Forecasting and Optimizing Dual Media Filter Performance via Machine Learning

Author:
Moradi, Sina; Omar, Amr; Zhou, Zhuoyu; Agostino, Anthony; Gandomkar, Ziba; Bustamante, Heriberto; Power, Kaye; Henderson, Rita; Leslie, Greg

Publication details:
Water Research
v. 235
Medium: Print-Electronic
0043-1354 (ISSN); 1879-2448 (ISSN)

Publication Date:
2023-03

Publisher DOI:
https://doi.org/10.1016/j.watres.2023.119874

License:
https://creativecommons.org/licenses/by/4.0/
Link to license to see what you are allowed to do with this resource.

Downloaded from http://hdl.handle.net/1959.4/unsworks_83084 in https://unsworks.unsw.edu.au on 2024-05-18

Forecasting and Optimizing Dual Media Filter Performance via Machine Learning

Sina Moradi[1,2], Amr Omar[3], Zhuoyu Zhou[4], Anthony Agostino[1], Ziba Gandomkar[5], Heriberto Bustamante[6], Kaye Power[6], Rita Henderson[1,3], Greg Leslie[1,2,3*]

[1] Algae & Organic Matter Laboratory, School of Chemical Engineering, University of New South Wales, Sydney 2052, Australia
[2] UNESCO Centre for Membrane Science & Technology, School of Chemical Engineering, University of New South Wales, Sydney 2052, Australia
[3] School of Chemical Engineering, University of New South Wales, Sydney 2052, Australia
[4] School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, 518172, China
[5] Discipline of Medical Imaging Sciences, Faculty of Medicine and Health, University of Sydney, Sydney 2006, Australia
[6] Sydney WaterCorporation, Sydney, Australia

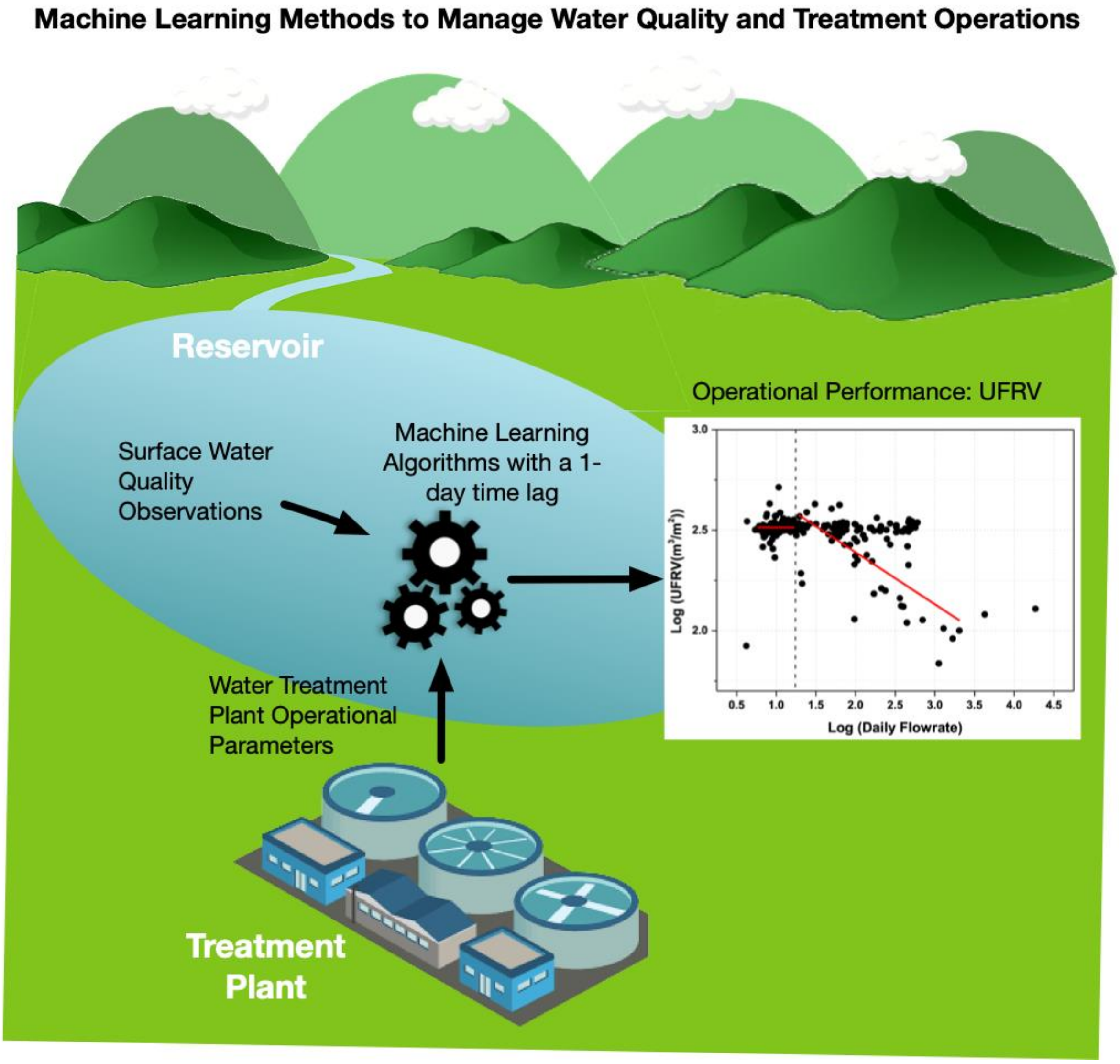
Abstract

Four different machine learning algorithms, including Decision Tree (DT), Random Forest (RF), Multivariable Linear Regression (MLR), Support Vector Regressions (SVR), and Gaussian Process Regressions (GPR), were applied to predict the performance of a multi-media filter operating as a function of raw water quality and plant operating variables. The models were trained using data collected over a seven year period covering water quality and operating variables, including true colour, turbidity, plant flow, and chemical dose for chlorine, $KMnO_4$, $FeCl_3$, and Cationic Polymer (PolyDADMAC). The machine learning algorithms have shown that the best prediction is at a 1-day time lag between input variables and unit filter run volume (UFRV). Furthermore, the RF algorithm with grid search using the input metrics mentioned above with a 1-day time lag has provided the highest reliability in predicting UFRV with a RMSE and R^2 of 31.58 and 0.98, respectively. Similarly, RF with grid search has shown the shortest training time, prediction accuracy, and forecasting events using a ROC-AUC curve analysis (AUC over 0.8) in extreme wet weather events. Therefore, Random Forest with grid search and a 1-day time lag is an effective and robust machine learning algorithm that can predict the filter performance to aid water treatment operators in their decision makings by providing real-time warning of the potential turbidity breakthrough from the filters.

Keywords: Filtration Performance, Machine Learning Approach, Hyper-parameter Optimisation, Unit Filter Run Volume

[*] Corresponding author: Greg Leslie, E-mail address: g.leslie@unsw.edu.au

Graphical Abstract


Machine Learning Methods to Manage Water Quality and Treatment Operations
Reservoir
Surface Water Quality Observations
Machine Learning Algorithms with a 1-day time lag
Water Treatment Plant Operational Parameters
Treatment Plant
Operational Performance: UFRV
Log (UFRV(m³/m²))
3.0
2.5
2.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
Log (Daily Flowrate)

Nomenclature

Symbols			
C	Cost of Constraints Violation	MLR	Multivariable Linear Regression
$Coef_0$	Constant Parameter in the Sigmoid or Polynomial Kernel Function	MSE	Mean Square Error
Cp	Complexity Parameter	NOM	Natural Organic Matter
FN	False-Negative	NTU	Nephelometric Turbidity Unit
FP	False-Positive	PolyDADMAC	Cationic Polymer
FPR	False-Positive Rate	RBF	Radial Basis Function
IQR	Interquartile Range	RF	Random Forest
$K_{(x_i,x_j)}$	Kernel Function	RMSE	Root Mean Square Error
mtry	Number of candidate variables considered at each split	ROC	Receiver Operator Characteristics
p	Number of variables in the input matrix	RS	Random Search

Q	Quartile	SVR	Support Vector Regression
TN	True-Negative	TOC	Total Organic Carbon
TP	True-Positive	UFRV	Unit Filter Run Volume
TPR	True-Positive Rate	WFP	Water Filtration Plant
x_i, x_j	Data Points		
$\widehat{y_l}$	MLR Model Prediction Value	**Greek Symbols**	
		β_0	Intercept Coefficient
Abbreviations		β_i	Regression Coefficient
AUC	Area Under the Curve	$\hat{\beta}$	Optimal Regression Parameter
DMG	Dual Media Gravity	γ	Kernel Function Coefficient
DOC	Dissolved Organic Carbon	ΔT	Time Lag
DT	Decision Tree	ε	Random Error
GPR	Gaussian Process Regression	λ	Penalty Term
GS	Grid Search	φ	Mapping to a high dimensional feature space factor

1 Introduction

Extreme weather events, such as intense and frequent heavy rain events, can affect water catchments and their performance due to the increased concentrations of suspended materials, natural organic matter (NOM), and inorganic substances in source waters [1, 2]. It has been shown that the water treatment plant's production rate can decrease by ~40% due to having weaker flocculants because of higher NOM in the feedwater after heavy rainfall [3]. These impose additional burdens on water treatment plants, requiring additional maintenance, chemical use, and waste production [4]. Hence, the use of a reliable forecasting model for filter performance can, not only help in controlling the performance of the water treatment plant, but also in predicting the production efficiency depending on the influent water quality.

