**Title: Evaluating Clustering Algorithms for Prediction of Rock Type for Oil and Gas Applications using United Kingdom Core Data**

**Autho**rs: Munish Kumara,b and Tan Wen Xuanb

**Affiliations:** aERCE Singapore, 48 Tras Street, #03-01, Singapore 078987

bSingapore University of Social Sciences, School of Business,463 Clementi Rd, Singapore 599494

Corresponding author. Email: mkumar@erce.energy

**Abstract:** Rock typing is important in the oil and gas industry when it come to locating optimal drilling spots wells for optimization. Studies on rock typing mainly revolves around rock typing indices as methods to rock type, with newer studies combining machine learning as well. However, comparison of unsupervised machine learning for rock typing against different algorithms and against a modified traditional method of rock typing has not been widely researched. This paper aims to compare the performance between 5 unsupervised machine learning models against a modified traditional rock typing method. The data for model training is the UK core data with 2000 data points. The modified traditional rock typing method is derived from the IMLR clustering technique and is a benchmark for the results of the unsupervised machine learning model. The 5 unsupervised machine learning model are the KMeans, SOM + KMeans, DBSCAN, BIRCH and GMMs which are chosen from different family of clustering. The study’s results showed that KMeans and BIRCH rock typing performance are the best aside from the modified method when applied on the UK core dataset.

**One-Sentence Summary:** This paper aims to explore the performance of rock typing using different unsupervised machine learning algorithms against a modified traditional rock typing method.

**Keywords (minimum 6):** Oil and Gas, Rock Typing, Unsupervised Machine Learning, KMeans, Density-Based Spectral Clustering of Application with Noise (DBSCAN), Self-Organising Map (SOM), Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), Gaussian Mixture Models (GMMs).

1. **Introduction**

Rock typing is the characterization of reservoir rocks into similar groups that are petrophysical similar (Hollis et. al., 2020). Much research has been done on rock typing reservoir rocks of reservoirs for determining optimal rock types for drilling. Rock Typing studies typically consist of the application of indices like FZI or FZI\* onto data that uses additional data alongside core data like well logs data. Drilling wells in sedimentary basins is the main method of obtaining oil and gas after processing. Prior the production phase of oil and gas, rock samples are extracted from various areas of the well the determine permeability and porosity, which are flow capacity of rock and storage capacity. Rock typing is used to determined areas with medium to high porosity and permeability to optimize drilling efforts by generalizing rock types of the rock samples for the whole well.

Rock typing studies mostly uses rock typing indices on many different variations of data for rock typing. There are studies that usage of machine learning is included in their methodology to complement the indices to rock type. Some studies even use machine learning solely without the indices. However, a limitation on the studies is that it mostly uses supervised machine learning and a mixture of data which can be expensive to acquire.

The comparison of multiple unsupervised machine learning methods for rock typing against the traditional method of rock typing on core is something that has not been researched extensively and hence there is value for research.

The literatures on rock typing historically focuses on 3 kinds, traditional rock typing, rock typing utilizing machine learning combined with indices and rock typing solely with machine learning. Rock typing indices are formulas that experts have come up with to perform rock typing on various data like the core, well logs, and others like the SCAL data. In the Table below we can see some common indices and indices that are touched upon in literatures studied.

**[Table of equations] #In appendix**

The literatures on rock typing using indices are the more common literatures of rock typing. Faramarzi-Palanga and Mirzaei-Paiaman (2021) used FZI, FZI\*, FZIM and MFZI indices on Special Core Analysis (SCAL) data to find flow. Shahat et. al. (2021) proposed a new approach using well-logs data, the Resistivity Zone Index (RZI). Soleymanzadeh et. al. (2019) explores variations of rock typing under variating pressure using the FZI, FZI\* and Winland method.

