Data-Centric Engineering (0000), 1–23
doi:10.1017/xxxx

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Machine Learning Approaches for Prediction of Serious Fluid Leakage from Hydrocarbon Wells

Mehdi Rezvandehy[1]* and Bernhard Mayer[1]

[1]University of Calgary, Department of Geoscience, 2500 University Drive NW, Alberta, T2N 1N4, Canada
*Corresponding author. E-mail: mehdi.rezvandehy@ucalgary.ca

Received xx xxx xxxx

Keywords: Energy Wells; Imputation; Probability Estimation; Imbalanced Class Classification; Resampling

Abstract

The exploitation of hydrocarbon reservoirs may potentially lead to contamination of soils, shallow water resources and greenhouse gas emissions. Fluids such as methane or CO2 may in some cases migrate towards the groundwater zone and atmosphere through and along hydrocarbon wells. Field tests in hydrocarbon producing regions are routinely conducted for detecting serious leakage to prevent environmental pollution. The challenge is that testing is costly, time-consuming, and sometimes labor-intensive. In this study, machine learning approaches were applied to predict serious leakage with uncertainty quantification for wells that have not been field tested in Alberta, Canada. An improved imputation technique was developed by Cholesky factorization of the covariance matrix between features, where missing data are imputed via conditioning of available values. The uncertainty in imputed values was quantified and incorporated into the final prediction to improve decision making. Next, a wide range of predictive algorithms and various performance metrics were considered to achieve the most reliable classifier. However, a highly skewed distribution of field tests towards the negative class (non-serious leakage) forces predictive models to unrealistically underestimate the minority class (serious leakage). To address this issue, a combination of oversampling and undersampling was applied. By investigating all the models on never-before-seen data, an optimum classifier with minimal false negative prediction was determined. The developed methodology can be applied to identify the wells with the highest likelihood for serious fluid leakage within producing fields. This information is of key importance for optimizing field test operations to achieve economic and environmental benefits.

Impact Statement

Field test operations to detect methane and CO2 leakages from hydrocarbon wells are expensive and time-consuming. Most wells do not have leakage or are labeled as non-serious leakage which repair is not required until abandonment. However, serious leakages are critical and should be identified and prioritized for immediate amendment to prevent environmental pollution. In this work, a reliable predictive model was trained by correlating the result of historical field tests with the well properties that may influence the likelihood of leakage such as age, depth, production/injection history, well density, and deviation, among others. The trained model can reliably predict the likelihood of serious leakage for wells that have not been field tested. Those wells with the highest probability of serious leakage can be prioritized for field test operation. This leads to develop cost-effective field testing and thereafter environmental benefits.

1. Introduction

Exploitation of oil and gas reservoirs has raised public concerns regarding potential contamination of soils, shallow ground water and increases in greenhouse gas emissions (Cherry et al., 2014; Shindell et al., 2009; Brandt et al., 2014). In cases where hydrocarbon wells are not properly sealed, there is the potential that gas such as methane and/or sequestered CO2 can migrate outside the surface casing of wellbores towards shallow aquifers, soils and the atmosphere, or travel inside the wellbore to be vented to the atmosphere via surface casing vent flows (SCVF) causing environmental concerns. Monitoring for such gas migration and surface casing vent flows is required by regulators to detect serious leakage associated with existing oil and gas wells, and to prioritize the amendment of the leakiest wells (Montague et al., 2018; Watson and Bachu, 2009; Abboud et al., 2021).

The Alberta Energy Regulator (AER) in Alberta, Canada, operates such field tests for energy wells within the province. The AER applies two field tests for identification of fluid migration after a well is completed to produce hydrocarbon or to inject any fluid: 1) Surface casing vent flow (SCVF) is the flow of gas (methane, CO2 etc.) out of the casing annulus or surface casing. SCVF is often referred to as internal migration. 2) Gas Migration (GM) is a flow of any gas that is detectable at surface outside of the outermost casing string. GM is often referred to as seepage or external migration (Alberta Energy Regulator, 2003).

The test for SCVF can be a simple bubble test where a small hose attached to the surfacecasing vent is immersed into a container filled with water. Observing bubbles indicates the well has SCVF and further testing is required to measure pressure and flow rate. Wells with positive SCVF are considered serious in the province of Alberta under one or several of the following conditions: a) gas-flow rates higher than 300 m^3/d, b) stabilized pressure > 9.8 kPa/m, c) saline water, d) liquid-hydrocarbons, and e) hydrogen sulphide (H2S) flow (Alberta Energy Regulator, 2003).

Monitoring for GM in the vicinity of hydrocarbon wells is generally more challenging and often approached with measurements using gas detectors near the soil surface. GM in soils happens when gas migrates outside of the cemented surface casing. The gas source is frequently natural gas from geological formations in the intermediate zone (e.g., below the base of groundwater protection but above the targeted hydrocarbon reservoir) with migration often facilitated by poor surface-casing cement. Monitoring for GM is traditionally accomplished by inserting gas measurement probes into the soil to a depth of at least 50 cm in a test pattern radiating out from the wellbore, and subsequently gas concentrations and flow rates are measured. A GM is serious if there is a high flow rate or public safety hazard or off-lease environmental damage, such as groundwater contamination (see Alberta Energy Regulator (2003) for more information). Wells with positive SCVF/GM are classified as non-serious if none of the conditions for serious category are met. In Alberta, repair for serious SCVF/GM leakage is required within 90 days; otherwise, repair is deferred to abandonment (Alberta Energy Regulator, 2003; Kang et al., 2014; Watson and Bachu, 2009; Montague et al., 2018).

Efficient and cost-effective testing of all hydrocarbon-producing wells is a major challenge in areas with large numbers of producing or injection wells. The AER requires testing for all wells only within a small specific area in central and eastern Alberta and only for wells completed since 1995 (Montague et al., 2018; Abboud et al., 2021). There are many wells in other parts of Alberta including abandoned and orphaned wells for which no SCVF/GM test have been conducted. Montague et al. (2018) (Montague et al., 2018) applied predictive models (machine learning) based on known well properties to generate a binary result for gas migration: whether the well is positive for SCVF/GM test or not. The wells only within the small test region (central and eastern Alberta) was included in the study and no predictions were made regarding the seriousness of fluid migration. Most leakages are non-serious (Alberta Energy Regulator, 2022) and repair is not required until abandonment; in contrast, serious leakages are critical and should be identified and prioritized for amendment to prevent environmental pollution. The objective of this work was to reliably train a model to predict the probability of serious

fluid leakage using oil and gas wells for which SCFV and GM measurements are available (Fig1) and subsequently make predictions for wells for which SCVF/GM tests have not been conducted.

2. Data Preparation

Fig1 shows a location map of classification for SCVF/GM test results obtained by AER (Alberta Energy Regulator, 2022) between January 1984 and November 2021 within Alberta, Canada. Based on 27404 tests that were conducted, 83.8% of the wells were classified as non-serious, while 16.2% of the tested

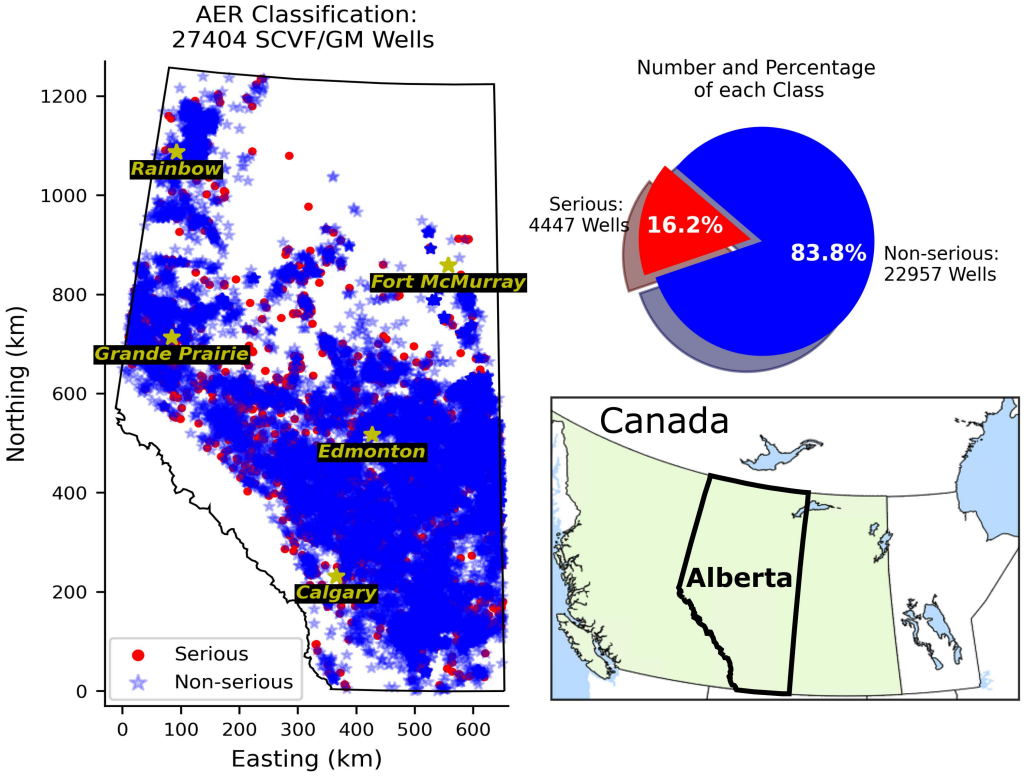

Figure 1. *Location map of Alberta Energy Regulator (AER) classification for test results (serious and non-serious) of surface casing vent flow (SCVF) and gas migration (GM) for energy wells in Alberta, Canada. The majority of wells are non-serious (83.8%)..*

wells had serious leakage that required immediate fixing to avoid environmental impacts. We considered known physical properties of the wells as training features to predict the probability of serious fluid leakage in the province of Alberta, Canada. The properties were retrieved from geoSCOUT, a large database of well characteristics in Alberta (geoSCOUT, 2022). Table 1 shows the 22 physical properties that were considered for each well displayed in Fig1. They include the following: properties 1 and 2 define deviated and horizontal wells (True/False). Properties 3 to 10 describe surface casing and production casing specifications of each well. Production-casing and surface-casing grades are string variables (text). Properties 11 and 12 are measured depth and the temperature of the borehole, respectively. Property 13 is the geological formation (text) targeted for production or injection. Property 14 shows the status of the well in text such as suspended, issued, abandoned. Property 15 is the age of each

well in months (counted from January 2022). Property 16 is the regional well density calculated as the total number of hydrocarbon wells with positive SCVF/GM test within 10km × 10km area around each well. Property 17 indicates the type of surface abandonment such as plate or cement; 18 is time in month since abandonment (counted from January 2022). Properties 19 to 22 are cumulative gas, oil, and water production and total months in production.

