PREDICTION AND TESTING OF WELL LOG PARAMETERS BASED ON MACHINE LEARNING

Report submitted to the

OIL & NATURAL GAS CORPORATION LTD.

In fulfilment of the requirement for the award of the completion of training by

Mr. Isham Sinha

Of

PES University,

Bangalore, Karnataka

Under the guidance of

Mr. Alok K Dudwe,

Sr.Prog.Officer, ONGC

During Summer Training at

DATABASE GROUP,

OIL AND NATURAL GAS CORPORATION LIMITED,

NBP GREEN HEIGHTS, BANDRA-KURLA COMPLEX,

MUMBAI

CERTIFICATE

This is to certify that Mr. Isham Sinha, a student of 2nd Year B.Tech from PES University, Bangalore, Karnataka has carried out a project titled “PREDICTION AND TESTING OF WELL LOG PARAMETERS BASED ON MACHINE LEARNING ” at ONGC NBP Green Heights, Bandra-Kurla Complex, Mumbai. He has successfully completed the projects and submitted the project report, which is the original work carried by him.

(   )

Mr. Alok K. Dudwe

Sr. Prog. Officer ONGC

ACKNOWLEDGEMENT

I am grateful to my mentor **Mr. Alok K. Dudwe, Sr. Programming Officer** for his valuable help & guidance in the completion of summer training at ONGC NBP Green Heights, Bandra-Kurla Complex, Mumbai.

I would also like to thank Mrs. Sunita Kashyap **DGM (Prog)** for providing seamless support and right suggestions in the development of the project. I thank them for sharing their immense programming skills and knowledge along with their technical support and advice during my entire project tenure.

Isham Sinha

ABSTRACT

This project focuses on developing a machine learning-based approach for predicting well log parameters in the petroleum industry. It proposes a comprehensive approach for well data cleaning and prediction of missing values. The approach utilises advanced data cleaning techniques, statistical analysis, and machine learning algorithms to rectify missing values. Well log datasets are preprocessed to extract relevant features capturing spatial and temporal variations. Various machine learning algorithms are employed, trained on the datasets, and evaluated using metrics such as root mean squared error. The interpretability of the models is explored, providing insights into the influencing factors. Real-world well log datasets are used for validation, demonstrating the effectiveness of the approach in improving reservoir characterization and decision-making.

In this project, I was asked to experiment with a real world dataset, and to explore how machine learning algorithms can be used to find the patterns in data. I was expected to gain experience using common data-mining and machine learning library; pandas, scikit-learn, numpy, matplotlib.pyplot, etc.

INDEX

OVERVIEW

ONGC:

Oil and Natural Gas Corporation Ltd. (ONGC) is recognized as the Numero Uno E&P company in the world and 20th among leading global Energy majors as per 'Platts Top 250' Global Energy Company Ranking. It has been ranked 449th in the Fortune Global 500 list of world's biggest corporations for the year 2018. ONGC has scripted India's hydrocarbon saga by discovering 6 of the presently 7 producing Basins of India, in the last 50 years, discovering over 7.1 billion tons of In-place Oil & Gas.

ONGC has recoverable reserves of over 1 billion tons of Oil and Gas, and produces more than 1.1 million Barrels of oil and oil Equivalent Gas (BOE) per day, meeting around 80% of India's domestic production of Oil and Gas. It owns and operates more than 22,000KM of pipelines in India, including nearly 4,500KM of subsea pipelines – the longest in India. All the installations of ONGC- India's Greenest company- are certified for quality, Health, Safety and environment Management (QHSE), making ONGC unique in the world in this regard.

EPINET:

Since its inception, ONGC has been carrying out extensive Exploration & Production activities in sedimentary basins of India. In this process, it acquired a huge volume of G & G, Drilling and Production data. With passage of time and advancements in technology, the generation of data has become manifold. However, to manage such a vast volume of data for effective use in E & P activities, it is required to be integrated and managed properly. This led to conceptualization of EPINET i.e. Exploration and Production Information Network, in the year 1999.

EPINET Phase-I

ONGC had conceived the EPINET project in Phased manner. The main data types managed in the FINDER environment of EPINET Phase-I consisted of Well Completion Reports, Seismic Navigation, Original Log Traces, Geo Laboratory, Production & Field Reservoir studies.

EPINET Phase-II

EPINET Phase-II data flow mechanism from Assets to Basins and Basins to Corporate server has been established and is being monitored on a regular basis.

IIWS:

The Interactive Interpretation Workstation Centre (IIWS) of Western Offshore Basin, commonly called the IIWS, is a major Centre where G&G interpretation of seismic and related exploration data, are carried out for hydrocarbon prospecting. Located on 5th Floor of NBP Green Heights, BKC, Bandra-East, Mumbai, the Centre is primarily used for carrying out 2D & 3D Seismic Interpretation with the help of State of the art software and high end graphics Workstations. Besides these, two Data Viewing Rooms have been set up with workstations installed for carrying out G&G data viewing by outside expert consultants.

The servers, peripheral units and network switches are all housed in a closed room, where the environment factors such as the temperature and humidity are maintained by precision air conditioners. This server room was expanded and all the equipment and ACs have been reoriented. The false floors have been replaced with fire-resistant /fire-proof tiles. The IIWS Centre has been christened a name called 'MANASKRUTI'. This centre was inaugurated by Shri Dinesh Kumar Pande, Director (Exploration), on 23rd Feb., 2010.

Introduction

The efficient analysis of well data plays a critical role in the oil and gas industry, enabling accurate reservoir characterization and decision-making. However, the quality of well data can be compromised due to various factors, including measurement errors, equipment malfunctions, and human errors. Consequently, the presence of missing values and erroneous data poses significant challenges for reliable data analysis and interpretation. This project focuses on addressing these challenges by proposing a comprehensive approach for well data cleaning and prediction of well log parameters.

The proposed approach leverages advanced data cleaning techniques, statistical analysis, and machine learning algorithms to identify and rectify missing values in well data. Initially, the data is preprocessed to handle outliers and inconsistencies. Next, a series of data imputation techniques, such as mean imputation, regression imputation, and k-nearest neighbours imputation, are applied to estimate the missing values based on the available data patterns and correlations..

The performance of the proposed approach is evaluated using real-world well datasets obtained from multiple oil and gas fields. A comparative analysis is conducted to assess the effectiveness of different data cleaning techniques and machine learning algorithms in terms of accuracy, computational efficiency, and scalability. The results demonstrate that the proposed approach significantly improves the quality of well data by accurately predicting values and enhancing the reliability of subsequent data analysis tasks.

This project contributes to the field of well data analysis by providing a robust and systematic framework for data cleaning and missing value prediction. The developed techniques and methodologies can be readily applied by oil and gas companies to enhance the accuracy and reliability of their well data analysis, leading to improved reservoir characterization, optimised production strategies, and more informed decision-making in the oil and gas industry.

1.1 Literature review:

The Oil and Natural Gas Corporation (ONGC) Ltd. is one of India’s largest and most successful companies. ONGC decided to implement the Exploration and Production Information Network (EPINET) project consisting of people, processes, tools, data and a hierarchy of corporate, regional and working project databases in a phased manner.

