

Estimating oil and gas recovery factor via machine learning

by

Alireza Roustazadeh

B.S., Texas A&M University, 2018

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Geology
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2022

Approved by:

Co-Major Professor
Mohammad B. Shadmand

Approved by:

Co-Major Professor
Behzad Ghanbarian

Copyright

© Alireza Roustazadeh 2022.

Abstract

With recent advances in artificial intelligence, machine learning (ML) approaches have become an attractive tool in petroleum engineering, particularly for reservoir characterizations. A key reservoir property is hydrocarbon recovery factor (RF) whose accurate estimation would provide decisive insights to drilling and production strategies. Therefore, this study aims to estimate the hydrocarbon RF from various reservoir characteristics, such as porosity, permeability, pressure, and water saturation via the ML. We applied three regression-based models including the extreme gradient boosting (XGBoost), support vector machine (SVM), and stepwise multiple linear regression (MLR) and various combinations of three databases to construct ML models and estimate the oil and/or gas RF. Using two databases and the cross-validation method, we evaluated the performance of the ML models. In each iteration 90 and 10% of the data were respectively used to train and test the models. The third independent database was then used to further assess the constructed models. For both oil and gas RFs, we found that the XGBoost model estimated the RF for the train and test datasets more accurately than the SVM and MLR models. However, the performance of all the models were unsatisfactory for the independent databases. Results demonstrated that the ML algorithms were highly dependent and sensitive to the databases based on which they were trained. Results of statistical tests revealed that such unsatisfactory performances were because the distributions of input and output features in the train datasets were significantly different from those in the independent databases (p -value < 0.05).

Table of Contents

List of Figures	v
List of Tables	vi
Acknowledgements	vii
Dedication	viii
Chapter 1 - Introduction	1
Chapter 2 - Materials and Methods	5
2.1. Databases	5
2.1.1. Commercial database	5
2.1.2. Bureau of Ocean Energy Management (BOEM) Atlas of Gulf of Mexico	5
2.1.3. Tertiary Oil Recovery Information System (TORIS) database	5
2.1.4. Gas Information System (GasIS) database	6
2.2. Data preparation	8
2.3. Machine learning algorithms	12
2.3.1. XGBoost	12
2.3.2. Support Vector Machine	13
2.3.3. Forward selection stepwise multiple linear regression	13
2.4. Learning curves	13
2.5. Hyperparameter tuning	14
2.6. Feature importance	16
2.7. Workflow of the constructed models	16
2.8. Model evaluation criteria	17
Chapter 3 - Results	18
3.1. Learning curves	18
3.2. Oil RF estimations	19
3.3. Gas RF estimations	24
3.4. Feature importance	28
Chapter 4 - Discussion	31
Chapter 5 - Conclusion	37
References	38

List of Figures

Figure 2.1. Oil and gas recovery factors against permeability (millidarcy) and porosity (ft ³ /ft ³) for different databases used in this study.	7
Figure 2.2. Imputation steps used to replace missing values in all the databases used to construct machine learning models.	11
Figure 2.3. The scheme of different possible learning curves and their over/underfitting interpretations due to sample size.	14
Figure 2.4. The workflow used here to develop the ML-based models and evaluate them. Normalization means converting the distribution of a feature to become approximately Gaussian, while standardization means scaling data between 0 and 1.	17
Figure 3.1. Learning curves of the XGBoost models developed for the (top) oil and (bottom) gas RFs using the CA (Commercial combined with Atlas) database.	19
Figure 3.2. The estimated oil RF by the XGBoost algorithm versus that measured for the train and test datasets as well as the independent database. CA represents Commercial combined with Atlas, TC denotes TORIS combined with Commercial, TA is TORIS combined with Atlas.	22
Figure 3.3. The estimated oil RF by the XGBoost algorithm versus that measured for the train and test datasets as well as the independent database. GC denotes GasIS combined with Commercial, GA is GasIS combined with Atlas, CA represents Commercial combined with Atlas.	27
Figure 3.4. Feature importance of the XGBoost model in the estimation of oil RF (top) and gas RF (bottom) models for the databases CA (with oil RF as target variable) and CA (with gas RF as target variable). Since the data were scaled between 0 and 1, the blue color denotes the smaller values close to 0, while the red color represents greater values near 1. The positive SHAP values indicate positive influence on the model construction and more accurate prediction. However, the negative SHAP values show the negative impact on the model.	30

List of Tables

Table 2.1. Features selected to construct the machine learning-based models and numerical intervals for each feature used in oil and gas databases.....	9
Table 2.2. Optimized hyperparameters of the XGBoost models developed for the oil and gas RFs obtained on the CA (Commercial combined with Atlas) database.	15
Table 2.3. Optimized hyperparameters of the SVM models developed for the oil and gas RFs obtained on the CA (Commercial combined with Atlas) database.	15
Table 3.1. Calculated RMSE, R2 and r values for the three algorithms XGBoost, SVM, and MLR used to estimate the oil RF. The train datasets of TC, TA, CA, and TCA databases consisted of 1502, 4779, 4654, and 5461 number of samples, respectively.	21
Table 3.2. Two samples with similar input features but different oil RFs (top two) and two samples with different input features but similar oil RFs (bottom two).	23
Table 3.3. Calculated RMSE, R2 and r values for the three algorithms XGBoost, SVM, and MLR used to estimate the gas RF. The train datasets of GC, GA, CA, and GCA databases consisted of 5838, 10040, 10251, and 15878 number of samples, respectively.	26
Table 3.4. Two samples with similar features but different gas RF (top two) and two samples with different features but similar gas RF (bottom two).....	28
Table 4.1. p-values determined using the t-test by comparing the RF values in the train datasets with those in the test datasets as well as those in the independent databases for both oil and gas at the significance level of 0.05 (confidence level of 0.95). The <i>p-values</i> were calculated using the paired t-test and <i>unequal variances</i>	33
Table 4.2. Results of the Kolmogorov-Smirnov test for the top three input features and RFs in the CA database and the corresponding independent databases TORIS (for oil) and GasIS (for gas). The significance level is 0.05.	35

Acknowledgements

The author is grateful to Vahid Taslimitehrani, Realtor.com, Frank Male, University of Texas at Austin, and Amirhossein Yazdavar, United Healthcare INC, for fruitful discussions and comments.

The author would like to acknowledge Department of Geology at Kansas State University, for financial supports through William J. Barret Fund for Excellence in Geology and Gary & Kathie Sandlin Geology Scholarship. Alireza Roustazadeh, Behzad Ghanbarian, and Mohammad B. Shadmand acknowledge Kansas State University for support through faculty start-up fund. The Python codes as well as means, standard deviations, and distributions of input features used in this study are available at: <https://github.com/alirezaro93/Estimating-hydrocarbon-recovery-factor-at-reservoir-scale-via-machine-learning>

Dedication

I would like to dedicate this work to my parents and siblings who through their hard work, love, and endless support, allowed me to reach the pinnacles of success and happiness. This work and any endeavor I have had in the past were not possible if I did not have them in my life. I hope this work would be a small gratitude to them and honors their love and support.

Chapter 1 - Introduction

Successful exploration and production of a hydrocarbon reservoir have always been challenging in petroleum engineering. Because of high fluctuations in oil and gas prices, it is necessary to determine, if a field is economically feasible to be invested or not. A conventional reservoir goes through different stages during its lifetime, namely exploration, appraisal, development, production, and lastly abandonment. Once a reservoir is discovered (exploration stage), the main challenge is to determine whether it has high potential to produce substantial amount of hydrocarbon and subsequently yield enough profit. Even for a currently producing reservoir, one still needs to analyze its performance and production rate to assure profits above economical margins.

The performance of a reservoir influences its economic feasibility and is a function of reservoir quality. Reservoir performance may be quantified by the initial production rate of a reservoir and the decline in its production rate. Another indicator is the amount of recovered hydrocarbon from the initial volume of hydrocarbon in a reservoir (Beaumont et al. 2019), also known as recovery factor (RF). The RF, ranging between 0 and 1, is a quantity to evaluate the fate of a reservoir. It is a function of displacement mechanisms, such as water drive, gas cap, rock compaction drive, solution gas, and gravity drainage (Schlumberger 2020a). Its value may be determined at different stages of a reservoir lifetime. However, in this study we refer to *ultimate* recovery factor by RF.

There exist different methods to determine the RF during the lifetime of a reservoir. When a field is in its appraisal phase and no production data are available, analogs and empirical formulas, e.g., history matching and volumetric reserve estimation, are commonly used. Such methods, however, generally come with substantial uncertainties due to the lack of adequate

analogs (Aliyuda & Howell 2019). During the production phase, dynamic properties of a field vary with production and time and, thus, the RF determination is influenced by enhanced and improved recovery methods as well as more data being collected from a field. Once approaching the abandonment phase, the emphasis should be more on economical margin, and how close the field production is to such a margin. At the plateau production phase, the RF is typically determined by simulations and dynamic reservoir modeling. However, at late stages of production and when the plateau production is no longer in place, the decline curve analysis should be analyzed. Various methods used at different field life stages may lead to different RF determinations and uncertainties (Aliyuda & Howell 2019). In addition to the decline curve analysis, one may apply the material balance method to determine the reserve and, accordingly, the recovery factor (Maselugbo et al., 2017; Omoniyi & Adeolu, 2014). Both the material balance and production decline curve analysis methods calculate reserves and recovery factor from the performance of a field (Omoniyi & Adeolu 2014).

The aforementioned methods are time consuming, and their results may have a wide range of uncertainties (Ling et al. 2013; Parish et al. 1993; Muskat & Woods 1944). Moreover, the RF is a function of multiple variables e.g., permeability, temperature, and gas oil ratio (GOR), which may not be necessarily incorporated in such methods. In addition, those methods are not cost efficient. With recent advances in artificial intelligence and data analytics, one may construct a machine learning-based model to estimate the RF from other available reservoir characteristics. Such models may provide a cost efficient and accurate platform as well as better insights into reservoir characterization and hydrocarbon production, if implemented appropriately.

