Estimate The Presence Of Impurities In Iron Ore Using IBM Watson Machine Learning

1 INTRODUCTION

1.1 Overview

Aim of the project is to predict the impurities present in an Iron ore. The ores are usually rich in iron oxides and vary in colour from dark grey, bright yellow, or deep purple to rusty red. The iron is usually found in the form of magnetite, hematite, goethite, limonite or siderite. Usually, Magnetite Iron ore concentrate contains an impurity of 3–7% of silica. Estimation of silica involves a lot of chemical analysis which is time-consuming and involves high operational cost. In order to cut down the operational cost and also to help engineers by predicting at a faster rate, we make use of Machine Learning (ML). So the main goal of this project is to build a Machine Learning model to predict the impurities present in an Iron ore.

1.2 Purpose

Using IBM Watson Studio we train the data set using Random Forest Regression algorithm that help to train the model with the help of machine learning services provided by the IBM. Using the dataset which have the existing sample data of the quality determining experimented values, Machine learn and study the variation according to the values of average air flow, average floating level, percentile of iron feed, amina flow, ore pulp pH and ore pulp density. So according to these factors impurities quality can be predicted machine will learn about it using the algorithm. As the impurity is measured every hour, if we can predict how much silica (impurity) is in the ore concentrate, we can help the engineers in the plant to take measurements in early stages of manufacturing. To help the environment by reducing the amount of ore that goes to tailing as you reduce silica in the ore concentrate .

2 LITERATURE SURVEY

2.1 Existing problem

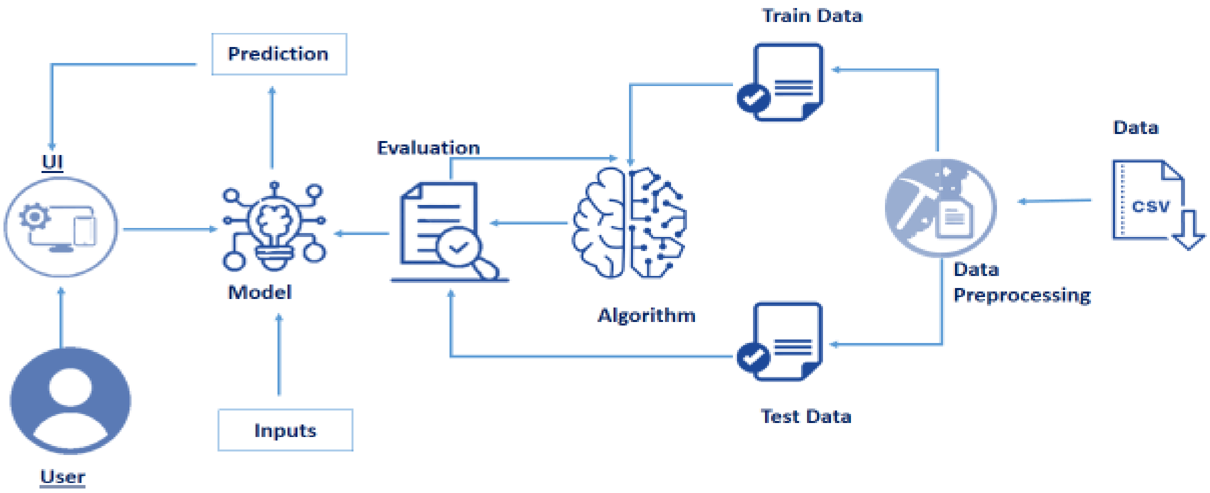
In the project we have to predict the **% Silica Concentrate**. Silica Concentrate is the impurity in the iron ore which needs to be removed. The current process of detecting silica takes many hours. The developed process of silica and iron recovery consisted of size classification, magnetic separation, and removal of impurities by leaching. Up to about 80% gangue is removed by chemical leaching with sodium hydroxide, sulphuric, hydrochloric and nitric acids. More alumina and silica were removed during magnetic separation followed by flotation process because the iron-silica and iron-alumina aggregated minerals were separated more by their difference in floatability, because there are weak magnetic particles and was difficult to separate them during two stage magnetic separation.

