an extrapolated best setting of (*X*2, *X*7) = (-2,-2).
4. Such extrapolation only makes sense if *X*2 and *X*7 are continuous factors. Such is not the case here. In this example, *X*2 is solute (-1 = sugar and +1 = glycerol) and *X*7 is flask clamping (-1 is unclamped and +1 is clamped). Both factors are discrete, and so extrapolated settings are not possible.

5.6.2.10. Summary of Conclusions

Most Important Factors

The primary goal of this experiment was to identify the most important factors in maximizing the sonoluminescent light intensity.

Based on the preceding graphical analysis, we make the following conclusions.

- Four factors and three groups of 2-factor interactions are important. A rank-order listing of factors is:

  1. *X*2: Solute (effect = -78.6)
  2. *X*7: Clamping (effect = -78.1)
  3. *X*1**X*3 (Molarity*pH) or
     *X*2**X*7 (Solute*Clamping)
     (effect = 70.0)
  4. *X*1: Molarity (effect = 66.2)
  5. *X*3: pH (effect = 63.5)
  6. *X*2**X*3 (Solute*pH) or
     *X*4**X*5 (Gas*Water Depth)
     *X*1**X*7 (Molarity*Clamping)
     (effect = -63.5)
  7. *X*1**X*2 (Molarity*Solute) or
     *X*3**X*7 (Ph*Clamping)
     (effect = -59.6)

- Thus, of the seven factors and 21 2-factor interactions, it was found that four factors and at most seven 2-factor interactions seem important, with the remaining three factors and 14 interactions apparently being unimportant.

Best Settings The best settings to maximize sonoluminescent light intensity are

- *X*1 (Molarity) + (0.33 mol)
- *X*2 (Solute) - (sugar)
- *X*3 (pH) + (11)
- *X*4 (Gas) . (either)
- *X*5 (Water Depth) + (full)
- *X*6 (Horn Depth) - (5 mm)
- *X*7 (Clamping) - (unclamped)

with the *X*1, *X*2, *X*3, and *X*7 settings especially important.

NIST SEMATECH HOME TOOLS & AIDS SEARCH BACK NEXT

5.6.2.11. Work This Example Yourself

View Dataplot Macro for this Case Study

This page allows you to repeat the analysis outlined in the case study description on the previous page using Dataplot . It is required that you have already downloaded and installed Dataplot and configured your browser to run Dataplot. Output from each analysis step below will be displayed in one or more of the Dataplot windows. The four main windows are the Output window, the Graphics window, the Command History window, and the Data Sheet window. Across the top of the main windows there are menus for executing Dataplot commands. Across the bottom is a command entry window where commands can be typed in.

Data Analysis Steps	Results and Conclusions
Click on the links below to start Dataplot and run this case study yourself. Each step may use results from previous steps, so please be patient. Wait until the software verifies that the current step is complete before clicking on the next step.	*The links in this column will connect you with more detailed information about each analysis step from the case study description.*
1. Get set up and started. 1. Read in the data.	1. You have read 8 columns of numbers into Dataplot: variables Y, X1, X2, X3, X4, X5, X6, and X7.
2. Plot the main effects. 1. Ordered data plot. 2. Dex scatter plot. 3. Dex mean plot.	1. Ordered data plot shows 2 points that stand out. Potential important factors are X1, X2, X3, and X7. 2. Dex scatter plot identifies X1, X2, X3, and X7 as important factors. 3. Dex mean plot identifies X1, X2, X3, and X7 as important factors.

3. Plots for interaction effects 1. Generate a dex interaction effects plot.	1. The dex interaction effects plot shows several important interaction effects.
4. Block plots for main and interaction effects 1. Generate block plots.	1. The block plots are not particularly helpful in this case.
5. Youden plot to identify important factors 1. Generate a Youden plot.	1. The Youden plot identifies X1, X2, X3, and X7 as important factors. It also identifies a number of important interactions (X1*X3, X1*X2, X2*X3).
6. |Effects| plot to identify important factors 1. Generate |effects| plot.	1. The |effects| plot identifies X2, X7, X1*X3, X1, X3, X2*X3, and X1*X2 as important factors and interactions.
7. Half-normal probability plot to identify important factors 1. Generate half-normal probability plot.	1. The half-normal probability plot identifies X2, X7, X1*X3, X1, X3, X2*X3, and X1*X2 as important factors and interactions.

