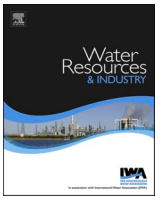
Water Resources and Industry 29 (2023) 100209

Contents lists available at ScienceDirect

Water Resources and Industry

journal homepage: www.elsevier.com/locate/wri



Application of machine learning algorithms for nonlinear system forecasting through analytics — A case study with mining influenced water data

Kagiso Samuel More, Christian Wolkersdorfer *

SARChI Chair for Mine Water Management, Department of Environmental, Water and Earth Sciences, Tshwane University of Technology, Private Bag X680, Pretoria, 0001, South Africa

ARTICLE INFO

Keywords:
Mining influenced water
Python programming language
Forecasting models
Data analytics
South Africa

ABSTRACT

Various techniques have been researched and introduced in water treatment plants to optimise treatment and management processes. This paper presents a solution that can help treatment plants to work more effectively and reach their mine water management goals. Using Python 3.7.1 programming language within an Anaconda 4.11.0 platform, neural networks and regression tree algorithms were compared to find the best performing model after the data had undergone robust data pre-processing and exploratory data analysis statistical techniques. The main aim was to use this best performing model to forecast mining influenced water (MIW) parameters. This approach will help the treatment plant operators in knowing the future MIW chemistry, and they can eventually plan ahead of time what chemicals and methods to use to treat and manage polluted MIW. Westrand mine pool water near Randfontein, South Africa is used as a case study, in which historical data (2016–2021) from shaft N° 9 is used to train and test the algorithms. These algorithms included the artificial neural network (ANN), deep neural network (DNN), gradient boosting and random forest regression trees, while the multivariate long short-term memory (LSTM) was used to generate new data for the best performing algorithm. Different data pre-processing approaches were explored, including data interpolation and anomaly detection. These processes were carried out to highlight the most important part of completing a machine learning related project, which is data analytics. Finally, the random forest regression tree algorithm showed the overall best performance and was used to forecast Fe and acidity concentrations of MIW for 60 days. It could be shown that artificial intelligence techniques are capable to optimise and forecast mine water treatment plant parameters, and it is imperative to perform robust statistical analysis on the data before attempting to build forecasting models.

1. Introduction and background

Mining influenced water (MIW), especially acid mine drainage (AMD), but also circum-neutral and alkaline drainage, is a threat to the former mining areas in South Africa and worldwide [1]. It is therefore imperative that any mine water is treated before it can enter local water courses. Due to the temporal changes and the longevity of the mine water quality [2,3], mine or treatment plant operators have to make provisions for the variability in the discharge qualities. If these changes can be quantified before they are about to occur,

* Corresponding author.
E-mail addresses: kagiso@moredatabytes.tech (K.S. More), Christian@Wolkersdorfer.info (C. Wolkersdorfer).

https://doi.org/10.1016/j.wri.2023.100209
Received 21 July 2022; Received in revised form 4 April 2023; Accepted 12 April 2023
Available online 14 April 2023

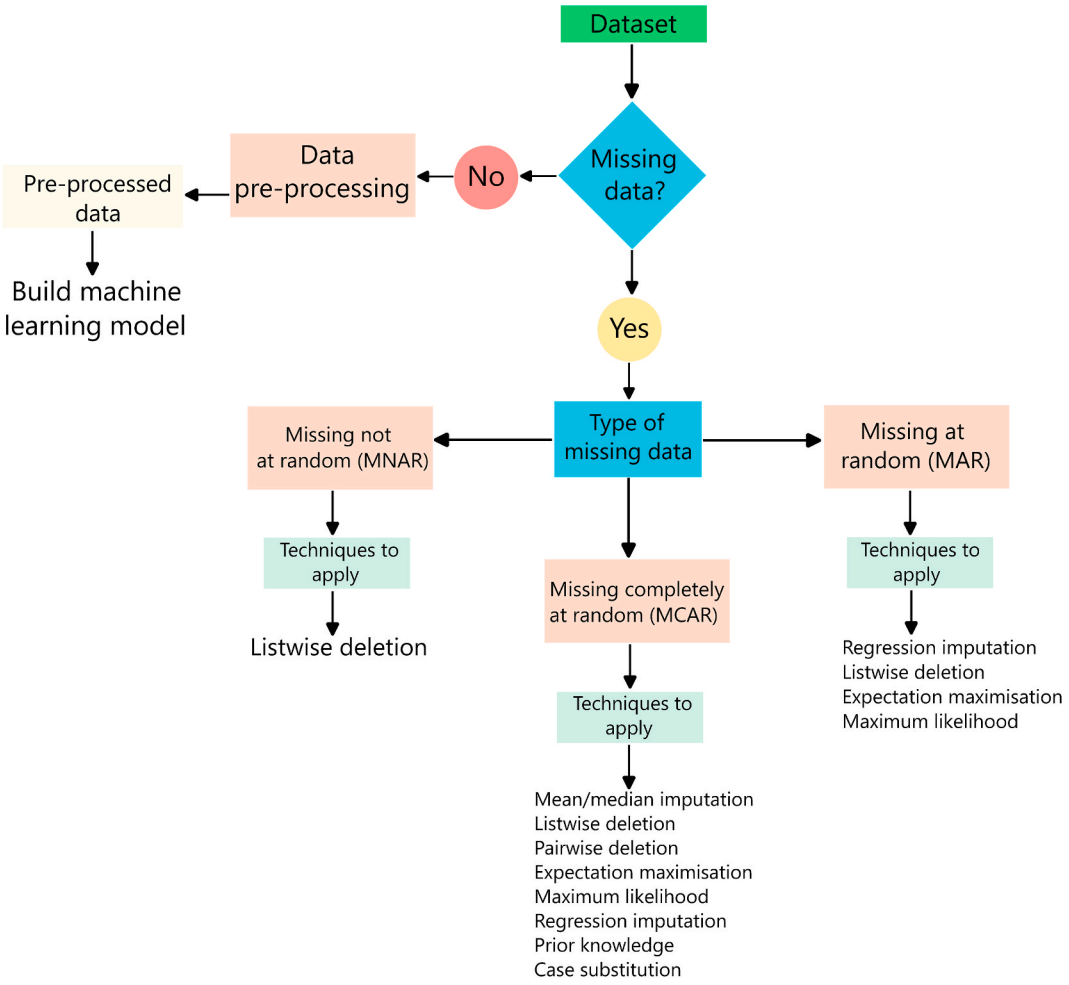


Fig. 1. Recommended techniques for each missing data type [modified and supplemented after [15]].

chemical stock, electricity or employee planning could be optimised to avoid over or under stocking or having too many personnel on site. Several mining companies are introducing advanced digital technologies in the treatment plants to treat and manage MIW effectively [4]. This study, therefore, applied machine learning (ML) techniques to forecast MIW parameters for 60 days using Fe and acidity exemplified by the former Randfontein Estates gold mine located in Randfontein, South Africa (Westrand mine pool). This kind of approach has never been tested on mine water dataset except for when More and Wolkersdorfer [5] only used regression tree models to forecast pH and EC values of mine water. The approach in this study is different due to the robust statistical techniques applied and multiple algorithms compared before deciding on the final model to use for forecasting analysis. Algorithms tested include the multi-layer perceptron (artificial) neural network (ANN), deep neural network (DNN), random forest and gradient boosting tree. Multivariate long short-term memory (LSTM) was used to generate new data for the best performing algorithm to forecast acidity and Fe of MIW. In the Westrand mine pool, MIW is pumped from shafts N№ 8 and N№ 9 to the treatment plant, and the main aim here is to forecast the mine water chemistry so that the plant operators can be prepared for changing water qualities ahead of time for optimal chemical dosing.

Usually, the traditional models fail to use all available parameters to forecast other parameters. These traditional models, such as auto regressive integrated moving average (ARIMA) or Box-Jenkins, assume that time series data are linear processes [6,7]. Additionally, they forecast data of an individual time series by analysing the underlying data structure and using its patterns and trends. In many cases, real world scenarios are nonlinear [8], and thus, relying only on the traditional time series forecasting techniques is highly disadvantageous and would be inappropriate for time series datasets of MIW. Mine water parameters produce a nonlinear dataset, thus ML models were applied in this study. ML models such as the neural networks have gained overwhelming attention over the past years in nonlinear time series forecasting [e.g. Refs. [9,10]] and have yielded positive results. These techniques, including regression tree models, use the whole dataset's structure and analyse the relationships between the data of the parameters in the whole dataset to forecast the future patterns and trends.

In addition to applying ML models, thorough data pre-processing and exploratory data analysis need to be practiced to produce models that can forecast the data with accuracy and precision. Missing data and anomalies are frequently encountered while collecting

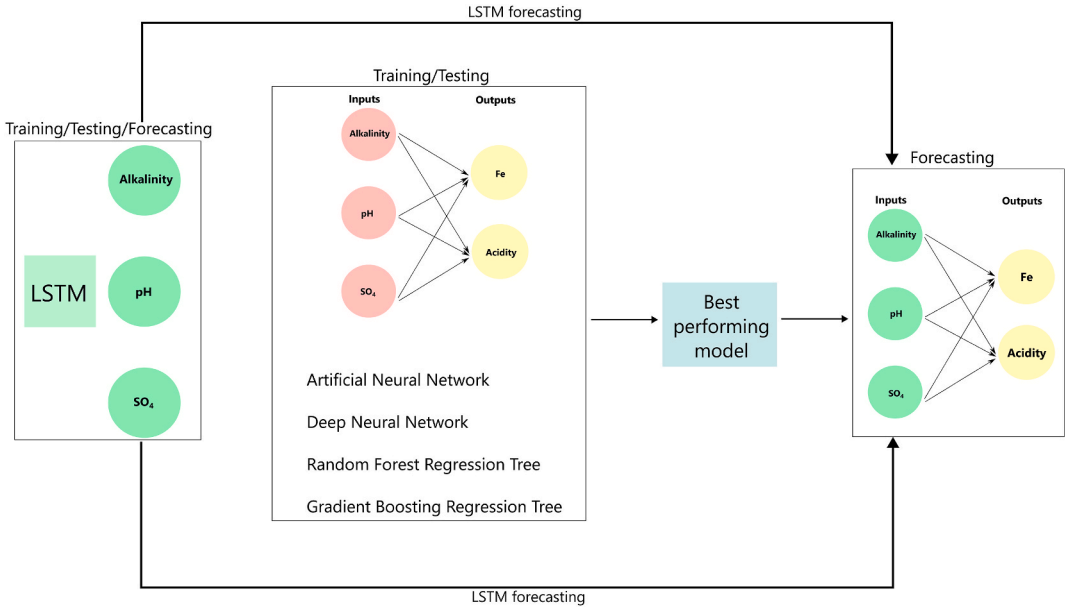


Fig. 2. Machine learning mechanism for the Westrand mine pool water treatment plant data.

