PROCEEDINGS, 50th Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, February 10-12, 2025
SGP-TR-229

Impact of Automated Time Series Feature Engineering on Enhancing Incident Prediction Rates in Geothermal Drilling

Aira H. Aspiras[1]*, Sadiq J. Zarrouk[1], Ralph Winmill[2], and Andreas W. Kempa-Liehr[1]

[1]Department of Engineering Science and Biomedical Engineering, The University of Auckland, Private Bag 92019, New Zealand,

[2]Contact Energy Ltd, Wairakei Power Station, Taupo, New Zealand

* aira.aspiras@auckland.ac.nz

Keywords: Geothermal drilling; Rate of penetration; Machine learning; Time series; Feature engineering

ABSTRACT

Improving overall drilling efficiency in geothermal systems involves two key strategies: increasing the rate of penetration (ROP) and minimising non-productive time (NPT). Drilling operations generate vast amounts of time-series data, including parameters like weight on bit, rotation speed, mud flow rate, and temperature. This data can be highly variable and noisy, making it challenging to interpret without sophisticated analysis. Machine learning (ML) offers significant promise in providing a data-based decision support system for drilling operations to address geothermal challenges through its predictive modelling and adaptive learning capabilities. By applying machine learning algorithms to raw data and engineered features, the system can identify patterns that precede hole-related non-productive time (NPT) incidents, such as stuck pipe situations or borehole instability. This paper compares the improvement in the incident prediction rate of the model arising from the use of automated time-series feature engineering techniques as compared to the traditional naïve, manual, and domain-specific feature generation practices. Early detection is crucial in taking corrective actions before problems escalate. By addressing issues before they result in significant NPT, the system can lead to more efficient drilling schedules and less wasted time, ultimately contributing to improved operational performance. As this system matures, it has the potential for widespread deployment across various drilling operations.

1. INTRODUCTION

1.1 Background of the Study

Stuck pipe situations occur when the drilling assembly is unable to rotate and move up or down (API Recommended Practice 54, 2019). Some common reasons for such incidents are: (1) formation-related issues like collapsing or sloughing formation, (2) inadequate hole cleaning leading to hole pack off, (3) ledges and keyseats from a bad wellbore geometry. Freeing stuckpipes incurs significant rig time, albeit with a low success rate, resulting in delays in drilling operations and overshooting the planned cost of the well (Finger & Blankenship, 2012; Nmegbu & Ohazuruike, 2014).

Drilling in geothermal reservoirs comes with inherent challenges that can often lead to precarious stuckpipe situations. The fractures and faults, which are some of the best sources of permeability (Grant & Bixley, 2011) for wells, are also sources of formation instability (Nmegbu & Ohazuruike, 2014) and losses during drilling. Due to these huge fractures, it is also common to drill geothermal reservoirs on total losses or without any returns to the surface (Finger & Blankenship, 2012), making hole cleaning inefficient and current formation characteristics unknown.

Ensuring good hole conditions during drilling is not only time-consuming but also tedious for drilling personnel tasked to monitor the drilling progress. Several drilling data from mud logs are being transmitted to the surface in real-time by as much as a datapoint per second, and drilling personnel are expected to be able to analyse and infer the hole conditions and respond accordingly on top of day-to-day rig site operations. Because of this characteristic nature of drilling, adequate experience and exposure to different drilling scenarios are necessary to be able to detect potential issues from the smallest hints in the drilling trends and execute appropriate drilling measures in relation to the trends observed.

The emergence of advanced artificial intelligence (AI) technologies, in conjunction with the extensive acquisition of drilling data in easily accessible digital formats, enables the application of machine learning (ML) as an innovative approach to addressing drilling challenges and optimising various drilling applications like incident detection.

Existing machine learning studies focus on a wide variety of drilling-related topics, but they generally fall under three (3) main categories: Review Papers, Incident Detection, and Optimisation. A specific area of interest lies in incident detection studies that focus on drilling issues like predicting stuck pipe events, estimating their likelihood, and identifying specific mechanisms causing the issue. Works of Gurina et al. (2019, 2022b, 2022a) and Antipova et al. (2019) all aim to detect drilling anomalies using Gradient Boosting and its variants, while works of Shadizadeh et al. (2010), Al-Baiyat et al. (2012), Jahanbakshi et al. (2012), and Rostami and Manshad (2014) mainly used Artificial Neural Networks (ANN) and Support Vector Machine (SVM) as machine learning models. Consistent with the No-Free-Lunch theorem, there is still no concrete answer which model performs best for incident prediction. In addition, all of them were conducted in oil and gas fields and none in geothermal fields, presenting an opportunity to explore similar uses for geothermal drilling.