Machine learning (ML) techniques have been previously used to estimate the RF (Aliyuda et al. 2020; Alpak et al. 2019; Chen et al. 2019; Sharma et al. 2010). For example, Lee & Lake

(2015) applied various methods including multiple linear regression (MLR), MLR with sequential feature selection, artificial neural networks (ANN), and Bayesian network to estimate both oil and gas RFs. In their study, the ANN and MLR with sequential feature selection showed better performance than the other two methods. Aliyuda & Howell (2019) applied the MLR and support vector machine (SVM) with Gaussian kernel on 93 reservoirs from the Norwegian Continental Shelf. 75 reservoirs were from the Norwegian Sea, the Norwegian North Sea, and the Barents Sea, and the remaining 18 reservoirs from the Viking Graben in the UK sector of the North Sea. Aliyuda & Howell (2019) found that the SVM model outperformed the MLR model.

In another study, Chen et al. (2019) applied the ANN approach to develop predictive oil RF models with different sets of input data using the database TORIS and identified 19 principal features (out of 70) in the construction of their ML model. Tewari et al. (2019) used six different approaches, i.e., random tree, random forest, SVM, bagging, radial basis function, and multilayer perceptron and compared them with their proposed ensemble estimator (E2) model. Ensemble methods are classifier algorithms that develop multiple classifiers and make inferencing using the weighted vote of multiple estimations that they have made (Dietterich 2000). Tewari et al. (2019) showed that their E2-based model outperformed the other six methods.

In the literature, most ML studies divide a database into train and test splits. Although the performance of ML models is evaluated using the test dataset, an independent database with no data overlap is not used to further assess the accuracy and reliability of ML models constructed. Therefore, the main objectives of this study are to: (1) apply the extreme gradient boosting (XGBoost) approach to develop an ML-based model, (2) compare its performance with that of multiple linear regression (MLR) and support vector machine (SVM) in the estimation of oil and

gas RFs using reservoirs from around the world, and (3) address the database dependence of accuracy and uncertainty in ML-based models.

Three main hypothesis were devised for this project and they were: (1) An efficient machine learning model proposed for estimating the RF should identify reservoir drive mechanism attributes (e.g., pressure and temperature production rate) and reservoir quality factors (e.g., porosity and permeability) as important features. (2) The chosen algorithm should create the model based on reservoir drive mechanism attributes and reservoir quality factors, such as porosity, permeability, initial reserves, reservoir pressure and temperature, and (3) the models constructed should have a satisfactory and relatively accurate estimations on new and unseen data

Chapter 2 - Materials and Methods

2.1. Databases

2.1.1. Commercial database

The commercial database analyzed in this study is the same one that Lee & Lake (2015) used. This database contains more than 1200 samples including reservoirs with different geologic formations from around the world. The oil RF values were reported for more than 600 samples, while the gas RF values only available for more than 300 cases. For each sample representing a reservoir, there exist more than 200 numerical and categorical features, such as porosity, permeability, reservoir fractures, and recovery factor.

2.1.2. Bureau of Ocean Energy Management (BOEM) Atlas of Gulf of Mexico

This database contains the oil and gas reserves of the outer continental shelf region of Gulf of Mexico (Burgess et al. 2019). There exist more than 13000 samples and more than 80 features for each sample. This is the most complete database with only a few missing values. It includes samples with both oil and gas recovery factor values.

2.1.3. Tertiary Oil Recovery Information System (TORIS) database

TORIS is a suite of databases gathered by the National Petroleum Council and used by the US Department of Energy to identify and characterize US oil resources and further strategize their plans. It consists of more than 1300 reservoirs (samples) and more than 60 categorical and numerical features for each sample. TORIS is an old database strictly compiled for oil reservoirs and was used in combination with other databases to estimate the oil RF only.

2.1.4. Gas Information System (GasIS) database

GasIS is a merged database, a combination of multiple pre-existing datasets. It is the first national public database containing information on multiple reservoir properties and ultimate reservoir recovery for gas reservoirs. Created by the Department of Energy (Hugman et al., 2000), GasIS consists of gas reservoirs with more than 19000 samples and 186 categorical and numerical features. The database, however, has many missing values. After cleaning up by omitting the samples with missing RF values, only 8524 samples were remained.

The Commercial and Atlas databases contain both oil and gas RF data, while the GasIS and TORIS databases only the gas and oil RF data, respectively. To avoid data overlap among the databases used in this study, we assured samples were presented in only one database and not repeated in other databases. Figure 2.1 presents the oil and gas RFs versus the permeability (millidarcy) and porosity (ft^3/ft^3) for the databases used in this study. This figure shows the range of each variable and the trend between the RF and permeability as well as porosity within each database. As can be seen, the plots are scattered, and there is no strong correlation between the RF and permeability or porosity.

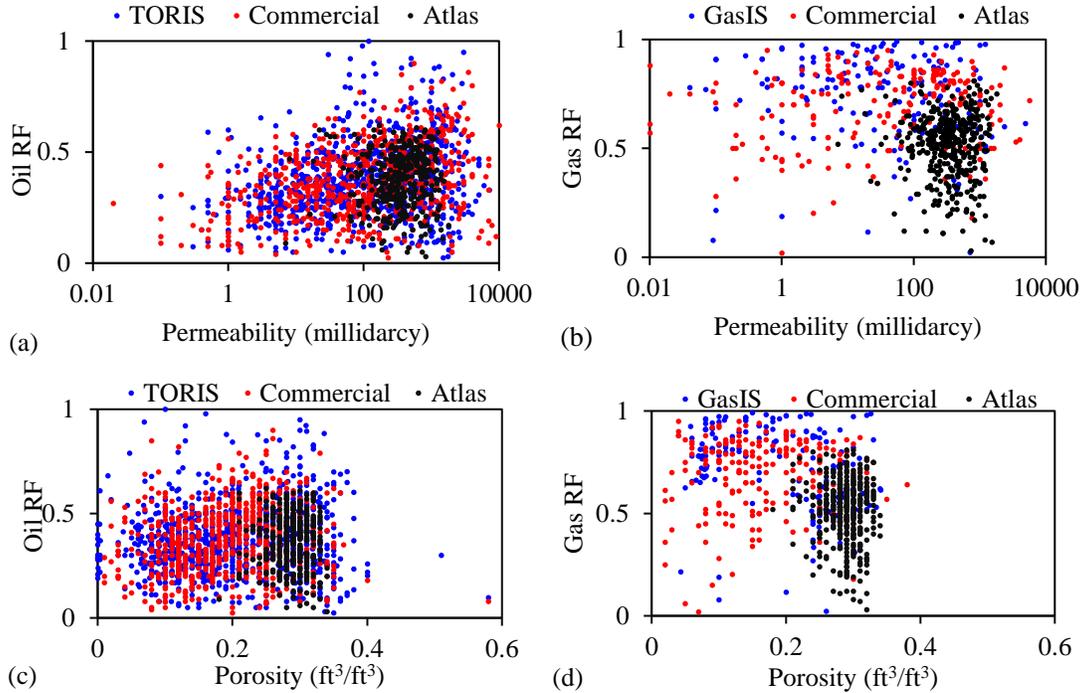


Figure 2.1. Oil and gas recovery factors against permeability (millidarcy) and porosity (ft³/ft³) for different databases used in this study.

To construct the ML-based models and address their database-dependent accuracy and reliability in the estimation of the oil and gas RFs, the databases collected in this study were merged using different combinations. For example, for the oil RF we combined the TORIS and Commercial databases (named TC), TORIS and Atlas databases (named TA), Commercial and Atlas databases (named CA), and all the three databases (named TCA). Similar approach was used for the gas RF; CA represents Commercial combined with Atlas, GC denotes GasIS combined with Commercial, GA is GasIS combined with Atlas, and GCA represents GasIS combined with commercial and Atlas. Whenever two databases were merged, the third one was used as an independent database for further evaluation of the constructed ML models.

2.2. Data preparation

We first selected those features that were common in all the oil or gas databases e.g., porosity, permeability, pressure, reservoir thickness, temperature, and reserves. All the features that had reservoir information after the exploration phase e.g., production rate and cumulative production were removed simply because we aimed to propose a model estimating the RF from data collected at early stages of a reservoir's life. To ensure that the selected features were not highly correlated with each other, Spearman's rank correlation coefficient (also known as Spearman's Rho test) was used to determine the correlation between any two features. Table 2.1 lists those features used in the ML algorithms and their limits in both oil and gas databases. Reserves in the GasIS and Atlas databases indicate the amount of hydrocarbon remaining in reservoirs that are commercially recoverable. Although it is not clear what reserves represent in the Commercial and TORIS databases, we assumed the same definition as in GasIS and Atlas.

After the feature selection, we sorted the data based on the target feature (i.e., RF) and omitted samples with missing RF values followed by removing outliers. To do that, each feature was capped from a lower to an upper limit depending on each feature nature and its histogram. Finally, we removed any samples that had more than 55% of its values missing.

Table 2.1. Features selected to construct the machine learning-based models and numerical intervals for each feature used in oil and gas databases.

Oil	Oil databases	Gas	Gas databases
API Gravity	$1 \leq B_o \leq 3$	GOR (MSCF/RB)	$0 \leq \text{GOR} \leq 10^9$
B_o (RB/STB)	$0 \leq \text{GOR} \leq 60$	Water saturation	$0 \leq S_w \leq 1$
GOR* (MSCF/RB)	$0 \leq S_w \leq 1$	Temperature (F)	$0 \leq \text{Reserves} \leq 4 \times 10^8$
Water saturation	$0 \leq \text{Reserves} \leq 5 \times 10^{11}$	Pressure (psi)	$0 \leq \phi \leq 1$
Temperature (°F)	$0 \leq \phi \leq 1$	Thickness (ft)	$0 \leq \text{Gas RF} \leq 1$
Pressure (psi)	$0 \leq \text{Oil RF} \leq 1$	Reserves (MMSCF)	
Thickness (ft)		Permeability (mD)	
Reserves (STB)		Porosity (ft ³ /ft ³)	
Permeability (mD)		Area (acre)	
Porosity (ft ³ /ft ³)		Gas RF	
Area (acre)			
Oil RF			

API: American Petroleum Institute, B_o : oil formation volume factor (which depends on temperature and pressure), GOR: Gas Oil Ratio, MSCF/RB: *Thousands* Standard Cubic feet per Reservoir Barrel, STB: Stock Tank Barrel, S_w : water saturation, ϕ : porosity, MMSCF: *Million* Standard Cubic Feet, RF: recovery factor.