MIW data, as this was the case with the data used for this study. Missing data compromise the statistical power of the study, while anomalies result in over-fitting or under-fitting of the models. Therefore, it is advantageous to apply suitable data interpolation and anomaly detection techniques on time series data before attempting to build forecasting models. Missing data in time series can occur due to several problems known as the missingness mechanism. The missingness mechanism can be in three different forms: missing completely at random (MCAR), missing at random (MAR) or missing not at random (MNAR) [Fig. 1; [11–13]]. A variable is MCAR if the probability of missingness is the same for all units, and it is MAR if the probability of missingness is depending only on available information. Additionally, missing data in MNAR are incomplete data that cannot be verified or predicted.

In this study, the data used consists of missing data that cannot be ignored, because doing so would lead to biased results, meaning that the type of missing data are MNAR. Therefore, numerical statistical modelling was investigated and suitable techniques used to interpolate the missing data. Approaches investigated include the basis-spline (B-Spline) curves, non-uniform rational basis spline (NURBS) curves and wavelet transform; however, only B-Splines were suitable to be applied in this study. Numerical modelling assumes that the time series data corresponds to an unknown function and the main aim is to fit the function and use it to interpolate the missing values [14]. In case of anomalies, statistical profiling and predictive confidence level (used for the data in this study) approaches were investigated in this study. Statistical profiling involves calculating measures of central tendency of the historical data and examining them, while the predictive confidence level approach uses the historical data to build a predictive model to get the overall trend, seasonality or cyclic pattern of the data, and ultimately detect the anomalies.

2. Study design

2.1. Introduction

Machine learning models in this study were developed in a three-step process, i.e. initial forecasting using the multivariate LSTM model, choosing the best performing model by comparing the ANN, DNN, random forest and gradient boosting tree models and finally performing the final forecasting analysis (Fig. 2). An LSTM model was used to forecast the values of alkalinity, pH and SO_4, and these forecasted values were supplied to the trained and tested best performing model to give the final concentrations of Fe and acidity for 60 days.

2.2. Multivariate long short-term memory (LSTM)

Multivariate Long Short-Term Memory systems (LSTMs) are a special type of recurrent neural network (RNN) and are mostly favoured because of the disadvantages that normal RNNs have. Normal RNNs have no long-term memory, cannot use information from distant past, and cannot learn patterns with long dependencies [16]. A way to overcome these issues is by introducing an LSTM which has memory cells that enable them to learn long-term patterns [16,17]. LSTM's default behaviour is remembering patterns and trends for a long period. They have a chain-like structure, similar to RNNs. However, the structure of the repeating module differs: for RNN, the repeating module is made up of a simple structure, such as a single tanh (hyperbolic tangent) layer, while LSTM's repeating module consist of four neural network layers interacting in a unique way (Fig. 3).

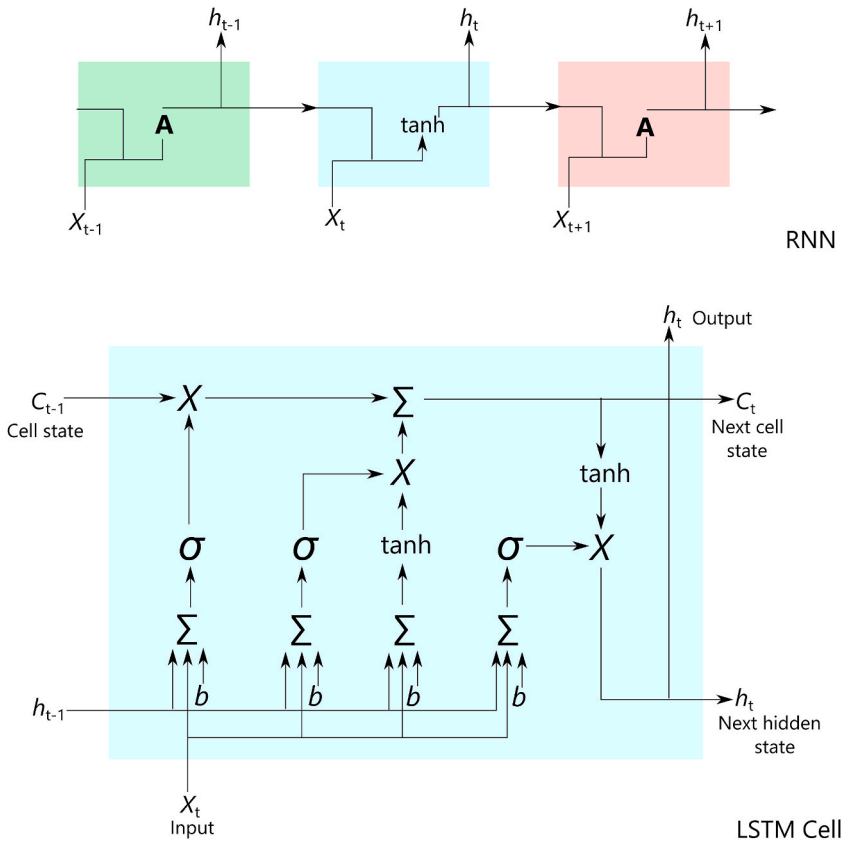


Fig. 3. Mechanism structure for RNN (above) and LSTM cells (below). Inputs are labelled as X_t: current input, C_{t-1}: memory from last LSTM unit and h_{t-1}: output of last LSTM unit. Outputs are labelled as C_t: new updated memory and h_t: current output. Nonlinearities are labelled as σ: sigmoid layer and tanh: tanh layer. Vector operations are labelled as X: scaling of information and $\sum$: adding information; b: bias.

2.3. Artificial neural network (ANN) and deep neural network (DNN)

Artificial Neural Networks (ANN) and Deep Neural Networks (DNN) are from the same group of algorithms, but only differ by the number of hidden layers. A typical neural network is made up of the input layer, hidden layer and output layer (Fig. 4), and if the network has one hidden layer then it is an ANN structure. A neural network that consists of two or more hidden layers is referred to as a DNN model. A fully connected layer in the neural network structure is practically composed of the weights and the bias of each neuron, and the input size controls the number of weights. Each neuron has its own activation function [6,9,18–20]. An input layer introduces values into the network, and it has no activation function. Hidden layer(s) perform the network's computations. Furthermore, the number of neurons in the input layer depends on the parameters that will be used in the network as inputs. An output layer makes final prediction for the network, and its neurons depend on the parameters that need to be predicted, while hidden layers can have any number of neurons stacked together. Hidden and output layer neurons have an activation function such as the sigmoid, rectified linear unit (ReLU) or softmax (normalised exponential function).

2.4. Regression tree algorithms

Regression trees are one of the key algorithms used in complex structures such as mine water dataset. They are useful when the data has no obvious linear relationship between the input and output parameters. In this study, random forest and gradient boosting regression trees were used. Decision trees make the foundation of both the algorithms. Regression tree algorithms have a tree-like structure which consists of root node, branches and leaf nodes [Fig. 5; [22,23]]. A random forest algorithm aims to reduce the variance in complex trees while gradient boosting aims to decrease the bias.

3. A review of selected data interpolation and anomaly detection techniques for time series

3.1. Basis spline (B-spline): data interpolation

Basis Spline (B-Spline) curves are an interpolation technique in which the order chosen for the curve is independent of the control

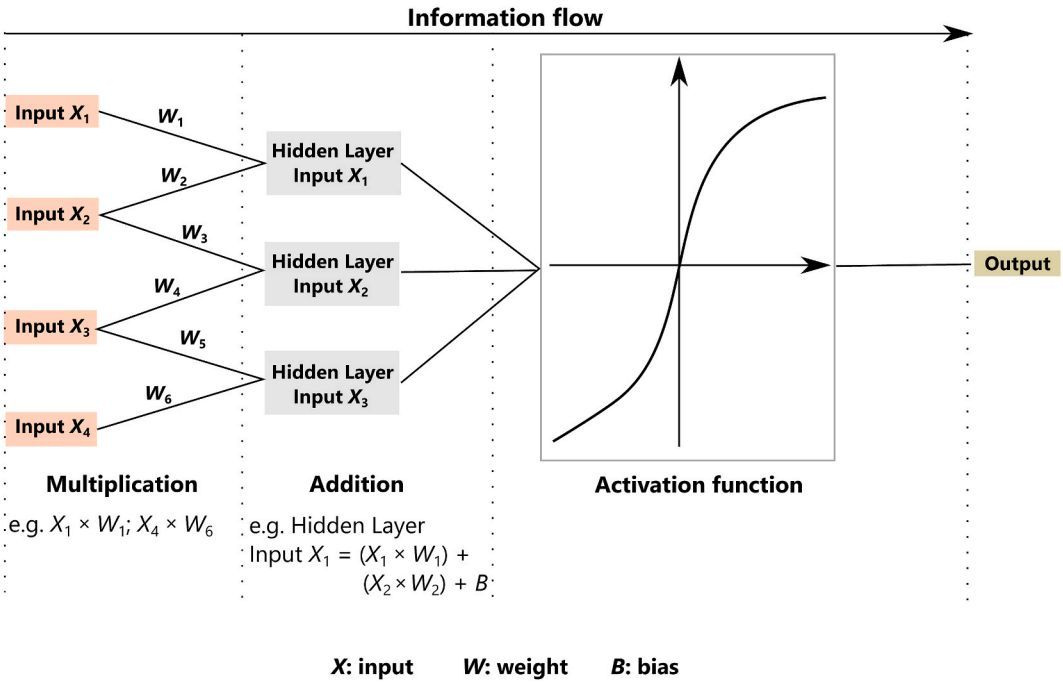


Fig. 4. A neural network's mechanism [modified after [21]].

