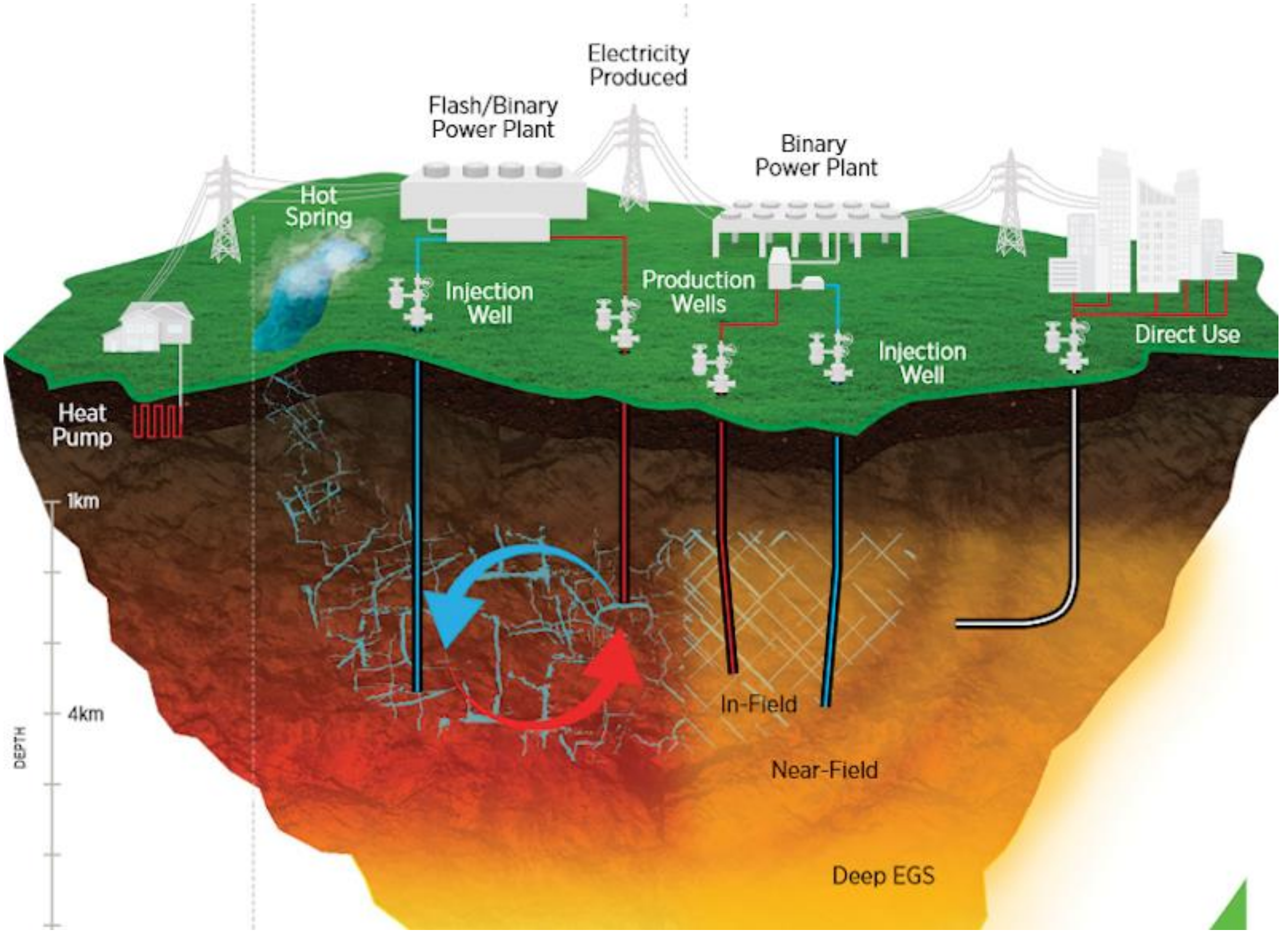


Real Time Prediction of Hardness-Based Geothermal Formation

DSA 5103 – Intelligent Data Analytics
Group 11 | Course Project

Chinedu Joseph Nwosu | Ayomide Hamzat | Steven Taylor | Daniel Tetteh

Contents

1. Executive Summary

This report presents the accomplishments of the exploration of several supervised machine learning algorithms in the identification of different Geothermal formation based on rock hardness. Several emerging technologies have shown prospects in the identification of Geothermal formation during drilling; however, these technologies are limited to the extreme temperature and pressure conditions of Geothermal formations, hence it is pertinent to explore alternate measures to identify formation types in order to aid in the selection of appropriate drilling bits and reduce the associated risk of drilling hard formation. The primary objective of this report is to present the findings of explored supervised machine learning algorithms for hardness-based formation type prediction.

During drilling, the rate of penetration (ROP) is set as the objective function in which maximizing the ROP yields shorter drilling time, thus increased savings. The surface drilling parameters are utilized to maximize ROP and are subject to the encountering formation, thus, it is important to determine the formation type in order to enable the decision of the type of the drilling bit and the optimum controllable drilling parameters needed to optimize the rate of drilling. Hence, in this project we assume that surface drilling parameters are invariably correlated to formation type and can be utilized to identify the different formation types.

Prior to the development of the formation type model, Initial clustering analysis using Kmeans, and Kmodes was implemented to reduce the high factor level of the target variable using a combination of scenarios. After several attempts, the Mohrs scale was eventually used to collapse the lithology into 3 formation types based on their hardness. Seven (7) classification models were developed to predict the hardness-based formation type with Random Forest yielding the best cross validation performance of 99.3% accuracy.

The findings from this project illustrate the significance of using machine learning algorithms in solving drilling optimization problems. The analysis inferred or deduced from this project can be applied to geothermal drilling and oil and gas drilling operations, thus a wide range of field application. This study recommends more acquisition of drilling data from a wide range of wells to significantly capture an equal amount of drilling data from different formations in order to increase further the generalization of the developed model.

2. Problem Background

2.1 Problem Understanding (Problem, Description and Background)

The demand for clean energy has spurred several studies to explore renewable energy sources with the goal to reduce the rising global carbon footprint. Geothermal energy is one of the numerous explored renewable sources and can simply be referred to as the heat energy extracted from the earth's crust which can be used as a source for power generation as well as in the utilization for heating, and cooling of buildings. The resources drilled to extract this heat energy is characterized by very hard rocks, making it a challenge for drilling as operators risk the occurrence of tool failures when utilizing inappropriate drilling bits in the encountered formation. This challenge results in a financial implication on the overall cost of drilling, as additional cost associated with tool replacement, delay in drilling, i.e., rented equipment being charged per day and other operational logistics needed to keep the drilling rig running during this delay time. Conventionally, rock hardness is determined by experimentally identifying the type of rock and classifying the rock type which aids to select appropriate drilling bit for subsequent drilling of forward well sections or depths. This approach is time costly, and its application is based on a trial-and-error analysis as further well sections could have varied well lithologies. Well logging has been preferred to this experimental approach, as operators are able to utilize expensive logging tools to acquire petrophysical properties of the drilled rocks in near real time, which effectively aids the classification of the rock types. Despite this advantage, these logging tools are limited to extreme temperature operating conditions as in the case of geothermal exploration, thus, making their applications in high-temperature resources very limited.

The objective of this project is to develop a classification model to predict hardness-based rock type using real time surface drilling parameters. The significance of this proposed project lies in the ability to reduce the dependencies of expensive logging tools, thus, cutting cost while delivering near similar potential value by leveraging on historical data. The Geothermal Data Repository (GDR) contains drilling data of several geothermal wells alongside their corresponding well logging reports containing rock mineralization per formation depth drilled.

2.2 Data Description/Understanding

The data utilized for this study was obtained from the three (3) geothermal wells: well 58-32 , well 16A(78)-32) located in the Utah FORGE site and the Fallon well 21-31 located in west-central Nevada (Figure 2-1), which is publicly accessible from the Geothermal Data Repository (GDR) webpage (https://gdr.openei.org). Table 1 represent a summary dictionary of the drilling data used in this analysis.

