**IBM NAANMUDHALVAN**

**PHASE 5**

**DOMAIN – ELECTRICITY PRICE PREDICTION**

**PROBLEM STATEMENT:**

Create a predictive model that utilizes historical electricity prices and relevant factors to forecast future electricity prices, assisting energy providers and consumers in making informed decisions regarding consumption and investment

**PROBLEM DEFNITION:**

* The problem is to develop a predictive model that uses historical electricity prices and relevant factors to forecast future electricity prices.
* The objective is to create a tool that assists both energy providers and consumers in making informed decisions regarding consumption and investment by predicting future electricity prices.
* This project involves data preprocessing, feature engineering, model selection, training, and evaluation.

**DESIGN THINKING:**

* Data Source: Utilize a dataset containing historical electricity prices and relevant factors like date, demand, supply, weather conditions, and economic indicators.
* Data Preprocessing: Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations.
* Feature Engineering: Create additional features that could enhance the predictive power of the model, such as time-based features and lagged variables.
* Model Selection: Choose suitable time series forecasting algorithms (e.g., ARIMA, LSTM) for predicting future electricity prices.
* Model Training: Train the selected model using the preprocessed data.
* Evaluation: Evaluate the model's performance using appropriate time series forecasting metrics (e.g., Mean Absolute Error, Root Mean Squared Error).

**PROBLEM STATEMENT:**

Electricity is a basic human need and definitely one of the most important factors of societal progress. In recent decades however, electricity has entered the market as a tradeable commodity and the power industry of many countries has been **deregulated**. In Spain, the Electric Power Act 54/1997 exposed all of the stakeholders to **high amounts of uncertainty** as the price of electricity is determined by countless factors and also, due to the fact that electricity cannot be stored in large quantities efficiently . With the emergence of this new market, the need for reliable forecasting methods at all scales (hourly, daily, long-term, etc.) has also emerged and has become a large area of research.

**SOLUTION:**

**DATA COLLECTION:**

The dataset that we choose to analyze the electricity price prediction. Utilize a dataset containing historical electricity prices and relevant factors like date, demand, supply, weather conditions, and economic indicators.

**DATA PREPROCESSING:**

Clean and preprocess the data by handle missing values, and convert categorical features into numerical representations **.**Also remove punctuation marks, HTML tags, URL’s, successive whitespaces, convert the text to lower case, strip whitespaces from the beginning and the end of the reviews.

**EXPLORATORY DATA ANALYSIS:**

Explore the data to understand its characteristics, observe your dataset, find any missing values, categorize your values , find the shape of dataset ,find relationships in dataset and locate any outliers in dataset.

**DATA VISUALIZATION:**

Visualizations consisting of histograms, box plots, scatter plots, line plots, heat maps, and bar charts assist in identifying styles, trends, and relationships within the facts to present key findings and insights.

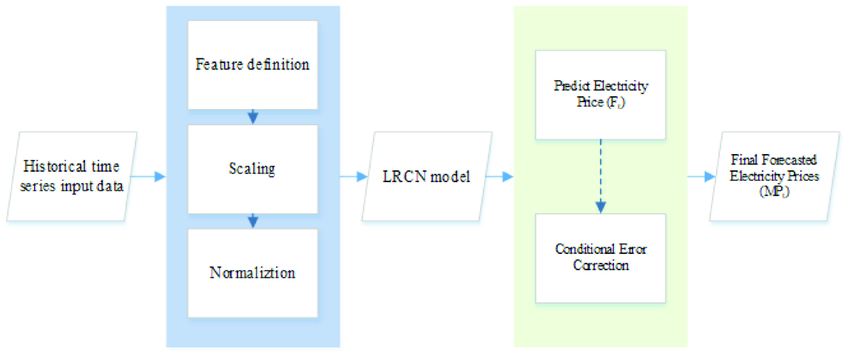
**FEATURE ENGINEERING:**

Create additional features that could enhance the predictive power of the model, such as time-based features, data related features, rolling window feature, expanding window feature, domain specific features and lagged variables.

Data Science is not a field where theoretical understanding helps you to start a carrier. It totally depends on the projects you do and the practice you have done that determines your probability of success. Feature engineering is a very important aspect of machine learning and data science and should never be ignored. The main goal of Feature engineering is to get the best results from the algorithms.

.

Feature engineering refers to the process of using domain knowledge to select and transform the most relevant variables from raw data when creating a predictive model using machine learning or statistical modelling . The goal of feature engineering and selection is to improve the performance of machine learning (ML) algorithms.