Filter performance can be expressed in terms of daily water production, filter run time, unit filter run volume (UFRV), effluent turbidity, and pressure head loss. Of these, UFRV—defined as the volume of water filtered through a unit surface area—is a useful way of normalising performance for daily water production and run time, which depending on the loading rate, can range from 200 $\mathrm{m}^3/\mathrm{m}^2$ to 400 $\mathrm{m}^3/\mathrm{m}^2$ [5]. Understanding changes in UFRV as a function of feedwater quality and plant operating variables, such as coagulant dose, provides an effective and robust way to combine large data sets into useful information on plant performance and potentially provide real-time warning of problems such as turbidity breakthroughs from the filters. Given the complexity, nonlinearity, and numerous variables in the filtration models, developing data-driven artificial intelligence (AI) models have been applied for water quality monitoring as they improve the prediction capability for accurately assessing water quality parameters [6-9].

Decision tree (DT) algorithms involve real-time water quality monitoring from multivariate data collected from different sensors [10, 11]. This identifies high-quality ground-water zones [12], flood modelling [13], and algal growth prediction [14]. It was reported that an improved DT learning method could forecast and evaluate the water quality of Chao Lake in Hong Kong and can assess the trophic status of Poyang Lake in China [6, 7]. Furthermore, the predictive performance of DT models can be considerably improved by aggregating many independent individual trees, known as random forests (RF) [15]. The random forest algorithm employs the bootstrap aggregation process to combine a set of DTs, where each tree is constructed using the best split for each node among a subset of randomly chosen predictors. RF averages noisy (but unbiased models) to reduce the prediction variance to mitigate the DT algorithm's poor performance. RF provides a multivariate, nonparametric, and nonlinear regression, where the final prediction in regression is made by averaging each tree's prediction [15, 16]. Training RF models are less computationally expensive compared to other machine learning algorithms [17] as they are straightforward to use [18] and can handle highly correlated predictor variables [17]. Other advantages of the RF algorithm are reducing variance and consistency, while not increasing the bias of the predictions [19, 20]. Recently, a holistic review of the implementation of the RF algorithm in water resource applications has shown the effectiveness of these models for prediction and inference purposes in water resources [21].

Other promising machine learning techniques are the support vector regression (SVR) and Gaussian process regression (GPR), especially in environmental studies, from soil moisture prediction [22] to rapid detection of organic contamination events in the water distribution systems [23], and coagulant dosage prediction in water treatment plants [24]. These algorithms essentially benefit from the "kernel trick", which efficiently maps the data into a high-dimensional feature space using a nonlinear function [8]. Despite the good predictive performance, SVR and GPR algorithms are sensitive to the choice of hyper-parameters (i.e., parameters that cannot be inferred during the model's training). Hence, selecting proper kernel functions and hyper-parameters is crucial in applying SVR and GPR algorithms to deliver correct results [25, 26]. In general, the main hyper-parameter selection techniques are gradient-based approaches [27], such as (1) grid search (GS), where the search space of parameters is split into groups of possible parameters to be tested uniformly [28], and (2) random search (RS), where possible values for parameters are randomly picked, which exhibited more efficient in high-dimensional search spaces [25, 29]. In a previous study, we compared different machine learning algorithms (i.e., multivariable linear regression (MLR), SVR, and GPR with different kernel functions) to quantify variations in NOM in the raw water reservoir as a function of climatological and water quality factors [30]. Four independent variables: (1) precipitation, (2) temperature, (3) leaf area index, and (4) turbidity, were selected to develop and train each machine learning model. It was found that model accuracy was very sensitive to the time-lag function, which is used to average climate observations prior to pairing with DOC observations. The SVR model with a quadratic kernel function and a 12-day time-lag function provided the highest reliability in predicting the DOC observations with a RMSE and R^2 of 1.9 and 0.71, respectively.

In this study, we extend the machine learning approach to predict filter performance in terms of UFRV in a treatment plant operation on the same drinking water reservoir. The main objective of this paper is to employ different machine learning algorithms, including multivariable linear regression, decision tree algorithm, random forests, support vector regression, and Gaussian process regression, to predict the filter performance in a water filtration plant in terms of influent water quality conditions. Random-search and grid-search procedures are used to tune the hyper-parameters of the different kernel functions embedded in SVR and GPR to predict filter performance. The results of regression and classification models were discussed in terms of error rates and classification precision.

2 Methodology

2.1 Dataset

The data were extracted from the Supervisory control and data acquisition (SCADA) system of the Nepean Water Filtration Plant (WFP), located in south Sydney, Australia. The Nepean WFP treats surface water from the Nepean reservoir by pre-oxidation (chlorine and $KMnO_4$), coagulation, and flocculation. Two-step filtration processes are used: roughing filtration and dual media gravity (DMG) filtration, followed by final chlorine disinfection. DMG filters at Nepean WFP consist of two types of tightly packed filtering materials: a layer of anthracite coal (media depth: 600mm), and layers of fine sand (media depth: 300mm) and gravel (media depth: 75mm). The plant uses ferric chloride ($FeCl_3$) as the primary coagulant and polyDADMAC as a secondary coagulant [31]. The data consisted of long timescale measurements of physicochemical water quality parameters that include and are not limited to turbidity, dissolved organic carbon, color, and pH, as well as filter performance indicator as UFRV. This dataset was obtained from August 2014 to May 2020, including data from a heavy rainfall event that happened in February 2020.

The Nepean catchment areas could receive more than 100 mm of rainfall over one month [32]. In extreme weather events, such as the period between the 7th and the 10th of February 2020, the catchment received 390 mm of rain [33]. In such a flash-flooding event, the inflows to the WFP could peak at 80 – 100 NTU. Consequently, an extreme rainfall event could impose a serious challenge on filtration performance to meet the drinking water Guidelines.

2.2 Data processing

Data integration, outliers removal, and feature selection are implemented to process the source data for subsequent modelling. In addition, influent water quality data and chemical dosing parameters were integrated as a model input, and the filter performance indicators (e.g., UFRV) were used as the model output.

2.2.1 Outlier detection and boxplot analysis

Boxplot analysis was conducted as it provides insightful visualization for outlier detection. For each variable, possible outliers were labelled by computing the interquartile range (IQR) [34]. Any data points less than or greater than the lower and upper fences, respectively, were eliminated (see Eq 1 and Eq 2, where Q_1 and Q_3 are the First and Third quartiles, respectively) [34].

$$Outlier = Q_1 - 1.5 \times IQR \qquad \text{Eq 1}$$

$$Outlier = Q_3 + 1.5 \times IQR \qquad \text{Eq 2}$$

Note that a cautious approach was used when removing the outliers as it could negatively affect the model accuracy. For example, data from heavy rainfall events might be categorized as outliers using the boxplot analysis. As such, this data was not removed in the outlier removal process and was still used in the data processing. On the other hand, a grouped outlier set due to unavoidable device errors or incorrect measurements was regarded as an outlier and removed from the dataset.

2.2.2 Correlation Analysis

Pearson and Spearman correlation analyses were applied to find the correlation coefficient between the different parameters to exclude multi-collinearity and to extract the possible relationships between each parameter. Appropriate input variables were selected beforehand to lower the computational time and avoid overfitting the model to the training data.

2.3 Establishing the Machine Learning algorithms

Multivariable linear regression, decision tree, random forest, support vector regression, and Gaussian process regression algorithms were employed to find the best machine learning algorithm that better estimates the filtration performance. The receiver operating characteristic (ROC) was conducted to investigate whether the developed machine learning models could predict extreme water quality events. To visualize the performance of the multi-class classification problem, the area under the curve (AUC) was computed by generating a confusion matrix, and the true-positive rate (TPR) and false-positive rate (FPR) metrics were calculated as follows:

$$TPR = \frac{TP}{TP + FN} \qquad \text{Eq 3}$$

$$FPR = \frac{FP}{FP + TN} \qquad \text{Eq 4}$$

where TP and FP are true-positives, representing the correctly predicted extreme events with UFRV < 150 m^3/m^2, and false-positives, representing the model's false alarms where UFRV < 150 m^3/m^2, respectively. In contrast, TN and FN are true-negatives, representing the correct predictions of normal treatment conditions with UFRV > 150 m^3/m^2, and false-negatives, representing the failures to predict the occurrence of extreme events (i.e., failed alarms).

The ROC curve illustrates the trade-off between the TPR and FPR for a suite of possible thresholds [35]. The AUC values could vary between 0.5 and 1, where values near 1 suggest excellent performance and values near 0.5 denote poor forecasting accuracy, not differing from random-classifier [36, 37].

2.3.1 Multivariable Linear Regression

The multivariable linear regression is a baseline model to evaluate the added benefit of using a more complex model than the conventional linear models. Eq 5 represents the relationship between the dependent variable (UFRV), and the k independent variables, using the MLR model.

$$\widehat{UFRV} = \beta_0 + \sum_{i=1}^{k} (\beta_i x_i) + \varepsilon \qquad \text{Eq 5}$$

β_0 is the intercept coefficient, β_i is the regression coefficient, and ε is the random error. To estimate the regression coefficients, the ordinary least squares were used to find the parameters that minimize the model's mean squared error (MSE) of the model, as implied by Eq 6.

$$\hat{\beta} = \arg\min \left(\sum (y_i - \hat{y_i})^2\right) \qquad \text{Eq 6}$$

The UFRV is denoted as $\hat{y} = \{\hat{y}_i | i = 1, 2, \dots n\}$), $\hat{\beta}$ represents the optimal regression parameter, and $\hat{y_i}$ represents the MLR model prediction value calculated by Eq 5.