Table 1. *22 physical properties for each well in Fig1 retrieved from geoSCOUT (2022) (geoSCOUT, 2022).*

Physical Properties of Wells	
1- Deviated Hole (T/F)	12- Borehole Temperature (degC)
2- Horizontal Hole (T/F)	13- Prod./Inject. Formation
3- Surface-Casing Depth (m)	14- Well Status
4- Surface-Casing Size (mm)	15- Month Well Spudded
5- Surface-Casing Weight (kg/m)	16- Well Density (n/10km X 10km)
6- Production-Casing Depth (m)	17- Surface Abandonment Type
7- Production-Casing Size (mm)	18- Surface Abandonment Month
8- Production-Casing Weight (kg/m)	19- Cumulative GAS Prod. (e3m3)
9- Production-Casing Grade	20- Cumulative OIL Prod. (m3)
10- Surface-Casing Grade	21- Cumulative WATER Prod. (m3)
11- Measured Depth (m)	22- Total Production Month

Since a model should be trained first and then reasonably evaluated, the SCVF/GM test results in Fig1 was split into training (72%), validation (18%) and test sets (10%) as shown in Fig2. The percentage of non-serious (83.8%) and serious (16.2%) classes for the training, validation, and test sets should be identical to the results for the entire dataset shown in Fig1. The reason of having a test set as well as a validation set is to avoid overfitting and to evaluate the model based on a never-before-seen dataset. Developing such models always involves tuning hyperparameters; feedback that signals the performance of the model on a validation set is used for tuning. Although the model is never directly trained on the validation set, tuning the configuration of the model based on its performance on the validation set can quickly result in overfitting to the validation set. In other words, some information about the validation data leaks into the model whenever a hyperparameter is tuned. Therefore, the model should not have any access to any information about the test set; it is only applied at the end of the project to test the performance of the model. This process is also applied for all data processing steps (normalization, imputation, text handling). For example, the same statistics for normalization and imputation of the training set should be applied for the validation set and test set. Categorical variables should be efficiently converted to numbers before feeding predictive algorithms. To enhance the performance of algorithms, a target encoding technique (Kaggle, 2022) was utilized in this study for converting text to numbers for categorical variables. Target encoding is replacing the mean of target for each category. This may lead to potential overfitting. A weighting factor was used to smooth the calculated means and prevent overfitting (Kaggle, 2022). The calculated smoothed means of categories in the training set was applied to replace categories in the validation and test sets. By comparing the performance of training and validation sets, it was ensured that overfitting did not occur.

3. Methodology

Binary classification was applied using the 22 physical properties summarized in Table 1 as training features, while using the SCVF/GM test results (AER classification) as target with serious leakage as positive class (value 1) and non-serious leakage as negative class (value 0). Two major challenges were

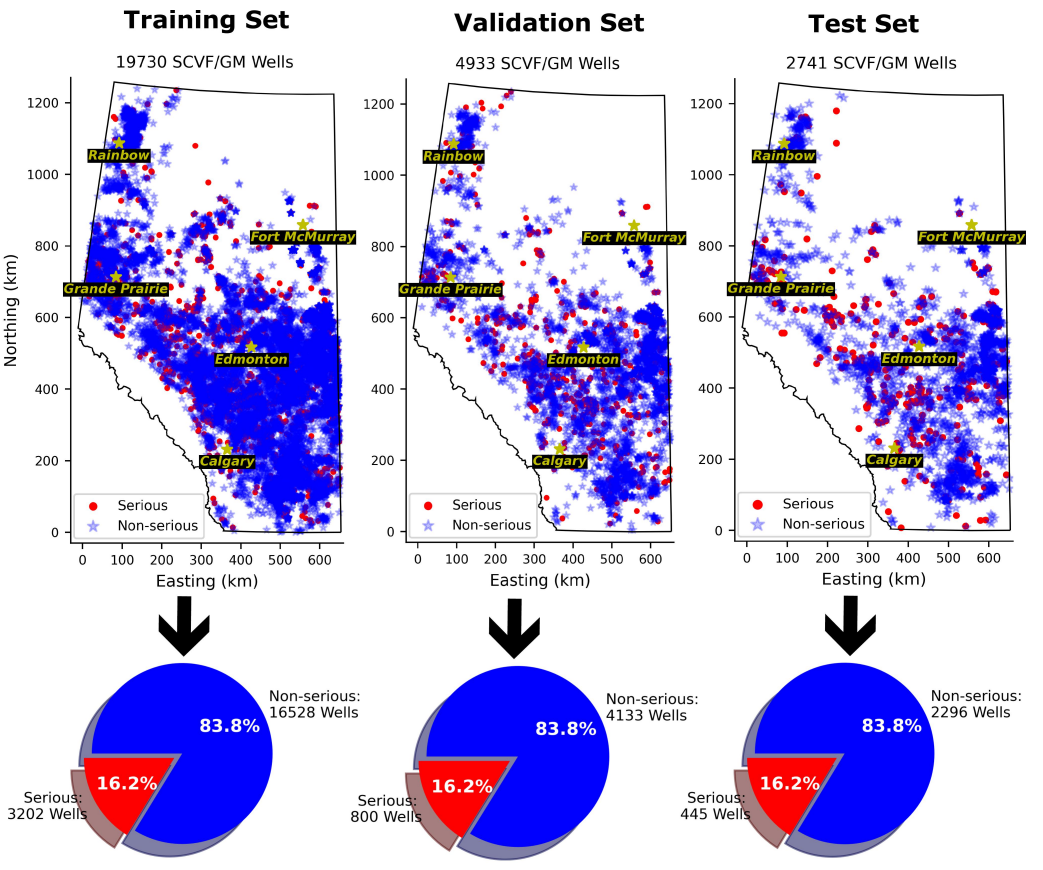

Figure 2. *AER classification shown in Fig1 separated into training, validation and test sets. Location map (top) and pie chart (bottom) for training (left), validation (middle) and test sets (right)..*

encountered. The first was the high number of missing data for the retrieved properties of the wells from geoSCOUT database that made the application of conventional imputation techniques inefficient. Therefore, a new approach was developed to impute missing values by conditioning using available data and quantifying the uncertainty of imputed values. The second challenge was the imbalanced number of classes with the negative class (non-serious) = 83.8% and the positive class (serious) =16.2%. A highly skewed distribution towards the negative class leads to forcing the predictive models to an unrealistically high classification for the negative class. Applying a random classifier for this dataset leads to high accuracy that can be misleading since accuracy is only reflecting the underlying class distribution (Montague et al., 2018; Brownlee, 2020). There are metrics that have been designed to arrive at more truthful results when working with imbalanced classes. Those performance metrics and approaches including over and undersampling were applied to resolve the issue of imbalanced data. Several predictive algorithms and aggregation of models were used to enhance the performance of the predictive tool. The final trained model can subsequently be applied to predict the probability of serious fluid leakage for the wells with known properties for which SCVF/GM tests are not available.

3.1. Imputation

Fig3 shows the percentage (left vertical axis) and number (right vertical axis) of missing data for the training set shown in Fig2. Some features (properties) have a significant number of missing data. Eliminating of the wells with missing data for those features is not reasonable since it would remove valuable information for other features. Missing data for each feature could be simply replaced with a constant value for example mean or median of the feature. However, this may lead to unreliable predictions and artifacts since a large population of data would have similar values. The K nearest neighbours (K-NN) is an algorithm that applies feature similarity. Missing data are imputed by finding the K's closest neighbours. K-NN is sensitive to outliers, and it does not include the uncertainty in imputed missing values. There are complex techniques such as multivariate imputation by chained equation (MICE) (Buuren and Groothuis-Oudshoorn, 2010) and deep learning (DataWing, 2022). However, the imputation using these techniques can be quite slow and computationally expensive for large datasets. They may also need special software, distributional assumption, and the uncertainty in imputation of missing data can not be taken into account. Multivariate Bootstrapping (Khan and Deutsch, 2016; Rezvandehy and Deutsch, 2017; Rezvandehy et al., 2019) is another approach that can be applied to quantify the uncertainty in the distribution of each feature, and then replace each missing value with a random sample from the distribution. This approach is fast, quantifies the uncertainty in imputation and the correlation between features are reproduced. However, the missing data cannot be simulated by conditioning on non-missing values. Therefore, a new imputation technique to quantify associated uncertainty in imputation by conditioning on non-missing values was developed in this study.

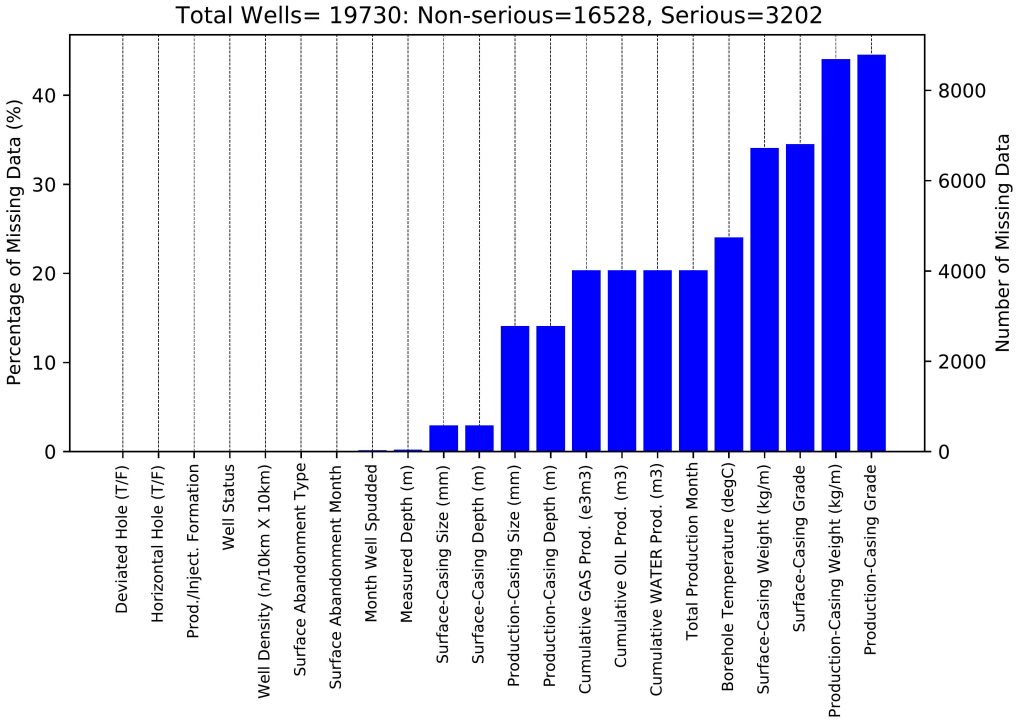

Figure 3. *Bar chart of missing values for the wells for the training set in Fig2 for 22 selected features (well properties from geoSCOUT database)..*

We pursued a method of imputing missing data using LU (lower–upper) simulation based on triangular decomposition of the correlation (standardized covariance) matrix. Many theoretical developments

of LU simulation have been pursued in geostatistics for geomodeling and spatial resampling (Davis, 1987; Journel and Bitanov, 2004; Deutsch and Journel, 1998; Khan and Deutsch, 2016; Rezvandehy and Deutsch, 2017). A modified LU conditional simulation (Davis, 1987) is suggested here for imputation by conditioning non-missing values to impute missing data for each well. This approach respects the correlation between features and quantifies the uncertainty in imputation of missing data. This method is fast and requires much lower computational power for large data sets compared with other techniques. The procedure can be summarized as follows:

