Walmart

1 INTRODUCTION

1.1 Overview

This report discusses our key messages and findings from the analysis we have done for Walmart. Given that they are one of the biggest retailers in the world, Walmart must have precise sales projections for all of its divisions. Every department's sales can be influenced by a variety of factors, so it's critical to pinpoint the major ones that drive sales and utilise them to build a model that can help predict sales with some degree of accuracy.

We used a dataset from the "Walmart Store Sales Forecasting" study that was made accessible on Kaggle for this undertaking. This dataset contains weekly sales information over a three-year period for 45 establishments and 99 departments. Additionally, we had data particular to a location and a store, such as the size of the store, the unemployment rate, the weather, and any current promotions. We sought to create a regression model that can anticipate sales while also being computationally effective and scalable using these criteria. The main hurdles we encountered in our analysis were caused by the big dataset, which presented various computational difficulties and forced us to change how we approached the issue. We also encountered substantial difficulties in determining the appropriate factors on which the analysis could be conducted.

1.2 Purpose

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. The major goal of this study is to forecast Walmart's sales using historical data and determine whether variables like weather, unemployment, gasoline prices, etc. impact the stores under study's weekly sales. Additionally, this study attempts to determine whether sales are generally greater on special occasions like Christmas and Thanksgiving than on non-holiday days so that retailers can concentrate on developing marketing initiatives that boost sales and boost profits.

2 LITERATURE SURVEY

2.1 Existing problem

Walmart runs several promotional markdown sales throughout the year on days immediately following the prominent holidays in the United States; it becomes crucial for the organization to determine the impact of these promotional offerings on weekly sales to drive resources towards such key strategic initiatives.

2.2 Related Work

Studies have previously been performed to predict sales for retail industry corporations based on the availability of relevant historic data. Several authors from the Fiji National University and The University of the South Pacific analyzed the Walmart dataset to predict sales ("Walmart's Sales Data Analysis - A Big Data Analytics Perspective," 2017). Tools like Hadoop Distributed File Systems (HDFS), Hadoop MapReduce framework, and Apache Spark along with Scala, Java, and Python high-level programming environments were used to analyze and visualize the data. Their study also aimed at understanding whether the factors included in the dataset have any impact on the sales of Walmart.

In 2015, Harsoor and Patil (Harsoor & Patil, 2015) worked on forecasting Sales of Walmart Stores using big data applications: Hadoop, MapReduce, and Hive so that resources are managed efficiently. This paper used the same sales data set that has been used for analysis in this study, however, they forecasted the sales for the upcoming 39 weeks using Holt's winter algorithm. The forecasted sales are visually represented in Tableau using bubble charts.

Michael Crown (Crown, 2016), a data scientist, performed analysis on a similar dataset but instead focused on the usage of time series forecasting and non-seasonal ARIMA models to make his predictions. He worked on ARIMA modeling to create one year of weekly forecasts using 2.75 years of sales data, with features of the store, department, date, weekly sales, and holiday data. Performance was measured using normalized root-mean-square error (NRMSE). Forecasting has not been limited to just business enhancement. Several researchers have tried to utilize machine learning and statistical analysis to build predictive models that can accurately predict the weather, monitor stock prices and analyze market trends, predict illnesses in a patient, etc. Likewise, in 2017, Chouskey and Chauhan (Chouksey & Chauhan, 2017) worked on creating a weather forecasting model that accurately predicts the weather and sends out weather warnings for people and businesses so that they can better prepare for the unforeseeable weather. The authors make use of MapReduce and Spark to create their models and gather data from various weather sensors; weather forecasts can be essentially important as they influence all human aspects and the authors have made use of various parameters like temperature, humidity, pressure, wind speed, etc. to make better predictions.

Another approach followed by Rajat Panchotia (Panchotia, 2020) to create a predictive model using linear regression throws light on the various regression techniques and the metrics that need to be defined when creating such models. He talks about the importance of defining techniques that should be considered, like studying the number of independent variables and type of dependent variables, determining the best fit, etc., based on the nature of data and the most accurate regression model that should be selected based on results obtained. In his article, he also emphasizes on the use of regression coefficients, p-values, variable selection, and residual analysis to study the performance of regression models. While Panchotia only focuses on studying the direct relationship between the independent and dependent variables of the dataset, another theory by James Jaccard and Robert Turrisi (Jaccard & Turrisi, 2018) involves observing the change in the relationship between an independent and dependent variable as a result of the presence of a third variable, called the moderator variable.

Kassambara (kassambara, 2018), in his article, throws light on the implementation of interaction effects with a multiple linear regression in R. Taking a basic multiple regression model as a base where he tries to predict sales based on advertising budgets spent on youtube and facebook, he tries to create an additive model based on two relevant predictors (budget for youtube and budget for facebook). His model assumes that the effect on sales of youtube advertising is independent of the effect of facebook advertising and subsequently creates a regression model. With an R2 score of 0.98, he observes that there is an interactive relationship between the two predictor variables (youtube and facebook advertising) and this additive model performs better than the regular regression model.

A further extension of predictive techniques relevant to this study involves the implementation of random forest algorithms to create predictive models. A study conducted by researchers at the San Diego State University (Lingjun et al., 2018) highlights the importance of this tree-based machine learning algorithm over other regression methods to create predictive models for the higher education sector. With their study, the authors use a standard classification and regression tree (CART) algorithm along with feature importance to highlight the importance of using random forest algorithms with prediction problems in Weka and R and compare their efficacy with several other models like lasso regression and logistic regression.

2.3 Proposed solution

Our proposed solution 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.

A Machine Learning model like, a regression model can provide robust prediction given the dataset satisfies its linearity assumptions. Furthermore, machine learning forecasting is not a black box; the influence of model inputs can be weighed and understood so that the forecast is intuitive and transparent. Machine Learning models can also be updated and become adaptable to the changes in dataset. And also, through machine learning help, relation between markdown events and weekly sales can be utilize in correct manner using machine learning model.