2.2 Proposed solution

To eliminate all these difficulties and problems we proposed a system with the help of some analysis and modelling of data we can give a good approximation of silica concentrate which will reduce a lot of time and effort required for processing iron ore by using the random forest regression algorithm.

3 THEORITICAL ANALYSIS

3.1 Block diagram



3.2 Hardware / Software designing

IBM Watson Studio - IBM Watson Studio helps data scientists and analysts prepare data and build models at scale across any cloud. IBM Watson Machine Learning - IBM Watson Machine Learning helps data scientists and developers accelerate AI and machine learning deployment. IBM Cloud Object Storage - IBM Cloud Object Storage makes it possible to store practically limitless amounts of data, simply and cost effectively. Machine Learning Services - Machine learning as service is an umbrella term for collection of various cloud-based platforms that use machine learning tools to provide solutions that can help ML teams with: out-of-the box predictive analysis for various use cases, data pre-processing, and model training.

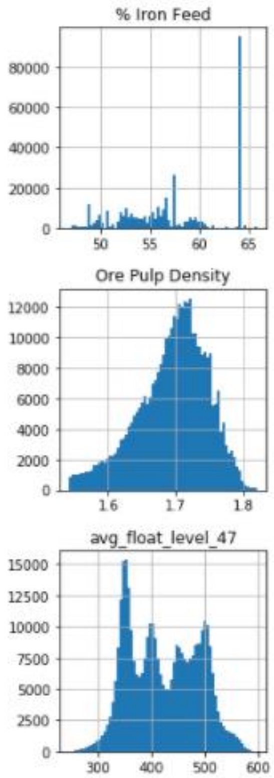
3 EXPERIMENTAL INVESTIGATIONS

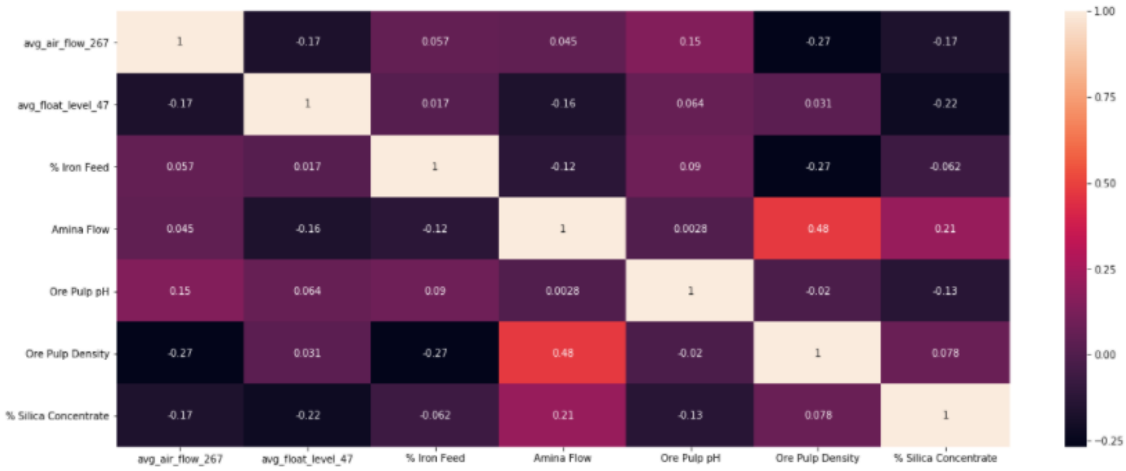
Here we are going to build a machine learning model that predicts the impurities present in an iron ore based on the following parameters:

- average air flow
- average floating level
- percentile of iron feed
- amina flow
- ore pulp pH
- ore pulp density

