**WELL LOG FACIES ANALYSIS & CLASSIFICATION USING MACHINE LEARNING TECHNIQUES**

L Akhilesh

200107045

Submission Date: April 25, 2024



Final Project Submission

Course Name: Applications of AI and ML in chemical engineering

Course Code: CL653

# **Contents**

1. Executive Summary
2. Introduction
3. Methodology
4. Implementation Plan
5. Testing and Deployment
6. Results and Discussion
7. Conclusion and Future Work
8. References
9. Appendices
10. Auxiliaries

# **Executive Summary**

The project aims to solve the task of classifying different geological facies based on their different characteristics; which was experimentally determined. The properties of these rocks were found through the process of wireline logging; which is a well-logging method that uses a variety of downhole tools to measure the physical properties of rocks.

Classification of geological facies is a challenging task due to the complex and heterogeneous nature of the subspace formations. This requires expertise in geology and proper understanding of well log measurements.

To come up with a solution for this problem we can make use of machine learning. We can try to employ different classification algorithms of machine learning on known datasets and use the best performing model to determine the facies type in the future.

We can expect to have different models built using different classification algorithms. Further we can compare the results and decide which model best classifies the facies.

# **Introduction**

Facies classification is crucial in chemical engineering because it aids in the characterization of subsurface reservoirs, which is essential for optimizing hydrocarbon extraction processes. Understanding the distribution and properties of different geological facies helps in predicting reservoir behaviour, estimating reservoir capacity, and designing efficient reservoir management strategies, thereby maximizing the recovery of valuable resources, and thereby ensuring the economic viability if the oil and gas production operations.

For the problem considered, the dataset contains logs from the largest gas fields in North America, the Hugoton and Panoma Fields.

Facies (types of rocks) are studied from core samples in every half foot and matches with logging data in the well location. Feature variables include five from wireline log measurements and two geologic constraining variables that are derived from geologic knowledge.

The variables are:

* GR: this wireline logging tools measure gamma emission
* ILD\_log10: this is resistivity measurement
* PE: photoelectric effect log
* DeltaPHI: Phi is a porosity index in petrophysics.
* PNHIND: Average of neutron and density log.
* NM\_M: nonmarine-marine indicator
* RELPOS: elative position

The discrete facies (classes of rocks) are:

* SS: Nonmarine sandstone
* CSiS: Nonmarine coarse siltstone
* FSiS: Nonmarine fine siltstone
* SiSH: Marine siltstone and shale
* MS: Mudstone (limestone)
* WS: Wackestone (limestone)
* D: Dolomite
* PS: Packstone-grainstone (limestone)
* BS: Phylloid-algal bafflestone (limestone)

The main objectives of this project are:

* To develop a machine learning model to accurately classify geological facies in the Hugoton and Panoma Fields using wireline log measurements and geological constraining variables.
* Enhance the reservoir characterization and support decision-making in reservoir development by providing precise facies classification.

# **Methodology:**

## Data Source:

The data originally comes from the measurements made in the gas fields in North America, the Hugoton and Panoma Fields.

The data was originally mentioned in the article called “Computers & Geosciences”, Volume 33, Issue 5, May 2007. It is mentioned under the topic; “Comparison of four approaches to a rock facies classification problem.” This article is written by Martin K. Dubois, Geoffrey C. Bohling and Swapan Chakrabarti.

<https://www.sciencedirect.com/science/article/pii/S0098300406001956?via%3Dihub>

Further the data was also used in class exercises in The University of Kansas. On top of this, it can be found on Kaggle and also in the other link mentioned below.

<https://www.kaggle.com/code/imeintanis/log-facies-nn-bayesian-optimization-skopt/input>

<https://github.com/akhiroxxx/AI-ML-Project-CL>

## Data Preprocessing:

Preprocessing steps done in the project are:

* Making selected features like Well Name and Formation Type as Categorical Data to improve performance; ensuring they are not treated like data values but instead categorically.
* Removing rows with invalid data; which could be due to wrong/null values present. This was observed for PE (Photoelectric log).
* Removing data of one selective well alone and later using it as blind data to check for performance of models.
* Standardizing the data to zero mean and unit variance to better apply the ML models for their better performance.
* Splitting the training data intro test set and training set.

## **Model Architecture:**



**Hugoton & Panoma Gas Fields**

Data Set

(Log Analysis)

Data Preprocessing

Keep Blind Data separate For Final Evaluation Of Models And Split the rest

Train Data (75%)

Evaluation Metrics

(F1–Score)

Test Data (25%)

Model Training

Models Obtained

Model Evaluation

Evaluation Metrics

(F1–Score)

Test On Blind Dataset

Model Selection

## **Tools and Technologies:**

The programming language used was Python. This is because it has a very good documentation and is one of the most profound languages when it comes to implementing ML algorithms; due to its ease of handling relational data.