1. **Normal Score Transformation** (Deutsch and Journel, 1998). Quantile-quantile transformation is applied to convert distribution of each feature **z** to a Gaussian distribution with mean=0 and standard deviation=1, which is required for LU simulation.
2. **Correlation Matrix of Features**. A correlation matrix (standardized covariance matrix) $\boldsymbol{\rho}$ for n features is shown in Fig4-a. The diagonal elements of this correlation matrix $\boldsymbol{\rho_{11}}, \boldsymbol{\rho_{22}}, ..., \boldsymbol{\rho_{nn}}$ are 1 representing the correlation of each feature to itself.
3. **Cholesky Decomposition**. The correlation matrix is then discomposed by Cholesky decomposition as $\boldsymbol{\rho} = \mathbf{LU}$, where **L** is the lower triangular matrix with all elements above diagonal elements is zero, and **U** is upper triangular matrix with zero values below diagonal elements. Only **L** is required for the LU simulation. Fig4-a shows the lower triangular matrix **L** achieved from Cholesky decomposition.
4. **Modified LU Conditional Simulation**. A vector of uncorrelated standard normal deviate **w** with mean=0, standard deviation=1 is simulated for each feature. The length of **w** for each feature is the number of data (here is the total number of wells for the training set). LU unconditional simulation can be simply calculated by $\mathbf{y} = \mathbf{Lw}$. Fig4-b shows how to generate a LU unconditional simulation achieving correlated Gaussian realization **y**. The unconditional simulation can be used for oversampling to improve the imbalance number of classes for classification (see Section 3.4). However, conditional simulation is needed to simulate missing data conditioned based on non-missing values. For conditioning non-missing features, each array of **w** vector with known values needs to be converted to $\mathbf{w^c}$ that is a function of the non-missing features. For example, if feature 1 z_1 and feature 2 z_2 are available, LU conditional simulation can be applied to keep z_1 and z_2 unchanged and simulate y_3 to y_n (missing data) conditioned based on z_1 and z_2 as shown in Fig4-c. This conditioning requires to convert w_1 and w_2 to w_1^c and w_2^c as follows:

$$w_1^c = \frac{z_1}{L_{11}} \quad , \quad w_2^c = \frac{z_2 - L_{21} w_1^c}{L_{22}} \tag{1}$$

where L_{11}, L_{21} and L_{22} are elements of the lower triangular matrix **L** from Cholesky decomposition (Fig4-c). This process needs to be repeated to calculate all w^c for conditioning non-missing features. The n^{th} w is calculated as:

$$w_n^c = \frac{z_n - L_{nn-1} w_{n-1}^c}{L_{nn}} \tag{2}$$

w for missing data should change for each feature and each instance (random sampling from Gaussian distribution) leading to quantification of the uncertainty in missing data after simulation. The main challenge for this modified LU simulation is the ordering of missing and non–missing features for each row of data. If missing data are placed first followed by non–missing values, the conditioning cannot be applied since w_1 for missing data are randomly sampled and then w_2^c for non–missing values are calculated based on Equation 1: **Lw** cannot enforce the correlation between simulated and non–missing values. However, if non–missing values are placed first followed by missing data, the conditioning will be properly applied because of calculating w_1^c for non–missing values before w_2. Therefore, non–missing values must be placed first followed by missing data for each instance (row of data). This requires reconstructing the correlation matrix for each instance to be consistent with the order of features. This ordering is not important within missing and non–missing features. Fig5 shows how to change the order of features and correlation

matrix based on non–missing and missing data. There are four features and four rows. Fig5-b shows how to change the order of raw data in Fig5-a and the related correlation matrix for each row is shown in Fig5-c. All four features of row 1 have non–missing values and therefore changing the order of features is not required. However, for row 2 to 4 the order of features should be changed to start with non–missing values. The correlation matrices should be consistent for each row of data. For the LU conditional simulation (Fig4-c), Cholesky decomposition must be calculated for each covariance matrix separately.

5. **Back-transform from Gaussian to Original Space**. The simulated values in Gaussian space must be back-transformed to original space. This requires to lookup through the standard Gaussian distribution to find the CDF (cumulative distribution function) probability (P) of each simulated value. Then, lookup through the original distribution of related feature to find the P-quantile of the simulated value in the original space. This ensures that non-missing values remain unchanged.

Steps 4 to 5 are repeated for all data in the training set to impute all missing data. Due to random sampling from the distribution of each feature, this approach quantifies the uncertainty in imputation of missing data by running the process described above many times (for example 100 times). Standard normal deviate $\mathbf{w}$ should be different for each run to simulate different values for missing data while keeping the non-missing values unchanged and respecting the correlation between features. Moreover, implementation of the above outlined steps is straightforward especially with Python programming language. Since Cholesky decomposition of the correlation matrix should be applied only once per each unique order of features, the process is fast and efficient for big datasets.

To evaluate the efficiency of the proposed imputation technique a synthetic example is considered with four correlated features with 10000 data as shown in Fig6-a. Features 1 and 2 are Gaussian and lognormal distributions, respectively while features 3 and 4 are triangular distributions with different statistics (mean and mode). Fig6-b shows the correlation matrix between features (below diagonal elements) and percentage of missing data for each bivariate feature (above diagonal elements). The highest percentage of missing data for bivariate distributions is between feature 1 and feature 2 (51%), and the lowest is between feature 3 and feature 4 (32%). Fig7 shows a scatter plot matrix including histograms of each feature on diagonal elements before imputation (a) and after imputation (b). The correlation between features ($\rho_{x,y}$), the shape of univariate (histograms) and the bivariate distributions are reproduced after imputation. Therefore, the technique is highly suitable for imputation of realistic data.

Fig8 shows a correlation matrix for 22 well properties of the AER classification (SCVF/GM test results), before imputation (Fig8-a) and after imputation (Fig8-b) for the training set. The correlations between features after imputation have been reproduced. An example is provided at the bottom of Fig8 that shows the crossplot between Surface-Casing Depth (m) and Production Casing Depth before and after imputation: imputed data (stars) have the same correlation as non-missing values (circles). The imputation can be repeated multiple times to quantify the associated uncertainty of imputed values. The same approach must be applied for imputation of missing data in validation and test sets; however, the correlation matrix and feature distribution of the training set must be used to prevent information leak into these datasets. The training data is subsequently ready to feed a machine learning algorithm for binary classification.

3.2. Predicting Algorithms

A wide range of machine learning algorithms was applied using Python's scikit-learn package (Pedregosa et al., 2011). A brief explanation of each approach is as follows:

- *Stochastic Gradient Descent* is a good place to start for large data sets (Géron, 2019). Gradient descent provides a general idea of how to minimize a cost function by iteratively tweaking parameters. However, it can be very slow for large data sets: Stochastic Gradient Descent is a

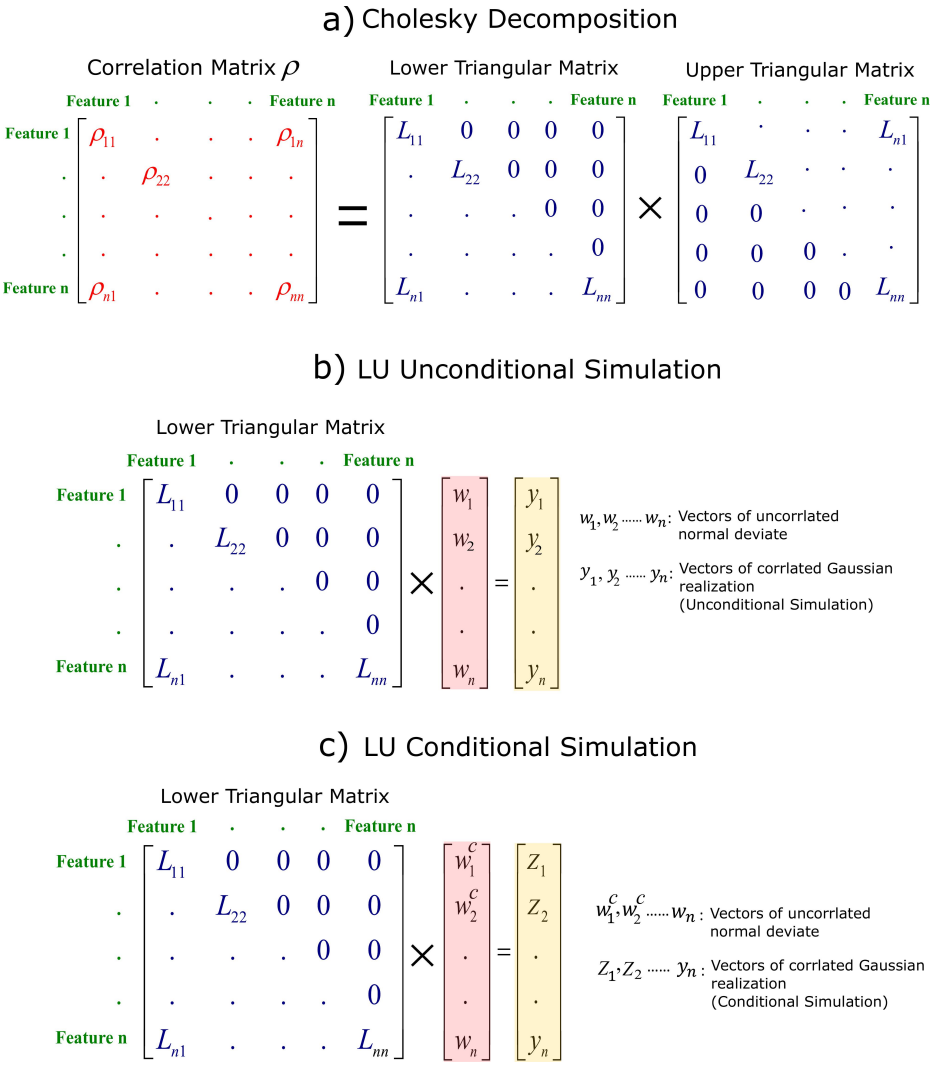

Figure 4. *a) Cholesky decomposition of correlation matrix for n features (well properties). b) LU unconditional simulation. c) LU conditional simulation..*

efficient because it just picks a random instance at every iteration and computes the gradients based only on that single instance (Bottou, 2012; Géron, 2019).

- *Logistic Regression* is a simple approach to estimate the probability of a particular class. It calculates a weighted sum of the input features (plus a bias term) and uses a sigmoid function to estimate the probability of each class. (Lemon et al., 2003; Zhu et al., 2019; Robles-Velasco et al., 2020; Rekha et al., 2019).
- *Support Vector Machine* is a powerful algorithm to perform linear or nonlinear classification. The fundamental idea is to have the largest possible margin between the classes. It predicts the class of a new instance by computing a decision function with optimum parameters (Géron, 2019; Gandhi, 2018; Escobar and Morales-Menendez, 2019).
- *Random Forest* are among the most versatile and reliable machine learning algorithms for non-linear and complex data. Decision Tree is the fundamental component of Random Forest; it is applied based on a flowchart structure in which each node denotes a test, each branch represents the result

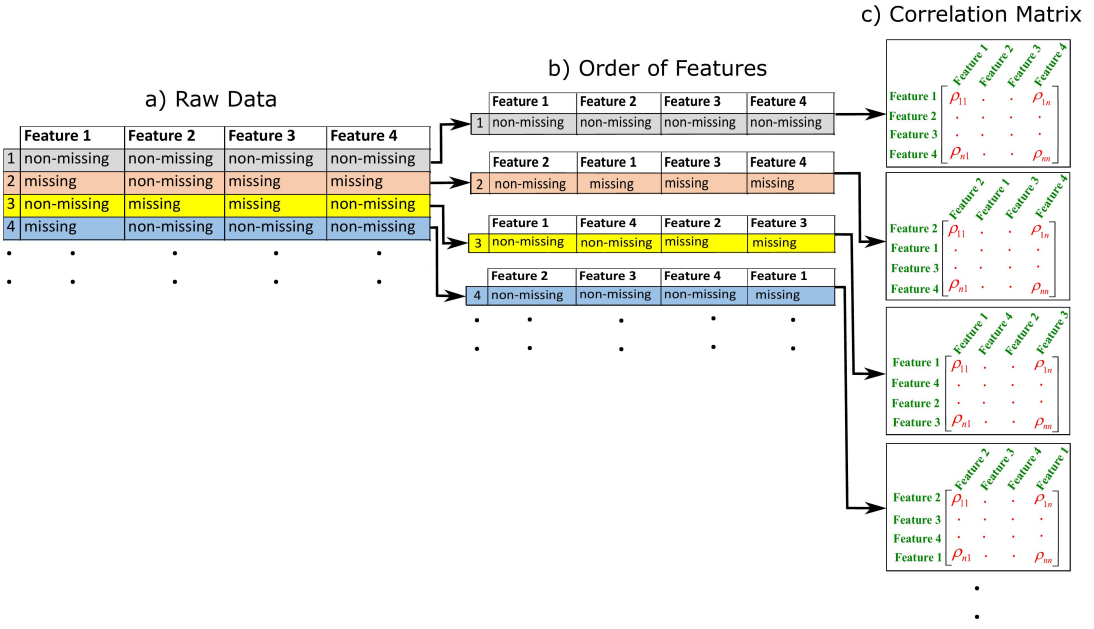

Figure 5. *a) Schematic illustration of four features with four rows of data with missing and non-missing values. b) Change the order of features for raw data to have non-missing values first followed by missing data. c) Correlation matrix for each row in b..*

a) Synthetic Data

	Feature 1	Feature 2	Feature 3	Feature 4
1	NaN	1.97	1.60	NaN
2	6.51	NaN	NaN	4.13
3	4.19	0.89	NaN	0.64
4	NaN	1.35	1.93	0.50
•	•	•	•	•
•	•	•	•	•
9998	5.72	NaN	NaN	0.77
9999	NaN	NaN	1.56	2.57
10000	3.32	0.62	3.68	0.13

