**PRODUCT DEMAND PREDICTION WITH MACHINE LEARNINGS**

**Phase 5: Project Documentation & Submission**

**PROBLEM DEEFINITION :**

The problem is to create a machine learning model that forecasts product demand based on historical sales data and external factors. The goal is to help business optimize inventory management and production planning to efficiently meet customer needs. This project involves data collection, data preprocessing, feature engineering, model selection, training, and evaluation.

**DESIGN THINKING:**

**DATA COLLECTION:**

Data collection is the process of gathering and collecting information or data from various sources or instruments for the purpose of analysis, research, decision-making, or record-keeping. It is a fundamental step in many fields, including science, business, healthcare, social sciences, and more. Here are some key aspects of data collection:

**1.Purpose**: Data collection is typically conducted with a specific purpose or objective in mind. This could include conducting research, monitoring a process, making informed decisions, or solving a problem. The purpose guides what data needs to be collected and how it should be collected.

**2.Data Sources**: Data can be collected from various sources, including:

**Surveys and Questionnaires**: Asking individuals or groups of people specific questions to gather information. This method is commonly used in social sciences and market research.

**Observation**: Directly observing and recording events, behaviors, or phenomena. This is often used in fields like ecology, psychology, and ethnography.

**Sensor Data**: Collecting data from sensors, such as temperature sensors, GPS devices, or IoT devices. This is prevalent in fields like environmental monitoring and industrial automation.

**Web Scraping**: Extracting data from websites and online sources, often used for data analytics and business intelligence.

**Existing Databases:** Accessing and extracting data from pre-existing databases or records, such as customer databases, financial records, and government databases.

**3.Data Collection Methods**: Depending on the data source and the nature of the data, different methods may be employed. These methods can be quantitative (numeric data) or qualitative (non-numeric data). Common methods include structured interviews, surveys, experiments, focus groups, and more.

**4.Sampling**: In cases where it's impractical or too costly to collect data from an entire population, researchers often use sampling techniques to collect data from a representative subset of the population. This helps generalize findings to the entire population.

**5.Data Quality**: Ensuring the quality and accuracy of collected data is crucial. Data should be collected in a consistent and unbiased manner. Data cleaning and validation processes are often applied to remove errors and anomalies.

**6.Ethical Considerations**: Data collection must adhere to ethical principles, including informed consent when dealing with human subjects, data privacy, and protection of sensitive information.

**7.Data Recording:** Collected data is typically recorded and stored in a structured format, such as spreadsheets, databases, or data warehouses. Proper documentation is important to keep track of the data's origin and context.

**8.Data Analysis:** After data collection, the next step is often data analysis, where statistical, computational, or analytical techniques are applied to derive insights, make predictions, or draw conclusions from the collected data.

Data collection is a fundamental step in the data lifecycle, and the quality and reliability of collected data can have a significant impact on the outcomes of research, decision-making, and problem-solving processes. It is essential to plan and execute data collection carefully to ensure that the collected data is fit for its intended purpose.

**Dataset Link:**https://raw.githubusercontent.com/amankharwal/Website-data/master/demand.csv

Development phase 1:

CONTEXT

Walmart is a renowned retail corporation that operates a chain of hypermarkets. Here, Walmart has provided a data c Business Objectives

combining of 45 stores including store information and monthly sales. The data is provided on weekly basis. Walmart tries to find the impact of holidays on the sales of store. For which it has included four holidays’ weeks into the dataset which are Christmas, Thanksgiving, Super bowl, Labor Day. Here we are owing to Analyze the dataset given. Before doing that, let me point out the objective of this analysis.

**Business Objectives**

Our Main Objective is to predict sales of store in a week. As in dataset size and time related data are given as feature, so analyze if sales are impacted by time-based factors and space- based factor. Most importantly how inclusion of holidays in a week soars the sales in store

# Importing Necessary Libraries and Data

In[1]

import matplotlib as mpl

import seaborn as sns import numpy as np *# To use np.arrays*

import pandas as pd *# To use dataframes*

from pandas.plotting import autocorrelation\_plot as auto\_corr

*# To plot*

import matplotlib.pyplot as plt

%matplotlib inline

*#For date-time*

import math

from datetime import datetime

from datetime import timedelta

*# Another imports if needs*

import itertools

import statsmodels.api as sm

import statsmodels.tsa.api as smt

import statsmodels.formula.api as smf

from sklearn.model\_selection import train\_test\_split

from statsmodels.tsa.seasonal import seasonal\_decompose as season

from sklearn.metrics import mean\_squared\_error, mean\_absolute\_error

from sklearn.metrics import accuracy\_score, balanced\_accuracy\_score

from sklearn.model\_selection import cross\_val\_score

from sklearn.pipeline import make\_pipeline, Pipeline

from sklearn.ensemble import RandomForestRegressor

from sklearn import metrics

from sklearn.linear\_model import LinearRegression

from sklearn import preprocessing

from statsmodels.tsa.holtwinters import ExponentialSmoothing

from statsmodels.tsa.stattools import adfuller, acf, pacf

from statsmodels.tsa.arima\_model import ARIMA

!pip install pmdarima

from pmdarima.utils import decomposed\_plot

from pmdarima.arima import decompos

from pmdarima import auto\_arima

import warnings

warnings.filterwarnings("ignore")