Also, it has inbuilt libraries which can implement the ML model on the data without the need to actually having to code it up.

Other tools which were used are:

* NumPy: To handle mathematical functions on arrays and matrices efficiently.
* Pandas: To handle tabular data and perform operations on them
* Matplotlib: To graphically visualize and analyse data.
* Seaborn: To make graphical data more attractive and informative.
* Sklearn-preprocessing: To perform operations like scaling, transforming and normalization for optimal performance and stability of the algorithms.

# IMPLEMENTATION PLAN:

# **TESTING & DEPLOYMENT:**

## Testing Strategy:

For ensuring that the final model used is relevant and gives close to accurate result, we can make use of blind dataset.

This blind dataset can be derived from the training dataset itself, by removing it initially before doing any training on it.

Later evaluation can be done on the blind dataset using evaluation metrics like F1 Score, precision or recall to decide if the model is accurate or not

## Deployment Strategy:

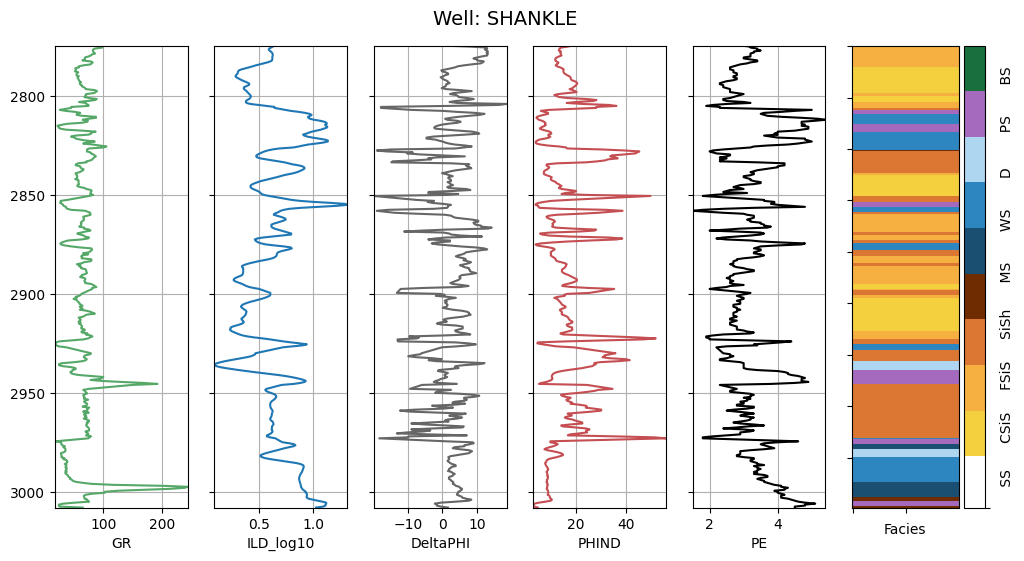
* Model Serialization: Serialize the trained model using libraries like ‘joblib’ or ‘pickle’ to save the model’s state and parameters.
* API Development: Develop a RESTful API using frameworks like Flask or FastAPI to expose the model for real-time predictions and integration with other systems.
* Containerization: Containerize the application using Docker to ensure consistent and isolated environments, facilitating scalability and deployment across different platforms.
* Scalability: Deploy the containerized application on cloud platforms like AWS, Azure or GCP to leverage auto-scaling capabilities and handle increased loads efficiently.
* Monitoring and Logging: Implement monitoring and logging mechanisms to track the model’s performance, detect anomalies, and maintain system health for continuous improvement and troubleshooting.
* Version Control & Updates: Implement version control to manage different versions of the model and facilitate seamless updates and rollback strategies for maintenance and enhancements.

## Ethical Considerations :

* Bias and Fairness: Ensure the model does not perpetuate or amplify existing biases present in the training data, which could result in unfair or discriminatory outcomes.
* Transparency and Interpretability: Provide transparency in the model’s decision-making process and ensure it is interpretable, enabling stakeholders to understand and trust the model’s predictions and recommendations.
* Data Privacy and Security: Implement robust data privacy and security measures to protect sensitive and confidential information, complying with relevant data protection regulations like GDPR or CCPA.
* Accountability and Responsibility: Establish clear accountability and responsibility for the model’s outcomes and decisions to mitigate potential risks and ensure proper handling of errors or unintended consequences.