**SOLUTION OVERVIEW:**

**DATASET:**

Electricity price prediction

**ABOUT:**

* Loading a dataset
* Preprocessing dataset
* Data cleaning
* Data transformation
* Data reduction

**PROGRAM:**

**LOAD THE DATASET:**

import pandas as pd

a=pd.read\_csv('Electricity.csv')

a

38014 rows × 18 columns

a.head(5)

a.tail(5)

a.isnull()

a.shape

(38014, 18)

a.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 38014 entries, 0 to 38013

Data columns (total 18 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 DateTime 38014 non-null object

1 Holiday 38014 non-null object

2 HolidayFlag 38014 non-null int64

3 DayOfWeek 38014 non-null int64

4 WeekOfYear 38014 non-null int64

5 Day 38014 non-null int64

6 Month 38014 non-null int64

7 Year 38014 non-null int64

8 PeriodOfDay 38014 non-null int64

9 ForecastWindProduction 38014 non-null object

10 SystemLoadEA 38014 non-null object

11 SMPEA 38014 non-null object

12 ORKTemperature 38014 non-null object

13 ORKWindspeed 38014 non-null object

14 CO2Intensity 38014 non-null object

15 ActualWindProduction 38014 non-null object

16 SystemLoadEP2 38014 non-null object

17 SMPEP2 38014 non-null object

dtypes: int64(7), object(11)

memory usage: 5.2+ MB

a.nunique()

DateTime 38014

Holiday 15

HolidayFlag 2

DayOfWeek 7

WeekOfYear 52

Day 31

Month 12

Year 3

PeriodOfDay 48

ForecastWindProduction 29312

SystemLoadEA 36166

SMPEA 8661

ORKTemperature 32

ORKWindspeed 53

CO2Intensity 25115

ActualWindProduction 2940

SystemLoadEP2 36171

SMPEP2 9277

dtype: int64

a.isnull().sum()

DateTime 0

Holiday 0

HolidayFlag 0

DayOfWeek 0

WeekOfYear 0

Day 0

Month 0

Year 0

PeriodOfDay 0

ForecastWindProduction 0

SystemLoadEA 0

SMPEA 0

ORKTemperature 0

ORKWindspeed 0

CO2Intensity 0

ActualWindProduction 0

SystemLoadEP2 0

SMPEP2 0

dtype: int64

a.duplicated().any()

False

**SPLITTING DATA:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x=a[['SMPEA','CO2Intensity']]  y=a['Year']  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)  x\_train   | **SMPEA** | **CO2Intensity** | | --- | --- | | **35021** | 58.58 | 381.83 | | **13975** | 43.98 | 479.47 | | **2601** | 25.53 | 379.63 | | **12345** | 55.83 | 646.40 | | **15306** | 127.88 | 478.23 | | **...** | ... | ... | | **824** | 32.99 | 392.69 | | **5987** | 44.48 | 529.00 | | **16996** | 37.47 | 653.11 | | **10444** | 70.84 | 459.40 | | **2994** | 39.28 | 423.49 |   30411 rows × 2 columns |

|  |
| --- |
| y\_train |

35021 2013

13975 2012

2601 2011

12345 2012

15306 2012

...

824 2011

5987 2012

16996 2012

10444 2012

2994 2012

Name: Year, Length: 30411, dtype: int64

|  |
| --- |
| y\_test |

5395 2012

28895 2013

24962 2013

17195 2012

27915 2013

...

30087 2013

5042 2012

35100 2013

21138 2013

34691 2013

Name: Year, Length: 7603, dtype: int64

|  |
| --- |
| x=a.iloc[:,:-1].values  y=a.iloc[:,1].values  print(x,y) |

|  |
| --- |
| [['01/11/2011 00:00' 'None' 0 ... '600.71' '356.00' '3159.60']  ['01/11/2011 00:30' 'None' 0 ... '605.42' '317.00' '2973.01']  ['01/11/2011 01:00' 'None' 0 ... '589.97' '311.00' '2834.00']  ...  ['31/12/2013 22:30' "New Year's Eve" 1 ... 280.91 962.0 3460.29]  ['31/12/2013 23:00' "New Year's Eve" 1 ... 302.46 950.0 3563.99]  ['31/12/2013 23:30' "New Year's Eve" 1 ... 308.01 1020.0 3517.08]] ['None' 'None' 'None' ... "New Year's Eve" "New Year's Eve"  "New Year's Eve"] |

**MODEL EVALUATION:**

**LINEAR REGRESSION:**

|  |
| --- |
| **fromsklearn.linear\_modelimport**LinearRegression,Ridge,Lasso  *# Linear Regression*  linear\_model=LinearRegression()  linear\_mse,linear\_rmse,linear\_mae,linear\_r2=perform\_cross\_validation(linear\_model,X,y,num\_folds)  print("Linear Regression:")  print(f"Average MSE: {np.mean(linear\_mse) / np.mean(y) \* 100:.2f}%")  print(f"Average RMSE: {np.mean(linear\_rmse) / np.mean(y) \* 100:.2f}%")  print(f"Average MAE: {np.mean(linear\_mae) / np.mean(y) \* 100:.2f}%")  print(f"Average R-squared: {np.mean(linear\_r2) \* 100:.2f}%")  print("**\n**") |