In[2]

pd.options.display.max\_columns=100

In[3]

df\_store = pd.read\_csv('../input/walmart-sales-forecast/stores.csv')

In[4]

df\_train = pd.read\_csv('../input/walmart-sales-forecast/train.csv')

In[5]

df\_features = pd.read\_csv('../input/walmart-sales-forecast/features.csv')

In]6]

df\_store.head()

Out[6]:

|  | Store | Type | Size |
| --- | --- | --- | --- |
| 0 | 1 | A | 151315 |
| 1 | 2 | A | 202307 |
| 2 | 3 | B | 37392 |
| 3 | 4 | A | 205863 |
| 4 | 5 | B | 34875 |

In[7]

df\_train.head()

Out[7]:

|  | Store | Dept | Date | Weekly\_Sales | IsHoliday |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 1 | 2010-02-05 | 24924.50 | False |

# **Date**

In[15]

df['Date'].head(5).append(df['Date'].tail(5))

Out[20]:

0 2010-02-05

1 2010-02-05

2 2010-02-05

3 2010-02-05

4 2010-02-05

421565 2012-10-26

421566 2012-10-26

421567 2012-10-26

421568 2012-10-26

421569 2012-10-26

Name: Date, dtype: object

# **Cleaning Process**

* The data has no too much missing values. All columns was checked.
* I choose rows which has higher than 0 weekly sales. Minus values are 0.3% of data. So, I dropped them.
* Null values in markdowns changed to zero. Because, they were written as null if there were no markdown on this department.

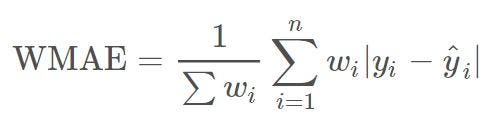
# **Explorations & Findings**

* There are 45 stores and 81 department in data. Departments are not same in all stores.
* Although department 72 has higher weekly sales values, on average department 92 is the best. It shows us, some departments has higher values as seasonal like Thanksgiving. It is consistant when we look at the top 5 sales in data, all of them belongs to 72th department at Thanksgiving holiday time.
* Although stores 10 and 35 have higher weekly sales values sometimes, in general average store 20 and store 4 are on the first and second rank. It means that some areas has higher seasonal sales.
* Stores has 3 types as A, B and C according to their sizes. Almost half of the stores are bigger than 150000 and categorized as A. According to type, sales of the stores are changing.
* As expected, holiday average sales are higher than normal dates.
* Christmas holiday introduces as the last days of the year. But people generally shop at 51th week. So, when we look at the total sales of holidays, Thankgiving has higher sales between them which was assigned by Walmart.
* Year 2010 has higher sales than 2011 and 2012. But, November and December sales are not in the data for 2012. Even without highest sale months, 2012 is not significantly less than 2010, so after adding last two months, it can be first.
* It is obviously seen that week 51 and 47 have higher values and 50-48 weeks follow them. Interestingly, 5th top sales belongs to 22th week of the year. This results show that Christmas, Thankgiving and Black Friday are very important than other weeks for sales and 5th important time is 22th week of the year and it is end of the May, when schools are closed. Most probably, people are preparing for holiday at the end of the May.
* January sales are significantly less than other months. This is the result of November and December high sales. After two high sales month, people prefer to pay less on January.
* CPI, temperature, unemployment rate and fuel price have no pattern on weekly sales.

# **First Trial with Random Forest**

Generally, Rondom Forest Regressor gives good results when we tune it well. So, to find simple baseline model, I will use RandomForestRegressor in this notebook. Also, feature importance for model can be found in this notebook.

Our metric for this project is weighted mean absolute error (WMAE):



where

* n is the number of rows
* ŷ i is the predicted sales
* yi is the actual sales
* wi are weights. w = 5 if the week is a holiday week, 1 otherwise

linkcode

With this metric, the error at holiday weeks has 5 times weight more than normal weeks. So, it is more important to predict sales at holiday weeks accurately. All results for trails can be found at the end of this notebook.