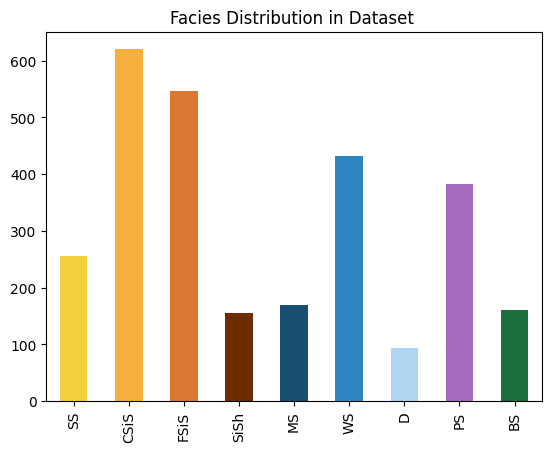
# **RESULTS AND DISCUSSION:**

## Findings:

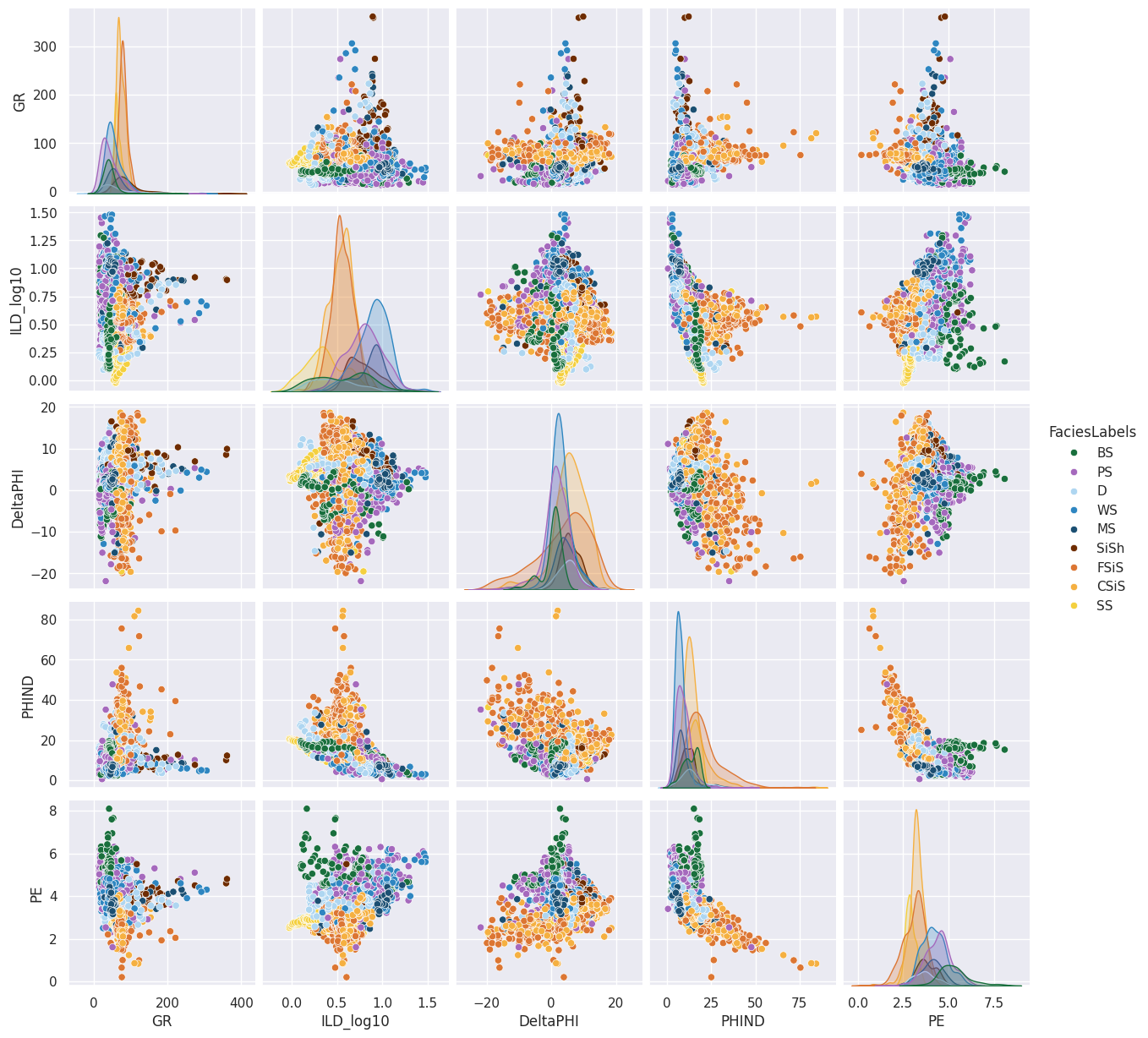
Some of the graphs through which significant insights were obtained have been attached below.

In well logging, plotting the log values with respect to depth is a common practice because it provides a vertical profile of the subsurface formations and their properties. This allows geoscientists and engineers to interpret and analyse the geological and petrophysical characteristics of the formations at different depths. The above figure shows the variation of every feature variable with depth for the well name: Shankle. Such plots have been made for other wells too.

To understand the distribution of the different types of facies in the dataset is a part of explatory data analysis.