In recent years, machine learning used to complement indices for rock typing have been on the rise, using both supervised and unsupervised machine learning algorithms. In the oil and gas industry, machine learning research is more focused on anomaly detection, production, energy efficiency, risk, and failure detection. There are fewer studies on machine learning for rock typing on core data comparatively to the traditional methods. Mohammadian et. al. (2022), approached rock typing FZIM\* index with supervised and unsupervised algorithms on core data where performance was measured against the FZI and FZI\* methods. Zakyan et. al. (2022), proposed method of rock typing log data with artificial neural network as an enhanced clay typing methodology. The limitation however is that the methodology is only for log data.

Rock typing with only machine learning is also something studied in the recent years. Mohamed et. al. (2019), performed rock typing using unsupervised and supervised machine learning algorithms. It was performed on the well-logs data and models selected are the KNN, SVM, Random Forest, Neural Network and K-Means.

Other than the above literatures, to this author’s knowledge, rock typing using solely machine learning on core data is an area that has not been widely research in the oil and gas industry.

This paper will take a different angle by comparing the performance of 5 selected unsupervised machine learning algorithms used for rock typing against a benchmark method which is a modified traditional rock typing method based on the IMLR technique.

1. **Scope & Methods**

This paper will rock type using traditional method and unsupervised machine learning algorithms and compare their performance using the traditional method as a benchmark. This will be a methods paper which should be applied and tested on other datasets. The dataset used for model training is the UK core dataset with 2000 data points from the original 100k data points dataset obtained from an online source. The algorithms selected are K-Means, Self-Organizing Map with K-Means, DBSCAN, BIRCH and GMMs. The objective of the paper is to evaluate performance of the unsupervised machine learning models against the benchmark model to see how well they fare against when rock typing on this set of core data. Ideally, the method should be further researched using more algorithms and dataset.

* 1. **Data Preparation and Data Understanding**

A total of eleven regions’ core data is prepared and retrieved from the following sources: [(Data Underground, n.d.), (GeoProvider, n.d.), (UK National Data Repository, n.d.)]. The regions are Africa, Australia, Brazil, Myanmar, the North Sea, Pakistan, Papua New Guinea (PNG), the UK, the USA, China, and the Philippines. The modeling is trained with the UK core dataset prepared in this step of the study.

Before data understanding, merging of the dataset and selection of appropriate columns must be done due to the number of files there are and, in some datasets, the number of columns there are. Some of which, these files number about 2000 files, and some datasets have over 200 columns. Moreover, most of the data column and rows in most files is empty and is not useful. Hence it is important to refine the dataset before moving on to data understanding and cleaning.

The general data preparation process is shown below in Figure 1. Separate files are firstly compiled and compiled into a single file for the different regions. The data columns and rows are selected based null values, problematic data as well as basic knowledge of the critical columns related to rock typing. The file’s data is then adjusted by filling empty data and data type are adjusted. The outliers are then removed to prevent skew of data from affecting the overall model training and results. The data is then min max normalized. Additional columns that would be used for the modified method’s way of rock typing is then created. The output from the data preparations would be 2 csv files, one being the unnormalized file and the other being the min max normalized file with the additional columns. The process is iterative and would repeat due to errors caused by calculation of additional columns like zero division error.

Figure 2: Generic data preparation flowchart

* + 1. **UK Dataset Initial Preparation**

For the UK dataset, there were close to two thousand files that needs to be merged. The format of the column was similar in the separate files which were confirmed through random sampling. To merge the separate files into a single dataset, we need to shift subfolder files all into a single location before running a code to merge the data. For the UK dataset, the process was the most tedious as it has multiple files with all sorts of issues, the other region’s dataset preparation was much more simplified.

For the UK dataset preparation, the two thousand files were all in their subfolders. The first step was to shift all the files with the file extension of .xls and .xlsx into a single location via a code to extract out from the individual subfolders. Then the .xls files were then placed in one location and .xlsx in another for merging. The .xls files were merged into a single file and the .xlsx files were merged into a single file separately. The corrupted file was removed using trial and error to remove corrupted files when merging both file types. Both the resulting file was saved as a .csv file. The two files were then merged into a single .csv file. At this point, the initial columns were selected by manually looking at the column headers as well as using a code to observe the total null value for each header.