2.3.2 Decision Tree

Although the decision tree method has many advantages, such as fast calculation speed, high efficiency, and relatively insensitive to missing values [11], they are considered noisy models [38] and tend to overfit the model to the training samples [39]. Intuitively, the tree construction does not continue beyond the current node if the cost of adding another branch from the current node is higher than the complexity parameter (Cp), which is calculated as:

$$Cp = \sum_{terminal\ nodes} Missclass_i + \lambda \times (split) \qquad \text{Eq 7}$$

where λ is the penalty term, also known as the regularization rate that is used to tune the over impact of regularization on the complexity error. A Cp value of 1 represents a tree with only 1 split that does not account for variable interactions. In this work, the optimal value for Cp was determined for each model using the hyperparameter optimization techniques discussed in section 2.4.1.

2.3.3 Random Forests

The "*mtry*" parameter (i.e., the number of candidate variables considered at each split) is optimized using grid and random search techniques to run the RF algorithm. The *mtry* default value was selected at p/3, where p is the number of variables in the input matrix [15].

2.3.4 Kernel-based regression models

The kernel function denotes an inner product in feature space and is represented as:

$$K_{(x_i, x_j)} = \varphi(x_i)\phi(x_j) \qquad \text{Eq 8}$$

Where φ is the mapping to a high dimensional feature space. Choosing the right kernel function and fine-tuning its hyper-parameters depends on the problem and the information extracted. Consequently, the predicted filter performance (UFRV denoted as $\hat{y} = \{\hat{y}_i | i = 1, 2, \dots n\}$) is determined as:

$$\hat{y} = \sum_{i=1}^{N} \alpha_i K_{(x_i, x)} + b \qquad \text{Eq 9}$$

Support vector regression and gaussian process regression were used as Kernel-based regression models. The SVR model can concurrently minimize model dimensions and estimation errors, while having decent generalization ability and less prone to over-fitting [40]. GPR requires a relatively small training data set that includes stability, flexibility, generalization capacity, and flexible kernel functions [18, 41]. In this study, the kernel functions used for the SVR and GPR algorithms and their adjustable hyper-parameters are presented in Table 1.

2.3.5 Adjustable hyper-parameters

Table 1 lists commonly used kernel functions for SVR and GPR, and the adjustable parameters for selected machine-learning algorithms. In these kernel functions, x_i and x_j are two different data points in the training data set. Degree defines the dimension of the polynomial function, and Coef_0 is the constant parameter in the sigmoid or polynomial kernel function. The gamma parameter (γ) defines how far the influence of a single training example reaches. Cost (C) is the cost of constraints violation, which means when the value is small, the penalty for misclassification is reduced, which means having a strong generalization ability. For example, in the application of SVR with kernel function, the hyperparameters, namely γ (the coefficient of the kernel function) and C (the regularisation parameter of the optimisation problem) are key points in the training process of the SVR model. The hyperparameter C controls the trade-off between minimising the model's complexity and minimising the training error. The hyper-parameter γ represents the width of the Radial Basis Function (RBF) kernel, and it determines whether the model will tend to over-fit the training data or it would make the model not flexible enough for complex function approximation [42].

Table 1: Set of common kernel functions used in this study for selected machine learning algorithms along with their adjustable parameters

Machine learning algorithm	Kernel	Equation	Adjustable parameters
SVR	Linear kernel	$K_{(x_i,x_j)} = x_i \cdot x_j + c$	
	Polynomial kernel	$K_{(x_i,x_j)} = \left(\alpha x_i \cdot x_j + c\right)^d$	Degree, scale, offset
	Radial basis function (RBF) kernel	$K_{(x_i,x_j)} = exp\left(-\gamma\left(x_i - x_j\right)^2\right)$	Gamma, cost
	Sigmoid kernel	$K_{(x_i,x_j)} = \tanh\,(\alpha x_i \cdot x_j + c)$	Scale, offset
GPR	RBF kernel	$K_{(x_i,x_j)} = exp\left(\frac{-1}{2\sigma^2}\left(x_i - x_j\right)^2\right)$	Sigma
	Laplacian Kernel	$K_{(x_i,x_j)} = exp\left(-\frac{\lVert x_i - x_j \rVert}{\sigma}\right)$	Sigma
	Hyperbolic Tangent kernel	$K_{(x_i,x_j)} = \tanh\,(\alpha x_i \cdot x_j + c)$	Scale, offset
	ANOVA Kernel	$K_{(x_i,x_j)} = \sum_{k=1}^{n} \exp\left(-\sigma\left({x_i}^k - {x_j}^k\right)^2\right)^d$	Sigma, degree
	Bessel Kernel	$K_{(x_i,x_j)} = \frac{J_{v+1}(\sigma\lVert x_i - x_j \rVert)}{\lVert x_i - x_j \rVert^{-n(v+1)}}$	Sigma, order, degree
DT	NA	NA	Cp
RF	NA	NA	mtry

The mapping that the kernel functions are represented to transform the non-linear input space to a high-dimensional feature space where linear regression is possible depends on the intrinsic topological structure of the data. This requires the kernel type and hyper-parameters to be optimised to approximate the ideal mapping [43]. This study focused on commonly used kernel functions, namely, the RBF, the polynomial, the sigmoid (hyperbolic tangent), the laplacian, and the Bessel kernel for SVR and GPR as supervised machine learning algorithms (outlined in Table 1). The predictive performance of SVR and GPR machine learning algorithms depends exclusively on the suitability of the selected hyper-parameters. While hyperparameter tuning has widely been applied to find a good combination of control parameters in the model, there has yet to be much discussion on which hyperparameter optimisation technique is best in machine learning model development.

As there is no exact method to obtain the best possible set of hyper-parameters, search algorithms are usually applied to find the optimal set of hyper-parameters [44-46]. Hence, in this paper, two separate search algorithms, random search and grid search, were implemented in combination with a 10-fold cross-validation procedure on each candidate parameter vector for adjusting hyper-parameters of SVR and GPR to guarantee the maximum possible quality of the final machine learning algorithms. It should be noted that grid search and random search were selected as hyperparameter optimisers as they are among the more popular methods for hyperparameter optimisation. Other hyperparameter optimisation methods, such as the Bayesian optimisation method used in other water treatment plant applications [47] were considered, but required biased inputs, such as operator experience, were not used in this study which focussed only water quality and filter performance data. could be explored in future studies. The selection of the initial ranges of parameters is a common problem in both grid search and random search algorithms, which can be selected either based on experience with the regression problem studied, or using large ranges of parameters [28]. The usage of large parameter ranges implies an increase in the search space and the training time of the machine learning algorithms. Table 4 shows the range of values for hyper-parameters explored in this work [25, 28, 48-51]. The results of the random search hyper-parameter optimisation algorithm were compared to the best results found in the grid search.

2.4 Time-lag function

All data indicators (i.e., operational data, chemical dosing, water quality parameters, and filtration performance) were temporally paired with each other. Theoretically, it takes as long as the residence time for water to go through different steps in a water filtration plant and pass through the filters. In reality, however, the water quality conditions and treatment regime at a given time might not lead to the recorded filter performance at that same time. Hence, a time-lag function is used to represent the delay between recorded filter performance indicators, and the associated operational data and water quality conditions. Time lags from zero up to three days for model input variables were considered. When the time lag is zero (ΔT=0), the filter performance indicators are in sync with the time of the model input variables. Whereas when the time lag is two (ΔT=2), the model input variables are for a time that is 2 days ahead of the time of the filtration performance indicators. In other words, the model simply uses the previous value as the prediction for the future.

$$\hat{y}^{T} = \sum_{i=1}^{N} \alpha_i K_{(x_i,x)}{}^{T-\Delta T} + b \qquad \text{Eq 10}$$

As the time lag was unknown, the performance of machine learning algorithms was evaluated as a function of time lag, while the lag that provided the strongest performance was selected.

2.5 Performance and accuracy assessment of machine learning algorithms

The framework to train, validate, and test the machine learning algorithms to predict filtration performance is presented in Figure 1. A 10-fold cross-validation was utilised for performance assessment of machine learning algorithms in which the training data are divided into 10 subsets of approximately equal size [30]. The resulting machine learning models are established by training on nine subsets, and one subset was retained to test the model. The procedure is repeated 10 times with each subset used for testing once, while the error of the machine learning model is determined by averaging the test errors over the 10 trials. The root mean squared error (RMSE) and mean square error (MSE) were employed as the statistical indicators for estimation performance. This approach was used to model plant performance over a range of conditions from normal to extreme. The extreme events occur over short temporal scales and are infrequent and there is insufficient data to train and independently test the model.

Once the optimum machine learning model and the optimal time lag were determined, the final model could be used to optimise the plant performance by changing the influencing treatment parameters. To evaluate whether developed models can predict the events with low filtration performance, the ROC and AUC were used by plotting the true-positive rate versus the false-positive rate at different thresholds. The values of AUC-ROC range between 0.5 and 1, where 1 denotes a model's excellent forecast capacity and 0.5 indicates its poor predictive accuracy [52]. This was programmed, trained, and tested using Matlab by MathWorks [53].