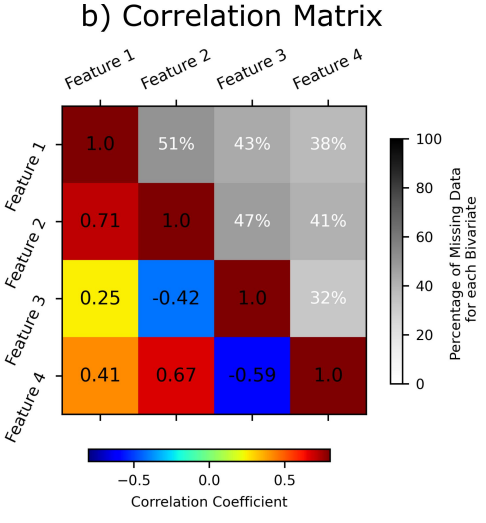

Figure 6. *a) Synthetic example of four features with 10000 data. NaNs are missing data. b) Correlation matrix between features (below diagonal elements) and percentage of missing data for each bivariate feature (above diagonal elements). Maximum percentage of missing data is 51% for bivariate distribution of Feature 1 and 2..*

of the test, and each leaf node represents a class label. The Random Forest randomly creates and

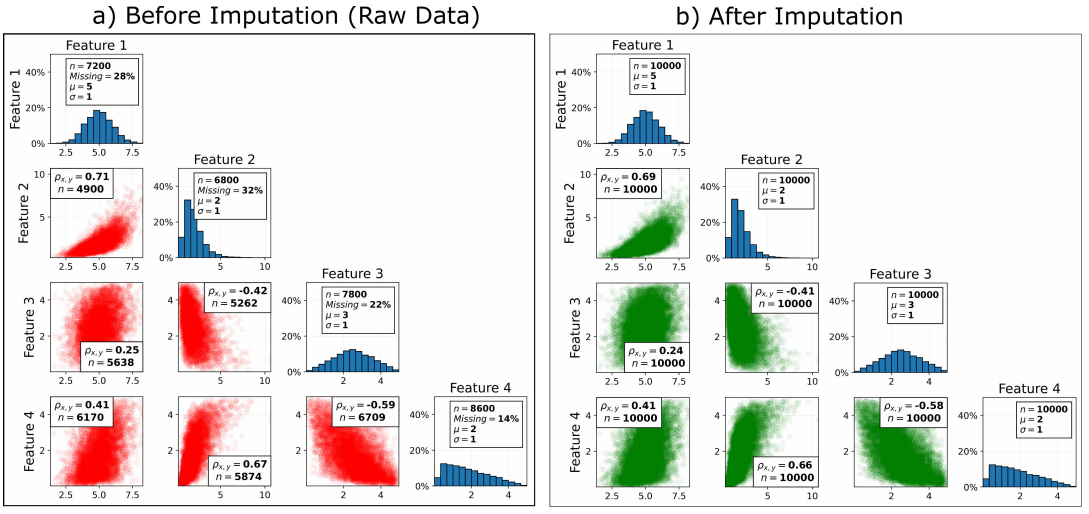

Figure 7. *Scatter plot matrix for synthetic example of four features with 10000 data before imputation (a) and after imputation (b). Histograms of each feature are shown on diagonal elements. n is number of non-missing values for each univariate and bivariate distribution and $\rho_{x,y}$ is the correlation coefficient for each bivariate distribution. μ is the mean and σ is the standard deviation..*

merges multiple decision trees and predicts a class that gets the most votes among all trees. Despite its simplicity, it is one of the most powerful machine learning algorithms available today (Géron, 2019; Gariazzo et al., 2020; Hashimoto et al., 2019; Lassalle et al., 2019; Ozigis et al., 2020).

- *Adaptive Boosting* can be applied for any predictor mentioned above to enhance the performance and turn into a stronger learner. The general idea is to use a base classifier, then correct the base classifier by paying attention to the training instances that are underfitted. This leads to a new classifier focusing more on the hard cases. Decision Tree, which is a weak learner, is typically applied as the base classifier in Adaptive Boosting (Géron, 2019; Chen et al., 2020).
- *Deep Neural Network* is a specific subfield of machine learning for tackling a very complex problem. In comparison with shallow learning or artificial neural networks, deep learning usually involves more successive layers of representations that are learned from training data. The large network's architecture may lead to some problems such as vanishing/exploding gradients, overfitting, computational cost, and slow training. However, these problems can be resolved by tuning some hyperparameters (Chollet, 2018).

For a sanity test, predictions from simple rule of thumb called Dummy Classifier were compared with the results from the algorithms described above. Dummy Classifier was used as a simple baseline with other classifiers being expected to have higher performance. This approach is especially useful for imbalanced datasets (Pedregosa et al., 2011).

Ensemble Learning is usually applied near the end of a project when a number of good and promising predictors are built to integrate them into an even stronger predictor. It works by aggregating the predictions of a group of predictors. Hard Voting is a simple Ensemble Learning that aggregates the predictions of each classifier and predicts the class that gets the most votes. Soft Voting is another Ensemble Learning that works by averaging the probability of each class and predict a class with the highest probability. Soft Voting often achieves higher performance than Hard Voting due to giving more weight to highly confident votes (Géron, 2019).

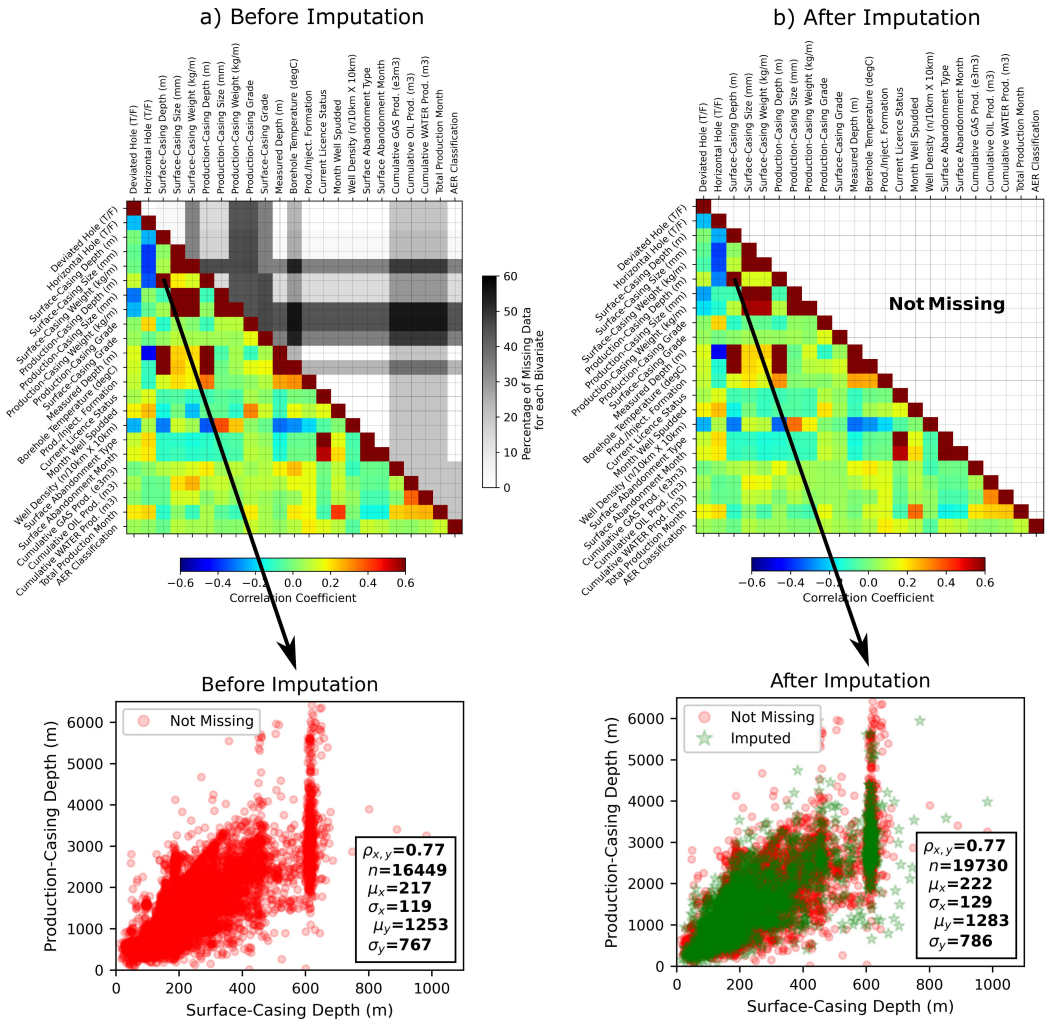

Figure 8. *Correlation matrix (below diagonal elements) for 22 well properties and AER classification (SCVF/GM test results), before imputation (a) and after imputation (b). The percentage of missing data for bivariate distribution are shown above diagonal elements. Cross plots between Surface-Casing Depth (m) and Production-Casing Depth (m) before and after imputation are shown at the bottom. n is number of non-missing values for each univariate and bivariate distribution and $\rho_{x,y}$ is the correlation coefficient for each bivariate distribution. μ is the mean and σ is the standard deviation..*

3.3. Performance Measurement

Binary classification was applied for each algorithm listed above for training, validation, and finally test sets. The performances were compared to achieve the most reliable classifier. K-fold cross-validation was utilized to get a clean prediction for the training set (to prevent overfitting): it splits the training set into K-folds, and then predicts each fold using a model trained on the remaining folds. Evaluating a classifier is often more challenging than a regressor. The most common approach for assessment is accuracy, which is calculated by the number of true predicted over the total number of data. However, accuracy alone may not be practical for performance measurement of classifiers, especially in the case of skewed datasets. Accuracy should be considered along with other metrics. A confusion matrix is a much better way to evaluate the performance of a classifier. The general idea is to consider the number

of times instances of negative class are misclassified as positive class and vice versa. In a confusion matrix, each column represents a predicted class, while each row signifies an actual class (Géron, 2019). Although the confusion matrix represents a lot of information, sometimes more concise metrics (including accuracy) is preferred as below.