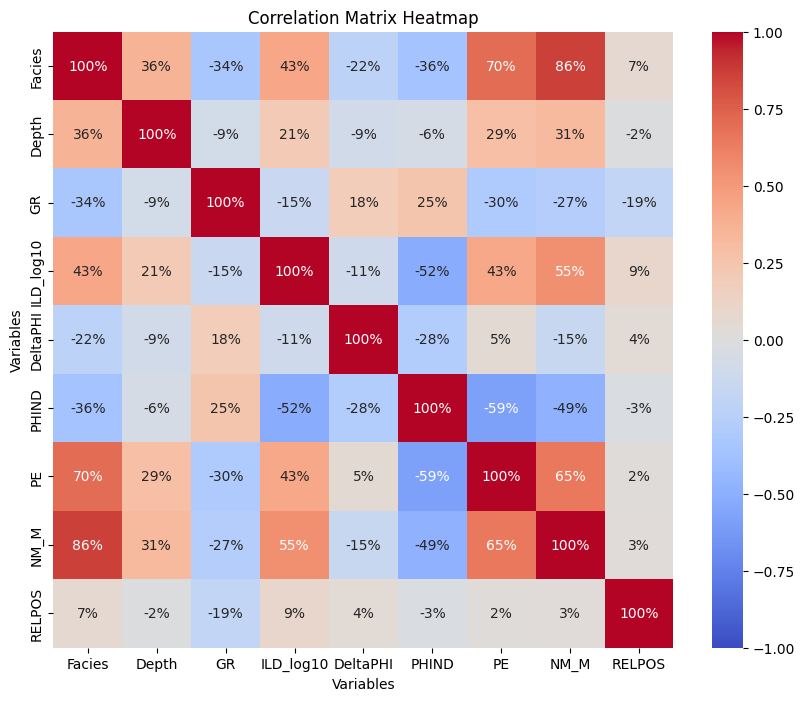


We also visualise how different facies types can vary with the different feature variables through cross plot(scattered). This is a general practise in geosciences.

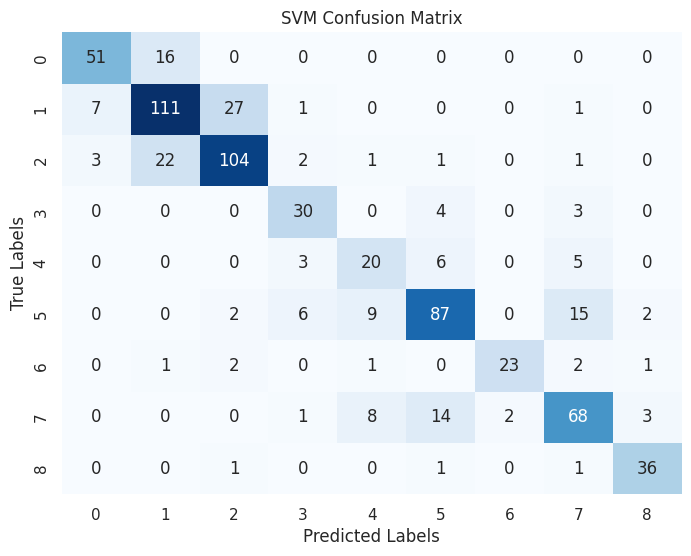


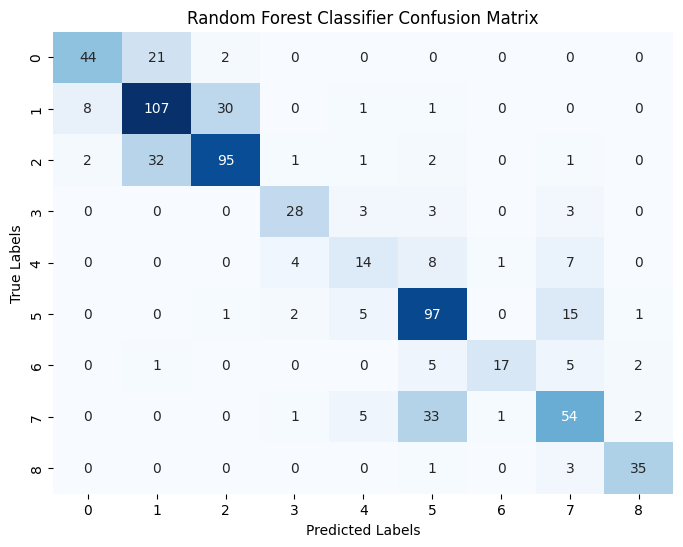
The correlation matrix is used to determine how each feature linearly varies with the other; through which we can also determine features which are closely linked to one another. The figure below represents the correlation matrix for the dataset used.