Missing value imputation is a common strategy employed to tackle incomplete datasets (Lin & Tsai 2020). An oversimplified imputation method is to replace missing values of a feature with its mean, which typically results in rough approximations. In this study, however, we first sorted all the samples based on their RF values ascendingly, then divided each feature into smaller subsamples (SS) of 10 entries and assured that the ratio (R) of the missing values (MS) to the total size of the sub-sample (TSS) was less than 10%. We next replaced the missing values with the mode of each subsample. The logic behind this imputation method is to better estimate the missing

values for one feature based on similarities in the RF values. The workflow of the proposed imputation method is summarized in Figure 2.2.

We used the Gaussian rank transformation to convert the distribution of each feature to approximately Gaussian. This method ranks each value of a feature based on its magnitude from lowest to highest (larger values of higher ranks). To remove bias and minimize uncertainties, one should have dimensionless input and output features with similar ranges and magnitude variations between 0 and 1 (Sola & Sevilla 1997). For this purpose, using the minimum and maximum values of each feature, the input and output data were scaled using the equation $X_{STD} = (x - x_{min}) / (x_{max} - x_{min})$ followed by $X_{scaled} = X_{STD} \times (max - min) + min$ (Pedregosa et al. 2011) in which min and max are, respectively, desired lower and upper limit values. In this study, the limits were chosen to be 0 (min) and 1 (max).

To construct the ML models, the data were randomly divided into two splits; 90% to train the model and 10% to test it. The 90%-10% split was selected so that the ML algorithms have as many samples as possible for training. Jaiswal et al. (2020) also used the same train and test split ratio in their study. For cross validation and optimizing the model parameters, the train dataset was used to create smaller datasets (called folds). All the preprocessing steps carried out on the test datasets and the independent databases were performed using the same parameters used for the train datasets (Kapoor & Narayanan 2021). This means that the imputation performed on the train and test datasets were performed separately using the mode of each feature specifically in each database. However, no imputation was applied to the independent databases. Instead, any sample with missing value was removed. This is because the goal of further evaluating the proposed ML models using the independent databases was to critically assess them based on real data not

imputed ones. As stated earlier, samples within the independent databases were not available in either the train or test datasets (repeated samples were removed to avoid data overlap).

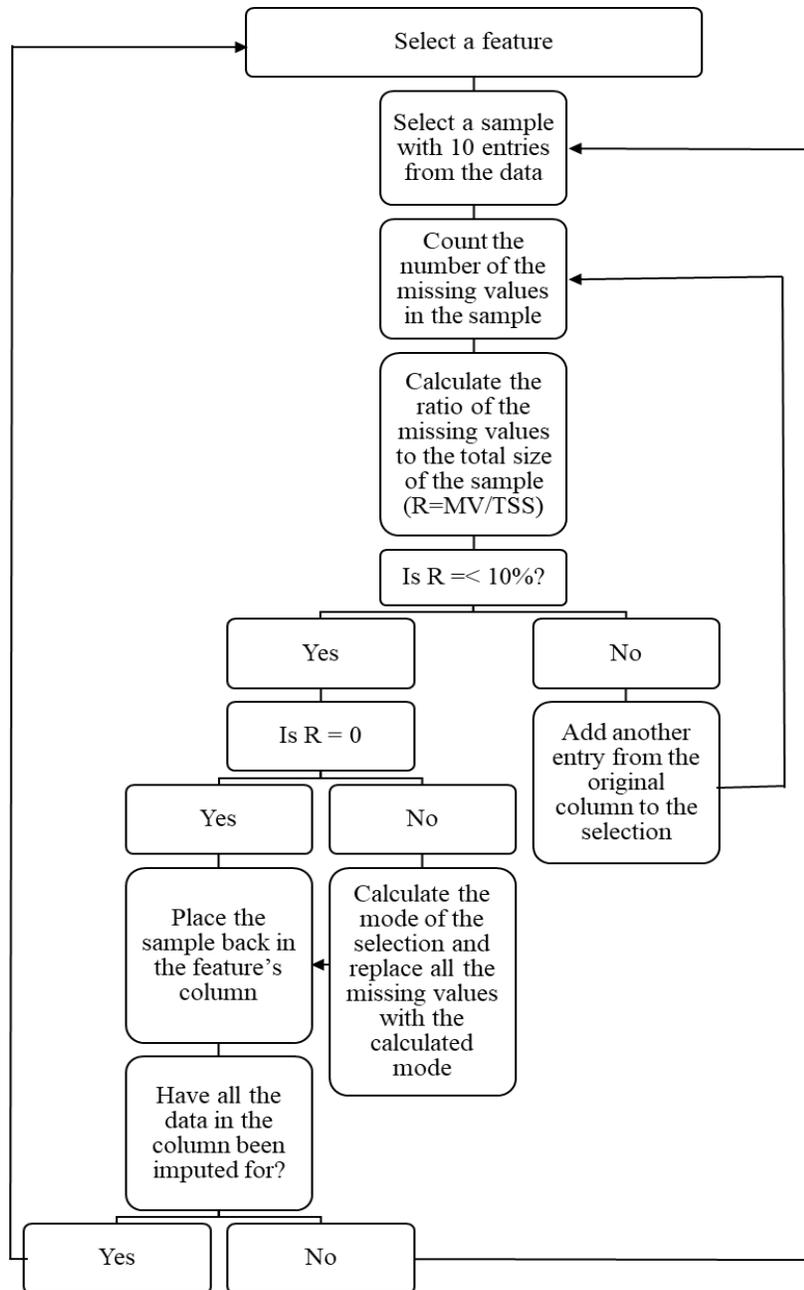


Figure 2.2. Imputation steps used to replace missing values in all the databases used to construct machine learning models.

2.3. Machine learning algorithms

In this study, we used the Python programming and applied extreme gradient boosting (XGBoost), support vector machine (SVM), and multiple linear regression (MLR) each of which is briefly described in the following. Although there are various ML algorithms, we selected these three because the XGBoost is among the latest ML algorithms developed. It is open access and has a user-friendly platform. The SVM is widely used in the petroleum industry for ML purposes (Tan et al. 2021; Chawshin et al. 2021; Olukoga & Feng 2021; Maiti et al. 2021; Xu et al. 2020). The MLR is simple and among the basic ML models. It is to compare with the other two more complicated ML algorithms.

2.3.1. XGBoost

The XGBoost is a decision tree based ensemble machine learning algorithm based on a gradient boosting (Chen et al. 2016). It is a supervised algorithm and can be used for both regression and classification tasks (Osman et al. 2021). The XGBoost is based on a gradient boosting framework that convert weak learners to strong ones in a sequential manner (Tang et al. 2021). In each iteration, gradient boosting machines (GBM) identify weak learners and try to empower them using higher weights and eventually convert them to strong learners, which improves the performance. Compared to the GBM, the XGBoost is an improvised version that yields more accurate results with less amount of computing resources. Some of those optimizations are parallelized implementation and L1 and L2 regularization to avoid overfitting (Chen et al. 2016).

2.3.2. Support Vector Machine

The SVM is a supervised algorithm that aims to construct a model to predict the category of unseen data using many different classes found in a set of training samples based on the generalized portrait algorithm (Zhou et al. 2016; Smola & Scholkopf 2004). In the support vector regression (Vapnik 1995), hyperplanes would be constructed, which acts as decision boundaries and make estimations of an output. Such hyperplanes are created through kernels, functions that convert input information into desired type for a support vector hyperplane. Hyperplanes are constrained by boundary lines that are epsilon (ϵ) units away around the hyperplane to construct a border among samples. The SVM regression is carried out by finding the best fit hyperplane that houses most number of data and are within the ϵ value rather than minimizing the error between measured and estimated values.

2.3.3. Forward selection stepwise multiple linear regression

The stepwise MLR is a simple and basic machine learning model in which those input variables significantly contributing to outputs are detected via the t-test and then used to establish a linear-based relationship (Smith 2018). In each iteration, a feature is added to the model until there is no statistically significant parameter left (forward stepwise). The performance of the model is optimized through this process of feature addition (Jahns 1966).

2.4. Learning curves

In the ML, over- and/or under-fitting may happen because of insufficient or excessive number of training samples. To avoid it, we investigated learning curves for each algorithm by constructing a basic model using the first sample in the train dataset. Predictions were then carried out for the same sample in the train dataset as well as the first sample in the test dataset. Next, a similar basic model was constructed based on the first two samples in the train dataset and used to

estimates the RF values for the first two samples in the train and test datasets. This process was iterated until all the samples in the train dataset were included in the basic model. The effect of adding samples was quantified by calculating the RMSE value. Figure 2.3 shows different schematic learning curves and over- or under-fitting interpretations. The closer the train and test learning curves are, the less the over-fitting or under-fitting would be. Note that the test learning curve far below and above the train learning curve, respectively, indicate substantial under- and over-fitting.

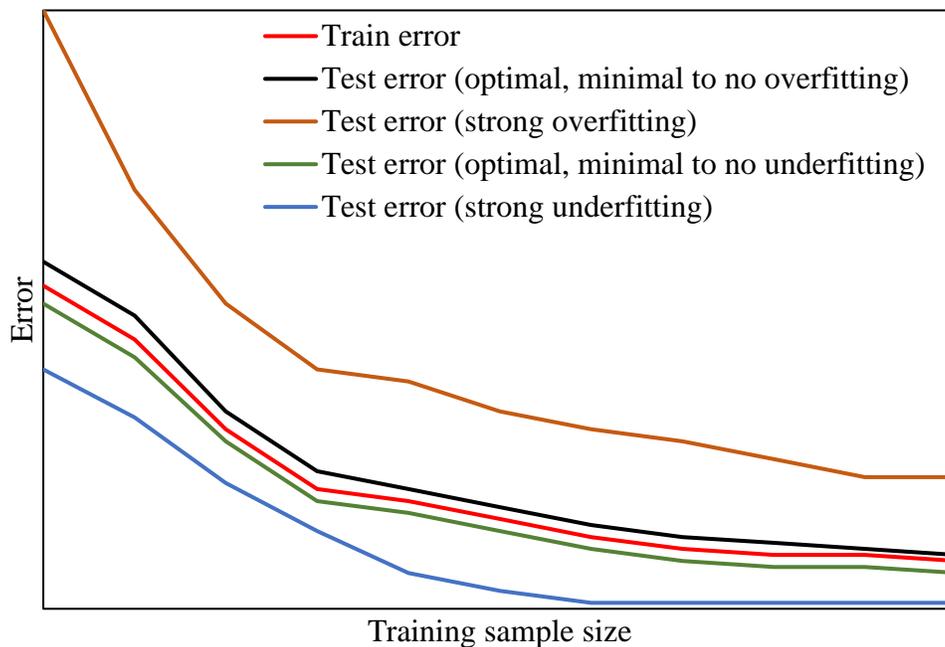


Figure 2.3. The scheme of different possible learning curves and their over/underfitting interpretations due to sample size.