The different data classes handled under this category are Well, Log, Seismic, Laboratory, Reservoir, Drilling, Production and Well stimulation. The OPU (Own, Populate & Use) model is used for the implementation of the project. The data generators are also the data owners and have a role in populating the same in the database with subsequent use of the same in the various E & P activities.

Objective of EPINET is to provide accurate and unambiguous data to all user groups with maximum metadata details in minimum time and thereby minimising the search time by creating a comprehensive database with the concept of maximum data on line.

Escalating demand for hydrocarbons has intensified exploration and development of oil and gas fields in different parts of the world by various Oil Companies. ONGC is among such companies to enhance exploration and production activities using latest Data Management technologies. ONGC Corporate management has taken up Information Management and the E&P data in electronic form through EPINET since 1999.

1.2 Scope of the Project:

The scope of this project encompasses the development and evaluation of a comprehensive approach for well data cleaning and prediction of well log parameters. It includes the investigation and implementation of various data cleaning techniques, such as outlier detection, error correction, and missing value imputation, specifically tailored for well datasets. The project also involves exploring different machine learning algorithms for accurate missing value prediction based on the cleaned data.

The scope further encompasses the evaluation and validation of the proposed approach using real-world well datasets obtained from multiple oil and gas fields. The evaluation includes assessing the effectiveness of different data cleaning techniques and machine learning algorithms in terms of accuracy, computational efficiency, and scalability.

The scope of the project is limited to the domain of well data analysis in the oil and gas industry. However, the methodologies and techniques developed in this project may have broader applications in other fields where missing value prediction and data cleaning are essential, such as environmental monitoring, finance, and healthcare.

Overall, the project aims to provide a robust framework for well data cleaning and prediction of missing values, contributing to the advancement of well data analysis techniques and benefiting the oil and gas industry in terms of improved reservoir characterization and informed decision-making.

1.3 Problem Definition

The problem addressed in this project is the need for accurate prediction of well log parameters in the petroleum industry. Well log parameters, such as porosity, permeability, and lithology, are crucial for reservoir characterization and hydrocarbon resource estimation. Traditional prediction methods rely on complex mathematical models and expert knowledge, which can be time-consuming, subject to uncertainties, and may not capture the full complexity of the data.

The primary problem is to develop a machine learning-based approach that can accurately predict well log parameters based on available well log datasets. This involves addressing challenges such as data preprocessing, feature engineering, model selection, and evaluation. It is necessary to extract relevant features from raw well log data and train machine learning models to capture the complex relationships between input features and target parameters.

Additionally, ensuring the generalisation capability and interpretability of the predictive models is crucial. The models should be capable of providing accurate predictions on unseen well log data while also offering insights into the factors influencing the predictions. Achieving these goals will improve the efficiency, accuracy, and robustness of well log parameter predictions.

The project aims to develop a methodology that enhances the prediction of well log parameters using machine learning techniques. By addressing the challenges of traditional methods and leveraging the power of data-driven models, the project seeks to improve reservoir characterization, formation evaluation, and decision-making processes in the petroleum industry.

System Requirements

Software used:

Visual Studio Code, Jupyter Notebook

Operating System:

Windows OS, Mac OS, Ubuntu

Languages used:

Python3

Libraries used:

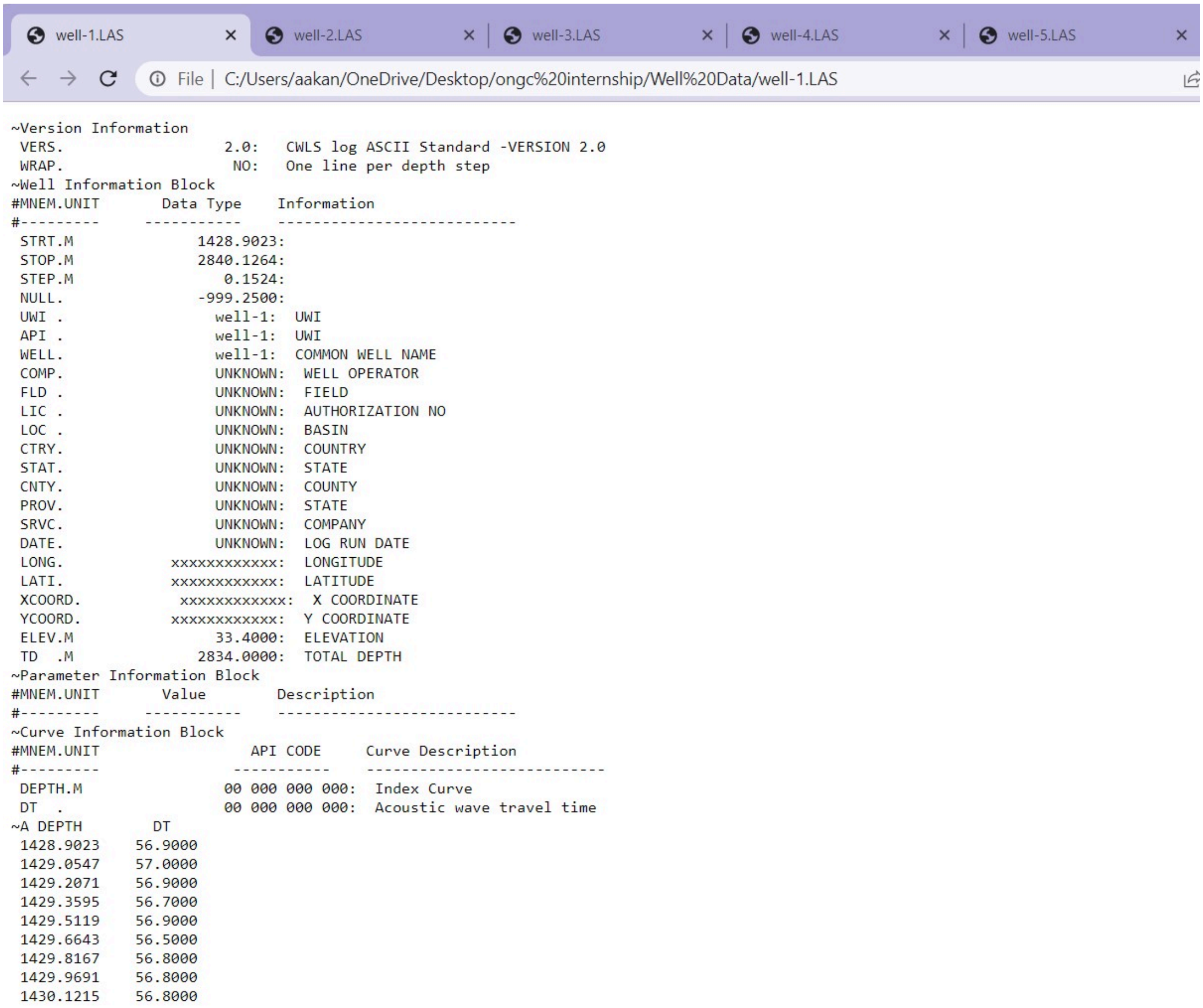
Pandas, scikit-learn, numpy, matplotlib.pyplot

Implementation

1. Data Acquisition:

Obtaining well datasets from relevant sources, ensuring the inclusion of various data types, such as production rates, pressure measurements, temperature readings, and fluid compositions. Ensure the datasets are representative of different oil and gas fields.

