Journal of Petroleum Exploration and Production Technology (2020) 10:2055–2068
https://doi.org/10.1007/s13202-020-00857-w

ORIGINAL PAPER - EXPLORATION ENGINEERING

Drilling stuck pipe classification and mitigation in the Gulf of Suez oil fields using artificial intelligence

Haytham H. Elmousalami[1,2,3] · Mahmoud Elaskary[1,4]

Received: 20 December 2019 / Accepted: 17 February 2020 / Published online: 20 March 2020

Abstract

Developing a reliable classification model for drilling pipe stuck is crucial for decision-makers in the petroleum drilling rig. Artificial intelligence (AI) includes several machine learning (ML) algorithms that are used for efficient predictive analytics, optimization, and decision making. Therefore, a comparison analysis for ML models is required to guide practitioners for the appropriate predictive model. Twelve ML techniques are used for drilling pipe stuck such as artificial neural networks, logistic regression, and ensemble methods such as scalable boosting trees and random forest. The drilling cases of the Gulf of Suez wells are collected as an actual dataset for analyzing the ML performance. The key contribution of the study is to automate pipe stuck classification using ML algorithms and mitigate the pipe stuck cases using the genetic algorithm optimization. Out of 12 AI techniques, the results presented that the most reliable algorithm was extremely randomized trees (extra trees) with 100% classification accuracy based on testing dataset. Moreover, this research presents a public open dataset for the drilled wells at the Gulf of Suez to be used for the future experiments, algorithms' validation, and analysis.

Keywords Artificial intelligence (AI) · Ensemble machine learning · XGBoost · Drilling stuck pipe prediction · Optimization · Genetic algorithm

Abbreviations

AdaBoost	Adaptive boosting
AI	Artificial intelligence
ANNs	Artificial neural networks
CART	Classification and regression trees
DNNs	Deep neural networks
DT	Decision tree
Extra Trees	Extremely randomized trees
GA	Genetic algorithms
kNN	K-nearest neighbors
KW	Kilowatts
MA	Moving average
IoT	Internet of things
ML	Machine learning
GOS	The Gulf of Suez
MRA	Multiple regression analysis
MSE	Mean square error
ReLU	Standard rectified linear unit
RF	Random forest
RFR	Random forest regression
RGF	Regularized greedy forest
RMSE	The root mean squared error
RNN	Recurrent neural network
SVM	Support vector machine
SVR	Support vector regression
XGBoost	Extreme gradient boosting
KPIs	Key performance indicators
CNN	Convolutional neural networks
EC	Evolutionary computing

Introduction

Drilling for oil and gas is one of the riskiest activities on Earth. The drilling pipe stuck issue is one of the most critical drilling problems which costs more than $250 million per

✉ Haytham H. Elmousalami
Haythamelmousalami2014@gmail.com

Mahmoud Elaskary
Mahmoud.Elaskary@gpc.com.eg

[1] General Petroleum Company (GPC), Nasr City, Egypt

[2] Department of Construction and Utilities, Faculty of Engineering, Zagazig University, Zagazig, Egypt

[3] Faculty of Computers and Artificial Intelligence, Cairo University, Giza, Egypt

[4] Faculty of Petroleum and Mining Engineering, Suez University, Suez, Egypt

year. Complications related to stuck pipe can account for nearly half of total well cost, making stuck pipe one of the most expensive problems that can occur during a drilling operation. This problem may reache to the drill string loss or the complete loss of the well (Shadizadeh et al. 2010; Siruvuri et al. 2006). The stuck pipe occurs due to several mechanisms including improper hole cleaning, wellbore stability, poor well trajectory, improper drilling fluid, hole assembly design, and differential sticking forces. The risk of mechanical or differentially stuck pipe can be minimized by adopting drilling variables. Pore pressure rises the probability of the pipe stuck. Moreover, lower mud densities can increase the risk of wellbore instability and mechanical sticking.

Pipe stuck risk can be effectively managed and mitigated based on reliable pipe stuck model. Model types can be divided into three main categories: empirical, physical, and mathematical (Noshi and Schubert 2018). However, empirical and physical models cannot capture high predictive accuracy and generalization. On the other hand, mathematical models statistically need dataset to be developed. In the oil and gas industry, huge dimensions of hourly real-time production data can be measured such as pressure, flow rate, and temperature profiles using sensors and internet of things (IoT) devices on the surface of down hole. Such observed data are known as the big data characterized by volume, velocity, and variety (Mishra and Datta-Gupta 2017). The main motivations to automate stuck classification and mitigation are as follows:

1. To provide a proactive prediction tool that can early predict the stuck occurrence based on the key drilling stuck predictors.
2. To provide a reliable tool that can avoid the stuck cases and optimize the drilling parameters.
3. To present a comprehensive comparison among ML algorithms for pipe stuck prediction.
4. To identify the importance of each predictor for drilling pipe stuck using sensitivity analysis.
5. To present a novel dataset for drilling pipe stuck classification and mitigation in the Gulf of Suez (GOS).

Love (1983) was the first one to use the past data to develop a predictive model for success rate of freeing stuck drill pipe using a trial-and-error method for key predictors' selection. ML can be applied to identify stuck pipe incidents where the predictors have been collected based on historical data, reports of stuck pipe, and published literature. The collected predictors have been ranked to identify the key predictors. After validation and testing processes, the model showed promising results where the proposed model enhanced the describing and monitoring of the drilling data streams (Alshaikh et al. 2019). Using real-time drilling operations, a framework for the early accurate detection of stuck pipe has been developed based on random forests. The model has automated data extraction module and reliable prediction classifier that helps drilling engineers and the rig crew to predict the stuck pipe risk (Magana-Mora et al. 2019). Natural language processing and ML can be developed for the analysis of drilling data. The objective is to improve reservoir management and determine the non-productive time and extract crucial information. The model shows successful performance in the fields in North and South America and fields located in the Middle East (Castiñeira et al. 2018).

ANNs have been used for the stuck drill pipe prediction in Maroon field where the model is capable of producing reliable results (MoradiNezhad et al. 2012). Chamkalani et al. (2013) have proposed a new methodology based on SVM for stuck pipe prediction. ANNS and SVM have been implemented for stuck pipe prediction where both models present accurate result of 83% based on binary classification (Albaiyat 2012). Based on 40 oil wells, multivariate statistics has been conducted for prediction of stuck pipe. Multivariate statistical analysis consisted of regression analysis and discriminate analysis with success rate up to 86% (Shoraka et al. 2011). A convolutional neural network (CNN) approach has been used to predict the stuck occurrence in the Gulf of Mexico. Back-propagation learning rule and sigmoid-type nonlinear activation functions have been used to develop the model. The model presents reliable results for stuck prediction based on the collected data (Siruvuri et al. 2006). This literatures review did not reveal any comparison for different ML techniques designed to prevent the sticking of the drill pipe.

Based on the literature survey, there is no a comprehensive comparison study of the different AI algorithms for the drilling stuck pip prediction. The key objective of this research is evaluating the classification accuracy of different AI models to produce the most accurate classification model. Moreover, this research aims to present a comprehensive performance comparison for AI model to guide the researchers and practitioners during the drilling stuck classification modeling. This research consists of five steps as follows as illustrated in Fig. 1:

1. The past literature has been reviewed to know the past practices for drilling stuck modeling.
2. Real data of cases of the drilling pipe stuck have been gathered. The data have been quantitatively collected based on the site records for each drilling well.
3. The third step includes a model development based on AI models where a total of 12 predictive models have been built.
4. The fourth step is the models' validation to select the most accurate model.
5. The fifth step is to analyze the results and conduct a sensitivity analysis to identify the contribution of the parameters on the pipe stuck.
6. Finally, an optimization system has been incorporated into prediction model to optimize drilling parameters to mitigate stuck and partially stuck cases.