3 THEORITICAL ANALYSIS

3.1 Block diagram

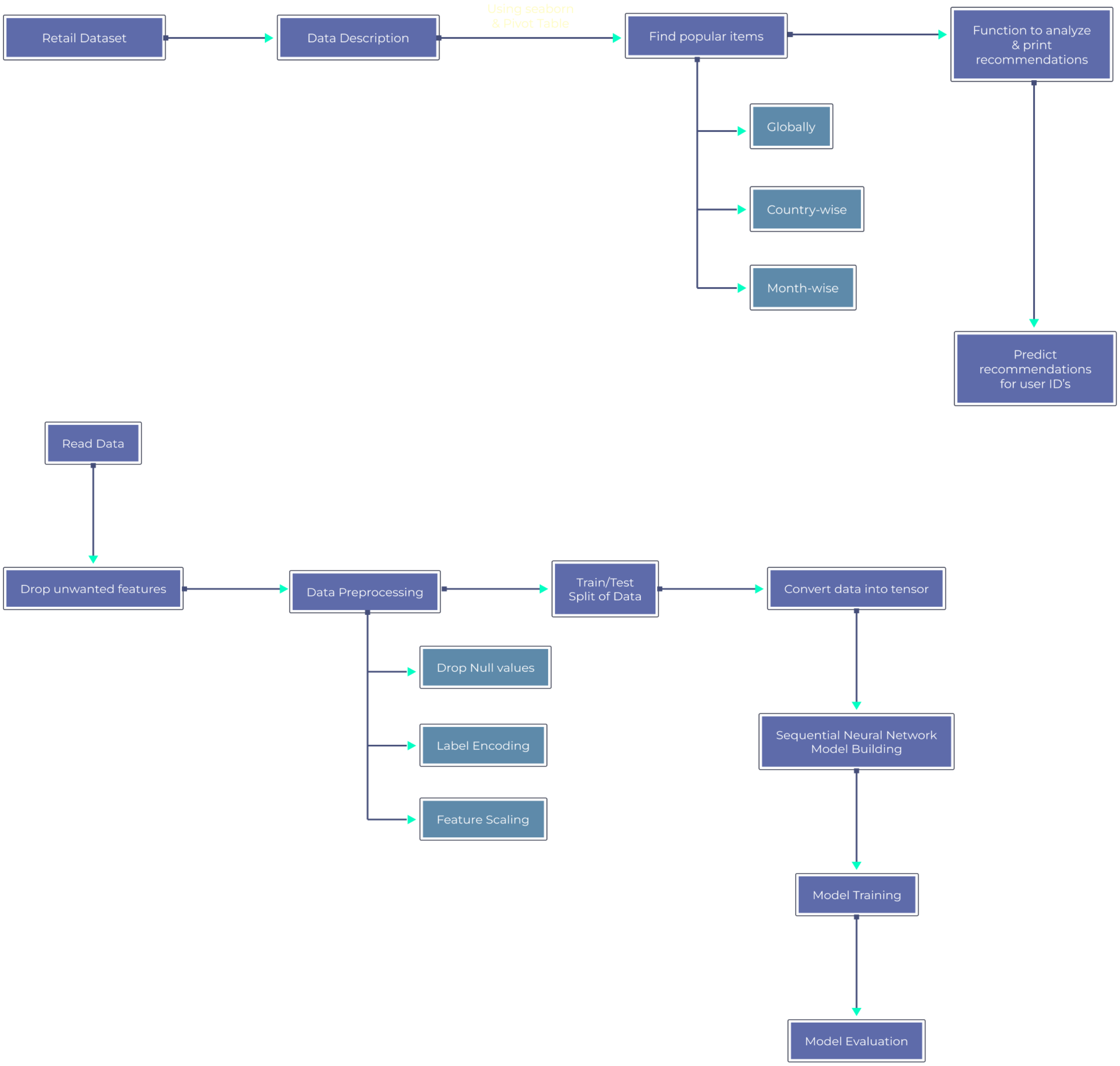


Fig 3.1- Diagrammatic overview of the project.

3.2 Hardware / Software designing

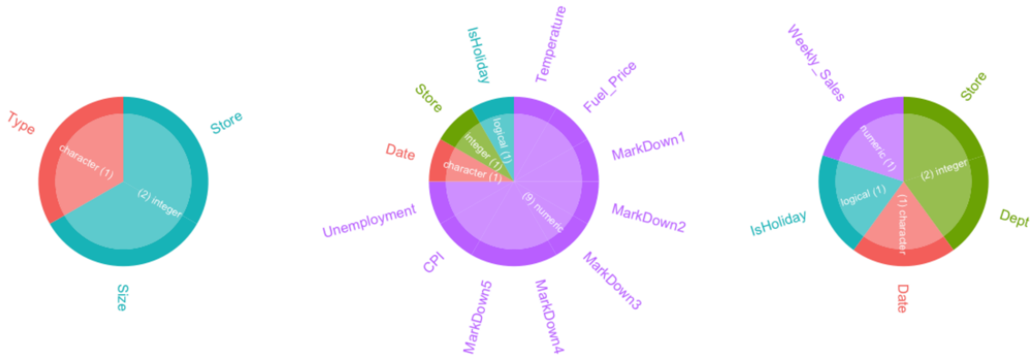
- Data: Kaggle datasets
- Software: Microsoft Excel - to handle the raw examination of the .csv file (as backup).
- Software: Anaconda Navigator - installed and equipped with Jupyter Notebooks to load in the .csv file for analysis
- Software: Windows 10 - running on a PC with at least 16 gigabytes of ram.
- Zendesk: - to log IT related support issues should they arise. (10

The analysis for this study has been performed using some main tools: R, Python, Flask and IBM. The models and Exploratory Data Analysis have been executed using development tools like PyCharm and Jupyter. Several packages have been used to perform the initial and final outcome EDA for the analysis such as numpy, scikit-learn, matplotlib, seaborn, pamdarima, matplotlib, seaborn, etc have been implemented. Packages like numpy, pandas, etc. have been used for data wrangling and manipulation. For the models that have been created, several packages like 'scikit-learn', 'xgboost', etc have been applied.

4 EXPERIMENTAL INVESTIGATIONS

The EDA resembles an essential primary investigation and attempts to take a gander at the connections and nature of the various sections accessible to us.Its components will respond to questions connected with the number and nature of segments and lines in the dataset, missing qualities, circulation of numeric and all out factors, connection coefficients, and so forth.

Several other packages like 'ggplot2', 'matplotlib', 'seaborn', and 'plotly' have also been used in this study to create visualizations that provide information about weekly sales by store and department, weekly sales on holidays versus on normal days, weekly sales based on region, store type and store size, average sales per year, change in sales as a result of factors like CPI, fuel price, temperature, and unemployment, etc in the form of heatmaps, correlation matrix (Kedia et al., 2013), histograms, scatterplots and several more. These visualizations are accompanied by brief descriptions that will discuss the findings and scope for potential modeling that will be performed in the next stages of this project.



