

Date	15 March 2024
Team id	SWUID20240034764
Project title	Predicting Full Load Electrical Power Output of a Base Load Operated Combined Cycle Power Plant Using Machine Learning
Maximum marks	4 Marks

Initial Model Training Code

Model Validation and Evaluation Report The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

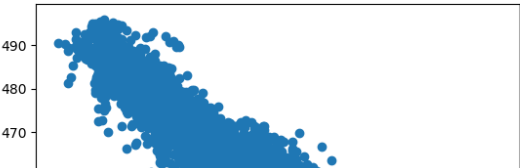
```
[3]: import pandas as pd
[4]: import numpy as np
[5]: import matplotlib.pyplot as plt
[6]: import seaborn as sns
[5]: import pickle
[21]: data = pd.read_csv(r"C:\Users\NEW\Desktop\projects\HL01.csv")

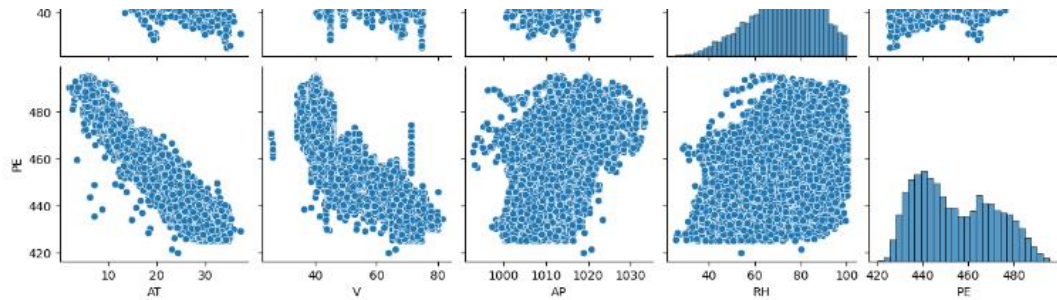
data = pd.read_csv(r"C:\Users\NEW\Desktop\projects\HL01.csv")

[40]: data.describe().T
```

	count	mean	std	min	25%	50%	75%	max
AT	9568.0	19.651231	7.452473	1.81	13.5100	20.345	25.72	37.11
V	9568.0	54.305804	12.707893	25.36	41.7400	52.080	66.54	81.56
AP	9568.0	1013.259078	5.938784	992.89	1009.1000	1012.940	1017.26	1033.30
RH	9568.0	73.308978	14.600269	25.56	63.3275	74.975	84.83	100.16
PE	9568.0	454.365009	17.066995	420.26	439.7500	451.550	468.43	495.76

```
[20]: plt.scatter(data['AT'], data['PE'])
[20]: <matplotlib.collections.PathCollection at 0x23adc2c1e80>
```





```
[45]: x = data.drop(['PE'], axis=1)
      y = data['PE']

[46]: from sklearn.model_selection import train_test_split
      xtrain, xtest, ytrain, ytest = train_test_split(x, y, test_size = 0.2, random_state = 0)

[47]: xtrain.shape

[47]: (7654, 4)

[48]: xtest.shape

[48]: (1914, 4)

[49]: from sklearn.linear_model import LinearRegression
      from sklearn.tree import DecisionTreeRegressor
      from sklearn.ensemble import RandomForestRegressor

[50]: from sklearn.linear_model import LinearRegression
      LRmodel = LinearRegression()
      LRmodel.fit(xtrain, ytrain)
      from sklearn.tree import DecisionTreeRegressor
      DTRmodel = DecisionTreeRegressor()
      DTRmodel.fit(xtrain, ytrain)
```

Model Validation and Evaluation Report:

model	Classification report	accuracy	Prediction matrix
Lr model	<pre>from sklearn.linear_model import LinearRegression LRmodel = LinearRegression() LRmodel.fit(xtrain, ytrain)</pre>	95	0.9325315554761302
Dt model	<pre># Initializing the model DTRmodel=DecisionTreeRegressor() # Train the data with Linear Regression model DTRmodel.fit(xtrain, ytrain) DTRpred=DTRmodel.predict(xtest) # Checking for accuracy score with actual data DTRscore=r2_score (ytest, DTRpred) DTRscore</pre>	90	0.9650934927089813
Rf model	<pre># Random Forest Regressor from sklearn.ensemble import RandomForestRegressor # Initializing the model RFmodel=RandomForestRegressor() # Train the data with Random Forest model RFmodel.fit(xtrain, ytrain) RFpred=RFmodel.predict(xtest) #Checking for accuracy score with actual data RFscore=r2_score (ytest, RFpred) RFscore</pre>	94	0.9212701843289313