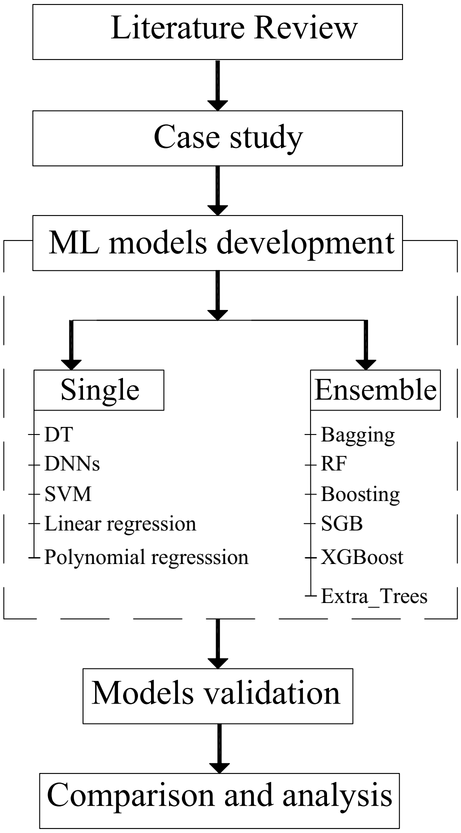

Fig. 1 Research methodology

Application to drilled wells in the Gulf of Suez

The process of data acquisition is the most difficult and critical part of any statistical learning (Elmousalami et al. 2018a, Elmousalami 2020). As shown in Fig. 2, a total number of 103 wells were drilled offshore and onshore during the five-year period from 2010 to 2015 by General Petroleum Company (GPC) and petroleum sector in Egypt. These data were collected using sensors and measuring devices on the drilling rig where these sensors are validated based on quality control and safety procedures before and during the drilling process. Moreover, these data have been handled to the drilling experts and engineers to check its quality and reliability where all outliers and missing data have been removed. The data contained 26 stuck and 77 non-stuck and partially stuck cases. The type of stuck pipe is mechanical pipe sticking due to poor hole cleaning, wellbore collapse, and key-seating. The parameter set includes a total of seven drilling parameters recorded on a daily basis as illustrated in Table 1.

The drilling pipe stuck issue could be a dynamic problem which could exist at different time periods of a drilling project. Thus, using a binary string cannot effectively represent the whole problem. Therefore, the output can be three general groups of data: stuck, partially stuck, and non-stuck. The output probability ranges from 0 to 1 where the range from 0 to 0.4 represents non-stuck case, the range from 0.4 to 0.7 represents partially stuck, and the range from 0.7 to 1 represents the stuck case.

Correlation analysis has been done to identify the key performance indicators (KPIs) as shown in Fig. 3. Scatterplots of all the independent variables with each other were drawn to check the collinearity among the variables. Of the two variables which showed collinearity, the one that showed a weak correlation with the outcome was dropped. This deletion was also based on discussions with the experts, common wisdom, and knowledge about the subject and statistics. The characteristic of the formation along the drilling trajectory has been excluded from the collected features because the formation of the collected dataset has the same characteristic in the Gulf of Suez fields. Moreover, the proposed classification model aims to classify the stuck case based on the least number of the input parameters.

Machine learning methods

ML algorithms are scalable algorithms used for pattern recognition and obtaining useful insight based on the collected data (LeCun et al. 2015, Bishop 2006). AI and ML are general purpose techniques which can be applied for several applications (Elmousalami 2020; Witten et al. 2016). The ML models in this study can be applied in the abroad area of oil and gas industry where modeling methodology can valid for different projects types. ML can be single or ensemble type. Single ML models are such as SVM, DT, and ANN. On the other hand, ensemble ML models are bagging, booting, XGBoost, and random forest. Before training the ML algorithms, the data input values have been normalized using min–max feature scaling (Dodge and Commenges 2006). The normalization process improves the computation for each classifier.

Single AL model

Support vector machines (SVM)

SVMs are supervised learning algorithms that can be used for both classification and regression applications

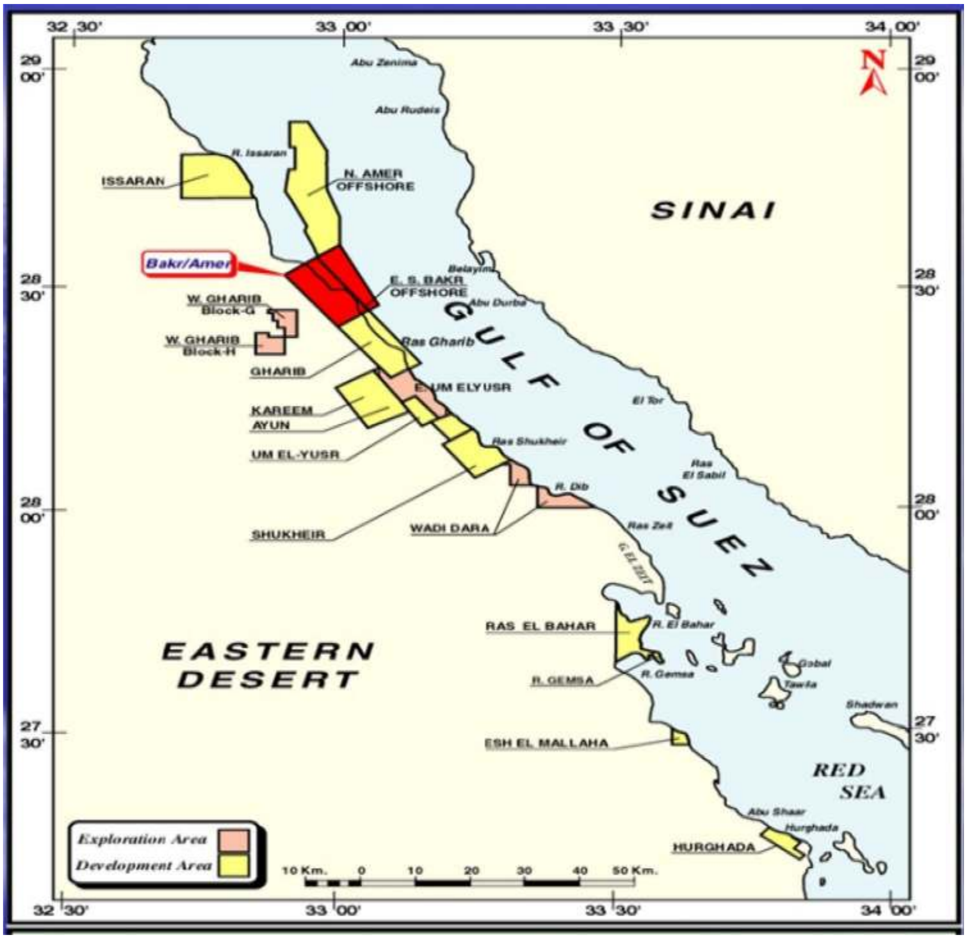

Fig. 2 Oil fields map in the Gulf of Suez

Table 1 The drilling pipe stuck parameters

Notation	Predictor	Predictor description	Unit
P1	Mud pump circulation rate	Mud pump circulation rate produces less stuck probability with good hole cleaning	Gallon per minute(gpm)
P2	Mud type	Mud type can be water-based mud model or oil-based mud model	Binary
P3	Total drilling time	The total duration of drilling operation in the well	Hour
P4	Rate of penetration	Rate of penetration is annular velocity where high rate with bad cleaning gives more stuck probability	Meter/hour
P5	Maximum inclination	Maximum angle of inclination from vertical where more inclination produces more stuck probability	Degree
P6	String rotation	More rate of string rotation produces less stuck case	Revolution per minute (rpm)
P7	Drilled depth	The actual measured depth during the well drilling process	Meter
O	The model output	The output is probability ranges from 0 to 1 where the range from 0 to 0.4 represents non-stuck case, the range from 0.4 to 0.7 represents partially stuck, and the range from 0.7 to 1 represents stuck case	Multi-classes