1.2 Objective, Scope and Limitation

This study aims to investigate the effects of automated feature engineering on incident detection machine learning models and explore how it can be used to improve computing efficiency and redefine how we look at drilling data relating to stuckpipe incidents. This is in support of the ultimate goal of developing a usable early warning incident detection system to assist drillers and engineers when monitoring drilling operations, especially in drilling reservoirs in blind conditions.

The drilling data used in this study was acquired from drilling the reservoir section of a deep geothermal well in New Zealand where severe losses were experienced, and thus, was drilled blind to the target depth (TD). The incidents denoted as non-productive time (NPT) is limited only to hole-related drilling issues, primarily stuckpipes and tight holes which were identified by the domain experts who drilled the well.

2. MODEL DEVELOPMENT

The general models were developed using Python (Version 3.12.2) as the main programming language, Matplotlib (Version 3.8.3) for the different data visualisations, Pandas (Version 2.2.1) for data manipulation and analysis, Sci-kit Learn (Version 1.4.1) for machine learning modules, tsfresh (Version 0.20.2) for automated feature engineering, and Minitab® (Version 22) for all statistical computation and data visualisation.

2.1 Parametric Design

Two (2) submodels using different drilling inputs will be evaluated across six (6) machine learning (ML) algorithms, as shown in Table 1. The secondary model will contain a subset of drilling inputs from the primary model, resulting in 12 models. Furthermore, another 12 models will also be run to cover cases where a Pressure-While-Drilling (PWD) sensor is not available.

Table 1 Factors for Investigation

Model/ Drilling Parameters	ML Algorithms	Data Availability
Primary model: All	Decision Tree	With PWD Data
	Random Forest	Without PWD Data
Secondary model: Critical Few	AdaBoosted DT	
	AdaBoosted RF	
	GBoost	
	XGBoost	

2.1.1 Machine Learning Algorithms

Decision Tree (DT) is a popular, simple machine learning method used for classification and regression. It starts at a root node and splits into two subsets at each step, continuing until it reaches a leaf node, which shows the final outcome (Breiman, 2001).

Random Forest (RF) is a machine learning algorithm based on Decision Trees. It improves on Decision Trees by creating multiple trees and making predictions by averaging or voting on the results from each tree (Breiman, 2001).

Adaptive Boosting (AdaBoost) was introduced by Freund and Schapire (Freund & Schapire, 1997). It trains a series of simple models, called weak learners, one after another. Each model focuses more on the mistakes made by the previous one. At the end, all the models are combined through weighted voting to create a stronger final model. Adaptive boosting was applied to both Decision Tree and Random Forest in this study.

Gradient Boosting (Gboost) is a popular ensemble method that combines weak models to create a stronger one. It works in steps, improving the model with each iteration, and is used for both classification and regression. The model is trained on labeled data to make predictions or understand relationships between features and outcomes (Friedman, 2001).

Extreme Gradient Boosting (XGBoost) is a more advanced variation of Gradient Boosting. Similar with GBoost, it builds a series of simple models (weak learners) one after another, each one trying to correct the mistakes of the previous model. XGBoost uses techniques that makes the processing more efficient and less likely to overfit the data (Chen & Guestrin, 2016).

2.1.2 Drilling Parameters

Table 2 summarises the drilling inputs used in this study. The data presented in bold letters are the additional data acquired if no Pressure-While-Drilling (PWD) was run.

Table 2 Drilling Inputs Used

Code	Drilling Parameters	Description
Hole_Depth	Hole Drilled	Depth of hole drilled
Hook	Average Hookload	Amount of force referring to the overall drilling load, including the weight of the drill string in air, the BHA, and drag, reduced by buoyant and friction forces, among others
WOB	Average Weight on Bit	Downward force exerted by the BHA through the bit into the formation during drilling operations
RPM_Surface	Average Surface Rotation	Rotary speed at which the BHA is rotated at surface
Flow_Out	Pump Flow out	Rate at which drilling fluid returns to the surface
SPP	Standpipe Pressure	Total frictional pressure drop in the hydraulic system affected by Mud properties, downhole drilling conditions, and equipment health
Flow_In	Pump Flow In	Rate at which drilling fluid is pumped into the system
ROP	Rate of Penetration	Rate at which the hole section is drilled
TRQ_ave	Average Surface Torque	Average rotational force required to overcome all the frictional forces between the drill string and the formation while drilling measured at surface
TRQ_max	Maximum Surface Torque	Maximum rotational force required to overcome all the frictional forces between the drill string and the formation while drilling measured at surface
Mud_In	Average Mud Weight In	Measure of the density of drilling fluid pumped into the system
Mud_Out	Average Mud Weight Out	Measure of the density of drilling fluid returns to the surface
Temp_In	Mud Temperature In	Temperature of the drilling fluid pumped into the system
Temp_Out	Mud Temperature Out	Temperature of the drilling fluid returns to the surface
MWD_Temp	Measurement-While-Drilling Temperature	Temperature of the drilling fluid downhole and transimitted in real time to surface
Block_Pos	Block Position	Position of the traveling block along a vertical axis
Ann_Pres	**Annular Pressure**	Measure of the fluid pressure in the annulus downhole near the bit between the drillstring and casing or borehole wall
Int_Pres	**Internal Pressure**	Measure of fluid pressure downhole inside the pipe
ECD	**Equivalent Circulating Density**	Calculated from the Annular Pressure and Mud density and expressed in units similar to MW

