Product demand prediction with machine learning

Name:Swathi.T

Reg.no:912421104048

Phase 2 submission document

INTRODUCTION:

A product company plans to offer discounts on its product during the upcoming holiday season. The company wants to find the price at which its product can be a better deal compared to its competitors. For this task, the company provided a dataset of past changes in sales based on price changes. You need to train a model that can predict the demand for the product in the market with different price segments.

Data source:

DatasetLink:
https://www.kaggle.com/datasets/chakradharmattapalli/product-demand-prediction-with-machine-learning

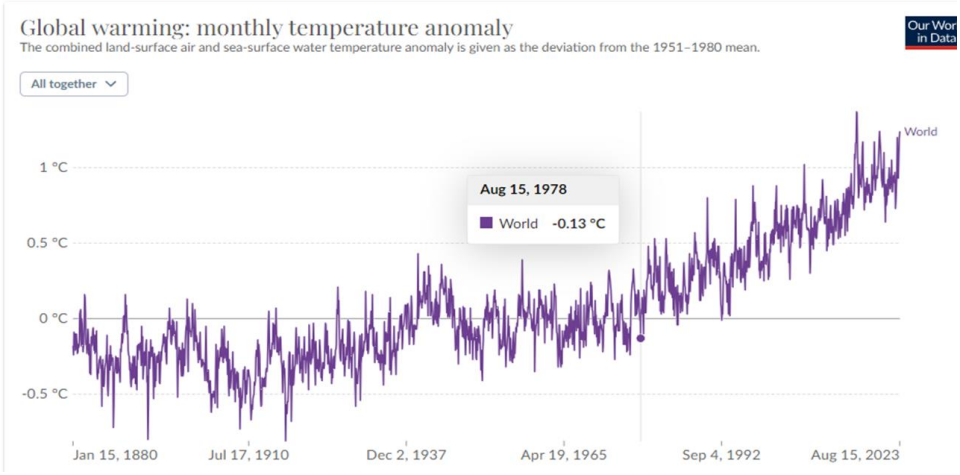
Demand forecasting:

> Demand forecasting is the process of predicting what customers' appetite will be for existing products or services, determining what adjustment you should make and what new offerings will spark interest. But predicting what people will want, in what quantities and when is no small feat.

Design thinking:

- Data collection
- Data preparation
- Feature engineering
- Model selection
- Model training
- Evaluation

Data about climate change



Data preprocessing:

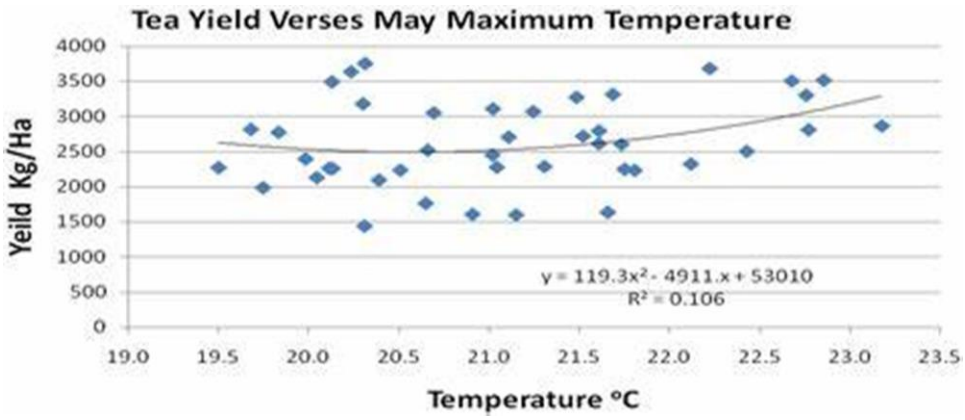
Data preprocessing can refer to manipulation or dropping of data before it is used in order to ensure or enhance performance,[1] and is an important step in the data mining process. The phrase "garbage in, garbage out" is particularly applicable to data mining and machine learning projects. Data collection methods are often loosely controlled, resulting in out-of-range values,

impossible data combinations, and missing values, amongst other issues.