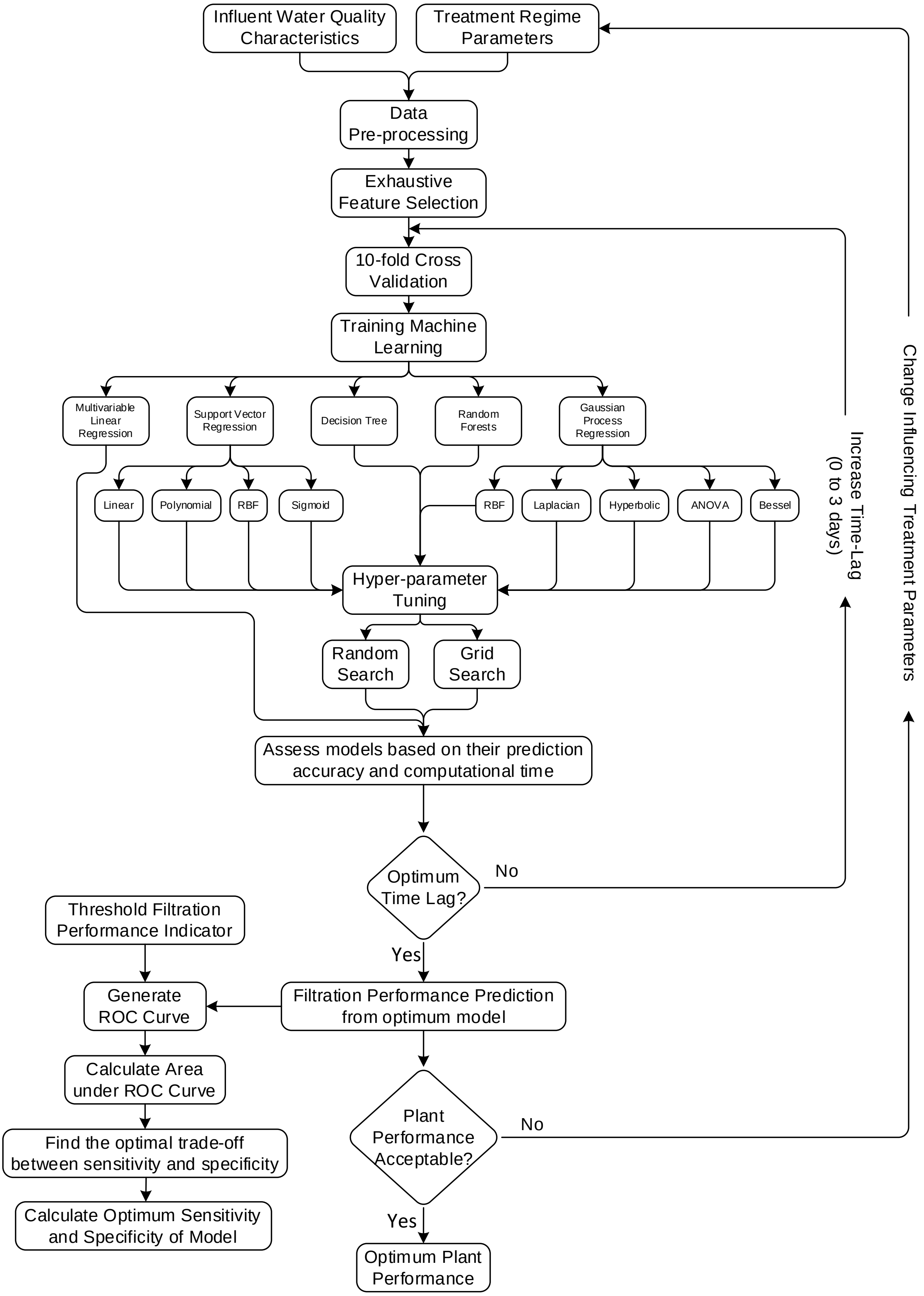

Influent Water Quality Characteristics
Treatment Regime Parameters
Data Pre-processing
Exhaustive Feature Selection
10-fold Cross Validation
Training Machine Learning
Multivariable Linear Regression
Support Vector Regression
Decision Tree
Random Forests
Gaussian Process Regression
Linear
Polynomial
RBF
Sigmoid
RBF
Laplacian
Hyperbolic
ANOVA
Bessel
Hyper-parameter Tuning
Random Search
Grid Search
Assess models based on their prediction accuracy and computational time
Optimum Time Lag?
No
Yes
Increase Time-Lag (0 to 3 days)
Change Influencing Treatment Parameters
Threshold Filtration Performance Indicator
Generate ROC Curve
Filtration Performance Prediction from optimum model
Calculate Area under ROC Curve
Plant Performance Acceptable?
No
Find the optimal trade-off between sensitivity and specificity
Calculate Optimum Sensitivity and Specificity of Model
Yes
Optimum Plant Performance

Figure 1: The overall framework to train, validate, and test the machine learning algorithms to predict filtration performance

3 Results and Discussion

3.1 Data processing

The Pearson correlation analysis observations are listed in Table 2. It was noticed that UFRV was not influenced by temperature (r=0.02), pH (r=-0.03), total Mn (r=0.02), and alkalinity (r=0.05) (Table 2). Hence, the parameters with no correlations were excluded as potential variables. DOC/TOC was also removed as potential explanatory variables because the number of measurements was not enough and considering them as variables meant that Dissolved Organic Carbon (DOC) and Total Organic Carbon (TOC) data had to be removed as explanatory variables data. Parameters with a correlation coefficient of more than 0.4 with the target variable (UFRV) were selected as independent potential input variables. To comprehensively consider the amount of data in the dataset, $KMnO_4$ was also considered as an input parameter to investigate the potential effects of oxidants on filtration performance. Hence, UFRV was correlated with seven parameters: (1) true colour, (2) turbidity, (3) flow, (4) chlorine, (5) $KMnO_4$, (6) $FeCl_3$, and (7) Cationic Polymer (PolyDADMAC). The statistical descriptions of the influencing factors, including mean, median, maximum, minimum, and standard deviation, are shown in Table S1.

Table 2: Pearson correlation analysis between filter performance indicator (UFRV) and potential explanatory variables

Method: Pearson	Turbidity	DOC	TOC	True Colour	Flow	Temperature	pH	total Mn	Alkalinity	$KMnO_4$	Chlorine	Ferric Chloride	Cationic Polymer	UFRV
Turbidity	**1.00**													
DOC	0.60	**1.00**												
TOC	0.58	0.96	**1.00**											
True Colour	0.67	0.65	0.68	**1.00**										
Flow	-0.37	-0.39	-0.39	-0.40	**1.00**									
Temperature	-0.24	0.22	0.27	-0.14	0.01	**1.00**								
pH	-0.06	-0.13	-0.15	-0.17	-0.04	0.10	**1.00**							
total Mn	0.04	-0.19	-0.01	0.24	0.07	0.25	0.06	**1.00**						
Alkalinity	-0.29	-0.45	-0.23	-0.25	-0.12	0.09	0.54	0.31	**1.00**					
$KMnO_4$	0.01	0.43	0.57	0.43	-0.12	-0.35	0.17	0.34	0.43	**1.00**				
Chlorine	0.54	0.65	0.54	0.66	-0.60	-0.01	0.27	0.31	0.34	0.29	**1.00**			
Ferric Chloride	0.57	0.56	0.42	0.61	-0.62	0.06	0.18	0.20	0.05	0.01	0.86	**1.00**		
Cationic Polymer	0.54	0.53	0.31	0.66	-0.64	0.11	-0.09	-0.07	-0.28	0.06	0.51	0.86	**1.00**	
UFRV	**-0.52**	-0.32	-0.30	**-0.57**	**0.61**	0.02	-0.03	0.02	0.05	0.21	**-0.43**	**-0.49**	**-0.48**	**1.00**

Five models with a different number of predictor parameters, as presented in Table 3, were considered to analyse their contribution to the accuracy of the predicted filtration performance indicator. For example, the difference between Model-1 and Model-2 is that $FeCl_3$ has been included in Model-2 to determine whether considering $FeCl_3$ will improve the accuracy of the predicted model.

Table 3: Set of influent water quality and operational input variables used in each model for UFRV estimation

Model	Model Predictor Variables						
Model-1	Flow	True Colour	Turbidity				
Model-2	Flow	True Colour	Turbidity	$FeCl_3$			
Model-3	Flow	True Colour	Turbidity	$FeCl_3$	PolyDADMAC		
Model-4	Flow	True Colour	Turbidity	$FeCl_3$	PolyDADMAC	Chlorine	
Model-5	Flow	True Colour	Turbidity	$FeCl_3$	PolyDADMAC	Chlorine	$KMnO_4$

3.2 Hyper-parameter tuning with Grid Search and Random Search

The optimal hyper-parameters for each supervised machine learning model with the lowest RMSE were computed by GS and RS techniques. RS randomly generated a set of candidate parameters from the same tuning range for GS as in Table 4. Table 4 also presents the optimum parameter values only for Model-4 with a one-day time lag. The optimal values of the hyper-parameters with respective kernel functions computed by RS and GS techniques for all five models and different time lags up to 3 days are shown in Table S2. The optimum hyper-parameters for each machine-learning algorithm (Table S2) were applied in selected kernel functions to compare how well each machine-learning algorithm can estimate the UFRV of filters. Whereas the Cp was used to adjust the DT model performance for predicting the UFRV. 10-fold cross-validation was considered in developing the DT model to avoid overfitting [54]. For the grid search technique, a grid network was established in the range of 0 to 1 with Cp to find the optimal value of Cp in the present study. Eventually, the Cp value of 0 was defined as the optimal value for the DT algorithm for 5 selected models and different time-lag values (see Figure S1 in the Supplementary Information document).