2.2 Methodology

Presented in Figure 1 is the machine learning methodology used in this study to generate different ML models. The model shall follow a Supervised ML technique where data is trained through a set of labelled data performed by domain experts to make predictions.

- Step 1: Time series data were acquired in csv format and were inspected for missing data and outliers as determined by the domain expert. The previous “good” value replaced any value that was out of the range or missing.
- Step 2: After performing the data quality check, they were then processed into two (2)-hour window sets rolling every hour. It was designed to forecast the presence of an incident by learning from the automatically generated features from each window. A total of 216 windows were generated using tsFresh.
- Step 3-5: ML model training and calibration are performed iteratively through hyperparameter tuning until satisfactory ML performance is achieved. The predictions were validated using a time-series split, ensuring that the algorithm learns only from previously seen values. Model performance was evaluated according to the following metrics: (1) Receiver Operating Characteristic Curve (ROC-AUC) and (2) Matthew’s Correlation Coefficient (MCC) Scores.

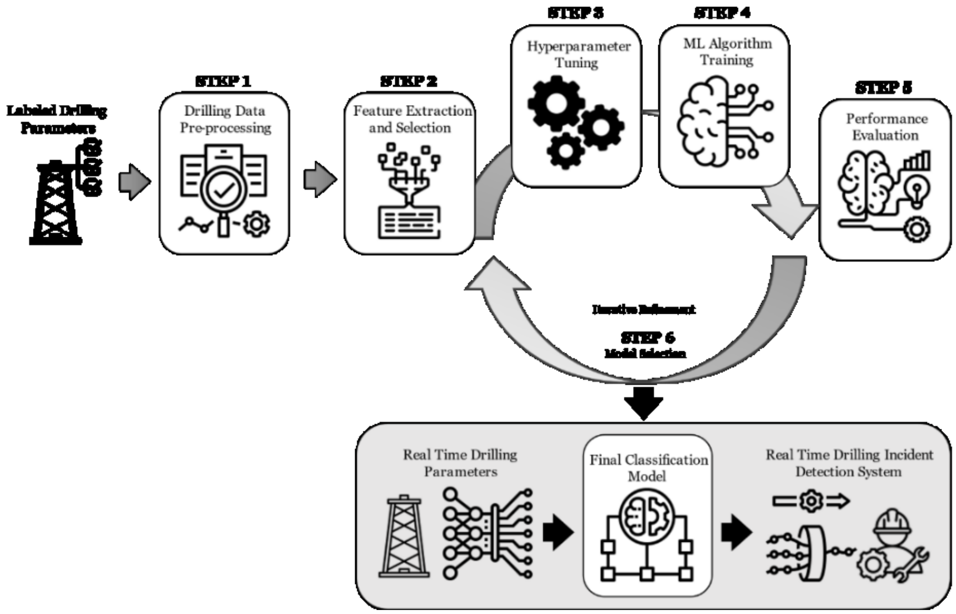

Figure 1 Machine Learning requires extensive data pre-processing techniques prior to loading into the selected ML model, after which the model needs to be cross-validated for fitness.

3. RESULTS AND DISCUSSION

The following analysis investigates how automated feature engineering improves ML predictions – specifically, if it can be used to identify the critical few drilling inputs needed to produce decent incident detection results from the same model.

A total of 19 drilling parameters are available from the raw data, and these were used as inputs in the primary ML model. Using tsfresh (Christ et al., 2018), features were generated from each window set and were fitted to the predicted values. Relevant features were identified and ranked according to their p-values (fdr level=0.01). Only the drilling parameters included in the top ten (10) features become part of the 'Critical Few'.

Table 3 shows the top ten (10) features and the 'Critical Few' drilling parameters, namely:

- Hole Depth
- Internal Pressure
- ECD
- Annular Pressure
- Average Torque.