2.5. Hyperparameter tuning

For the XGBoost, the hyperparameters tuning was implemented using an in-house code. At each step, two hyperparameters were optimized and cross validated using 10 folds. For the database TC, however, we used 3 folds due to the limited number of samples in this database. The hyperparameters tuning process was iterated until all the parameters were tuned.

For the SVM, the hyperparameters were optimized through a grid search using the 10-fold cross-validation pipeline except for the database TC for which we used 3 folds. Tables 2.2 and 2.3 present the name and value of the optimized parameters, respectively, for the XGBoost and SVM models and the CA database.

Table 2.2. Optimized hyperparameters of the XGBoost models developed for the oil and gas RFs obtained on the CA (Commercial combined with Atlas) database.

Parameters	Oil RF	Gas RF
Max depth	4	4
Minimum child weight	6	3
Learning rate	0.05	0.1
Subsample	0.9	0.9
Column sample by tree	1	1
Objective	reg:squarederror	reg:squarederror
Evaluation metric	rmse	rmse
Alpha	0.3	0.8
Lambda	0.04	0.08
Column sample by level	1	0.9
Gamma	0.01	0.01
Max delta step	0.1	0.1
Number of boosting rounds	999	999

Table 2.3. Optimized hyperparameters of the SVM models developed for the oil and gas RFs obtained on the CA (Commercial combined with Atlas) database.

Parameters	Oil RF	Gas RF
C	10	100
Epsilon	0.1	0.01
Gamma	1	1
Kernel	rbf	rbf

2.6. Feature importance

The feature importance was carried out using the SHapley Additive exPlanations (SHAP) for the XGBoost and SVM models. SHAP describes how input features contribute to the estimation of a target variable (Lundberg & Lee 2017). Based on the theory proposed by Shapley (1953), it determines the output of a game according to each player's contribution.

The feature importance for the MLR models was performed by investigating the regression coefficient of the input features (Male et al. 2020). This is because both the input and output variables were scaled between 0 and 1, and, thus, the greater the coefficient would be, the more important the feature is.

2.7. Workflow of the constructed models

Figure 2.4 shows the workflow including the data processing and model development. All models were constructed using the same procedure but different ML algorithms (i.e., XGBoost, SVM, and MLR) and combinations of databases. After each model was constructed, it was used to estimate the oil and/or gas RFs for the train and test datasets as well as the independent database. A total of 24 models were developed using the three algorithms. Each model was evaluated statistically using the RMSE, R^2 , and r values and visually using the scatterplots and the spread of data points around the 1:1 line.

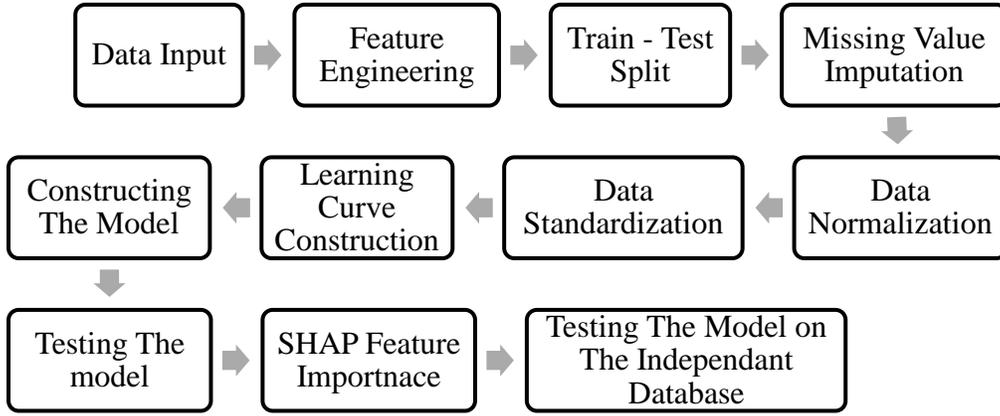


Figure 2.4. The workflow used here to develop the ML-based models and evaluate them. Normalization means converting the distribution of a feature to become approximately Gaussian, while standardization means scaling data between 0 and 1.

2.8. Model evaluation criteria

To evaluate the accuracy and reliability of the developed ML-based models, the root mean square error (RMSE), Pearson’s correlation coefficient (r), and coefficient of determination (R^2) were calculated as follows

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (x_e - x_m)^2}{N}} \quad (1)$$

$$r = \frac{Cov(x_e, x_m)}{\sigma_{x_m} \sigma_{x_e}} \quad (2)$$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (3)$$

where x_e is the estimated value, x_m is the measured value, N is the number of samples, $Cov(x_m, x_e)$ is the covariance of variables x_m and x_e , σ_{x_m} and σ_{x_e} are the standard deviations of measured and estimated values, respectively, SS_{res} is the residual sum of squares, and SS_{tot} is the total sum of squares. One should note that based on the definition of R^2 given in Eq. (3), its value could be positive or negative depending on the ratio SS_{res}/SS_{tot} (Edwards 1976).

Chapter 3 - Results

3.1. Learning curves

Figure 3.1 shows the learning curves for the XGBoost models developed for estimating the oil and gas RFs using the CA database. At lower training sample sizes (< 3000), the test learning curves are far above the training learning curves indicating over-fitting for both oil and gas RFs. However, at greater training sample sizes, both training and test learning curves converge to some constant value close to each other. Results presented in Figure 5 indicate that the size of the train dataset was large enough. Furthermore, the RMSE value of the train dataset near the RMSE value of the test dataset demonstrates that the constructed models were not over- or under-fitted when the entire train dataset was used. Similar results were obtained for the other algorithms and databases.

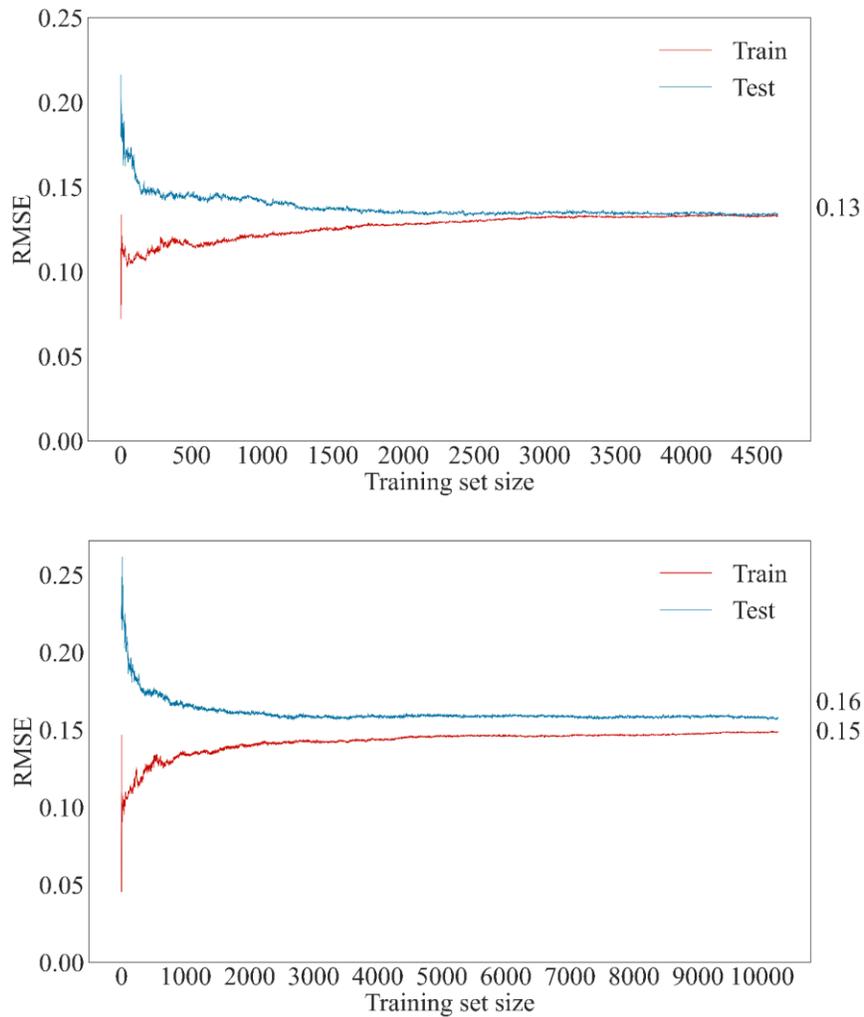


Figure 3.1. Learning curves of the XGBoost models developed for the (top) oil and (bottom) gas RFs using the CA (Commercial combined with Atlas) database.

3.2. Oil RF estimations

The RMSE, R^2 , and r values for the XGBoost, SVM, and MLR models and different combinations of the oil RF databases are summarized in Table 3.1. Comparing the RMSE, R^2 , and r values reported for the train datasets and different models shows that the XGBoost approach estimated the oil RF more accurately than the SVM and MLR models. Such values for the test datasets indicate that both the XGBoost and SVM models estimated the oil RF with similar accuracy but better than the MLR model. Results including the RMSE, R^2 , and r values for the

independent databases, however, demonstrate that the accuracy of all the three models was almost similar (Table 3.1). Recall that the TCA database contains all the data from the TORIS, Commercial, and Atlas databases. Therefore, the independent database results are not available for this combination. The negative R^2 values obtained for the independent databases indicate that the constructed models could not accurately estimate the RF for new data from a different database. $R^2 < 0$ means that the estimations are worse than those based on the average value of the measured data. From Eq. (3), one should expect $R^2 < 0$ when the SS_{res} value is greater than the SS_{tot} value. The small Pearson's correlation coefficient values ($r < 0.30$) also indicate that the estimated RFs were weakly correlated to the measured values in the independent databases.