Overall, the steps for preparing the data for the eleven regions vary from one another but the process is mainly the merging of files or sheets and the initial selection of columns. Following the data preparation, we can then move on to data understanding.

* + 1. **UK Dataset Understanding**

Figure 3 shows the general distribution of the UK dataset. Figure 4 shows the histogram chart of the UK dataset porosity values and Figure 5 shows the histogram chart of the UK dataset permeability values. General understanding of the dataset was also understood like the data type, null value total and statistical data.

Chart, scatter chart

Description automatically generated

Figure 3: Porosity Permeability scatterplot of the UK dataset

Chart, histogram

Description automatically generated

Figure 4: Porosity histogram of the UK dataset

Chart

Description automatically generated with medium confidence

Fig 5: Permeability histogram of the UK dataset

* 1. **Methods**

In the study, training of models was done using the normalized UK core dataset and visualization was done using the unnormalized dataset by tagging cluster results obtained from training of the models.

* + 1. **Selection of dataset sizing for modelling**

The original UK core dataset contained over 100k rows which is very large for training of models. To determine optimal sizing of datasets that saves training time yet not sacrifice results too much, K-Means algorithms was used. 3 separate sized dataset was tested using K-Means algorithm, 2000, 40000, full sized data point dataset. Random state was fixed, and initialization of K-Means algorithm was set to k-means++ for range of 1 to 10 clusters. An elbow plot was then used to determine the optimal number of clusters and then for 3 datasets, the optimal cluster is 4 based on the individual inertia plots. Figure 6 shows the combined inertia plot of the 3 unnormalized dataset.

Chart, line chart

Description automatically generated

Fig 6: K-Means Inertia Plot for the 2k, 40k and whole dataset normalized data

Each of the 3 datasets was run using K-Means with the optimal cluster derived from inertia chart, the labels obtained from the model training were tagged back to the corresponding unnormalized data. Charts were then plotted to visualize the cluster output for the 3 datasets using the 1000 random samples with same random state. The x-axis is porosity and y-axis are log-scaled permeability. The result shown in Fig 7 shows similarity of clustering results and hence the 2000 rows dataset is selected for computational efficiency. The reduction in data improves model training time and not overly affecting the results to a large extent. Both the normalized data and the un-normalized one is used to derive the smaller sets of 2000 rows with both sampling at the same random state.

Chart, scatter chart

Description automatically generated Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Fig 7: Porosity vs Log-Scaled Permeability Scatter Plots of K-Means for Different Sample Sizes of the Un-Normalized Dataset

* + 1. **Traditional Rock Typing and Failure of its Application**

We attempted to rock type using the traditional rock typing method by referencing two techniques used, the iterative multi-linear regression (IMLR) clustering technique and the analysis by least square regression method (Khalid et al., 2019; Abdulelah et al., 2018) and using either technique. In Khalid et al. (2019). The IMLR technique was one of the more accurate rock typing techniques. We failed to apply the techniques but failed as data points do not have clear separation of a Log RQI (y-axis) and Log Porosity Index (x-axis) scatterplot where points that lie on the same line with slope being 1 will indicate the points having similar hydraulic flow units. Which this indicates that they are likely the same rock type. In Fig 8 shows a Log RQI and Log Porosity Index scatterplot where there are no clear lines of slope = 1 to segregate the datapoints by.

Chart, scatter chart

Description automatically generated

Fig 8: Log Porosity Index vs Log RQI Scatterplot of the Normalized Training Dataset

* + 1. **Modified Rock Typing Method**

We came up with a modified version that is with reference to the two techniques previously mentioned. This method is used as a benchmark to compare the results from the machine learning algorithms with. The normalized and unnormalized dataset is first sorted by ascending FZI which is accomplished in the data preparation stage. The process flow can be seen in fig 9. The method is adapted from the IMLR and the least square regression method and adjusted to become the modified rock typing method used for benchmarking. Firstly, the log porosity index (y-axis) and log RQI (x-axis) scatterplot is plotted. Then from range of 1 till 10, the dataset is split into parts based on the loop number. Each mini dataset is used to train, fit, and predict linear regression and the average r-squared value is recorded. A r-squared vs number of clusters plot is plotted, and the optimal cluster is chosen. The method repeats using the optimal cluster number and the cluster labels are tagged back to the unnormalized data for visualization and label summaries are generated.