(Elmousalami 2019a, 2020). SVM optimizes hyperplanes distance and the margin as shown in Fig. 4. Hyperplane distance can be maximized based on two classes of boundaries using the following equation (Vapnik 1979):

$$\text{Linear SVM} = \begin{cases} W \cdot X_i + b \geq 1, & \text{if } y_i \geq 0 \\ W \cdot X_i + b < -1, & \text{if } y_i < 0 \end{cases} \tag{1}$$

For $i = 1,2,3,\ldots,m$, a positive slack variable (ξ) is added for handling the nonlinearity as displayed in Eq. (2):

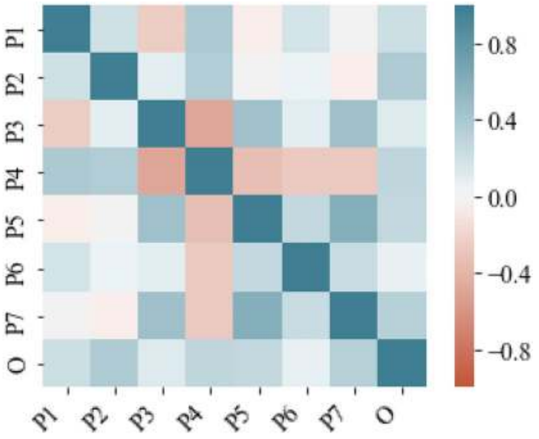

Fig. 3 The predictors correlation heat map

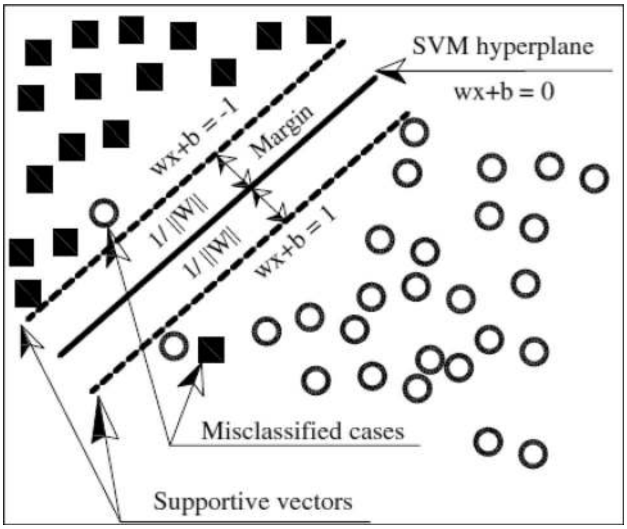

Fig. 4 Linear support vector machine

$$y_i\left(W.X_i + b\right) \geq 0 - \xi, \quad i = 1, 2, 3, \ldots \ldots m \tag{2}$$

Accordingly, the objective function will be as shown in Eq. (3):

$$\text{Min} \sum_{i=0}^{i=m} \frac{1}{2} w \cdot w^T + C \sum_{i=0}^{i=m} \xi_i \tag{3}$$

Decision trees (DTs)

Decision tree (DT) is a statistical learning algorithm that is dividing the collected data into logical rules hierarchically (Elmousalami 2019b; Breiman et al. 1984) as shown in Fig. 5. Splitting algorithm is repetitively used to formulate each node of the tree. Classification and regression trees (CART) and C4.5/C5.0 algorithms are the most common tree models used in the research and practical community. This model is applied for both classification and continues prediction applications (Curram and Mingers 1994). DT algorithm can interpret data and feature importance based on the generated logical statement for each tree node. However, DT is not a robust and stable algorithm against noisy and missing data (Perner et al. 2001).

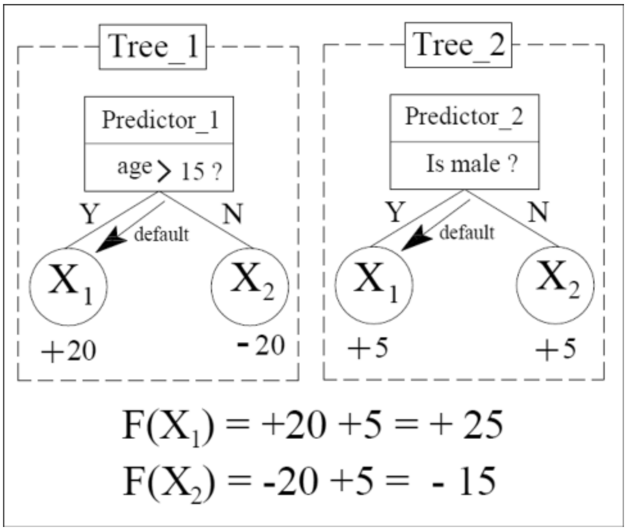

Fig. 5 Additive function concept

Logistic regression

Logistic regression (logit regression) is a predictive regression analysis which is appropriate for the dichotomous (binary) dependent variable (Hosmer et al. 2013). Logistic regression is used to explain data and to describe the relationship between one dependent binary variable and one or more independent variables. No outliers exist in the data, and there should be no high correlations (multicollinearity) among the predictors (Tabachnick and Fidell 2013). Mathematically, logistic regression can be defined as follows:

$$P = \frac{1}{1 + e^{-(a+bX)}} \tag{4}$$

where P is the classification probability, e is the base of the natural logarithm and (a) and (b) are the parameters of the model. Adding more predictors to the model can result in overfitting, which reduces the model generalizability and increases the model complexity.

K-nearest neighbor classifier (KNN)

KNN algorithm is building a nonparametric classifier (Altman 1992, Weinberger et al. 2006). KNN is an instance-based learning used for classification or regression applications. The object is classified by a majority vote of its neighbors in the training set. If $K = 1$, then the case is simply assigned to the class of its nearest neighbor. Many distance functions can be applied to measure the similarity among the instances such as Euclidian, Manhattan, and Minkowski (Singh et al. 2013).

Gaussian Naive Bayes algorithm

Gaussian Naive Bayes classifier is an algorithm for classification technique which assumes independency among predictors (Patil and Sherekar 2013). Naive Bayes is useful for very large datasets and known to outperform even highly sophisticated classification methods. Bayes theorem computes posterior probability $P\ (c|x)$ from $P(c)$, $P(x)$, and $P(x|c)$ as shown in Eq. 5:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{5}$$

where $P(c|x)$ represents the posterior probability of the target class (c, target) given the input predictors (x, attributes); $P(c)$ represents the prior probability of the target class; $P(x|c)$ is the likelihood which is the probability of predictor given class; $P(x)$ is the prior probability of predictor. Naive Bayes algorithm works by computing likelihood and probabilities for each class. Naive Bayesian formula computes the posterior probability for each class where the highest posterior probability class is the prediction outcome (Kohavi 1996).

Artificial neural networks (ANNs)

ANNs are computational systems biologically inspired by the design of natural neural networks (NNN). Key abilities of ANNs are generalization, categorization, prediction, and association (LeCun et al. 2015). ANNs have high ability to dynamically figure out the relationships and patterns between the objects and subjects of knowledge based on nonlinear functions (Elmousalami et al. 2018b). The feed-forward network such as multilayer perceptron networks (MLPs) applies the input vector (x), a weight matrix (W), an output vector (Y), and a bias vector (b). It can be formulated as Eq. (6) and Fig. 6.

$$Y = f(W \cdot x + b) \tag{6}$$

where f (.) includes a nonlinear activation function.