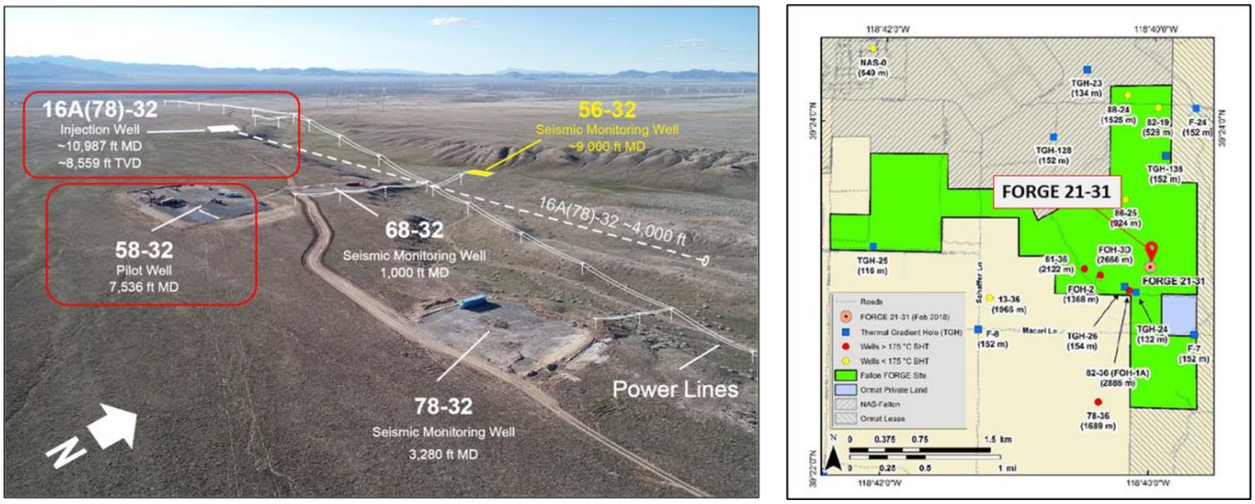


Figure 2-1: Well locations at the three (3) wells (Courtesy Utah FORGE web and Kraal, K. O., & Ayling, B. (2019)).

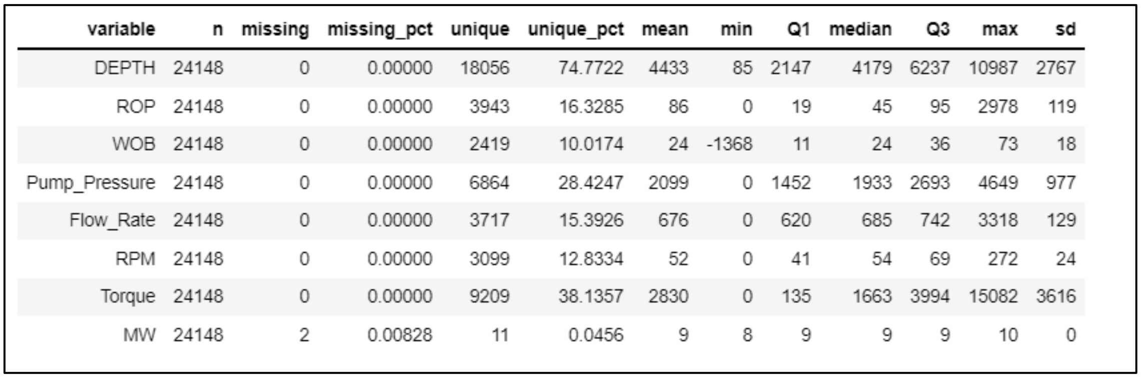
Table 1: Summary dictionary of the drilling log data

Variables	Field Units	Description
Rate of penetration (ROP)	ft/hr	The speed rate at which drilling occurs
Weight on bit (WOB)	klbs	The amount of weight applied on the drilling bit to drill
Pump pressure	psi	This is a measure of resistance to flow of drilling fluid
Flow rate	gpm	This is the rate at which drilling fluid is being pumped
Rotary speed (RPM)	RPM	This is the speed of rotation per time
Torque	psi	This is the force required to rotate the drilling bit on impact of a rock
Mud weight (MW)	ppg	This is the density of the drilling fluid used to circulate cuttings out of a well

2.2.1 Numeric Data Quality Report

The drilling log data was divided into numeric and factor variables in order to gain initial insights from the discretized dataset. The figure 2-2, provides information about the statistical description of the raw numeric data with inferences summarized below.

- The drilling data has 8 numeric features with a total of 24,148 observations
- The numeric features record a total of 0.8% missingness across the entire dataset
- The data contains drilling data with maximum depth of 10,987ft. The large uniqueness equally observed is attributed to the different combinations of implemented drilling data across varying formation rocks with varying hardness type
- The observations are depth matched and vary according to the occurrence of naturally occurring formation, thus depth feature will be irrelevant to our analysis
- The numeric features are expected to be of positive values ; however, we can see WOB has negative values present, thus this is an outlier requiring capping
- Mud weight initially gives the impression of being a factor variable, however applying domain knowledge which suggests that mud weight can be varied based on the additives of the drilling mud formulated and the operational suitability of the fluid in the drilling window, we considered it to be a numeric feature.



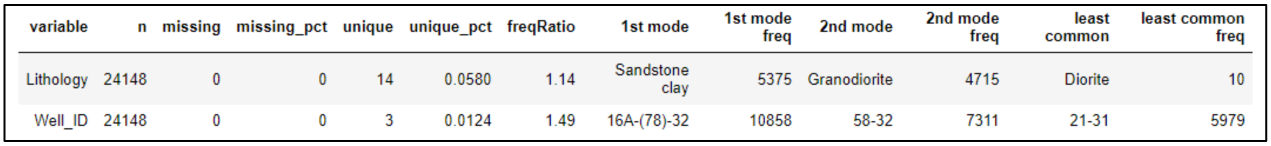
variable	n	missing	missing_pct	unique	unique_pct	mean	min	Q1	median	Q3	max	sd
DEPTH	24148	0	0.00000	18056	74.7722	4433	85	2147	4179	6237	10987	2767
ROP	24148	0	0.00000	3943	16.3285	86	0	19	45	95	2978	119
WOB	24148	0	0.00000	2419	10.0174	24	-1368	11	24	36	73	18
Pump_Pressure	24148	0	0.00000	6864	28.4247	2099	0	1452	1933	2693	4649	977
Flow_Rate	24148	0	0.00000	3717	15.3926	676	0	620	685	742	3318	129
RPM	24148	0	0.00000	3099	12.8334	52	0	41	54	69	272	24
Torque	24148	0	0.00000	9209	38.1357	2830	0	135	1663	3994	15082	3616
MW	24148	2	0.00828	11	0.0456	9	8	9	9	9	10	0

Figure 2-2: Numeric data quality report

2.2.2 Factor Data Quality Report

The figure 2-3, provides information about the statistical description of the raw factor data with inferences summarized below.

- The drilling data has only 2 factor variables with a total of 24,148 observations, of which one of the feature; “Well_ID” will not be a useful parameter as it’s a well identifier and not a drilling parameter
- The factor data has no identified missingness
- It was also observed that Granodiorite was the most occurring lithology across the wells.
- The lithology feature which serves as our target variable has a high factor level of about 14 levels, thus there is need for consideration of masking these levels into clusters as technically some of these lithologies can be aggregated based on some clustering analysis or factor collapsing based formation property such as rock hardness.



variable	n	missing	missing_pct	unique	unique_pct	freqRatio	1st mode	1st mode freq	2nd mode	2nd mode freq	least common	least common freq
Lithology	24148	0	0	14	0.0580	1.14	Sandstone clay	5375	Granodiorite	4715	Diorite	10
Well_ID	24148	0	0	3	0.0124	1.49	16A-(78)-32	10858	58-32	7311	21-31	5979

Figure 2-3: Factor data quality report

2.3 Exploratory Data Analysis

In this section, we looked at some useful trends present in our dataset to better understand the possible relationships in our dataset.