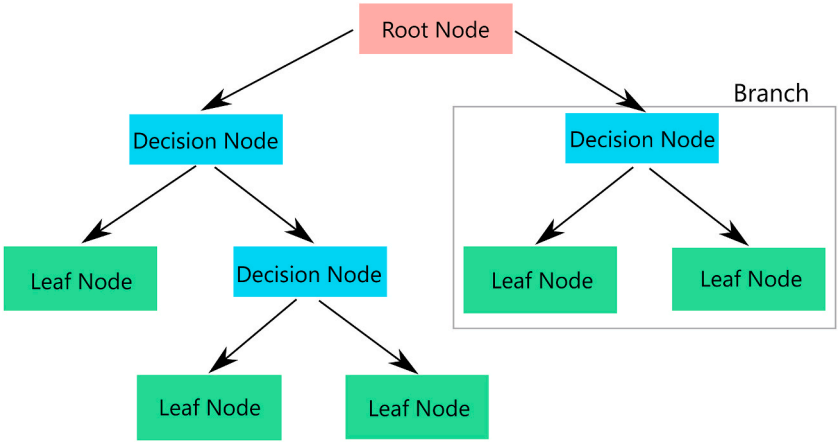


Fig. 5. Components of regression tree structure.

points [24,25]. Control points are used to determine the shape of a spline curve. B-Splines are unique and more advantageous when compared to other interpolation or approximation techniques such as splines and the Bezier curves [26]. Splines follow the general shape of the curve, while the Bezier curve generally follows the shape of a defining polygon [24,26,27]. In the B-Spline curve, the independency of the order of the curve over the local control points makes it a reliable interpolation technique. Thus, the B-Spline curve provides the local control through control points in every segment of the curve [25,28]. For example, a curve may have 40, 60 or 100 points and there will still be freedom to fix the curve to any shape, e.g. quadratic, cubic or higher order.

As can be seen, a B-Spline curve is not a single curve, but it is made up of a number of curve segments and all have the same continuity requirement depending on the order of the curve [24,29]. B-Splines can be applied for the open and closed curves, and changing any of the control points changes only a specific segment of the curve (Fig. 6), while in techniques like Bezier, the whole curve changes. Therefore, applying B-Spline interpolation in time series ensures that the missing data are interpolated within the population space and are not distorted or isolated.

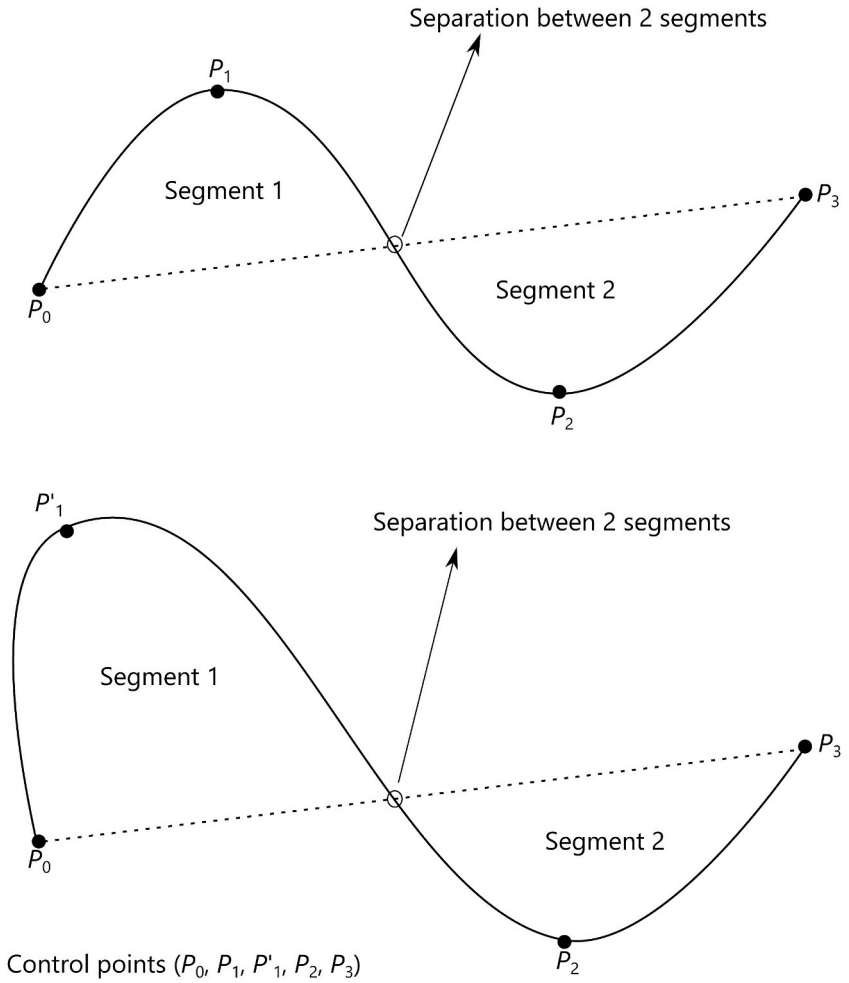


Fig. 6. B-Spline curve behaviour when changing the control point position. Changing the position of control point P_1 only changed segment 1, while segment 2 remained intact.

3.2. Non-uniform rational basis spline (NURBS): data interpolation

Non-uniform rational basis spline (NURBS) curves are a modification or rational version of B-Spline curves. Their advantage is that they can create smoother surfaces with fewer control points [30]. NURBS are mathematical representations of complex structures, be it two- or three-dimensional objects, e.g. cars, buildings, cones or simple curves. They are the computer graphics design industry standard when it comes to creating or interpolating complex objects [30–32]. Generally, they work exceptionally well in three-dimensional modelling, enabling the designer to easily manipulate control points and the contours' smoothness. The non-uniform in NURBS refers to the idea that some segments or sections of a defined shape can be easily manipulated relative to other sections of the overall shape with control points being associated with weights (positive numbers). When these control points all have the same weight, the curve is called non-rational [28,31,33] and the rationality of NURBS means that the curves have the possibility of being rational, i.e. NURBS have the ability to give more weight to the control points in the overall curve shape.

3.3. Wavelet transform: data interpolation

Wavelet transform is a technique derived from the Fourier Transform (FT) and Short-Time Fourier Transform (STFT). Limitations that arise from the FT and STFT techniques gave birth to the wavelet transform [24,34]. The FT provides frequency information of a signal that represents frequencies and their magnitude. However, it does not tell when in time these frequency components exist [35, 36]. Therefore, it is ideal for signals that do not change with time, i.e. signals that have a constant frequency throughout. Consequently, FT's disadvantage is that it lacks capability to provide frequency information for a localised signal region in time. STFT was therefore developed to overcome the poor time resolution of the FT. Thus, STFT explains the time frequency representation of the signal and it assumes that a certain portion of the non-stationary signal is stationary [37,38].

The main limitation of STFT is that high frequency components appear as short bursts, thus needing higher time resolution [39,40].

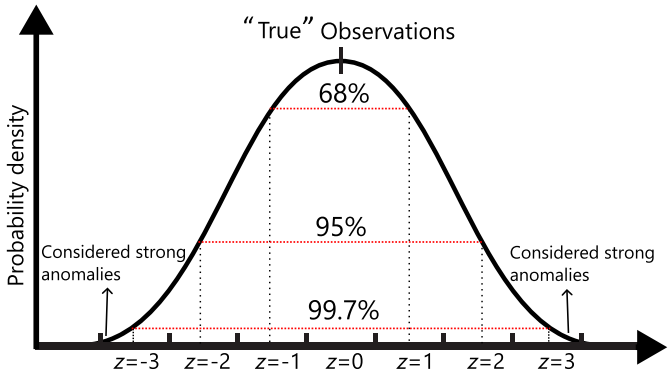


Fig. 7. An example of anomaly detection using Z-score analysis.

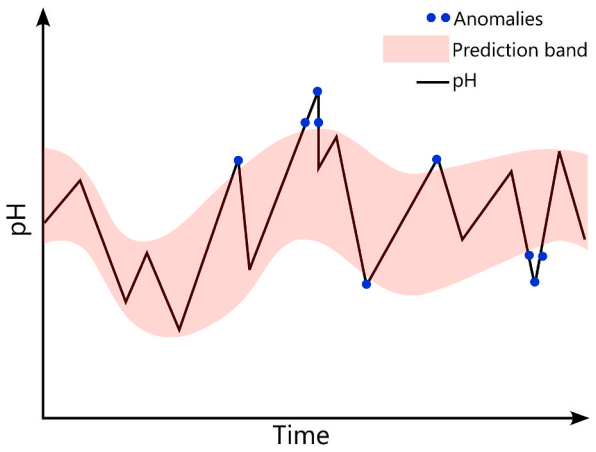


Fig. 8. An example of applying predictive confidence level bands to detect anomalies.

So, the wavelet transform improves on this shortfall, i.e. it results in analysing a signal into different frequencies at different resolutions. Therefore, a wavelet is a rapidly decaying wave-like oscillation that has zero mean and exists for a finite duration [24,34,38,41–44]. Wavelet interpolation is a key factor in time series data due to the data's stationarity properties. It is advantageous to apply this technique when the function is not smooth and is oscillating.

PyWavelets, an open source wavelet transform software for Python, can be used to interpolate the missing values for oscillating data [45]. PyWavelets combines a simple high level interface with low level C and Cython performance. Python consists of different types of wavelet families, e.g. Haar, Morlet, Gaussian or Frequency B-Spline wavelets. Therefore, for accurate results, a wavelet family which fits best with the supplied data must be chosen. Each wavelet family is useful for a different purpose as they have different smoothness, shape and compactness.

3.4. Statistical profiling approach: anomaly detection

Statistical profiling is a simple and fast process that involves calculating measures of central tendency of the historical data and examining them [14]. This can be done by calculating the mean, median or the moving average of the data. Standard deviation can also be calculated and be used to set up the uppermost and lower bounds of the dataset as anomaly thresholds. This technique is known as the Z-score analysis in which the anomaly threshold is set by the three-standard deviations above and below the mean [e.g. Refs. [5,46]]. Values that are outside the three-standard deviation thresholds are considered strong anomalies (Fig. 7). Simple moving average can also be used for anomaly detection: it is applied to capture the pattern in time series. The difference between the actual and simple moving average can be computed to determine the tolerance band and identify anomalies.

3.5. Predictive confidence level approach: anomaly detection

Another way of detecting anomalies in time series data is by using the historical data to build a predictive model to get the overall trend, seasonality or cyclic pattern of the data. The model error can be analysed between the predicted and actual values, and use that to compute a confidence interval (Fig. 8). The values falling beyond the confidence band can be regarded as anomalies. For example, an

Table 1
Mine water quality dataset from shaft N° 9 of the gold mine in Randfontein from 2016 to 03–07 to 2021-07-13; n: number of measurements, $\overline{x}$: average, σ: standard deviation, min.: minimum value, max.: maximum value. pH average calculated as $-\log10[\sum C_i/n]$, where C is the proton activity (https://www.wolkersdorfer.info/pH_en); measured values and units as reported by the plant.