Data types

The second section under this Exploratory Data Analysis looks at advanced and extensive visualizations that answer some crucial questions about the Walmart dataset, as listed in the purpose statement. After inspecting crucial elements in each of the data frames about the types of variables, their distribution, correlation, and association, etc. using ‘inspectdf’, more detailed and summarized information about the weekly sales for each department/store and the effect of various factors on the weekly sales are studied here. This is performed using a combination of R and Python packages like ‘ggplot2’, ‘matplotlib’, ‘seaborn’, ‘plotly’, and several others. This section will aim at looking at the following aspects of the Walmart dataset and also possibly look at some more crucial information that stems out from the below mentioned criteria:

- Identifying store as well as department-wide sales in Walmart
- Identifying sales based on store size and type
- Identifying how much sales increase during holidays
- Correlation between the different factors that affect sales
- Average sales per year
- Weekly sales as per region temperature, unemployment

5 FLOWCHART

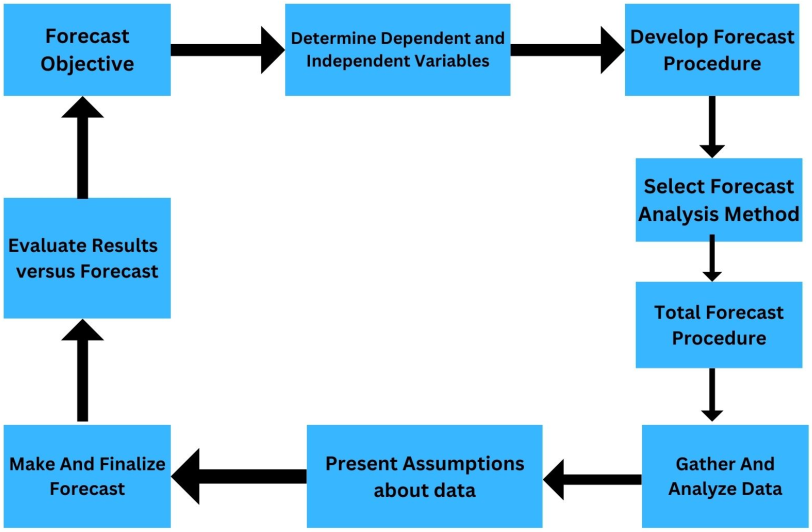
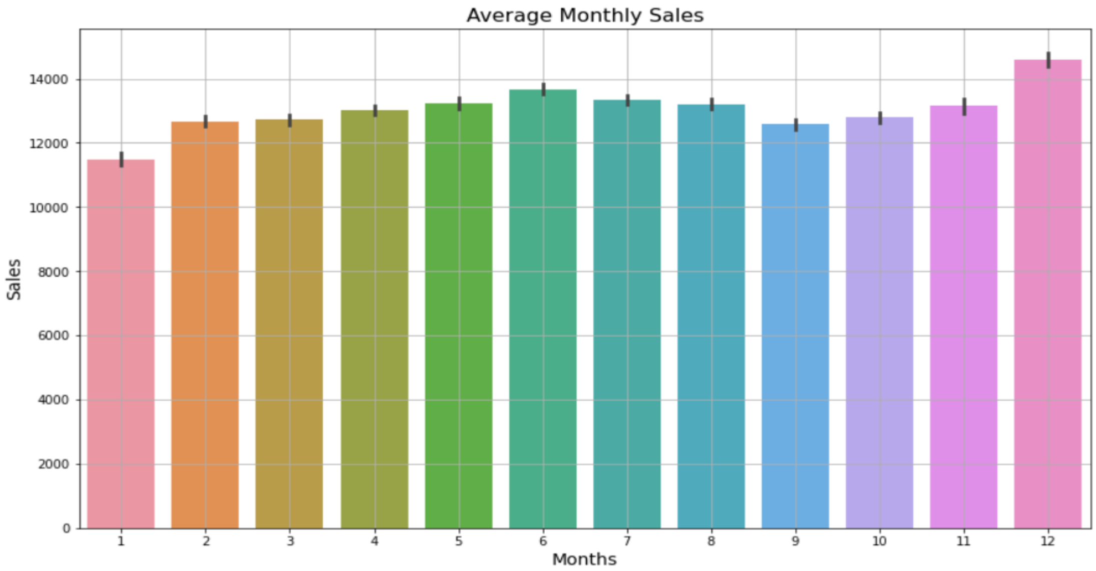
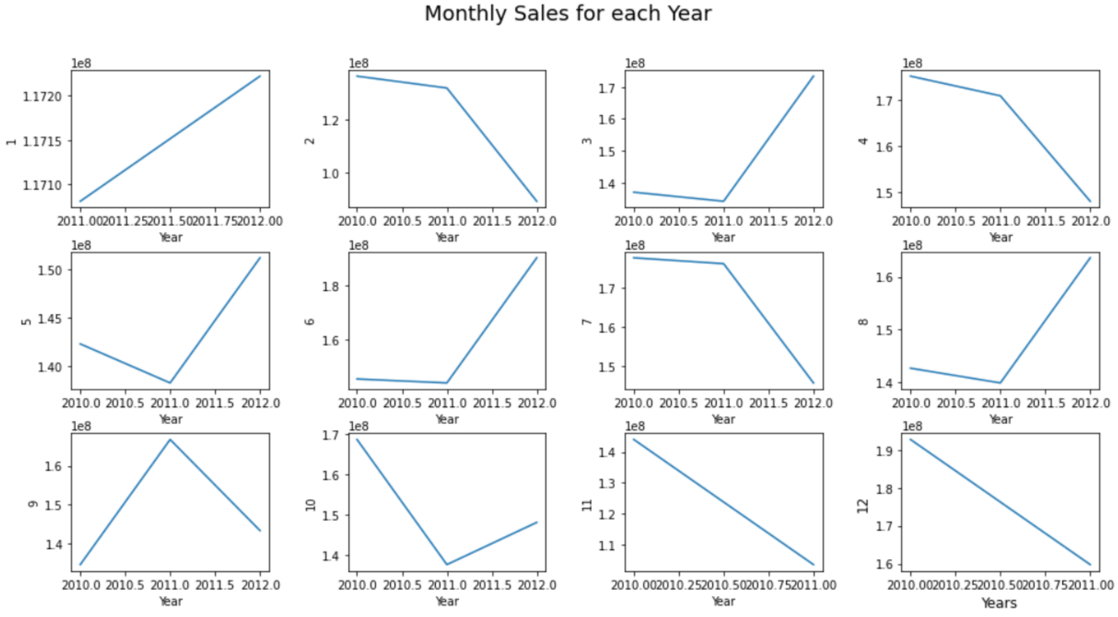