1. **Accuracy**= $\frac{TP+TN}{TP+TN+FP+FN}$, where TP=true positive, TN=true negative, FP=false positive, FN=false negative. It is the proportion of correct predictions over total number of data.
2. **Sensitivity** (Recall)= $\frac{TP}{TP+FN}$: the proportion of correct positive predictions to the total positive classes.
3. **Precision**=$\frac{TP}{TP+FP}$: the proportion of correct positive prediction to the total positive predicted values.
4. **Specificity**=$\frac{TN}{TN+FP}$: true negative rate or the proportion of negatives that are correctly identified.

The harmonic mean of precision and sensitivity can also be calculated as another metric (Montague et al., 2018; Géron, 2019); it gives much more weight to low values, so a classifier only represents a high harmonic mean if both precision and sensitivity are high. However, for the classification in this study, precision is less important than sensitivity. Fig9 shows the confusion matrix of Dummy Classifier on the left (a), and confusion matrix of Stochastic Gradient Descent on the right (b) for the training set. The lower the values are for off-diagonal elements of a confusion matrix, the higher is the performance of the classifier. The metrics accuracy, sensitivity and specificity are higher for the

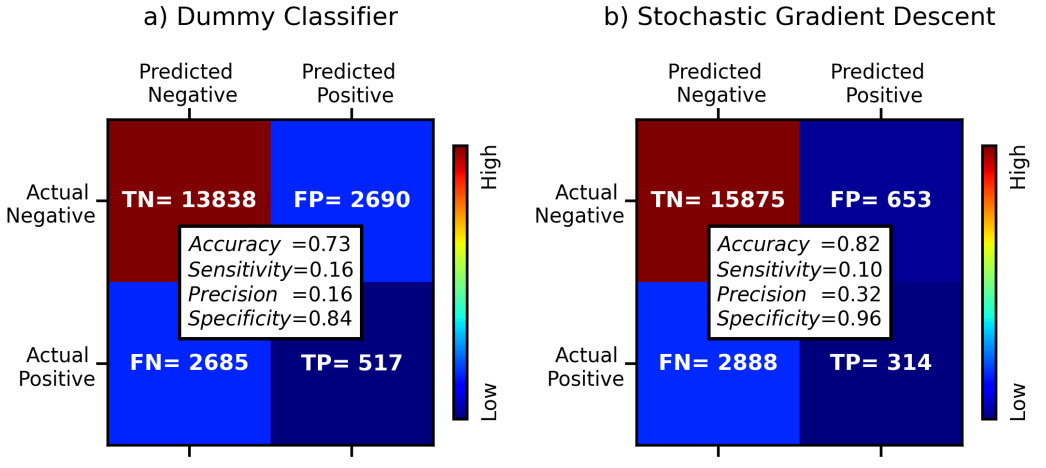

Figure 9. *Confusion matrix for Random classifier (a) and Stochastic Gradient Descent (b) including accuracy, sensitivity, precision and specificity, where TP=true positive, TN=true negative, FP=false positive, FN=false negative..*

Stochastic Gradient Descent because of lower number of FN and FP than for the Dummy Classifier. However, precision and sensitivity are significantly low: sensitivity is lower than Dummy Classifier. Utilizing stronger classifiers might improve these metrics. If no classifier can address this problem, improving either precision or sensitivity may be considered depending on requirement of the problem. In this study, it should be fine if precision is relatively low, which means having noticeable false alerts of serious leakages while they are non-serious. However, we expect the classifier to have a high sensitivity that can detect serious leakages with a minimal number of false negatives. Sensitivity can be manually increased by tweaking the threshold for assigning a class to each instance. The probability 0.5 is often used as a threshold. For example, if the predicted probability is above 0.5, the instance is labeled as

class 1; otherwise, class 0 is assigned. If the threshold manually decreases, the prediction will have more class 1 that leads to higher sensitivity. This approach seems very straightforward to get any desired sensitivity; however, it may not be practical to apply tweaking threshold if the predicted sensitivity is very low and decreasing the threshold will artificially increase sensitivity and decrease significantly other metrics. Since the low sensitivity is caused by the imbalanced data in this work, class distribution can be adjusted to reasonably increase sensitivity.

3.4. Resampling

Resampling techniques are used to adjust the class distribution of training data (the ratio between the different classes) to feed more balanced data into predictive models; thereby creating a new transformed version of the training set with a different class distribution. Two main approaches for randomly resampling an imbalanced dataset are:

- **Undersampling:** This approach deletes random instances of a majority class from the training set. In the transformed version of the training set, the number of instances for the majority class are reduced. This process is repeated to achieve the desired class distribution, such as an equal number of instances for each class. This approach may be more efficient when there are a lot of instances in the majority class. A drawback of undersampling is eliminating the instances that may be important, useful, or critical for fitting a robust decision boundary (He and Ma, 2013; Brownlee, 2021). In this study, undersampling was considered for the instances with missing well properties. Although missing data are already imputed, it is better to remove random instances that have imputed values instead of non-missing instances.
- **Oversampling:** This approach adds random instances (duplicate) of the minority class to the training set. Instances from the minority class are selected and added to the training data to generate a new balanced training set. The samples are chosen from the original training set. Oversampling may be applied when there are limited instances in the minority class. A disadvantage of this technique is that it increases the likelihood of overfitting because of including the exact copies of the minority class examples (Fernández et al., 2018; Brownlee, 2021). In this study, we used LU unconditional simulation for oversampling to avoid the inclusion of exact duplicates as discussed in Section 3.1 and shown in Fig4-b. This approach is fast and includes associated uncertainty for resampling.

Combining both oversampling and undersampling can lead to improved overall performance in comparison with performing one approach in isolation (Brownlee, 2021). We combined both techniques with equal percentage: undersampling with a selected percentage is applied to the majority class to reduce the bias on that class, while also applying the same percentage for oversampling of the minority class to improve the bias towards these instances. Resampling was applied for different ratios of $\frac{Class\,1}{Class\,0}$, where Class 1 and Class 0 are the proportions of serious and non-serious leakage, respectively. For each ratio, the metrics accuracy, specificity, sensitivity, and AUC (area under the curve) were calculated. The ratio that has similar performance for all the metrics is the most reliable ratio.

4. Results and Discussion

Fig10 shows the performance of the predictive algorithms based on the four metrics achieved from the confusion matrix for the training set (a) and validation set (b) and test set (c). The hyperparameters for each algorithm are fine-tuned to enhance performance. The trained models derived using the training set are applied for prediction of the validation set and test set to confirm that overfitting is not occurring. The comparison between Fig10-a to c shows that the performances are very similar except for Deep Neural Network having a lower precision for the test set indicating overfitting for this algorithm. Specificity is the highest and sensitivity is lowest for almost all classifiers. The Dummy classifier has

the lowest values for all metrics except for sensitivity that is close to the predictive algorithms. A highly skewed distribution towards negative class leads to have sensitivity close to that of the Dummy classifier even for powerful algorithms. In order to achieve better comparison of the classifiers, a tool called receiver operating characteristic (ROC) curve was used to measure performance. The ROC curve plots the true positive rate (sensitivity) against the false positive rate (1-specificity). Every point on the ROC curve represents a chosen cut-off even though it cannot be seen. The most common way to compare classifiers is to measure the area under the curve (AUC). A perfect classifier has an AUC equal to 1, whereas a purely random classifier has an AUC equal to 0.5. Fig11 shows ROC curves of classifiers for

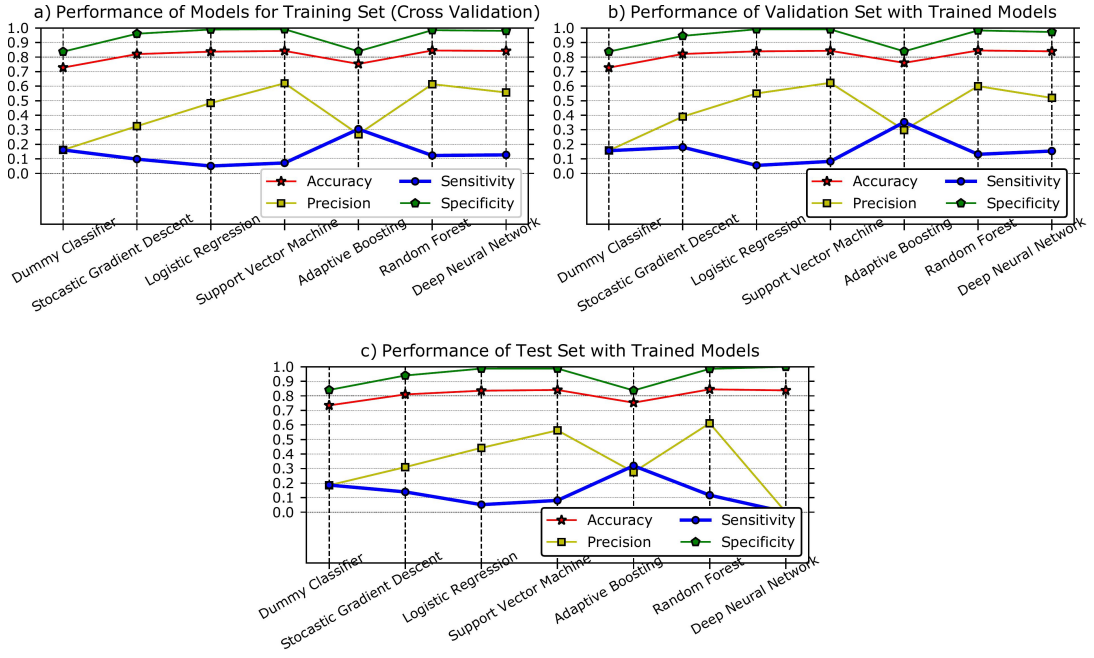

Figure 10. *Performance of predictive algorithms for training set (a), validation set (b) and test set (c) based on the metrics accuracy, sensitivity, precision, and specificity achieved from confusion matrix. Specificity is the highest and sensitivity is lowest for almost all classifiers. .*

the training set (a), validation set (b) and test set (c). The AUC is shown for each algorithm. The Dummy classifier has the lowest AUC (≈ 0.5). Random Forest and Deep Neural Network have the highest AUC; however, due to overfitting, AUC for Deep Neural Network decreased for the test set. Therefore, it is concluded that the Random Forest is the most reliable classifier with AUC$\approx$0.75. Using the above-described approach, the importance of each feature (Table 1) to predict serious leakage was determined by 100 realizations of Random Forest with 100 realizations of imputation, one realization at a time to quantify the importance of each feature with full picture of uncertainty. The results are summarized in Fig12. Each bar shows the mean percentage of importance with uncertainty interval (variance). Month Well Spudded (e.g., the age of the wells in months) has the highest importance to predict fluid leakage followed by six features that have similar performance: Total Production Month, Surface-Casing Depth, Production Casing Depth, Measured Depth, Cumulative WATER Prod and Cumulative GAS Prod. The features do not have strong linear correlations with the target (see Fig8 for linear correlation of the well properties with AER classification). However, increasing Month Well Spudded, Total Production Month, Measured Depth, Cumulative GAS Prod and Cumulative WATER Prod probably lead