Ensemble methods and fusion learning

Ensemble methods and fusion learning are data mining techniques to fuse several ML algorithms such as ANNs, DT, and SVM to boost the overall performance and accuracy (Hansen and Salamon 1990). Ensemble methods can merge several ML algorithms such as DT, SVM, or ANNs. Each single ML used in the ensemble method is called a base learner where the final decision is taken by the ensemble model. K is an additive function to predict the final output as given in Eq. (7):

$$\hat{y}_i = \sum_{k=1}^{K} f_k(X_i), \quad f_k \in F \tag{7}$$

where $\hat{y}_i$ represents the predicted dependent variable. Each f_k is an independent tree structure and leaf weights $w{\cdot}x_i$ are

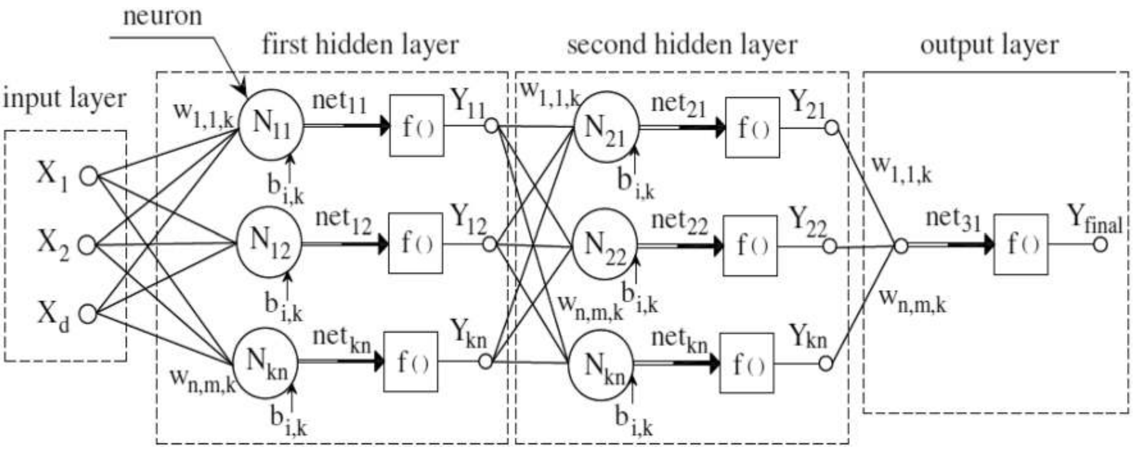

Fig. 6 Multilayer perceptron network (MLP)

the independent variables. F is the regression trees space. Ensemble methods include several methods such as bagging, voting, stacking, and boosting (Elmousalami 2019c; 2020). The ensemble learning models deal effectively with the issues of complex data structures, high-dimensional data, and small sample size (Breiman 1996; Dietterich 2000; Kuncheva 2004).

Breiman (1999) proposed bagging technique as shown in Fig. 7a. Bagging applies bootstrap aggregating to train several base learners for variance reduction (Breiman 1996). Bagging draws groups of training data with replacement to train each base learner. Random forest (RF) is a special case of the bagging ensemble learning techniques. RF draws bootstrap subsamples to randomly create a forest of trees as shown in Fig. 7b (Breiman 2001). Using adaptive resampling, boosting method can be established for enhancing the performance of weak base learners (Schapire 1990) as shown in Fig. 7c. Adaptive boosting algorithm (AdaBoost) has been proposed by Schapire et al. (1998). Serially, AdaBoost draws the data for each base learner using adaptive weights for all instances. These adaptive weights guide the algorithm to minimize the prediction error and misclassified cases (Bauer and Kohavi 1999).

Extreme gradient boosting (XGBoost) is a gradient boosting tree algorithm. XGBoost uses parallel computing to learn faster and diminish computational complexity (Chen and Guestrin 2016). The following equation uses regularization term to the additive tree model to avoid overfitting of the model as shown in the following equation:

$$L(\phi) = (x + a)^n = \sum_{k=0}^{n} l\left(\hat{y}_i, y_i\right) + \sum_{k=1}^{K} \Omega\left(f_k\right), \quad \text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda \left\| w^2 \right\| \tag{8}$$

where L represents a differentiable convex cost function (Friedman 2001). Moreover, XGBoost assigns a default direction into its tree branches to handle missing data in the training dataset. Therefore, no effort is required to clean the training data. Stochastic gradient boosting (SGB) is a boosting bagging hybrid model (Breiman 1996). SGB iteratively improves the model's performance by injecting randomization into the selected data subsets to enhance fitting accuracy and computational cost (Schapire et al. 1998).

Extremely randomized trees algorithm (extra trees) is tree-based ensemble method which can be applied for both supervised classification and regression cases (Vert 2004). Extra trees algorithm essentially randomizes both cut-point choice and attribute during tree node splitting. The key advantage of extra trees algorithm is the tree structure randomization which enables the algorithm to be tuned for the optimal parameters' selection. Moreover, extra trees have high computational efficiency based on a bias/variance analysis (Vert 2004).

In ML, many parameters are assessed and improved during the learning process. By contrast, a hyperparameter is a variable whose value is set before training. The performance of the ML algorithms depends on the tuning parameter. The objective of hyperparameters optimization is to maximize the predictive accuracy by finding the optimal hyperparameters for each ML algorithm. Manual search, random search, grid search, Bayesian optimization, and evolutionary optimization are the most common techniques used for ML hyperparameters optimization. However, manual search random search and grid search are brute force techniques which needs unlimited trails to cover all possible combinations to get the optimal hyperparameters (Bergstra et al. 2011). On the other hand, Bayesian optimization and evolutionary optimization are automatic hyperparameters optimization

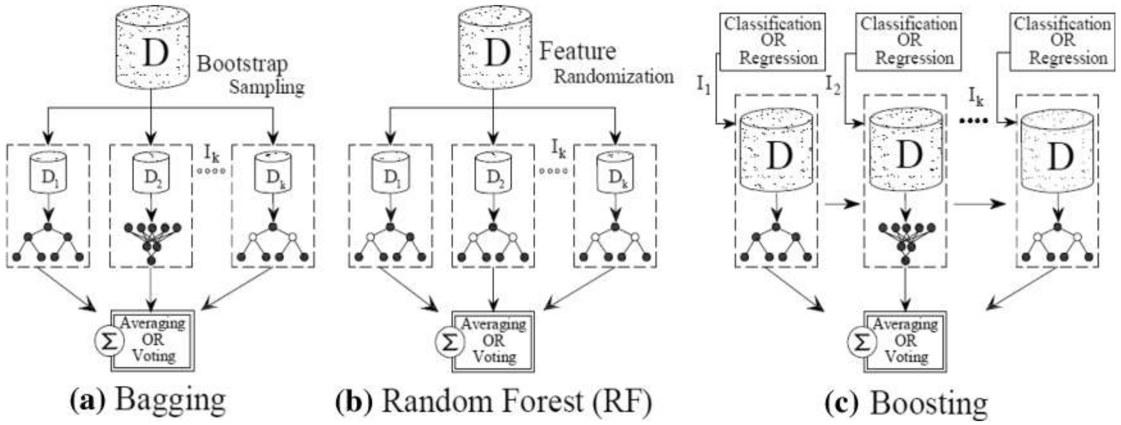

Fig. 7 **a** Bagging, **b** RF, and **c** boosting which selects the optimal parameter with less human intervention (Shahriari et al. 2015). Moreover, these techniques can solve the curse of dimensionality. Therefore, this study used genetic algorithms to select the global optimal setting for each model before training stage. Starting with a random population, the iterative process of selecting the strongest and producing the next generation will stop once the best-known solution is satisfactory for the user. Objective function is defined as minimization of classification accuracy (Acc in Eq. 10) for each classifier. Classification accuracy (Acc) computes the ratio between the correctly classified instances and the total number of samples as in Eq. (9):

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{9}$$

where TP is the true positive; FP the false positive; TN the true negative; FN the false negative. The domain space is defined as the range of the all possible hyperparameters for each algorithm as shown in Table 2. This study applied a decision tree algorithm as based learners for all ensemble methods. Accordingly, the proposed ensemble models and decision tree have been classified as tree-based models which have the same parameter as shown in Table 2. The maximum number of iterations to be run is defined as 10,000 iterations.