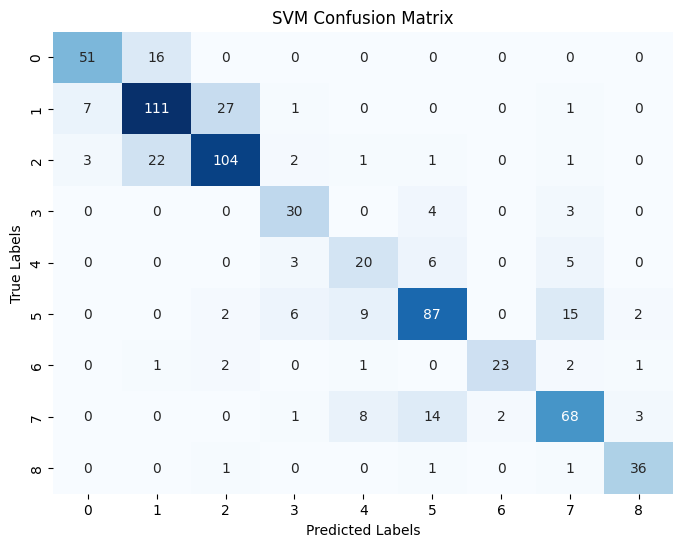
After employing the different ML models like:

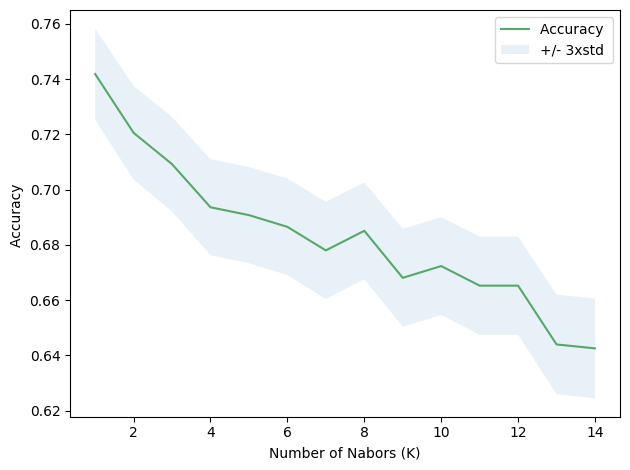
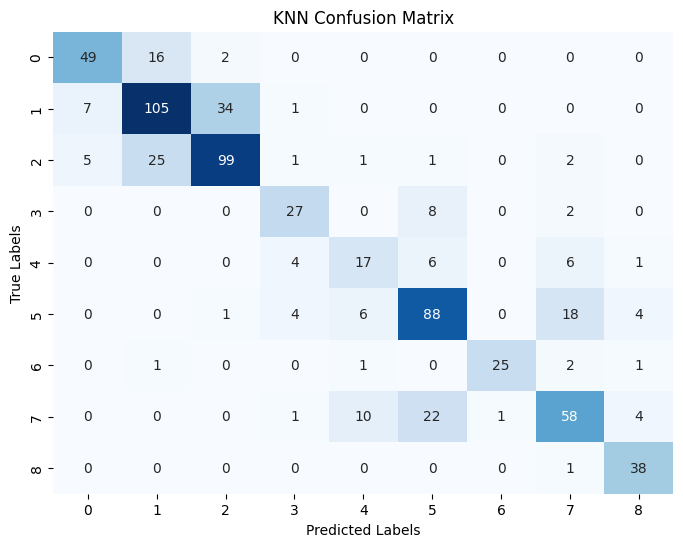
* SVM
* KNN
* Logistic Regression
* Random Forest Classifier
* Gaussian Naïve Bayes

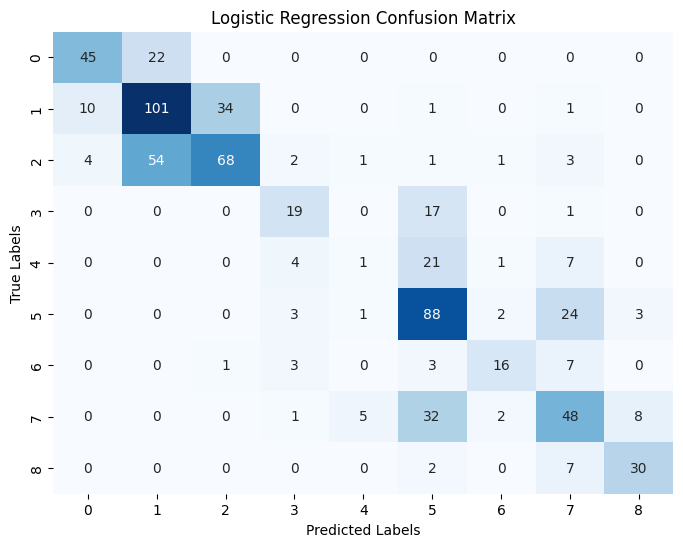
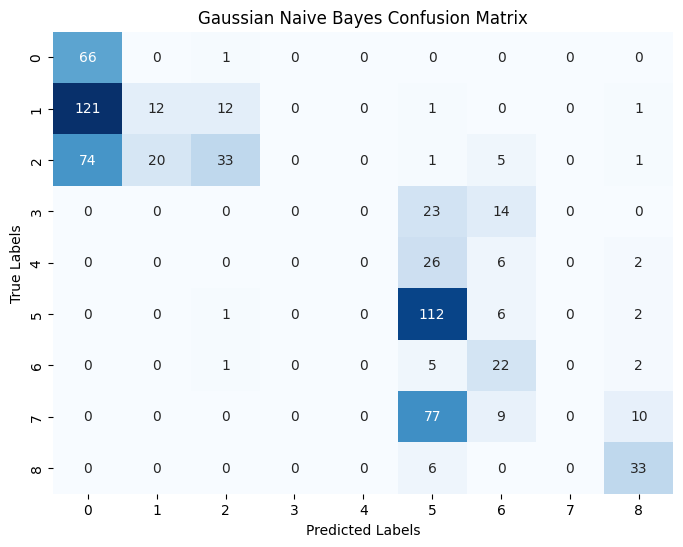
on the dataset; F1 scores and confusion matrices were obtained.