For this I was provided with 5 las files containing all the well data, namely well-1.LAS to well-5.LAS


well-1.LAS
well-2.LAS
well-3.LAS
well-4.LAS
well-5.LAS
File | C:/Users/aakan/OneDrive/Desktop/ongc%20internship/Well%20Data/well-1.LAS

```
~Version Information
 VERS.                  2.0:   CWLS log ASCII Standard -VERSION 2.0
 WRAP.                   NO:   One line per depth step
~Well Information Block
#MNEM.UNIT       Data Type    Information
#---------     -----------    ---------------------------
 STRT.M              1428.9023:
 STOP.M              2840.1264:
 STEP.M                 0.1524:
 NULL.               -999.2500:
 UWI .                 well-1:  UWI
 API .                 well-1:  UWI
 WELL.                 well-1:  COMMON WELL NAME
 COMP.                 UNKNOWN:  WELL OPERATOR
 FLD .                 UNKNOWN:  FIELD
 LIC .                 UNKNOWN:  AUTHORIZATION NO
 LOC .                 UNKNOWN:  BASIN
 CTRY.                 UNKNOWN:  COUNTRY
 STAT.                 UNKNOWN:  STATE
 CNTY.                 UNKNOWN:  COUNTY
 PROV.                 UNKNOWN:  STATE
 SRVC.                 UNKNOWN:  COMPANY
 DATE.                 UNKNOWN:  LOG RUN DATE
 LONG.            xxxxxxxxxxxx:  LONGITUDE
 LATI.            xxxxxxxxxxxx:  LATITUDE
 XCOORD.           xxxxxxxxxxxx:  X COORDINATE
 YCOORD.          xxxxxxxxxxxx:  Y COORDINATE
 ELEV.M                33.4000:  ELEVATION
 TD  .M              2834.0000:  TOTAL DEPTH
~Parameter Information Block
#MNEM.UNIT       Value        Description
#---------     -----------    ---------------------------
~Curve Information Block
#MNEM.UNIT                 API CODE     Curve Description
#---------               -----------    ---------------------------
 DEPTH.M                 00 000 000 000:  Index Curve
 DT  .                   00 000 000 000:  Acoustic wave travel time
~A DEPTH        DT
 1428.9023    56.9000
 1429.0547    57.0000
 1429.2071    56.9000
 1429.3595    56.7000
 1429.5119    56.9000
 1429.6643    56.5000
 1429.8167    56.8000
 1429.9691    56.8000
 1430.1215    56.8000
```

2. Data Preprocessing:

Perform initial data preprocessing steps, including data cleaning and formatting. Handle missing values and outliers using appropriate techniques, such as deletion, imputation, or interpolation. Address any inconsistencies or errors in the dataset.

3. Exploratory Data Analysis:

Conduct exploratory data analysis to gain insights into the characteristics of the well dataset. Analyse distributions, correlations, and patterns within the data to inform subsequent cleaning and prediction steps.

4. Missing Value Detection:

Identify missing values within the dataset. Utilise statistical techniques, visualisation, or pattern recognition algorithms to locate missing values and determine the extent of missingness.

5. Data Cleaning:

Apply data cleaning techniques specific to well datasets. These may include outlier detection and correction methods, error handling, and data transformation techniques. Ensure the dataset is free from inconsistencies and errors that could affect subsequent analysis.

6. Feature Engineering:

Feature engineering plays a crucial role in developing accurate machine learning models. The collected raw well log data may contain redundant or irrelevant information. Engineers and data scientists can apply domain knowledge to extract meaningful features or create new ones that help improve the predictive models.

7. Machine Learning Model Selection and Training:

Choosing the appropriate machine learning model is crucial for accurate prediction and testing of oil well log parameters. There are various algorithms to consider, including linear regression, support vector machines (SVM), decision trees, random forests, and deep learning models like convolutional neural networks (CNN) and recurrent neural networks (RNN).

The choice of model depends on factors like the nature of data, the complexity of the problem, and the desired accuracy. Ensembles of models, such as random forests or gradient boosting, are often preferred for their ability to capture complex relationships and reduce bias.

8. Value Prediction:

Utilise the trained machine learning models to predict well log parameters in the dataset. Apply the models to the incomplete portions of the dataset to complete the missing values based on the learned patterns and relationships.

9. Training and Validation:

After selecting a suitable model, it needs to be trained using the prepared well log dataset. The dataset is split into training and validation sets. The training set is used to teach the model to identify patterns and make predictions, while the validation set helps evaluate the model's performance and prevent overfitting.

During the training process, the model learns the underlying relationships between the input features and the desired output (i.e., the predicted well log parameters). This process involves adjusting the model's internal parameters through optimization techniques like gradient descent, to minimise the difference between the predicted values and the actual measurements.

10. Iterative Refinement:

Iterate and refine the approach as necessary based on the evaluation results. Fine-tune the cleaning techniques, explore different machine learning algorithms, or consider additional data preprocessing steps to improve the accuracy and reliability of the well dataset.

11. Testing and Evaluation:

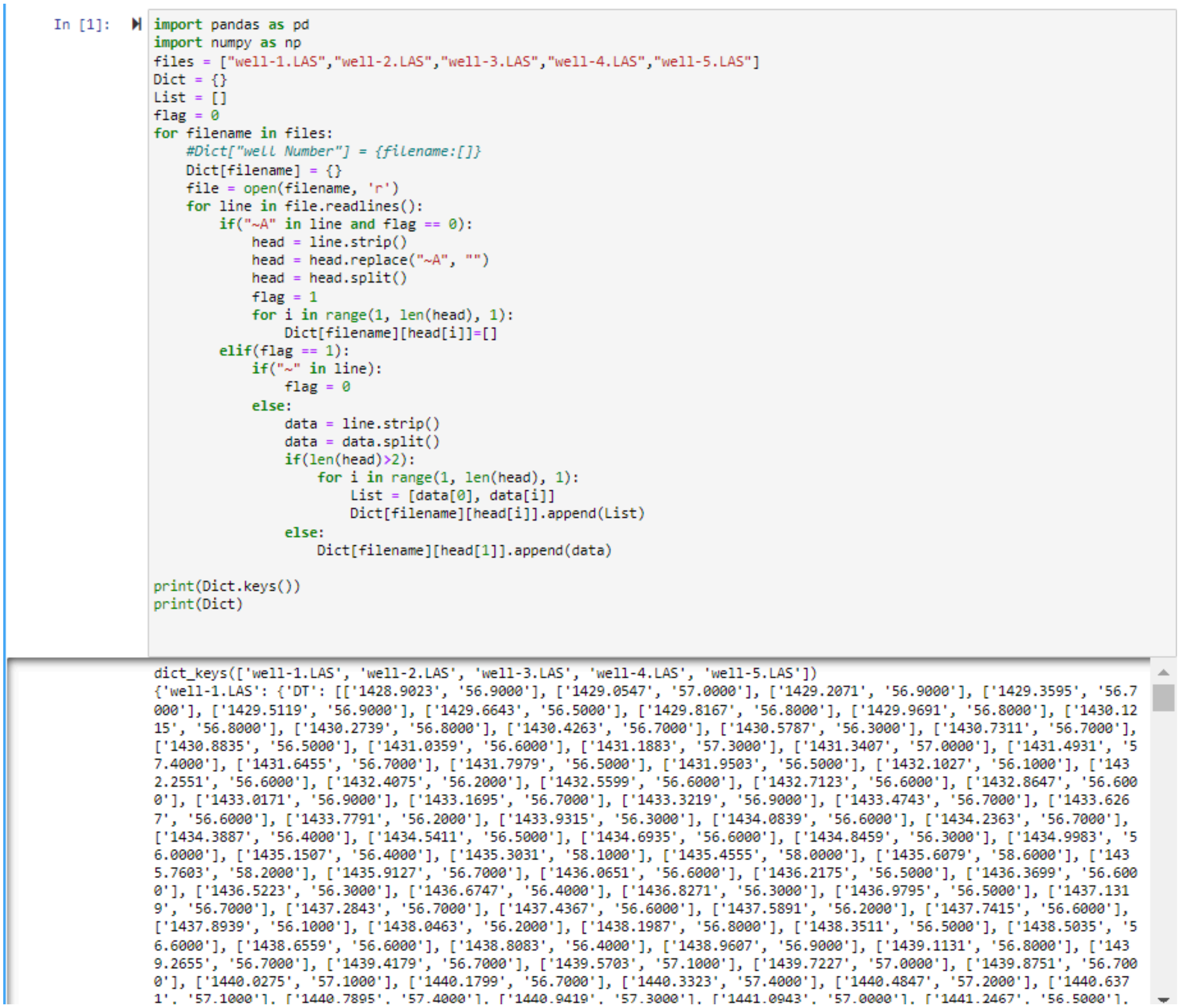
Once the model is trained, it can be tested on unseen or holdout well log data to assess its performance. The model predicts the well log parameters based on the test data, and the predictions