Development phase 2:

**FEATURE ENGINEERING:**

Feature engineering is the process of creating new features or transforming existing features in a dataset to improve the performance of machine learning models. It involves selecting, enhancing, or modifying the input variables (features) used in a model to make them more informative or suitable for the task at hand. Effective feature engineering can lead to better model accuracy and generalization. Here are some key aspects of feature engineering:

**1.Feature Selection**:

Sometimes, datasets contain many features, some of which may be irrelevant or redundant. Feature selection involves choosing the most relevant features while discarding less important ones. Common techniques for feature selection include statistical tests, correlation analysis, and domain knowledge.

**2.Feature Transformation:**

Feature transformation involves changing the representation of features to make them more suitable for a given machine learning algorithm or problem. Common transformations include.

**MODEL TRAINING:**

Model training is a crucial step in the development of machine learning models. It involves using a dataset to teach a model to make predictions or classify data based on patterns and relationships it learns from the data. The primary goal of model training is to optimize the model's parameters or weights to minimize its prediction errors and improve its performance. Here are the key steps and concepts involved in model training.

**1.Data Preparation**:

Before training a model, you need to prepare the data. This includes data preprocessing steps such as cleaning, transformation, and feature engineering. The dataset is typically split into three parts: a training set, a validation set, and a test set.

**2.Selecting a Model:**

You need to choose an appropriate machine learning algorithm or model architecture based on the nature of your problem (classification, regression, clustering, etc.) and the characteristics of your data. Different algorithms have different strengths and weaknesses.

**3.Initialization:**

Initialize the model's parameters or weights with some initial values. The choice of initialization can affect the convergence of the training process.

**4.Loss Function:**

Define a loss function (also known as a cost function or objective function) that quantifies how well the model's predictions match the actual target values in the training data. Common loss functions include mean squared error for regression problems and cross-entropy for classification problems.

**5.Optimization Algorithm:**

Choose an optimization algorithm, such as gradient descent, stochastic gradient descent (SGD), or variants like Adam or RMSprop. The optimization algorithm iteratively updates the model's parameters to minimize the loss function.

**6.Training Loop:**

The model is trained through an iterative process in which it makes predictions on the training data, calculates the loss, and updates its parameters using the optimization algorithm. This process is repeated for a specified number of iterations (epochs) or until convergence.

**7.Validation**:

After each training epoch or a certain number of iterations, the model's performance is evaluated on the validation set using appropriate evaluation metrics. This helps monitor the model's progress and detect overfitting.

**8.Hyperparameter Tuning:** Fine-tune the hyperparameters of the model, such as learning rate, batch size After each training epoch or a certain number of iterations, the model's performance is evaluated on the validation set using appropriate evaluation metrics. This helps monitor the model's progress and detect overfitting.

, and regularization strength, to optimize its performance on the validation set. This is typically done through grid search, random search, or more advanced techniques like Bayesian optimization.

**9.Early Stopping:**

Implement early stopping to prevent overfitting. If the model's performance on the validation set starts to degrade, training can be stopped to avoid training the model too long and fitting noise in the data.

**10.Test Evaluation:**

Once training is complete, the final model is evaluated on the test set to assess its generalization performance. This step provides an estimate of how well the model will perform on new, unseen data.

**11.Model Deployment:**

If the model meets the desired performance criteria, it can be deployed for real-world use. Deployment involves integrating the model into an application or system where it can make predictions on new data.

**12.Monitoring and Maintenance:**

After deployment, it's essential to monitor the model's performance in production and periodically retrain it with fresh data to ensure it remains accurate and up-to-date.

Model training is an iterative and resource-intensive process that requires careful attention to data quality, algorithm selection, and hyperparameter tuning. It is a fundamental step in creating machine learning solutions that can make accurate predictions or decisions based on data.

**PYTHON PROGRAM:**

import pandas as pd

import numpy as np

import plotly.express as px

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.tree import DecisionTreeRegressor

file = pd.read\_csv("https://raw.githubusercontent.com/amankharwal/Website-data/master/demand.csv")

data = file

data.head()

#

| ID | Store ID | Total Price | Base Price | Units Sold |
| --- | --- | --- | --- | --- |
| 0 | 1 | 8091 | 99.0375 | 111.8625 | 20 |
| 1 | 2 | 8091 | 99.0375 | 99.0375 | 28 |
| 2 | 3 | 8091 | 133.9500 | 133.9500 | 19 |
| 3 | 4 | 8091 | 133.9500 | 133.9500 | 44 |
| 4 | 5 | 8091 | 141.0750 | 141.0750 | 52 |

data.isnull()

data.dropna()

#

data.isna().sum()

