Int. J. Nonlinear Anal. Appl. In Press, 1–16
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.25730.3112

A machine-learning approach for optimal ionic concentration determination in smart-water EOR applications

Ehsan Bahonar[a,*], Sadegh Salmani[b], Mahshid Rajabi[b]

[a]*Faculty of Petroleum and Natural Gas Engineering, Sahand University of Technology, Tabriz, Iran*

[b]*Ahwaz Faculty of Petroleum Engineering, Petroleum University of Technology, Ahvaz, Iran*

(Communicated by Seyed Hossein Siadati)

Abstract

The smart water-enhanced oil recovery (EOR) process is a pioneering tertiary recovery method in the petroleum industry. Meanwhile, more than half of oil reserves in the world are carbonate. Accordingly, considering the technical and financial aspects, the determination of the accurate concentration of presented ions in smart water is very important. Although several experimental studies considered this issue, no appropriate statistical method has been suggested to deal with this problem during smart water injection in carbonate rocks. In the present article, five different multi-target regression machine learning (ML) algorithms (i.e., Random Forest, Decision Tree, K-Nearest Neighbours, Lasso and Linear), were used to predict the ionic concentration in imbibition tests. A completely reliable dataset of imbibition test results, which were gathered from the literature, was employed in the learning process of the algorithms and examining their accuracy. After data processing, feature extraction, splitting data and building candidate ML models, an exact hyperparameter tuning was carried out to evaluate the ML models and select the best model. It is found that the Random Forest algorithm is the best-acting approach with the lowest total root mean squared error (RMSE) of 1.231 and the highest score of 0.981 for predicting ionic concentration in smart water EOR applications. In conclusion, the proposed model is the most efficient approach as compared with commonly used costly laboratory tests, which can be a good candidate for predicting the concentration of ions in smart water injection processes.

Keywords: Wettability alteration, Enhanced oil recovery, Machine learning, Carbonate rocks, Smart water
2020 MSC: 62C25, 90B50

1 Introduction

The initial production from an oil reservoir which is called *Primary Production*, mainly happens due to the reservoir's internal pressure where hydrocarbons naturally rise to the surface. *Secondary Recovery* operations contain water and gas injection into the reservoir to maintain reservoir pressure at a high level and displace hydrocarbons toward the surface level. Hydrocarbons that have not already been extracted from the reservoir through the primary and secondary stages of the recovery process, undergo the *Tertiary Recovery* step which is also known as *Enhanced Oil Recovery* (EOR). Chemical flooding is one of the most practical EOR methods which helps to free trapped oil in the

*Corresponding author
Email addresses: ebahonar@gmail.com (Ehsan Bahonar), salmani.sgh@gmail.com (Sadegh Salmani), mahshidrajabi96@gmail.com (Mahshid Rajabi)

Received: November 2021 *Accepted:* January 2022

reservoir by altering the fluid properties by changing the surface tension to overcome capillary barriers in the reservoir [1].

Enhanced oil recovery in carbonate rocks which are usually naturally fractured is one of the main recovery procedures in mature fields. Carbonate rocks contain more than 50% of oil reserves in the world [44] and are neutral or preferentially oil-wet which usually decreases the efficiency of the recovery process by water flooding [18, 58, 2]. Among all recovery mechanisms, water-based EOR methods have been the most common way of hydrocarbon recovery by many oil companies. In water-based EOR, not only the quantity of water but also the quality of injection water is very important and should be highlighted as a determining factor for enhancing recovery efficiency [4]. Spontaneous imbibition is one of the main mechanisms in fractured reservoirs in which water displaces oil from matrix to fracture due to capillary forces, which highlights the importance of understanding the wettability characteristics of the rock surfaces [45, 6, 35]. *Smart water* injection is the most common and efficient approach for chemical Enhanced Oil Recovery due to its high efficiency and low cost. Effective mechanisms involved in smart water injection into carbonate rocks and sandstones include fines migration, pH increase, multi-ion exchange (MIE), salting in, and wettability alteration. For better effectiveness of these mechanisms, it is required to predict the optimal water ions in the imbibition and flooding processes [3]. Studies on seawater injection into the North Sea chalk reservoirs showed that modifying and designing the composition of the injection brine can change the wettability condition of the rock surfaces more toward water-wet [33]. This process is termed as *smart water*, *designed water*, or *engineered water*. Smart water refers to synthesized water whose type and content of its ions are selected in a smart way to achieve certain recovery conditions during water injection for hydrocarbon recovery.

Zhang et al. used outcrop chalk from Stevns Klint (a white chalk cliff), near Copenhagen, Denmark, to investigate the effect of Ca^{2+}, in the presence of SO_4^{2-}, on wettability alteration. This outcrop chalk had a low permeability of 2 to 5 $[mD]$ and a high porosity of 45 to 50%. Besides, all cores had a length of 7 $[cm]$ and a diameter of 3.75 $[cm]$ and were drilled from the same outcrop. Spontaneous imbibition tests, Chromatographic wettability tests, interfacial tension, and zeta potential measurements were performed using oil samples with acid numbers of $AN = 2.07$ $[mgKOH/g]$ and $AN = 0.55$ $[mgKOH/g]$ and brine samples (artificial Ekofisk formation brine) with different concentrations of Ca^{2+} and SO_4^{2-}. They showed that when the oil has a higher AN, the change in Ca^{2+} in the reservoir water will not have much effect on the recovery factor (RF) and also showed that the higher the concentration of Ca^{2+} in the presence of SO_4^{2-} in both reservoir water and injected water, the higher the value of RF [69].

Strand et al. investigated the temperature effects on Enhanced Oil Recovery from mixed-wet outcrop chalk using seawater injection. This outcrop was obtained from Stevns Klint (a white chalk cliff), near Copenhagen, Denmark, and had a low permeability of 2 to 5 $[mD]$ and a high porosity of 45 to 50%. In their experiments, crude oil with low AN of 0.7 $[mgKOH/g]$ and high AN of 1.9 $[mgKOH/g]$ was used. They concluded that, firstly, at temperatures below 100 °C, the amount of oil recovery through the imbibition process was not significant compared to the recovery by flooding. Secondly, at temperatures above 100 °C, the amount of oil recovery increased through the imbibition process, but still less than the flooding process [59].

Zhan et al. studied the secondary and tertiary recovery processes by changing the salinity of injection and connate waters. Core plugs were cut from Berea sandstone and had a diameter of 3.8 $[cm]$ and length of 7.6 $[cm]$ with permeability to N_2 ranging from 600 $[mD]$ to 1100 $[mD]$ and porosity ranging from 16.5 to 33%. Three different oil samples with acid numbers of 0.33, 0.17, and 0.16 $[mgKOH/g]$ were used in their experiments. They concluded that if low salinity water is used as the reservoir brine and high salinity water as displacing water, the amount of oil recovery would be high. They also concluded that if both the reservoir water and the displacing water have low salinity, the amount of oil recovery would be high as well [71].

Fathi et al. examined the effect of ionic composition and salinity of smart water on oil recovery at different temperatures of 100, 110 and 120 °C under flooding and imbibition processes where crude oil with an acid number of $AN = 1.9$ $[mgKOH/g]$ was used. Outcrop chalks from Stevns Klint of Denmark, with a low permeability of 1 to 2 $[mD]$ and a high porosity of about 45% were used. Their work confirmed that not only the impact of active ions such as Ca^{2+}, Mg^{2+}, and SO_4^{2-} are important for wettability alteration but also non-active ions such as Na^+ and Cl^- have important effects on wettability alteration which can be considered as a double-layer mechanism. The high ultimate recovery factor and the high speed of the imbibition process were the results of the reduction in NaCl content in water. Consequently, ion-engineered water flooding can be a relatively smart EOR technique for tuning the ionic composition of the injecting brine [24]. Used two types of oil with low AN of 0.5 $[mgKOH/g]$ and high AN of 2.0 $[mgKOH/g]$, as well as a temperature range of 70 to 120 °C, to investigate and find an optimal concentration of water ions in the process of imbibition on the chalky cores from Stevns Klint, Denmark [25]. They found that when the concentration of SO_4^{2-} in the seawater is quadrupled, the recovery factor will be higher than when the water is depleted in NaCl. Besides, the impact of changing concentrations of Ca^{2+} at a temperature of 100 °C was not noticeable, but

at a temperature of 120 °C, a significant change was observed.