4.1 Experimental Analysis

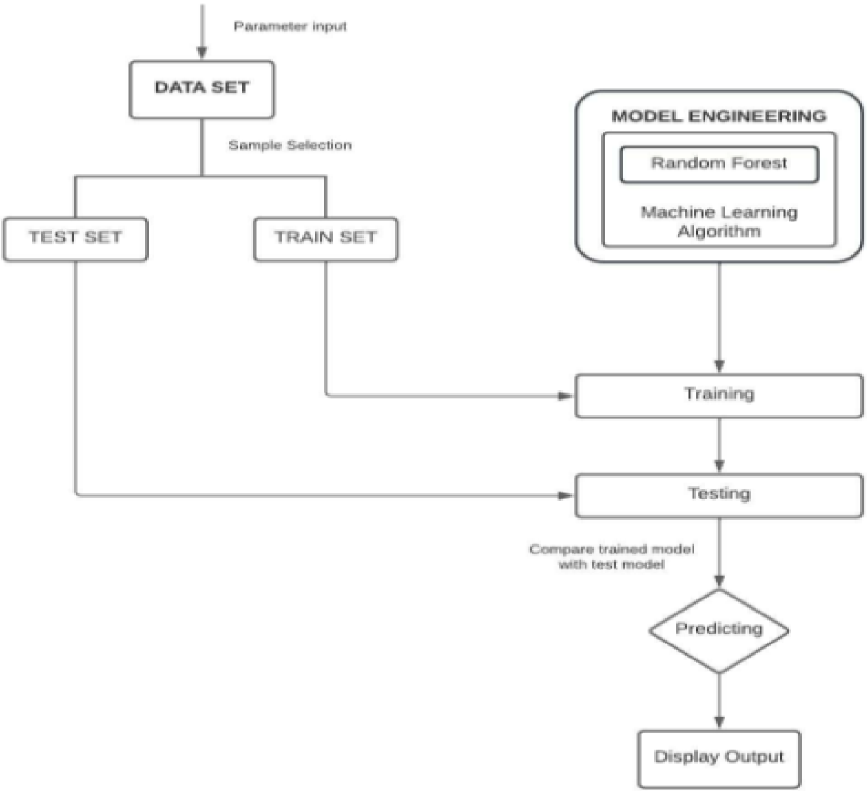
Visualization



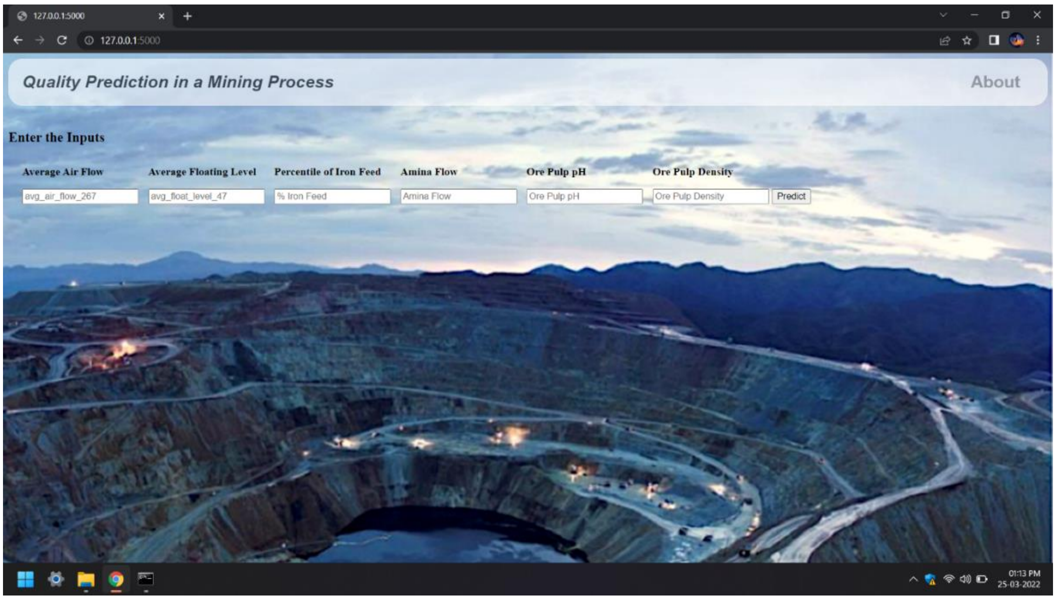


Above plot shows the correaltions between the features. From the plot we can find out the features which affects the % Silica Concentrate the most