Figure 3.2 shows the results of the oil RF estimated via the XGBoost method against the measured value for the train and test datasets as well as the independent databases. As can be seen, the estimations are around the 1:1 line for the train and test datasets. Generally speaking, the value of RMSE decreases from top to bottom; RMSE = 0.120 (train) and 0.134 (test) for the TC database, while 0.035 (train) and 0.111 (test) for the TCA database. The TCA database is the combination of three databases with the largest number of samples. Results imply that the performance of the XGBoost model increased as the number of samples in the train dataset increased. Figure 3.2 also shows the results for the independent databases and the combinations TC, TA, and CA. Substantial greater RMSE values reported for the independent databases compared to those for the train and test datasets presented in Table 3.1 and Figure 3.2 clearly demonstrate that the ML models depend on the databases used to construct them. This means although all the models provided reasonable estimates of the oil RF for the test datasets, they failed to accurately estimate the RF for the independent databases. We address the database-dependent accuracy and reliability of the proposed models in further detail in the Discussion section.

Table 3.1. Calculated RMSE, R² and r values for the three algorithms XGBoost, SVM, and MLR used to estimate the oil RF. The train datasets of TC, TA, CA, and TCA databases consisted of 1502, 4779, 4654, and 5461 number of samples, respectively.

Algorithm		XGBoost Regressor								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
TC		0.120	0.45	0.70	0.134	0.31	0.57	0.174	-0.14	0.17
TA		0.071	0.81	0.91	0.112	0.53	0.73	0.204	-0.65	0.18
CA		0.065	0.84	0.93	0.100	0.61	0.78	0.260	-1.50	-0.04
TCA		0.035	0.96	0.98	0.111	0.56	0.75	-	-	-
Algorithm		Support Vector Machine Regressor								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
TC		0.135	0.30	0.55	0.135	0.29	0.55	0.169	-0.08	0.29
TA		0.101	0.63	0.80	0.113	0.53	0.74	0.256	-1.62	0.09
CA		0.100	0.61	0.78	0.100	0.61	0.78	0.285	-1.97	0.02
TCA		0.110	0.57	0.75	0.130	0.41	0.65	-	-	-
Algorithm		Stepwise Multiple Linear Regression								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
TC		0.146	0.19	0.43	0.148	0.15	0.40	0.169	-0.08	0.30
TA		0.125	0.46	0.65	0.122	0.42	0.66	0.205	-0.68	0.19
CA		0.123	0.44	0.66	0.117	0.46	0.68	0.261	-1.50	0.02
TCA		0.129	0.38	0.62	0.134	0.37	0.61	-	-	-

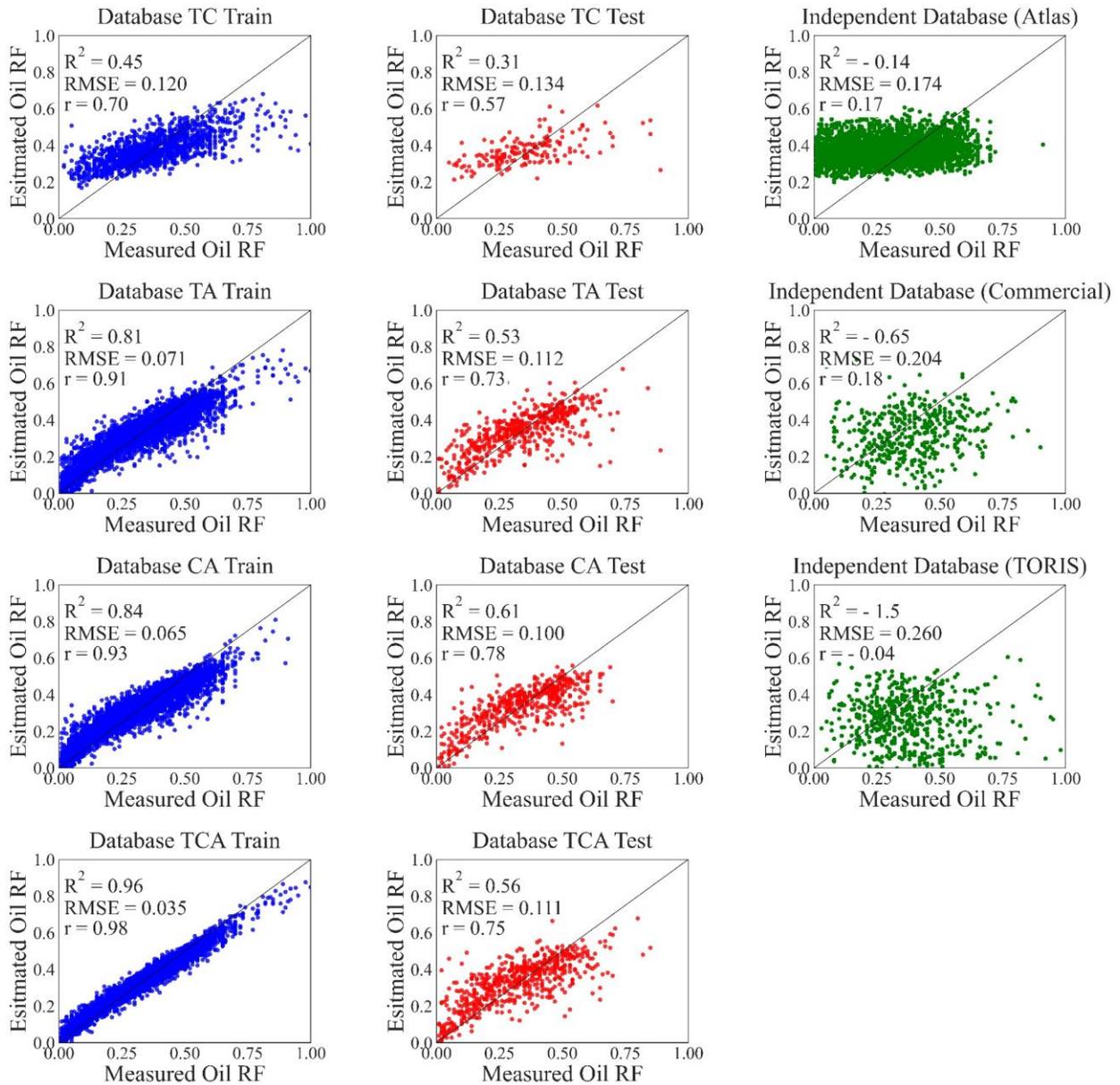


Figure 3.2. The estimated oil RF by the XGBoost algorithm versus that measured for the train and test datasets as well as the independent database. CA represents Commercial combined with Atlas, TC denotes TORIS combined with Commercial, TA is TORIS combined with Atlas.

For the combination TC shown in Figure 3.2, the independent database is Atlas for which the estimated oil RFs are mainly restricted between 0.2 and 0.6, while the measured values mostly vary from 0 to 0.75. The reason for such a trend in the oil RF estimations is not clear yet. This, however, could be because the constructed model using the TC database estimates the oil RF for the train and test datasets mainly in the range of 0.2 and 0.6 (Figure 3.2).

Generally speaking, the XGBoost algorithm tends to overestimate at the lower oil RFs, while underestimate at the higher RF values for the train and test datasets, as depicted in Figure 3.2. Similar results were reported by Talluru & Wu (2017) and Lee & Lake (2015). Such a trend may be attributed to the fact that the databases used in this study contained samples with similar oil RFs, while their input features were substantially different. Such a complexity in data would decrease the efficiency of ML algorithms to adequately find and learn patterns among input features and target variables. Table 3.2 lists two samples with different oil RFs but similar input values and two other samples with similar oil RFs but different input features.

Table 3.2. Two samples with similar input features but different oil RFs (top two) and two samples with different input features but similar oil RFs (bottom two).

API gravity	B_o (RB/STB)	GOR (MSCF/RB)	S_w (-)	T (°F)	P (psi)	Thickness (ft)	Reserves (STB)	k (mD)	ϕ (ft ³ /ft ³)	Area (acre)	Oil RF
35	1.5	0.75	0.56	212	3823	141.1	2.46×10^9	200	0.23	20501	0.59
37	1.1	0.7	0.25	226	4357	141	2.56×10^9	25	0.14	20501	0.15
30	1.3	0.54	0.33	171	2404	673	1.54×10^{10}	1	0.09	73885	0.3
42	2	0.95	0.35	281	5163	10	2.4×10^6	62	0.16	742	0.3

API: American Petroleum Institute, B_o : oil formation volume factor, GOR: gas oil ratio, S_w : water saturation, T: temperature, P: pressure, k: permeability, ϕ : porosity, RF: recovery factor.

3.3. Gas RF estimations

The RMSE, R^2 , and r values for different combinations of the databases used to estimate the gas RF were calculated and presented in Table 3.3. Comparing the RMSE, R^2 , and r values reported for the three ML algorithms indicates that, in general, the XGBoost model estimated the gas RF more accurately than the SVM and MLR models for the train datasets, similar to the oil results. For the XGBoost, we found that as the train dataset became larger, the RMSE decreased from 0.190 for the smallest database (i.e., GC) to 0.054 for the largest database (i.e., GCA). For the SVM and MLR, however, the RMSE value for the GCA database was greater than those for the GA and CA databases.

For the test datasets, we found that the accuracy of both XGBoost and SVM models were similar. Although the SVM model estimated the gas RF more accurately than the XGBoost for the GC and GA databases, the XGBoost outperformed the SVM model for the GCA database. Both models provided similar results for the CA database. We also found that the MLR model estimations were not as accurate as the XGBoost and SVM predictions.