Forecast Objective
Determine Dependent and Independent Variables
Develop Forecast Procedure
Select Forecast Analysis Method
Total Forecast Procedure
Gather And Analyze Data
Present Assumptions about data
Make And Finalize Forecast
Evaluate Results versus Forecast

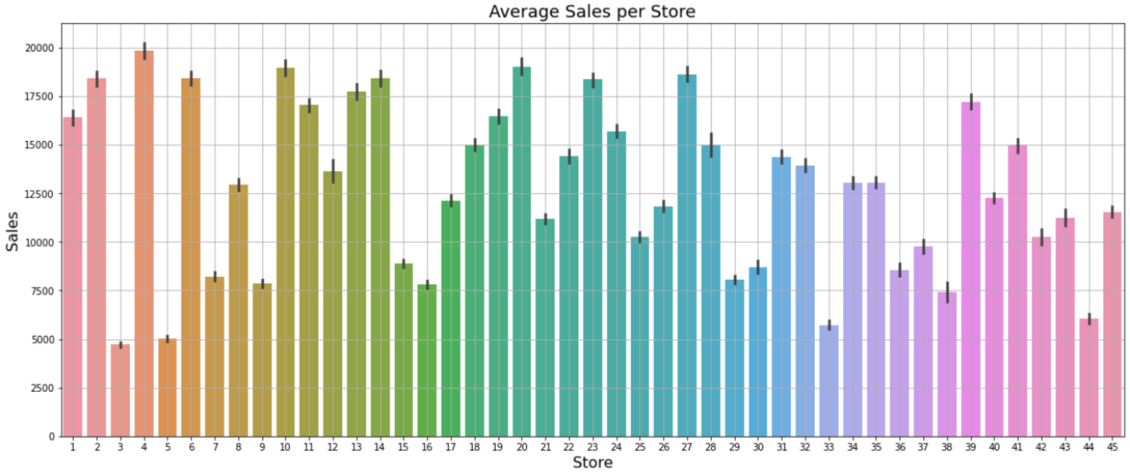
Diagram showing the control flow of the solution