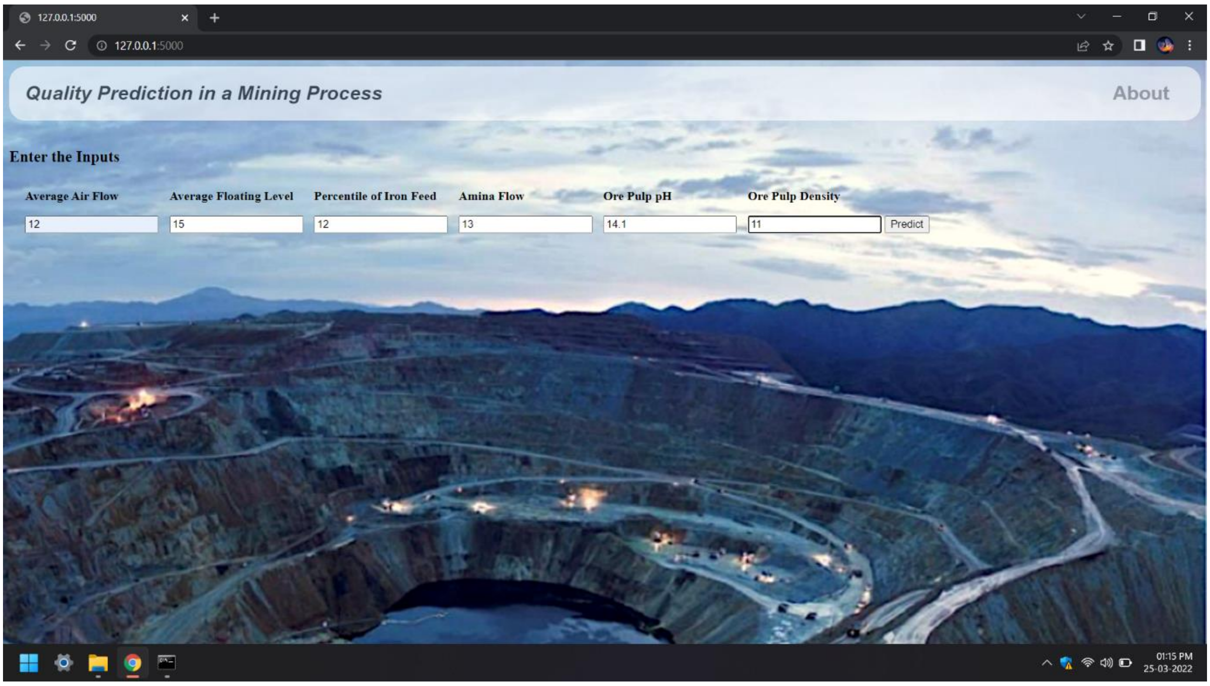
5 FLOWCHART



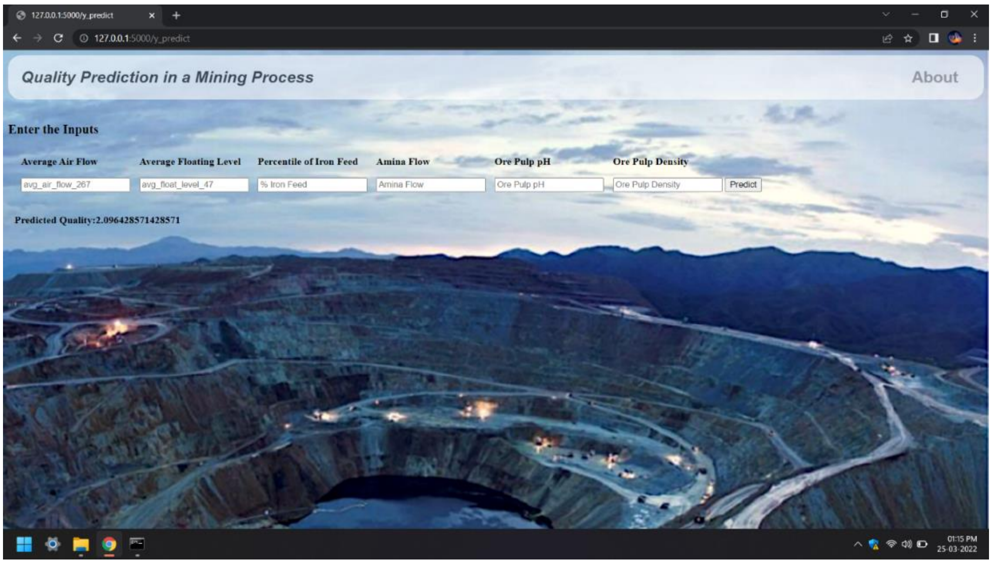
6 RESULT



When we input the values in each input fields after finding the average air flow, average floating level, percentile of iron feed, amina flow, ore pulp pH and ore pulp density.



Input as average air flow: 12 , average floating level:15, percentile of iron feed: 12, amina flow:13, ore pulp pH: 14.1, ore pulp density: 11



Here we got the predicted quality as 2.096428571428571

7 ADVANTAGES AND DISADVANTAGES

7.1 Advantages

1. Faster Claim Settlements And Save Time

We can estimate the output very fast compared to other system and save time due to machine learning, so we can estimate the presence of impurities in iron ore using the machine learning techniques.

2. Automation of Everything

Machine Learning is responsible for cutting the workload and time. By automating things we let the algorithm do the hard work for us. Automation is now being done almost everywhere. The reason is that it is very reliable. Also, it helps us to think more creatively.

Due to ML, we are now designing more advanced computers. These computers can handle various Machine Learning models and algorithms efficiently. Even though automation is spreading fast, we still don’t completely rely on it. ML is slowly transforming the industry with its automation.

3. Wide Range of Applications

ML has a wide variety of applications. This means that we can apply ML on any of the major fields. ML has its role everywhere from medical, business, banking to science and tech. This helps to create more opportunities. It plays a major role in customer interactions.

Machine Learning can help in the detection of diseases more quickly. It is helping to lift up businesses. That is why investing in ML technology is worth it.