Parameter	n	$\overline{x}$	σ	Min.	Max.
Acidity, mg/L $CaCO_3$	1123	406	337	48	1484
Alkalinity, mg/L $CaCO_3$	1123	155	55	70	298
EC, mS/m	1123	347	47	187	497
Fe, mg/L	1111	175	158	14	668
Mn, mg/L	1111	28.0	7.0	10.0	45.2
pH, —	1123	6.5	0.3	5.8	9.4
SO_4, mg/L	989	2436	303	1833	3184
Temperature, °C	1123	19.7	2.1	9.5	26.2
Turbidity, NTU	1116	22	36	0.7	275

Table 2
ADF test application on the mine water dataset to test for stationarity. N° lags: number of lags, n: number of observations used for ADF regression and critical values calculation. Critical values at $\alpha = 0.01$, 0.05 and 0.10 (−3.437, −2.864 and −2.568).

Parameter	ADF Statistic	p-value	N° lags	n
Acidity	−1.474	0.546	6	981
Alkalinity	−1.572	0.498	4	983
EC	−1.953	0.307	18	969
Fe	−1.580	0.493	11	976
Mn	−1.548	0.510	6	981
pH	−2.210	0.202	7	980
SO_4	−1.440	0.563	8	979
Temperature	−3.238	0.018	16	971
Turbidity	−2.739	0.068	22	965

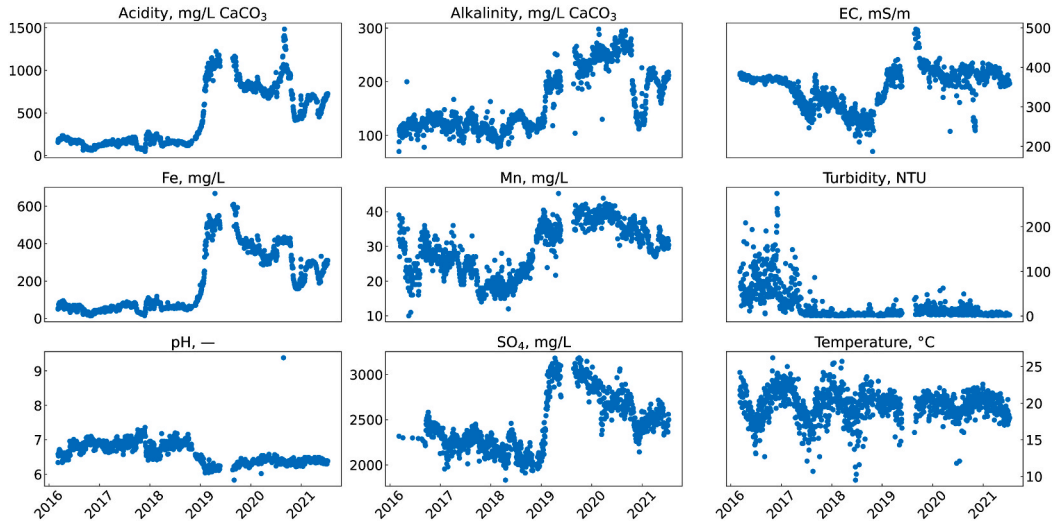


Fig. 9. Temporal mine water data development of Shaft N° 9 in the Western Basin of the Witwatersrand mines from 2016 to 2021.

ARIMA model can be built and use the mean absolute percentage error (MAPE) to come up with a confidence band [47,48]. Other ML or deep learning based algorithms such as the LSTM or LSTM autoencoder produce accurate results for time series data, and thus can be used to find anomalies [49]. This technique is highly dependable on the accuracy and good performance of the predictive model. Therefore, the model must be tuned to ensure that it produces a low error.

4. Dataset

4.1. Background

Sampling and monitoring of mine water is usually done on a regular basis to carefully examine its chemistry. This study uses the

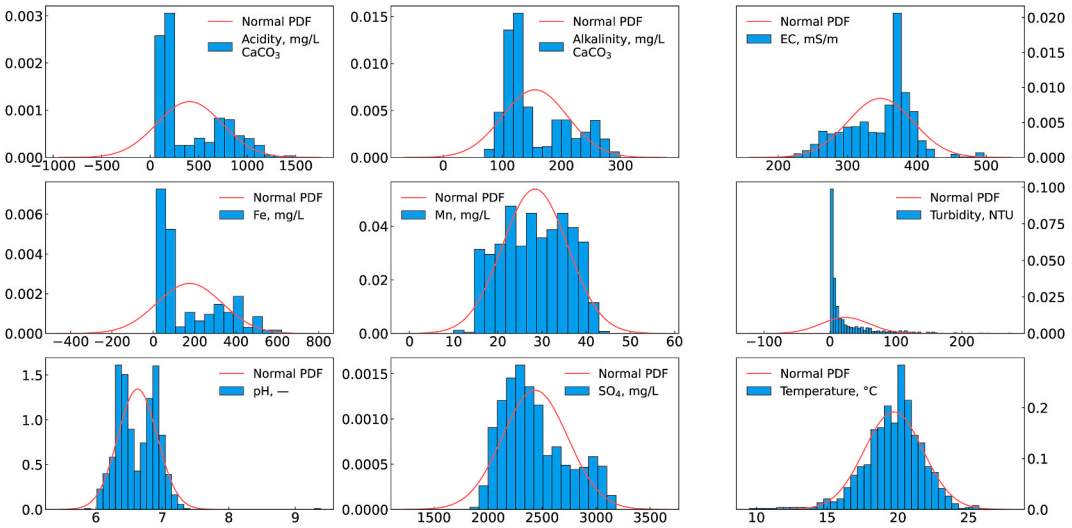


Fig. 10. Histograms with fitted normal probability distribution function (PDF) curves. Normal PDF curves were drawn using the SciPy 1.7.0 module by utilising the mean and standard deviations of the data.

Table 3
Normality tests using the Kolmogorov-Smirnov, Shapiro-Wilk and Anderson-Darling tests; *n*: number of observations. Critical values for $\alpha = 0.15$, 0.10, 0.05, 0.025 and 0.01 (0.574, 0.654, 0.0784, 0.915 and 1.088) (Anderson-Darling).

		Kolmogorov-Smirnov		Shapiro-Wilk		Anderson-Darling
Parameter	*n*	Test statistic	*p*-value	Test statistic	*p*-value	Test statistic
Acidity	1123	1.00	0.000	0.819	0.000	89.1
Alkalinity	1123	1.00	0.000	0.861	0.000	66.9
EC	1123	1.00	0.000	0.946	0.000	27.9
Fe	1111	1.00	0.000	0.815	0.000	89.2
Mn	1111	1.00	0.000	0.967	0.000	10.4
pH	1123	0.99	0.000	0.935	0.000	16.0
SO_4	995	1.00	0.000	0.942	0.000	18.5
Temperature	1123	1.00	0.000	0.972	0.000	5.4
Turbidity	1118	0.90	0.000	0.626	0.000	147.5

South African Westrand mine water treatment plant's data gathered between 2016 and 03–07 and 2021-07-13. The data contains nine parameters, i.e. acidity, alkalinity, electrical conductivity (EC), Fe, Mn, pH, SO_4, temperature and turbidity, which were used in the units applied by the plant operators. The parameters do not have an equal number of observations, i.e. some of the measurements are missing. The highest number of observations of an individual parameter is 1123 (Table 1). Robust data analytics approaches which included data interpolation and anomaly detection were conducted to prepare the data to be used in the machine learning models.

4.2. Stationarity test

Stationarity tests are critical as several statistical applications and models are computed based on its results. For example, numerical models are often applied to time series data when it is non-stationary, and probabilistic models are sometimes useful and accurate when the dataset is stationary [14]. Data interpolation and anomaly detection approaches are guided by stationarity tests. Therefore, the Augmented Dickey Fuller (ADF) test was used on the dataset to test the stationarity of the time series (Table 2). A stationary dataset will have a *p*-value that is highly significant (<0.05). Additionally, scatter plots for the dataset were drawn to visualise the patterns, trends and seasonality, and identify any stationarity or non-stationarity properties of the data (Fig. 9).

From the statistical results, the *p*-value obtained for temperature is less than 0.05. Therefore, the null hypothesis is rejected, i.e. temperature time series is stationary. However, the graphs show that temperature data are oscillating, a prominent seasonality can be observed in series. For the other parameters, the *p*-value for the time series is greater than 0.05. Therefore, the null hypothesis is accepted, i.e. the dataset for the parameters is non-stationary. Finally, the statistical results show that numerical modelling can be used to interpolate missing measurements and detect anomalies.

4.3. Normality test

Normality tests identify if the data has been sampled from a normal distribution. When data are plotted on a frequency distribution,

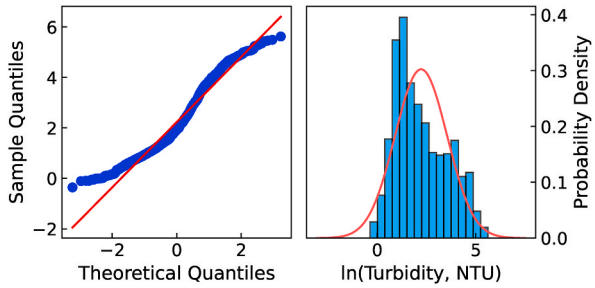


Fig. 11. Q–Q plot (left) and histogram with a normal PDF curve (right) for the transformed turbidity data.