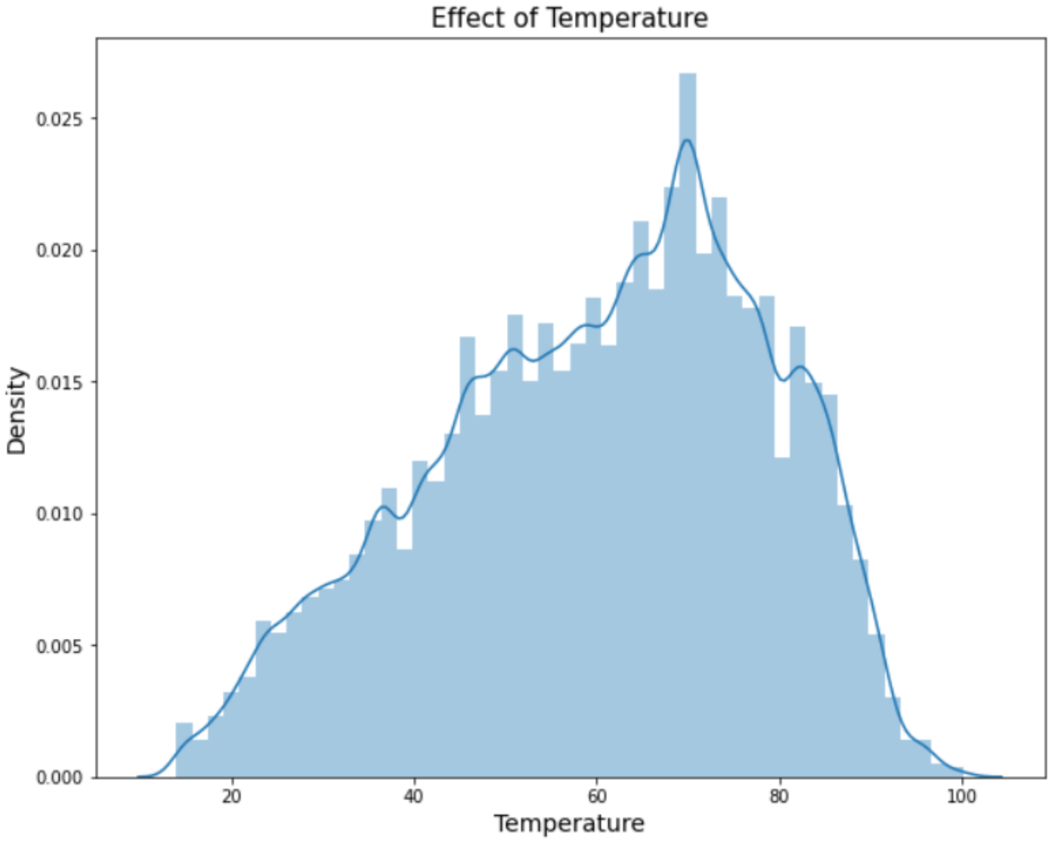
6 RESULT

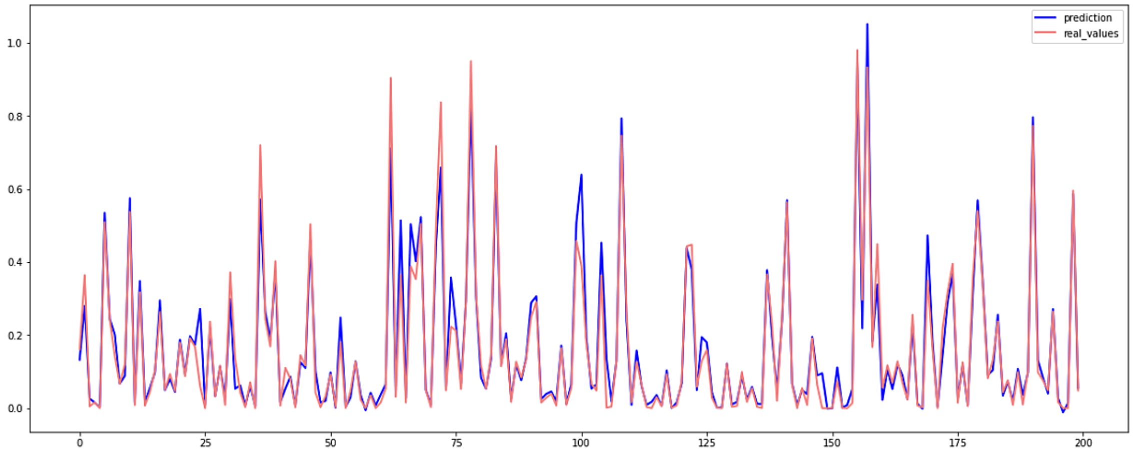
The main purpose of this study was to predict Walmart's sales based on the available historic data and identify whether factors like temperature, unemployment, fuel prices, etc affect the weekly sales of particular stores under study. This study also aims to understand whether sales are relatively higher during holidays like Christmas and Thanksgiving than normal days so that stores can work on creating promotional offers that increase sales and generate higher revenue. As observed through the exploratory data analysis, store size and holidays have a direct relationship with high Walmart sales. It was also observed that out of all the store types, Type A stores gathered the most sales forWalmart. Additionally, departments 92, 95, 38, and 72 accumulate the most sales for Walmart stores across all three store types; for all of the 45 stores, the presence of these departments in a store ensures higher sales. About the specific factors provided in the study (temperature, unemployment, CPI, and fuel price), it was observed that sales do tend to go up slightly during favorable climate conditions as well as when the prices of fuel are adequate. However, it is difficult to make a strong claim about this assumption considering the limited scope of the training dataset provided as part of this study. By the observations in the exploratory data analysis, sales also tend to be relatively higher when the unemployment level is lower. Additionally, with the dataset provided for this study, there does not seem to be a relationship between sales and the CPI index. Again, it is hard to make a substantial claim about these findings without the presence of a larger training dataset with additional information available. Interaction effects were studied as part of the linear regression model to identify if a combination of different factors could influence the weekly sales for Walmart. This was necessary because of the presence of a high number of predictor variables in the dataset. While the interaction effects were tested on a combination of significant variables, a statistically significant relationship was only observed between the independent variables of temperature, CPI and unemployment, and weekly sales (predictor variable). However, this is not definite because of the limitation of training data. Relationships between independent and target variables were tried to be identified through EDA components like the correlation matrix and scatter plots, feature importance plots created as part of the random forest and gradient boosting models as well as the interaction effects. It was discovered that, although, there were no significant relationships between weekly sales and factors like temperature, fuel price, store size, department, etc. in the correlation matrix (Figure 22), some significant relationships were observed between weekly sales and store size and department in the feature importance plots created as part of the gradient boosting and random forest models. Considering that the performance of both these models was significantly better than the performance of the regression models, it can be concluded that a non-linear statistically significant relationship exists between these independent and target variables. Finally, the tuned Gradient Boosting model, with the lowest WMAE score, is the main model used to create the final predictions for this study. These predictions can be found in the ‘sampleSubmissionFinal.csv’ file and a visualization of the project was carried out in Flask using HTML.


Average Monthly Sales
Sales
0
2000
4000
6000
8000
10000
12000
14000
Months
1
2
3
4
5
6
7
8
9
10
11
12

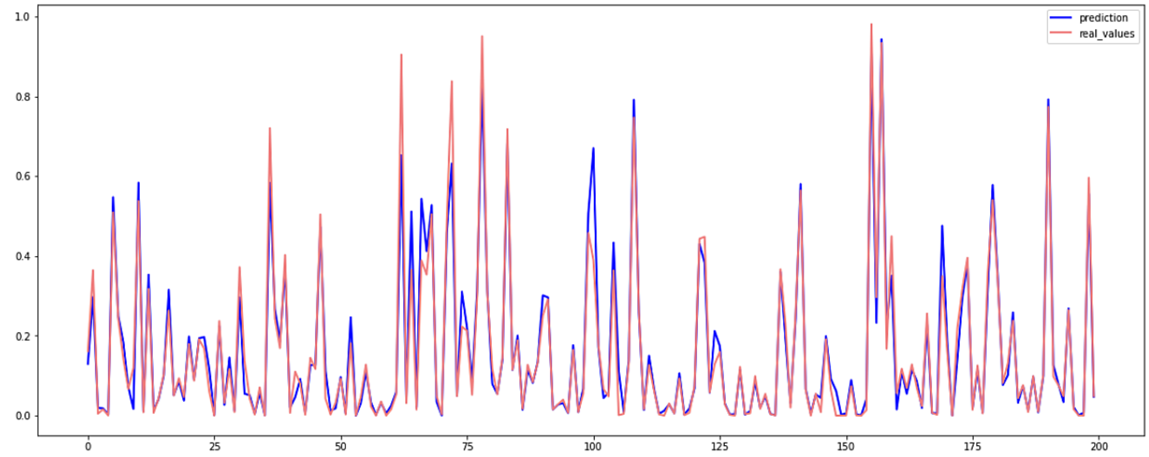

Monthly Sales for each Year
1e8
1
1.1710
1.1715
1.1720
2011.00
2011.25
2011.50
2011.75
2012.00
Year
2
3
4
5
6
7
8
9
10
11
12
2010.0
2010.5
2011.0
2011.5
2012.0
2010.00
2010.25
2010.50
2010.75
2011.00
Years


Average Sales per Store
Sales
0
2500
5000
7500
10000
12500
15000
17500
20000
Store
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45