Similar to the results of oil RF, the performance of all the algorithms for the independent databases was not satisfactory. As listed in Table 3.3, the RMSE values for the independent databases are substantially greater than those for the train and test datasets. Recall that the database GCA was the combination of the GasIS, Commercial, and Atlas databases, and, thus, there was no independent database for further evaluations. Based on the RMSE values reported in Table 3.3, the XGBoost model generally estimated the gas RF for the independent databases more accurately than the SVM and MLR models. Only for the GC database, the MLR model had a slightly better performance than the XGBoost and SVM models. In the assessment of the ML models using the

independent databases, we also found $R^2 < 0$, similar to the oil RF results (Table 4), which indicate unsatisfactory estimations, confirmed by small r values.

The substantially smaller RMSE, R^2 , and r values for the independent databases compared to those for the train and test datasets demonstrate the database dependence of the ML performances. We address this in further detail in the Discussion section.

Table 3.3. Calculated RMSE, R² and r values for the three algorithms XGBoost, SVM, and MLR used to estimate the gas RF. The train datasets of GC, GA, CA, and GCA databases consisted of 5838, 10040, 10251, and 15878 number of samples, respectively.

Algorithm		XGBoost Regressor								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
GC		0.190	0.36	0.62	0.217	0.18	0.43	0.331	-1.30	0.20
GA		0.057	0.93	0.97	0.093	0.82	0.90	0.694	-18.70	-0.1
CA		0.064	0.92	0.96	0.094	0.82	0.91	0.553	-5.60	-0.05
GCA		0.054	0.95	0.98	0.158	0.58	0.76	-	-	-
Algorithm		Support Vector Machine Regressor								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
GC		0.221	0.19	0.44	0.212	0.14	0.38	0.356	-1.70	0.12
GA		0.079	0.87	0.93	0.079	0.86	0.93	0.825	-26.90	0.03
CA		0.083	0.86	0.93	0.094	0.82	0.91	1.342	-37.70	0.12
GCA		0.160	0.56	0.75	0.165	0.54	0.74	-	-	-
Algorithm		Stepwise Multiple Linear Regression								
Dataset		Train			Test			Independent		
Database		RMSE	R ²	r	RMSE	R ²	r	RMSE	R ²	r
GC		0.220	0.14	0.37	0.228	0.10	0.32	0.316	-1.10	0.34
GA		0.128	0.66	0.81	0.130	0.64	0.80	1.070	-45.90	-0.02
CA		0.137	0.61	0.78	0.132	0.64	0.80	0.862	-14.97	-0.15
GCA		0.191	0.38	0.61	0.184	0.43	0.66	-	-	-

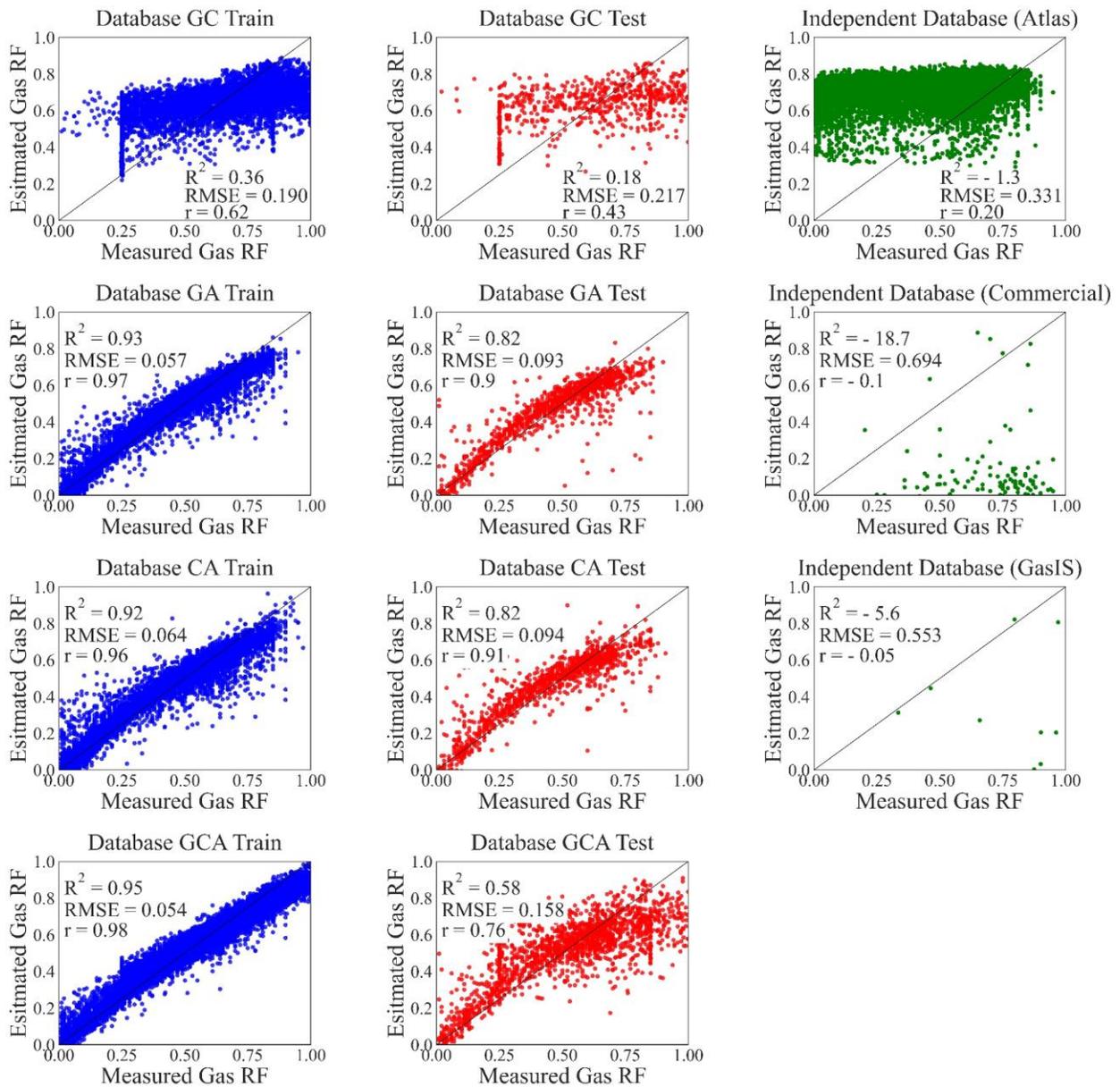


Figure 3.3. The estimated oil RF by the XGBoost algorithm versus that measured for the train and test datasets as well as the independent database. GC denotes GasIS combined with Commercial, GA is GasIS combined with Atlas, CA represents Commercial combined with Atlas

As shown in Figure 3.3, we found that the constructed XGBoost models tend to overestimate the gas RF at smaller RF values, while underestimate at higher RF values regardless of the data combination set. As stated earlier, this could be explained by the fact that some samples in the train dataset had similar input features, while their gas RFs were different or vice versa. Table 3.4 lists two samples with similar input features but different gas RF values as well as two other samples with different input features but similar gas RFs.

Table 3.4. Two samples with similar features but different gas RF (top two) and two samples with different features but similar gas RF (bottom two).

GOR (MSCF/RB)	S_w (-)	T (°F)	P (psi)	Thickness (ft)	Reserves (MMSCF)	k (mD)	ϕ (ft ³ /ft ³)	Area (acres)	Gas RF
0	0.10	160	3486	75	1×10^6	2	0.08	25946	0.74
0	0.20	223	4394	69	7.1×10^5	2	0.05	25946	0.86
0	0.30	230	5318	1299	1.41×10^9	70	0.15	1150027	0.78
0	0.08	268	1967	299	2.12×10^6	200	0.23	11120	0.74

GOR: gas oil ratio, S_w : water saturation, T: temperature, P: pressure, k: permeability, ϕ : porosity, RF: recovery factor.

3.4. Feature importance

To investigate the significance of input variables, we performed the feature importance analysis. For the XGBoost, we used SHAP that determines which features have more impact on the constructed model and how the magnitude of features affect model predictions. Figure 3.4 shows the results of SHAP for the oil and gas RF and the CA (Commercial combined with Atlas) database. Interestingly, the constructed models used influential features in petroleum industry (e.g., porosity, B_o , and GOR) to estimate the RF. They also shed light on features traditionally not considered to be influential, such as permeability and temperature. Comparing the oil and gas RF graphs shown in Figure 3.4 implies the similarities in the feature importance selection, which further demonstrates how the pattern recognition in the oil and gas models were similar indicating

both models used similar protocols to select important features. This means the workflow used to construct the ML models is logical with no data leakage (Kapoor & Narayanan 2021). Similar feature importance results were obtained for other combinations of databases. For the oil RF and the XGBoost algorithm, the top 4 features selected in the databases TA, CA, and TCA were similar. The feature importance, however, was different in the TC database, which could be due to its limited number of samples. For the gas RF, the top 3 features (i.e., reserves, area, and thickness) were similar in the databases GA, CA, and TCA. Similar to the oil RF results, the GC database had a different feature importance selection. This may be because of the small size of this database compared to the other ones.

Results of SHAP showed that the SVM prioritized features very similar to the XGBoost. The top 4 features for oil (i.e., reserves, area, thickness, and GOR) and the top 3 features for gas (i.e., reserves, area, and thickness) were exactly the same variables selected by SHAP and the XGBoost model. This further supports the workflow proposed to construct the ML models.

For the MLR, we investigated the feature importance based on the coefficient of input features, the greater the coefficient the higher the impact on the model. Results showed that the variable reserves was the top feature selected by the MLR algorithm for both oil and gas RFs models except for the TC and GC databases. We found that each MLR model had a different order of feature importance. However, temperature and GOR for gas and temperature and API oil gravity for oil were the commonly-used influential features.