12. Documentation and Reporting:

Document the entire implementation process, including the steps taken, techniques employed, and the rationale behind decisions made. Prepare a comprehensive report detailing the findings, evaluation results, and recommendations for further improvements.

By following these implementation steps, the project aims to develop a robust framework for well data cleaning and prediction of values, enhancing the quality and usability of well datasets for reservoir characterization and decision-making in the oil and gas industry.

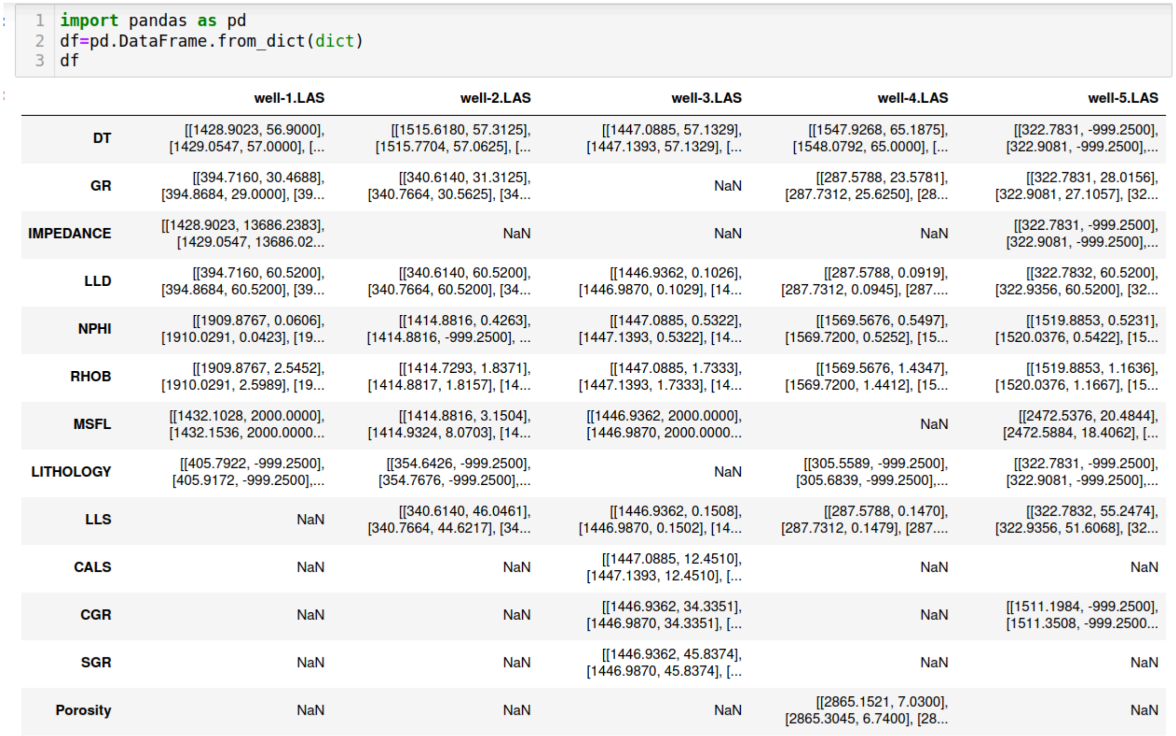
Code and Results



In [1]:

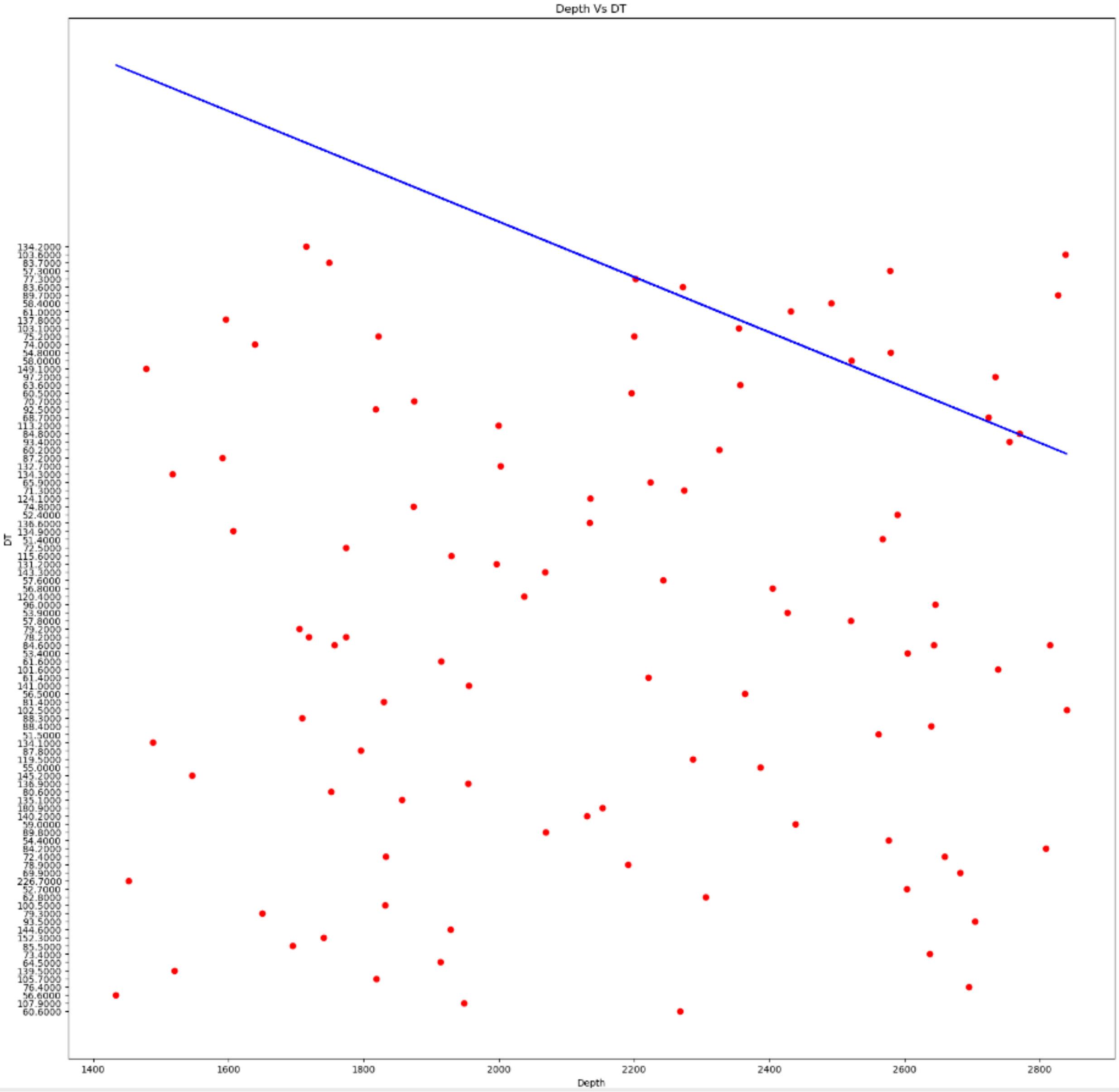
```
import pandas as pd
import numpy as np
files = ["well-1.LAS","well-2.LAS","well-3.LAS","well-4.LAS","well-5.LAS"]
Dict = {}
List = []
flag = 0
for filename in files:
    #Dict["well Number"] = {filename:[]}
    Dict[filename] = {}
    file = open(filename, 'r')
    for line in file.readlines():
        if("~A" in line and flag == 0):
            head = line.strip()
            head = head.replace("~A", "")
            head = head.split()
            flag = 1
            for i in range(1, len(head), 1):
                Dict[filename][head[i]]=[]
        elif(flag == 1):
            if("~" in line):
                flag = 0
            else:
                data = line.strip()
                data = data.split()
                if(len(head)>2):
                    for i in range(1, len(head), 1):
                        List = [data[0], data[i]]
                        Dict[filename][head[i]].append(List)
                else:
                    Dict[filename][head[1]].append(data)

print(Dict.keys())
print(Dict)
```