As a data-driven study on enhanced oil recovery, Dang et al. presented a novel EOR process called *Hybrid Low Salinity Chemical Flooding*, where the low salinity water flooding and the chemical flooding are combined [22]. They used artificial intelligence for its mechanistic modelling and showed that one of the most important problems during the mechanistic modelling of that hybrid EOR process was the change of relative permeability due to different factors such as the presence of salinity, polymers, and surfactants. To overcome this challenge, they implemented a multilayer non-linear network for multi-dimensional interpolation of the relative permeability during the hybrid EOR process. Romanuka et al. performed a screening study using different carbonate rocks. Spontaneous imbibition tests were carried out on various rock samples such as limestone, dolomite, and chalk with different values of porosity and permeability [52]. Puntervold et al. conducted a laboratory test to determine the optimum values of NaCl and SO_4^{2-} in smart water injection to obtain maximum oil recovery factor (RF) at 90 °C [51]. In their experiment, with complete depletion of NaCl from smart water, the amount of recovery factor increased by 8%, although was not a linear function of the NaCl concentration. To achieve maximum recovery factor, more than 90% of NaCl got removed from the smart water and the concentration of SO_4^{2-} was approximately fourfold. As a result, the recovery factor was increased by 10% of the original oil in place.

Shariatpanahi et al. compared the process of imbibition in dolomite and calcite rocks with experiments and previous papers on sandstone and chalk [54]. They concluded that Mg^{2+} had a greater effect on the water wetness of calcite compared to Ca^{2+}. They also showed that, since the dolomite surface had both ions of Mg^{2+} and Ca^{2+} as positively charged components, it could be stated that the adsorption of active polar carboxylic onto dolomite rock was less than for calcite. Moreover, when the water was 10 times diluted, it reached incremental oil of 10 to 15% in dolomite rock. Although laboratory imbibition tests are promising methods for EOR applications, they are expensive and time-consuming. besides, the Design of Experiment (DOE) is another tool to predict optimal ionic concentration. In this work, an accurate machine learning algorithm is used for the determination of ionic concentration in smart water EOR processes. The proposed model showed good agreement with different imbibition tests.

2 Dataset Overview

The process of data collection for use as training and test datasets will be discussed in this section. In this work, the results of imbibition tests which are carried out by other researchers are employed to generate a 29-dimentional feature space dataset. The database on which the model was trained includes 127 sets of data obtained from different experimental studies from the literature [24, 25, 36, 51, 52, 59, 69, 71] and these studies are discussed briefly in the introduction section. Each data point includes lithology, porosity (ϕ), permeability (k), initial water saturation (S_{wi}), imbibing time (t), recovery factor (RF), imbibition test temperature (T_0), temperature in which density and viscosity are measured (T_1), acid number (AN), crude oil density (ρ_o), crude oil viscosity (μ_o), formation water compositions, imbibing fluid compositions, ionic strength (I), and total dissolved solids (TDS). Records of some data points existing in the data bank used in this study are summarized in Table 1. Statistical parameters of the employed databank in this paper are presented in Table 2.

Table 1: Summarized dataset.

Reference	Lithology	ϕ	k	S_{wi}	t_{imb}	RF	T_0	AN	ρ_o	T_1	μ_o
[69]	chalk	47.3 - 49.9	2 - 5	21.4 - 26.3	30	9.2 - 54.6	70	0.55	0.803	25	2.56
							100	2.07	0.806		3.05
					10	20.6 - 65.8	130				
[69]	chalk	47.2 - 49.8	2 - 5	0 - 30	65	8.9 - 30.9	70	2.07	0.806	25	3.05
					19	22.5 - 29.4	100				
					5	58.1 - 63.2	130				
					15 - 30	17.5 - 67.3	70	0.55	0.8.3		2.56
[59]	chalk	43.7 - 48.8	1 - 3	10	90	8 - 30.7	90	0.7	0.8	20	25
					12	10.2 & 10..8	110	1.9			
					19	10.8 & 11.9	120				
[71]	sandstone	22.7 - 23.3	1067 - 1174	22.4 - 25.8	4	42.5 - 73.8	75	0.17	0.903	75	7.5
[24]	chalk	45 - 47	1 - 3	8 & 9	49	36.1 - 44.3	100	1.9	0.8115	20	3.38
					71.5	23.4 - 60.3	110				
					34	14.6 - 69.4	120				
[24]	chalk	44 - 48	1 - 3	9 - 11	37	16.5 - 62.2	90	0.5	0.798	25	2.6
					27	39.6 - 48.5	70				
					61	10.6 - 30.3	100	2	0.815		3.38
					59	34.1 - 47.8	120				

[52]	chalk	49 - 51	5.5 - 6.2	15	111	41 - 45	60	0.92	0.843	70	3.93
	limestone	23 - 31	1.9 - 56.5	10 - 20	25.4 & 33	3 - 9.5	70	0.42	0.813		2.36
					12	10.28	120		0.779	120	0.97
	dolomite	12 - 25	10.8 - 235	11 - 17	42.3	0.12 - 2.29	70	0.92	0.843	70	3.93
					64	2.53 - 10.43	85	0.07	0.831	85	4.84
					4.1 & 6	5.6 & 8.7	70	0.92		70	3.93
[51]	chalk	45 - 55	1 - 5	10	25 - 55	22 - 61.2	90	0.5	0.801	20	2.3
[54]	chalk	45.35 - 49	1 - 3	10	5.1 - 7.9	8.1 - 65	25	0.34	0.804	20	2.5
								0.17	0.8		
	dolomite	20	201 & 235	15	4.1 & 6	5.6 - 26.3	70	0.52	0.847	20	20.8
[36]	chalk	43 & 46	3 - 5	10	7.6 & 11.9	5.6 - 26.3	50	0.34	0.808	25	3.2

Table 2: Statistical description of the dataset.

Category	Parameter	Unit	Mean	Std	Min	Q1	Median	Q3	Max
Experiment	ϕ	%	42.32	10.43	12.00	44.25	46.98	48.50	55.00
	k	mD	47.95	200.56	1.50	1.70	3.50	3.50	1174.00
	S_{wi}	%	14.59	7.03	0.00	10.00	11.00	22.65	29.20
	t_{imb}	Day	34.15	23.73	4.00	12.00	30.00	50.56	111.00
	RF	%	31.53	20.49	0.12	10.80	30.90	47.25	73.80
	T_0	°C	83.86	28.13	25.00	70.00	85.00	100.00	130.00
	AN	mg of KOH/g	1.04	0.74	0.07	0.50	0.55	1.90	2.07
	ρ_o	g/cm^3	0.81	0.02	0.78	0.80	0.81	0.81	0.90
	T_1	°C	35.75	25.41	20.00	20.00	25.00	25.00	120.00
	μ_o	cp	3.36	2.45	0.97	2.50	3.05	3.38	20.80
Formation water	Na^+	mol/lit	1.22	0.76	0.00	0.99	1.00	1.34	2.85
	K^+	mol/lit	0.00	0.00	0.00	0.00	0.00	0.01	0.01
	Mg^{2+}	mol/lit	0.04	0.09	0.00	0.01	0.03	0.03	0.66
	Ca^{2+}	mol/lit	0.13	0.16	0.00	0.01	0.03	0.23	0.57
	Cl^-	mol/lit	1.56	1.04	0.00	1.07	1.20	1.41	3.87
	HCO_3^-	mol/lit	0.00	0.00	0.00	0.00	0.00	0.01	0.01
	SO_4^{2-}	mol/lit	0.00	0.02	0.00	0.00	0.00	0.00	0.13
	Ionic strength	mol/lit	1.66	1.20	0.00	1.11	1.45	1.45	4.39
	TDS	g/lit	86.62	58.10	0.00	62.80	62.83	82.05	222.19
Injection water	Na^+	mol/lit	0.91	0.99	0.00	0.19	0.45	1.01	3.51
	K^+	mol/lit	0.01	0.00	0.00	0.00	0.01	0.01	0.01
	Mg^{2+}	mol/lit	0.05	0.06	0.00	0.04	0.05	0.05	0.66
	Ca^{2+}	mol/lit	0.09	0.15	0.00	0.01	0.01	0.05	0.57
	Cl^-	mol/lit	1.17	1.30	0.00	0.32	0.54	1.08	3.98
	HCO_3^-	mol/lit	0.00	0.00	0.00	0.00	0.00	0.00	0.01
	SO_4^{2-}	mol/lit	0.02	0.03	0.00	0.00	0.01	0.02	0.13
	Ionic strength	mol/lit	1.30	1.39	0.00	0.47	0.66	1.12	4.39
	TDS	g/lit	66.31	71.21	0.00	26.30	33.40	62.83	230.77

A boxplot is used to identify the outlier data and to show the distribution of data. Outlier data are those data points higher than the upper extreme or lower than the lower extreme in the plot as shown in Figure 1. As mentioned in the literature, earlier studies demonstrate a strong and consistent association between increasing Smart water injection efficiency and diverse water ion concentration. Therefore, a histogram of them was drawn in Figure 2 to acquire the best vision of this concentration on formation water and injected water ions as well.