Table 4: Adjustable parameters for each supervised machine learning model and tuned optimum parameter values

Supervised model	Kernel Function	Tuned parameters	Search range	Optimal parameter setting (Model-4 with one-day time lag)	
				Grid Search	Random Search
SVR	Polynomial	Degree	[2, 9]	4	3
		Cost	$[2^{-2}, 2^{15}]$	1	1
		Gamma	$[2^{-15}, 2^{3}]$	0.01	0.1

	RBF	Cost	[2^{-2}, 2^{15}]	3	82
		Gamma	[2^{-15}, 2^{3}]	0.8	0.15
	Sigmoid	Scale	[0, 10]	0	1
		Offset	[0, 10]	1	0.1
GPR	RBF	Sigma	[10^{-3}, 10^{3}]	0.65	0.7
	Laplacian	Sigma	[10^{-3}, 10^{3}]	0.5	0.5
	Hyperbolic Tangent	Scale	[0, 10]	1	3
		Offset	[0, 10]	5	1
	Bessel	Sigma	[10^{-3}, 10^{3}]	1	0.2
		Order	[1, 10]	1	1
		Degree	[1, 5]	5	5
DT		Cp	[0, 1]	0	0.02
RF		mtry	[1, number of model predictors]	2	2

3.3 Machine learning model performance assessment

Once the optimal values of the hyper-parameters using GS and RS techniques were determined, the performance of the optimisation process for selected machine learning algorithms was evaluated. It is important to consider the optimal hyper-parameter measures to determine the best predictive machine learning algorithm for filtration performance prediction. Figure 2 compares the effects of increasing the time-lag from 0 to 3 days between UFRV and model input variables (Model-1 to -5 in Table 3) on the performance of developed machine learning algorithms, which include MLR, DT, RF, and SVR with different kernel functions, as well as GPR with different kernel functions that are based on R-squared values. Figure 2 also shows whether incorporating more parameters as model input would improve UFRV predictions.

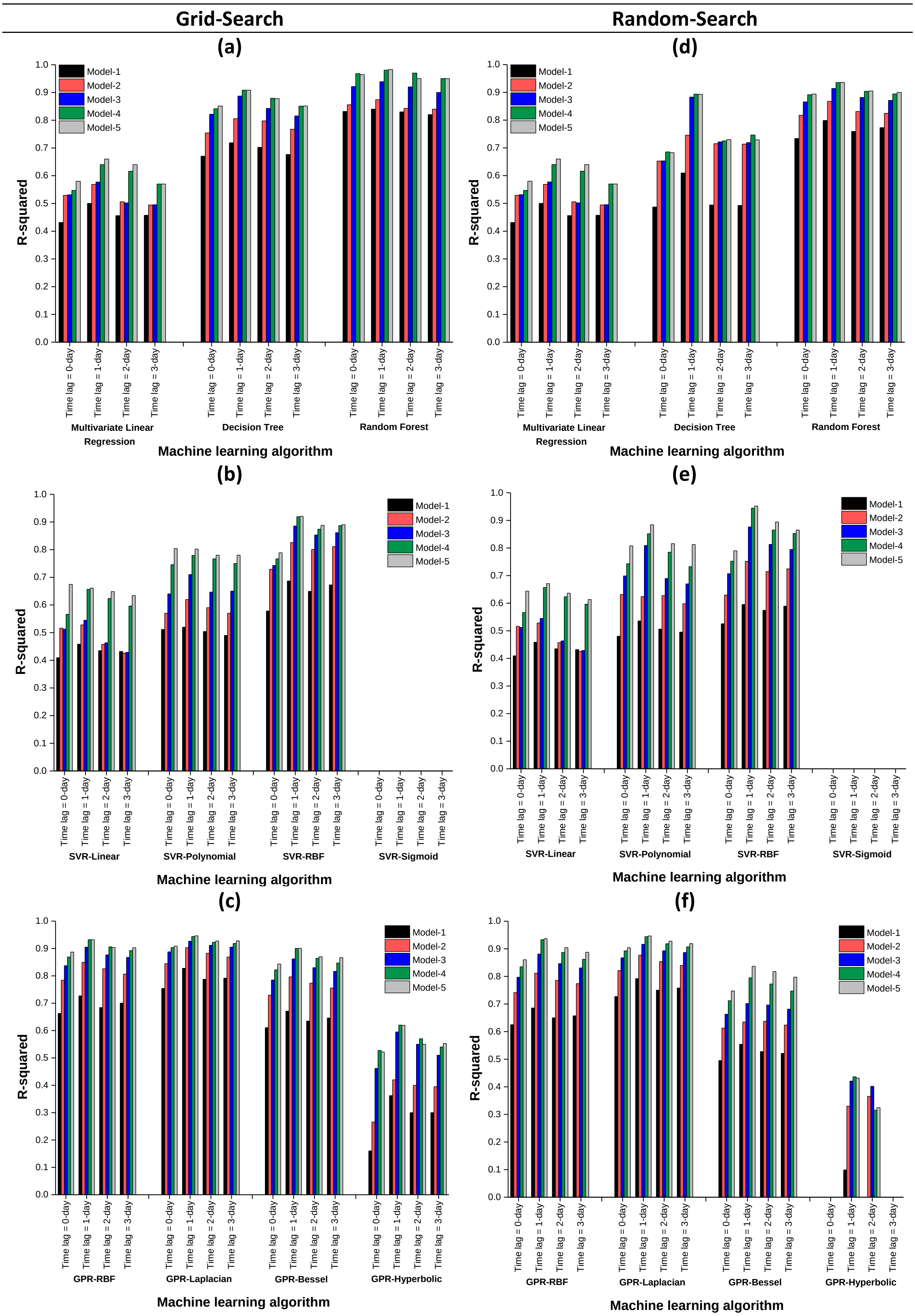


Figure 2: R-squared values of estimated UFRV from different machine learning algorithms by increasing the time-lag between UFRV and model input variables (Model-1 to -5) from 0 to 3 days

Figure 2 shows that considering a 1-day time lag between input variables and UFRV as model output resulted in better predictions than time-lags of 0, 2, and 3 days for all the machine learning algorithm. For example, the results show that by applying the MLR algorithm to Model-4 (input variables: true colour, turbidity, flow, $FeCl_3$, PolyDADMAC, and Chlorine), the R-squared values for having the time-lag of 0, 1, 2, and 3 days to be 0.54, 0.64, 0.61, and 0.57 respectively. This implies that the regression model for UFRV prediction has a higher predictive power when using a time-lag of 1-day in this dataset. Also, Model-4 and Model-5 showed better performance in terms of R-squared than Model-1, Model-2, and Model-3. However, incorporating $KMnO_4$ as a model input variable (i.e., the difference between Model-4 and Model-5, see Table 3) did not enhance the model performance in terms of R-squared with 1-day time lag using SVR-RBF, GPR-RBF, GPR-Bessel, and DT algorithms. In other machine learning algorithms, it only improved the model predictions marginally.

It is also essential to identify the root mean square error (RMSE) of the machine learning models to identify the best model in UFRV prediction, as shown in Figure 3. The results of Figure 2 and Figure 3 revealed that RF with grid search for Model-4 (model input variables are true colour, turbidity, flow, $FeCl_3$, PolyDADMAC, and Chlorine) and 1-day time lag provided high reliability in predicting UFRV (R^2=0.98). The DT algorithm with Model-4 and 1-day time lag yielded a weaker performance compared to those of the RF model (R^2=0.90, Figure 2 (a)). Among the SVR algorithms with different kernel functions (i.e., linear, polynomial, RBF, and sigmoid), the SVR algorithm with the sigmoid kernel function (SVR-Sigmoid model) provided the lowest performance (R^2=0.00). Its performance with both grid search and random search was even lower than those of the MLR model (Figure 2 (b) and Figure 2 (e)). However, the SVR with RBF kernel functions for Model-4 with a 1-day time lag performed better than the other SVR algorithms with an RMSE of 33.68 (Figure 3 (b)) and R^2 of 0.92 (Figure 2 (b)). The GPR algorithm with RBF, Laplacian, and Bessel kernel functions provided a good performance based on the RMSE, and R^2 (Figure 2 (c) and Figure 2 (f)). The GPR algorithm with hyperbolic kernel function provided the poorest performance in this study. Thus, the developed RF algorithm with grid search for Model-4 with a 1-day time lag performed better than the SVR-based model, the GPR model, and the DT model, with an RMSE of 31.58 (Figure 3 (a)) and R^2 of 0.98 (Figure 2 (a)).