8. Cumulative residual standard deviation plot 1. Generate a cumulative residual standard deviation plot.	1. The cumulative residual standard deviation plot results in a model with 4 main effects and 3 2-factor interactions.
9. Dex contour plot 1. Generate a dex contour plot using factors 2 and 7.	1. The dex contour plot shows X2 = -1 and X7 = -1 to be the best settings.

5.7. A Glossary of DOE Terminology

Definitions for key DOE terms

This page gives definitions and information for many of the basic terms used in DOE.

- ***Alias***: When the estimate of an effect also includes the influence of one or more other effects (usually high order interactions) the effects are said to be *aliased* (see *confounding*). For example, if the estimate of effect D in a four factor experiment actually estimates (D + ABC), then the main effect D is aliased with the 3-way interaction ABC. **Note**: This causes no difficulty when the higher order interaction is either non-existent or insignificant.
- ***Analysis of Variance (ANOVA)***: A mathematical process for separating the variability of a group of observations into assignable causes and setting up various significance tests.
- ***Balanced Design***: An experimental design where all cells (i.e. treatment combinations) have the same number of observations.
- ***Blocking***: A schedule for conducting treatment combinations in an experimental study such that any effects on the experimental results due to a known change in raw materials, operators, machines, etc., become concentrated in the levels of the blocking variable. **Note**: the reason for blocking is to isolate a systematic effect and prevent it from obscuring the main effects. Blocking is achieved by restricting randomization.
- ***Center Points***: Points at the center value of all factor ranges.

***Coding Factor Levels*:** Transforming the scale of measurement for a factor so that the high value becomes +1 and the low value becomes -1 (see *scaling*). After coding all factors in a 2-level full factorial experiment, the design matrix has all orthogonal columns.

Coding is a simple linear transformation of the original measurement scale. If the "high" value is X_h and the "low" value is X_L (in the original scale), then the scaling transformation takes any original X value and converts it to (X - a)/b, where
$a = (X_h + X_L)/2$ and $b = (X_h - X_L)/2$.
To go back to the original measurement scale, just take the coded value and multiply it by "b" and add "a" or, X = b(coded value) + a.

As an example, if the factor is temperature and the high setting is 65ºC and the low setting is 55ºC, then a = (65 + 55)/2 = 60 and b = (65 - 55)/2 = 5. The center point (where the coded value is 0) has a temperature of 5(0) + 60 = 60ºC.

- ***Comparative Designs***: A design aimed at making conclusions about one a priori important factor, possibly in the presence of one or more other "nuisance" factors.
- ***Confounding***: A confounding design is one where some treatment effects (main or interactions) are estimated by the same linear combination of the experimental observations as some blocking effects. In this case, the treatment effect and the blocking effect are said to be *confounded*. Confounding is also used as a general term to indicate that the value of a main effect estimate comes from both the main effect itself and also contamination or bias from higher order interactions. Note: Confounding designs naturally arise when full factorial designs have to be run in blocks and the block size is smaller than the number of different treatment combinations. They also occur whenever a fractional factorial design is chosen instead of a full factorial design.
- ***Crossed Factors:*** See *factors below*.
- ***Design***: A set of experimental runs which allows you to fit a particular model and estimate your desired effects.
- ***Design Matrix***: A matrix description of an experiment that is useful for constructing and analyzing experiments.
- ***Effect:*** How changing the settings of a factor changes the response. The effect of a single factor is also called a *main effect*. **Note:** For a factor A with two levels, scaled so that low = -1 and high = +1, the effect of A is estimated by subtracting the average response when A is -1 from the average response when A = +1 and dividing the result by 2

(division by 2 is needed because the -1 level is 2 scaled units away from the +1 level).