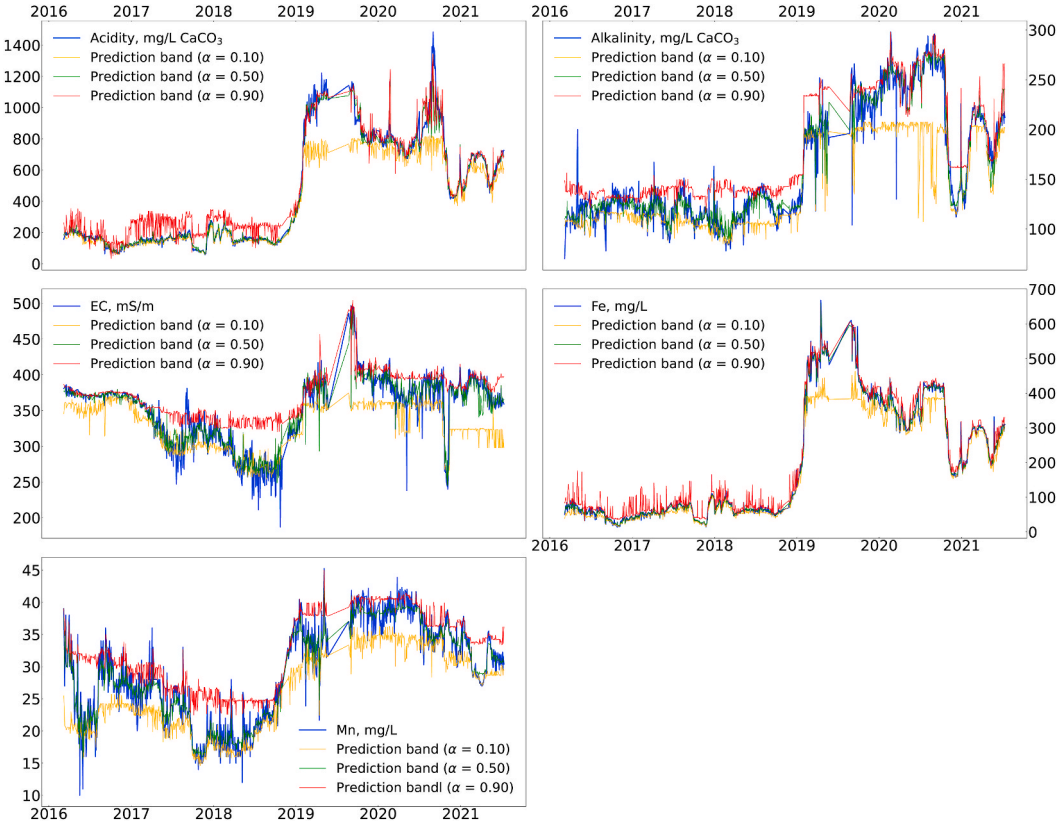


Fig. 12. Gradient boosting regressors with prediction intervals of $\alpha = 0.10$, 0.50 and 0.90 to detect anomalies in mining influenced water data. Only showing graphs for acidity, alkalinity, EC, Fe and Mn due to the data size.

the normal distribution can be explained by a bell-curve shape with majority of the observations being around the mean value. From the graphs plotted, the data are log-normally, bi-modally and multi-modally distributed and only temperature shows a close to Gaussian distribution (Fig. 10). Mining influenced water data are continuous data, thus the normality test is a crucial process for deciding statistical methods and measures of central tendency to perform data analysis. Apart from graphical methods, there are several statistical techniques applied to test for normality of data.

In this study, the SciPy 1.7.0 module was used to compute the Kolmogorov-Smirnov, Shapiro-Wilk, and Anderson-Darling normality tests (Table 3). Statistical tests conducted using the Kolmogorov-Smirnov and Shapiro-Wilk methods show *p*-values for the parameters to be below the 5% significance level, meaning that the data do not follow a normal distribution. Using the Anderson-Darling test, the test statistics are well above the critical values at $\alpha = 0.15$, 0.10, 0.05, 0.025 and 0.01 (0.574, 0.654, 0.784, 0.915 and 1.088), which also implies that the data do not follow a normal distribution.

4.4. Data transformation

As has been shown in the previous section, the dataset is not normally distributed. Several statistical methods, especially time-series

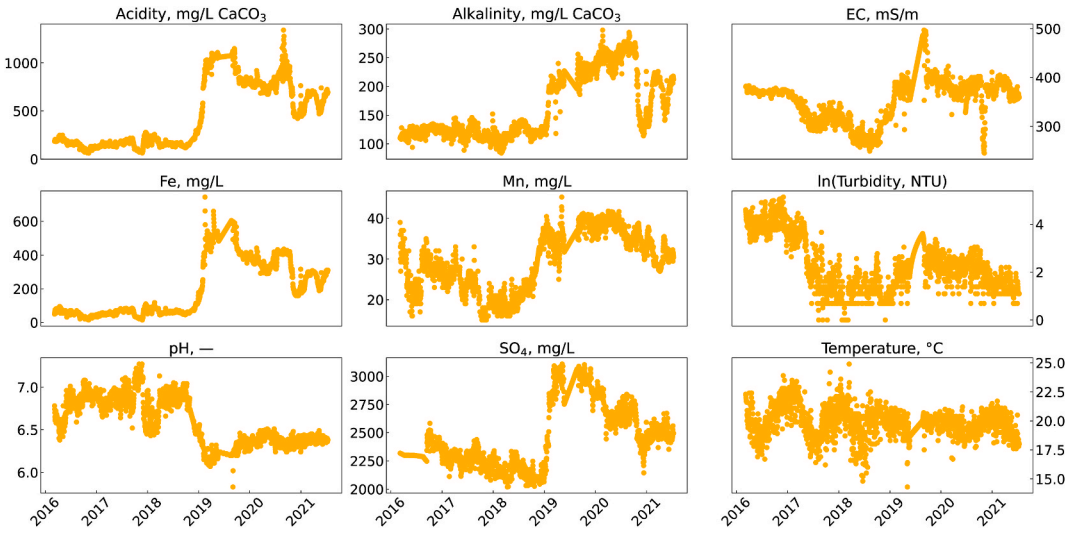


Fig. 13. Temporal mine water data development of Shaft N^{o} 9 in the Western Basin of the Witwatersrand mines from 2016 to 2021 including interpolated data.

forecasting techniques, are based on the assumption that the data are normally distributed. Thus, building forecasting models with untransformed data often results in inaccurate forecasting results. Therefore, data transformation is taking data that are not normally distributed and transforming it to a close-to normal distribution [50–53]. Data transformation does not change the relationship of the variables for mathematical and statistical purposes. So, the procedure is a statistically necessary step towards building models that can forecast data with accuracy and precision. From the results, it can be seen that alkalinity, EC and Mn have a multi-modal distribution, and acidity, Fe, pH and SO_4 show a bi-modal distribution. Temperature shows a close-to normal distribution and turbidity has a log-normal distribution. Therefore, only turbidity will be transformed. There are several ways to transform the data, such as log-normal, square-root, reciprocal, or Box-Cox transformation. In this study, a natural log-normal transformation was used: each variable of x was replaced by $\ln(x)$. Finally, quantile-quantile (Q–Q) and histogram graphs were plotted to test the normality of the transformed turbidity data (Fig. 11). In a Q–Q plot, for a normally distributed data, observations lie approximately on a straight line. Therefore, the graphs show that turbidity, after being transformed, is close-to a normal distribution, slightly showing a bi-modal distribution.

5. Anomaly detection and data interpolation

The dataset used for this study contains sample times with missing measurements resulting from equipment malfunctioning or because no sample was taken and has a small percentage of outliers identified by the statistical analysis conducted in the previous sections. There are several statistical methods to interpolate missing values and detect anomalies for time series data, some of which have been discussed earlier in this study. Predictive confidence level approach with gradient boosting regression tree algorithm was used to detect anomalies. In this approach, Python's Scikit-Learn 0.24.2 library was used to build the gradient boosting regression model. This model was fitted on the data with three prediction bands of $\alpha = 0.10$, 0.50 and 0.90 (Fig. 12). The $\alpha = 0.10$ prediction band represents the lower bound of the data (real observations plotting below this band are considered possible anomalies), while the $\alpha = 0.90$ prediction band represents the upper bound of the data (real observations plotting above this band are considered possible anomalies). For the mid-prediction, alpha was set to 0.50 and this predicts the median of the original data. Therefore, values plotting on the $\alpha = 0.50$ prediction band were used to replace the possible anomalies.

For the build-up of gradient boosting regression model, for each target output all other parameters were used as input variables, e.g. when acidity was set as the target output, alkalinity, EC, Fe, Mn, pH, SO_4, temperature and turbidity were set as input variables. This model used 100 trees, a maximum tree depth of 2 and learning rate of 0.05 to perform predictions, and the data were split into training (80%) and testing (20%) sets. Gradient boosting model's objective was to predict the values of the parameters that will possibly be used to build the forecasting models. There are several hidden features, some of which are not included in the data, which affect each parameter. Therefore, the uncertainty in the estimates was shown by predicting the lower ($\alpha = 0.10$), middle ($\alpha = 0.50$) and upper ($\alpha = 0.90$) bands of the observations. The loss function of the gradient boosting model was changed to quantiles with selected prediction intervals (alphas). This configuration ensures that the model performs predictions which correspond to percentiles.

Numerical analysis modelling was applied to interpolate the missing measurements. Numerical analysis assumes that time series data trend and pattern represent an unknown function [14]. The main task when applying this technique is to find a suitable function for the data so it can be used to interpolate the missing values. For reasons described above, B-Spline interpolation technique, using Python's SciPy library, was used in this study. B-Spline interpolation is a form of interpolation where a continuous curve has various piecewise polynomials whose gradients match up at the measured data. Interpolation in this form takes place between two points that

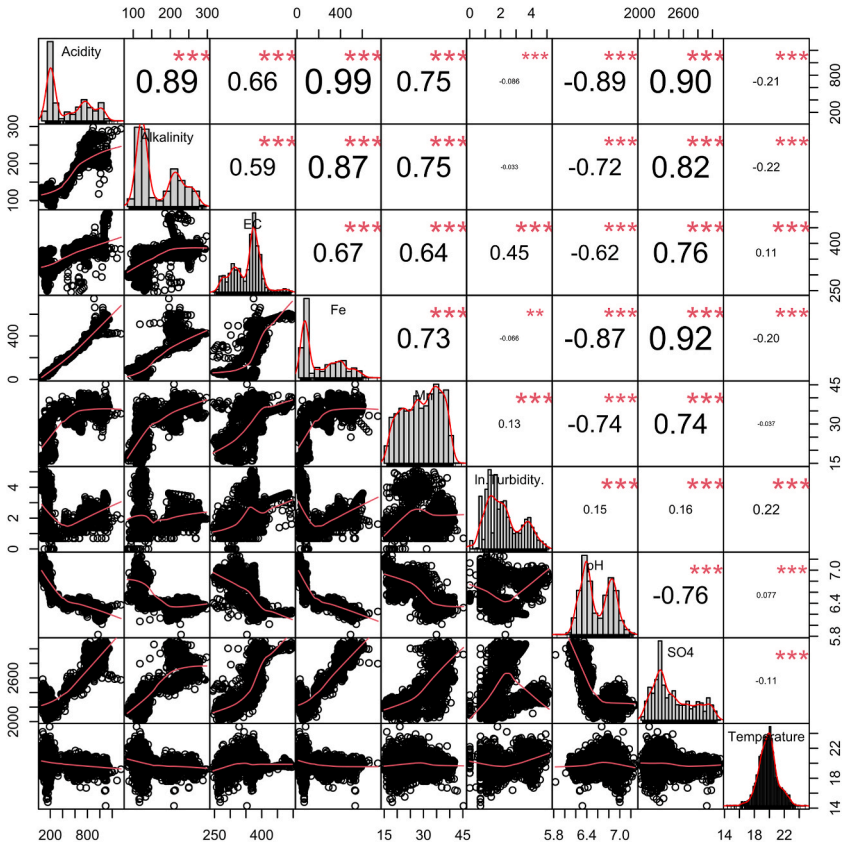


Fig. 14. Correlation chart with the distribution of each parameter shown on the diagonal; on the bottom of the diagonal are the bivariate scatter plots with a fitted line; on the top of the diagonal are the values of the correlation with the significance levels shown as asterisks: ***$p < 0.001$, **$p < 0.01$, *$p < 0.05$; correlation coefficient font size is related to the relationship between the parameters – the stronger the relationship the bigger the font size and vice versa.

have missing values, i.e. a function is derived between the points and interpolation takes place. The polynomials are continuous up to their second derivative, and this process occurs for every paired-point that has missing values between them [24,54]. Finally, the interpolated values fitted well into the known population, and these interpolated values did not change the trends and patterns of the original data (Fig. 13). This implies that B-Spline interpolation is an accurate technique for non-stationary time series data.