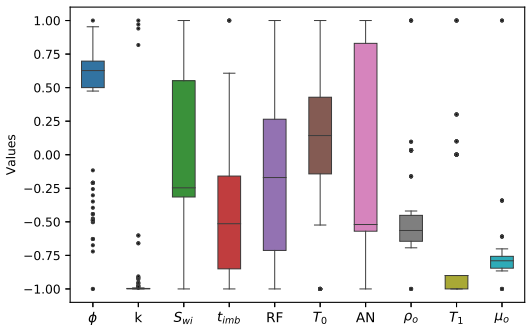


Figure 1: boxplot of the expriment data

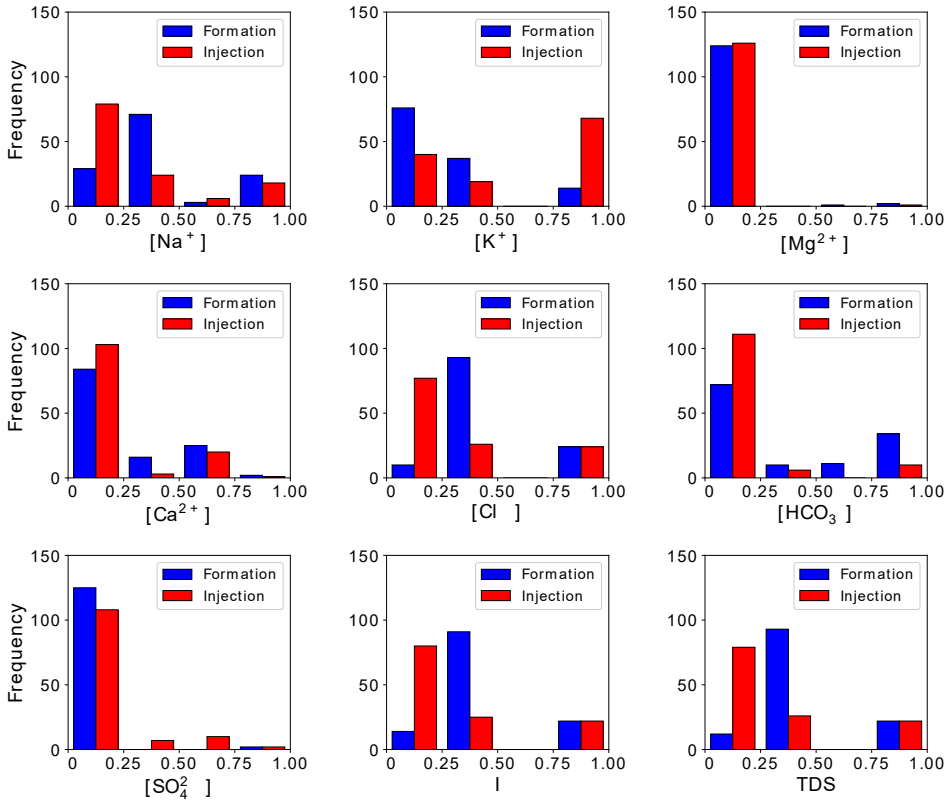


Figure 2: histogram of formation water and injected water ions

Furthermore, to investigate the relations between the outlier data and to show the distribution of data, a Confusion matrix was plotted. This plot draws Confusion matrix plots between all parameters and calculates the Pearson correlation coefficients. This coefficient represents the linear relations between the two parameters, see Figure 3. It should be noted that outlier data points are higher than the upper extreme or under the lower extreme in the plot. Mainly, this Figure was utilized to check potential correlations between independent and dependent data and to display whether any variables are similar to each other. Finally, Input data were visualized and investigated, and the outlier data were identified and deleted, see Figure 4. Five machine learning algorithms were exploited to estimate terminal ionic concentration. The development procedures of the models are discussed below.

3 Machine Learning Framework

As mentioned earlier, the purpose of this work is to build predictive Multi-Target Regression (MTR) models. Multi-Target Regression (MTR), also known as multi-output/multivariate regression is mainly the task of predicting multiple continuous output variables (called targets) using some input variables (called features). MTR has applications in many fields such as geoscience, economics, energy, healthcare, etc. ([27, 73, 12, 43, 72, 64, 11]). The main challenge in building MTR models is the appropriate modelling of target dependencies between target parameters. A naive approach towards Multi-Target Regression (MTR) is using a combination of single-output regression models instead of multi-output regression models. There are several problems associated with using a combination of single-output regression models for a multi-output regression task. Mainly, concatenating multiple single-output models takes longer to train and is computationally expensive. Besides, they optimize for the single target rather than all the target variables together and do not use the relationships between the target variables. In this work, we build Multi-Target Regression (MTR) models using different algorithms proposed in the literature and use them for the prediction of ionic contents in smart water flooding processes, see Figure 5.