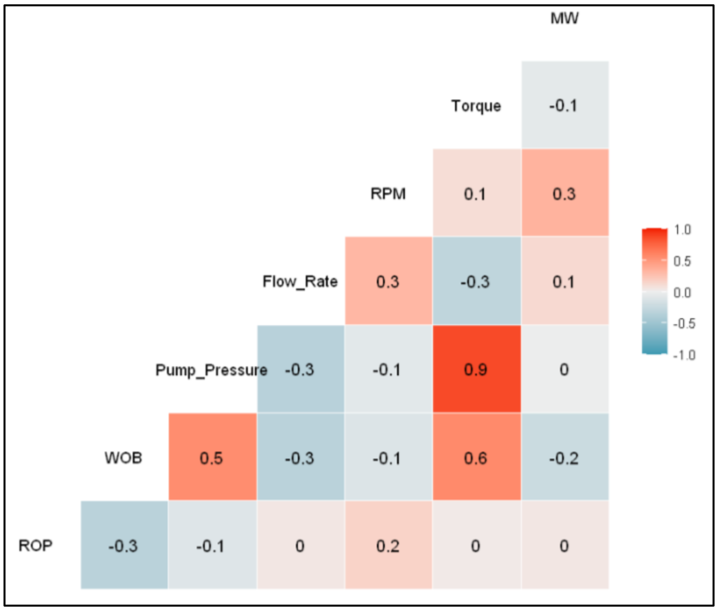


Figure 2-4: Correlation Plot of Useful Drilling Parameters

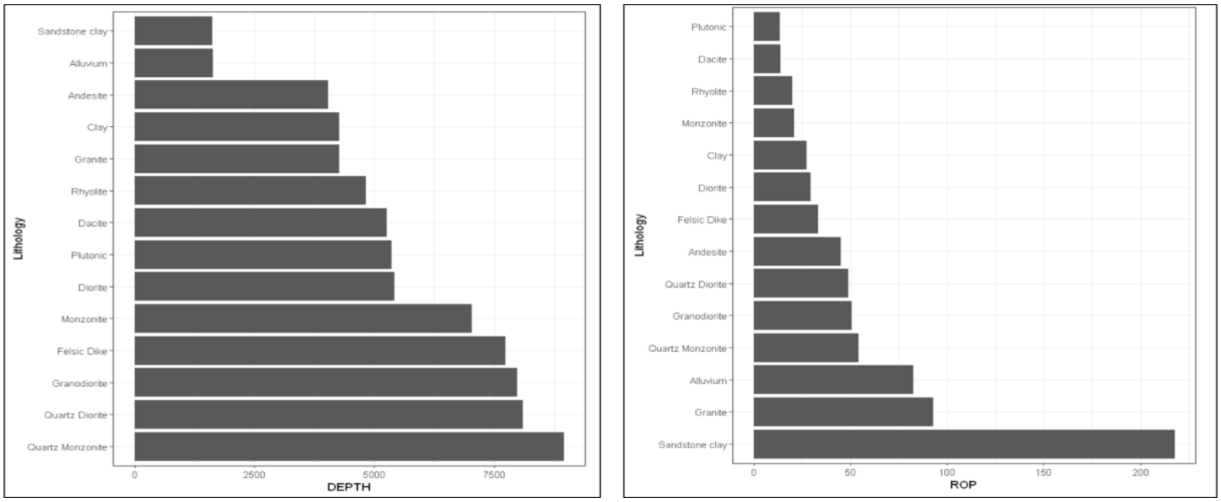


Figure 2-5: Lithology with respect to depth (left) and ROP (right)

- From the correlation plot in figure 2-4, we can see the presence of multicollinearity between pump pressure and torque.
- From the left plot (figure 2-5) it can be inferred that over 50% of the lithology types are found at depth greater than 5000ft.
- The left plot (figure 2-5) also infers that sandstone clay is found at the shallower section, thus existence of looseness. This is validated in the right plot (figure 2-5) where we can see that we experienced the highest rate of penetration, i.e., drilling at this section was very hard.

- None of the plots in figure 2-5 does justice in providing information needed to classify the lithology types, thus the need to explore linear discriminant analysis to determine the power of separability needed to predict the high level of lithologies (i.e., taking the distinct lithologies as distinct formation types based on hardness).
- Clustering analysis is an approach to segment these lithologies into groups with similar attributes. Factor collapsing is equally another approach that is domain based where we use theories in literature to collapse the high level of possible target features.

3. Methodology

3.1 Outlier Detection and Treatment and Handling of Missing Values

In this analysis, univariate analysis was conducted to identify the present of outliers in the dataset. The figure 3-1 shows some of the visually identified outliers present in some of the features. The right visual in figure 3-2 shows the implementation of a capping methodology to sub select "dataframes" within non extremeness ranges of identified outliers. Also, the capping of values was backed by domain knowledge as some values out of the specified relevant ranges are impossible to have and could be attributed to equipment error or operational pauses during drilling operation. Mud weight has 2 missing values as seen in figure 3.2 (left) and based on the tight range of values from the numerical quality report, mean imputation was considered to address the missingness.

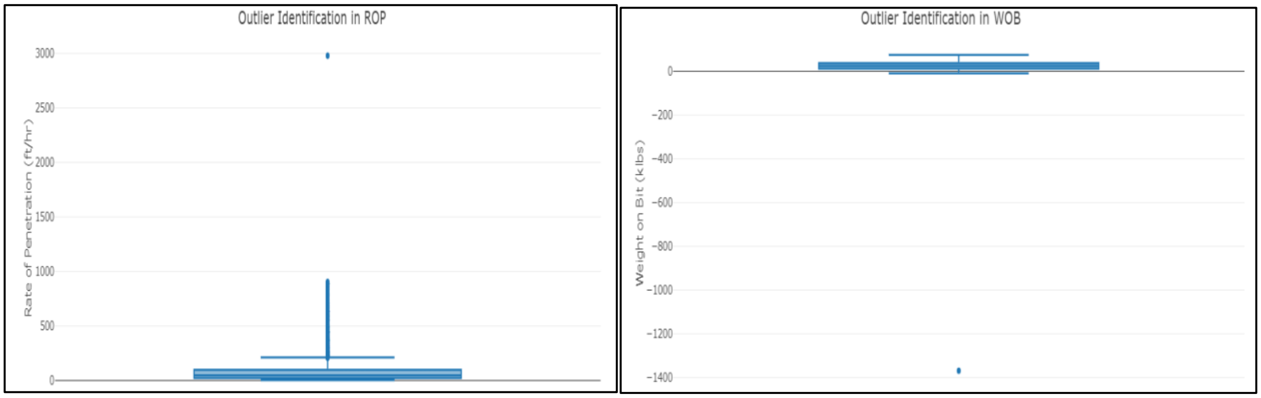
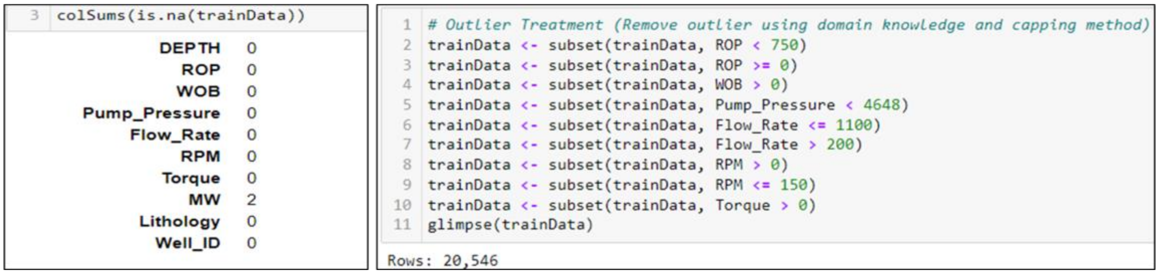


Figure 3-1: Outliers present above 1000ft/hr (left), Outliers present below 0 klbs in the weight on bit (right)



```
colSums(is.na(trainData))

         DEPTH  0
           ROP  0
           WOB  0
 Pump_Pressure  0
     Flow_Rate  0
           RPM  0
        Torque  0
            MW  2
     Lithology  0
       Well_ID  0
```

```
# Outlier Treatment (Remove outlier using domain knowledge and capping method)
trainData <- subset(trainData, ROP < 750)
trainData <- subset(trainData, ROP >= 0)
trainData <- subset(trainData, WOB > 0)
trainData <- subset(trainData, Pump_Pressure < 4648)
trainData <- subset(trainData, Flow_Rate <= 1100)
trainData <- subset(trainData, Flow_Rate > 200)
trainData <- subset(trainData, RPM > 0)
trainData <- subset(trainData, RPM <= 150)
trainData <- subset(trainData, Torque > 0)
glimpse(trainData)
Rows: 20,546
```

Figure 3-2: Outlier treatment using a capping methodology

3.2 Feature Engineering, Feature Extraction, and Identifying the Target Variable

In this section, the goal is to develop a target variable for hardness prediction. Lithology is a function of hardness but is not hardness itself. We tried a number of approaches for this as highlighted below. We tried an unsupervised data driven approach as well as domain knowledge to carry out this section as shown below:

3.2.1 Clustering analysis

Scenario 1: Clustering Analysis On Encoded Lithology

A new data frame was created with just the lithology column and was further encoded using one-hot encoding, thus generating a total of 14 binary features for clustering analysis. In this scenario we explored two algorithms, Kmeans and Kmodes algorithms and utilized the purity of the identified clusters as our evaluation metric.

Scenario 2: Clustering Analysis on Encoded Lithology And ROP

Based on domain knowledge and from preliminary inference in our data understanding section, we can see that ROP is a strong indicator of lithology as a high ROP will most likely indicate a softer formation and a low ROP will depict a harder rock. In this scenario a new dataframe was created with scaled ROP and the encoded lithology, i.e a total of 1 continuous feature and 14 binary features was used for the clustering analysis in this scenario. Due to the addition of numeric features together with numerical encoded features, only Kmeans was explored.

Scenario 3: Clustering Analysis on Encoded Lithology And Drilling Data

In this scenario, we explored clustering the 14 binary lithological features with the entire scaled 7 drilling features.

Evaluation Of Clustering Analysis

In each scenario, silhouette analysis was conducted in order to select the optimum number of clusters for each method (Kmeans and Kmodes). The average silhouette width gives a measure of the quality of a clustering, i.e., it gives an understanding of how well an observation point lies within its assigned cluster. Hence, the higher the average silhouette width the better the clustering. In order to evaluate the different clustering models developed, Purity was utilized as an evaluation metric. Purity is simply a clustering evaluation metric that determines the fraction of all correctly matched class and cluster labels in the entire dataset. Generally, as we increase the number of clusters, we tend to increase the percentage of purity. From the analysis conducted (figure 3-3 & 3-4), we can see that using Kmeans on the scenario 2 analysis yielded the highest purity value, hence we investigate further the cluster groups as shown in figure 3-5 to validate with domain knowledge.