6. Modelling data

Data analytics approaches were conducted to detect anomalies and interpolate the missing observations by robust statistical techniques described above, which provided a "clean" dataset that could be used for modelling. A correlation chart of nine parameters, all with 1955 observations, was computed using R 4.1.2 on RStudio 1.1.456 and this produced a chart with cross plots, distribution plots and Pearson correlation coefficients (r) (Fig. 14). Parameters of concern at the Westrand mine water treatment plant are acidity and Fe. Therefore, their relationships with other parameters were thoroughly examined. The statistical analyses show that both acidity and Fe have high correlations with alkalinity, EC, Mn, pH and SO_4. Additionally, alkalinity, pH and SO_4 have good relationships with acidity and Fe with r above 0.8. Therefore, alkalinity, pH and SO_4 were used as input parameters for final forecasting, with acidity and Fe being the target outputs (Fig. 2). Turbidity and temperature were not used to build the forecasting models because they have poor relationships with the rest of the parameters, as displayed by the correlation coefficients. In total, seven parameters from the available nine were used to develop the models.

7. Model development and evaluation

All the models were trained and tested independently and only the best performing model was used in the project to create the final forecasting. Evaluation metrics used for the models include the mean squared error (MSE) and mean absolute error (MAE). By statistical definition, MSE is a measure of the average squared difference between predicted and actual values in a dataset, while MAE refers to the measure of the average absolute difference between predicted and actual values [55]. The forecasted concentrations and values of alkalinity, pH and SO_4 by the LSTM model were supplied to the best performing model to forecast Fe and acidity

Table 4
Structure variations of the multivariate LSTM model.

Model type	Lookback period	Epochs	Validation split
Structure variation 1	150 days	30	15%
Structure variation 2	250 days	20	20%
Structure variation 3	300 days	50	25%

Table 5
Performances (MSEs and MAEs) for the multivariate LSTM, ANN, DNN, random forest and gradient boosting models; RF: random forest, GB: gradient boosting.

	LSTM1	LSTM2	LSTM3	ANN		DNN		RF		GB	
	MSE	MSE	MSE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Training	0.0532	0.0485	0.0481	0.0337	0.1308	0.0342	0.1313	0.0094	0.0620	0.0236	0.1039
Testing	0.0851	0.0790	0.0592	0.0258	0.1190	0.0279	0.1234	0.0192	0.0798	0.0254	0.1050
Comment	Good performance.			Data leaking. Not a good generalisation model.		Data leaking. Not a good generalisation model.		Good performance.		Acceptable performance.	
Decision				Do not use.		Do not use.		Use.		Do not use.	

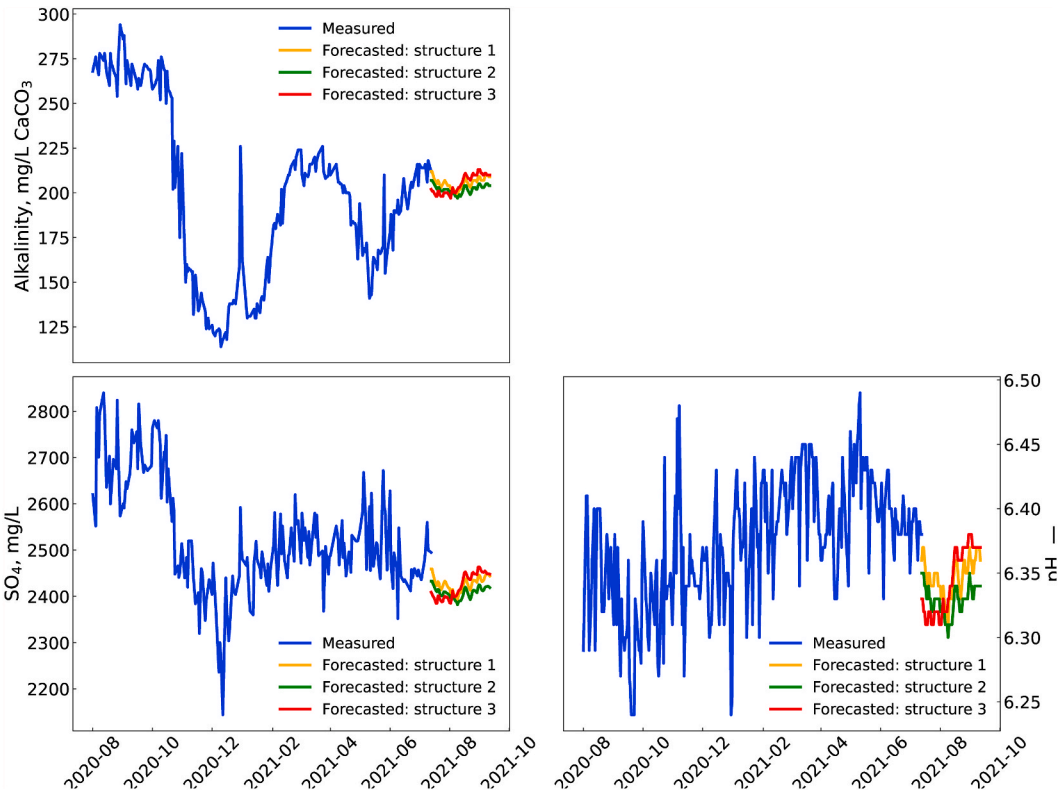


Fig. 15. Forecasted concentrations of alkalinity (top), SO_4 (bottom left) and pH values (bottom right) using multivariate LSTM model. Historical data was used from 2020 to 08-01 to allow better visualisation of the forecasting results.

concentrations. For the LSTM model, three structures were developed with different numbers of the "lookback period" which explains the number of previous time-steps the model needs to use to predict the subsequent time-step, epochs, and validation split (Table 4). A single hidden layer multivariate LSTM model with 32 memory units based on the structure variations explained was compiled, and a ReLU activation function was used throughout. For all the structure variations, the models were fitted with a batch size of 32. The adaptive moment estimation (Adam) optimiser and MSE loss function were used in this model, and the low validation and training loss implies good model fitting on the new and training data (Table 5).

Different techniques were applied to tune the hyper-parameters for the neural network models, including grid search optimisation technique, keras tuner and "trial and error" method. For the ANN model, the configuration that yielded better results was a model with

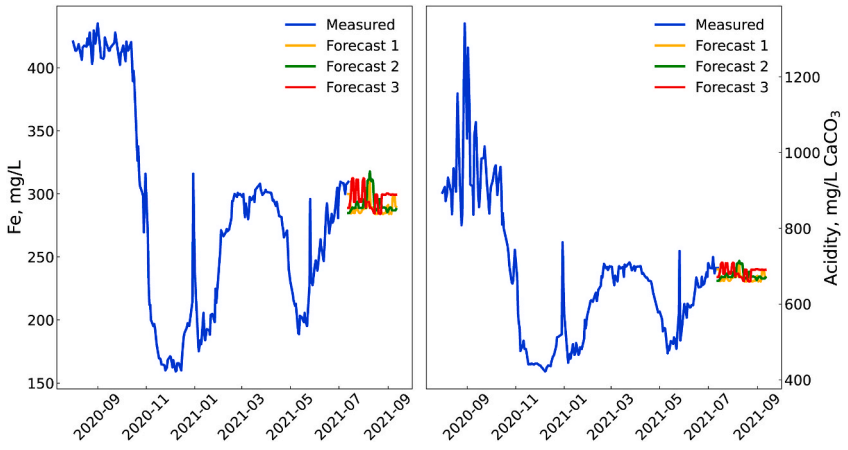


Fig. 16. Forecasted concentrations of Fe and acidity using the random forest model. Historical data was used from 2020 to 08-01 to allow better visualisation of the forecasting results.

input layers that consisted of three neurons of alkalinity, pH and SO_4, while Fe and acidity were used in the output layer with a ReLU activation function. The model had one hidden layer of four neurons and a ReLU activation function. The DNN architecture consisted of two hidden layers, the first one with four neurons and another with two neurons, both with a ReLU activation function. Like ANN, input parameters were alkalinity, pH and SO_4, with target outputs being Fe and acidity. Both the neural network models used a test size of 20%, and the models were compiled using the Adam optimiser. They were then fitted for 500 epochs with a batch size of 64. The model performances were tested using the MSE and MAE accuracy metrics (Table 5). A random forest regression tree model was built using 150 trees and a maximum depth of 8, with a test split of 20%. A gradient boosting regression tree model was compiled using 100 trees, maximum depth of 5 and learning rate of 0.05, with a test size of 20%. Regression tree model performances were also evaluated using MSE and MAE (Table 5).

8. Results and discussions

Machine learning models developed in this study consisted of a multivariate LSTM model which was used to forecast alkalinity, pH and SO_4 for 60 days using three different structure variations (Fig. 15). Furthermore, ANN, DNN, random forest and gradient boosting models were trained and tested using historical data. Random forest displayed the overall best performance and was used to forecast Fe and acidity concentrations for the same period as the LSTM model (Fig. 16). The multivariate LSTM forecasting has three different forecasted concentrations for alkalinity and SO_4 as well as the pH values, which then resulted in the random forest model also forecasting three different concentrations for Fe and acidity. With random forest using alkalinity, pH and SO_4 as input parameters, the forecasted concentrations and values by the LSTM model were fed to the trained random forest model. Therefore, the random forest model showed good performance and forecasted Fe and acidity concentrations with accuracy as the values fall within the population and follows the trend. In addition, the results suggest that ML models can be widely applied in mine water time series forecasting analysis.