To compare machine learning algorithms, the identical blind validating cases are used to test the algorithms performance. The datasets have been divided into training set (80%) and validation set (20%), where the validation cases are excluded from the training data to ensure the generalization capability. This study applied tenfold cross-validation (10 CV) approach using the validation data set (20% of the whole data set). The K-fold cross-validation boosts the performance of validation process using a limited dataset.

Classification accuracy (Acc), specificity, and sensitivity are scalar measures for the classification performance. Moreover, receiver operating characteristic (ROC) is graphical measure for classification algorithm (Tharwat 2018). The receiver operating characteristic (ROC) curve is a two-dimensional graph in which the true positive rate (TPR) is represented in the y-axis and false positive rate (FPR) is in the x-axis (Sokolova et al. 2006a, b; Zou 2002):

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{10}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} \tag{11}$$

Based on ROC, the perfect classification happens when the classifier curve possesses through the upper left corner

Table 2 Optimal hyperparameters settings

Model class	Optimal hyperparameters settings
Tree-based models	Minimum number of samples for node splitting: 2 sample Minimum number of samples be stored in a tree leaf: 1 sample Maximum number of features for splitting a node: log2 (number of the features) Maximum number of levels allowed in each tree: expanded until all leaves are pure or until all leaves contain less than minimum number of samples for node splitting. The function to measure the quality of a split: the mean squared error (MSE) Maximum number of trees in the ensemble: 20 trees.
ANNs	Number of hidden layers: 3 layers Number of neurons in each layer: 10 neurons/layer Activation function: rectified linear unit function (Relu) Weight optimization: stochastic gradient-based optimizer
Polynomial regression	Degree polynomial features: second order
Logistic Regression	Intercept and coefficients weights of the input features
SVM	Penalty parameter = 1 Tolerance for stopping criteria = $1e^{-4}$ Epsilon parameter in the epsilon-insensitive loss function = 0.00 Kernel: radial base function (RBF)
Naïve Bayes	Prior probabilities of the classes = None Variance smoothing = 10-9
KNNs	Number of neighbors = 5 neighbors Weights = ’uniform’ Leaf size = 30 Algorithm used to compute the nearest neighbors: Ball Tree Algorithm

of the graph. At such a corner point, all positive and negative samples are correctly classified. Therefore, the steepest curve has better performance. Area under the ROC curve (AUC) is applied to compare different classifiers in the ROC curve based on the scalar value. The AUC score is ranging between zero and one. Therefore, no realistic classifier has an AUC score lower than 0.5 (Metz 1978; Bradley 1997). ROC curves for each classifier must be potted to show the performance of classifier against different thresholds. In addition, the cost function is represented in the following equation:

$$\text{Error} = \frac{1}{N}\sum_{i=1}^{N} L\left\{\hat{Y}^{(i)} \neq Y^{(i)}\right\} \tag{12}$$

where Error: TN + FP, N: the number of cases, $\hat{Y}^{(i)}$ is the predicted value, $Y^{(i)}$ is the actual value, and L is the loss function. In the current study, weights are added to the error formula (Eq. 10) to emphasize the weight of the true negative cases where the case is stuck in the reality and the model predicted it as a non-stuck case. To handle such case, Eq. 11 is added to Eq. 10 to formulate Eq. 13:

$$W^{(i)} = \begin{cases} 1 & \text{if} \quad X^{(i)} \text{is nonstuck case} \\ 10 & \text{if} \quad X^{(i)} \text{ is stuck case} \end{cases} \tag{13}$$

where $X^{(i)}$ is the actual classification of the oil well stuck case.

$$\text{Modified Error} = \frac{1}{\sum W^{(i)}}\sum_{i=1}^{N} W^{(i)} L\left\{\hat{Y}^{(i)} \neq Y^{(i)}\right\} \tag{14}$$

Results and discussion

In engineering practice, the operator and decision-makers have to select a mathematical model regarding accuracy, the easiness of implementation, generalization, and uncertainty. The scope of the current study focused on the accuracy and the generalization ability of the developed algorithms. Based on validation dataset, accuracy (Acc), and AUC, 12 classifiers were validated as displayed in Table 3. The classifiers have been descendingly sorted from $C1$ to $C12$ based on AUC as shown in Table 3. This study presents that extra trees classifier ($C1$) is the most accurate for pipe stuck classification. Based on ROC comparison as shown in Fig. 8, extra trees classifier was in the first place. Based on the test data set, extra trees classifier ($C1$) yielded an overall correct classification of 100%, which means that 100% of the time this model was able to identify correctly the wells belonging to the given predictors. DT, RF, XGBoost, and AdaBoost produce RF produced 0.83 and 0.74 for AUC and accuracy, respectively. Ensemble methods such as [extra trees ($C1$), bagging ($C8$), RF ($C3$), AdaBoost ($C5$), and SGB ($C9$)] have produced a high acceptable performance.

Table 3 The classifiers' accuracy

Notation	Model	Algorithm type	AUC	Accuracy
C1	Extra Trees	Ensemble	1.000	1.000
C2	DT	Single	0.833	0.744
C3	RF	Ensemble	0.833	0.744
C4	XGB	Ensemble	0.833	0.744
C5	AdaBoost	Ensemble	0.833	0.744
C6	Naive Bayes	Single	0.817	0.501
C7	SVM	Single	0.808	0.610
C8	Bagging	Ensemble	0.808	0.610
C9	SGB	Ensemble	0.808	0.610
C10	ANNs	Single	0.592	0.287
C11	KNN	Single	0.575	0.265
C12	Logistic	Single	0.500	0.231

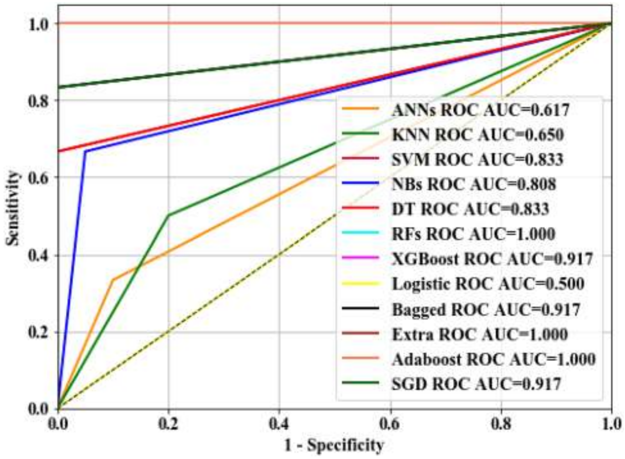

Fig. 8 Average ROC curve for different classifiers

High-dimensional data can be effectively handled using ensemble machine learning. In addition, ensemble machine learning solves small sample size and complex data structures problems (Breiman 1996, Schapire et al. 1998). On the other hand, ensemble ML increases the model complexity (Kuncheva 2004). Accordingly, noisy data can be effectively computed by random forests algorithm than decision tree algorithm (Breiman 1996; Dietterich 2000). However, the RF algorithm is unable to interpret the importance of features or the mechanism of producing the results. On the other hand, ANNs, KNN, and logistic regression produced the least performance based on AUC of 0.592, 0.575, and 0.500, respectively.