Table 3 Top 10 Features Generated – with PWD data

feature	type	p_value	relevant
Int_Pres__ratio_beyond_r_sigma__r_3	real	4.607809e-14	True
Ann_Pres__number_peaks__n_5	real	4.489890e-12	True
Int_Pres__ratio_beyond_r_sigma__r_2.5	real	7.962780e-12	True
Hole_Depth__last_location_of_maximum	real	1.380609e-10	True
ECD__number_peaks__n_5	real	2.095895e-10	True
Hole_Depth__ratio_beyond_r_sigma__r_2.5	real	6.255923e-10	True
Int_Pres__quantile__q_0.1	real	1.478757e-09	True
Int_Pres__quantile__q_0.2	real	6.863116e-08	True
Ann_Pres__number_peaks__n_3	real	8.822999e-08	True
TRQ_ave__mean_n_absolute_max__number_of_maxima_7	real	2.114809e-07	True

A secondary ML model was run using only the 'Critical Few' drilling inputs, down to five (5) from the previous 19, resulting in a 71% reduction in total features generated and a 45% reduction in relevant features, improving the efficiency of the runs (Table 4).

Table 4 All vs Critical Few - Features differences – with PWD

Data Availability	Drilling Parameters	Features	Relevant Features
With PWD Data	All (19)	16443	451
	Critical (5)	4698 (↓71%)	246 (↓45%)

However, to determine suitability for use, it needs to be investigated whether the use of 'Critical Few' can predict outcomes as well as the primary models using all the drilling parameters. Figure 2 shows comparisons of the ROC AUC and MCC Scores using 'All' vs 'Critical Few' in the different ML algorithms evaluated. The area under the ROC curve is a plot of the true positive rate (TPR) on the Y axis and the false positive rate (FPR) on the X axis. ROC_AUC measures how well a classifier performs in comparison to a false classification (Sokolova & Lapalme, 2009). However, it still treats misclassifications equally (Provost & Fawcett, 2001) and does not perform well in highly imbalanced datasets such as this. Matthews Correlation Coefficient (MCC) provides a balanced measure that considers true positives, true negatives, false positives, and false negatives and is a robust metric for binary classification and imbalanced datasets(Matthews, 1975). To further gauge the model's performance and prevent overfitting, a specific cross validation technique specifically dealing time series data called blocked time series split method (blocks=4) was employed. Blocked time series splits divide the training set into two parts of folds at each step, making sure the validation set always comes before the training set. Margins are then added at two positions, to stop future data from leaking into the model (Bergmeir & Benítez, 2012).

Based on the combined results, the 'Critical few' models performed similarly and, in some cases, even better than the results of the primary models where all drilling inputs were used, especially in the MCC scores. In addition, previously unsatisfactory MCC scores of <0.6 greatly improved using the Critical Few drilling inputs only. This confirms that automated feature engineering can be used indirectly to identify which drilling parameters really matter when dealing with potential stuckpipe situations. Furthermore, from the results in Figure 2, AdaRF, Ada DT, and XGBoost are the top three(3) performing ML algorithms.