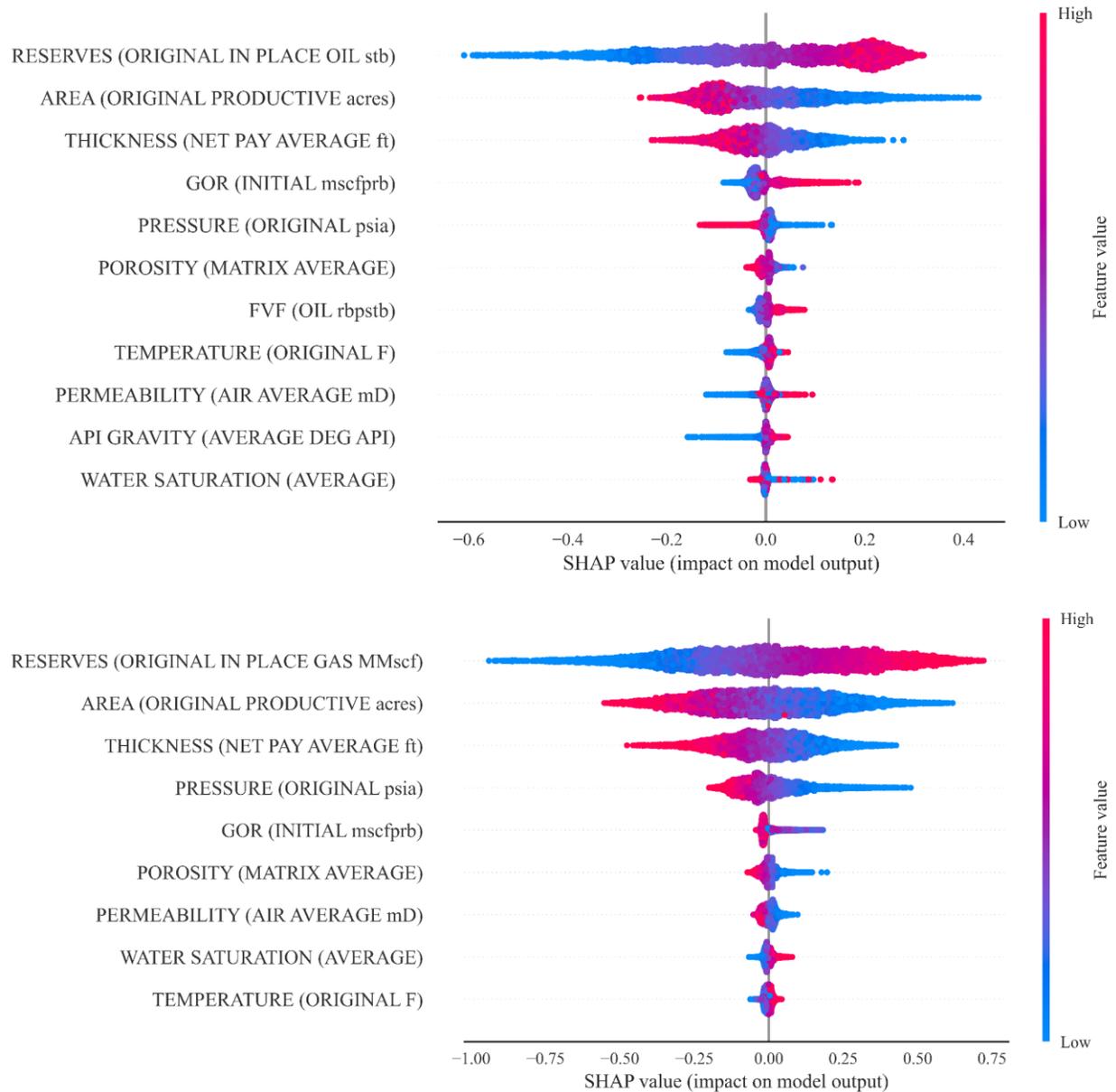


Figure 3.4. Feature importance of the XGBoost model in the estimation of oil RF (top) and gas RF (bottom) models for the databases CA (with oil RF as target variable) and CA (with gas RF as target variable). Since the data were scaled between 0 and 1, the blue color denotes the smaller values close to 0, while the red color represents greater values near 1. The positive SHAP values indicate positive influence on the model construction and more accurate prediction. However, the negative SHAP values show the negative impact on the model.

Chapter 4 - Discussion

As stated earlier, the RF is a function of reservoir's displacement (also known as reservoir drive mechanism or natural drive). Reservoir drive mechanisms are natural forces within a reservoir that move hydrocarbons through it (Schlumberger 2020b). In the order of efficiency, reservoir drive mechanisms are water drive (bottom or edge), gas cap, solution gas, gravity drainage, and rock or compaction drive. A reservoir may have one or a combination of several drive mechanisms. Drive mechanisms are themselves functions of reservoir qualities, which are capacities of a reservoir to store and transmit oil or gas. The quality of a reservoir is controlled by pore throat size distribution and pore geometry (including natural fractures), pore volume, permeabilities to hydrocarbon, water saturation (hydrocarbon pore volume), lateral continuity, number, position of flow units and containers, reservoir pressure, and drive mechanism (Beaumont et al. 2019). Accordingly, an efficient and reliable ML model should identify reservoir drive mechanism attributes (e.g., pressure and temperature) as well as reservoir quality factors (e.g., porosity and permeability) as the most important features (as shown in Figure 8) to accurately estimate the RF. We should note that the reservoir drive mechanisms were not available for all the databases in this study. However, features attributing to the drive mechanisms and reservoir quality e.g., porosity, thickness, permeability, and water saturation were available.

ML models are developed mainly for prediction purposes. Therefore, they are expected to provide reasonable estimations, if evaluated using a new database including data not used to train and test such models. In the literature, most ML models were not assessed via an independent database. Schaap & Leij (1998) are among the pioneers who addressed the database dependence of ML models and their results. Those authors applied neural networks to train and test models for estimating capillary pressure curve and permeability in soils. Schaap & Leij (1998) found that their

ML models were optimal for databases based on which models were trained. However, the accuracy and reliability of ML models dropped down when evaluated using other independent databases. Schaap & Leij (1998) concluded that the performance of ML models may substantially depend on data used to train and evaluate them. The train-test split is a process through which one would expose a model to unseen data and evaluate the performance of the model. The problem with that is train and test data are still derived from the same database. No matter how randomly a database is split, train and test data would still be similar to some extent.

ML models produce reliable results, if and only if, test and train data are similar both experimentally (e.g., measurement methods and timing of the measurements) and statistically (feature distributions). One should expect the reliability of estimations for test data to drop significantly, if important features were measured at different stages of exploration or even production rather than one single stage. For instance, the feature pressure may vary over time at different stages of a field life. Accordingly, one should consistently use feature values measured at similar life stages of a field using similar methods across all reservoirs. Otherwise, such differences would impact the accuracy of an ML model. For example, for the databases used in this study, it is not clear whether porosity was measured on core samples or determined using sonic or density logs across all the databases. Such differences in measurements may substantially affect the ML results.

As can be deduced from Tables 4 and 6, the performance of the constructed ML models, were unsatisfactory for the independent datasets. Using the t-test method, we statistically analyzed the similarity in the data. More specifically, we compared the features in the train datasets with those in the test datasets as well as those in the independent databases. Results presented for the oil and gas RFs in Table 4.1 showed no significant differences between the train and test datasets

at the 0.05 significance level ($p\text{-value} > 0.05$). However, we found $p\text{-values} \ll 0.05$ when the oil and gas RFs in the train datasets were compared to those in the independent databases (Table 4.1). This means although the train and test datasets statistically showed similar RF values, the train datasets had significantly different RFs than the independent databases for both oil and gas. Similar results were obtained for the input features.

Table 4.1. p-values determined using the t-test by comparing the RF values in the train datasets with those in the test datasets as well as those in the independent databases for both oil and gas at the significance level of 0.05 (confidence level of 0.95). The p-values were calculated using the paired t-test and unequal variances.

	Database	p-value train and test RFs	Similar data?	p-value train and independent RFs	Similar data?
Oil	TC	0.18	Yes	2.65×10^{-33}	No
	TA	0.31	Yes	2.82×10^{-13}	No
	CA	0.42	Yes	1.18×10^{-12}	No
	TCA	0.12	Yes	-	-
Gas	GC	0.27	Yes	0	No
	GA	0.51	Yes	1.00×10^{-45}	No
	CA	0.65	Yes	3.95×10^{-3}	No
	GCA	0.87	Yes	-	-

The similarities in the train and test data (Table 4.1) might be because they come from the same combined database. However, the data in the independent databases are not necessarily similar to those in the train datasets. The dissimilarities in the train and independent data (Table 4.1) are probably the main reason why performances of the ML models evaluated using the independent datasets were unsatisfactory.

In addition to similarity in the data, we also investigated the similarity in the distributions of top three important input features (i.e., reserves, area, and thickness) as well as the RF. More

specifically, using the two-sample Kolmogorov-Smirnov test we determined whether two samples are from the same distribution at the significance level of 0.05. Table 4.2 presents the results for both oil and gas and the CA database. The statistic values less than the critical value (D) and p-values greater than the significance level indicate that features in the train and test datasets come from the same distribution for both oil and gas. However, we found the statistic values greater than the critical value (D) and p-values substantially less than the significance level for the features in the train datasets and independent databases meaning that they should be from different distributions. Similar results were obtained for other database combinations.

Table 4.2. Results of the Kolmogorov-Smirnov test for the top three input features and RFs in the CA database and the corresponding independent databases TORIS (for oil) and GasIS (for gas). The significance level is 0.05.

Database	Feature	D train and test datasets	Statistic train and test datasets	p-value train and test datasets	Samples from same distributions ?	D train dataset and independent database	Statistic train dataset and independent database	p-value train dataset and independent database	Samples from same distributions ?
CA (oil RF)	RF	0.063	0.001	1.00	Yes	0.060	0.90	0.0	No
	Reserves		0.024	0.95	Yes		0.81	0.0	No
	Area		0.027	0.88	Yes		0.47	4.59×10^{-104}	No
	Thickness		0.039	0.48	Yes		0.32	2.04×10^{-46}	No
CA (gas RF)	RF	0.042	0.001	1.00	Yes	0.017	0.92	3.33×10^{-10}	No
	Reserves		0.028	0.40	Yes		0.55	0.0	No
	Area		0.016	0.96	Yes		0.38	0.0	No
	Thickness		0.035	0.15	Yes		0.25	0.0	No

D: Critical Value for the Two-sample Kolmogorov-Smirnov test (2-sided)

We recommend that before one uses the constructed ML models to estimate the oil or gas RF, statistical tests should be conducted to investigate whether the distributions of input features in our train datasets are statistically similar to those in a new database for which the oil or gas RF estimations are desired. If the RF estimation is required for one sample (or reservoir), then the values of important features should be close to the values of means in the train dataset studied here.