Diagram

Description automatically generated

Fig 9: Process of the Modified Rock Typing Method

* + 1. **K-Means**

K-Means was chosen as it is one of the most widely used clustering methods. It is a centroid-based clustering method, which is a simple yet efficient algorithm. For K-Means, an initial number of clusters must be inputted where random starting points will be initialized (Sharma, 2022). Those points would act as cluster centroids and the data points would assign to the nearest cluster and the center point will shift to be the average of all the data points in that cluster (Sharma, 2022). The process repeats until all data points belong to a cluster (Sharma, 2022). An important step is that the cluster needs to be chosen beforehand and this was done by fitting K-Means with multiple K-clusters and compare with an inertia plot. The process applied for our K-Means model training is as follows in Fig 10.

Diagram

Description automatically generated

Fig 10: Process of K-Means Method

* + 1. **SOM + K-Means**

A Self-Organizing Map (SOM) is an artificial neural network that uses competitive learning to weight adjust weights in neurons (Ralhan, 2018). We built the SOM using the MiniSOM library (Vettigli, 2018). We initialized a large SOM grid of 10X10 with default sigma and learning rates before applying K-Means to get a smaller number of clusters. The optimal cluster was decided with the inertia plot. Below is a simple flow of the process.

Diagram

Description automatically generated

Fig 11: Process of SOM+K-Means Method

* + 1. **Density-Based Spatial Clustering of Application with Noise (DBSCAN)**

DBSCAN is a clustering algorithm that belongs to the density-based clustering family (Kumar, 2021). It clusters the data points together based on a certain defined maximum distance between two points called epsilon (Kumar, 2021). Those that are sufficiently close together belong to a single cluster and those points that don’t have any data points near the defined epsilon are considered outliers (Kumar, 2021). In our case, we chose the outlier points to be a cluster instead of noise due to the points coming from a certain area. There are two parameters required for the DBSCAN algorithm, epsilon, and min samples, where epsilon is the max distance between points allowed to be considered as part of a cluster, and min samples are the minimum points that are required for a formation of a cluster (Kumar, 2021). The nearest neighbor of 4 for all data points was used to find the average distance between each point from its 4 nearest neighbors (Kumar, 2021). The average distance of the points was plotted in ascending order and the point of the greatest bend is used as the epsilon value (Kumar, 2021). The kneed package was used to identify that point. The knee is the point of the greatest bend to be used as the epsilon value for DBSCAN. Fig 12 shows the knee point of the dataset to and Fig 13 shows the process flow of DBSCAN.

A picture containing chart

Description automatically generated

Fig 12: Elbow Plot with Knee Point Identified by Kneed for Epsilon Value

Diagram

Description automatically generated

Fig 13: Process of DBSCAN Method

* + 1. **Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH)**

The BIRCH method is a hierarchical clustering method that generates a small summary of the large dataset to cluster with as much information retained as possible which is referred to as a CF Tree (Verma, 2021). The three parameters needed are threshold, branching factor, and n \_cluster (Verma, 2021). The threshold represents the number of data points leaf node sub-clusters can have in a CF tree, the branching factor represents how many sub-clusters can a leaf node have and the n\_clusters represent how many clusters we require (Verma, 2021).