DT presents an alternative to the black box existing in ANNs based on formulating logic statements (Perner et al. 2001). Furthermore, splitting procedure of DT can compute the high-dimensional data (Prasad et al. 2006). On the other hand, DT produced poor performance for noisy, nonlinear data or time series data (Curram and Mingers 1994). Therefore, tree-based models and ensemble models produce super performance than single algorithms. DT (CART) is inherently used as a based learner for the ensemble methods. Naive Bayes, SVM, bagging, and SGB produced a moderate accuracy where AUC ranged 0.817 to 0.808 and accuracy ranged from 0.501 to 0.61. Table 4 summarizes the limitations and strengths of each classifier. Table 4 guides the researchers and drilling engineers to select the appropriate ML model based on the algorithms' characteristics.

Classifiers computational cost

The prediction accuracy should not be only the evaluation criterion for selecting the optimal ML algorithms. The computational costs (e.g., memory usage and computational time) of the algorithms are also significant criteria during the data processing. Figure 9 illustrates the computational time of the twelve developed algorithms. All models showed an acceptable computational time where the highest time was consumed by logistic regression and KNN algorithms of 192 s and 184 s, respectively. Conversely, XGBoost was the fastest algorithm. On the other hand, Fig. 9 shows that extra trees and DT consumed high memory of 205 and 197 MBs, respectively. ANNs and bagging DT consumed the least memory for classification.

Accordingly, using ensemble algorithms require more computational resources such as extra trees, AdaBoost, and bagging. Therefore, the memory usages that were used by all the ML algorithms were acceptable as at least 4 GB RAM memory. As a result, XGBoost, RFR, and DNNs were the most efficient algorithms based on the computational cost criterion. However, the computational cost (time and memory consumed) of the ML algorithms would exponentially increase with increasing data dimensions such as data features or data size.

A sensitivity analysis for the predictors was done to evaluate the impact of each predictor on the model's performance. The F-score is the harmonic average of the precision and recall, where an F-score reaches its best value at 1 (perfect precision and recall) and worst at 0 (Sokolova et al. 2006a). Moreover, F-score calculates how many times this variable is split. Different ML model would have a different interpolation regarding the input parameters sensitivity. Accordingly, the sensitivity analysis has been done for the most accurate classifier [extra trees ($C1$)].

As illustrated in Fig. 10, the sensitivity analysis indicated that drilled depth (P_7) had the highest impact on the output (drilling pipe stuck). String rotation (P_6) and maximum

Table 4 Algorithms comparison

Algorithm	Strengths	Weaknesses	Interpretation	Uncertainty	Missing values and noisy data
C1	Handing data randomness	Black box nature and sufficient data	No	No	Yes
C2	Working on both linear and nonlinear data, and producing logical expressions	Poor results on too small datasets, overfitting can easily occur	Yes	No	No
C3	Accurate and high performance on many problems including nonlinear	No interpretability, need to choose the number of trees	No	No	Yes
C4	High scalability, handing missing values, high accuracy, low computational cost	No uncertainty and interpretation	No	No	Yes
C5	High scalability, and high adaptability	Depends on other algorithms performance	No	No	Yes
C6	high accuracy and fast algorithm for classification problem	No uncertainty and interpretation	No	No	No
C7	Easily adaptable, works very well on nonlinear problems, not biased by outliers	Compulsory to apply feature scaling, more difficult to understand	No	No	No
C8	Providing higher performance than a single algorithm	Depending on other algorithms performance	No	No	Yes
C9	Handing difficult examples	Highly sensitive to noisy data	No	No	Yes
C10	Works on small size of dataset	Linear assumptions	Yes	No	No
C11	simple and fast algorithm for classification problem	selecting the optimal number of clustering point (k)	No	No	No
C12	Capturing complex patterns, processing big data and high-performance computing	Sufficient training data and high cost computation	No	No	No

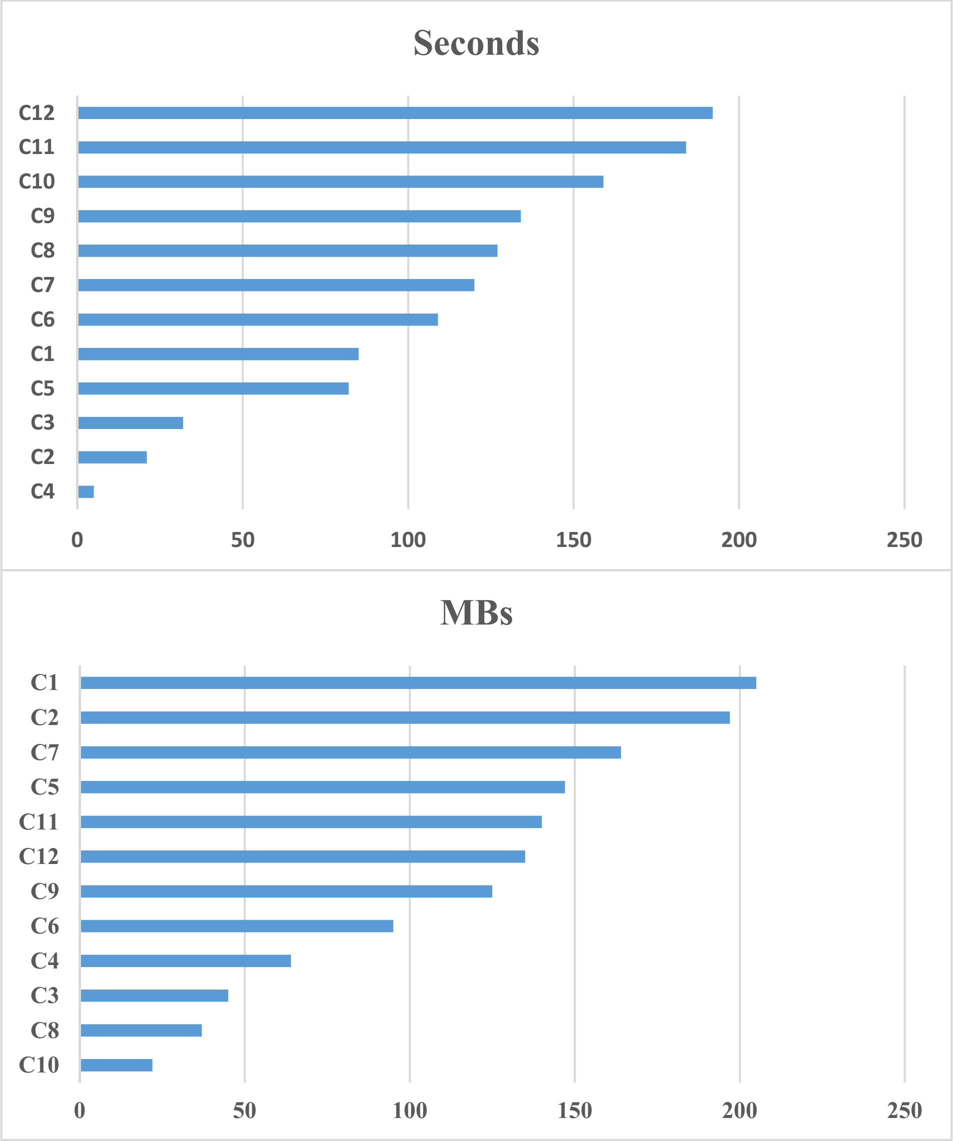

Fig. 9 Computational speed and memory for each classifier

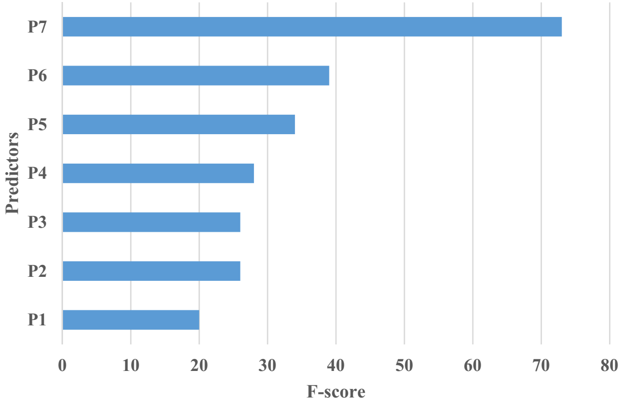

Fig. 10 Sensitivity analysis

inclination (P_5) approximately had the same impact on the output. Similarly, rate of penetration, total drilling time, and mud type had the same impact on the output. The engineering and scientific insights that can be drawn from the sensitivity analysis are as follows:

1. All seven input parameters that are mentioned in Table 1 have significant impact on the pipe stuck classification.
2. Drilled depth (P_7) is the key classifier for stuck cases identification where more drilled depth means more stuck probability percentage.
3. String rotation (P_6) comes in the second place impacting on the stuck probability. Therefore, drilling engineers must accurately calculate the suitable string rotation.
4. Maximum inclination (P_5), rate of penetration, total drilling time, and mud type have approximately the same impact on pipe stuck classification.