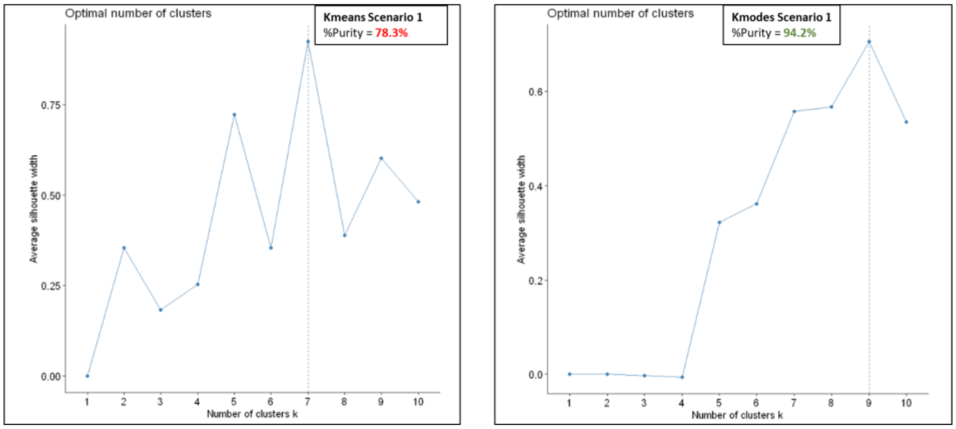


Figure 3-3: Clustering analysis of scenario 1 using Kmeans (left) and Kmodes (right)

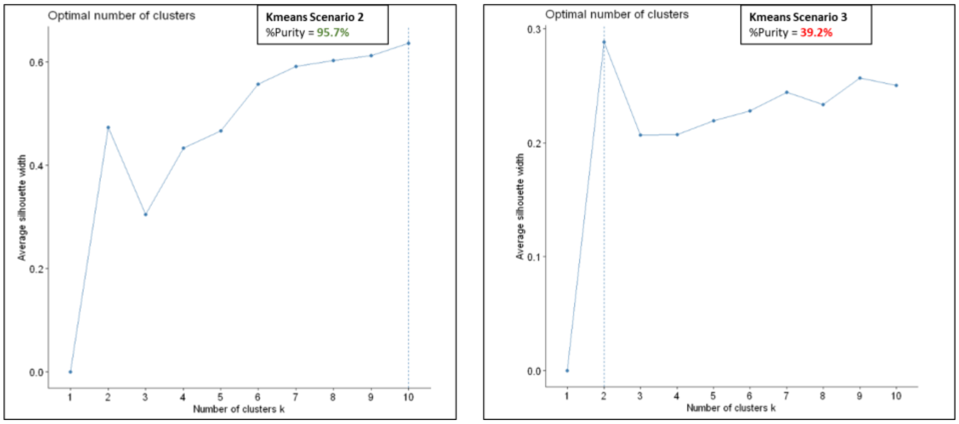
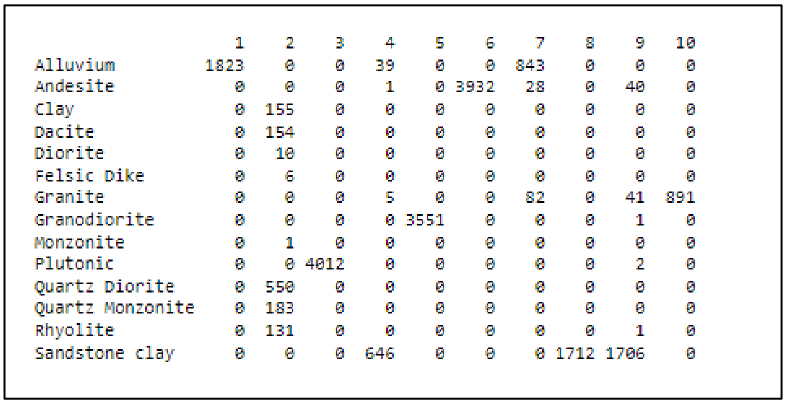


Figure 3-4: Clustering analysis of scenario 2 (left) and 3 (right) using Kmeans



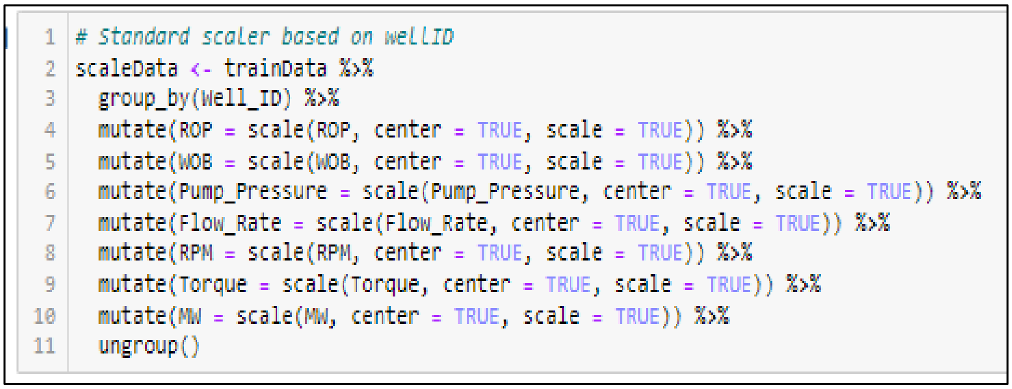
```
                     1    2    3    4    5    6    7    8    9   10
Alluvium          1823    0    0   39    0    0  843    0    0    0
Andesite             0    0    0    1    0 3932   28    0   40    0
Clay                 0  155    0    0    0    0    0    0    0    0
Dacite               0  154    0    0    0    0    0    0    0    0
Diorite              0   10    0    0    0    0    0    0    0    0
Felsic Dike          0    6    0    0    0    0    0    0    0    0
Granite              0    0    0    5    0    0   82    0   41  891
Granodiorite         0    0    0    0 3551    0    0    0    1    0
Monzonite            0    1    0    0    0    0    0    0    0    0
Plutonic             0    0 4012    0    0    0    0    0    2    0
Quartz Diorite       0  550    0    0    0    0    0    0    0    0
Quartz Monzonite     0  183    0    0    0    0    0    0    0    0
Rhyolite             0  131    0    0    0    0    0    0    1    0
Sandstone clay       0    0    0  646    0    0    0 1712 1706    0
```

Figure 3-5: Lithology clustering using best clustering algorithm

From figure 3-5, we can see that there are some levels of misclassifications in Alluvium, Andesite, Granite, and Sandstone clay. Although the efficiency of the clustering model, it is however not entirely reliable as this would impact our model development if originally misclassified labels were utilized.

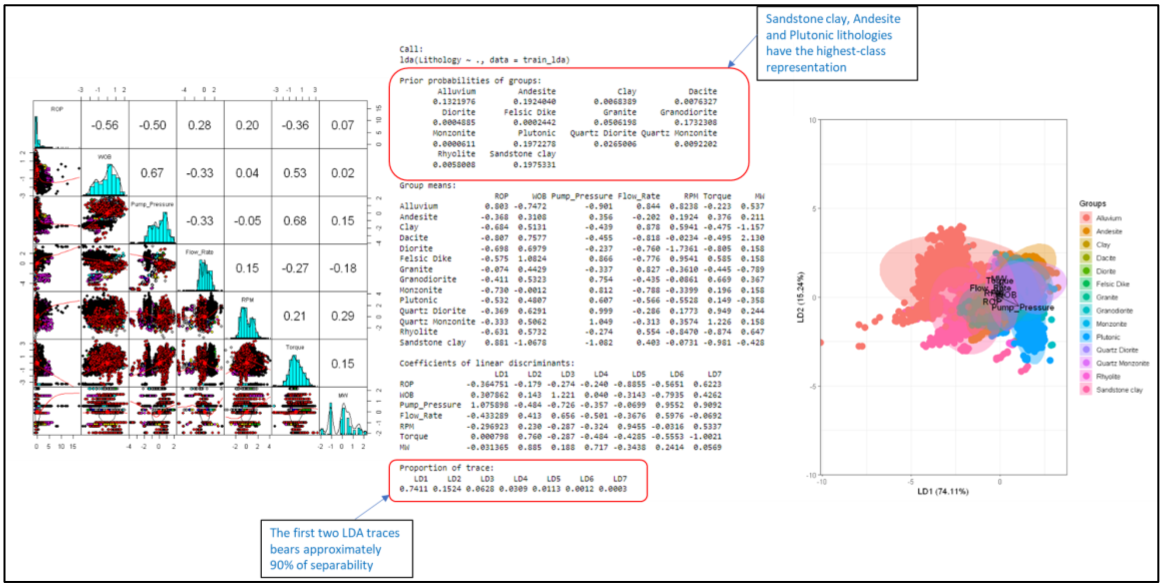
3.2.2 Linear Discriminant Analysis

The linear discriminant analysis was utilized to determine the efficiency of separability or classification using its supervised approach and the already known lithologies. However, from the defined analysis, the results were not intuitively useful as we observed an expected overlap between the variables, hence the need to find an effectively way to cluster our target levels. To use this method, we had to scale the data. Recall the dataset being used for this study, is a stacked data of drilling data from different wells, hence it is important to consider this grouping while scaling the dataset as the different wells bear different distributions that could impact the transformation of the dataset. Drilling data was grouped by “Well_IDs” and scaled to retain the diverse data distribution present in the dataset.



```r
# Standard scaler based on wellID
scaleData <- trainData %>%
  group_by(Well_ID) %>%
  mutate(ROP = scale(ROP, center = TRUE, scale = TRUE)) %>%
  mutate(WOB = scale(WOB, center = TRUE, scale = TRUE)) %>%
  mutate(Pump_Pressure = scale(Pump_Pressure, center = TRUE, scale = TRUE)) %>%
  mutate(Flow_Rate = scale(Flow_Rate, center = TRUE, scale = TRUE)) %>%
  mutate(RPM = scale(RPM, center = TRUE, scale = TRUE)) %>%
  mutate(Torque = scale(Torque, center = TRUE, scale = TRUE)) %>%
  mutate(MW = scale(MW, center = TRUE, scale = TRUE)) %>%
  ungroup()
```