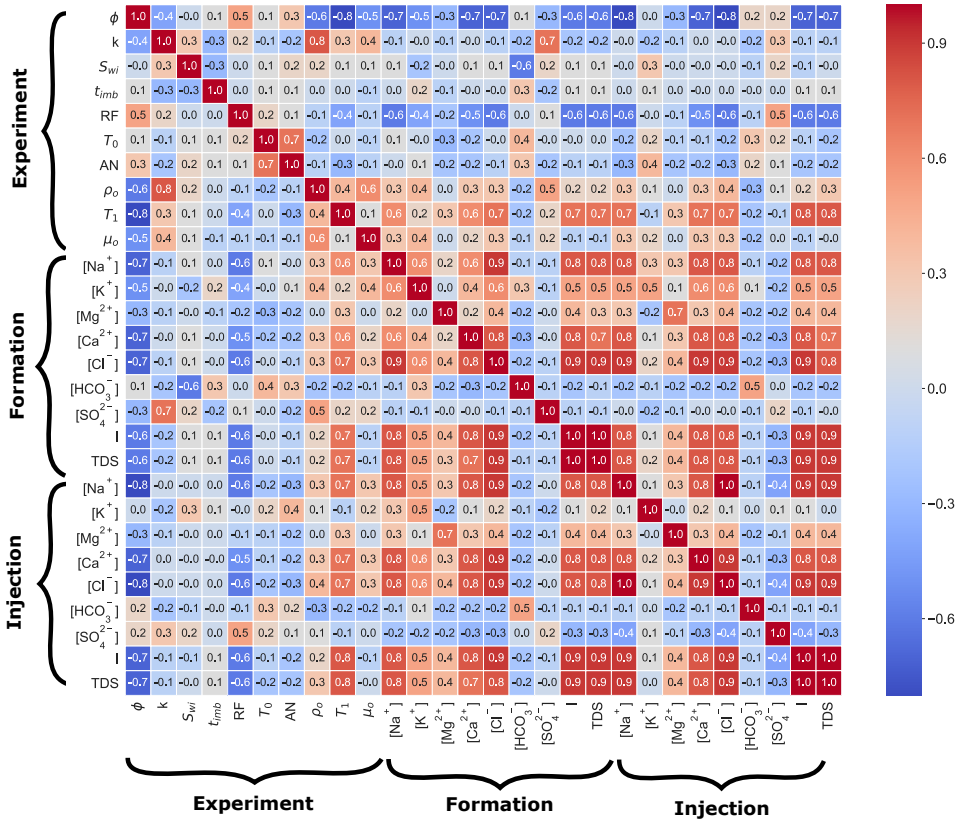


Figure 3: Confusion matrix of the dataset

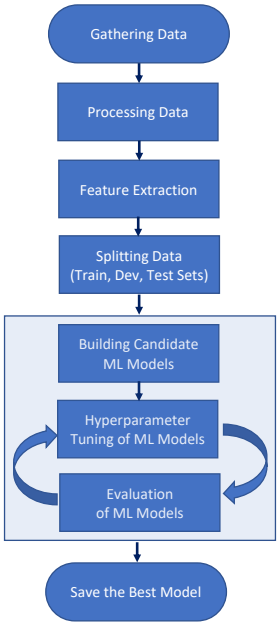
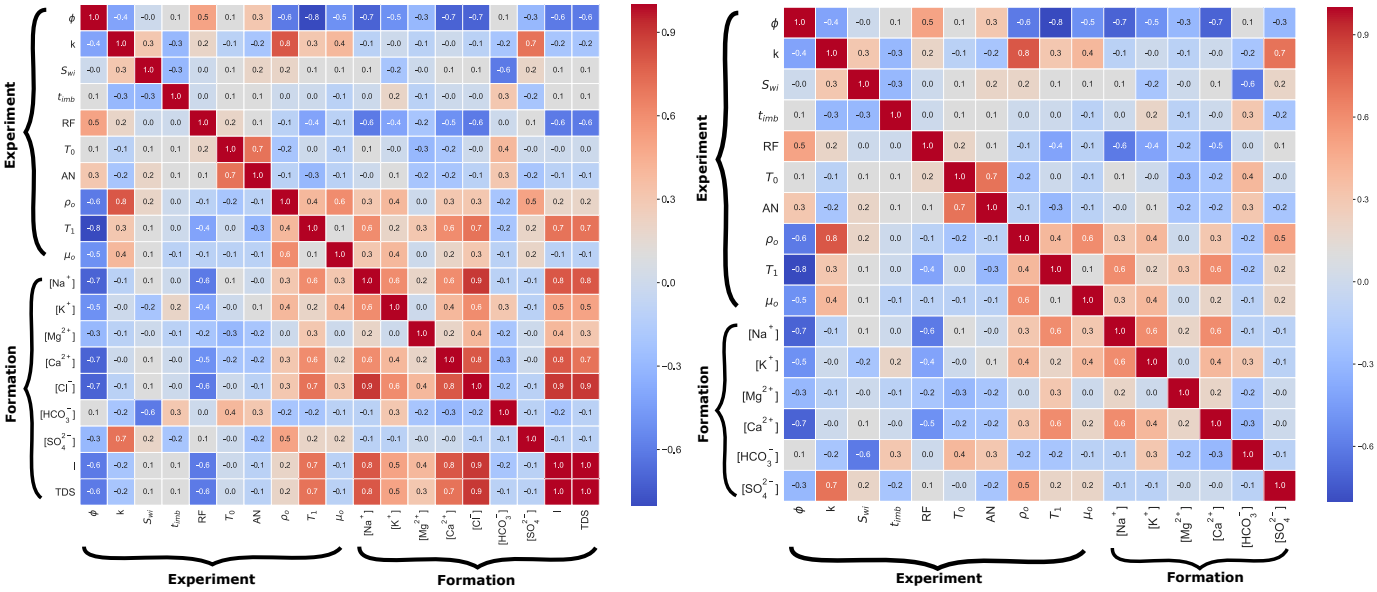


Figure 5: Prediction steps by using machine learning algorithms.

3.1 Model Evaluation Metrics

In order to evaluate the performance of different regression models, we use and report Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) as the evaluation metrics.



(a) Input features prior to removal of highly correlated ones

(b) Input features after removal of highly correlated ones

Figure 4: Input features part of the dataset - experiment setup properties and formation water components

3.1.1 Mean Absolute Error (MAE)

MAE measures the equally weighted average of the errors between the set of predictions ($\hat{y}_j$) and the set of actual observation values (y_j) in a set of n samples as

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|\,. \tag{3.1}$$

3.1.2 Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) is the square root of the average of squared differences between predicted and real data and can be expressed as

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}. \tag{3.2}$$

Root Mean Squared Error increases with the variance of the frequency distribution of error magnitudes. Although both MAE and RMSE are negatively-oriented metrics meaning that lower values are better, RMSE gives a relatively high weight to large errors and becomes important when existing large errors are particularly undesirable.

3.2 Multiple Linear Regression

Linear regression is a straightforward and fundamental approach to build predictive models. If there is more than one independent variable, we can exploit multiple linear regression (MLP). The form of this model for P-predictors is:

$$y_i = \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + \epsilon, \tag{3.3}$$

where X_i is i^{th} predictor, β_0 is intercept, β_i is i^{th} coefficient, and ϵ denotes the error. The best-fitting line is calculated by minimizing a cost function as the residual sum of squares between the real and predicted data in the dataset. The cost function is described as follows [7, 26].

$$\text{Cost} = \sum_{i=1}^{n}(\hat{y}_i - y_i)^2. \tag{3.4}$$

3.3 Lasso Regression

It is possible to use the lasso algorithm, which is classified as a modified linear regression. Lasso regression is similar to ridge regression with a slight modification in its cost function. Ridge regression pushes coefficients to approach zero (approximately); therefore, all independent variables remain in this model. The lasso model pushes them equal to be equal zero, so the lasso model reduces the number of predictors, and the algorithm applies feature selection automatically [7, 26]. The cost function is considered

$$\text{Cost} = \sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p}(\beta_j X_{ij})^2 + \lambda \sum_{j=1}^{p} |\,\beta_j|. \tag{3.5}$$

3.4 *K*-Nearest-Neighbour (KNN) Regression

K-Nearest-Neighbour (KNN) regression is a non-parametric regression model which is based on the k-nearest neighbours algorithm [7] where the input is comprised of the k neighbouring (closest) training examples in the feature space. KNN for regression has been used for different applications ([5, 34, 39, 42, 50] where a target value is predicted by local interpolation of the other target values associated with the nearest neighbours in the training set. In principle, the KNN-based regression uses an inverse distance weighted average of the K nearest neighbours to predict the value of a target sample in the dataset. The *Minkowki* function (L^p norm) can be used as the measure of similarity (distance function) for continuous data which is expressed as

$$D(x^i, x^j) = \left[\sum_{i}^{k}(x_d^i - x_d^j)^{\,p}\right]^{\frac{1}{p}}, \tag{3.6}$$

where for p values of 1 or 2, the Equation (3.6) respectively becomes *Manhattan* or *Euclidean* distance function. In general, KNN regression gives good predictive results for low-dimensional spaces in case enough data samples are available. This is because there would be enough nearby data points available for getting accurate predictions. However, as the number of dimensions increases the KNN regression loses its accuracy since the distance measure becomes meaningless and cannot represent a solid similarity metric for a significantly large number of dimensions. On the other hand, KNN regression is quite robust in case of having noisy training data as a weighted distance function is used.

There have been ways suggested on how to choose the value of k which is a model hyperparameter that needs to be tuned for each dataset under study independently [66, 32]. Intuitively, for a very low value of k, the underlying noise in the data will have a high influence on the results and the KNN model will overfit on the training data leading to a high error rate on the validation set. On the other hand, for a high value of k, building the model becomes computationally expensive, and the KNN model may perform poorly on the training and validation set. It has also been shown that a small value of k leads to the most flexible fit, having low bias but high variance, and a large value of k leads to a smoother decision boundary, having lower variance but higher bias, making it more resilient to outliers.

In this study, we use Euclidean distance as the similarity measure for reporting the results as based on our tests no large difference was observed compared to Manhattan distance for our dataset. For information retrieval, we take advantage ofthe KD-tree data structure as an efficient representation of the data in the KNN model. For the KD-tree we investigate the influence of the leaf size as an important hyperparameter that influences the accuracy of the information retrieval.

3.5 Decision Tree (DT) Regression

Decision trees (DT) have been used as a non-parametric machine learning model for different problems [70, 65, 38, 16, 41, 62, 53, 57] to perform a regression or classification task in a multistage hierarchical decision-making approach (tree-like structure). The main component of a DT model (tree) consists of a root node (i.e. all the data), a set of internal nodes (i.e. splits), and a set of external nodes (leaves). Each node of the decision tree structure makes a binary decision that separates classes from one another. In principle, a decision tree is based on the task of breaking down a complex decision into several simpler decisions, which may lead to a solution that is easier to interpret. In a decision tree, features of data (i.e. bands) are predictor variables and the classes or values to be mapped or predicted are target variables. In a decision tree model, features that carry a significant amount of information are automatically selected for the regression task and the remaining features are suppressed leading to more computational efficiency. In other words, in DT models, feature selection and regression are performed simultaneously which helps get over the curse of dimensionality problem (the Hughes phenomenon [67]), since merely a small number of features take part in building the model.