5. Mud pump circulation rate had the least impact on the output.

Drilling stuck pipe mitigation module

The drilling pipe stuck issue could easily exist for various reasons in the field applications. Unless the model could provide an effective way to design the drilling project and avoid the issue, predicting whether pipe sticking would happen or not has very little value for field operation. Therefore, once the well condition had been classified as stuck or partially stuck case, the optimization system is needed to determine the optimal values of the seven input parameters. As a result, an optimization system has been incorporated into optimal classification algorithm [extra trees model (C1)] to convert the seven input parameters form stuck or partially stuck case into a non-stuck case. The optimization system has used the genetic algorithm (GA) to optimize the seven input parameters as shown in Fig. 11.

The concept of evolutionary computing (EC) is based on the Darwin's theory: survival for the fittest (Darwin 1859). Genetic algorithm (GA) is a branch of EC applied for optimization and searching applications (Holland 1975; Siddique and Adeli 2013). A chromosome can be represented as a vector (C) consisting of (n) genes denoted by (ci) as follows: $C = \{c1, c2, c3\ldots$ ci}. Each chromosome (C) represents a point in the n-dimensional search space (Elmousalami 2020). In the current case study, the chromosomes represent the seven input parameters. Each chromosome consists of seven genes, where seven genes represent the well drilling parameters (P_1, P_2, P_3, P_4, P_5, P_6, P_7), respectively, as shown in Table 1. Each gene consists of one of the membership functions (MF$_i$) where (I) is ranging through the boundary condition for each variable (P_1: P_7). The number of chromosomes (initial population) is set as 10 chromosomes, and the number of generations is determined to be 10,000 generations. Crossover probability and mutation probability are set to be 0.7 and 0.03, respectively. Accordingly, a group of the initial population of chromosomes have been identified to be evaluated through fitness function.

Fitness function (F) is the function that evaluates the quality of the possible solutions. Crossover and mutation processes are used for developing new offspring generations. The objective is to minimize the stuck probability to be in the range of non-stuck case [0,0.4]. Therefore, the objective function of the GA is the minimization of the stuck probability by optimizing the seven input parameters to reach the characteristics of a non-stuck well. The fitness function can be formulated as Eq. (15) where the objective is to minimize the fitness function as follows:

$$F = \text{Minmization}\left(\hat{y}_i\right) \tag{15}$$

where (F) is a fitness function and $\hat{y}_i$ is the predicted classification based on extra trees model (stuck probability).

To maintain the variables within the reasonable limits, the seven input parameters have been constrained to defined boundary for each parameter. The boundary constraints have ranged for the minimum and maximum values of each parameter. Moreover, functional constraints have been added based on design criteria such as the summation of the solids % and water % not exceeding 100%. However, relatively

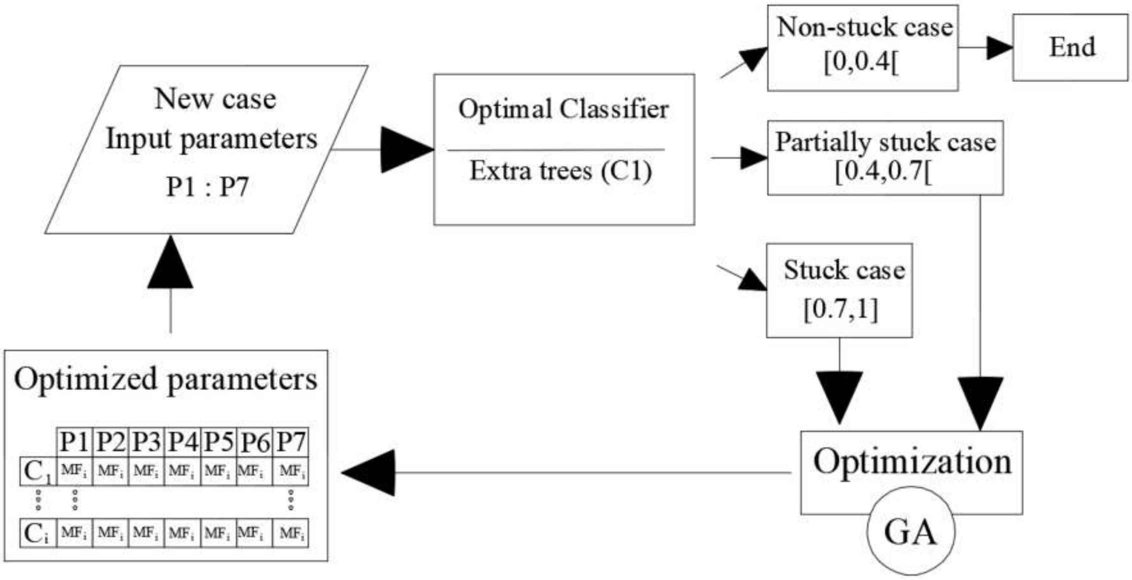

Fig. 11 Stuck mitigation system

high degree of judgment is required to logically select any combination of the seven parameters for drilling process.

Conclusion

Complications of the stuck pipe can account for approximately half of total well cost, making stuck pipe one of the most expensive problems that can occur during a drilling operation (Muqeem et al. 2012). Therefore, the key contribution of the study is to automate the classification and the mitigation of the drilling pipe stuck for the drilled wells at the Gulf of Suez (GOS). Out of 12 machine leaning algorithms, the results presented that the most reliable algorithm was the extremely randomized trees (extra trees) with 100% classification accuracy based on testing dataset. On the other hand, genetic algorithm can optimize the drilling parameters to mitigate the risk of drilling pipe stuck.

The methodology addressed in this study enables the oil and gas drilling industry in GOS to evaluate the risk of stuck pipe occurrence before the well drilling procedure. A comprehensive comparison of ML algorithms has been provided for drilling piping stuck prediction. More data mean more generalization of the trained algorithms. The key limitation of this study is the size of the collected data. However, the collected dataset is sufficient to train the classifiers and to avoid the overfitting problem. Therefore, the future research is to apply this research framework to different datasets in oil fields. The future work will rely on deep learning where deep learning is a powerful tool for pattern recognition. The big data of the drilling projects will be modeled using deep learning algorithms such as deep neural networks and convolutional neural networks.

Acknowledgements I would like to thank drilling engineer Mahmoud M. Elaskary (senior drilling engineer at General Petroleum Company) for his support and his effort for collecting the dataset of the current study.