4. Scope of Improvement

Machine Learning is the type of technology that keeps on evolving. There is a lot of scope in ML to become the top technology in the future. The reason is, it has a lot of research areas in it. This helps us to improve both hardware and software.

In hardware, we have various laptops and GPUs. These have various ML and Deep Learning networks in them. These help in the faster processing power of the system. When it comes to software we have various UIs and libraries in use. These help in designing more efficient algorithms.

4. Efficient Handling of Data

Machine Learning has many factors that make it reliable. One of them is data handling. ML plays the biggest role when it comes to data at this time. It can handle any type of data.

Machine Learning can be multidimensional or different types of data. It can process and analyse these data that normal systems can't. Data is the most important part of any Machine Learning model. Also, studying and handling of data is a field in itself.

7.2 Disadvantages

1. Possibility of High Error

In ML, we can choose the algorithms based on accurate results. For that, we have to run the results on every algorithm. The main problem occurs in the training and testing of data. The data is huge, so sometimes removing errors becomes nearly impossible. These errors can cause a headache to users. Since the data is huge, the errors take a lot of time to resolve.

2. Algorithm Selection

The selection of an algorithm in Machine Learning is still a manual job. We have to run and test our data in all the algorithms. After that only we can decide what algorithm we want. We choose them on the basis of result accuracy. The process is very much time-consuming.