3.6 Random Forest (RF) Regression

Random Forest (RF) algorithm, first proposed by Breiman in 2001 [14], as an extension to the idea of decision trees has been used in practical applications mainly for purposes such as building predictive models (classification and regression), selection of features, data preprocessing, and predictive performance assessment [9, 10, 23, 37, 63]. Random Forest algorithm has been applied to several scientific research areas including but not limited to oil and gas ([19, 31]) agriculture ([37, 40, 68]), remote sensing ([8, 61], land classification ([29, 30]), biology and genomics ([17, 46, 48]), etc. Random-based methods are ensemble learning algorithms that use bootstrap aggregations (Bag) of classification and regression trees (CARTs) as the basis of the learning process. Random Forest algorithm as a supervised learning approach is proven to reduce the variance without increasing the bias during a prediction process, to well adapt to sparse data, to reach the minmax rate of convergence independent of noisy predictor variables, to capture nonlinear dependencies between predictor and dependent variables, to effectively handle small sample sizes, and to effectively handle missing data ([13, 15, 21, 47, 55, 56, 74]).

Besides, classification, Random Forest based approaches have been used for single/multiple output regression purposes. Svetnik et al. elaborate on the theory and details of using Random Forest for regression tasks [63]. In principle, RF-based regression is based on an ensemble model comprised of decision trees by which one/multiple continuous variable(s) are predicted as the average of the predictions from all the trees in the ensemble model.

In RF based regression task, an ensemble model of multiple regression trees is built from separate bootstrap samples of the training data using the Classification and Regression Tree (CART) algorithm [60]. The branches of each tree get subdivided and grow as long as the minimum number of observations for each leaf node is greater than a predetermined margin whereas, unlike the regression trees, the branches don't get pruned. The descriptor value that gets selected for branch splitting at any fork in any tree is selected from random subset values with a predetermined size. Random Forest has three main tuning hyperparameters as m_{try} being the number of descriptor values for each split, n_{tree} being the number of trees, and *nodesize* being the minimum number of nodes below which leaves are not further get subdivided. Comprised of multiple members of the dataset, a bootstrap sample is used during the tree growth by a random selection with replacement. The samples of the dataset which are not selected for the training of the trees get included as part of another subset so-called *out-of-bag (oob)* that can be used to evaluate the performance of the Random Forest model and provide an unbiased estimation of its generalization error. Random Forest also includes an approach for assessing the importance of each input feature where each feature is replaced by random noise and then the resulting decrease/increase in the model's outcome is considered as a measure of feature importance ([15, 49, 20]).

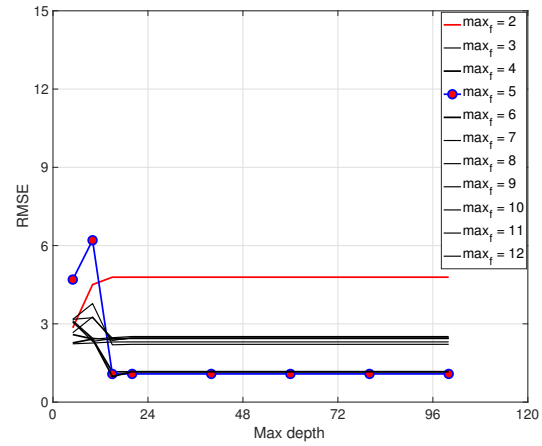
As briefly mentioned above, a machine learning model typically has several parameters that need to be learned from the data in the training procedure. Unlike model parameters, some hyperparameters must be determined outside the actual training procedure. Likewise, Random Forest, KNN, and Decision Tree have some special hyperparameters. Finding the best hyperparameters could have a significant effect on the prediction accuracy of these models.

Therefore, they should be optimized before the actual training process begins. In this study, several models were trained with different values of hyperparameters, while the RMSE and MAE were recorded for each model. Figures 6,7, and 8 show the effort to find optimized hyperparameters by minimizing the RMSE and MAE for two examples. Once the hyperparameters are optimized, the next step is to train a model on the entire dataset under the best hyperparameters.

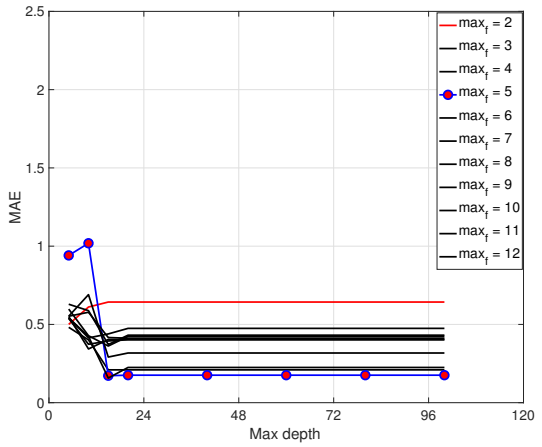
As an example in KNN regression, the P-value is a hyperparameter that determines the formula for calculating the distances. If p is equal to one, the algorithm will use the Manhattan distance while applying Euclidean distance if p is equal to two. Another hyperparameter in KNN is the number of neighbours to use. As a second example, in the Random Forest algorithm, a few hyperparameters must be defined. For instance, N-estimators determine the number of trees in the forest. It is necessary to optimize this parameter as adding growing more trees will notably slow down the training process; on the other hand, a model with limited trees may lose accuracy. Max-depth, Min-samples-leaf and max-features are other hyperparameters for Random Forest.

4 Results

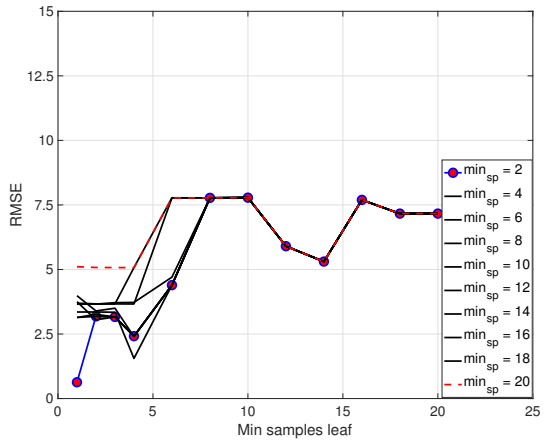
All development procedures to accurately predict ionic concentrations, the main aim of conducting MTR, briefly is described in the following. Firstly, outlier data was detected, and eliminated by using box plot tools. Then, all features were investigated using a confusing plot to determine the most correlated variables. In the following, five different algorithms were applied after optimizing their hyperparameters. The Random Forest algorithm makes the most accurate results compared to measured (actual) data for MTR. Moreover, For the reason of an easy comparison,



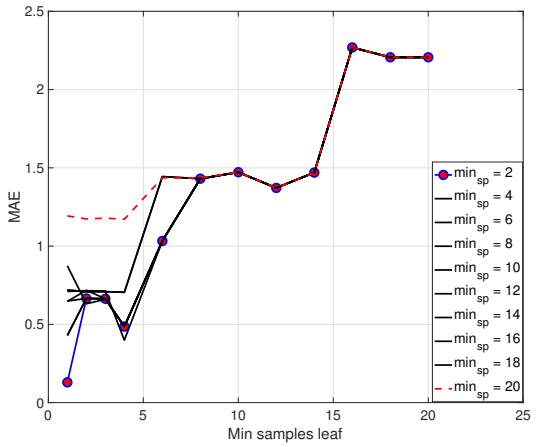
(a) DT1RMSE 1



(b) DT2MAE 1

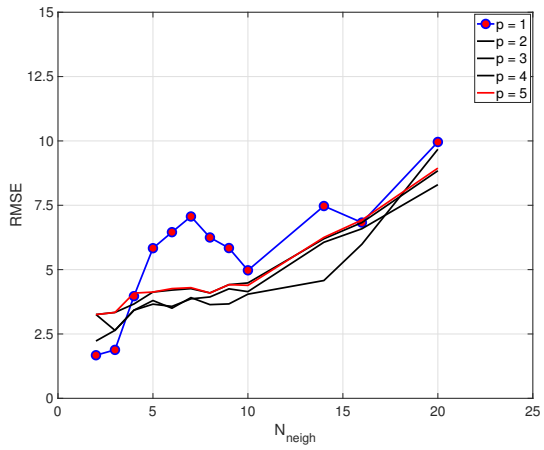


(c) DT3RMSE 1

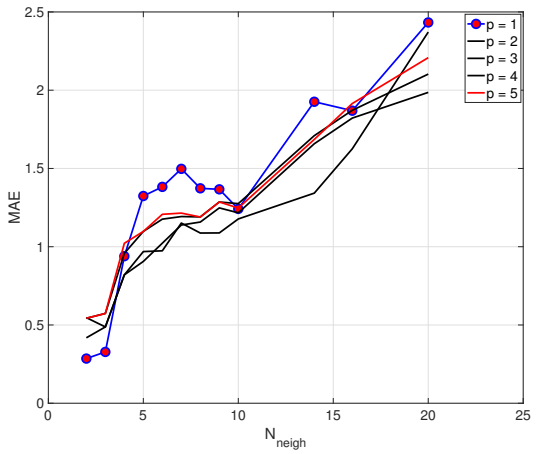


(d) DT4MAE 1

Figure 6: Determining the optimum value of hyperparameters, Decision Tree.