- ***Error***: Unexplained variation in a collection of observations. **Note**: DOE's typically require understanding of both random error and lack of fit error.
- ***Experimental Unit***: The entity to which a specific treatment combination is applied. **Note**: an experimental unit can be a
  - PC board
  - silicon wafer
  - tray of components simultaneously treated
  - individual agricultural plants
  - plot of land
  - automotive transmissions
  - etc.
- ***Factors****: Process inputs* an investigator manipulates to cause a change in the output. Some factors cannot be controlled by the experimenter but may effect the responses. If their effect is significant, these *uncontrolled factors* should be measured and used in the data analysis. **Note**: The inputs can be discrete or continuous.
  - ***Crossed Factors***: Two factors are *crossed* if every level of one occurs with every level of the other in the experiment.
  - ***Nested Factors***: A factor "A" is nested within another factor "B" if the levels or values of "A" are different for every level or value of "B". **Note**: Nested factors or effects have a hierarchical relationship.
- ***Fixed Effect***: An effect associated with an input variable that has a limited number of levels or in which only a limited number of levels are of interest to the experimenter.
- ***Interactions***: Occurs when the effect of one factor on a response depends on the level of another factor(s).
- ***Lack of Fit Error***: Error that occurs when the analysis omits one or more important terms or factors from the process model. **Note**: Including replication in a DOE allows separation of experimental error into its components: lack of fit and random (pure) error.
- ***Model***: Mathematical relationship which relates changes in a given response to changes in one or more factors.

- ***Nested Factors***: See *factors* above.
- ***Orthogonality***: Two vectors of the same length are orthogonal if the sum of the products of their corresponding elements is 0. **Note**: An experimental design is orthogonal if the effects of any factor balance out (sum to zero) across the effects of the other factors.
- ***Random Effect***: An effect associated with input variables chosen at random from a population having a large or infinite number of possible values.
- ***Random error***: Error that occurs due to natural variation in the process. Note: Random error is typically assumed to be normally distributed with zero mean and a constant variance. **Note**: Random error is also called experimental error.
- ***Randomization***: A schedule for allocating treatment material and for conducting treatment combinations in a DOE such that the conditions in one run neither depend on the conditions of the previous run nor predict the conditions in the subsequent runs. **Note**: The importance of randomization cannot be over stressed. Randomization is necessary for conclusions drawn from the experiment to be correct, unambiguous and defensible.
- ***Replication***: Performing the same treatment combination more than once. **Note**: Including replication allows an estimate of the random error independent of any lack of fit error.
- ***Resolution***: A term which describes the degree to which estimated main effects are aliased (or confounded) with estimated 2-level interactions, 3-level interactions, etc. In general, the resolution of a design is one more than the smallest order interaction that some main effect is confounded (aliased) with. If some main effects are confounded with some 2-level interactions, the resolution is 3. **Note**: Full factorial designs have no confounding and are said to have resolution "infinity". For most practical purposes, a resolution 5 design is excellent and a resolution 4 design may be adequate. Resolution 3 designs are useful as economical screening designs.
- ***Responses***: The output(s) of a process. Sometimes called dependent variable(s).
- ***Response Surface Designs***: A DOE that fully explores the process window and models the responses. **Note**: These designs are most effective when there are less than 5

factors. Quadratic models are used for response surface designs and at least three levels of every factor are needed in the design.

- ***Rotatability***: A design is *rotatable* if the variance of the predicted response at any point **x** depends only on the distance of **x** from the design center point. A design with this property can be rotated around its center point without changing the prediction variance at **x**. **Note**: Rotatability is a desirable property for response surface designs (i.e. quadratic model designs).
- ***Scaling Factor Levels***: Transforming factor levels so that the high value becomes +1 and the low value becomes -1.
- ***Screening Designs***: A DOE that identifies which of many factors have a significant effect on the response. **Note**: Typically screening designs have more than 5 factors.
- ***Treatment***: A treatment is a specific combination of factor levels whose effect is to be compared with other treatments.
- ***Treatment Combination***: The combination of the settings of several factors in a given experimental trial. Also known as a *run*.
- ***Variance Components***: Partitioning of the overall variation into assignable components.