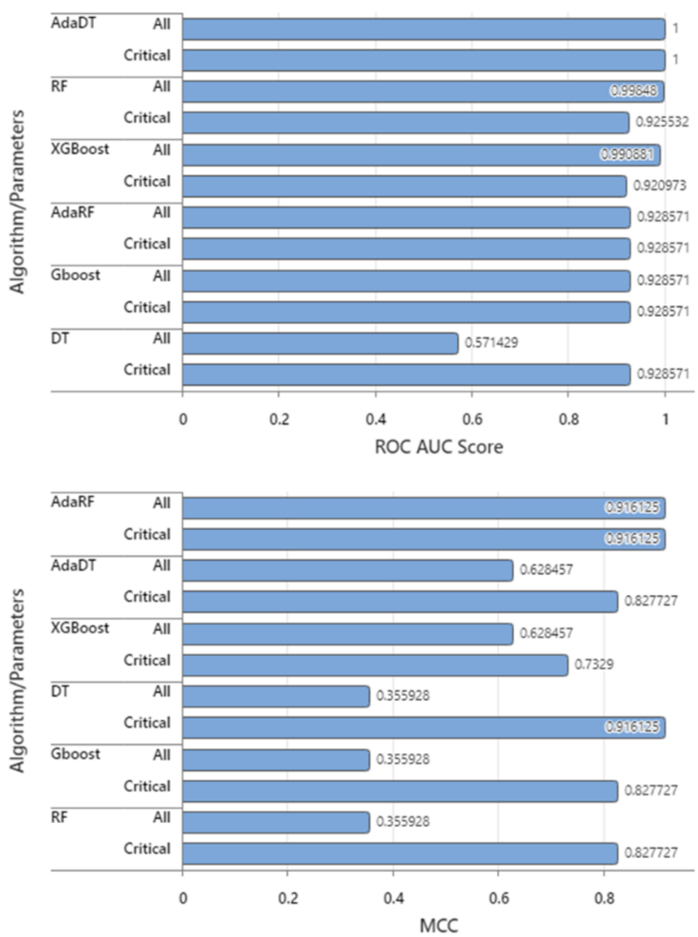

Figure 2 Secondary ML models using only 'Critical Few' parameters performed adequately in both ROC_AUC and MCC scores – with PWD

It is, however, important to note that 3 out of 5 parameters are PWD-acquired data and may not always be available. Thus, an analysis without these PWD data was also conducted. It is logical that PWD data are significant predictors as those are measurements from the downhole and thus reflect the real-time conditions of the hole below. Hole depth and torque are relevant drilling factors in identifying stuckpipes as most problems are related to depth due to formation in the area, and torque is the response of the rotating drill string, indicating whether there are some anomalies in the trend or not.

However, not all drilling assemblies are equipped with PWD. For the primary models with no-PWD data, 16 drilling parameter inputs were used, which generated 13,311 features, 299 of which were deemed relevant according to p-values (fdr level=0.01).

Using the same methodology previously, Table 5 shows the top ten (10) features and the 'Critical Few' parameters for non-PWD were also identified and listed below:

- Hole Depth,
- Average Torque
- Max Torque

Table 5 Top 10 Features Generated – without PWD data

feature	type	p_value	relevant
Hole_Depth__last_location_of_maximum	real	1.380609e-10	True
Hole_Depth__ratio_beyond_r_sigma__r_2.5	real	6.255923e-10	True
TRQ_ave__mean_n_absolute_max__number_of_maxima_7	real	2.114809e-07	True
TRQ_max__mean_n_absolute_max__number_of_maxima_7	real	3.029545e-07	True
TRQ_ave__change_quantiles__f_agg_"var"__isabs_...	real	3.029888e-07	True
TRQ_max__absolute_maximum	real	3.333634e-07	True
TRQ_max__maximum	real	3.333634e-07	True
TRQ_ave__cid_ce__normalize_False	real	3.475054e-07	True
TRQ_ave__change_quantiles__f_agg_"var"__isabs_...	real	3.475054e-07	True
TRQ_ave__maximum	real	4.813636e-07	True

A secondary ML model was run using only the 'Critical Few' parameters, down to three (3) from the previous 16, resulting in a 76% reduction in total features generated and a 459 reduction in Relevant features, improving the efficiency of the runs (Table 6)

Table 6 All vs Critical Few - Features differences – no PWD

Data Availability	Drilling Parameters	Features	Relevant Features
Without PWD Data	All (16)	13311	299
	Critical (3)	3132 (↓76%)	151 (↓49%)

Figure 3 shows the comparison of the ROC AUC and MCC Scores using 'All' vs 'Critical Few' in the different ML algorithms evaluated. Based on the ROC AUC scores, the secondary models performed similarly except in the case of DT, where the use of critical parameters showed better results. The same was seen in MCC scores in AdaDT, Gboost, and DT. However, runs of the Critical Few were inferior in XGBoost and Random Forest, showing mixed results for the different ML algorithms. This shows that automated feature engineering can be used indirectly to identify which parameters really matter when dealing with potential stuckpipe situations, but only in some ML models when no PWD data is available.