3. Data Acquisition

In ML, we constantly work on data. We take a huge amount of data for training and testing. This process can sometimes cause data inconsistency. The reason is some data constantly keep on updating. So, we have to wait for the new data to arrive. If not, the old and new data might give different results. That is not a good sign for an algorithm.

4. Time and Space

Many ML algorithms might take more time than you think. Even if it's the best algorithm it might sometimes surprise you. If your data is large and advanced, the system will take time. This may sometimes cause the consumption of more CPU power. Even with GPUs alongside, it sometimes becomes hectic. Also, the data might use more than the allotted space.

8 APPLICATIONS

To predict the impurities present in the iron ore machine learning applications are used.

9 CONCLUSION

In this study, a multi-target regression problem is handled to predict quality in a mining process. The aim is to construct a robust model that simultaneously estimates the amount of silica and iron concentrates in the ore. Several approaches are implemented and compared to be able to handle more than one target variable. We tried to observe the performance of a multi target regression approach when target features are highly correlated. At the end, it is noticed that this approach can also be efficient in manufacturing data when a related attribute is not given to the algorithm as an input parameter. Instead, that feature can also be evaluated as an output variable by being added to the existing target feature. We have observed that this alteration did not create an adverse effect on the regression performance. Finally, the experimental results demonstrate the superiority of AdaBoost regressor .

10 FUTURE SCOPE

Achieving predictions on contaminant levels has a high impact on quality insurance and it can help technicians and engineers to make adjustments in advance to improve the quality of the final product, and thus profits. In this paper, exploratory research is performed on the problem of silica concentrate estimation for iron ore using machine learning techniques, with the goal to identify algorithms that may be suitable for industry soft sensor development.

11 BIBILOGRAPHY

https://www.kaggle.com/datasets/edumagalhaes/quality-prediction-in-a-mining-process

https://medium.com/@sivashankarimalathyveda/quality-prediction-in-a-mining-process-using-machine-learning-ea6450b6a5d9

http://ivyproschool.com/blog/advantages-and-disadvantages-of-machine-learning-in-2020/