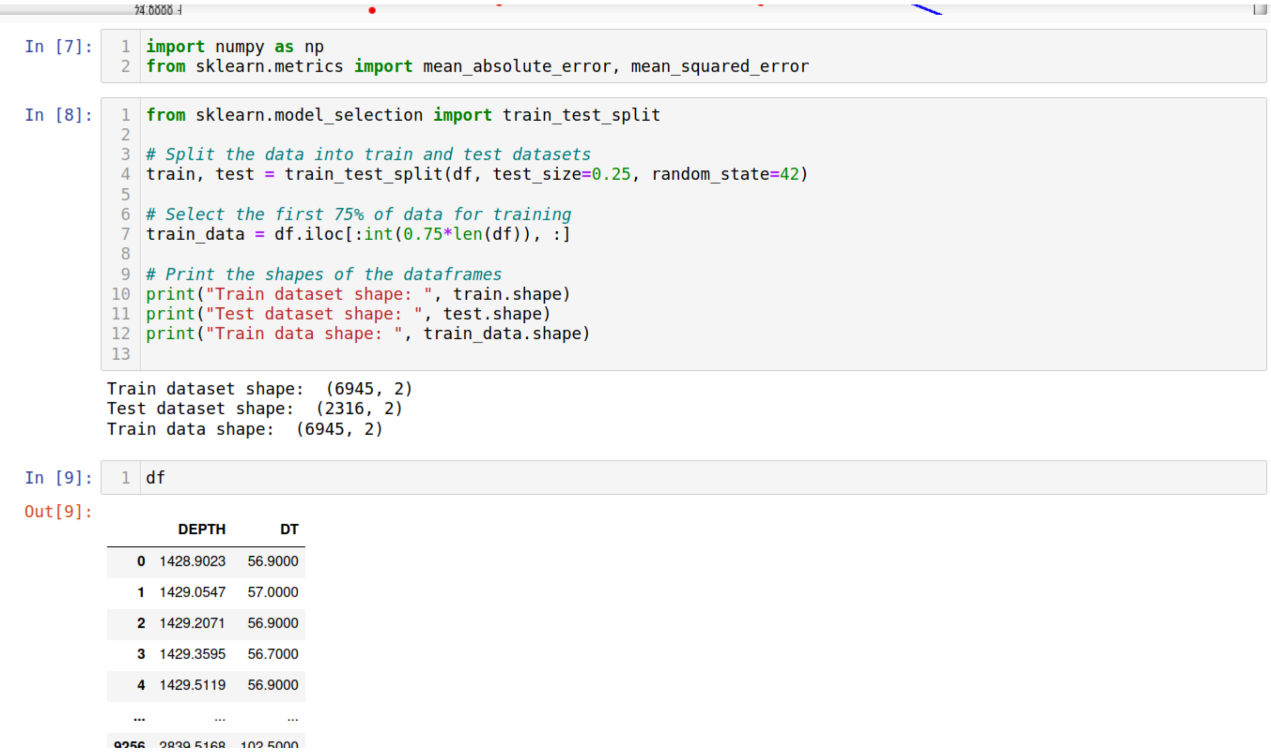
```
dict_keys(['well-1.LAS', 'well-2.LAS', 'well-3.LAS', 'well-4.LAS', 'well-5.LAS'])
{'well-1.LAS': {'DT': [['1428.9023', '56.9000'], ['1429.0547', '57.0000'], ['1429.2071', '56.9000'], ['1429.3595', '56.7
000'], ['1429.5119', '56.9000'], ['1429.6643', '56.5000'], ['1429.8167', '56.8000'], ['1429.9691', '56.8000'], ['1430.12
15', '56.8000'], ['1430.2739', '56.8000'], ['1430.4263', '56.7000'], ['1430.5787', '56.3000'], ['1430.7311', '56.7000'],
['1430.8835', '56.5000'], ['1431.0359', '56.6000'], ['1431.1883', '57.3000'], ['1431.3407', '57.0000'], ['1431.4931', '5
7.4000'], ['1431.6455', '56.7000'], ['1431.7979', '56.5000'], ['1431.9503', '56.5000'], ['1432.1027', '56.1000'], ['143
2.2551', '56.6000'], ['1432.4075', '56.2000'], ['1432.5599', '56.6000'], ['1432.7123', '56.6000'], ['1432.8647', '56.600
0'], ['1433.0171', '56.9000'], ['1433.1695', '56.7000'], ['1433.3219', '56.9000'], ['1433.4743', '56.7000'], ['1433.626
7', '56.6000'], ['1433.7791', '56.2000'], ['1433.9315', '56.3000'], ['1434.0839', '56.6000'], ['1434.2363', '56.7000'],
['1434.3887', '56.4000'], ['1434.5411', '56.5000'], ['1434.6935', '56.6000'], ['1434.8459', '56.3000'], ['1434.9983', '5
6.0000'], ['1435.1507', '56.4000'], ['1435.3031', '58.1000'], ['1435.4555', '58.0000'], ['1435.6079', '58.6000'], ['143
5.7603', '58.2000'], ['1435.9127', '56.7000'], ['1436.0651', '56.6000'], ['1436.2175', '56.5000'], ['1436.3699', '56.600
0'], ['1436.5223', '56.3000'], ['1436.6747', '56.4000'], ['1436.8271', '56.3000'], ['1436.9795', '56.5000'], ['1437.131
9', '56.7000'], ['1437.2843', '56.7000'], ['1437.4367', '56.6000'], ['1437.5891', '56.2000'], ['1437.7415', '56.6000'],
['1437.8939', '56.1000'], ['1438.0463', '56.2000'], ['1438.1987', '56.8000'], ['1438.3511', '56.5000'], ['1438.5035', '5
6.6000'], ['1438.6559', '56.6000'], ['1438.8083', '56.4000'], ['1438.9607', '56.9000'], ['1439.1131', '56.8000'], ['143
9.2655', '56.7000'], ['1439.4179', '56.7000'], ['1439.5703', '57.1000'], ['1439.7227', '57.0000'], ['1439.8751', '56.700
0'], ['1440.0275', '57.1000'], ['1440.1799', '56.7000'], ['1440.3323', '57.4000'], ['1440.4847', '57.2000'], ['1440.637
1', '57.1000'], ['1440.7895', '57.4000'], ['1440.9419', '57.3000'], ['1441.0943', '57.0000'], ['1441.2467', '56.5000'],
```



```python
import pandas as pd
df=pd.DataFrame.from_dict(dict)
df
```