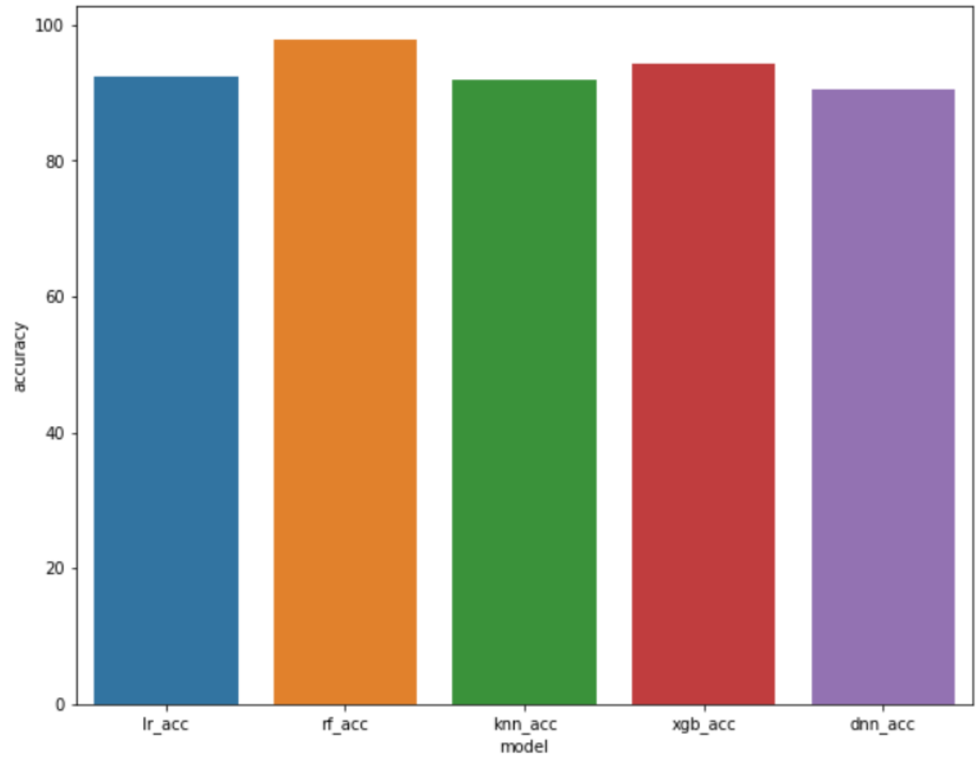




Linear Regression Model



XgBoost Model



Comparing Models

7 ADVANTAGES & DISADVANTAGES

- *Unexpected system outages (i.e., internet connectivity)*- In the rare and unlikely event of an internet outage, especially when working from a home office, it is crucial to coordinate with IT the procurement of mobile wireless hotspots. IT will oversee and monitor any issues about gateway security, and privacy, but this must be assured due to the sensitive nature of this project. Zendesk support tickets will be used to log IT related issues.
- *Security risks (i.e., privacy, data integrity)*- In the event of a data breach or leak of information, all parties must be made aware of the policies and procedures.
- *Staffing resources unavailable (i.e., temporary or permanent reduction in staff hours dedicated to this project)*- With any project, a dedicated staff may experience untimely and unforeseen emergencies leading to a reduction in hours. For this reason, we have a team of two data analysts. Both will be cross trained in each other's daily routines and handling of tasks in the event of absence by one or the other. Especially now with the pandemic, untimely absenteeism needs to be taken with a grain of salt.
- *Limited productivity due to distributed teams*- Staff deliverables between analysts and data scientists may not be met or produced in a well-timed manner. To maximize workflow efficiency, proper communication channels will be established from the onset. For example, the team will utilize Microsoft Teams in lieu of face-to-face meetings, where such communications and collaboration will be facilitated by the project manager.

8 APPLICATIONS

In uncertain times like these, business leaders would kill to have predictable revenue. Many of them are still grappling with how to forecast revenue for the next year, which is often the starting point for drawing up annual budgets for the organization. With distributed sales teams, businesses are now relying on their ability to forecast, now more than ever, to drive their entire growth strategy.

Sales forecasting is both a science and an art. Decision makers rely on these forecasts to plan for business expansion and to determine how to fuel the company's growth. So, in many ways, sales forecasting affects everyone in the organization.

- A sales forecast helps every business make better business decisions. It helps in overall business planning, budgeting, and risk management.

- Sales forecasting allows companies to allocate resources for future growth efficiently and manage their cash flow.

- Sales forecasts help sales teams achieve their goals by identifying early warning signals in their sales pipeline and course-correct before it’s too late

- Sales forecasting also helps businesses to estimate their costs and revenue accurately based on which they can predict their short-term and long-term performance.

9 CONCLUSION

In conclusion, we discover that our regression equation predicts the weekly sales with an accuracy of 84.5%. It can be used by Walmart to improve sales forecasting. They must concentrate on the inventory planning for important divisions like 38, 92, and 95. The current markdowns must be changed because they are not having the desired effect on sales. They must concentrate on the year-end inventory because weeks 51 and 52 are very important for forecasting sales.

10 FUTURE SCOPE

Walmart can concentrate more on the e-commerce parts of the business thanks to advancing technology and rising consumer demand. By drawing ideas from Amazon's business strategy, Walmart may significantly expand its online retail operation and generate enormous profits.

They can analyze the inventory data as well to optimize their inventory. They can analyze the sales targets and incentives that are given for employees to arrive at achievable sales targets for employees to motivate them better

Due to the organization's already existing stores and warehouses, it is simpler for them to expand across the country, reducing the need for physical stores and saving their consumers money on fuel by delivering goods right to their door. Additionally, it makes it much simpler to pinpoint customer purchasing trends.

11 BIBLIOGRAPHY

Brownlee, J. (2016). Feature importance and feature selection with xgboost in python. https : / / machinelearningmastery . com / feature - importance - and - feature - selection-with-xgboost-in-python/

Crown, M. (2016). Weekly sales forecasts using non-seasonal arima models. http : / / mxcrown.com/walmart-sales-forecasting/

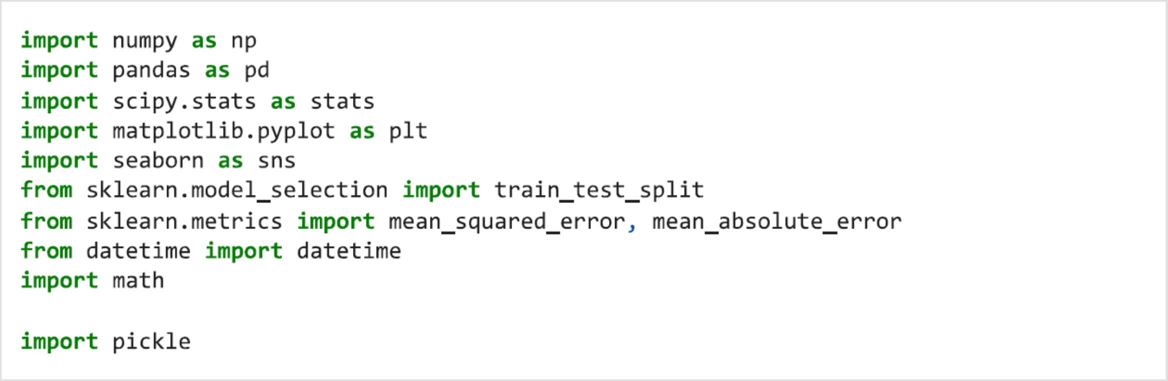
https://www.chargebee.com/blog/importance-of-sales-forecasting/#:~:text=Sales%20forecasting%20allows%20companies%20to,and%20manage%20its%20cash%20flow.&text=Sales%20forecasting%20also%20helps%20businesses,term%20and%20long%2Dterm%20performance.