# ID 0

Store ID 0

Total Price 1

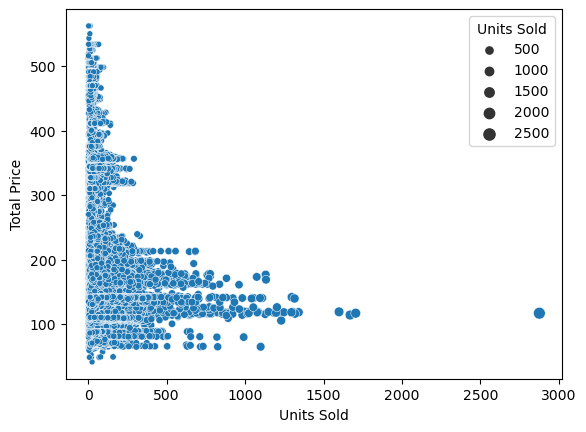
Base Price 0

Units Sold 0

dtype: int64

sns.scatterplot(data, x='Units Sold', y='Total Price',size= 'Units Sold')

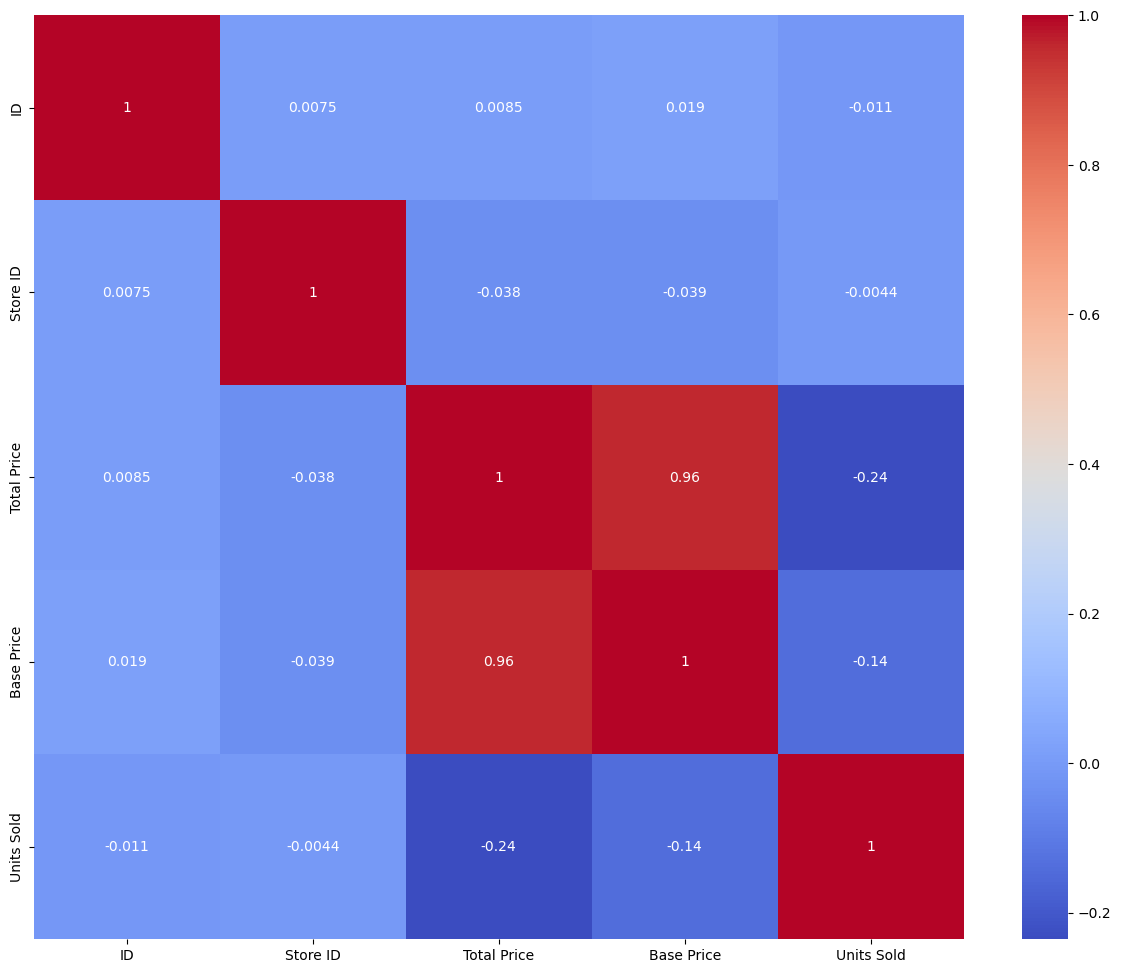
# <Axes: xlabel='Units Sold', ylabel='Total Price'>

Out put:

data.corr()

#

| ID | Store ID | Total Price | Base Price | Units Sold |
| --- | --- | --- | --- | --- |
| ID | 1.000000 | 0.007464 | 0.008473 | 0.018932 | -0.010616 |



**EVALUATION:**

Evaluation in the context of machine learning and data analysis refers to the process of assessing the performance and quality of a model or a system based on specific criteria and metrics. It is a critical step to determine how well a model or system is performing and whether it meets the intended goals. Here are key aspects of evaluation.