	well-1.LAS	well-2.LAS	well-3.LAS	well-4.LAS	well-5.LAS
DT	[[1428.9023, 56.9000], [1429.0547, 57.0000], [...	[[1515.6180, 57.3125], [1515.7704, 57.0625], [...	[[1447.0885, 57.1329], [1447.1393, 57.1329], [...	[[1547.9268, 65.1875], [1548.0792, 65.0000], [...	[[322.7831, -999.2500], [322.9081, -999.2500],...
GR	[[394.7160, 30.4688], [394.8684, 29.0000], [39...	[[340.6140, 31.3125], [340.7664, 30.5625], [34...	NaN	[[287.5788, 23.5781], [287.7312, 25.6250], [28...	[[322.7831, 28.0156], [322.9081, 27.1057], [32...
IMPEDANCE	[[1428.9023, 13686.2383], [1429.0547, 13686.02...	NaN	NaN	NaN	[[322.7831, -999.2500], [322.9081, -999.2500],...
LLD	[[394.7160, 60.5200], [394.8684, 60.5200], [39...	[[340.6140, 60.5200], [340.7664, 60.5200], [34...	[[1446.9362, 0.1026], [1446.9870, 0.1029], [14...	[[287.5788, 0.0919], [287.7312, 0.0945], [287....	[[322.7832, 60.5200], [322.9356, 60.5200], [32...
NPHI	[[1909.8767, 0.0606], [1910.0291, 0.0423], [19...	[[1414.8816, 0.4263], [1414.8816, -999.2500], ...	[[1447.0885, 0.5322], [1447.1393, 0.5322], [14...	[[1569.5676, 0.5497], [1569.7200, 0.5252], [15...	[[1519.8853, 0.5231], [1520.0376, 0.5422], [15...
RHOB	[[1909.8767, 2.5452], [1910.0291, 2.5989], [19...	[[1414.7293, 1.8371], [1414.8817, 1.8157], [14...	[[1447.0885, 1.7333], [1447.1393, 1.7333], [14...	[[1569.5676, 1.4347], [1569.7200, 1.4412], [15...	[[1519.8853, 1.1636], [1520.0376, 1.1667], [15...
MSFL	[[1432.1028, 2000.0000], [1432.1536, 2000.0000...	[[1414.8816, 3.1504], [1414.9324, 8.0703], [14...	[[1446.9362, 2000.0000], [1446.9870, 2000.0000...	NaN	[[2472.5376, 20.4844], [2472.5884, 18.4062], [...
LITHOLOGY	[[405.7922, -999.2500], [405.9172, -999.2500],...	[[354.6426, -999.2500], [354.7676, -999.2500],...	NaN	[[305.5589, -999.2500], [305.6839, -999.2500],...	[[322.7831, -999.2500], [322.9081, -999.2500],...
LLS	NaN	[[340.6140, 46.0461], [340.7664, 44.6217], [34...	[[1446.9362, 0.1508], [1446.9870, 0.1502], [14...	[[287.5788, 0.1470], [287.7312, 0.1479], [287....	[[322.7832, 55.2474], [322.9356, 51.6068], [32...
CALS	NaN	NaN	[[1447.0885, 12.4510], [1447.1393, 12.4510], [...	NaN	NaN
CGR	NaN	NaN	[[1446.9362, 34.3351], [1446.9870, 34.3351], [...	NaN	[[1511.1984, -999.2500], [1511.3508, -999.2500...
SGR	NaN	NaN	[[1446.9362, 45.8374], [1446.9870, 45.8374], [...	NaN	NaN
Porosity	NaN	NaN	NaN	[[2865.1521, 7.0300], [2865.3045, 6.7400], [28...	NaN


Depth Vs DT
DT
134.2000
103.6000
83.7000
57.3000
77.3000
83.6000
89.7000
58.4000
61.0000
137.8000
103.1000
75.2000
74.0000
54.8000
58.0000
149.1000
97.2000
63.6000
60.5000
70.7000
92.5000
68.7000
113.2000
84.8000
93.4000
60.2000
87.2000
132.7000
134.3000
65.9000
71.3000
124.1000
74.8000
52.4000
138.6000
134.9000
51.4000
72.5000
115.6000
131.2000
143.3000
57.6000
56.8000
120.4000
96.0000
53.9000
57.8000
79.2000
78.2000
84.6000
53.4000
61.6000
101.6000
61.4000
141.0000
56.5000
81.4000
102.5000
88.3000
88.4000
51.5000
134.1000
87.8000
119.5000
55.0000
145.2000
136.9000
80.6000
135.1000
180.9000
140.2000
59.0000
89.8000
54.4000
84.2000
72.4000
78.9000
69.9000
226.7000
52.7000
62.8000
100.5000
79.3000
93.5000
144.6000
152.3000
85.5000
73.4000
64.5000
139.5000
105.7000
76.4000
56.6000
107.9000
60.6000
1400
1600
1800
2000
2200
2400
2600
2800
Depth



In [7]:
```python
import numpy as np
from sklearn.metrics import mean_absolute_error, mean_squared_error
```