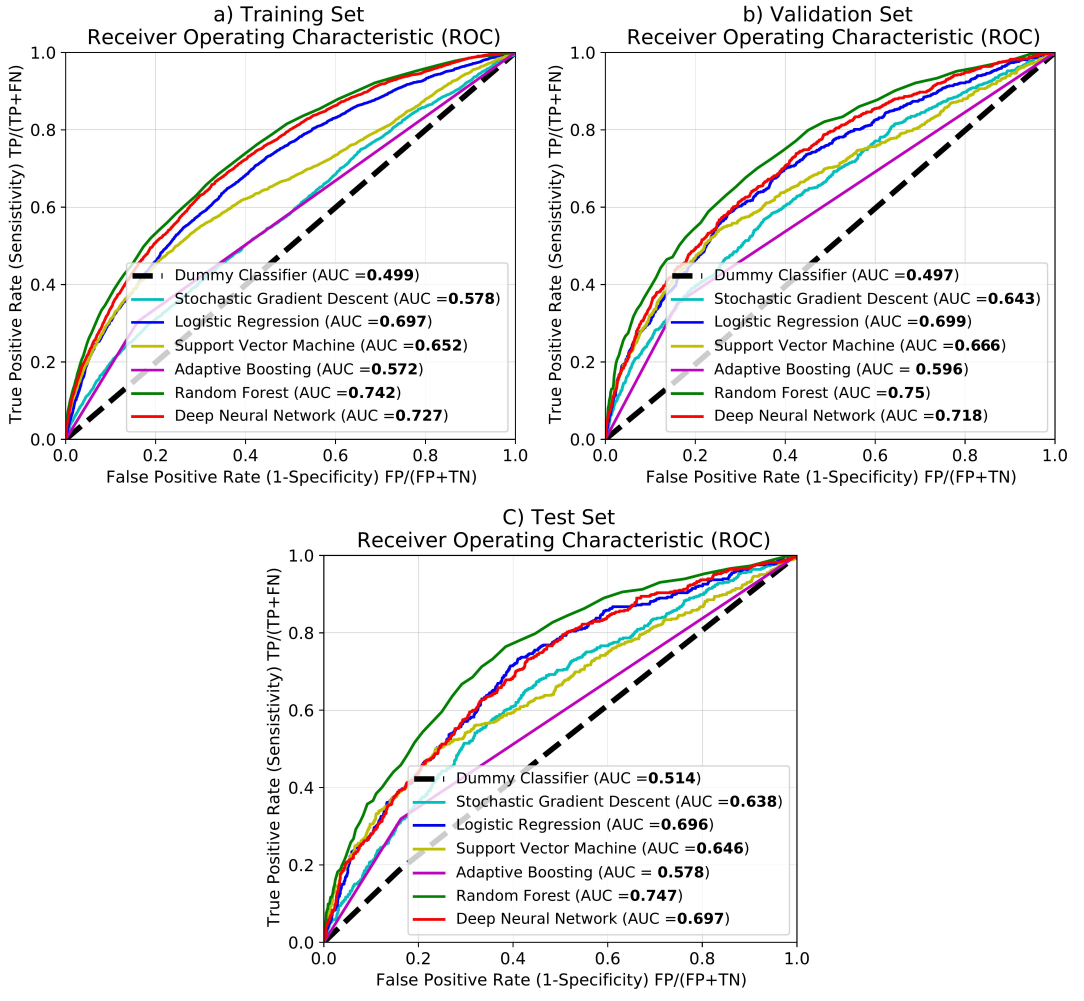

Figure 11. *ROC curves with calculated AUC for training set (a) and validation set (b) and test set (c). TP=true positive, TN=true negative, FP=false positive, FN=false negative. Random Forest has the highest AUC without overfitting..*

to more serious fluid leakages according to the positive correlation with the target in Fig8. It is interesting to note that the features Deviated Hole and Horizontal Hole have very low importance on impacting serious leakage. Furthermore, the size of production and surface casing, and the surface abandonment type appear to have negligible influence on the likelihood of serious fluid leakage.

It is suggested that the trained Random Forest model can be applied to predict the probability of serious fluid leakage for the wells without SCVF/GM field tests. The predictions will have high accuracy (>0.8) and specificity (>0.95), relatively low precision (0.6), but sensitivity will not be reliable: similar to prediction of Dummy Classifier (see Fig10) which is due to imbalanced class. Therefore, we combined undersampling and oversampling to increase sensitivity by adjusting class distribution. The resampling was applied for different ratios of $\frac{Class\,1}{Class\,0}$, where Class 1 is proportions of serious leakage and Class 0 is the proportion of non-serious leakage. Increasing this ratio may lead to an increase in sensitivity but a decrease in other metrics. The ratio that has similar performance for the metrics must be the most reliable ratio. Random Forest was applied for prediction since it has the highest

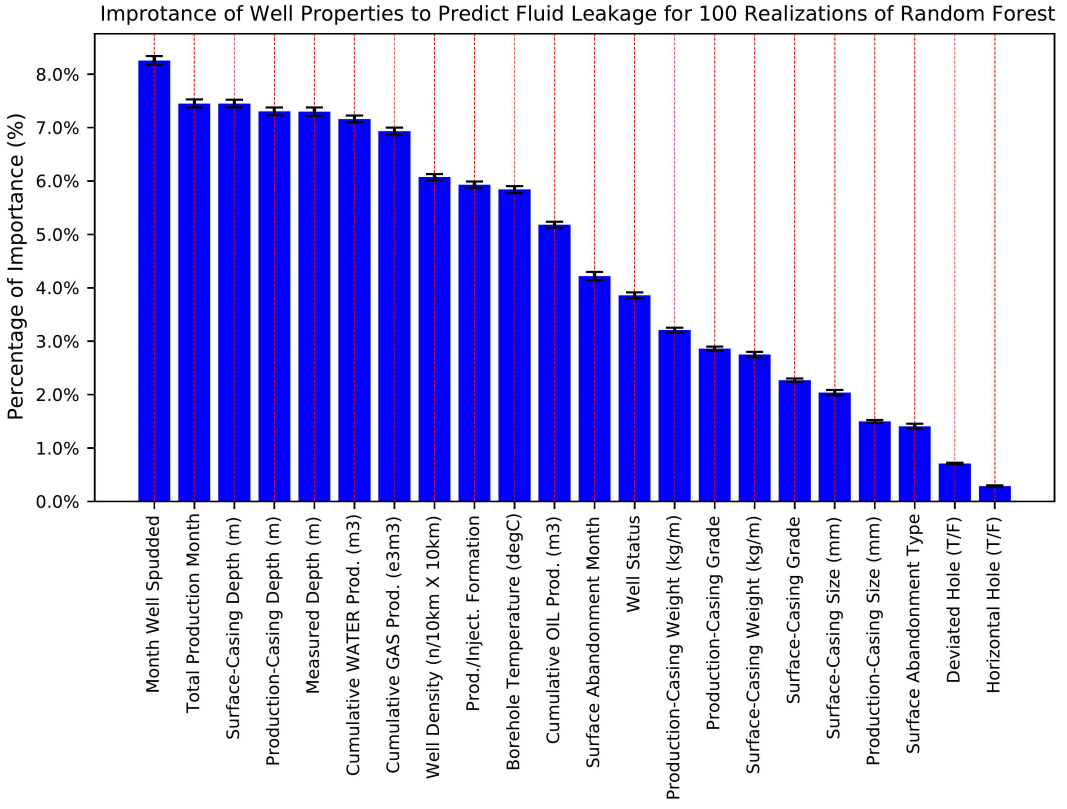

Figure 12. *100 realizations of Random Forest with 100 realizations of imputation, one realization at a time to quantify the importance of each feature with uncertainty for predicting target (AER classification). The feature Month Well Spudded (age of the wells in months) has the highest importance; Deviated Hole and Horizontal Hole have the lowest importance..*

performance (Figures 10 and 11). K-fold cross validation can be applied on the transformed version of the training set. Fig13-a shows a 5-fold cross validation after resampling the training set for 22 ratios of $\frac{Class\,1}{Class\,0}$ from 0.25 to 1.95. By increasing the ratio, AUC increases; sensitivity increases significantly; specificity decreases but accuracy almost remains unchanged. The metrics accuracy, specificity and sensitivity have equal performance (0.8) for the ratio 1.02. However, K-fold cross validation after resampling incorrectly leads to overoptimism and overfitting since the class distribution of the original data is different from the sampled training set (Santos et al., 2018). The correct approach is to resample within K-fold cross validation. Fig14 shows a schematic illustration of resampling within 5-fold cross validation. The training set is divided into 5 stratified splits: the folds of each split have the same class distribution (percentage) of the original data. Resampling was applied on the training folds of each split. A model was trained on the resampled training folds. The test fold, which preserves the percentage of samples for each class in the original dataset, was predicted with the trained model. This process was repeated for all 5-splits that leads to 5 models. Fig13-b shows resampling within 5-folds cross validation for the ratios of $\frac{Class\,1}{Class\,0}$. Compared with Fig13-a, AUCs have decreased significantly which confirms overoptimism and overfitting for applying K-fold cross validation after resampling. The ratio 1.27 in Fig13-b is the point where the line of all metrics cross: the model performs almost equally for two classes. Therefore, the ratio 1.27, which signifies there are more wells with serious

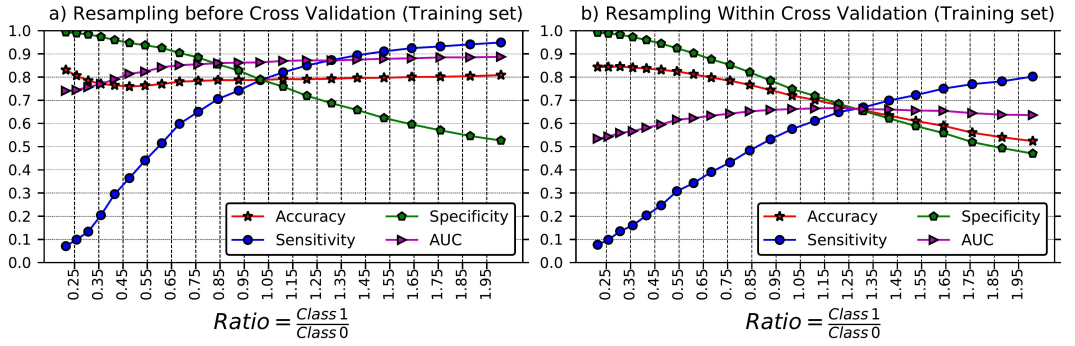

Figure 13. *Resampling before (a) and within (b) 5-fold cross validation for training set for the ratios of* $\frac{Class\,1}{Class\,0}$ *by Random Forest algorithm. Resampling before cross validation (a) is incorrect due to overoptimism and overfitting. The metrics are accuracy, specificity, sensitivity and AUC (area under the curve)..*

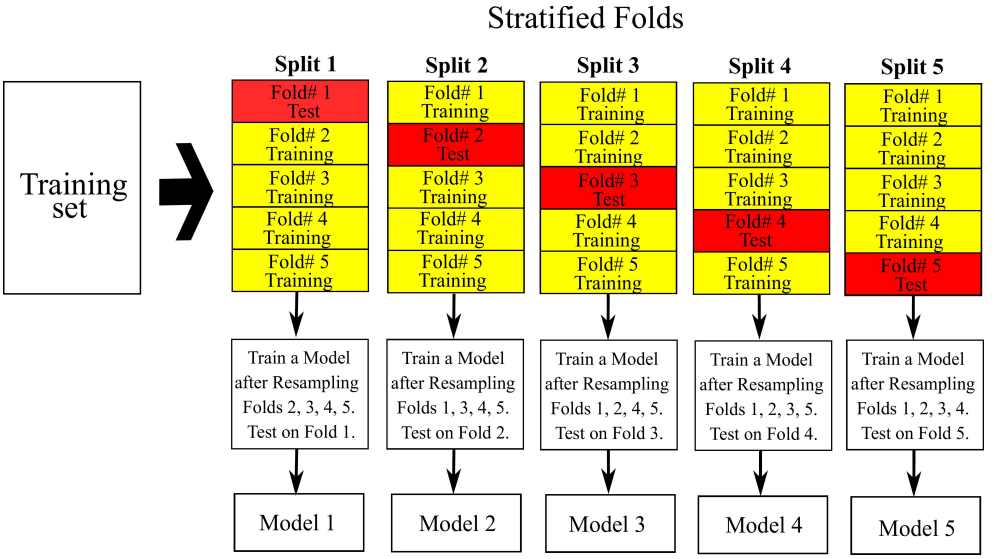

Figure 14. *Schematic illustration of resampling within 5-fold cross validation that leads to 5 models (models 1 to 5). Resampling is applied only on the training folds. A model is trained for the resampled training folds of each split. The trained model is used to predict the test fold which preserves the percentage of samples for each class in the original data set. .*

leakage in the training data, is selected as the ratio of two classes to achieve a more reliable sensitivity. The trained models 1 to 5 obtained from 5-fold cross validation (Fig14) were applied separately to predict validation using the test set. Then, the predictions from each model were aggregated and predicted the class that gets the most votes; since each model is trained on random subsets of training sets with random sampling, it is reasonable to aggregate the predictions. This process was repeated for other classifiers in Fig11 to compare the performances with Random Forest: the models 1 to 5 in Fig14 were trained with the classifiers and then aggregate the prediction of these models on validation and test sets. Fig15 shows the performance of the classifiers for the validation set (a) and test set (b) for the resampling ratio of $\frac{Class\,1}{Class\,0} = 1.27$ within 5-fold cross validation. All classifiers for validation and test set