Figure 3-6: Implementing Standard Scaling on Grouped Wells



Sandstone clay, Andesite and Plutonic lithologies have the highest-class representation

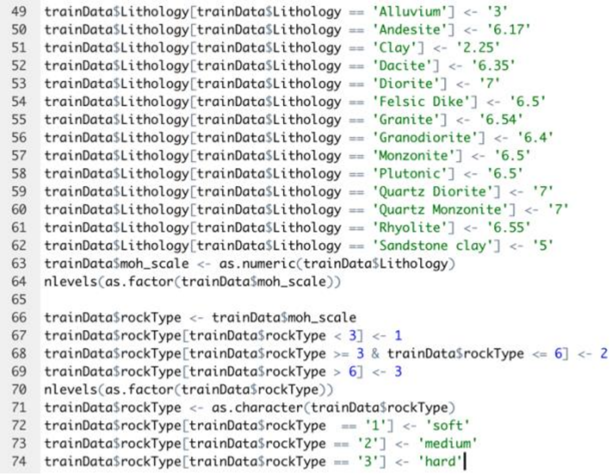
```
Call:
lda(Lithology ~ ., data = train_lda)

Prior probabilities of groups:
       Alluvium        Andesite            Clay          Dacite
      0.1321976       0.1924040       0.0068389       0.0076327
        Diorite     Felsic Dike         Granite    Granodiorite
      0.0004885       0.0002442       0.0506198       0.1732308
      Monzonite        Plutonic  Quartz Diorite Quartz Monzonite
      0.0000611       0.1972278       0.0265006       0.0092202
       Rhyolite  Sandstone clay
      0.0058008       0.1975331

Group means:
                    ROP     WOB Pump_Pressure Flow_Rate     RPM Torque     MW
Alluvium          0.803 -0.7472        -0.901     0.844  0.8238 -0.223  0.537
Andesite         -0.368  0.3108         0.356    -0.202  0.1924  0.376  0.211
Clay             -0.684  0.5131        -0.439     0.878  0.5941 -0.475 -1.157
Dacite           -0.807  0.7577        -0.455    -0.818 -0.0234 -0.495  2.130
Diorite          -0.698  0.6979        -0.237    -0.760 -1.7361 -0.805  0.158
Felsic Dike      -0.575  1.0824         0.866    -0.776  0.9541  0.585  0.158
Granite          -0.074  0.4429        -0.337     0.827 -0.3610 -0.445 -0.789
Granodiorite     -0.411  0.5323         0.754    -0.435 -0.0861  0.669  0.367
Monzonite        -0.730 -0.0012         0.812    -0.788 -0.3399  0.196  0.158
Plutonic         -0.532  0.4807         0.607    -0.566 -0.5528  0.149 -0.358
Quartz Diorite   -0.369  0.6291         0.999    -0.286  0.1773  0.949  0.244
Quartz Monzonite -0.333  0.5062         1.049    -0.313  0.3574  1.226  0.158
Rhyolite         -0.631  0.5732        -0.274     0.554 -0.8470 -0.874  0.647
Sandstone clay    0.881 -1.0678        -1.082     0.403 -0.0731 -0.981 -0.428

Coefficients of linear discriminants:
                   LD1    LD2    LD3    LD4     LD5     LD6     LD7
ROP          -0.364751 -0.179 -0.274 -0.240 -0.8855 -0.5651  0.6223
WOB           0.307862  0.143  1.221  0.040 -0.3143 -0.7935  0.4262
Pump_Pressure 1.075898 -0.484 -0.726 -0.357 -0.0699  0.9552  0.9092
Flow_Rate    -0.433289  0.413  0.656 -0.501 -0.3676  0.5976 -0.0692
RPM          -0.296923  0.230 -0.287 -0.324  0.9455 -0.0316  0.5337
Torque        0.000798  0.760 -0.287 -0.484 -0.4285 -0.5553 -1.0021
MW           -0.031365  0.885  0.188  0.717 -0.3438  0.2414  0.0569

Proportion of trace:
   LD1    LD2    LD3    LD4    LD5    LD6    LD7
0.7411 0.1524 0.0628 0.0309 0.0113 0.0012 0.0003
```

The first two LDA traces bears approximately 90% of separability

Figure 3-7: Linear Discriminant Analysis for Separability

3.2.3 Mohs Scale

In the field of Geology and Metallurgy, rocks can be classified based on its mineralization-based hardness using the Mohs scale, i.e., the more dominant a mineral is in a rock, the more the rock bears the Mohs hardness value assuming the rock is homogeneous of that mineral composition, else the total sum of the product of the fraction of individual mineral content in the rock and their corresponding hardness value, gives the hardness of the rock. In this study, we utilize the Mohs scale value of the different rock types from two sources, “911metallurgist” and “rock.comparenature” to score our Lithology.



```r
trainData$Lithology[trainData$Lithology == 'Alluvium'] <- '3'
trainData$Lithology[trainData$Lithology == 'Andesite'] <- '6.17'
trainData$Lithology[trainData$Lithology == 'Clay'] <- '2.25'
trainData$Lithology[trainData$Lithology == 'Dacite'] <- '6.35'
trainData$Lithology[trainData$Lithology == 'Diorite'] <- '7'
trainData$Lithology[trainData$Lithology == 'Felsic Dike'] <- '6.5'
trainData$Lithology[trainData$Lithology == 'Granite'] <- '6.54'
trainData$Lithology[trainData$Lithology == 'Granodiorite'] <- '6.4'
trainData$Lithology[trainData$Lithology == 'Monzonite'] <- '6.5'
trainData$Lithology[trainData$Lithology == 'Plutonic'] <- '6.5'
trainData$Lithology[trainData$Lithology == 'Quartz Diorite'] <- '7'
trainData$Lithology[trainData$Lithology == 'Quartz Monzonite'] <- '7'
trainData$Lithology[trainData$Lithology == 'Rhyolite'] <- '6.55'
trainData$Lithology[trainData$Lithology == 'Sandstone clay'] <- '5'
trainData$moh_scale <- as.numeric(trainData$Lithology)
nlevels(as.factor(trainData$moh_scale))

trainData$rockType <- trainData$moh_scale
trainData$rockType[trainData$rockType < 3] <- 1
trainData$rockType[trainData$rockType >= 3 & trainData$rockType <= 6] <- 2
trainData$rockType[trainData$rockType > 6] <- 3
nlevels(as.factor(trainData$rockType))
trainData$rockType <- as.character(trainData$rockType)
trainData$rockType[trainData$rockType  == '1'] <- 'soft'
trainData$rockType[trainData$rockType == '2'] <- 'medium'
trainData$rockType[trainData$rockType == '3'] <- 'hard'
```

Figure 3-8: Assigning Mohs hardness value to lithology types and grouping them as hard/soft/medium

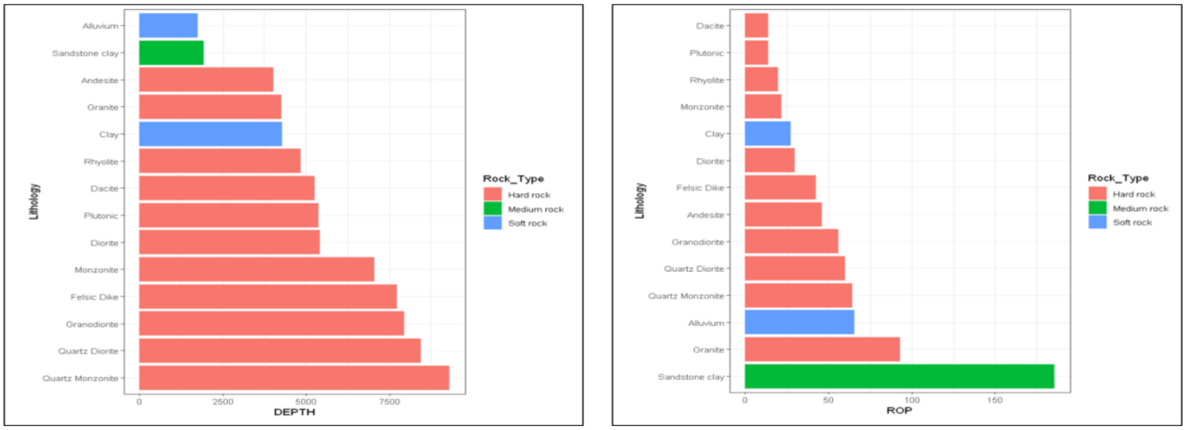


Figure 3-9: Hardness based rock types with respect to Depth and ROP

The Mohs scale classifies rocks of less than 3 to be soft rocks, rocks with hardness value of 3-6 to be medium rocks while rocks with hardness value greater than 6 to be hard rocks. These classification was implemented on the assigned Mohs scale to determine the rock hardness classification, which will be utilized as our target variable in this analysis. The above figure gives a better understanding of the classification of the different rocks, as the left plots infers that harder rocks are mostly found at the deeper section of the well while softer or medium rocks can be found top to mid sections of the well. These inference is further validated in the right plot, which shows that soft and medium rocks are drilled at a faster ROP, while harder rocks are drilled at low ROP. The unexpected trend identified in clay was confirmed from drilling reports to be attributed to clay swelling which is a known drilling problem in clay formation. This causes the ROP to be low for soft rocks (clay)