5. Process Improvement

5.8. References

Chapter specific references

Bisgaard, S. and Steinberg, D. M., (1997), "The Design and Analysis of 2^{k-p} Prototype Experiments," *Technometrics*, 39, 1, 52-62.

Box, G. E. P., and Draper, N. R., (1987), *Empirical Model Building and Response Surfaces*, John Wiley & Sons, New York, NY.

Box, G. E. P., and Hunter, J. S., (1954), "A Confidence Region for the Solution of a Set of Simultaneous Equations with an Application to Experimental Design," *Biometrika*, 41, 190-199

Box, G. E. P., and Wilson, K. B., (1951), "On the Experimental Attainment of Optimum Conditions," *Journal of the Royal Statistical Society, Series B*, 13, 1-45.

Capobianco, T. E., Splett, J. D. and Iyer, H. K., "Eddy Current Probe Sensitivity as a Function of Coil Construction Parameters." *Research in Nondesructive Evaluation*, Vol. 2, pp. 169-186, December, 1990.

Del Castillo, E., (1996), "Multiresponse Optimization via Constrained Confidence Regions," *Journal of Quality Technology*, 28, 1, 61-70.

Derringer, G., and Suich, R., (1980), "Simultaneous Optimization of Several Response Variables," *Journal of Quality Technology*, 12, 4, 214-219.

Draper, N.R., (1963), "Ridge Analysis of Response Surfaces," *Technometrics*, 5, 469-479.

Hoerl, A. E., (1959), "Optimum Solution of Many Variables Equations," *Chemical Engineering Progress*, 55, 67-78.

Hoerl, A. E., (1964), "Ridge Analysis," *Chemical Engineering Symposium Series*, 60, 67-77.

Khuri, A. I., and Cornell, J. A., (1987), *Response Surfaces*, Marcel Dekker, New York, NY.

Mee, R. W., and Peralta, M. (2000), "Semifolding 2^{k-p} Designs," *Technometrics*, *42*, 2, p122.

Miller, A. (1997), "Strip-Plot Configuration of Fractional Factorials," *Technometrics*, 39, 3, p153.

Myers, R. H., and Montgomery, D. C., (1995), *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John Wiley & Sons, New York, NY.

Ryan, Thomas P., (2000), *Statistical Methods for Quality Improvement*, John Wiley & Sons, New York, NY.

Taguchi, G. and Konishi, S., (1987), *Orthogonal Arrays and Linear Graphs*, Dearborn, MI, ASI press.

Well Known General References

Box, G. E. P., Hunter, W. G., and Hunter, S. J. (1978), *Statistics for Experimenters*, John Wiley & Sons, Inc., New York, NY.

Diamond, W. J., (1989), *Practical Experimental Designs*, Second Ed., Van Nostrand Reinhold, NY.

John, P. W. M., (1971), *Statistical Design and Analysis of Experiments*, SIAM Classics in Applied Mathematics, Philadelphia, PA.

Milliken, G. A., and Johnson, D. E., (1984), *Analysis of Messy Data, Vol. 1*, Van Nostrand Reinhold, NY.

Montgomery, D. C., (2000), *Design and Analysis of Experiments*, Fifth Edition, John Wiley & Sons, New York, NY.

Case studies for different industries

Snee, R. D., Hare, L. B., and Trout, J. R.(1985), *Experiments in Industry. Design, Analysis and Interpretation of Results*, Milwaukee, WI, American Society for Quality.

Case studies in Process Improvement, including DOE, in the Semiconductor Industry

Czitrom, V., and Spagon, P. D., (1997), *Statistical Case Studies for Industrial process Improvement*, Philadelphia, PA, ASA-SIAM Series on Statistics and Applied Probability.

Software to design and analyze experiments

In addition to the extensive design and analysis documentation and routines in Dataplot, there are many other good commercial DOE packages. This Chapter showed examples using "JMP" (by the SAS Institute, 100 SAS CampusDrive, Cary, North Carolina 27513-9905), as an illustration of a good commercial package.