The models were evaluated using MSE and MAE, and in all occasions, the error was substantially low. However, the neural network models' training loss was always greater than the validation loss, which meant that there is data leaking in the training process. Hyperparameters were rigorously tuned using grid search, keras tuner and "trial and error" approach, but training loss continued to be slightly greater than the validation loss. Therefore, the final decision was that the models cannot be used for final forecasting analysis. Regression tree models displayed better performance, with random forest performing better than the gradient boosting. Thus, random forest was used to perform final forecasting analysis. Random forest showed superior performance because it is good at handling nonlinear relationships and interactions between variables. It also has built-in mechanisms to prevent overfitting, such as bagging and random feature selection, making it more reliable than other models in situations where overfitting is a concern.

9. Comparing the measured and forecasted data for the final target outputs

Forecasted concentrations of Fe and acidity using random forest model were compared with the measured data by calculating the forecasting error (Table 6). Measured data only contains 23 observations while the forecasting period was for 60 days. This is because sampling was not carried out daily for this period, thus the forecasting error was only calculated for the available measured data. Furthermore, cross plots of measured and forecasted data with robust regression lines were computed (Fig. 17). Computed plots and calculations show both the low coefficients of determinations (r^2) and statistical significances. However, the calculated forecasting errors are relatively low with only four notable higher errors. The reason for such differences may be because of the sampling that was not conducted daily at the treatment plant.

Table 6
Forecasting error analysis for the random forest model; Error is calculated as: |[(measured–forecasted)/(measured)] × 100%|.

Measured Fe, mg/L	Forecast 1	|Error, %|	Forecast 2	|Error, %|	Forecast 3	|Error, %|	Measured Acidity, mg/L $CaCO_3$	Forecast 1	|Error, %|	Forecast 2	|Error, %|	Forecast 3	|Error, %|
309.6	300.0	3.1	284.9	8.0	288.9	6.7	726	692	4.7	661	8.9	671	7.6
307.2	299.4	2.5	285.6	7.0	293.5	4.4	724	691	4.6	663	8.4	680	6.1
306.4	286.9	6.4	288.9	5.7	312.4	2.0	724	667	7.8	671	7.3	709	2.1
301.6	284.9	5.6	289.1	4.1	293.5	2.7	724	661	8.7	672	7.2	680	6.1
304.0	288.7	5.0	293.5	3.4	311.4	2.4	728	671	7.8	680	6.6	708	2.8
296.8	285.6	3.8	288.9	2.7	295.8	0.3	702	663	5.5	671	4.4	685	2.4
263.2	284.9	8.2	288.9	9.8	293.5	11.5	662	661	0.1	671	1.4	680	2.7
307.8	284.9	7.5	288.9	6.1	295.8	3.9	742	661	10.9	671	9.5	685	7.7
304.0	286.9	5.6	288.9	5.0	295.8	2.7	708	667	5.7	671	5.2	685	3.3
289.6	288.9	0.2	295.8	2.2	312.0	7.7	700	671	4.1	685	2.2	708	1.2
280.8	288.9	2.9	293.5	4.5	295.8	5.4	700	671	4.1	680	2.8	685	2.2
284.0	288.9	1.7	295.8	4.2	293.5	3.4	710	671	5.5	685	3.5	680	4.2
287.2	293.5	2.2	310.2	8.0	293.5	2.2	710	680	4.2	706	0.6	680	4.2
276.8	288.9	4.4	311.2	12.4	289.4	4.5	680	671	1.3	707	4.0	673	1.1
278.4	288.9	3.8	310.1	11.4	288.7	3.7	684	671	1.9	706	3.2	671	1.8
264.0	289.4	9.6	310.2	17.5	286.9	8.7	648	673	3.8	706	9.0	667	3.0
256.0	284.2	11.0	287.9	12.5	299.4	16.9	610	660	8.1	670	9.8	690	13.2
255.2	284.5	11.5	286.9	12.4	299.6	17.4	614	660	7.6	667	8.7	691	12.5
248.0	284.2	14.6	287.9	16.1	299.4	20.7	600	660	10.0	670	11.7	690	15.1
197.6	289.7	46.6	286.9	45.2	299.6	51.6	518	671	29.6	667	28.9	691	33.4
198.4	299.4	50.9	286.9	44.6	299.6	51.0	530	691	30.3	667	25.9	691	30.4
208.8	299.4	43.4	286.9	37.4	299.4	43.4	540	690	27.9	667	23.6	690	27.9
208.8	290.4	39.1	286.9	37.4	299.4	43.4	536	673	25.5	667	24.5	690	28.8

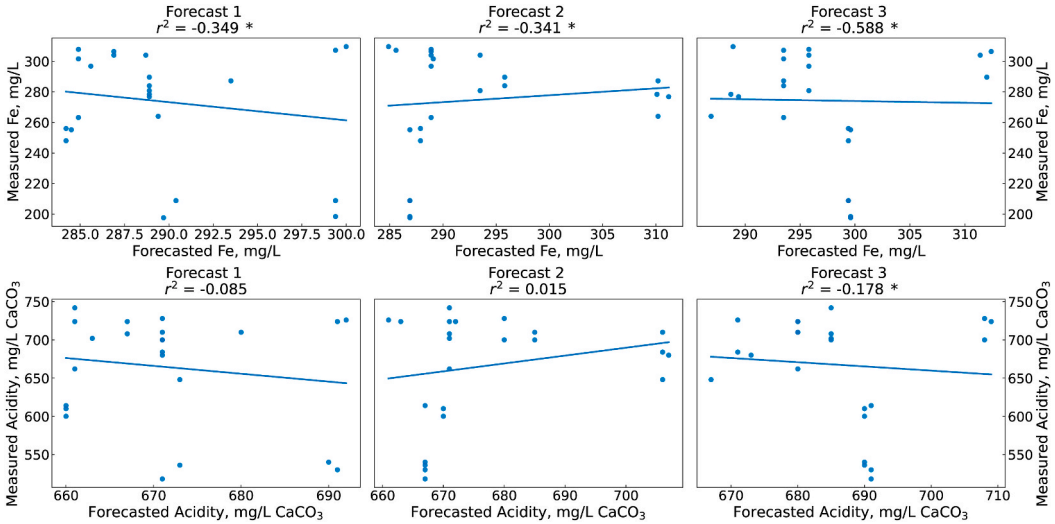


Fig. 17. Cross plots with robust regression lines comparing the measured and forecasted Fe and acidity concentrations using the random forest model results; ***$p < 0.001$, **$p < 0.01$, *$p < 0.05$.

10. Conclusions

Forecasted concentrations of Fe and acidity by the random forest model fall within the historical data population and follow its recent trend and pattern. Therefore, the proposed methodology can be applied with certainty and confidence in forecasting mine water chemistry. Machine learning forecasting approach proved that the application can use data from several parameters to forecast other parameters, i.e. the model was developed in a way that the computer learns the trends, patterns and seasonality of input data to forecast the target outputs. Parameters in a time series are related to each other and influence the outcomes in each parameter's dataset. Therefore, focusing only on one parameter to perform forecasting analysis would be inaccurate. Traditional statistical forecasting techniques such as ARIMA or Box-Jenkins, which forecast data of a parameter by learning its structure without relating it to other parameters, should be avoided in future applications. Thus, the forecasting technique proposed here will be a useful tool for water treatment plants because it will help in understanding changes in the mine water chemistry and volumes in advance.

It can be concluded that forecasting mine water chemistry by applying ML models is a relevant contribution in and addition to mine water treatment plants. Comparing the neural network and regression tree models, the results show that random forest regression tree model performed better than the other models. Finally, the results obtained in this study indicate that regression tree algorithms are powerful and important mechanisms to model and forecast the complex mine water time series data or nonlinear systems. These approaches were able to analyse the hidden patterns, trends and seasonality among the historical mine water dataset in a much better and accurate approach compared to traditional time series analysis and statistical techniques. Lastly, the findings of this study have revealed that transforming time series data before using it for modelling is sometimes necessary to achieve more accurate forecasting results.

CrediT author contribution statement

Conceptualisation: Kagiso Samuel More; Funding acquisition: Christian Wolkersdorfer; Software: Kagiso Samuel More; Supervision: Christian Wolkersdorfer; Writing – original draft: Kagiso Samuel More; Writing – review and editing: Christian Wolkersdorfer.

Funding

This work is funded and supported by the National Research Foundation (NRF Grant UID 86948 and 121723) South Africa under the SARChI Chair for Mine Water Management, the Tshwane University of Technology (TUT).

Statement

All authors certify that they have participated sufficiently in the work to take public responsibility for the content, including participation in the concept, design, analysis, writing, or revision of the manuscript.

Accountability

We have authority over manuscript preparation and decisions to submit the manuscript for publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

Thanks to the National Research Foundation (NRF Grant UID 86948 and 121723) South Africa under the SARChI Chair for Mine Water Management and the Tshwane University of Technology (TUT) for funding this project and supporting this research. Additional thanks go to Council for Geosciences (CGS), South Africa and the relevant authorities and mine operator Sibanye Gold for providing us with historical mine water data.