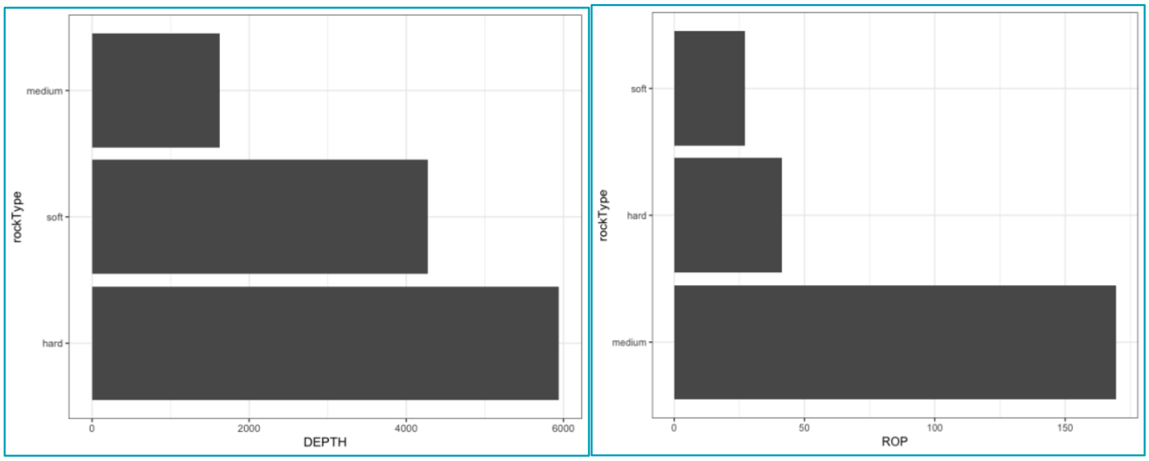


Figure 3-10: Transformed Hardness based rock types with respect to (left) Depth and (right) ROP

3.3 Modeling and Model Evaluation

Five machine learning models were trained using the clean and scaled drilling data (presented above). A multinominal logistic regression (MLR) model, a K-Nearest Neighbor (KNN) model, a support vector machine (SVM) model, an extreme gradient boosted (XGB) tree model, and a random forest model were developed to predict the rock type classification from drilling data. Hyperparameters for each model were tuned using 5-fold of cross-validation, over a range that was deemed relevant for each hyperparameter. The *train* function from the *caret* package was utilized for training and tuning the models and the best hyperparameters were chosen based on accuracy and kappa metrics. We also reserved 15% of the dataset for holdout validation to test the model on unseen data (Train/Test split of 85/15). The models were evaluated based on accuracy for cross validation because we have more than two classes in the target variable. We paid attention to the model sensitivity and specificity as well.

3.3.1 Multinomial Logistic Regression

The supervised learning algorithm of logistic regression is extended to make classification predictions for outcomes with more than two classes with MLR. The *multinom* method from the *nnet* package was used to develop the MLR model. The weight decay hyperparameter was tuned to improve the model, which regulates the model by applying weights to excessive parameters to prevent the model from responding to noise in the data and prevent over-fitting.

The best MLR model accuracy was obtained using a weight decay value of 1.3. Although an accuracy of almost 90% was obtained with this model, Figure 17 and 18 shows that there is a large bias toward predictions for hard rock types when trying to classify soft rocks. For this reason, the model has a poor ability to predict soft rock types and struggles to differentiate medium and hard rock types. This model poorly predicts soft rocks

3.3.2 K-Nearest-Neighbor

We used K-Nearest Neighbor classification to attempt the classification of these three classes of rock hardness using the caret package and tuning k- the number of neighbors. The results are shown below

The best KNN model accuracy was obtained using number of neighbors k = 5. We obtained over 90% accuracy, sensitivity and specificity indicating good model performance.

3.3.3 Support Vector Machines

Classifications are distinguished using SVM by mapping the data to a higher-dimensional feature space and creating a hyperplane between different categories. The higher-dimensional feature mapping is performed using various SVM kernel functions to transform the training data. Three different kernels were utilized to develop the SVM models using the *svmKernel* method from the *kernlab* package and were tuned to their specific hyperparameters.

Linear SVM Kernel

The linear SVM kernel uses the function $K(x_i, x_j) = x_i^T x_j$ to transform the training data. The cost hyperparameter was tuned to improve the model, which dictates the margin from the hyperplane used for classification by weighting the points that fall within that margin.

The best SVM model accuracy using a linear kernel was obtained using a cost value of 100. Although an accuracy of about 91% was obtained with this model, There were still considerable misclassifications present as seen in the confusion matrix (Appendix).

Polynomial SVM Kernel

The polynomial SVM kernel uses the function $K(x_i, x_j) = (1+x_i^T x_j)^p$ to transform the training data. The cost and degree hyperparameters were tuned to improve the model. The degree parameter sets the power of polynomial curve used for the hyperplane.

The best SVM model accuracy was obtained using a polynomial kernel was obtained with a cost value of 0.01 and a 5-degree polynomial. Using the polynomial kernel produced better results than the linear kernel with fewer misclassifications, less apparent bias, and an accuracy of 99.1% on data set aside for testing.

Radial SVM Kernel

The radial SVM kernel uses the function $K(x_i , x_j) = exp[-\sigma\Sigma(x_i-x_j)^2]$ to transform the training data. The sigma hyperparameter was tuned to improve the model, which dictates the boundary of the hyperplane used for classification to control the model variance.

The best SVM model accuracy using a radial kernel was obtained with a cost value of 10 and a sigma of 1. The radial kernel performed the best of any of the three SVM models with less bias and misclassifications and an accuracy of 99.4% on test data.

3.3.4 Tree Based Bagging and Boosting

Random Forest

Random forest is a decision tree method that consists if many decision trees such that they operate in an ensemble. Each individual tree gives a class prediction and the class with the most votes becomes the model's prediction. We used the caret package to build our XGBoost model and tuned the mtry hyperparameter

The best Random Forest model accuracy of 99.7% was obtained with mtry = 2 with very good discrimination between the classes as seen in Figure 4-2 through 4-4.

XGBoost

Extreme gradient boosting is a decision tree method that uses penalty weights to regulate the model and a gradient descent algorithm to maximize accuracy much faster than other decision tree methods. There are many tuning parameters available with XGBoost. The *xgbTree* method from the *xgboost* package was used to develop the XGB tree model. The number of trees (nrounds), maximum tree depth, and the shrinkage (the learning rate step size) hyperparameters were used to tune the XGB tree model.

The best XGBoost model accuracy was obtained with 300 rounds, a max depth of 4, and a shrinkage rate of 0.25. The XGBoost model had a 99.6% prediction accuracy on test data, a marginal improvement over the SVM model using a radial kernel but slightly underperforming the random forest model. The XGBoost model classified all the soft rocks accurately and differentiates between the medium and hard rock types very well, with few misclassifications and no apparent bias.

4. Results

4.1 Model Performance Summary

The RF tree model performed best of all the models developed to predict rock types from the drilling data, followed closely by some of the SVM models. A summary of all the models and their performance metrics is provided in Table 2.

Table 2. Summary of model performance.