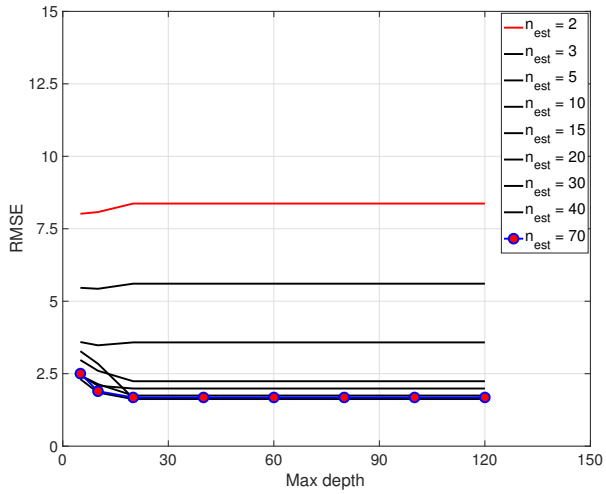


(a) KNN1RMSE-1

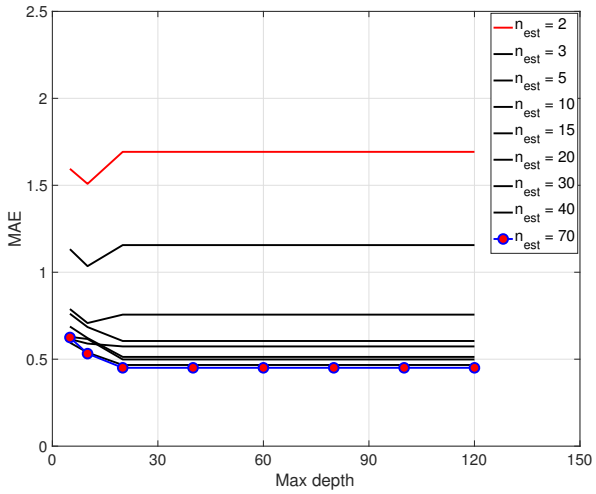


(b) KNN2MAE-1

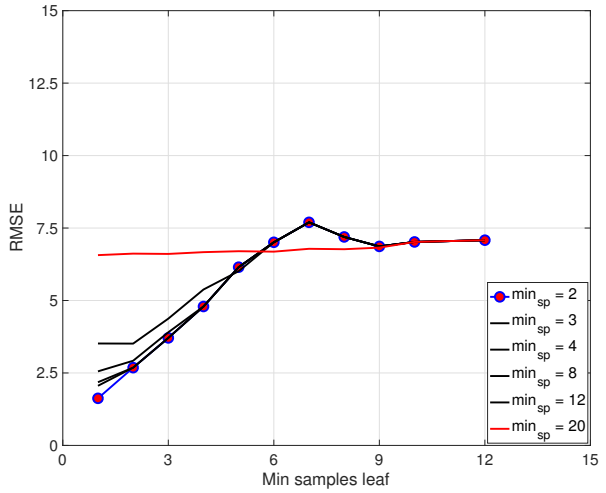
Figure 7: Determining the optimum value of hyperparameters, KNN.



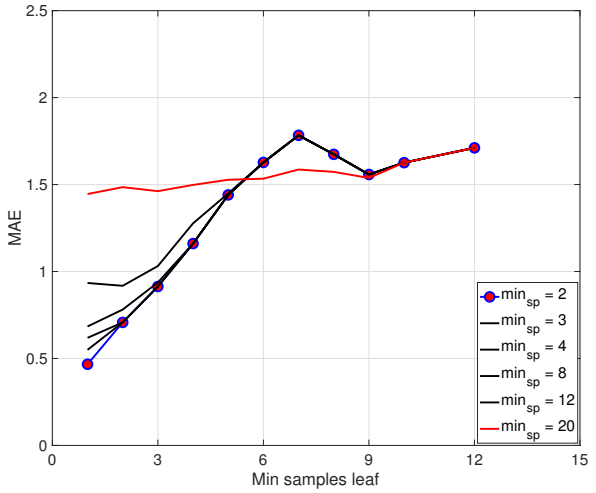
(a) RF3RMSE-1



(b) RF4MAE-1



(c) RF3RMSE-1



(d) RF4MAE-1

Figure 8: Determining the optimum value of hyperparameters, Random Forest.

the main results drawn of Figure 9 are summarized in Table 3. As can be seen, the Random Forest model showed the best performance in the test step.

Table 3: Statistical comparison between models

Regression Model	MAE	RMSE	Score
Linear	1.763	6.356	0.941
Lasso	3.188	10.891	0.826
k-Nearest Neighbors	0.328	1.882	0.730
Decision Tree	0.498	1.746	0.956
Random Forest	**0.259**	**1.231**	**0.981**