Myrianthous, G. (n.d.). Training vs testing vs validation sets. https : / / towardsdatascience.com/training-vs-testing-vs-validation-sets-a44bed52a0e1 Panchotia, R. (2020). Predictive modelling using linear regression. https: / /medium. com/swlh/predictive-modelling-using-linear-regression-e0e399dc4745

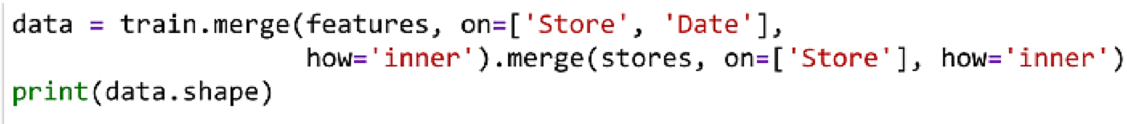
Walmart recruiting - store sales forecasting. (2014). https : / / www . kaggle . com / c / walmart-recruiting-store-sales-forecasting/data Walmart's sales data analysis - a big data analytics perspective. (2017). https://doi.org/ 10.1109/APWConCSE.2017.00028 Wikipedia, t. f. e. (n.d.). Walmart. https : / / en . wikipedia . org / w / index . php ? title = Walmart&oldid=1001006854 Yu, C. H., Lee, H. S., Lara, E., & Gan, S. (2018). The ensemble and model comparison approaches for big data analytics in social sciences. https://scholarworks.umass. edu/pare/vol23/iss1/17

12 APPENDIX

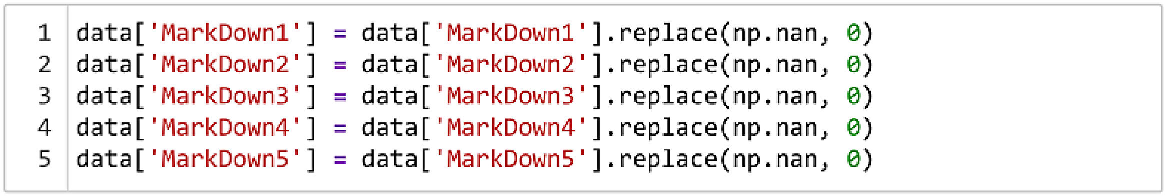
A. Source Code



```
import numpy as np
import pandas as pd
import scipy.stats as stats
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error
from datetime import datetime
import math

import pickle
```



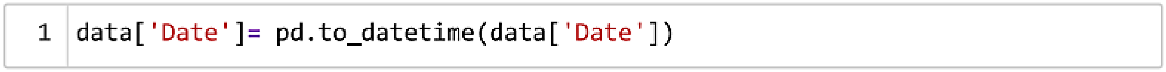
```
data = train.merge(features, on=['Store', 'Date'],
                   how='inner').merge(stores, on=['Store'], how='inner')
print(data.shape)
```



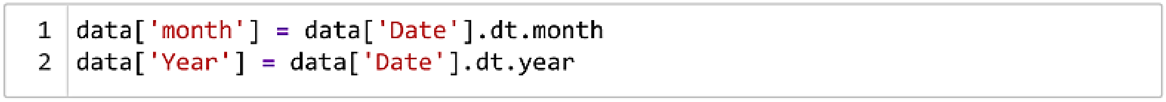
```
data['MarkDown1'] = data['MarkDown1'].replace(np.nan, 0)
data['MarkDown2'] = data['MarkDown2'].replace(np.nan, 0)
data['MarkDown3'] = data['MarkDown3'].replace(np.nan, 0)
data['MarkDown4'] = data['MarkDown4'].replace(np.nan, 0)
data['MarkDown5'] = data['MarkDown5'].replace(np.nan, 0)
```

```
data = data[data['Weekly_Sales'] >= 0]
```

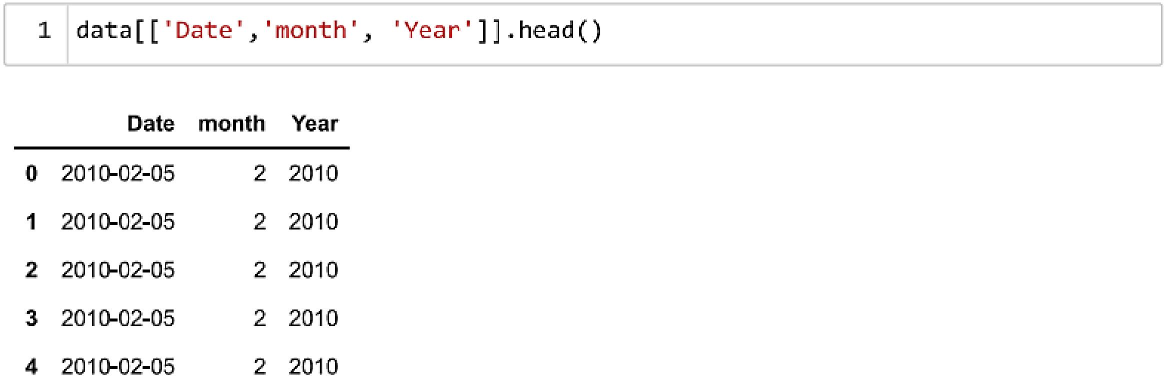
```
sorted_type = stores.groupby('Type')
plt.style.use('ggplot')
labels=['A store','B store','C store']
sizes=sorted_type.describe()['Size'].round(1)
sizes=[(22/(17+6+22))*100,(17/(17+6+22))*100,(6/(17+6+22))*100] # convert to
fig, axes = plt.subplots(1,1, figsize=(7,7))

axes.pie(sizes,
        labels=labels,
        explode=(0.0,0,0),
        autopct='%1.1f%%',
        pctdistance=0.6,
        labeldistance=1.2,
        radius=0.8,
        center=(0.5,0.5))
plt.show()
```



```
data['Date']= pd.to_datetime(data['Date'])
```