Chapter 5 - Conclusion

In this study, we used a total of 4 databases (Commercial, TORIS, Atlas, and GasIS) to construct machine learning models for the estimation of oil and gas recovery factors at the reservoir scale. For this purpose, we applied different combinations of the databases and three algorithms i.e., XGBoost, SVM, and MLR. Generally speaking, the XGBoost model estimated both oil and gas RFs more accurately than the SVM and MLR models for the train and test datasets. However, the performances of the three models were unsatisfactory when evaluated using the independent databases. Results showed that the constructed ML models and their performances depend on the databases used to train them. The statistical tests disclosed that the unsatisfactory accuracies of the ML models were because the input and output features in the train datasets were significantly different from those in the independent databases. We recommend evaluating similarities in the input features and their distributions before applying the constructed ML models to estimate the oil or gas RF.

References

- Aliyuda, K., Howell, J., & Humphrey, E. (2020). Impact of geological variables in controlling oil-reservoir performance: An insight from a machine-learning technique. *SPE Reservoir Evaluation & Engineering*, 23, 1314–1327.
- Aliyuda, Kachalla, & Howell, J. (2019). Machine-learning algorithm for estimating oil-recovery factor using a combination of engineering and stratigraphic dependent parameters. *Interpretation*, 7(3), SE151–SE159. <https://doi.org/10.1190/int-2018-0211.1>
- Alpak, F. O., Araya–Polo, M., & Onyeagoro, K. (2019). Simplified Dynamic Modeling of Faulted Turbidite Reservoirs: A Deep-Learning Approach to Recovery-Factor Forecasting for Exploration. *SPE Reservoir Evaluation & Engineering*, 22, 1240–1255.
- Beaumont, E. A., Foster, N. H., & Hartmann, D. J. (2019). Predicting Reservoir System Quality and Performance. *Exploring for Oil and Gas Traps*. <https://doi.org/10.1306/trhbk624c9>
- Burgess, Grant L.; Cross, Kellie K.; Kazanis, E. G. (2019). *OCS Estimated Oil and Gas Reserves Gulf of Mexico*. <https://www.data.boem.gov/Main/GandG.aspx>
- Chawshin, K., Gonzalez, A., Berg, C. F., Varagnolo, D., Heidari, Z., & Lopez, O. (2021). Classifying lithofacies from textural features in whole core CT-scan images. *SPE Reservoir Evaluation and Engineering*, 24(2), 341–357. <https://doi.org/10.2118/205354-PA>
- Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 13-17-Aug*, 785–794. <https://doi.org/10.1145/2939672.2939785>
- Chen, Y., Zhu, Z., Lu, Y., Hu, C., Gao, F., Li, W., Sun, N., & Feng, T. (2019). Reservoir

- recovery estimation using data analytics and neural network based analogue study. *Society of Petroleum Engineers - SPE/IATMI Asia Pacific Oil and Gas Conference and Exhibition 2019, APOG 2019*. <https://doi.org/10.2118/196487-ms>
- Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. *Multiple Classifier Systems*, 1–15.
- Edwards, A. L. (1976). *An introduction to linear regression and correlation*. W. H. Freeman.
- Hugman, R. H., Vidas, E. H., & Springer, P. S. (2000). *Development of the DOE Gas Information System (GASIS) – Final Report*. https://netl.doe.gov/sites/default/files/2018-04/Final_28139.pdf
- Ibrahim Ahmed Osman, A., Najah Ahmed, A., Chow, M. F., Feng Huang, Y., & El-Shafie, A. (2021). Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia. *Ain Shams Engineering Journal*, 12(2), 1545–1556. <https://doi.org/10.1016/j.asej.2020.11.011>
- Jahns, H. O. (1966). A Rapid Method for Obtaining a Two-Dimensional Reservoir Description From Well Pressure Response Data. *Society of Petroleum Engineers Journal*, 6(04), 315–327. <https://doi.org/10.2118/1473-pa>
- Jaiswal, V., Brown, A. A., & Yu, M. (2020). Virtual sensors for mooring line tension monitoring. *Proceedings of the Annual Offshore Technology Conference, 2020-May*. <https://doi.org/10.4043/30562-ms>
- Kapoor, S., & Narayanan, A. (2021). *(Ir)Reproducible Machine Learning : A Case Study*. <https://reproducible.cs.princeton.edu/>

- Lee, B. B., & Lake, L. W. (2015). Using data analytics to analyze reservoir databases. *Proceedings - SPE Annual Technical Conference and Exhibition, 2015-Janua*(September), 2481–2491. <https://doi.org/10.2118/174900-ms>
- Lin, W. C., & Tsai, C. F. (2020). Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, 53(2), 1487–1509. <https://doi.org/10.1007/s10462-019-09709-4>
- Ling, K., Wu, X., Zhang, H., & He, J. (2013). Tactics and pitfalls in production decline curve analysis. *SPE Production and Operations Symposium, Proceedings, 1*, 398–413. <https://doi.org/10.2118/164503-ms>
- Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems, 2017-Decem*(Section 2), 4766–4775.
- Maiti, S., Gupta, H., Vyas, A., & Kulkarni, S. D. (2021). Evaluating Precision of Annular Pressure Buildup (APB) Estimation Using Machine-Learning Tools. *SPE Drilling & Completion, October 2019*, 1–11. <https://doi.org/10.2118/196179-pa>
- Male, F., Jensen, J. L., & Lake, L. W. (2020). Comparison of permeability predictions on cemented sandstones with physics-based and machine learning approaches. *Journal of Natural Gas Science and Engineering*, 77(March), 103244. <https://doi.org/10.1016/j.jngse.2020.103244>
- Maselugbo, A. O., Onolemhemen, R. U., Denloye, A. O., Salufu, S. O., & Isehunwa, S. O. (2017). Optimization of Gas Recovery using Co-production technique in Water Drive Reservoir. *Journal of Petroleum and Gas Engineering*, 8(6), 42–48. <https://doi.org/10.5897/jpge2017.0269>

- Muskat, M., & Woods, R. W. (1944). An Analysis of Material-balance Calculations. *Trans. 160*, May, 124–139. <https://doi.org/10.2118/945124-G>
- Olukoga, T. A., & Feng, Y. (2021). *Practical Machine-Learning Applications in Well-Drilling Operations*. December, 849–867.
- Omoniyi, O. A., & Adeolu, S. (2014). Decline Curve Analysis and Material Balance, as Methods for Estimating Reserves (A Case Study of D4 and E1 Fields). *International Journal of Innovative Research and Development*, 3(11), 207–218.
http://internationaljournalcorner.com/index.php/ijird_ojs/article/view/135465
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825–2830.
<http://jmlr.org/papers/v12/pedregosa11a.html>
- Schaap, M. G., & Leij, F. J. (1998). Database-related accuracy and uncertainty of pedotransfer functions. *Soil Science*, 163(10), 765–779. <https://doi.org/10.1097/00010694-199810000-00001>
- Schlumberger. (2020a). Recovery Factor. In *Schlumberger Oilfield Glossary*. Schlumberger. https://www.glossary.oilfield.slb.com/en/Terms/r/recovery_factor.aspx#:~:text=The recoverable amount of hydrocarbon,to increase the recovery factor.
- Schlumberger. (2020b). Reservoir-Drive Mechanisms. In *Schlumberger Oilfield Glossary*. Schlumberger. https://www.glossary.oilfield.slb.com/en/Terms/r/reservoir-drive_mechanisms.aspx#:~:text=Natural forces in the reservoir,combination drive%2C and

gravity drainage.

Shapley, L. S. (1953). A Value for n-person Games. Contributions to the Theory of Games.

Annals of Mathematics Studies, 28(2), 307–317.

Sharma, A., Srinivasan, S., & Lake, L. W. (2010). Classification of oil and gas reservoirs based

on recovery factor: A data-mining approach. *Proceedings - SPE Annual Technical*

Conference and Exhibition, 1, 50–70. <https://doi.org/10.2118/130257-ms>

Smith, G. (2018). Step away from stepwise. *Journal of Big Data*, 5(1).

<https://doi.org/10.1186/s40537-018-0143-6>

Sola, J., & Sevilla, J. (1997). Importance of input data normalization for the application of neural

networks to complex industrial problems. *IEEE Transactions on Nuclear Science*, 44(3

PART 3), 1464–1468. <https://doi.org/10.1109/23.589532>

Talluru, G., & Wu, X. (2017). Using data analytics on dimensionless numbers to predict the

ultimate recovery factors for different drive mechanisms of Gulf of Mexico oil fields.

Proceedings - SPE Annual Technical Conference and Exhibition, 0.

<https://doi.org/10.2118/187269-ms>

Tan, Y., Tian, H., Xu, F., Jiang, D., Jiang, R., Lin, Y., & Zhang, J. (2021). Complex Decay

Prediction of Marine Machinery Using Multilabel SVM. *Journal of Ship Research*, 00(00),

1–10. <https://doi.org/10.5957/josr.10200052>

Tang, J., Fan, B., Xiao, L., Tian, S., Zhang, F., Zhang, L., & Weitz, D. (2021). A new ensemble

machine-learning framework for searching sweet spots in shale reservoirs. *SPE Journal*,

26(1), 482–497. <https://doi.org/10.2118/204224-PA>

Tewari, S., Dwivedi, U. D., & Shiblee, M. (2019). Assessment of big data analytics based ensemble estimator module for the real-time prediction of reservoir recovery factor. *SPE Middle East Oil and Gas Show and Conference, MEOS, Proceedings, 2019-March*.
<https://doi.org/10.2118/194996-ms>

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.

Xu, R., Deng, T., Jiang, J., Jobe, D., & Xu, C. (2020). Integration of NMR and conventional logs for vuggy facies classification in the arbuckle formation: A machine learning approach. *SPE Reservoir Evaluation and Engineering*, 23(3), 917–929. <https://doi.org/10.2118/201102-PA>

Zhou, X., Zhang, X., & Wang, B. (2016). Support vector machine: A survey. *Advances in Intelligent Systems and Computing*, 382(8), 269–278. https://doi.org/10.1007/978-3-662-47926-1_26