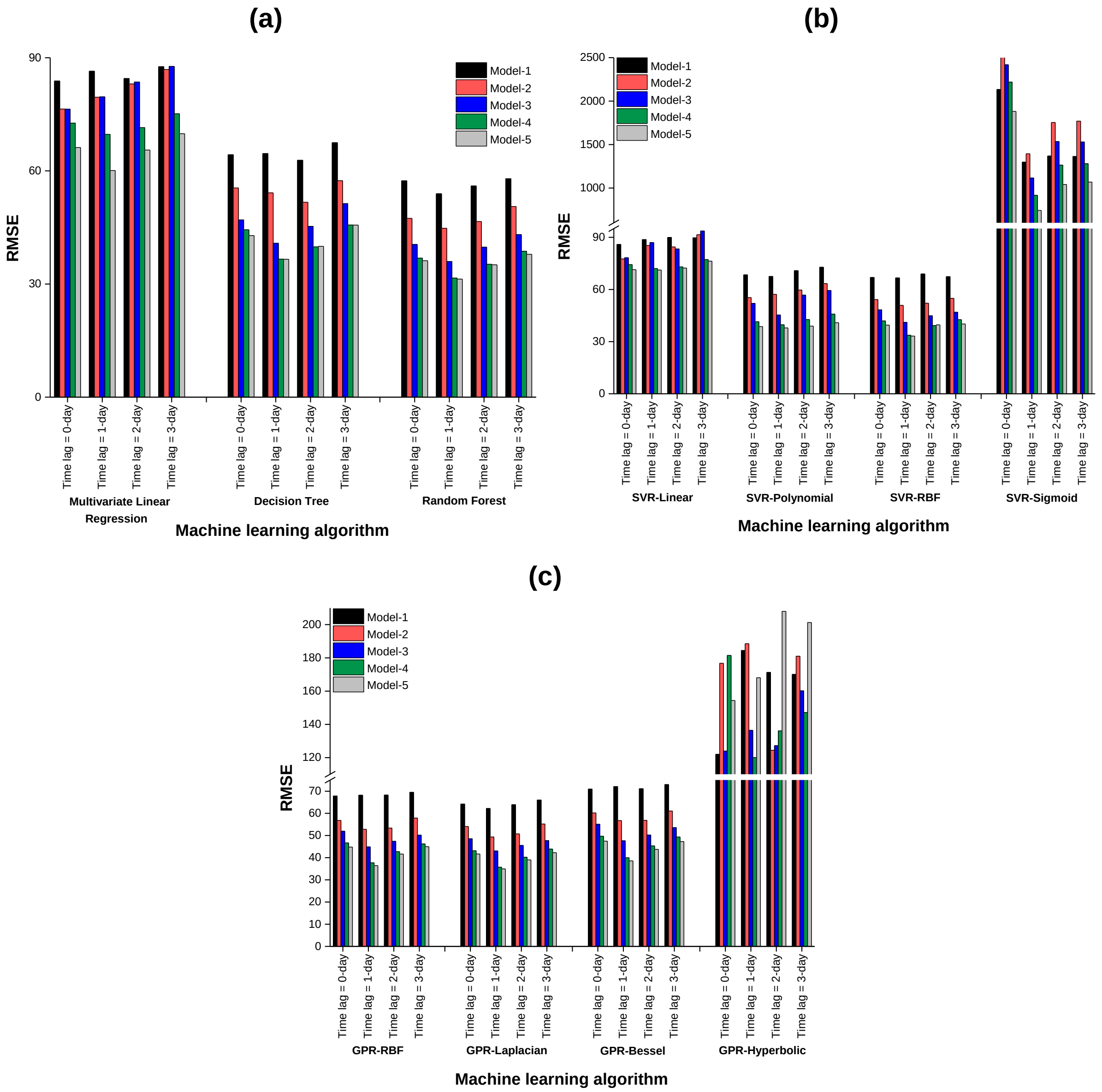


Figure 3: RMSE of estimated UFRV from different Machine learning algorithms by increasing the time-lag between UFRV and model input variables (Models-1 to -5) from 0 to 3 days

Figure 4 compared the results of hyperparameter tuning by the grid search and random search optimisation methods with a 1-day time lag between model output and input variables. Figure 4 shows that the outcomes from the grid search and random search hyperparameter optimisation methods were broadly similar in their performance measures that partially reflect previously reported findings [29].

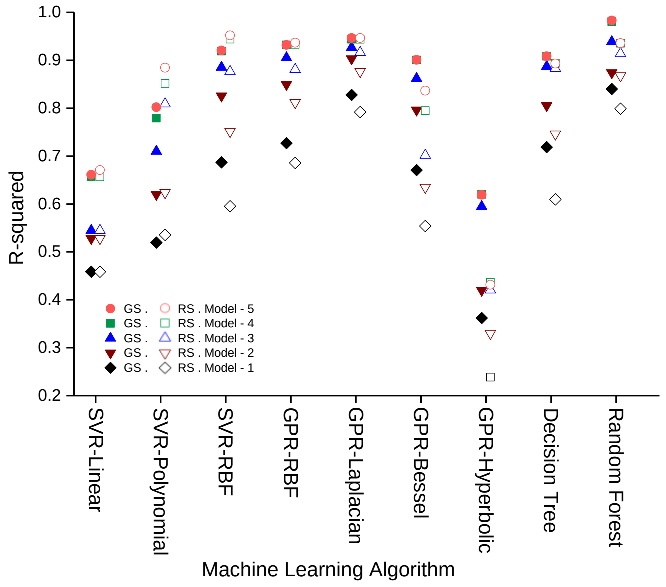


Figure 4: Comparison between the R-squared values of estimated UFRV from the grid search and random search hyperparameter optimization methods by the different machine learning algorithms with a 1-day time lag between UFRV and input variables

Interestingly, the difference in R-squared between the two hyper-parameter optimisation methods (random search and grid search) is marginal for kernel functions that only have one or two adjustable parameters (e.g., GPR-RBF). In contrast, there is a large difference in the R-squared of the final UFRV model between grid search and random search for the kernels in SVR and GPR algorithms with more than two adjustable parameters (e.g., SVR with a polynomial kernel function). This can be explained by the additional parameters that are optimised by the polynomial kernel function compared to only two parameters (γ and C) in the SVR with RBF kernel function.

3.4 Grid Search vs. Random Search

Once the machine learning models were analysed, a comparison to evaluate which model is more suitable for this dataset was carried out (i.e., the speed at each model converges relatively to the other selected algorithms). By normalising RMSE against the machine learning algorithm with the highest RMSE (SVR-Sigmoid in Figure 3) and relatively comparing the training time, Figure 5 depicts a summary of the performance obtained with different machine learning algorithms including MLR, DT, RF, SVR, and GPR with different kernel functions for Model-4 with a 1-day time lag between UFRV and model input variables. In this case, the performance was determined in terms of R-squared (maximum), the training time (minimum), and normalised RMSE (minimum) by each model.

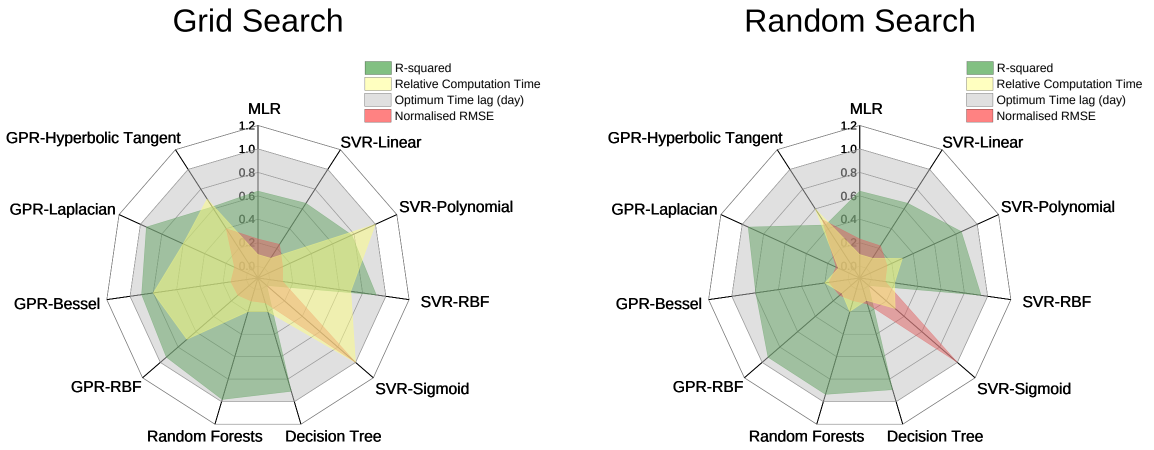


Figure 5: Performance comparison between machine learning algorithms predicting UFRV with a 1-day time lag between UFRV and model input variables including flow, true colour, turbidity, $FeCl_3$, PolyDADMAC, and Chlorine

Figure 5 shows that the SVR-sigmoid provided the worst performance in this dataset with the highest normalised RMSE, lowest R-squared, and the highest relative training time. However, the better models are RF, SVR-RBF, and GPR with RBF and Laplacian as they reported greater prediction accuracy in terms of the lowest RMSE and the highest R-squared. In addition, the random search hyper-parameter optimisation technique took less training time than the grid search considerably. For the RF algorithm, the relative training times using the grid search and random search optimisation techniques were similar, but the grid search method provided a higher R-squared value and a lower RMSE. Hence, the machine learning algorithms with top performance in terms of training time and prediction accuracy were RF with grid search, SVR-RBF with random search, GPR-Laplacian with random search, and GPR-RBF with random search, respectively.

3.5 ROC Analysis

The ROC-AUC curve analysis was carried out to find out whether developed machine learning models that have better performance than the others (RF, SVR-RBF, GPR-Laplacian, and GPR-RBF) could also predict extreme water quality events. The UFRV threshold for such events was determined by studying the changing relationship between discharge and UFRV of filters (hysteresis) during an individual storm event in February 2020. The hysteresis between discharge and UFRV is presented in Figure 6, and the UFRV threshold for such events was set to be 150 m^3/m^2 as the UFRV during an extreme rainfall event (high flowrates) was less than 150 m^3/m^2. Figure S2 in the Supplementary Information document shows the ROC curve for the extreme weather events (UFRV <150 m^3/m^2) using RF, SVR-RBF, GPR-Laplacian, and GPR-RBF algorithms and a 1-day time lag between UFRV and model input variables.