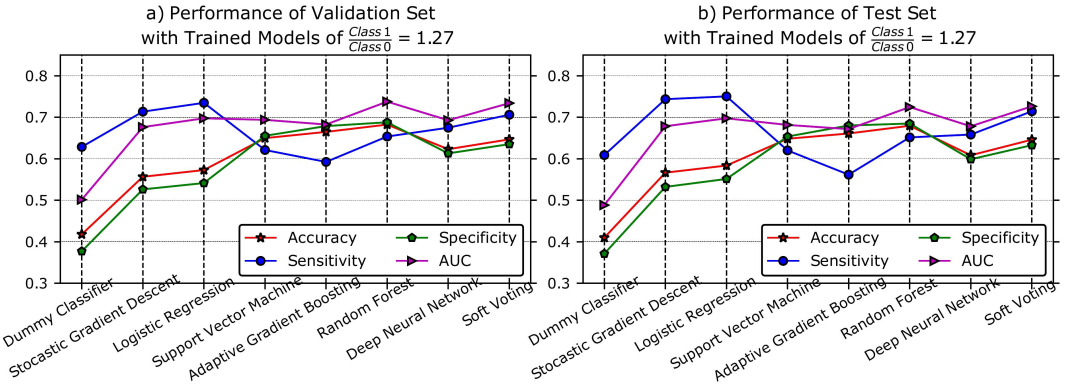

Figure 15. *a) Performance of the classifiers for validation set (a) and test set (b) for the sampling ratio of* $\frac{Class\,1}{Class\,0} = 1.27$ *within 5-fold cross validation. All classifiers for validation and test set have reasonably higher performance than Dummy Classifier. There is small overfitting in the validation set because of fine-tuning of hyperparameters. Soft voting, integration of Logistic Regression and Random Forest, is the most reliable classifier with high sensitivity and AUC (area under the curve), and reasonable accuracy and specificity..*

have a higher performance than the Dummy Classifier. Due to resampling, sensitivity has significantly increased (compared with Fig10). This leads to remarkable reduction for false negative predictions of wells with serious leakage. The performance for the test set is a little lower than for the validation set. This may be related to minor overfitting in the validation set while fine-tuning hyperparameters of classifiers. Predicting the probability of serious fluid leakage with known well properties (Table 1) for the wells without field test in Alberta will likely have similar performances as Fig15-b. Random Forest has the highest AUC but sensitivity is low. Logistic Regression has the highest sensitivity, but accuracy and specificity are relatively low compared with other algorithms. Soft Voting was applied at the end to integrate Random Forest and Logistic Regression by averaging the probability of each class and predict a class with the highest probability. Soft Voting has high sensitivity and AUC, and reasonable accuracy and specificity. Therefore, it is the most reliable classifier to predict the probability of serious fluid leakage for hydrocarbon wells without SCVF/GM field tests.

To apply the trained classifier for leakage detection, 1000 hydrocarbon wells without field tests were randomly selected in Alberta, Canada. Fig16-a shows the location map of the wells. The 22 physical properties of the wells (Table 1) used for training the classifier were utilized to predict the probability of serious fluid leakage for each well. Fig16-b shows percentage of missing data for the well properties. The distribution of the training set for each well property should be applied for normalization, text handling and imputation of these 1000 wells. Since some properties have more than 40% missing data, the uncertainty of imputed values should be incorporated in final prediction. This requires to run the developed LU conditional simulation for multiple realizations: each realization gives different imputed values. 100 realizations of imputation were generated. The trained classifier was applied 100 times using one realization of imputation at a time. This results in quantification of the uncertainty for the predicted probability of serious fluid leakage. The 100 predicted probabilities for each well can be aggregated by mean, median or 25^{th}, 75^{th} percentiles for decision making. Fig17-a shows the calculated mean of probability for serious fluid leakage of the 100 realizations for each well. The histogram of predicted probabilities for two wells (with means of probability for serious fluid leakage of 0.42 and 0.70) in southern Alberta is shown to represent that each well has 100 predicted probabilities achieved by running the classifier with 100 realizations of imputed values. Figures 17-b and c show the location map of the wells that have the mean of probability of serious fluid leakage higher than

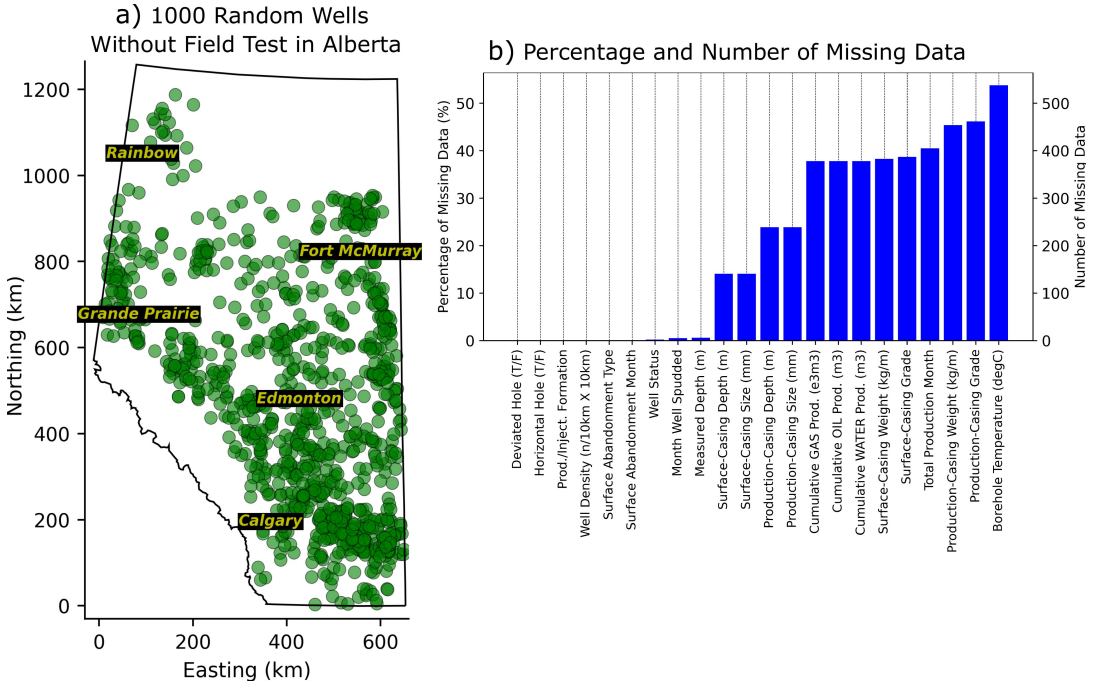

Figure 16. *a) Location map of 1000 random wells without field test in Alberta, Canada. b) Bar chart of missing values of the wells in (a) for 22 well properties retrieved from the geoSCOUT database. .*

0.6 and 0.8 in Fig17-a, respectively. There are 136 wells with the probability higher than 0.8 probably representing the leakiest wells for this random selection of 1000 wells. Therefore, field test operations can be optimized by targeting these 136 wells in Fig17-c first, followed by test for the rest of the wells shown in Fig17-b. To optimize field test procedures, the wells with probabilities of less than 0.4 should not be prioritized for field testing since they may not have serious leakage. There are over 300,000 hydrocarbon wells (Watson and Bachu, 2009) in Alberta. The described approach can be utilized to most efficiently detect serious fluid leakages for previously non-tested wells in Alberta. Furthermore, the developed methodology described in this paper can be applied for any producing oil and gas field to train an optimum classifier for leakage detection.

5. Conclusion

This paper has developed a framework enabling the prediction of serious fluid leakage from hydrocarbon wells. A wide range of machine learning algorithms were trained based on 22 physical properties from 19730 wells, evaluated by 4933 wells (validation set) and finally tested by 2741 wells (test set). A new imputation technique with low computational cost was developed using a modified LU conditional simulation to overcome the challenge of frequently missing data for selected well properties. Using this approach, the correlations of features were preserved after imputation. Due to random sampling from the distribution of well properties, the associated uncertainty in the imputation of missing data was quantified. Random Forest was found to have the highest performance among all classifiers. The importance of well proprieties to predict serious fluid leakage was quantified based on 100 realizations of Random Forest: each run used different imputed values to incorporate the uncertainty of imputation for feature importance measurement. Month Well Spudded (e.g. the age of the wells) was found to