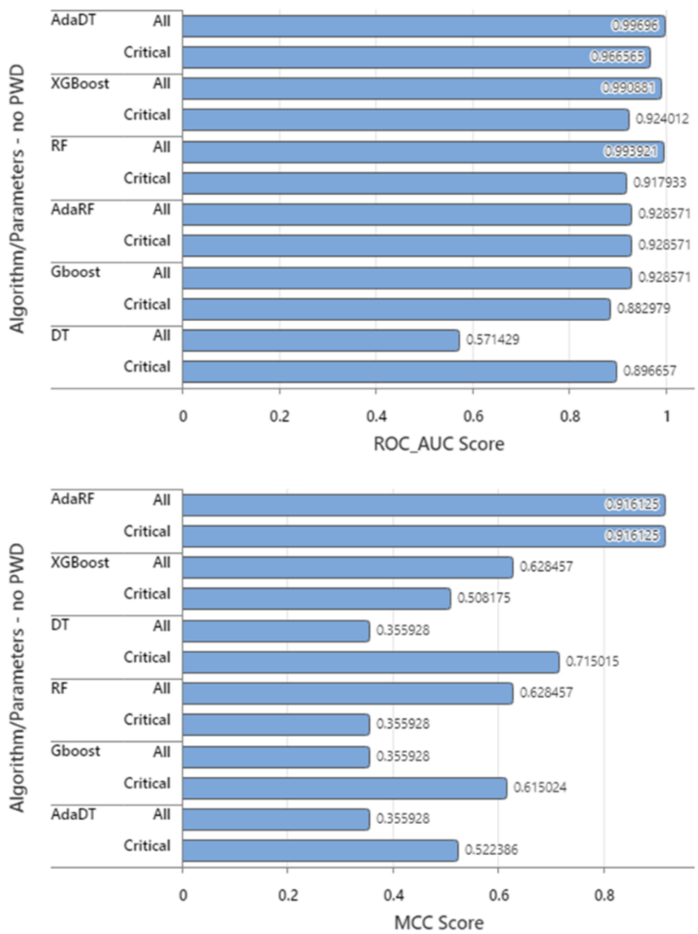

Figure 3 Secondary ML models using only ‘Critical Few’ parameters performed adequately in ROC_AUC and but was inferior for some ML algorithms in MCC scores – no PWD

Presented in Figures 4 and 5 are the main effects and interaction plots of the different factors investigated for both ROC AUC and MCC scores. Based on the combined results, it can be concluded that having PWD data outperforms not having downhole data as drilling inputs for incident detection as these parameters directly influence hole drilling incidents using features automatically generated by tsfresh.

Furthermore, using the critical drilling parameters improves the effectiveness of the prediction success rate and the Adaboosted Random Forest is the highest-performing ML model. Also, the interaction effects of having PWD and establishing the critical parameters significantly improved the predictability of the stuckpipe incidents.