References

[1] C. Wolkersdorfer, E. Mugova, V.S. Daga, P. Charvet, J.R.S. Vitule, Effects of mining on surface water – case studies, in: K. Irvine, D. Chapman, S. Warner (Eds.), The Encyclopedia of Inland Waters, second ed., Elsevier, Oxford, 2022, pp. 210–224, https://doi.org/10.1016/B978-0-12-819166-8.00036-0.
[2] M. Paul, T. Metschies, M. Frenzel, J. Meyer, The mean hydraulic residence time and its use for assessing the longevity of mine water pollution from flooded underground mines, in: B. Merkel, M. Schipek (Eds.), The New Uranium Mining Boom, Springer Geology, Springer, Heidelberg, 2011, pp. 689–699, https://doi.org/10.1007/978-3-642-22122-4_79.
[3] P. Younger, The longevity of mine water pollution – a basis for decision making, Sci. Total Environ. 194–195 (1997) 457–466, https://doi.org/10.1016/S0048-9697(96)05383-1.
[4] K.S. More, C. Wolkersdorfer, N. Kang, A.E. Elmaghraby, Automated measurement systems in mine water management and mine workings — a review of potential methods, Water Resour. Ind. 24 (2020) 1–12, https://doi.org/10.1016/j.wri.2020.100136.
[5] K.S. More, C. Wolkersdorfer, Predicting and Forecasting Mine Water Parameters Using a Hybrid Intelligent System, Water Resour. Manage., 2022, https://doi.org/10.1007/s11269-022-03177-2.
[6] M. Khashei, M. Bijari, An artificial neural network (p,d,q) model for timeseries forecasting, Expert Syst. Appl. 37 (2010) 479–489, https://doi.org/10.1016/j.eswa.2009.05.044.
[7] G.P. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing 50 (2003) 159–175, https://doi.org/10.1016/S0925-2312(01)00702-0.
[8] G. Zhang, B. Eddy Patuwo, M.Y. Hu, Forecasting with artificial neural networks: the state of the art, Int. J. Forecast. 14 (1998) 35–62, https://doi.org/10.1016/S0169-2070(97)00044-7.
[9] Y. Chen, B. Yang, J. Dong, A. Abraham, Time-series forecasting using flexible neural tree model, Inf. Sci. 174 (2005) 219–235, https://doi.org/10.1016/j.ins.2004.10.005.
[10] A. Jain, A.M. Kumar, Hybrid neural network models for hydrologic time series forecasting, Appl, Soft Comput. 7 (2007) 585–592, https://doi.org/10.1016/j.asoc.2006.03.002.
[11] S. Fielding, P.M. Fayers, C.R. Ramsay, Investigating the missing data mechanism in quality of life outcomes: a comparison of approaches, Health Qual. Life Outcome 7 (2009) 57, https://doi.org/10.1186/1477-7525-7-57.
[12] T.D. Little, T.D. Jorgensen, K.M. Lang, E.W.G. Moore, On the joys of missing data, J. Pediatr. Psychol. 39 (2014) 151–162, https://doi.org/10.1093/jpepsy/jst048.
[13] D.A. Newman, Missing data: five practical guidelines, Organ. Res. Methods 17 (2014) 372–411, https://doi.org/10.1177/1094428114548590.
[14] D. Wackerly, W. Mendenhall, R.L. Scheaffer, Mathematical Statistics with Applications, seventh ed., Thomson, Belmont, 2014.
[15] R. Houari, A. Bounceur, A.K. Tari, M.T. Kecha, Handling missing data problems with sampling methods, in: International Conference on Advanced Networking Distributed Systems and Applications, Bejaia, Algeria, 2014, pp. 99–104, https://doi.org/10.1109/INDS.2014.25. Paper presented at the 2014.
[16] N.K. Manaswi, Deep Learning with Applications Using python: Chatbots and Face, Object, and Speech Recognition with TensorFlow and Keras, Apress, Berkeley, 2018, https://doi.org/10.1007/978-1-4842-3516-4.
[17] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735.
[18] T. Ekemen Keskin, E. Özler, E. Şander, M. Düğenci, M.Y. Ahmed, Prediction of electrical conductivity using ANN and MLR: a case study from Turkey, Acta Geophys. 68 (2020) 811–820, https://doi.org/10.1007/s11600-020-00424-1.
[19] H.R. Maier, N. Morgan, C.W.K. Chow, Use of artificial neural networks for predicting optimal alum doses and treated water quality parameters, Environ. Model. Software 19 (2004) 485–494, https://doi.org/10.1016/S1364-8152(03)00163-4.
[20] S. Øyen, Forecasting Multivariate Time Series Data Using Neural Networks [Master of Science: Cybernetics and Robotics], Norwegian University of Science and Technology, Trondheim, 2018.
[21] A. Krenker, J. Bester, A. Kos, Introduction to the artificial neural networks, in: K. Suzuki (Ed.), Artificial Neural Networks — Methodological Advances and Biomedical Applications, InTech Open, Rijeka, 2011, https://doi.org/10.5772/15751.
[22] G. Biau, E. Scornet, A random forest guided tour, Test 25 (2016) 197–227, https://doi.org/10.1007/s11749-016-0481-7.
[23] Y. Zhang, A. Haghani, A gradient boosting method to improve travel time prediction, Transport. Res. C Emerg. Technol. 58 (2015) 308–324, https://doi.org/10.1016/j.trc.2015.02.019.
[24] A.Z. Averbuch, P. Neittaanmäki, V.A. Zheludev, Spline and Spline Wavelet Methods with Applications to Signal and Image Processing, Springer, Cham, 2014, https://doi.org/10.1007/978-3-319-22303-2.
[25] Y. Kineri, M. Wang, H. Lin, T. Maekawa, B-spline surface fitting by iterative geometric interpolation/approximation algorithms, Comput. Aided Des. 44 (2012) 697–708, https://doi.org/10.1016/j.cad.2012.02.011.
[26] S. Baydas, B. Karakas, Defining a curve as a Bezier curve, J. Taibah Univ. Sci. 13 (2019) 522–528, https://doi.org/10.1080/16583655.2019.1601913.
[27] X.-A. Han, Y. Ma, X. Huang, A novel generalization of Bézier curve and surface, J. Comput. Appl. Math. 217 (2008) 180–193, https://doi.org/10.1016/j.cam.2007.06.027.
[28] M. Hoffmann, I. Juhász, Shape Control of Cubic B-Spline and NURBS Curves by Knotmodifications, Paper presented at the 14th International Conference Information Visualisation, Washington DC, 2001, pp. 63–68, https://doi.org/10.1109/IV.2001.942040.
[29] M. Elbanhawi, M. Simic, R.N. Jazar, Continuous path smoothing for car-like robots using B-spline curves, J. Intell. Rob. Syst. 80 (2015) 23–56, https://doi.org/10.1007/s10846-014-0172-0.
[30] C. Shao, L. Xiao, Nurbs model for chaotic time series, in: Paper Presented at the 3rd International Conference on Computer Research and Development, Shanghai vol. 4, 2011, pp. 135–138.

[31] D.F. Rogers, An Introduction to NURBS: with Historical Perspective, Elsevier, Amsterdam, 2001, https://doi.org/10.1016/B978-1-55860-669-2.X5000-3.
[32] R. Sevilla, S. Fernández-Méndez, A. Huerta, NURBS-enhanced finite element method (NEFEM), Int. J. Numer. Methods Eng. 76 (2008) 56–83, https://doi.org/10.1002/nme.2311.
[33] R. Boudjemaa, A. Forbes, P. Harris, S. Langdell, Multivariate Empirical Models and Their Use in Metrology, Software Support for Metrology Programme, Middlesex, 2003, pp. 1–61.
[34] C. Torrence, G.P. Compo, A practical guide to wavelet analysis, Bull. Am. Meteorol. Soc. 79 (1998) 61–78, https://doi.org/10.1175/1520-0477 (1998)079.
[35] E.W. Hansen, Fourier Transforms: Principles and Applications, Wiley, Hoboken, 2014.
[36] V. Serov, Fourier Series, Fourier Transform and Their Applications to Mathematical Physics, Springer, New York, 2017, https://doi.org/10.1007/978-3-319-65262-7.
[37] T. Baba, Time-frequency analysis using short time Fourier transform, Open J. Acoust. 5 (2012) 32–38, https://doi.org/10.2174/1874837601205010032.
[38] K. Veer, R. Agarwal, Wavelet and short-time Fourier transform comparison-based analysis of myoelectric signals, J. Appl. Stat. 42 (2015) 1591–1601, https://doi.org/10.1080/02664763.2014.1001728.
[39] B. Boulet, Fundamentals of Signals and Systems, Thomson Learning, Boston, 2006.
[40] T.K. Hon, Time-Frequency Analysis and Filtering Based on the Short-Time Fourier Transform [Doctor of Philosophy], King's College London, London, 2013.
[41] E.W. Bolton, K.A. Maasch, J.M. Lilly, A wavelet analysis of Plio-Pleistocene climate indicators: a new view of periodicity evolution, Geophys. Res. Lett. 22 (1995) 2753–2756, https://doi.org/10.1029/95GL02799.
[42] A. Graps, An introduction to wavelets, IEEE Comput. Sci. Eng. 2 (1995) 50–61, https://doi.org/10.1109/99.388960.
[43] G. Lord, E. Pardo-Igúzquiza, I. Smith, A Practical Guide to Wavelets for Metrology, Software Support for Metrology Programme, Middlesex, 2000, pp. 1–44.
[44] P. Qi, G. Zhang, Y.J. Xu, L. Wang, C. Ding, C. Cheng, Assessing the influence of precipitation on shallow groundwater table response using a combination of singular value decomposition and cross-wavelet approaches, Water 10 (2018) 598, https://doi.org/10.3390/w10050598.
[45] G. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, A. O'Leary, PyWavelets: a Python package for wavelet analysis, J. Open Source Softw. 4 (2019) 1237, https://doi.org/10.21105/joss.01237.
[46] N.B. Chikodili, M.D. Abdulmalik, O.A. Abisoye, S.A. Bashir, Outlier detection in multivariate time series data using a fusion of K-medoid, standardized euclidean distance and Z-score, in: Paper Presented at the International Conference on Information and Communication Technology and Applications, Minna, Nigeria, 2020, pp. 259–271, https://doi.org/10.1007/978-3-030-69143-1_21.
[47] V. Kozitsin, I. Katser, D. Lakontsev, Online forecasting and anomaly detection based on the ARIMA model, Appl. Sci. 11 (2021) 3194, https://doi.org/10.3390/app11073194.
[48] Q. Yu, L. Jibin, L. Jiang, An improved ARIMA-based traffic anomaly detection algorithm for wireless sensor networks, Int. J. Distributed Sens. Netw. 12 (2016), 9653230, https://doi.org/10.1155/2016/9653230.
[49] B. Lindemann, B. Maschler, N. Sahlab, M. Weyrich, A survey on anomaly detection for technical systems using LSTM networks, Comput. Ind. 131 (2021), 103498, https://doi.org/10.1016/j.compind.2021.103498.
[50] M. Xie, T. Goh, X. Tang, Data transformation for geometrically distributed quality characteristics, Qual. Reliab. Eng. Int. 16 (2000) 9–15, https://doi.org/10/ctw9qp.
[51] S. Manikandan, Data transformation, J. Pharmacol. Pharmacother. 1 (2010) 126, https://doi.org/10.4103/0976–500X.72373.
[52] A. Azzalini, A. Capitanio, Statistical applications of the multivariate skew normal distribution, J. Roy. Stat. Soc. B. 61 (1999) 579–602, https://doi.org/10.1111/1467-9868.00194.
[53] F. Zhang, I. Keivanloo, Y. Zou, Data transformation in cross-project defect prediction, Empir. Software Eng. 22 (2017) 3186–3218, https://doi.org/10.1007/s10664-017-9516-2.
[54] C. Habermann, F. Kindermann, Multidimensional spline interpolation: theory and applications, Comput. Econ. 30 (2007) 153–169, https://doi.org/10.1007/s10614-007-9092-4.
[55] J.C. Davis, Statistics and Data Analysis in Geology, third ed., Wiley & Sons, New York, 2002.