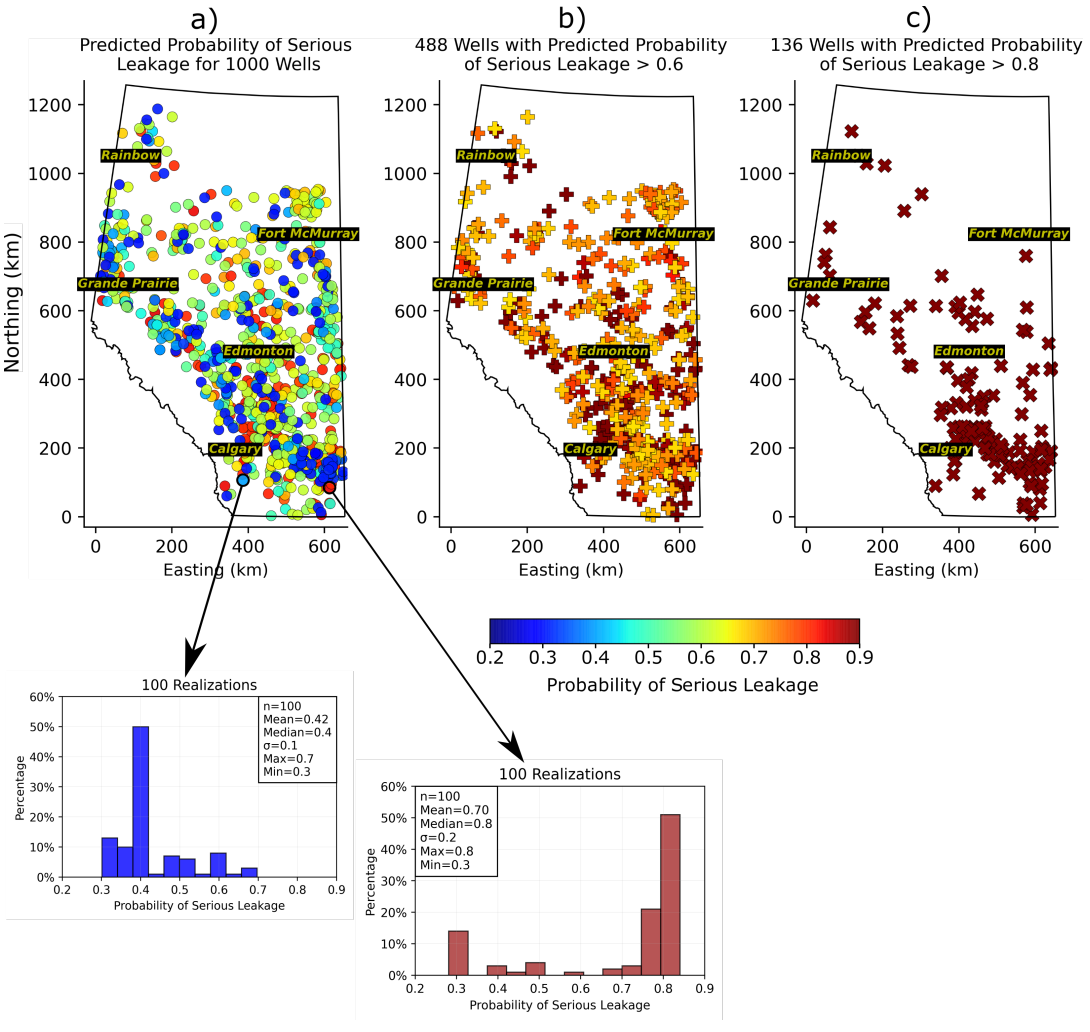

Figure 17. *a) Mean of 100 predicted probabilities of serious fluid leakage for 1000 wells. b) Location map for 488 wells with the probability higher than 0.6 in (a). c) Location map for 136 wells with the probability higher than 0.8 in (a).*

have the highest importance for potential fluid leakage followed by Total Production Month, Surface-Casing Depth, Production Casing Depth, Measured Depth, Cumulative WATER Prod and Cumulative GAS Prod. The well properties do not have a strong positive linear correlation with the target (AER classification). However, the presented results suggest that the oldest and deepest wells with high production are most likely to have serious fluid leakage issues. It is important to note that Deviated Hole and Horizontal Hole have almost no impact on serious fluid leakage. A challenge is the imbalanced number of classes that leads to low sensitivity forcing predictive models to classify more majority class (non-serious leakage). A combination of undersampling and oversampling techniques was considered to adjust the class distribution of training data and feed more balanced data to predictive model. Undersampling was randomly applied for the instances with missing well properties. LU unconditional simulation was utilized for oversampling because of reduced computational cost and quantifying uncertainty. K-fold cross-validation after resampling resulted in severe overoptimism and overfitting. The correct approach was to resample the training folds of each split for K-fold cross-validation to train

separate models. The most reliable class ratio 1.27 was chosen at the point where the metrics have similar performance. The predictions from each trained model on validation and test sets were aggregated by predicting the class that gets the most votes. Model training with more balanced class data reduces false negative estimation and increases the sensitivity. The integration of Random Forest and Logistic Regression as Soft Voting leads to a more reliable prediction. This classifier can be used to detect serious fluid leakage for oil and gas wells in Alberta. The framework developed in this paper can be applied to train a reliable classier for any producing oil and gas field. This should be helpful for the development of cost-effective field testing approaches that result in environmental advantages by identifying and prioritizing amendment of the leakiest wells.

Funding Statement. This research was supported by Canada First Research Excellence Fund (CFREF) and Global Research Initiative in Sustainable Low Carbon Unconventional Resources (GRI).

Competing Interests. The authors declare no competing interests exist.

Data Availability Statement. The test results of surface casing vent flow (SCVF) and gas migration (GM) for energy wells in Alberta, Canada was retrieved from https://www2.aer.ca/t/Production/views/COM-VentFlowGasMigrationReport/VentFlowGasMigrationReport.csv. Well properties for each well was retrieved from geoSCOUT (2022) software.

Author Contributions. Data curation: Mehdi Rezvandehy; Data visualisation: Mehdi Rezvandehy; Conceptualization: Mehdi Rezvandehy; Methodology: Mehdi Rezvandehy; Writing original draft: Mehdi Rezvandehy; Formal analysis: Mehdi Rezvandehy; Programming: Mehdi Rezvandehy; Review and editing: Bernhard Mayer; Funding acquisition: Bernhard Mayer; Supervision: Bernhard Mayer. All authors approved the final submitted draft.

References

Abboud, J., Watson, T., and Ryan, M. (2021). Fugitive methane gas migration around alberta's petroleum wells. *Greenhouse Gases: Science and Technology*, 11(1):37–51.

Alberta Energy Regulator (2003). Interim directive: ID 2003-01. www.aer.ca/documents/ids/id2003-01.pdf. Last checked on July 29, 2022.

Alberta Energy Regulator (2022). Vent Flow/Gas Migration Report. https://www.aer.ca/providing-information/data-and-reports/activity-and-data/general-well-data. Last checked on July 29, 2022.

Bottou, L. (2012). Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer.

Brandt, A. R., Heath, G., Kort, E., O'sullivan, F., Pétron, G., Jordaan, S., Tans, P., Wilcox, J., Gopstein, A., Arent, D., et al. (2014). Methane leaks from north american natural gas systems. *Science*, 343(6172):733–735.

Brownlee, J. (2020). *Imbalanced classification with Python: better metrics, balance skewed classes, cost-sensitive learning*. Machine Learning Mastery.

Brownlee, J. (2021). Retrieved from Machine Learning Mastery. https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/. Last checked on July 29, 2022.

Buuren, S. v. and Groothuis-Oudshoorn, K. (2010). mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, pages 1–68.

Chen, C., Zhou, L., Ji, X., He, G., Dai, Y., and Dang, Y. (2020). An adaptive modeling strategy integrating feature selection and random forest for fluid catalytic cracking processes. *Industrial & Engineering Chemistry Research*.

Cherry, J., Ben-Eli, M., Bharadwaj, L., Chalaturnyk, R., Dusseault, M. B., Goldstein, B., Lacoursière, J.-P., Matthews, R., Mayer, B., Molson, J., et al. (2014). Environmental impacts of shale gas extraction in canada. *The Expert Panel on Harnessing Science and Technology to Understand the Environmental Impacts of Shale Gas Extraction*.

Chollet, F. (2018). *Deep Learning with Python*. Manning Publications Co.

DataWing (2022). Retrieved from . https://datawig.readthedocs.io/en/latest/. Last checked on July 29, 2022.

Davis, M. W. (1987). Production of conditional simulations via the lu triangular decomposition of the covariance matrix. *Mathematical geology*, 19(2):91–98.

Deutsch, C. V. and Journel, A. G. (1998). *GSLIB: Geostatistical Software Library and User's Guide, Second Edition*. Oxford University Press.

Escobar, C. A. and Morales-Menendez, R. (2019). Process-monitoring-for-quality—a model selection criterion for support vector machine. *Procedia Manufacturing*, 34:1010–1017.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. (2018). *Learning from imbalanced data sets*, volume 10. Springer.

Gandhi, R. (2018). Support vector machine—introduction to machine learning algorithms. *Towards Data Science*.

Gariazzo, C., Carlino, G., Silibello, C., Renzi, M., Finardi, S., Pepe, N., Radice, P., Forastiere, F., Michelozzi, P., Viegi, G., et al. (2020). A multi-city air pollution population exposure study: Combined use of chemical-transport and random-forest models with dynamic population data. *Science of The Total Environment*, page 138102.

geoSCOUT (2022). Retrieved from . https://www.geologic.com/products/geoscout/. Last checked on July 29, 2022.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media.

Hashimoto, H., Wang, W., Melton, F. S., Moreno, A. L., Ganguly, S., Michaelis, A. R., and Nemani, R. R. (2019). High-resolution mapping of daily climate variables by aggregating multiple spatial data sets with the random forest algorithm over the conterminous united states. *International Journal of Climatology*, 39(6):2964–2983.

He, H. and Ma, Y. (2013). Imbalanced learning: foundations, algorithms, and applications.

Journel, A. G. and Bitanov, A. (2004). Uncertainty in N/G ratio in early reservoir development. *Journal of Petroleum Science and Engineering*, 44(1):115–130.

Kaggle (2022). Retrieved from . https://www.kaggle.com/ryanholbrook/target-encoding. Last checked on July 29, 2022.

Kang, M., Kanno, C. M., Reid, M. C., Zhang, X., Mauzerall, D. L., Celia, M. A., Chen, Y., and Onstott, T. C. (2014). Direct measurements of methane emissions from abandoned oil and gas wells in pennsylvania. *Proceedings of the National Academy of Sciences*, 111(51):18173–18177.

Khan, K. D. and Deutsch, C. V. (2016). Practical Incorporation of Multivariate Parameter Uncertainty in Geostatistical Resource Modeling. *Natural Resources Research*, 25(1):51–70.

Lassalle, G., Credoz, A., Hédacq, R., Bertoni, G., Dubucq, D., Fabre, S., and Elger, A. (2019). Estimating persistent oil contamination in tropical region using vegetation indices and random forest regression. *Ecotoxicology and environmental safety*, 184:109654.

Lemon, S. C., Roy, J., Clark, M. A., Friedmann, P. D., and Rakowski, W. (2003). Classification and regression tree analysis in public health: methodological review and comparison with logistic regression. *Annals of behavioral medicine*, 26(3):172–181.

Montague, J. A., Pinder, G. F., and Watson, T. L. (2018). Predicting gas migration through existing oil and gas wells. *Environmental Geosciences*, 25(4):121–132.

Ozigis, M. S., Kaduk, J. D., Jarvis, C. H., da Conceição Bispo, P., and Balzter, H. (2020). Detection of oil pollution impacts on vegetation using multifrequency sar, multispectral images with fuzzy forest and random forest methods. *Environmental Pollution*, 256:113360.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Rekha, S., Jeyanthy, P. A., and Devaraj, D. (2019). Multinomial logistic regression for fault type detection in bench mark fault model of wind energy conversion system. In *2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, pages 1–4. IEEE.

Rezvandehy, M. and Deutsch, C. V. (2017). Horizontal variogram inference in the presence of widely spaced well data. *Petroleum Geoscience*, 24(2):219–235.

Rezvandehy, M., Leung, J. Y., Ren, W., Hollands, B., and Pan, G. (2019). An improved workflow for permeability estimation from image logs with uncertainty quantification. *Natural Resources Research*, 28(3):777–811.

Robles-Velasco, A., Cortés, P., Muñuzuri, J., and Onieva, L. (2020). Prediction of pipe failures in water supply networks using logistic regression and support vector classification. *Reliability Engineering & System Safety*, 196:106754.

Santos, M. S., Soares, J. P., Abreu, P. H., Araujo, H., and Santos, J. (2018). Cross-validation for imbalanced datasets: avoiding overoptimistic and overfitting approaches [research frontier]. *ieee ComputatioNal iNtelligeNCe magaziNe*, 13(4):59–76.

Shindell, D. T., Faluvegi, G., Koch, D. M., Schmidt, G. A., Unger, N., and Bauer, S. E. (2009). Improved attribution of climate forcing to emissions. *Science*, 326(5953):716–718.

Watson, T. L. and Bachu, S. (2009). Evaluation of the potential for gas and co2 leakage along wellbores. *SPE Drilling & Completion*, 24(01):115–126.

Zhu, C., Idemudia, C. U., and Feng, W. (2019). Improved logistic regression model for diabetes prediction by integrating pca and k-means techniques. *Informatics in Medicine Unlocked*, 17:100179.