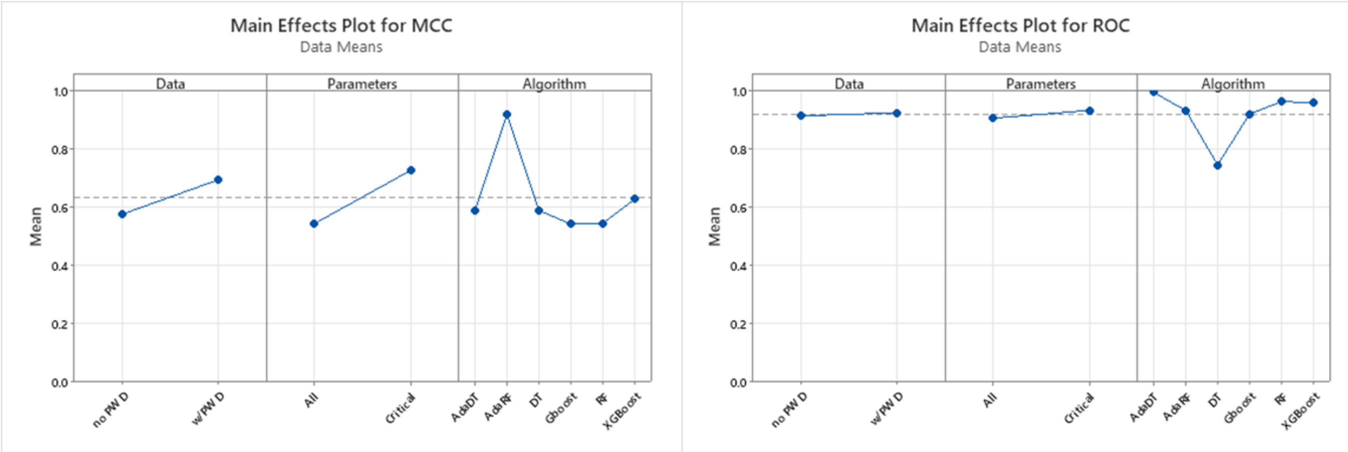

Figure 4 Main interaction plots for the different parameters

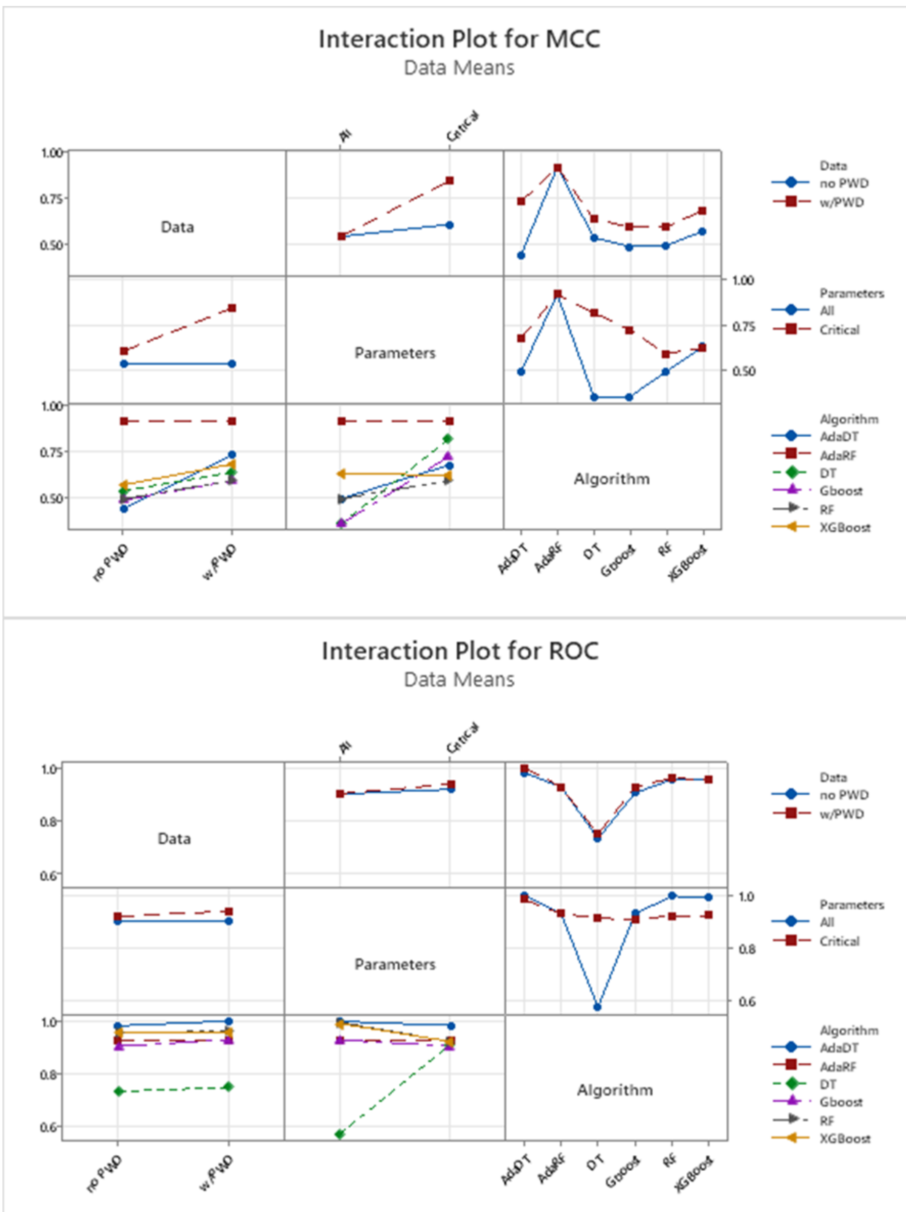

Figure 5 Interaction Plots of the different parameters

4. CONCLUSION

Drilling in geothermal reservoirs comes with inherent challenges that can often lead to precarious stuckpipe situations. The emergence of advanced artificial intelligence (AI) technologies, together with the extensive acquisition of drilling data, provides an opportunity to use machine learning (ML) as an innovative approach to addressing drilling challenges and optimising various drilling applications like incident detection. This study specifically aims to investigate the effects of automated feature engineering on incident detection machine learning models and explore how it can be used to improve computing efficiency and redefine how we look at drilling data relating to stuckpipe incidents, specifically, if it can be used to identify the critical few drilling inputs needed to produce decent incident detection results from the same model.

The use of ‘Critical Few’ can predict outcomes and the primary models using all the drilling parameters, especially those with PWD Data. (1) Hole Depth, (2) Internal Pressure, (3) ECD, (4) Annular Pressure, and (5) Average Torque were identified as the critical few drilling inputs needed in an automated feature engineering ML model to predict stuckpipe incidents successfully. However, for those without PWD data, (1) Hole Depth, (2) Average Torque and (3) Max Torque may be able to perform satisfactorily but not in all ML algorithms. Identifying the critical few reduced drilling inputs by as much as 80%, resulting in computing efficiency. This was only made possible through automated feature engineering and is not possible with naïve and manual techniques.