Methods:

- Clean and preprocess the data
- Handle missing values
- Convert categorical features into numerical representation

Linear regression:



ARIMA

ARIMA is a method for forecasting or predicting future outcomes based on a historical time series.

Data wrangling technique:

- Merging several data sources into one data-set for analysis.
- Identifying gaps or empty cells in data and either filling or removing them.
- Deleting irrelevant or unnecessary data.
- Identifying severe outliers in data and either explaining the inconsistencies or deleting them to facilitate analysis.

EXAMPLE PROGRAM:

```
import pandas as pd

import numpy as np

import plotly.express as px

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
```

from sklearn.tree import DecisionTreeRegressor

data = pd.read_csv("https://raw.githubusercontent.com/amankharwal/Website-data/master/demand.csv")

data.head()

OUTPUT:

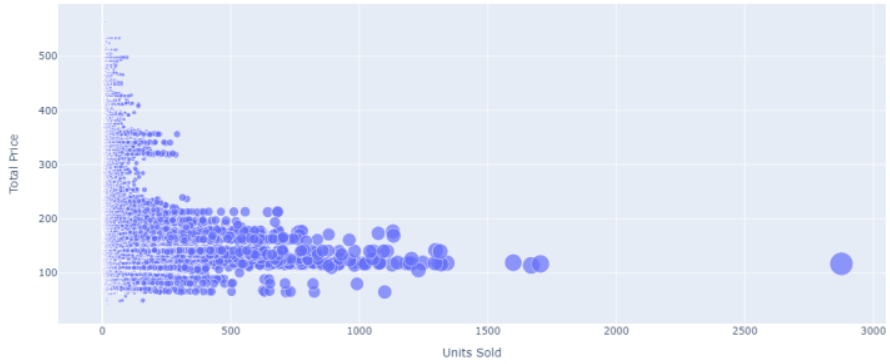
	ID	Store ID	Total Price	Base Price	Units Sold
0	1	8091	99.0375	111.8625	20
1	2	8091	99.0375	99.0375	28
2	3	8091	133.9500	133.9500	19
3	4	8091	133.9500	133.9500	44
4	5	8091	141.0750	141.0750	52

fig = px.scatter(data, x="Units Sold", y="Total Price",

size='Units Sold')

fig.show()

OUTPUT:



features = np.array([[133.00, 140.00]])

model.predict(features)

OUTPUT:

array([27.])

Prophet:

make an in-sample forecast

from pandas import read_csv

from pandas import to_datetime

from pandas import DataFrame

from fbprophet import Prophet

from matplotlib import pyplot

load data

```
path = 'https://raw.githubusercontent.com/jbrownlee/Datasets/master/monthly-car-sales.csv'
df = read_csv(path, header=0)
# prepare expected column names

df.columns = ['ds', 'y']
df['ds']= to_datetime(df['ds'])
# define the model

model = Prophet()
# fit the model

model.fit(df)
# define the period for which we want a prediction

future = list()
for i in range(1, 13):
date = '1968-%02d' % i
future.append([date])
future = DataFrame(future)
future.columns = ['ds']
```

```
future['ds']= to_datetime(future['ds'])
# use the model to make a forecast

forecast = model.predict(future)
# summarize the forecast

print(forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].head())
# plot forecast

model.plot(forecast)
pyplot.show()
```

Output:

	ds	yhat	yhat_lower	yhat_upper
0	1968-01-01	14364.866157	12816.266184	15956.555409
1	1968-02-01	14940.687225	13299.473640	16463.811658
2	1968-03-01	20858.282598	19439.403787	22345.747821
3	1968-04-01	22893.610396	21417.399440	24454.642588
4	1968-05-01	24212.079727	22667.146433	25816.191457

What is Mean Squared Error or MSE

•The Mean Absolute Error is the squared mean of the difference between the actual values and predictable values.

Conclusion:

*** In this phase-2 project conclusion, we will summarize the key findings and insights from the incorporating time series techniques .we will reiterate the impact of these time series techniques .These techniques provide valuable insights into patterns, seasonality, and trends within the data, enabling more accurate predictions and forecasts.**