Model	Method	Package	Hyperparameter	Selection	CV Performance		Test Performance - Sensitivity			Test Performance - Specificity		
					Accuracy	Kappa	Hard	Medium	Soft	Hard	Medium	Soft
MLR	multinom	nnet	weight decay	1.3	90.30%	0.781	94.14%	85.99%	5.88%	83.64%	94.18%	99.96%
KNN	KNN	caret	k	5	99.01%	0.978	99.76%	98.56%	97.06%	98.61%	99.72%	100.00%
Linear SVM	svmLinear	kernlab	cost	100	91.00%	0.799	94.09%	86.95%	82.35%	86.99%	94.18%	99.93%
Polynomial SVM	svmPoly	kernlab	cost, degree	0.001, 5	98.90%	0.975	99.71%	97.89%	97.06%	98.14%	99.67%	99.94%
Radial SVM	svmRadial	kernlab	cost, sigma	10, 1	99.20%	0.982	99.90%	98.37%	97.06%	98.51%	99.86%	99.97%
Random Forest	rf	caret	mtry	2	99.30%	0.985	99.95%	99.14%	97.06%	99.16%	99.91%	100.00%
XGBoost	xgbTree	xgboost	nrounds, maxdepth, shrinkage	300, 4, 0.25	99.50%	0.989	99.86%	99.23%	97.06%	99.26%	99.86%	99.97%

4.2 Key Findings from Analysis

Below is a plot of variable importance from our best Random Forest Model alongside the confusion matrix and detailed performance on test data:

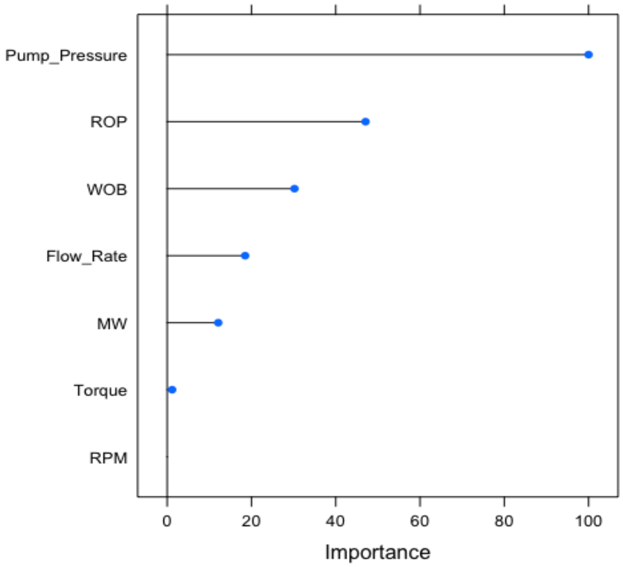
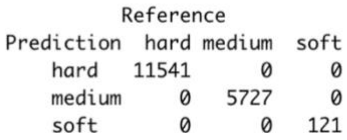
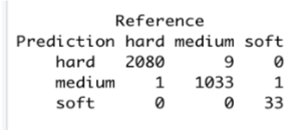


Figure 4-1: Random Forest Variable importance plot



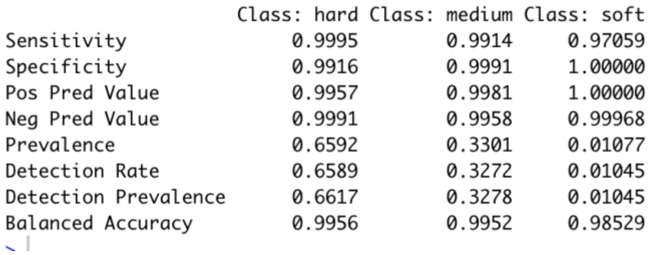
```
          Reference
Prediction  hard medium  soft
    hard   11541      0     0
    medium     0   5727     0
    soft       0      0   121
```

Figure 4-2: Random Forest confusion matrix on train data



```
          Reference
Prediction hard medium soft
    hard   2080      9    0
    medium    1   1033    1
    soft      0      0   33
```

Figure 4-3: Random Forest confusion matrix on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9995        0.9914     0.97059
Specificity               0.9916        0.9991     1.00000
Pos Pred Value            0.9957        0.9981     1.00000
Neg Pred Value            0.9991        0.9958     0.99968
Prevalence                0.6592        0.3301     0.01077
Detection Rate            0.6589        0.3272     0.01045
Detection Prevalence      0.6617        0.3278     0.01045
Balanced Accuracy         0.9956        0.9952     0.98529
```

Figure 4-4: Random Forest Model Evaluation Results

From the variable importance plot, we see that the pump pressure, then ROP are the most significant contributors to predicting the hardness of the rock. Recall that from our exploratory data analysis, we found that rocks of medium hardness exert very low pressures on the pump (< 1500 psi), this is followed by soft rocks and hard rocks. This confirms our hypothesis that the pump pressure is a good indicator of rock hardness. We also see that the random forest handled the high correlation between the Pump Presssure and the Torque by essentially using predominantly one of them to determine the hardness alongside other factors. The Number of drillbit rotations per minute is also not a good predictor of rock hardness.

5. Conclusion

We followed standard CRISP-DM in executing this project. We successfully built a model that predicts the rock hardness using drilling parameters. This tool could arm drilling engineers with valuable information that will aid them in proper tool selection when drilling geothermal wells.

Our approach started by first understanding the problem and the dataset. We came up with a data dictionary and a data quality report that helped us assess the missingness of our data and the distribution of the data in each column. We proceeded to prepare the dataset for modelling by removing outliers by capping values based on the upper and lower bounds of their boxplot distributions. We also imputed the missing values in the MW distribution my using mean imputation technique due to its very narrow distribution.

The next task was to identify our target variable. Predicting lithology would be an unreaslistic target because of the high number of unique variables in that dataset. After exploring different methods (including Clustering and LDA analysis) we ended up referring to literature and a concept known as the Mohs' Scale. This method classifies rock hardness based on the dominant mineral in the rocks. We obtained Mohs' scale values for the varying lithologies and grouped them into hard/medium/soft based on these values. The Mohs scale classifies rocks of less than 3 to be soft rocks, rocks with hardness value of 3-6 to be medium rocks while rocks with hardness value greater than 6 to be hard rocks. With this, we now had target data to predict and could proceed to modelling.

After scaling the dataset using the standard scaler method to address the large disparity in the scales of the predictors, we proceeded to try our 5 kinds of models to predict hardness. These models were the Multinomial Logistic Regression, the K-Nearest Neighbors model, Support Vector Machines (Linear, Polynomial and Radial), Random Forest Ensemble Model and XGBoost gradient boosting model. We used two validation methods for comparison – 5-fold cross validation and holdout validation, where we trained the model with 85% of the dataset and tested its performance on the remaining 15%. After training, validation and testing, random forest provided the best results across different metrics including Accuracy, Kappa, Sensitivity and Specificity cross all three classes of Rock Types. We were able to see from the model that the best indicator of hardness in a rock is the Pump Pressure with lower pressures indicative of medium rock hardness. Very high pump pressures and low penetration rates are usually seen in hard rocks while clay swelling usually causes high pump pressure and low rate of penetration in clay (soft rock).

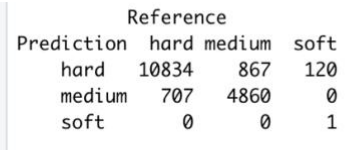
The main limitation we have we had in this project was having insufficient data for soft rocks. The data was highly skewed in favor of high and medium rocks with the only soft rocks available being clay. The ROP, Pump pressure and torque trends we observed for clay may not be representative for another soft rock. Thus, as a final recommendation, future work should be done to identify more records of soft rocks that are preferably not clay.

6. References

- Geothermal Data Repository (GDR) website (https://gdr.openei.org)
- https://rocks.comparenature.com/en/monzonite-rock/model-53-0
- https://www.911metallurgist.com/hardness-toughness-rocks/

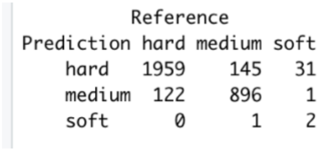
7. Appendix

7.1 Multinomial Linear Regression Performance



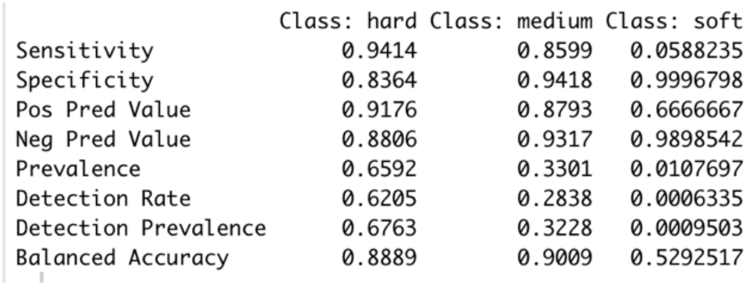
```
          Reference
Prediction  hard medium  soft
    hard   10834    867   120
    medium   707   4860     0
    soft       0      0     1
```

Figure 7-1: MLR performance on train data



```
          Reference
Prediction hard medium soft
    hard   1959    145   31
    medium  122    896    1
    soft      0      1    2
```

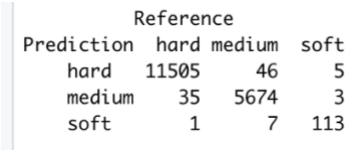
Figure 7-2: MLR performance on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9414        0.8599   0.0588235
Specificity               0.8364        0.9418   0.9996798
Pos Pred Value            0.9176        0.8793   0.6666667
Neg Pred Value            0.8806        0.9317   0.9898542
Prevalence                0.6592        0.3301   0.0107697
Detection Rate            0.6205        0.2838   0.0006335
Detection Prevalence      0.6763        0.3228   0.0009503
Balanced Accuracy         0.8889        0.9009   0.5292517
```