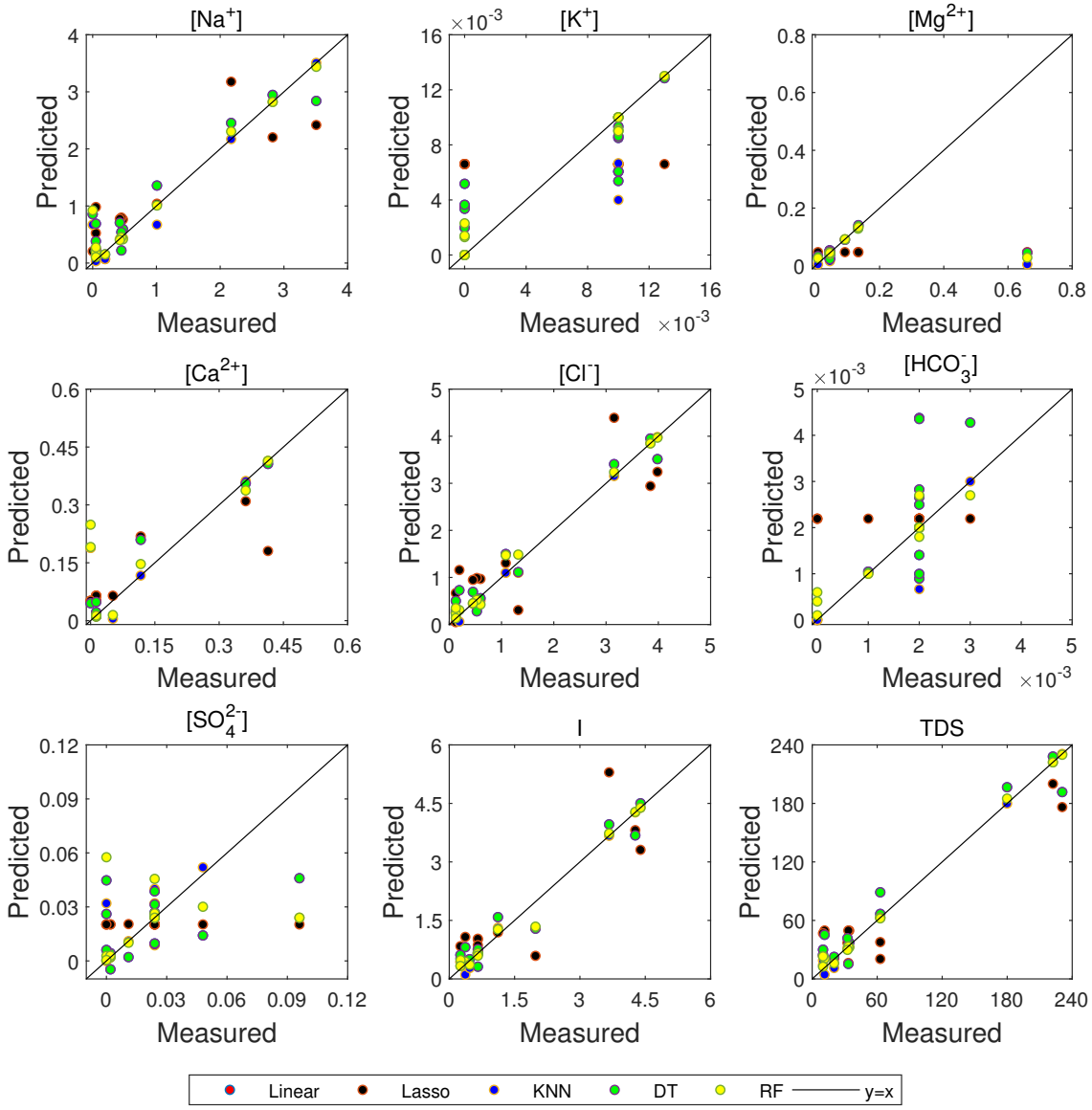


Figure 9: Real vs. predicted values for all algorithms

5 Conclusion

Considering that a wide range of factors mentioned in literature has a remarkable effect on ionic concentration measures simultaneously, therefore, being a multi-variant function, machine learning algorithms can be used to predict ionic concentration. In this paper, an innovative machine-learning approach was developed to accurately estimate the ionic concentrations for smart water injection without running expensive lab tests. An array of data, including a matrix of 29 features for 129 data sets, was collected and used for training different ML models. Regression models implemented in this study include Linear and other advanced regression models such as Lasso, k-nearest Neighbors, Decision Tree, and Random Forest. Hyperparameters for each model were first optimized before the training process. Models were then evaluated and ranked using errors and scores. Some of the key conclusions of this paper can be summarized as follows:

1. We have applied machine learning algorithms in a new field of petroleum engineering that showed high reliability and accuracy.

2. RMSE of predicted data versus lab for Linear, Lasso, k-Nearest Neighbors, Decision Tree, and Random Forest regression were 6.356, 10.891, 1.882, 1.746, and 1.231, respectively.

3. Random Forest exhibited the lowest error and highest accuracy for this particular data set.

References

[1] P. Ahmadi, H. Asaadian, S. Kord, and A. Khadivi, *Investigation of the simultaneous chemicals influences to promote oil-in-water emulsions stability during enhanced oil recovery applications*, J. Molecul. Liquids **275** (2019), 57–70.

[2] H.S. Al-Hadhrami and M.J. Blunt, *Thermally induced wettability alteration to improve oil recovery in fractured reservoirs*, SPE Reserv. Eval. Engin. **4** (2001), no. 3, 179–186.

[3] E.W. Al-Shalabi and K. Sepehrnoori, *A comprehensive review of low salinity/engineered water injections and their applications in sandstone and carbonate rocks*, J. Petrol. Sci. Engin. **139** (2016), 137–161.

[4] M.B. Alotaibi and H.A. Nasr-El-Din, *Chemistry of injection water and its impact on oil recovery in carbonate and clastics formations*, SPE Int. Conf. Oilfield Chem., SPE, 2009, pp. 121565.

[5] N.S. Altman, *An introduction to kernel and nearest-neighbor nonparametric regression*, Amer. Statist. **46** (1992), no. 3, 175–185.

[6] T. Austad and D.C. Standnes, *Spontaneous imbibition of water into oil-wet carbonates*, J. Petrol. Sci. Engin. **39** (2003), 363–376.

[7] E. Bahonar, Y. Ghalenoei, M. Chahardowli, M. Simjoo, *New correlations to predict oil viscosity using data mining techniques*, J. Petrol. Sci. Engin. **208** (2022), 109736.

[8] M. Belgiu and L. Drăgu, *Random forest in remote sensing: A review of applications and future directions*, ISPRS J. Photogram. Remote Sens. **114** (2016), 24–31.

[9] G. Biau, *Analysis of a random forests model*, J. Mach. Learn. Res. **13** (2012), 1063–1095.

[10] G. Biau and E. Scornet, *A random forest guided tour*, Test **25** (2016), 197–227.

[11] B. Boehmke and B.M. Greenwell, *Hands-on Machine Learning with R*, Chapman and Hall/CRC, 2019.

[12] H. Borchani, G. Varando, C. Bielza, and P. Larrañaga, *A survey on multi-output regression*, Wiley Interdiscip. Rev.: Data Min. Knowledge Discov. 5 (2015), no. 5, 216–233.

[13] A.L. Boulesteix, S. Janitza, J. Kruppa, and I.R. König, *Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics*, Wiley Interdiscip. Rev.: Data Min. Knowledge Disc. **2** (2012), no. 6, 493–507.

[14] L. Breiman, *Random forests*, Machine Learn. **45** (2001), 5–32.

[15] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*, 1st Edition, Chapman and Hall/CRC, 2017.

[16] M.Y. Chen, *Predicting corporate financial distress based on integration of decision tree classification and logistic regression*, Expert Syst. Appl. **38** (2011), 11261–11272.

[17] X. Chen and h. Ishwaran, *Random forests for genomic data analysis*, Genomics **99** (2012), 323–329.

[18] G.V. Chilingar and T.F. Yen, *Some notes on wettability and relative permeabilities of carbonate reservoir rocks, II*, Energy Sources **7** (1983), 67–75.

[19] H.E. Copeland, K.E. Doherty, D.E. Naugle, A. Pocewicz, and J.M. Kiesecker, *Mapping oil and gas development potential in the US intermountain west and estimating impacts to species*, PLoS ONE **4** (2009), 251–257.

[20] A. Criminisi, *Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning*, Found. Trends Comput. Graph. Vis. **7** (2011), 81–227.

[21] A. Criminisi, J Shotton, and E. Konukoglu, *Decision forests: A unified framework for classification, regression,*

density estimation, manifold learning and semi-supervised learning, Found. Trends® Comput. Graph. Vis. **7** (2012), no. 2–3, 81–227.

[22] C. Dang, L. Nghiem, E. Fedutenko, E. Gorucu, C. Yang, A. Mirzabozorg, N. Nguyen, and Z. Chen, *AI based mechanistic modeling and probabilistic forecasting of hybrid low salinity chemical flooding*, Fuel **261** (2020), 116445.

[23] B. Efron and T. Hastie, *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*, Cambridge University Press, 2016.

[24] S.J. Fathi, T. Austad, and S. Strand, *Smart water as a wettability modifier in chalk: the effect of salinity and ionic composition*, Energy Fuels **24** (2010), 2514–2519.

[25] S.J. Fathi, T. Austad, and S. Strand, *Water-based enhanced oil recovery (EOR) by smart water: Optimal ionic composition for EOR in carbonates*, Energy Fuels **25** (2011), 5173–5179.

[26] J. Gareth, W. Daniela, H. Trevor, and T. Robert, *An Introduction to Statistical Learning*, Springer, New York, 2013.

[27] J. Ghosn, and Y. Bengio, *Multi-Task Learning for Stock Selection*, Adv. Neural Inf. Process. Syst. **9** (1996), 946–952.

[28] J. Gillberg, P. Marttinen, M. Pirinen, A.J. Kangas, P. Soininen, M. Ali, A.S. Havulinna, M.-R. Järvelin, M. Ala-Korpela, and S. Kaski, *Multiple output regression with latent noise*, J. Machine Learn. Res. **17** (2016), no. 122, 1–35.

[29] P.O. Gislason, J.A. Benediktsson, and J.R. Sveinsson, *Random forest classification of multisource remote sensing and geographic data*, IEEE Int. Geosci. Remote Sens. Symp., 2004, pp. 1049–1052.

[30] P.O. Gislason, J.A. Benediktsson, and J.R. Sveinsson, *Random forests for land cover classification*, Pattern Recogn. Lett. **27** (2006), 294–300.

[31] E. Goel and E. Abhilasha, *Random forest: A review*, Int. J. Adv. Res. Comput. Sci. Software Engin. **7** (2017), 251–257.

[32] P. Hall, B.U. Park, and R.J. Samworth, *Choice of neighbor order in nearest-neighbor classification*, Ann. Statist. **36** (2008), 2135–2152.