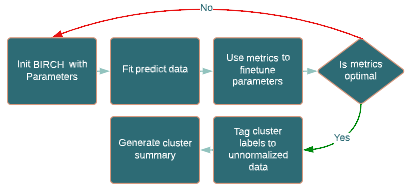


Fig 14: Process of BIRCH Method

* + 1. **Gaussian Mixture Models (GMMs)**

GMMs assume that the data consists of several Gaussian distributions and for each data, the points are assigned to a cluster based on probabilistic methods (Singh, 2022). For example, a data point that lies between two clusters has a different probability of being in either one using the GMMs (Singh, 2022). It assigns the data to the cluster it most likely belongs to. GMMs work like K-Means but the clusters can be non-circular, unlike K-Means (Singh, 2022). The initialization is selected as K-Means and the number of components is determined using Bayesian Information Criterion (BIC) (Singh, 2022). BIC is a model selection criterion where lower BIC means the model has lower penalty terms and thus the model is better (Datalab, 2019). The penalty term relates to number of parameters (Datalab, 2019). Fig 15 shows the optimal number components using the BIC plot and Fig 16 shows the process flow for GMMs method.

Chart, line chart

Description automatically generated

Fig 15: BIC Plot for Optimal Number of Components

Diagram

Description automatically generated

Fig 16: Process of Gaussian Mixture Model Method

1. **Results**

We used 3 methods of evaluation to compare the performances of all the methods. For our first method of model evaluation, the clustering methods were compared against the benchmark method using the porosity (x-axis) vs log scaled permeability (y-axis) scatter plot with the clusters shown as well as comparing cluster averages each method to see if any of the 6 clustering methods produce similar results. For our second method of model evaluation, using a sample dataset of 50 samples of the 2000 sample dataset used for training, we performed permeability predictions using our clustering algorithms and plotted residual models. The residual models were then compared against one another to observe how scattered each plot is. For the third method of model evaluation, we compare the unsupervised algorithms using the Silhouette Coefficient, Calinski-Harabaz Index, and the Davis-Bouldin Index

* 1. **Comparing Porosity vs Log Scaled Permeability Plots and Cluster Averages**

For all the methods, training was done using the normalized dataset while visualization was done on the unnormalized dataset with the cluster labels taken from trained model. Each cluster was assigned a color and the list was passed into the scatterplot to observe the clusters. Only 1000 sample points was used for plotting for clearer visualization. Porosity vs log scaled permeability was plotted for all the models and with porosity being the x-axis and y-axis being permeability that was displayed log scaled.

The average values of the columns for each cluster of each method were obtained from the unnormalized data. The average table for each method was then compared against one another to find any resemblance in the clusters. This method would not be accurate as values would be greatly differing due to the scale and differences in cluster groupings.

The results displayed below for the scatterplots show that the 5 cluster methods cluster grouping on the scatterplot do not resemble the modified method. As for the cluster average table comparisons, the cluster averages of each method do not resemble one another.