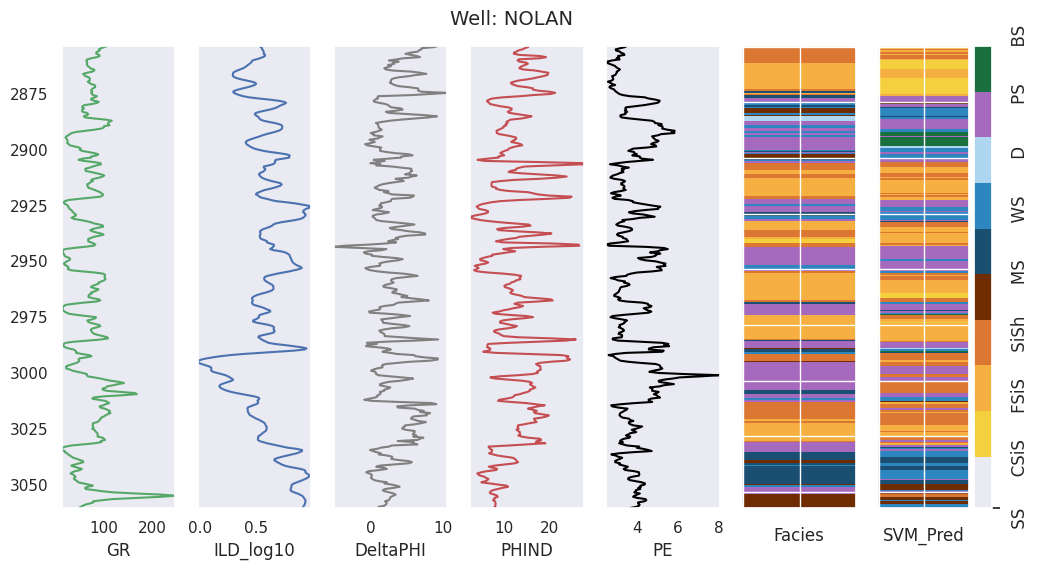
|  |  |  |
| --- | --- | --- |
| S. No. | Model Type | F1-Score |
| 1 | SVM | 0.75 |
| 2 | RFC | 0.70 |
| 3 | KNN | 0.72 |
| 4 | GNB | 0.30 |
| 5 | LR | 0.58 |

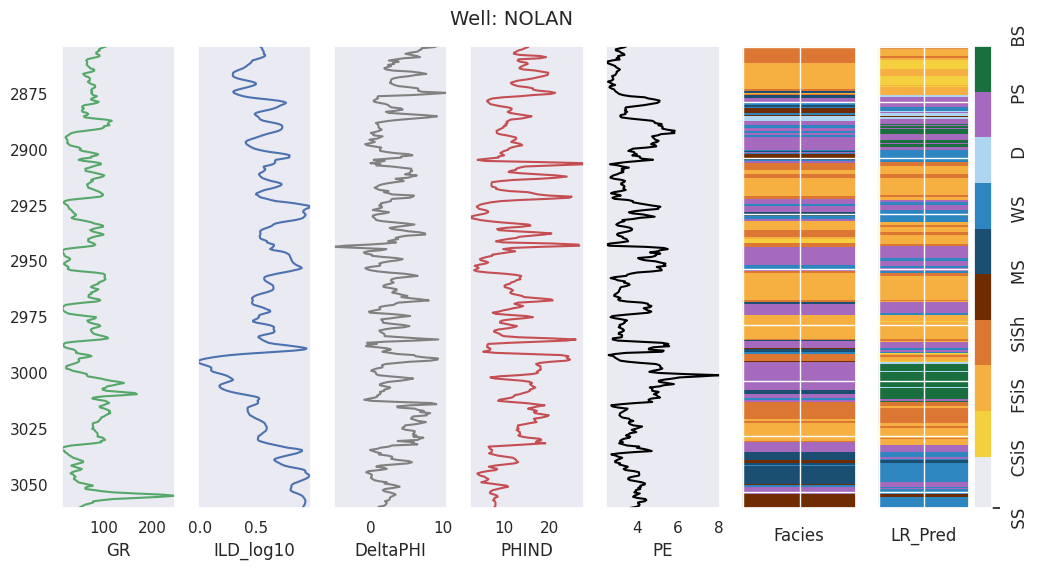
Applying the same model on the blind dataset;