APPENDIX

A. Source Code

```
import os, types

import pandas as pd

from botocore.client import Config

import ibm_boto3

def __iter__(self): return 0

client_e81fba6ff66d48bf8f1a419a065c725e = ibm_boto3.client(service_name='s3',ibm_api_key_id='YQATAJLj2ZIG4CqF-y8pdYOyeD5ROe0_l67JwJreDthv',
ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",

config=Config(signature_version='oauth'),

    endpoint_url='https://s3.private.eu.cloud-object-storage.appdomain.cloud')

body = client_e81fba6ff66d48bf8f1a419a065c725e.get_object(Bucket='estimatethepresenceofimpuritiesin-donotdelete-pr-yoriit6zucnnbb',Key='Mining_Dataset.csv')['Body']

if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

mining_data = pd.read_csv(body,decimal=',',parse_dates=["date"],infer_datetime_format=True, sep=',')

mining_data.head()

mining_data.describe()

for cols in mining_data.columns.tolist()[1:]:

    df = mining_data.loc[mining_data[cols] > 0]

df.info()

df=df.set_index('date')

import matplotlib.pyplot as plt

from matplotlib import style
```

```
df.hist(bins = 70, figsize = (17,15))
plt.show()
plt.suptitle('figure title', color='w')

df_copy=df
from scipy import stats
df_copy=df[(np.abs(stats.zscore(df)) < 2).all(axis=1)]
df_copy.info()
df_copy.hist(bins = 70, figsize = (17,15))
plt.show()
plt.suptitle('figure title', color='w')
df_copy.info()
df_copy.drop(columns=['% Iron Concentrate'],inplace=True)
df_copy.head()
import seaborn as sns
ml_mining_data=df_copy
plt.figure(figsize=(20, 20))
p = sns.heatmap(ml_mining_data.corr(), annot=True);
df=ml_mining_data.drop(['% Silica Feed', 'Starch Flow','Ore Pulp Flow','Flotation Column 01 Air Flow',
                    'Flotation Column 03 Air Flow', 'Flotation Column 04 Air Flow','Flotation Column 05 Air Flow',
                    'Flotation Column 01 Level','Flotation Column 02 Level', 'Flotation Column 03 Level',
                    'Flotation Column 05 Level','Flotation Column 06 Level'], axis = 1)
df.head()
df['avg_air_flow_267'] = df[['Flotation Column 02 Air Flow','Flotation Column 06 Air Flow',
```

```
                'Flotation Column 07 Air Flow']].mean(axis=1)
df['avg_float_level_47'] = df[['Flotation Column 04 Level', 'Flotation Column 07 Level']].mean(axis=1)
df.head()
plt.figure(figsize=(20, 10))
p = sns.heatmap(df.corr(), annot=True);
df_ml=df.drop(['Flotation Column 02 Air Flow','Flotation Column 06 Air Flow','Flotation Column 07 Air Flow',
        'Flotation Column 04 Level', 'Flotation Column 07 Level'], axis = 1)
df_ml.head()
cols = df_ml.columns.tolist()
cols
cols = cols[-1:] + cols[:-1]
cols = cols[-1:] + cols[:-1]
df_ml = df_ml[cols]
df_ml.head()
df_ml.info()
plt.figure(figsize=(20, 8))
p = sns.heatmap(df_ml.corr(), annot=True);
df_ml.hist(bins = 70, figsize = (10,10))
plt.show()
plt.suptitle('figure title', color='w')
export_csv = df_ml.to_csv(header= True, index=True)
# Importing of Libraries
import pandas as pd
import numpy as np
body = client_e81fba6ff66d48bf8f1a419a065c725e.get_object(Bucket='estimatethepres
```

```
enceofimpuritiesin-donotdelete-pr-
yoriit6zucnnbb',Key='Enhanced_Mining_dataset.csv')['Body']

if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__,
body )

dataset=pd.read_csv(body,sep=',')

dataset=dataset.set_index('date')

dataset.head()

x=dataset.iloc[:,:-1].values

y=dataset.iloc[:,-1].values

x

y

from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)

from sklearn.ensemble import RandomForestRegressor

model=RandomForestRegressor(n_estimators=42,criterion='mse')

model.fit(x_train,y_train)

y_pred=model.predict(x_test)

y_pred

from sklearn.metrics import r2_score

r2_score(y_test,y_pred)

model.predict([[251.448000,483.4510,55.20,557.434,10.06640,1.74000]])

import ibm_watson_machine_learning

from ibm_watson_machine_learning import APIClient

import json

import numpy as np

wml_credentials = {

    "apikey" :"q5j82Gu7TWz752cZ5UJnzWpK_CDN8DD_khVukUc5Lz5T",

    "url" : "https://eu-gb.ml.cloud.ibm.com"
```

```
}
wml_client = APIClient(wml_credentials)
wml_client.spaces.list()
SPACE_ID = "42d8d342-4ead-48b0-bc02-ef5435ee3ba4"
wml_client.set.default_space(SPACE_ID)
MODEL_NAME = "quality prediction"
DEPLOYMENT_NAME = "prediction deployment"
BEST_MODEL= model
software_spec_uid =
wml_client.software_specifications.get_id_by_name("default_py3.8")
model_props = {
   wml_client.repository.ModelMetaNames.NAME:MODEL_NAME,
    wml_client.repository.ModelMetaNames.TYPE:'scikit-learn_0.23',
   wml_client.repository.ModelMetaNames.SOFTWARE_SPEC_UID :
software_spec_uid
}
model_details = wml_client.repository.store_model(
    model = BEST_MODEL,
    meta_props = model_props,
    training_data = x_train,
    training_target =y_train
)
software_spec_uid
model_uid = wml_client.repository.get_model_id(model_details)
model_uid
deployment_props = {

wml_client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_N
AME,
```

```
    wml_client.deployments.ConfigurationMetaNames.ONLINE:{}
}
deployment =
wml_client.deployments.create(artifact_uid=model_uid,meta_props=deployment_props)
```