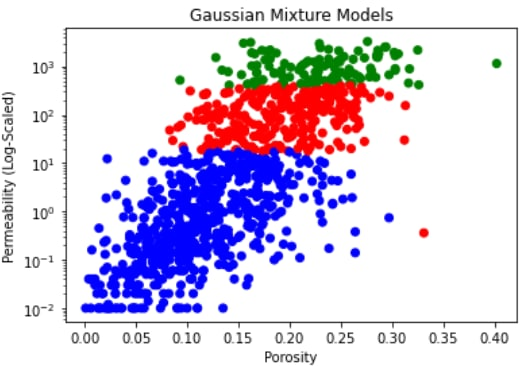
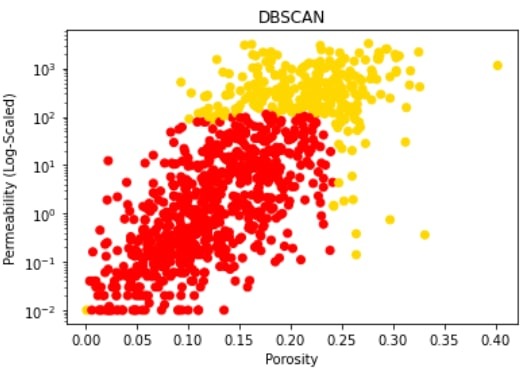
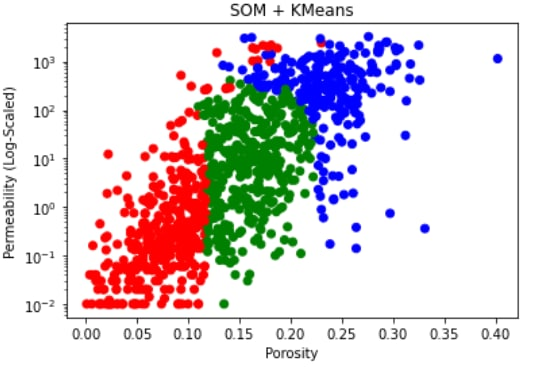
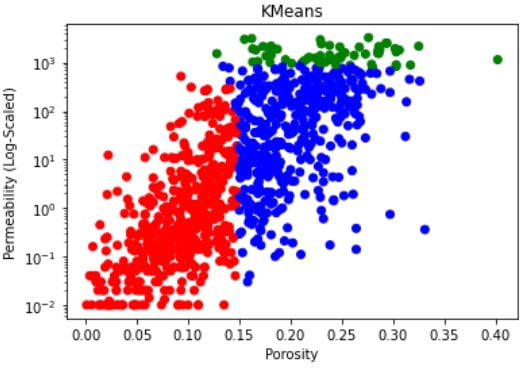
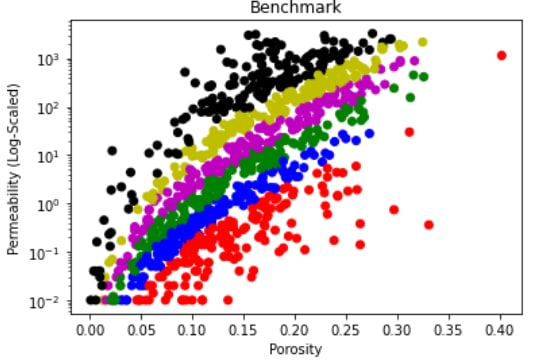
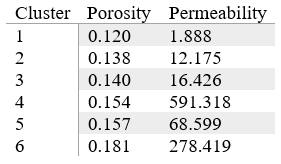
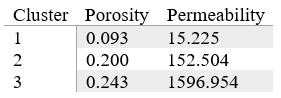


Fig 17: Porosity vs Log Scaled Permeability Scatterplot Cluster Distributions

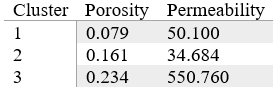
Benchmark



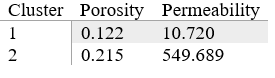
KMeans



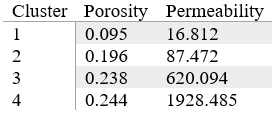
SOM + KMeans



DBSCAN



BIRCH



Gaussian Mixture Models

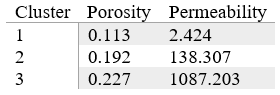


Table 1: Cluster Porosity and Permeability Averages

The cluster porosity and permeability averages of the 5 clustering algorithms comparing to the benchmark method do not look similar. The benchmark method has a total of 6 clusters whilst the 5 algorithms have 4 or less clusters which may indicate that it may be sufficient to just have a smaller number of rock types rather than the amount produced by the benchmark method. Based on Table 1 BIRCH and KMeans ae the most similar with the average porosity and permeability of cluster 1s in both algorithms to be very similar in terms of value. Likewise for KMeans’ cluster 3 to BRICH’s cluster 4.

* 1. **Comparing Residual Models**

For this approach of evaluation, we did permeability prediction, then calculate the residuals to get the residual models for each of the 6 methods. Permeability prediction is one of the applications that are related to rock typing and has been used in the field for a long time. For this evaluation, the purpose was to compare the residual models for all 6 methods that are obtained by permeability prediction which can show us the sample data deviates from each of the models. This can give us some insight into the performance of each model for rock typing.