Based on the main effects and interaction effects results, it was found that having PWD data outperforms not having those as drilling inputs for incident detection. Furthermore, using the critical drilling parameters improves the effectiveness of the prediction success rate and the Adaboosted Random Forest is the best-performing ML model. Also, the interaction effects of having PWD and establishing the critical parameters significantly improved the predictability of the stuckpipe incidents.

The findings in this study support the ultimate goal of developing a usable early warning incident detection system to support drillers and engineers when monitoring drilling operations, especially in drilling reservoirs in blind drilling conditions.

REFERENCES

Al-Baiyat, I., & Heinze, L. (2012, May). Implementing Artificial Neural Networks and Support Vector Machines in Stuck Pipe Prediction. *SPE Kuwait International Petroleum Conference and Exhibition*. https://doi.org/https://doi.org/10.2118/163370-MS

Antipova, K., Klyuchnikov, N., Zaytsev, A., Gurina, E., Romanenkova, E., & Koroteev, D. (2019). Data-Driven Model for the Drilling Accidents Prediction. *SPE Annual Technical Conference and Exhibition*, *2*, 195888. https://doi.org/https://doi.org/10.2118/195888-MS

API Recommended Practice 54. (2019). Occupational Safety and Health for Oil and Gas Well Drilling and Servicing Operations. In *API Recommended Practice 54, 4th Edition*.

Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, *191*, 192–213. https://doi.org/https://doi.org/10.1016/j.ins.2011.12.028

Breiman, L. (2001). Random forests. In *Machine Learning* (Vol. 45, pp. 5–32). Kluwer Academic Publisher. https://doi.org/https://doi.org/10.1023/A:1010933404324

Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. https://doi.org/10.1145/2939672.2939785

Christ, M., Braun, N., Neuffer, J., & Kempa-Liehr, A. W. (2018). Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing*, *307*, 72–77. https://doi.org/https://doi.org/10.1016/j.neucom.2018.03.067

Finger, J., & Blankenship, D. (2012). *Handbook of Best Practices for Geothermal Drilling*. https://doi.org/https://doi.org/10.2172/1325261

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, *55*(1), 119–139.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 1189–1232.

Grant, M., & Bixley, P. (2011). *Geothermal Reservoir Engineering* (2nd ed.). Academic Press. https://doi.org/https://doi.org/10.1016/C2010-0-64792-4

Gurina, E., Klyuchnikov, N., Antipova, K., & Koroteev, D. (2022a). Forecasting the abnormal events at well drilling with machine learning. *Applied Intelligence*, *52*(9). https://doi.org/https://doi.org/10.1007/s10489-021-03013-x

Gurina, E., Klyuchnikov, N., Antipova, K., & Koroteev, D. (2022b). Making the black-box brighter: Interpreting machine learning algorithm for forecasting drilling accidents. *Journal of Petroleum Science and Engineering*, *218*. https://doi.org/https://doi.org/10.1016/j.petrol.2022.111041

Gurina, E., Klyuchnikov, N., Zaytsev, A., Romanenkova, E., Antipova, K., Simon, I., Makarov, V., & Koroteev, D. (2019). Application of machine learning to accidents detection at directional drilling. *Journal of Petroleum Science and Engineering*, *184*. https://doi.org/https://doi.org/10.1016/j.petrol.2019.106519

Jahanbakhshi, R., Keshavarzi, R., Shoorehdeli, A., & Emamzadeh, A. (2012). Intelligent Prediction of Differential Pipe Sticking by Support Vector Machine Compared With Conventional Artificial Neural Networks: An Example of Iranian Offshore Oil Fields. *SPE Drilling and Completion*, *27*(04), 586–595. https://doi.org/https://doi.org/10.2118/163062-PA

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, *405*(2), 442–451. https://doi.org/https://doi.org/10.1016/0005-2795(75)90109-9

Nmegbu, C. G. J., & Ohazuruike, L. V. (2014). Wellbore Instability in Oil Well Drilling: A Review. *International Journal of Engineering Research and Development*, *10*(5), 11–20.

Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning*, *42*, 203–231.

Rostami, H., & Khaksar Manshad, A. (2014). A New Support Vector Machine and Artificial Neural Networks for Prediction of Stuck Pipe in Drilling of Oil Fields. *Journal of Energy Resources Technology*, *136*(2). https://doi.org/https://doi.org/10.1115/1.4026917

Shadizadeh, S. R., Karimi, F., & Zoveidavianpoor, M. (2010). Drilling Stuck Pipe Prediction in Iranian Oil Fields: An Artificial Neural Network Approach. *Iranian Journal of Chemical Engineering*, *7*(4), 29–41. https://www.ijche.com/article_10326.html

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/https://doi.org/10.1016/j.ipm.2009.03.002