

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testing is deprecated. Use the functions in the public API at pandas.testing instead.
import pandas.util.testing as tm
```

```
In [3]: df=pd.read_csv('/content/drive/My Drive/IBM Hack2020/T1.csv')
```

```
In [4]: df.drop(columns=df[['Date/Time','Theoretical_Power_Curve (KWh)']],inplace=True)
y=df['LV ActivePower (kW)']
df.drop(columns=['LV ActivePower (kW)'],axis=1,inplace=True)
```

```
In [5]: df['Wind Direction (°)']=(df['Wind Direction (°)']-df['Wind Direction (°)'].mean())/(df['Wind Direction (°)'].std())
df['Wind Speed (m/s)']=(df['Wind Speed (m/s)']-df['Wind Speed (m/s)'].mean())/(df['Wind Speed (m/s)'].std())
```

```
In [6]: y_train=y[:42283] #data till october
y_test=y[42283:]
X_train=df.iloc[:42283]
X_test=df.iloc[42283:]
```

```
In [8]: from sklearn.linear_model import LinearRegression
from sklearn.ensemble import VotingRegressor
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
import xgboost as xgb
from xgboost import XGBRegressor
from sklearn.metrics import accuracy_score,r2_score,mean_squared_error
xgr=XGBRegressor(base_score=0.5, booster='gbtree', colsample_bylevel=0.7,
                  colsample_bynode=1, colsample_bytree=0.3, gamma=0.2,
```

```

import xgboost as xgb
from xgboost import XGBRegressor
from sklearn.metrics import accuracy_score, r2_score, mean_squared_error
xgr=XGBRegressor(base_score=0.5, booster='gbtree', colsample_bylevel=0.7,
                  colsample_bynode=1, colsample_bytree=0.3, gamma=0.2,
                  importance_type='gain', learning_rate=0.03, max_delta_step=0,
                  max_depth=8, min_child_weight=25, missing=None, n_estimators=800,
                  n_jobs=1, nthread=None, objective='reg:linear', random_state=0,
                  reg_alpha=0.2, reg_lambda=0.8, scale_pos_weight=1, seed=None,
                  silent=None, subsample=0.1, verbosity=1)
sm=SVR(gamma='auto', C=50, epsilon=0.3)
rf=RandomForestRegressor(n_estimators=500, max_depth=4)
lr=LinearRegression()
model=VotingRegressor([('lr', lr), ('rf', rf), ('sm', sm), ('xgr', xgr)], weights=[1,1,2,3])
Model=model.fit(X_train, y_train)
y_pred=Model.predict(X_test)
print(y_pred)
print('R2', r2_score(y_test, y_pred))
print('RMSE', np.sqrt(mean_squared_error(y_test, y_pred)))

```

```

[16:54:48] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor of reg:squarederror.
[ 37.19686887  35.1144883  166.4837035 ... 1496.15758367 2002.28501619
 2197.22718458]
R2 0.8966747293978229
RMSE 441.4482250519333

```

```

In [ ]: from sklearn.externals import joblib
        joblib.dump(Model, 'ML_ensemble_Model.pkl')

```