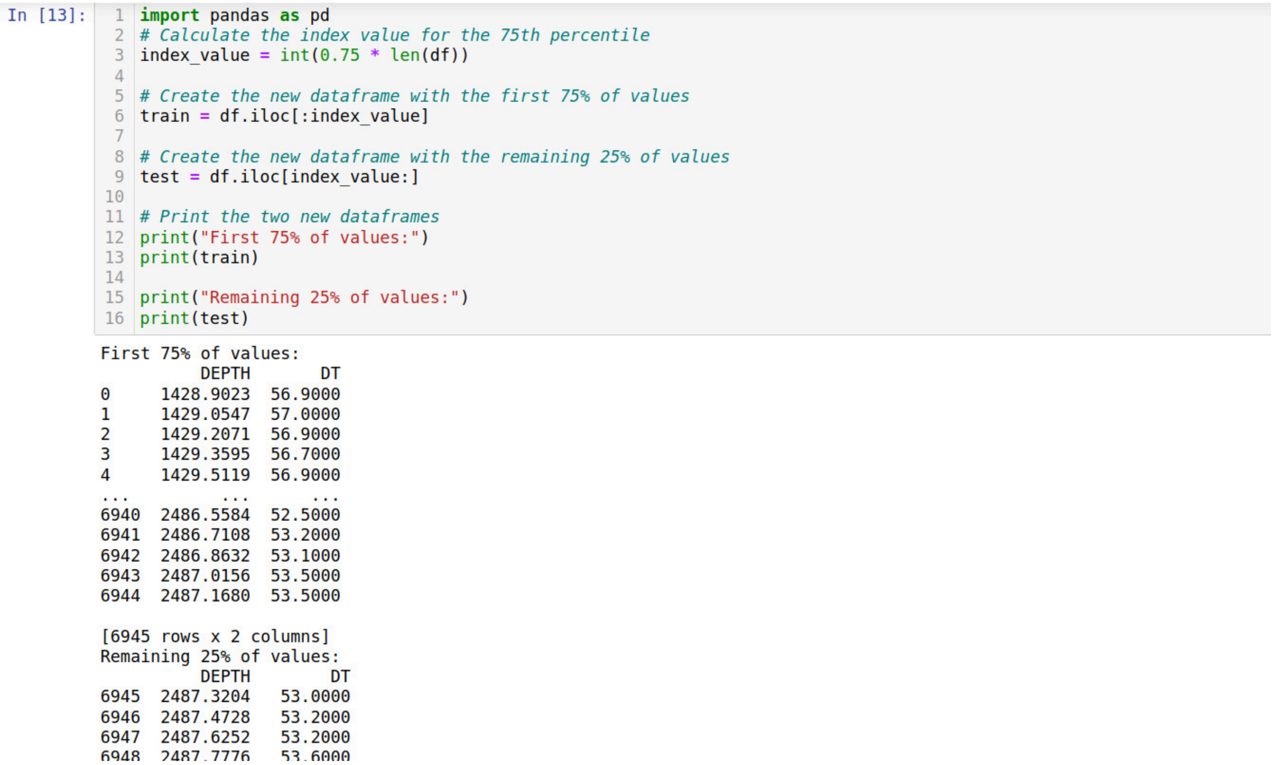
In [8]:
```python
from sklearn.model_selection import train_test_split

# Split the data into train and test datasets
train, test = train_test_split(df, test_size=0.25, random_state=42)

# Select the first 75% of data for training
train_data = df.iloc[:int(0.75*len(df)), :]

# Print the shapes of the dataframes
print("Train dataset shape: ", train.shape)
print("Test dataset shape: ", test.shape)
print("Train data shape: ", train_data.shape)

```

```
Train dataset shape:  (6945, 2)
Test dataset shape:  (2316, 2)
Train data shape:  (6945, 2)
```

In [9]:
```python
df
```

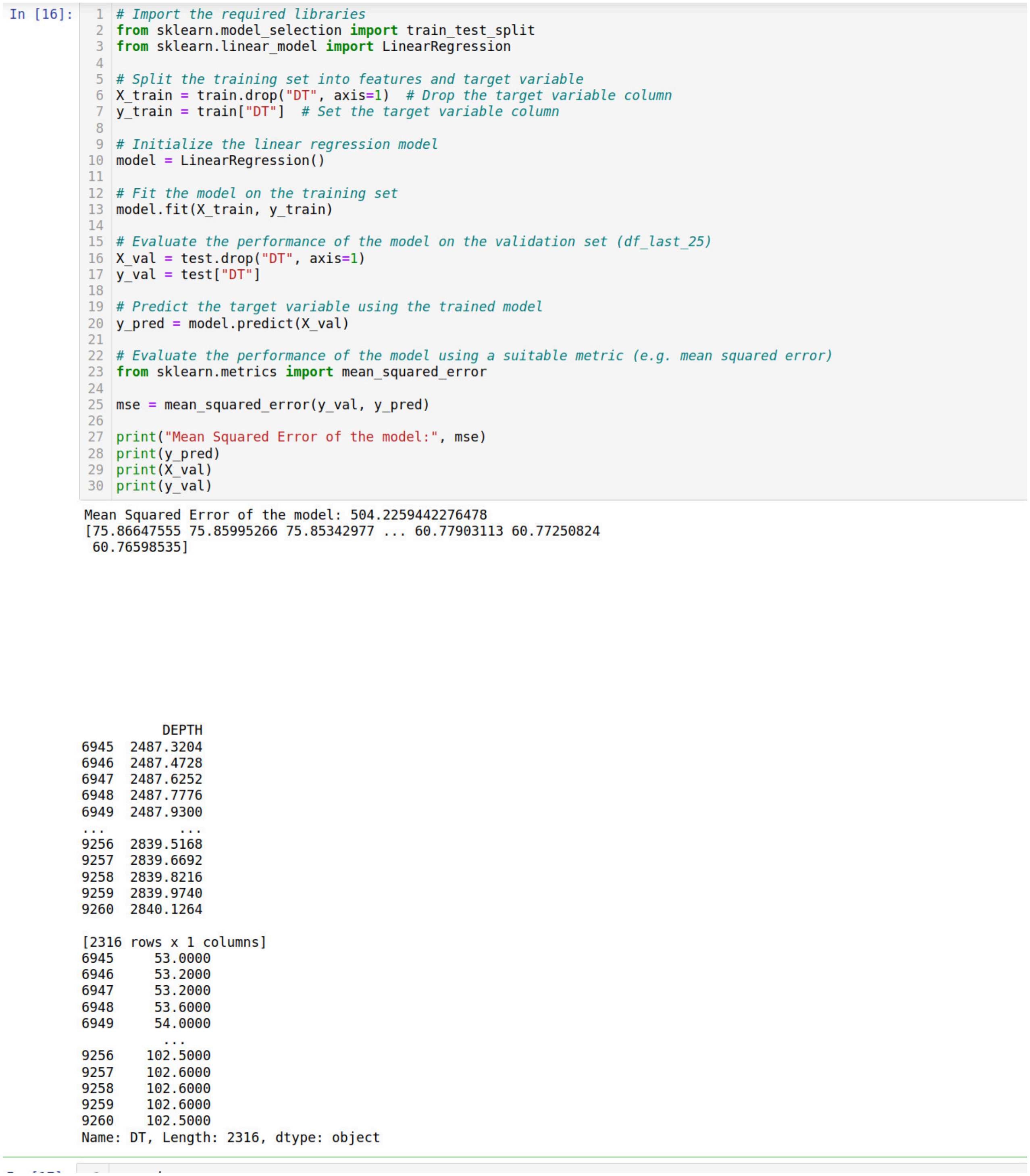
Out[9]:

	DEPTH	DT
0	1428.9023	56.9000
1	1429.0547	57.0000
2	1429.2071	56.9000
3	1429.3595	56.7000
4	1429.5119	56.9000
...	...	...
9256	2839.5168	102.5000