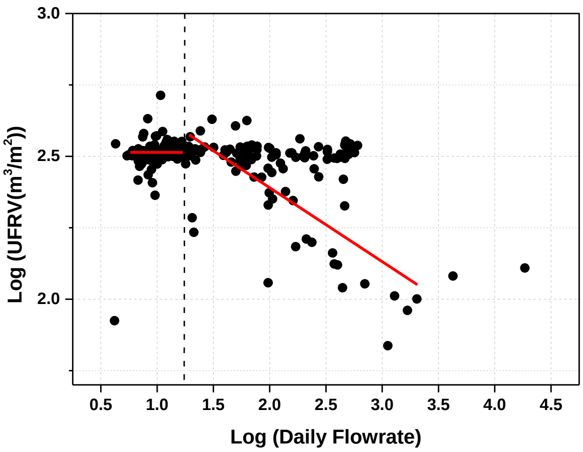


Figure 6: The hysteresis plot associated with the UFRV of filters

Model-5 with GPR-Laplacian algorithms yielded maximum AUC values of 0.86, while the RF, SVR-RBF, and GPR-RBF algorithms yielded AUC values of 0.83, 0.85, and 0.85, respectively (see Table 5). The results of AUC values indicate that machine learning algorithms with top performance in this study (i.e., RF with grid search, SVR-RBF with random search, GPR-Laplacian with random search, and GPR-RBF with random search), not only can predict UFRV with high prediction accuracy and lowest relative training time, but they can also provide high reliability in forecasting events with low UFRV (AUC was over 0.8, above the random level, as shown in Table 5).

Table 5: The ROC-AUC curve analysis results for the extreme weather events (UFRV < 150 $\mathrm{m}^3/\mathrm{m}^2$) using RF, SVR-RBF, GPR-Laplacian, and GPR-RBF algorithms

AUC-ROC	Machine Learning Algorithm			
	Random Forest	SVR-RBF	GPR-Laplacian	GPR-RBF
Model-1	0.78	0.75	0.76	0.73
Model-2	0.78	0.77	0.78	0.73
Model-3	0.79	0.80	0.80	0.74
Model-4	0.83	0.83	0.84	0.81
Model-5	0.83	0.85	0.86	0.85

4 Conclusion

In this study, eleven machine learning regression algorithms were applied to estimate the filter performance from water quality and operational data. The required input parameters were determined using an exhaustive feature selection technique, and two separate hyperparameter optimisation methods (grid search and random search) to optimise the parameter set. The key findings arising from the study are:

- A 1-day time lag between input variables and unit filter run volume as model output resulted in better predictions than time lags of 0, 2, and 3 days.
- The developed random forests algorithm with grid search using true colour, turbidity, flow, $FeCl_3$, PolyDADMAC, and Chlorine as model input variables, and considering a 1-day time lag performed better than the SVR-based model, the GPR model, and the DT model, with an RMSE of 31.58, and R^2 of 0.98.
- In terms of extreme wet weather events UFRV prediction, the UFRV threshold for such events was set to be 150 m^3/m^2 from the hysteresis between discharge and UFRV. The machine learning algorithms with top performance in terms of the training time, prediction accuracy, and forecasting events with low UFRV (AUC over 0.8) were RF with grid search, SVR-RBF with random search, GPR-Laplacian with random search, and GPR-RBF with random search, respectively.

In conclusion, the estimated UFRV of DMG filters in a direct filtration plant were in agreement with the actual measured values observed using machine learning-based algorithms with optimised hyper-parameters. Overall, this study showcases the potential of the machine-learning approach to utilise influent water quality and operating data to predict the filter performance in a water filtration plant.

Acknowledgements

This research was supported via the Australian Research Council's Linkage Projects funding scheme (project number LP160100620) which included support from Sydney Water and WaterNSW. In addition, the authors thank Water Research Australia for the PhD top-up scholarship provided to Anthony Agostino.

5 References

[1] S. J. Khan *et al.*, "Lessons and guidance for the management of safe drinking water during extreme weather events," *Environmental Science: Water Research & Technology,* 10.1039/C6EW00165C vol. 3, no. 2, pp. 262-277, 2017, doi: 10.1039/C6EW00165C.

[2] J. P. Ritson, N. J. D. Graham, M. R. Templeton, J. M. Clark, R. Gough, and C. Freeman, "The impact of climate change on the treatability of dissolved organic matter (DOM) in upland water supplies: A UK perspective," *Science of The Total Environment,* vol. 473-474, pp. 714-730, 2014/03/01/ 2014, doi: https://doi.org/10.1016/j.scitotenv.2013.12.095.

[3] P. Loganathan, M. Gradzielski, H. Bustamante, and S. Vigneswaran, "Progress, challenges, and opportunities in enhancing NOM flocculation using chemically modified chitosan: a review towards future development," *Environmental Science: Water Research & Technology,* 10.1039/C9EW00596J vol. 6, no. 1, pp. 45-61, 2020, doi: 10.1039/C9EW00596J.

[4] S. J. Khan, D. Deere, F. D. L. Leusch, A. Humpage, M. Jenkins, and D. Cunliffe, "Extreme weather events: Should drinking water quality management systems adapt to changing risk profiles?,"

Water Research, vol. 85, pp. 124-136, 2015/11/15/ 2015, doi: https://doi.org/10.1016/j.watres.2015.08.018.

[5] A. W. W. Association, *Operational Control of Coagulation and Filtration Processes*. American Water Works Association, 2011.

[6] H. Liao and W. Sun, "Forecasting and Evaluating Water Quality of Chao Lake based on an Improved Decision Tree Method," *Procedia Environmental Sciences,* vol. 2, pp. 970-979, 2010/01/01/ 2010, doi: https://doi.org/10.1016/j.proenv.2010.10.109.

[7] B. Li *et al.*, "Combining multivariate statistical techniques and random forests model to assess and diagnose the trophic status of Poyang Lake in China," *Ecological Indicators,* vol. 83, pp. 74-83, 2017/12/01/ 2017, doi: https://doi.org/10.1016/j.ecolind.2017.07.033.

[8] K. P. Singh, N. Basant, and S. Gupta, "Support vector machines in water quality management," *Analytica Chimica Acta,* vol. 703, no. 2, pp. 152-162, 2011/10/10/ 2011, doi: https://doi.org/10.1016/j.aca.2011.07.027.

[9] R. Grbić, D. Kurtagić, and D. Slišković, "Stream water temperature prediction based on Gaussian process regression," *Expert Systems with Applications,* vol. 40, no. 18, pp. 7407-7414, 2013/12/15/ 2013, doi: https://doi.org/10.1016/j.eswa.2013.06.077.

[10] S. Shakhari and I. Banerjee, "A multi-class classification system for continuous water quality monitoring," *Heliyon,* vol. 5, no. 5, p. e01822, 2019/05/01/ 2019, doi: https://doi.org/10.1016/j.heliyon.2019.e01822.

[11] H. Lu and X. Ma, "Hybrid decision tree-based machine learning models for short-term water quality prediction," *Chemosphere,* vol. 249, p. 126169, 2020/06/01/ 2020, doi: https://doi.org/10.1016/j.chemosphere.2020.126169.

[12] M. Jeihouni, A. Toomanian, and A. Mansourian, "Decision Tree-Based Data Mining and Rule Induction for Identifying High Quality Groundwater Zones to Water Supply Management: a Novel Hybrid Use of Data Mining and GIS," *Water Resources Management,* vol. 34, no. 1, pp. 139-154, 2020/01/01 2020, doi: 10.1007/s11269-019-02447-w.

[13] A. Mosavi, P. Ozturk, and K.-w. Chau, "Flood Prediction Using Machine Learning Models: Literature Review," *Water,* vol. 10, no. 11, 2018, doi: 10.3390/w10111536.

[14] N.-C. Jung, I. Popescu, P. Kelderman, D. P. Solomatine, and R. K. Price, "Application of model trees and other machine learning techniques for algal growth prediction in Yongdam reservoir, Republic of Korea," *Journal of Hydroinformatics,* vol. 12, no. 3, pp. 262-274, 2009, doi: 10.2166/hydro.2009.004.

[15] L. J. M. l. Breiman, "Random forests," vol. 45, no. 1, pp. 5-32, 2001.

[16] X. Hu *et al.*, "Estimating PM2.5 Concentrations in the Conterminous United States Using the Random Forest Approach," *Environmental Science & Technology,* vol. 51, no. 12, pp. 6936-6944, 2017/06/20 2017, doi: 10.1021/acs.est.7b01210.

[17] A. Ziegler and I. König, "Mining data with random forests: Current options for real-world applications," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* vol. 4, 01/01 2014, doi: 10.1002/widm.1114.

[18] S. Athey, J. Tibshirani, and S. Wager, "Generalized random forests," (in en), *Ann. Statist.,* vol. 47, no. 2, pp. 1148-1178, 2019/04 2019, doi: 10.1214/18-AOS1709.

[19] R. Genuer, "Variance reduction in purely random forests," *Journal of Nonparametric Statistics,* vol. 24, no. 3, pp. 543-562, 2012/09/01 2012, doi: 10.1080/10485252.2012.677843.

[20] G. Biau, L. Devroye, and G. Lugosi, "Consistency of Random Forests and Other Averaging Classifiers," *Journal of Machine Learning Research,* vol. 9, pp. 2015-2033, 09/01 2008, doi: 10.1145/1390681.1442799.

[21] H. Tyralis, G. Papacharalampous, and A. Langousis, "A Brief Review of Random Forests for Water Scientists and Practitioners and Their Recent History in Water Resources," *Water,* vol. 11, no. 5, 2019, doi: 10.3390/w11050910.