Linear Regression:

Average MSE: 18.89%

Average RMSE: 11.01%

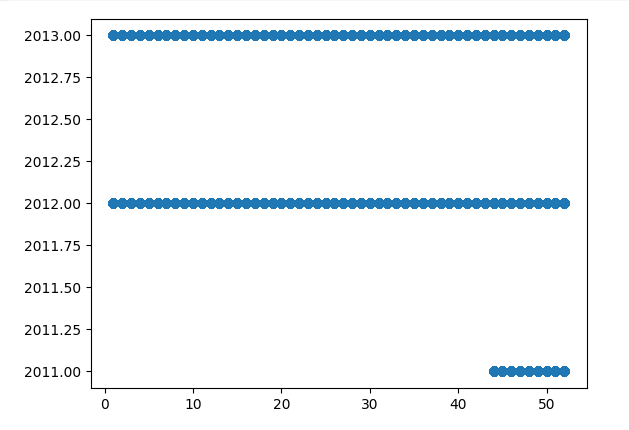
Average MAE: 8.38%

Average R-squared: 89.54%

**DATA VISUALIZATION:**

**SCATTER PLOT:**

|  |
| --- |
| import numpy  import matplotlib.pyplot as plt  numpy.random.seed(2)  x=a['WeekOfYear']  y=a['Year']  plt.scatter(x, y)  plt.show() |



**PIE CHART:**

df=a['Year'].head(10)

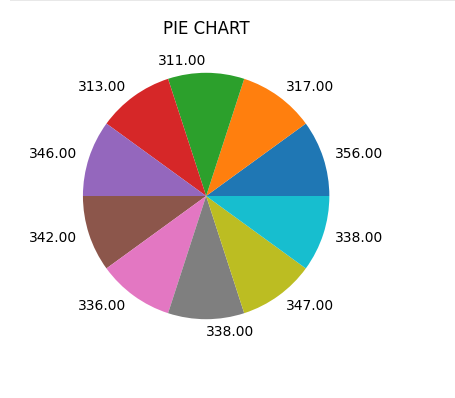
df1=a['ActualWindProduction'].head(10)

fig = plt.figure(figsize =(4,4))

plt.pie(df, labels= df1)

plt.title("PIE CHART")

plt.show()



**HEAT MAP:**

import numpy as np

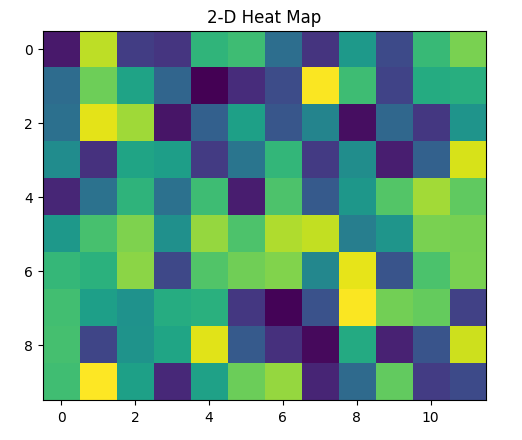
import matplotlib.pyplot as plt

a = np.random.random(( 10, 12 ))

plt.imshow( a )

plt.title( "2-D Heat Map" )

plt.show()



**HISTOGRAM:**

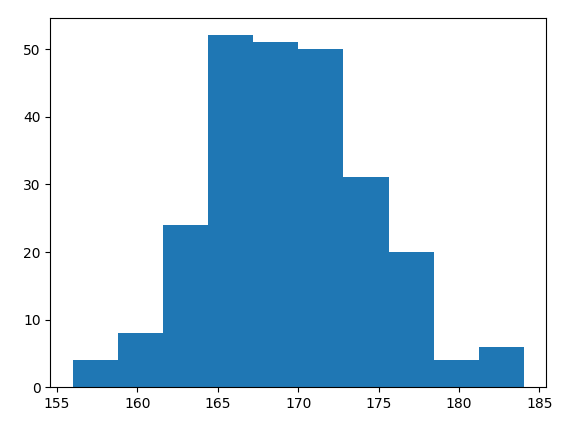
import matplotlib.pyplot as plt

import numpy as np

x = np.random.normal(170, 5, 250)

plt.hist(x)

plt.show()



**CONCLUSION:**

Predicting the price of electricity helps a lot of companies to understand how much electricity expenses they have to pay every year. By using the Electricity dataset we clearly outline the problem statement , design thinking process, data preprocessing steps, feature extraction techniques ,model training, machine learning algorithms ,data visualization techniques and evaluation metrics.

**NAME :KHAWSIK R**

**COLLEGE CODE : 4204**

**REGISTER NO : 420421106025**