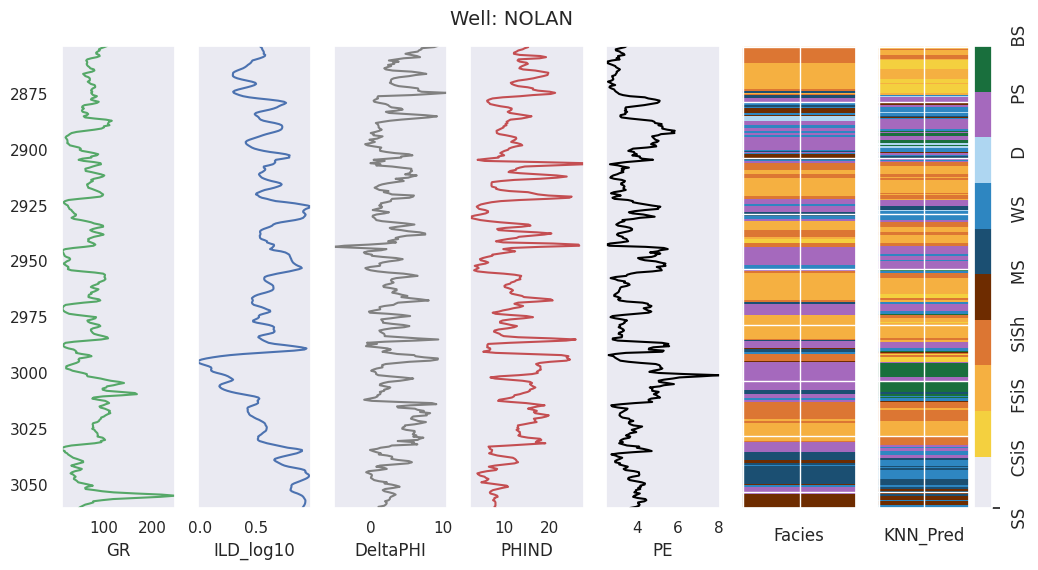
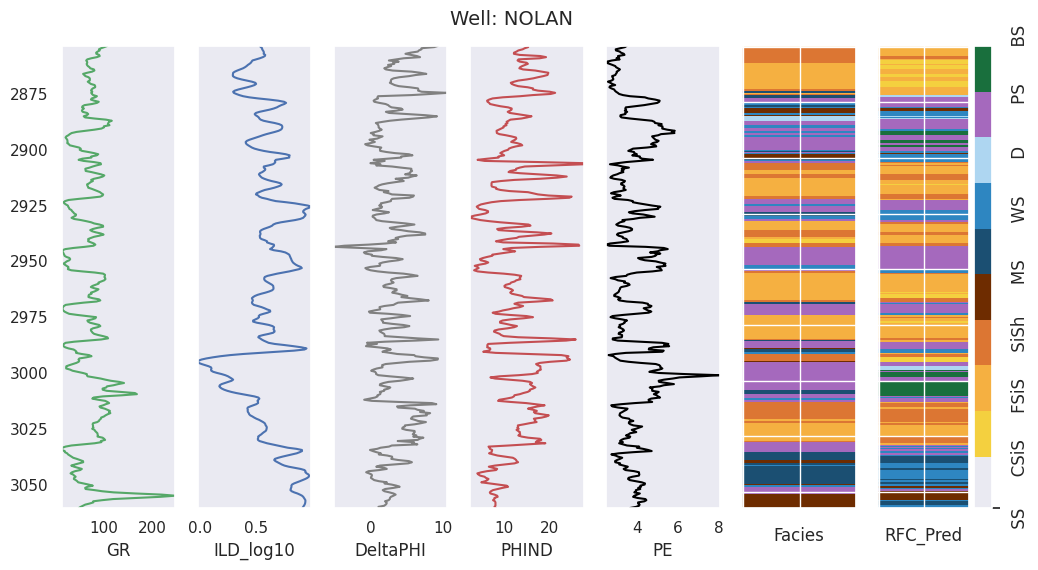
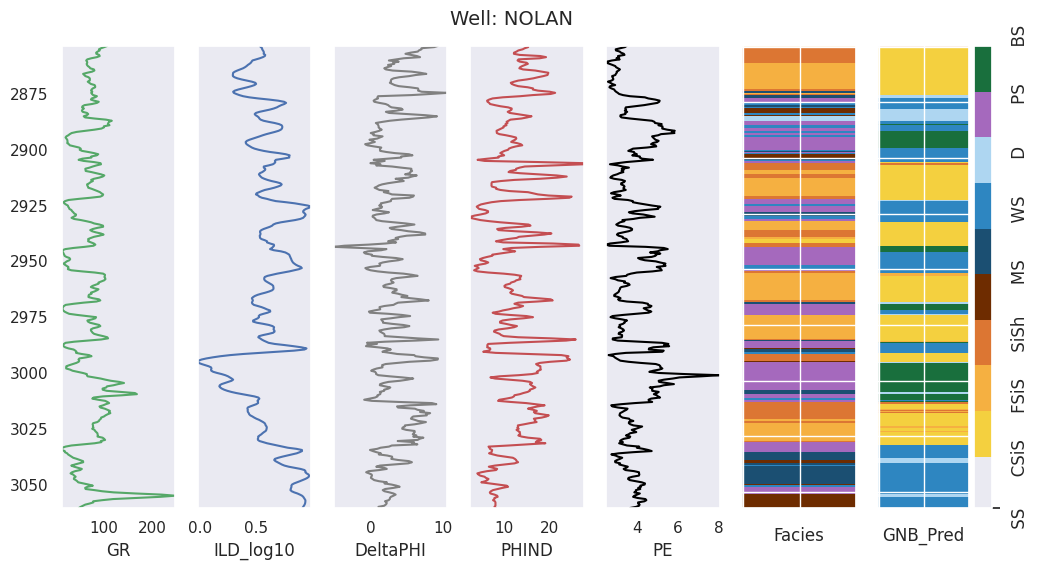
|  |  |  |
| --- | --- | --- |
| S. No. | Model Type | F1-Score |
| 1 | SVM | 0.49 |
| 2 | RFC | 0.54 |
| 3 | KNN | 0.50 |
| 4 | GNB | 0.07 |
| 5 | LR | 0.49 |