References

Albaiyat I (2012) Implementing artificial neural networks and support vector machines in stuck pipe prediction (Doctoral dissertation)

Alshaikh A, Magana-Mora A, Gharbi SA, Al-Yami A (2019) Machine learning for detecting stuck pipe incidents: data analytics and models evaluation. In: International petroleum technology conference

Altman NS (1992) An introduction to kernel and nearest-neighbor nonparametric regression. Am Stat 46(3):175–185

Bauer E, Kohavi R (1999) An empirical comparison of voting classification algorithms: bagging, boosting, and variants. Mach Learn 36(1–2):105–139

Bergstra JS, Bardenet R, Bengio Y, Kégl B (2011) Algorithms for hyper-parameter optimization. In: Jordan MI, Lecun Y, Solla SA (eds) Advances in neural information processing systems. MIT Press, Cambridge, pp 2546–2554

Bishop CM (2006) Pattern recognition and machine learning. Springer, New York, pp 1–58

Bradley AP (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn 30(7):1145–1159

Breiman L (1996) Bagging predictors. Mach Learn 26:123–140

Breiman L (1999) Pasting small votes for classification in large databases and on-line. Mach Learn 36(1–2):85–103

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Breiman L, Friedman JH, Olshen R, Stone C (1984) Classification and regression trees. Routledge, Wadsworth

Castiñeira D, Toronyi R, Saleri N (2018) Machine learning and natural language processing for automated analysis of drilling and completion data. In: SPE Kingdom of Saudi Arabia annual technical symposium and exhibition. Society of Petroleum Engineers

Chamkalani A, Pordel Shahri M, Poordad S (2013) Support vector machine model: a new methodology for stuck pipe prediction. In: SPE unconventional gas conference and exhibition 2013 Jan 28. Society of Petroleum Engineers

Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, pp 785–794

Curram SP, Mingers J (1994) Neural networks, decision tree induction and discriminant analysis: an empirical comparison. J Oper Res Soc 45(4):440–450

Darwin C (1859) The origin of species by means of natural selection or the preservation of favoured races in the struggle for life, Mentor Reprint 1958, New York

Dietterich TG (2000) An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Mach Learn 40(2):139–157

Dodge Y, Commenges D (eds) (2006) The Oxford dictionary of statistical terms. Oxford University Press, Oxford

Elmousalami HH (2019a) Intelligent methodology for project conceptual cost prediction. Heliyon 5(5):e01625

Elmousalami HH (2019b) Prediction of construction cost for field canals improvement projects in Egypt. arXiv preprint arXiv:1905.11804

Elmousalami HH (2019c) Comparison of artificial intelligence techniques for project conceptual cost prediction. arXiv preprint arXiv:1909.11637

Elmousalami HH (2020) Artificial intelligence and parametric construction cost estimate modeling: state-of-the-art review. J Constr Eng Manag 146(1):03119008

Elmousalami HH, Elyamany AH, Ibrahim AH (2018a) Evaluation of cost drivers for field canals improvement projects. Water Resour Manag 32:53–65

Elmousalami HH, Elyamany AH, Ibrahim AH (2018b) Predicting conceptual cost for field canal improvement projects. J Const En Manag 144(11):04018102

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232

Hansen LK, Salamon P (1990) Neural network ensembles. IEEE Trans Pattern Anal Mach Intell 12(10):993–1001

Holland JH (1975) Adaptation in natural and artificial systems. University Michigan Press, Ann Arbor

Hosmer DW, Lemeshow S, Sturdivant RX (2013) Applied logistic regression, vol 398. Wiley, Hoboken

Kohavi R (1996) Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In: Kdd, vol 96, pp 202–207

Kuncheva LI (2004) Combining pattern classifiers: methods and algorithms. Wiley, Hoboken

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436

Love T (1983) Stickiness factor: a new way of looking at stuck pipe. Oil and Gas Journal (Oct. 3, 1983) 87

Magana-Mora A, Gharbi S, Alshaikh A, Al-Yami A (2019) AccuPipePred: a framework for the accurate and early detection of stuck pipe for real-time drilling operations. In: SPE Middle East oil and gas show and conference. Society of Petroleum Engineers

Metz CE (1978) Basic principles of ROC analysis. In: Seminars in nuclear medicine. WB Saunders, vol 8, no 4, pp. 283–298

Mishra S, Datta-Gupta A (2017) Applied statistical modeling and data analytics: a practical guide for the petroleum geosciences. Elsevier, Amsterdam

MoradiNezhad M, Ashoori S, Hooshmand P, Mirzaee M (2012) Stuck drill pipe prediction with networks neural in maroon field. J Basic Appl Sci Res 2(6):5570–5575

Muqeem MA, Weekse AE, Al-Hajji AA (2012) Stuck pipe best practices-a challenging approach to reducing stuck pipe costs. In: SPE Saudi Arabia section technical symposium and exhibition. Society of Petroleum Engineers

Noshi CI, Schubert JJ (2018) The role of machine learning in drilling operations; a review. In: SPE/AAPG eastern regional meeting. Society of Petroleum Engineers

Patil TR, Sherekar SS (2013) Performance analysis of Naive Bayes and J48 classification algorithm for data classification. Int J Comput Sci Appl 6(2):256–261

Perner P, Zscherpel U, Jacobsen C (2001) A comparison between neural networks and decision trees based on data from industrial radiographic testing. Pattern Recogn Lett 22(1):47–54

Prasad AM, Iverson LR, Liaw A (2006) Newer classification and regression tree techniques: bagging and random forests for ecological prediction. Ecosystems 9(2):181–199

Schapire RE (1990) The strength of weak learnability. Mach Learn 5(2):197–227

Schapire RE, Freund Y, Bartlett P, Lee WS (1998) Boosting the margin: a new explanation for the effectiveness of voting methods. Ann Stat 26(5):1651–1686

Shadizadeh SR, Karimi F, Zoveidavianpoor M (2010) Drilling stuck pipe prediction in iranian oil fields: an artificial neural network approach. Petroleum University of technology, Abadan

Shahriari B, Swersky K, Wang Z, Adams RP, De Freitas N (2015) Taking the human out of the loop: a review of Bayesian optimization. Proc IEEE 104(1):148–175

Shoraka SAR, Shadizadeh SR, Shahri MP (2011) Prediction of stuck pipe in Iranian south oil fields using multivariate statistical analysis. In: Nigeria annual international conference and exhibition. Society of Petroleum Engineers

Siddique N, Adeli H (2013) Computational intelligence: synergies of fuzzy logic, neural networks and evolutionary computing. Wiley, Chichester

Singh A, Yadav A, Rana A (2013) K-means with three different distance metrics. Int J Comput Appl 67(10):13–17

Siruvuri C, Nagarakanti S, Samuel R (2006) Stuck pipe prediction and avoidance: A convolutional neural network approach. In: IADC/SPE drilling conference. Society of Petroleum Engineers

Sokolova M, Japkowicz N, Szpakowicz S (2006) Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Australasian joint conference on artificial intelligenceSpringer, Berlin, pp 1015–1021

Sokolova M, Japkowicz N, Szpakowicz S (2006) Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Australasian joint conference on artificial intelligence. Springer, Berlin, pp 1015–1021

Tabachnick BG, Fidell LS (2013) Using multivariate statistics (6. bs.)

Tharwat A (2018) Classification assessment methods. Appl Comput Inf. https://doi.org/10.1016/j.aci.2018.08.003

Vapnik V (1979) Estimation of dependences based on empirical data. Nauka, Moscow, pp 5165–5184, 27 (**in Russian**) (English translation: Springer, New York, 1982)

Vert JP, Tsuda K, Schölkopf B (2004) A primer on kernel methods. In: Schölkopf B, Tsuda K, Vert JP (eds) Kernel methods in computational biology. The MIT Press, London

Weinberger KQ, Blitzer J, Saul LK (2006) Distance metric learning for large margin nearest neighbor classification. In: Advances in neural information processing systems, pp 1473–1480

Witten IH, Frank E, Hall MA, Pal CJ (2016) Data mining: practical machine learning tools and techniques. Morgan Kaufmann, Burlington

Zou KH (2002) Receiver operating characteristic (ROC) literature research. On-line bibliography. http://splweb.bwh.harvard.edu

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.