For the benchmark method, using the sample sets’ porosity, the log porosity index values were calculated and used to predict the log RQI by using the linear regression prediction depending on the FZI of the sample which affects which linear regression model to be used for prediction as our benchmark method when training the model, splits the dataset into equal parts depending on the iteration number. As the FZI is already sorted in the data preparation stage, the linear regression fitted HFU is based on a specific range of non-overlapping FZI values. The sample data points FZI is compared against the ranges for the 6 FZI ranges that were determined when training the model and the corresponding linear regression would be used to predict log RQI of the sample data point. The predicted RQI can then be worked out from the predicted RQI and the predicted permeability can then be calculated by performing some calculations. The residuals are then obtained by having the actual sample permeabilities minus the predicted permeabilities. A scatterplot with predicted permeability on the x-axis and residual on the y-axis with a dashed horizontal line at point 0 is plotted. This process is repeated for all 6 methods. Some of the methods have a simple approach to calculating the predicted permeability by simply using the predict function like the KMeans algorithm using the sci-kit learn KMeans. However, most of the methods do not have such prebuilt functions. What was done is very similar to the process of getting the predicted permeability and residuals for the benchmark method. In short, each cluster is fitted with a linear regression that will be used for permeability prediction for the samples. The sample’s predicted cluster is first obtained for the sample set data points. The samples’ predicted permeability is then obtained by predicting using the different cluster’s linear regression prediction method. The residuals can then be derived, and the residual model plotted. The residual models for all 6 models are then plotted and will be shown below.

[still using lr rather than power functions, difficulty in plotting them]

[pending new charts]

* 1. **Comparing Unsupervised Machine Learning Algorithms Using Metrices**

This method of evaluation is comparing the unsupervised machine learning algorithms using metrics. The metrics chosen are the Silhouette Coefficient, the Calinski-Harabaz score, and the Davis-Bouldin Index. This evaluation excludes the benchmark as there was no straightforward way to for applying these metrics to the model. However, an implementation is possible to derive the metrics values for the benchmark method which was not touched on in this study. These metrics evaluate the uniqueness of clusters. For the Silhouette Coefficient and the Calinski-Harabaz index, the higher values are better. For the David-Bouldin Index, the lower values are better. The required inputs needed are the dataset and labels, of which the unsupervised algorithms used have implementations retrieve the labels. Below is an overview of all 3 metrics on the 5 unsupervised machine learning algorithms with values rounded to 3 decimal points.

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Silhouette Coefficient | Calinski-Harabaz Index | Davis-Bouldin Index |
| KMeans | 0.529 | 2556.740 | 0.642 |
| SOM+KMeans | 0.412 | 1505.616 | 0.859 |
| DBSCAN | 0.404 | 1180.614 | 1.041 |
| BIRCH | 0.474 | 2170.326 | 0.839 |
| GMMs | 0.263 | 1090.622 | 1.032 |

Table 2: Summary of the 5 Clustering Models’ Metrices

Based on Table 2, looking on an individual level for the silhouette coefficient, KMeans have the highest score, for the Calinski-Harabaz Index, KMeans and BIRCH have high scores and for the Davis-Bouldin Index, KMeans and BIRCH the lowest relative scores. From an overall viewpoint, both KMeans and BIRCH have good scorings based on the 3 metrics and have been shown to outperform the other algorithms.

1. **Limitations of Study and Conclusion**

In this study, we performed rock typing using our implementation of the traditional rock typing method as well as 5 other unsupervised machine learning algorithms, namely KMeans, SOM+KMeans, DBSCAN, BIRCH, and GMM. The scaled-down dataset used for training consists of 2000 samples of the UK core data with porosity and permeability data which is the petrophysical properties that are typically used for rock typing in the field. The own implementation of the traditional method is derived by closely studying several studies. It acts as a benchmark model for us to compare the other models. We then compared all 6 trained models with optimized parameters using 3 different approaches.

When comparing the cluster distributions in the porosity vs log-scaled permeability scatterplots and the cluster averages of the 6 models, we observe that none of the 5 models' cluster distributions and cluster averages resemble the benchmark method results. KMeans and BIRCH cluster distribution and cluster averages are close to one another.