From the above it can be noted that **SVM**, **RFC** and **KNN** give very consistent results; while **GNB** very poor results.

After plotting the model’s prediction performance on the blind dataset; we obtain the plots shown below.







The inference can be seen through the plots also as the colours of the predictions by SVM, KNN and RFC resemble the colour of original facies result.

There wasn’t any specific challenge that I faced while making this project; however the solution is not a strong one.

F1 scores in the range of 0.5 to 0.8 is good and such were the scores obtained by some models.

However when the model was tried on the blind data set; F1 score was close to the 0.5 range which is moderately appreciable.

# CONCLUSION AND FUTURE WORK:

## Conclusion:

The project focused on the classification of geological facies using machine learning techniques, specifically exploring the use of well log data to predict and differentiate between various subsurface rock types. Several machine learning classifiers were trained and evaluated, achieving an average F1 score of approximately 0.71 on the test data and successful results on the blind data too.

The project’s impact lies in its potential to optimize hydrocarbon extraction processes and enhance reservoir management strategies.

## Future Work:

Further improvements can be made in the Machine learning algorithm applied through:

* Feature Engineering and Selection
* Hyperparameter Tuning and Model Optimization
* Integrating ML using Deep Learning Approaches (like CNN,RNN)

In addition to this deploying the project along with real-time monitoring and adaptive learning to continuously update and refine the models based on new data can make the project relevant and dynamic always.

# References:

1. Dubois, D., Grader, A., Barry, P.H., et al. (2007). Hugoton and Panoma Fields, Kansas. Kansas Geological Survey, Open-file Report 2007-29.
2. University of Kansas. (n.d.). Class Exercise on Hugoton and Panoma Fields.
3. Scikit-learn Documentation. (n.d.). Machine Learning in Python.
4. TensorFlow Documentation. (n.d.). An end-to-end open-source machine learning platform.
5. Keras Documentation. (n.d.). The Python Deep Learning Library.
6. PyTorch Documentation. (n.d.). An open source machine learning framework.
7. Pandas Documentation. (n.d.). Data Structures and Data Analysis Tools.
8. NumPy Documentation. (n.d.). The fundamental package for scientific computing with Python.
9. Matplotlib Documentation. (n.d.). Visualization with Python.

10.Seaborn Documentation. (n.d.). Statistical Data Visualization.

# APPENDICES:

All the required graphs, plots related to data exploration, analysis and the inferences obtained have been included under Findings in the Results and Discussion Section.

# AUXILIARIES:

**Python File:**

https://github.com/akhiroxxx/AI-ML-Project-CL/blob/main/Well\_Log\_Facies\_Analysis\_And\_Classification.ipynb

**Data Source:**

https://github.com/akhiroxxx/AI-ML-Project-CL/blob/main/facies\_data.csv