[22] M. K. Gill, T. Asefa, M. W. Kemblowski, and M. McKee, "SOIL MOISTURE PREDICTION USING SUPPORT VECTOR MACHINES1," *JAWRA Journal of the American Water Resources Association,* vol. 42, no. 4, pp. 1033-1046, 2006/08/01 2006, doi: 10.1111/j.1752-1688.2006.tb04512.x.

[23] Q. Yu, H. Yin, W. Ke, H. Dong, and D. Hou, "Adaptive Detection Method for Organic Contamination Events in Water Distribution Systems Using the UV-Vis Spectrum Based on Semi-Supervised Learning," *Water,* vol. 10, p. 1566, 11/02 2018, doi: 10.3390/w10111566.

[24] K. Zhang, G. Achari, H. Li, A. Zargar, and R. Sadiq, "Machine learning approaches to predict coagulant dosage in water treatment plants," *International Journal of System Assurance Engineering and Management,* vol. 4, no. 2, pp. 205-214, 2013/06/01 2013, doi: 10.1007/s13198-013-0166-5.

[25] R. G. Mantovani, A. L. D. Rossi, J. Vanschoren, B. Bischl, and A. C. P. L. F. d. Carvalho, "Effectiveness of Random Search in SVM hyper-parameter tuning," in *2015 International Joint Conference on Neural Networks (IJCNN),* 12-17 July 2015 2015, pp. 1-8, doi: 10.1109/IJCNN.2015.7280664.

[26] O. Samuelsson, A. Björk, J. Zambrano, and B. Carlsson, "Gaussian process regression for monitoring and fault detection of wastewater treatment processes," *Water Science and Technology,* vol. 75, no. 12, pp. 2952-2963, 2017, doi: 10.2166/wst.2017.162.

[27] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing Multiple Parameters for Support Vector Machines," *Machine Learning,* vol. 46, no. 1, pp. 131-159, 2002/01/01 2002, doi: 10.1023/A:1012450327387.

[28] E. G. Ortiz-García, S. Salcedo-Sanz, Á. M. Pérez-Bellido, and J. A. Portilla-Figueras, "Improving the training time of support vector regression algorithms through novel hyper-parameters search space reductions," *Neurocomputing,* vol. 72, no. 16, pp. 3683-3691, 2009/10/01/ 2009, doi: https://doi.org/10.1016/j.neucom.2009.07.009.

[29] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," vol. 13, no. null %J J. Mach. Learn. Res., pp. 281–305, 2012.

[30] S. Moradi *et al.*, "Quantifying natural organic matter concentration in water from climatological parameters using different machine learning algorithms," *H2Open Journal,* vol. 3, no. 1, pp. 328-342, 2020, doi: 10.2166/h2oj.2020.035.

[31] A. S. M. Mohiuddin, P. Cox, and B. Blayney, "The impact of the Millennium Drought on water filtration plants," *Water e-Journal,* vol. 5, pp. 1-10, 01/01 2020, doi: 10.21139/wej.2020.002.

[32] Australian Government Bureau of Meteorology. "Climate Statistics for Australian Locations." http://www.bom.gov.au/climate/averages/tables/cw_066062.shtml (accessed November, 2022).

[33] Australian Government - National Emergency Management Agency. "Heavy rainfall and floods." https://knowledge.aidr.org.au/resources/heavy-rainfall-and-floods-new-south-wales-february-2020/ (accessed December, 2022).

[34] P. S. Horn, L. Feng, Y. Li, and A. J. Pesce, "Effect of outliers and nonhealthy individuals on reference interval estimation," (in eng), *Clinical chemistry,* vol. 47, no. 12, pp. 2137-45, Dec 2001.

[35] B. B. Mirus, M. D. Morphew, and J. B. Smith, "Developing hydro-meteorological thresholds for shallow landslide initiation and early warning," *Water,* vol. 10, no. 9, pp. 1-19, 2018, doi: 10.3390/w10091274.

[36] M. I. Sameen, B. Pradhan, and S. Lee, "Application of convolutional neural networks featuring Bayesian optimization for landslide susceptibility assessment," *CATENA,* vol. 186, p. 104249, 2020/03/01/ 2020, doi: https://doi.org/10.1016/j.catena.2019.104249.

[37] S. Moradi *et al.*, "Quantifying natural organic matter concentration in water from climatological parameters using different machine learning algorithms," *H2Open Journal,* 2020, doi: 10.2166/h2oj.2020.035.

[38] D. De Clercq, Z. Wen, F. Fei, L. Caicedo, K. Yuan, and R. Shang, "Interpretable machine learning for predicting biomethane production in industrial-scale anaerobic co-digestion," *Science of*

The Total Environment, vol. 712, p. 134574, 2020/04/10/ 2020, doi: https://doi.org/10.1016/j.scitotenv.2019.134574.

[39] M. Castrillo and Á. L. García, "Estimation of high frequency nutrient concentrations from water quality surrogates using machine learning methods," *Water Research,* vol. 172, p. 115490, 2020/04/01/ 2020, doi: https://doi.org/10.1016/j.watres.2020.115490.

[40] J. Qu and M. J. Zuo, "Support vector machine based data processing algorithm for wear degree classification of slurry pump systems," *Measurement,* vol. 43, no. 6, pp. 781-791, 2010/07/01/ 2010, doi: https://doi.org/10.1016/j.measurement.2010.02.014.

[41] M. J. a. p. a. Ebden, "Gaussian processes: A quick introduction," 2015.

[42] G. S. Naganathan and C. K. Babulal, "Optimization of support vector machine parameters for voltage stability margin assessment in the deregulated power system," *Soft Computing,* vol. 23, no. 20, pp. 10495-10507, 2019/10/01 2019, doi: 10.1007/s00500-018-3615-x.

[43] B. Üstün, W. J. Melssen, M. Oudenhuijzen, and L. M. C. Buydens, "Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization," *Analytica Chimica Acta,* vol. 544, no. 1, pp. 292-305, 2005/07/15/ 2005, doi: https://doi.org/10.1016/j.aca.2004.12.024.

[44] H. Fröhlich and A. Zell, "Efficient parameter selection for support vector machines in classification and regression via model-based global optimization," in *Proceedings of the International Joint Conference on Neural Networks*, 2005, vol. 3, pp. 1431-1436, doi: 10.1109/IJCNN.2005.1556085. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-33750124478&doi=10.1109%2fIJCNN.2005.1556085&partnerID=40&md5=78ebab5f98189ad56aa61b4511cb1d1a

[45] H. Chen, L. Xu, W. Ai, B. Lin, Q. Feng, and K. Cai, "Kernel functions embedded in support vector machine learning models for rapid water pollution assessment via near-infrared spectroscopy," *Science of The Total Environment,* vol. 714, p. 136765, 2020/04/20/ 2020, doi: https://doi.org/10.1016/j.scitotenv.2020.136765.

[46] H. Chen *et al.*, "Hyperparameter Estimation in SVM with GPU Acceleration for Prediction of Protein-Protein Interactions," in *2019 IEEE International Conference on Big Data (Big Data)*, 9-12 Dec. 2019 2019, pp. 2197-2204, doi: 10.1109/BigData47090.2019.9006024.

[47] D. Yunana *et al.*, "Developing Bayesian networks in managing the risk of Legionella colonisation of groundwater aeration systems," *Water Research,* vol. 193, p. 116854, 2021/04/01/ 2021, doi: https://doi.org/10.1016/j.watres.2021.116854.

[48] H. Tu and V. Nair, "Is one hyperparameter optimizer enough?," presented at the Proceedings of the 4th ACM SIGSOFT International Workshop on Software Analytics, Lake Buena Vista, FL, USA, 2018. [Online]. Available: https://doi.org/10.1145/3278142.3278145.

[49] R. G. Mantovani, A. L. D. Rossi, J. Vanschoren, B. Bischl, and A. C. P. L. F. Carvalho, "To tune or not to tune: Recommending when to adjust SVM hyper-parameters via meta-learning," in *2015 International Joint Conference on Neural Networks (IJCNN)*, 12-17 July 2015 2015, pp. 1-8, doi: 10.1109/IJCNN.2015.7280644.

[50] A. L. D. Rossi and A. C. P. L. F. d. Carvalho, "Bio-inspired Optimization Techniques for SVM Parameter Tuning," in *2008 10th Brazilian Symposium on Neural Networks*, 26-30 Oct. 2008 2008, pp. 57-62, doi: 10.1109/SBRN.2008.28.

[51] P. Ashrafi, Y. Sun, N. Davey, R. G. Adams, S. C. Wilkinson, and G. P. Moss, "Model fitting for small skin permeability data sets: hyperparameter optimisation in Gaussian Process Regression," vol. 70, no. 3, pp. 361-373, 2018, doi: 10.1111/jphp.12863.

[52] K. Taheri, H. Shahabi, K. Chapi, A. Shirzadi, F. Gutiérrez, and K. Khosravi, "Sinkhole susceptibility mapping: A comparison between Bayes-based machine learning algorithms," vol. 30, no. 7, pp. 730-745, 2019, doi: 10.1002/ldr.3255.

[53] *Matlab*. (2018). Massachusetts, United States.

[54] D. Pérez-Guaita, J. Kuligowski, B. Lendl, B. R. Wood, and G. Quintás, "Assessment of discriminant models in infrared imaging using constrained repeated random sampling – Cross validation," *Analytica Chimica Acta,* vol. 1033, pp. 156-164, 2018/11/29/ 2018, doi: https://doi.org/10.1016/j.aca.2018.05.019.