[33] L.D. Hallenbeck, J.E. Sylte, D.J. Ebbs, and L.K. Thomas, *Implementation of the Ekofisk field waterflood*, SPE Form. Eval. **6** (1991), 284–290.

[34] T. Hastie and R. Tibshirani, *Discriminant adaptive nearest neighbor classification and regression*, Adv. Neural Inf. Process. Syst. **8** (1995), 409–415.

[35] G. Hirasaki and D.L. Zhang, *Surface chemistry of oil recovery from fractured, oil-wet, carbonate formations*, SPE J. **9** (2004), 151–162.

[36] P.A. Hopkins, I. Omland, F. Layti, S. Strand, T. Puntervold, and T. Austad, *Crude oil quantity and its effect on chalk surface wetting*, Energy Fuels **31** (2017), 4663–4669.

[37] G.F. Hughes, *On the mean accuracy of statistical pattern recognizers*, IEEE Trans. Inf. Theory **14** (1968), 55–63.

[38] Z. Ibrahim and D. Rusli *Predicting students' academic performance: Comparing artificial neural network, decision tree and linear regression*, 21st Ann. SAS Malaysia Forum, 2007, pp.1–6.

[39] S.B. Imandoust and M. Bolandraftar, *Application of k-nearest neighbor (kNN) approach for predicting economic events: Theoretical background*, J. Engin. Res. Appl. **3** (2013), no. 5, 605–610.

[40] J.H. Jeong, J.P. Resop, N.D. Mueller, D.H. Fleisher, K. Yun, E.E. Butler, D.J. Timlin, K.M. Shim, J.S. Gerber, V.R. Reddy, and S.H. Kim, *Random forests for global and regional crop yield predictions*, PLoS ONE **11** (2016), 1–15.

[41] R.J. Lewis, *An introduction to classification and regression tree (CART) analysis*, Ann. Meet. Soc. Acad. Emergency Med. San Francisco, California, Vol. 14. San Francisco, CA, USA: Department of Emergency Medicine Harbor-UCLA Medical Center Torrance, 2000.

[42] C.D. Manning and H. Schütze *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.

[43] G. Melki, and A. Cano, V. Kecman, and S. Ventura, *Multi-target support vector regression via correlation regressor chains*, Inf. Sci. **415** (2017), 53–69.

[44] S. Mohammadi, S. Kord, and J. Moghadasi, *An experimental investigation into the spontaneous imbibition of surfactant assisted low salinity water in carbonate rocks*, Fuel **243** (2019), 142–154.

[45] N.R. Morrow, *Wettability and its effect on oil recovery*, J. Petrol. Technol. 42 (1990), no. 12, 1476–1484.

[46] C. Nguyen, Y. Wang, and H.N. Nguyen, *Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic*, J. Biomed. Sci. Engin. **6** (2013), 551–560.

[47] O. Okun and H. Priisalu, *Random forest for gene expression based cancer classification: overlooked issues*, Iberian Conf. Pattern Recog. Image Anal., Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 483–490.

[48] M. Pal, *Random forests for land cover classification*, IEEE Int. Geosci. Remote Sens. Symp., Proc.(IEEE Cat. No. 03CH37477). Vol. 6. IEEE, 2003, pp. 3510–3512.

[49] D.S. Palmer, N.M. O'Boyle, R.C. Glen, and J.B.O. Mitchell, *Random forest models to predict aqueous solubility*, J. Chem. Inf. Model. **47** (2007), 150–158.

[50] L.E. Peterson, *K-nearest neighbor*, Scholarpedia **4** (2009), no. 2, 1883.

[51] T. Puntervold, S. Strand, R. Ellouz, and T. Austad, *Modified seawater as a smart EOR fluid in chalk*, J. Petrol. Sci. Engin. **133** (2015), 440–443.

[52] J. Romanuka, J. Hofman, D.J. Ligthelm, B.M. Suijkerbuijk, A.H. Marcelis, S. Oedai, N.J. Brussee, A. van der Linde, H. Aksulu, and T. Austad, *Low salinity EOR in carbonates*, SPE Improved Oil Recovery Conf., SPE, 2012.

[53] P. Royston and D.G. Altman, *Risk stratification for in-hospital mortality in acutely decompensated heart failure*, Jama **293** (2005), no. 20, 2467–2468.

[54] S.F. Shariatpanahi, P. Hopkins, H. Aksulu, S. Strand, T. Puntervold, and T. Austad, *Water based EOR by wettability alteration in dolomite*, Energy Fuels **30** (2016), no. 1, 180–187.

[55] E. Scornet, *On the asymptotics of random forests*, J. Multivar. Anal. **146** (2016), 72–83.

[56] E. Scornet, G. Biau, J.P. Vert, *Consistency of random forests*, Ann. Statist. **43** (2015), 1716–1741.

[57] Y.Y. Song and L.U. Ying, *Decision tree methods: Applications for classification and prediction*, Shanghai Arch. Psych. **27** (2015), no. 2, 130–135.

[58] D.C. Standnes and T. Austad, *Wettability alteration in carbonates: Interaction between cationic surfactant and carboxylates as a key factor in wettability alteration from oil-wet to water-wet conditions*, Colloids Surfaces A: Physicochem. Engin. Aspects **216** (2003), 243–259.

[59] S. Strand, T. Puntervold, and T. Austad, *Effect of temperature on enhanced oil recovery from mixed-wet chalk cores by spontaneous imbibition and forced displacement using seawater*, Energy Fuels **22** (2008), no. 5, 3222–3225.

[60] V. Svetnik, A. Liaw, C. Tong, J.C. Culberson, R.P. Sheridan, and B.P. Feuston, *Random forest: A classification and regression tool for compound classification and QSAR modeling*, J. Chem. Inf. Comput. Sci. **43** (2003), no. 6, 1947–1958.

[61] K. Tatsumi, Y. Yamashiki, M.A.C. Torres, and C.L.R. Taipe, *Crop classification of upland fields using Random forest of time-series Landsat 7 ETM+ data*, Comput. Electron. Agricul. **115** (2015), 171–179.

[62] G.K.F. Tso and K.K.W. Yau, *Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks*, Energy **32** (2007), 1761–1768.

[63] 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 **11** (2019), no. 5, 910.

[64] X. Xi, V.S. Sheng, B. Sun, L. Wang, and F. Hu, *An empirical comparison on multi-target regression learning*, Comput. Mater. Continua 56 (2018), no. 2, 185–198.

[65] M. Xu, P. Watanachaturaporn, P.K. Varshney, and M.K. Arora, *Decision tree regression for soft classification of remote sensing data*, Remote Sens. Envir. **97** (2005), 322–336.

[66] Z. Yao and W.L. Ruzzo, *A regression-based K nearest neighbor algorithm for gene function prediction from heterogenous data*, BMC Bioinf. **7** (2006), 1–11.

[67] Z. Yu, F. Haghighat, B.C.M. Fung, and H. Yoshino, *A decision tree method for building energy demand modeling*, Energy Build. **27** (2015), 1637–1646.

[68] H.M. Zawbaa, M. Hazman, M. Abbass, and A.E. Hassanien, *Automatic fruit classification using random forest algorithm*, 14th Int. Conf. Hybrid Intell. Syst., 2014, pp. 164–168.

[69] P. Zhang and T. Austad, *Wettability and oil recovery from carbonates: Effects of temperature and potential determining ions*, Colloids Surfaces A: Physicochem. Engine. Aspects **279** (2006), 179–187.

[70] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, *Learning k for kNN classification*, ACM Trans. Intell. Syst. Technol. **8** (2017), no. 3, 1–19.

[71] Y. Zhang and N.R. Morrow, *Comparison of secondary and tertiary recovery with change in injection brine composition for crude oil/sandstone combinations*, SPE Improved Oil Recovery Conf., SPE, 2006.

[72] X. Zhen, M. Yu, X. He, and S. Li, *Multi-target regression via robust low-rank learning*, IEEE Trans. Pattern Anal. Machine Intell. **40** (2018), 497–504.

[73] X. Zhen, M. Yu, F. Zheng, I.B. Nachum, M. Bhaduri, D. Laidley, D and S. Li, *Multitarget sparse latent regression*, IEEE Trans. Neural Networks Learn. Syst. **29** (2017), no. 5, 1575–1586.

[74] A. Ziegler and I.R. König, *Mining data with random forests: Current options for real-world applications*, Wiley Interdiscip. Rev.: Data Min. Knowledge Disc. **4** (2014), 55–63.