The 2nd approach to the evaluation was to perform permeability prediction using a sample set of 50 samples and plot residual models for comparison. The model that has less spread from the residual = 0 point would mean that the model fits the data best. Based on the comparison of the residual models, KMeans and BIRCH models have a similar residual spread to the benchmark model with the other models showing poor performance when used for permeability prediction.

The 3rd approach compares the Silhouette Coefficient, Calinski-Harabaz Index, and the Davis-Bouldin Index metrics for the 5 unsupervised machine learning algorithms. Overall, KMeans and BIRCH are the best-performing algorithms based on 3 metrics.

Overall, when comparing the 5 unsupervised machine learning models to the benchmark model, KMeans and BIRCH show to be able to cluster the rock types the best and the performance is comparable to the performance of the benchmark model. Although the clustering algorithm models cluster distribution is not like the benchmark models’, it is still able to be used as a method of rock typing. Using unsupervised machine learning algorithms would bring about time savings and efficiency while not compromising the quality and accuracy of the clustering of rock types. However, this is not to say to solely replace the expertise of experts, data is only useful depending on who is using it. Experts will still need to take over after training the models to make sense of the results. This would bring us to progress in the oil and gas rock typing area.

The study only used the limited dataset of UK core data. There is room for improvement by including more algorithms, using more datasets to account for variability, and benchmarking on multiple traditional methods of rock typing.

**References**

Amaefule, J. O., Altunbay, M., Tiab, D., Kersey, D. G., & Keelan, D. K. (1993). *Enhanced Reservoir Hydraulic (Flow) Units and Predict Permeability in Uncored Intervals/Wells* [Paper Presentation]. Paper presented at the SPE Annual Technical Conference and Exhibition, Houston, Texas, October 1993. https://doi.org/10.2523/26436-ms

Datalab, A. (2019, January 16). *What is Bayesian Information Criterion (BIC)?* Medium. Retrieved from https://medium.com/@analyttica/what-is-bayesian-information-criterion-bic-b3396a894be6

Izadi, M. & Ghalambor, A. (2013). A new approach in permeability and hydraulic-flow-unit determination. *SPE Reserv. Eval. Eng.* **16**(03), 257–264.

Kolodzie, S. (1980). *Analysis of pore throat size and use of the Waxman–Smits equation to determine OOIP in Spindle Field, Colorado* [Paper Presentation]. Paper presented at the SPE Annual Technical Conference and Exhibition, Dallas, Texas, September 1980. https://doi.org/10.2118/9382-MS

Mirzaei-Paiaman, A., Ostadhassan, M., Rezaee, R., Saboorian-Jooybari, H. & Chen, Z. (2018). A new approach in petrophysical rock typing. *Journal of Petroleum Science and Engineering,* **166**, 445–464. https://doi.org/10.1016/j.petrol.2018.03.075

Nooruddin, H. A. & Hossain, M. E. (2011). Modified Kozeny–Carmen correlation for enhanced hydraulic flow unit characterization. *Journal of Petroleum Science and Engineering,* **80**(1), 107–115. https://doi.org/10.1016/j.petrol.2011.11.003

**Appendix 1 – Rock Typing Indices Equations**

|  |  |  |  |
| --- | --- | --- | --- |
| Authors | Indices | Equations | Equation number |
| (Amaefule et al, 1993) |  | where:  k -> permeability  -> porosity  -> normalized porosity  -> Rock Quality Index |  |
| (Mirzaei-Paiaman et al, 2018) | FZI\* | Where:  k -> permeability  -> porosity |  |
| (Nooruddin and Hossain, 2011) | FZIM | Where:  m -> cementation factor |  |
| (Izadi and Ghalambor, 2013) | MFZI | Where:  -> irreducible water saturation  -> Rock Quality Index |  |
| (Shahat et. al.,2021) | RFZI | Where:  -> resistivity index  -> true formation resistivity  -> formation resistivity when water saturation is 100%  -> formation water resistivity  -> surface grain volume  k -> permeability  -> shape factor |  |
| (Kolodzie, 1980) | Winland | Where:  -> uncorrected air permeability  -> porosity  r35 -> pore throat radius with 35% mercury saturation |  |