AI : MEASURE ENERGY CONSUMPTION

DEVELOPMENT:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import xgboost as xgb

from sklearn.metrics import mean\_squared\_error

def create\_features(df):

df = df.copy()

df['hour'] = df.index.hour

df['dayofweek'] = df.index.dayofweek

df['quarter'] = df.index.quarter

df['month'] = df.index.month

df['year'] = df.index.year

df['dayofyear'] = df.index.dayofyear

df['dayofmonth'] = df.index.day

df['weekofyear'] = df.index.isocalendar().week

return df

def process(fileHead, fileName, date):

df = pd.read\_csv(fileName)

df = df.set\_index('Datetime')

df.index = pd.to\_datetime(df.index)

train = df.loc[df.index < date]

test = df.loc[df.index >= date]

df = create\_features(df)

train = create\_features(train)

test = create\_features(test)

features = ['dayofyear', 'hour', 'dayofweek', 'quarter', 'month', 'year']

target = fileHead

x\_train = train[features]

y\_train = train[target]

x\_test = test[features]

y\_test = test[target]

reg = xgb.XGBRegressor(base\_score=0.5, booster='gbtree', n\_estimators=1000, early\_stopping\_rounds=50, objective='reg:linear', max\_depth=3, learning\_rate=0.01)

reg.fit(x\_train, y\_train, eval\_set=[(x\_train, y\_train), (x\_test, y\_test)], verbose=100)

test['prediction'] = reg.predict(x\_test)

score = np.sqrt(mean\_squared\_error(test[fileHead], test['prediction']))

print(f'RMSE Score on Test set: {score:0.2f}')

test['error'] = np.abs(test[target] - test['prediction'])

test['date'] = test.index.date

test.groupby(['date'])['error'].mean().sort\_values(ascending=False).head(10)

if \_\_name\_\_ == '\_\_main\_\_':

file\_header = ['AEP\_MW','COMED\_MW','DAYTON\_MW','DEOK\_MW','DOM\_MW','DUQ\_MW','EKPC\_MW','FE\_MW','NI\_MW']

data = [['../IBM Project/Dataset/AEP\_hourly.csv','02-01-2017'],

['../IBM Project/Dataset/COMED\_hourly.csv','08-01-2017'],

['../IBM Project/Dataset/DAYTON\_hourly.csv','10-01-16'],

['../IBM Project/Dataset/DEOK\_hourly.csv','06-01-17'],

['../IBM Project/Dataset/DOM\_hourly.csv','01-01-17'],

['../IBM Project/Dataset/DUQ\_hourly.csv','06-01-16'],

['../IBM Project/Dataset/EKPC\_hourly.csv','08-01-17'],

['../IBM Project/Dataset/FE\_hourly.csv','02-01-17'],

['../IBM Project/Dataset/NI\_hourly.csv','07-01-09']]

print("1.AEP\n2.COMED\n3.DAYTON\n4.DEOK\n5.DOM\n6.DUQ\n7.EKPC\n8.FE\n9.NI\n")

n = int(input())

if(n in range(1,10)):

process(file\_header[n-1],data[n-1][0],data[n-1][1])

else:

print("Invalid Input!")