In [13]:
```python
import pandas as pd
# Calculate the index value for the 75th percentile
index_value = int(0.75 * len(df))

# Create the new dataframe with the first 75% of values
train = df.iloc[:index_value]

# Create the new dataframe with the remaining 25% of values
test = df.iloc[index_value:]

# Print the two new dataframes
print("First 75% of values:")
print(train)

print("Remaining 25% of values:")
print(test)
```

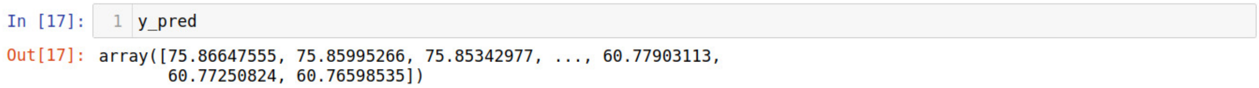
```
First 75% of values:
          DEPTH       DT
0     1428.9023  56.9000
1     1429.0547  57.0000
2     1429.2071  56.9000
3     1429.3595  56.7000
4     1429.5119  56.9000
...         ...      ...
6940  2486.5584  52.5000
6941  2486.7108  53.2000
6942  2486.8632  53.1000
6943  2487.0156  53.5000
6944  2487.1680  53.5000

[6945 rows x 2 columns]
Remaining 25% of values:
          DEPTH        DT
6945  2487.3204   53.0000
6946  2487.4728   53.2000
6947  2487.6252   53.2000
6948  2487.7776   53.6000
```



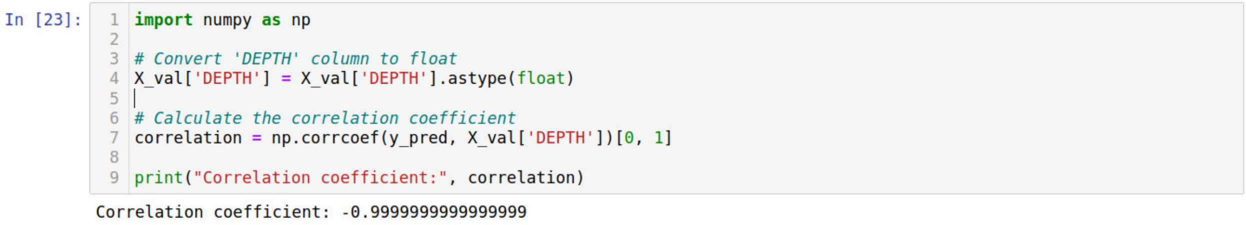
```
In [16]:
# Import the required libraries
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

# Split the training set into features and target variable
X_train = train.drop("DT", axis=1)  # Drop the target variable column
y_train = train["DT"]  # Set the target variable column

# Initialize the linear regression model
model = LinearRegression()

# Fit the model on the training set
model.fit(X_train, y_train)

# Evaluate the performance of the model on the validation set (df_last_25)
X_val = test.drop("DT", axis=1)
y_val = test["DT"]

# Predict the target variable using the trained model
y_pred = model.predict(X_val)

# Evaluate the performance of the model using a suitable metric (e.g. mean squared error)
from sklearn.metrics import mean_squared_error

mse = mean_squared_error(y_val, y_pred)

print("Mean Squared Error of the model:", mse)
print(y_pred)
print(X_val)
print(y_val)
```

```
Mean Squared Error of the model: 504.2259442276478
[75.86647555 75.85995266 75.85342977 ... 60.77903113 60.77250824
 60.76598535]
```

```
          DEPTH
6945  2487.3204
6946  2487.4728
6947  2487.6252
6948  2487.7776
6949  2487.9300
...         ...
9256  2839.5168
9257  2839.6692
9258  2839.8216
9259  2839.9740
9260  2840.1264

[2316 rows x 1 columns]
6945     53.0000
6946     53.2000
6947     53.2000
6948     53.6000
6949     54.0000
          ...
9256    102.5000
9257    102.6000
9258    102.6000
9259    102.6000
9260    102.5000
Name: DT, Length: 2316, dtype: object
```



```
In [17]: y_pred
Out[17]: array([75.86647555, 75.85995266, 75.85342977, ..., 60.77903113,
                60.77250824, 60.76598535])
```



In [23]:

```python
import numpy as np

# Convert 'DEPTH' column to float
X_val['DEPTH'] = X_val['DEPTH'].astype(float)

# Calculate the correlation coefficient
correlation = np.corrcoef(y_pred, X_val['DEPTH'])[0, 1]

print("Correlation coefficient:", correlation)
```

```
Correlation coefficient: -0.9999999999999999
```

Future Scope

The project has a promising future scope. Machine learning has the potential to revolutionise the oil and gas industry by enabling more accurate and efficient predictions of well log parameters. Here are some potential areas for future development and expansion of this project:

1. Integration with IoT devices: Internet of Things (IoT) devices can collect a vast amount of data from drilling operations. Integrating this data with machine learning algorithms can greatly improve the accuracy of well log parameter predictions.
2. Real-time monitoring: Machine learning can be used to provide real-time monitoring of well drilling operations. This can help identify potential hazards and prevent accidents.
3. Optimization of drilling operations: The accurate prediction of well log parameters can help optimise drilling operations and reduce costs.
4. Predictive maintenance: Machine learning algorithms can be used to predict equipment failure. This can help avoid unplanned downtime and reduce maintenance costs.
5. Automated decision-making: Machine learning can automate decision-making processes in drilling operations. This can help improve efficiency and reduce the need for human intervention.

With continued development and integration with new technologies, this project has the potential to greatly improve drilling operations and revolutionise the oil and gas industry.

Conclusion

In conclusion, this project has addressed the critical problem of accurately predicting well log parameters in the petroleum industry using machine learning techniques. By leveraging well log datasets, feature engineering, and advanced machine learning algorithms, the project has demonstrated the potential to improve the accuracy, efficiency, and robustness of well log parameter predictions.

The developed methodology provides a foundation for enhancing reservoir characterization, formation evaluation, and decision-making processes. It overcomes the limitations of traditional methods by leveraging data-driven models that can capture complex relationships within the well log data.

The future scope of this project is promising, with potential areas for further exploration. Advanced machine learning techniques, such as deep learning and ensemble methods, can be investigated to further improve predictive performance. Incorporating additional data sources and exploring uncertainty quantification techniques will enhance the comprehensive understanding of reservoirs and support more informed decision-making.

Transfer learning and domain adaptation techniques offer opportunities to leverage knowledge from related domains or transfer models between fields, improving predictions in data-scarce environments. Real-time prediction and automation can enable timely decision-making, while integration with decision support systems ensures seamless integration into existing workflows.

By pursuing these future directions, this project can contribute to advancing well log parameter prediction techniques, empowering the petroleum industry with more accurate predictions, informed decision-making, and optimised reservoir management strategies. Ultimately, this work aims to improve hydrocarbon resource estimation and maximise the value of oil and gas assets.

References

[1] www.stackoverflow.com/questions/

[2] https://datagy.io/mean-squared-error-python/

[3] https://towardsdatascience.com/

[4] https://www.geeksforgeeks.org/

[5] https://unix.stackexchange.com/

[6] https://realpython.com/python-data-cleaning-numpy-pandas/

[7] https://deepnote.com/

[8] https://scikit-learn.org/