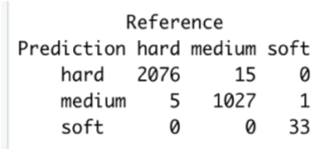
Figure 7-3: MLR General Model evaluation Results

7.2 KNN Classification Performance



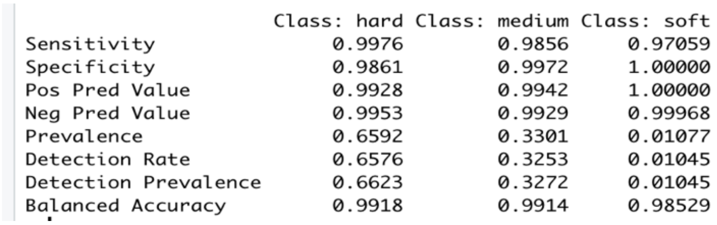
```
          Reference
Prediction  hard medium  soft
    hard   11505     46     5
    medium    35   5674     3
    soft       1      7   113
```

Figure 7-4: KNN performance on train data



```
          Reference
Prediction hard medium soft
    hard   2076     15    0
    medium    5   1027    1
    soft      0      0   33
```

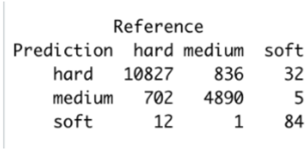
Figure 7-5: KNN performance on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9976        0.9856     0.97059
Specificity               0.9861        0.9972     1.00000
Pos Pred Value            0.9928        0.9942     1.00000
Neg Pred Value            0.9953        0.9929     0.99968
Prevalence                0.6592        0.3301     0.01077
Detection Rate            0.6576        0.3253     0.01045
Detection Prevalence      0.6623        0.3272     0.01045
Balanced Accuracy         0.9918        0.9914     0.98529
```

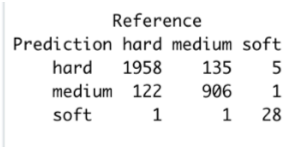
Figure 7-6: KNN general model evaluation results

7.3 Linear SVM Classification Performance



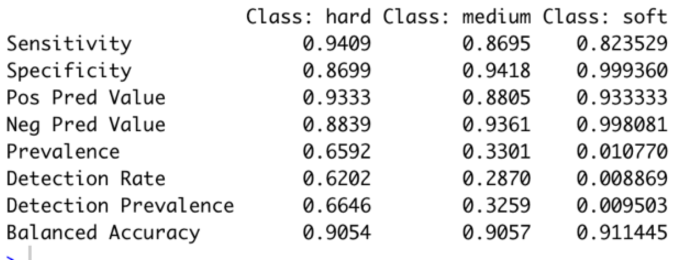
```
          Reference
Prediction  hard medium  soft
    hard   10827    836    32
    medium   702   4890     5
    soft      12      1    84
```

Figure 7-7: Linear SVM performance on train data



```
          Reference
Prediction hard medium soft
    hard   1958    135    5
    medium  122    906    1
    soft      1      1   28
```

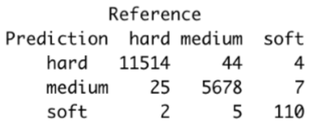
Figure 7-8: Linear SVM performance on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9409        0.8695    0.823529
Specificity               0.8699        0.9418    0.999360
Pos Pred Value            0.9333        0.8805    0.933333
Neg Pred Value            0.8839        0.9361    0.998081
Prevalence                0.6592        0.3301    0.010770
Detection Rate            0.6202        0.2870    0.008869
Detection Prevalence      0.6646        0.3259    0.009503
Balanced Accuracy         0.9054        0.9057    0.911445
```

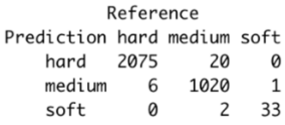
Figure 7-9: Linear SVM general model evaluation results

7.4 Polynomial SVM Classification Performance



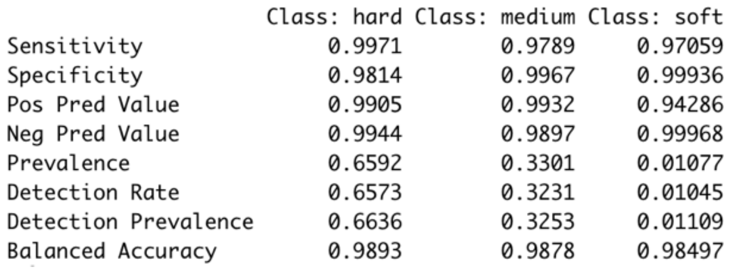
```
          Reference
Prediction  hard medium  soft
    hard   11514     44     4
    medium    25   5678     7
    soft       2      5   110
```

Figure 7-10: Polynomial SVM Performance (Confusion Matrix) on train data



```
          Reference
Prediction hard medium soft
    hard   2075     20    0
    medium    6   1020    1
    soft      0      2   33
```

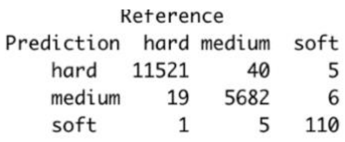
Figure 7-11: Polynomial SVM Performance (Confusion Matrix) on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9971        0.9789     0.97059
Specificity               0.9814        0.9967     0.99936
Pos Pred Value            0.9905        0.9932     0.94286
Neg Pred Value            0.9944        0.9897     0.99968
Prevalence                0.6592        0.3301     0.01077
Detection Rate            0.6573        0.3231     0.01045
Detection Prevalence      0.6636        0.3253     0.01109
Balanced Accuracy         0.9893        0.9878     0.98497
```

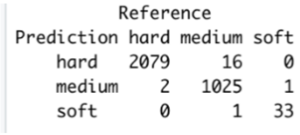
Figure 7-12: Polynomial SVM general model evaluation results

7.5 Radial SVM Classification Performance



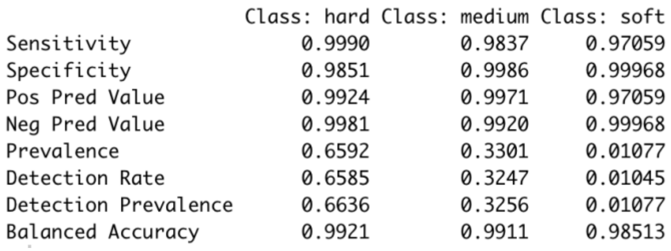
```
          Reference
Prediction  hard medium  soft
    hard   11521     40     5
    medium    19   5682     6
    soft       1      5   110
```

Figure 7-13: Radial SVM Performance (Confusion Matrix) on train data



```
          Reference
Prediction hard medium soft
    hard   2079     16    0
    medium    2   1025    1
    soft      0      1   33
```

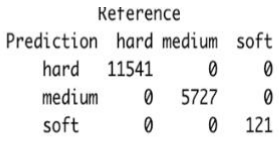
Figure 7-14: Radial SVM Performance (Confusion Matrix) on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9990        0.9837     0.97059
Specificity               0.9851        0.9986     0.99968
Pos Pred Value            0.9924        0.9971     0.97059
Neg Pred Value            0.9981        0.9920     0.99968
Prevalence                0.6592        0.3301     0.01077
Detection Rate            0.6585        0.3247     0.01045
Detection Prevalence      0.6636        0.3256     0.01077
Balanced Accuracy         0.9921        0.9911     0.98513
```

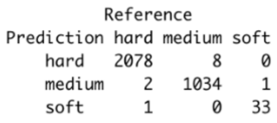
Figure 7-15: Radial SVM general model evaluation results

7.6 XGBoost Classification Performance



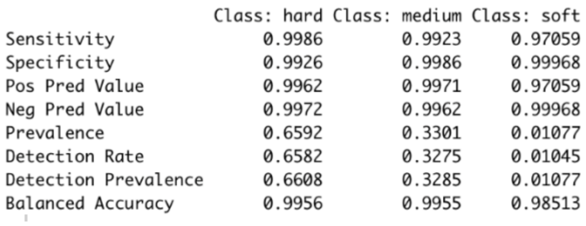
```
          Reference
Prediction  hard medium  soft
    hard   11541      0     0
    medium     0   5727     0
    soft       0      0   121
```

Figure 7-16: XGBoost Performance (Confusion Matrix) on train data



```
          Reference
Prediction hard medium soft
    hard   2078      8    0
    medium    2   1034    1
    soft      1      0   33
```

Figure 7-17: XGBoost Performance (Confusion Matrix) on test data



```
                     Class: hard Class: medium Class: soft
Sensitivity               0.9986        0.9923     0.97059
Specificity               0.9926        0.9986     0.99968
Pos Pred Value            0.9962        0.9971     0.97059
Neg Pred Value            0.9972        0.9962     0.99968
Prevalence                0.6592        0.3301     0.01077
Detection Rate            0.6582        0.3275     0.01045
Detection Prevalence      0.6608        0.3285     0.01077
Balanced Accuracy         0.9956        0.9955     0.98513
```

Figure 7-18: XGBoost general model evaluation results