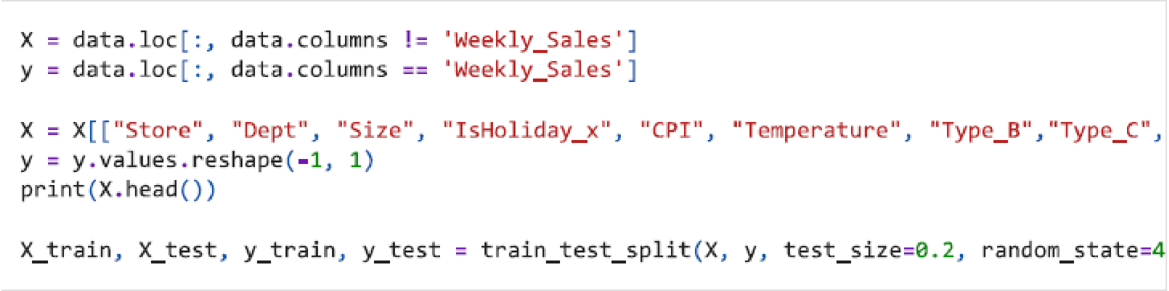


```
data['month'] = data['Date'].dt.month
data['Year'] = data['Date'].dt.year
```

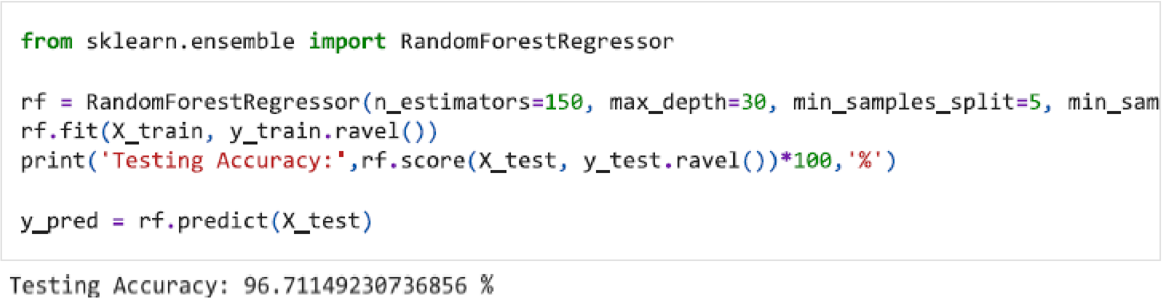


```
data[['Date','month', 'Year']].head()
```

	Date	month	Year
0	2010-02-05	2	2010
1	2010-02-05	2	2010
2	2010-02-05	2	2010
3	2010-02-05	2	2010
4	2010-02-05	2	2010

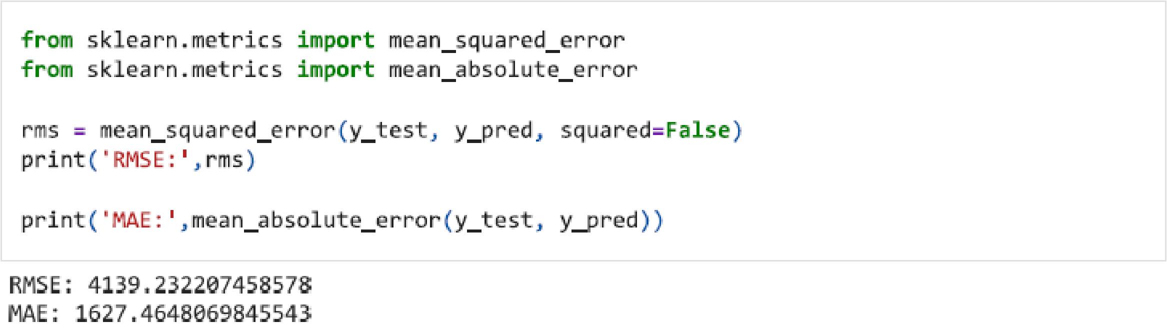


```
X = data.loc[:, data.columns != 'Weekly_Sales']
y = data.loc[:, data.columns == 'Weekly_Sales']

X = X[["Store", "Dept", "Size", "IsHoliday_x", "CPI", "Temperature", "Type_B","Type_C",
y = y.values.reshape(-1, 1)
print(X.head())

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=4
```



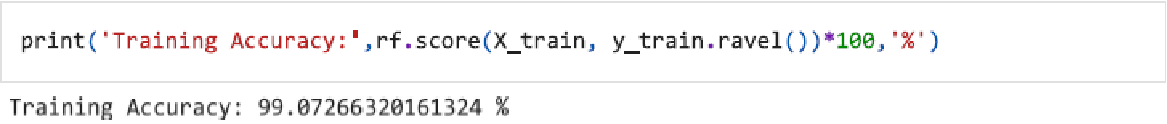
```
from sklearn.ensemble import RandomForestRegressor

rf = RandomForestRegressor(n_estimators=150, max_depth=30, min_samples_split=5, min_sam
rf.fit(X_train, y_train.ravel())
print('Testing Accuracy:',rf.score(X_test, y_test.ravel())*100,'%')

y_pred = rf.predict(X_test)
```

```
Testing Accuracy: 96.71149230736856 %
```



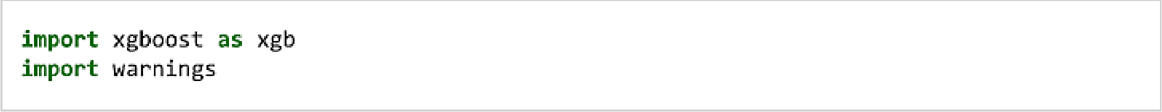
```
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error

rms = mean_squared_error(y_test, y_pred, squared=False)
print('RMSE:',rms)

print('MAE:',mean_absolute_error(y_test, y_pred))
```

```
RMSE: 4139.232207458578
MAE: 1627.4648069845543
```

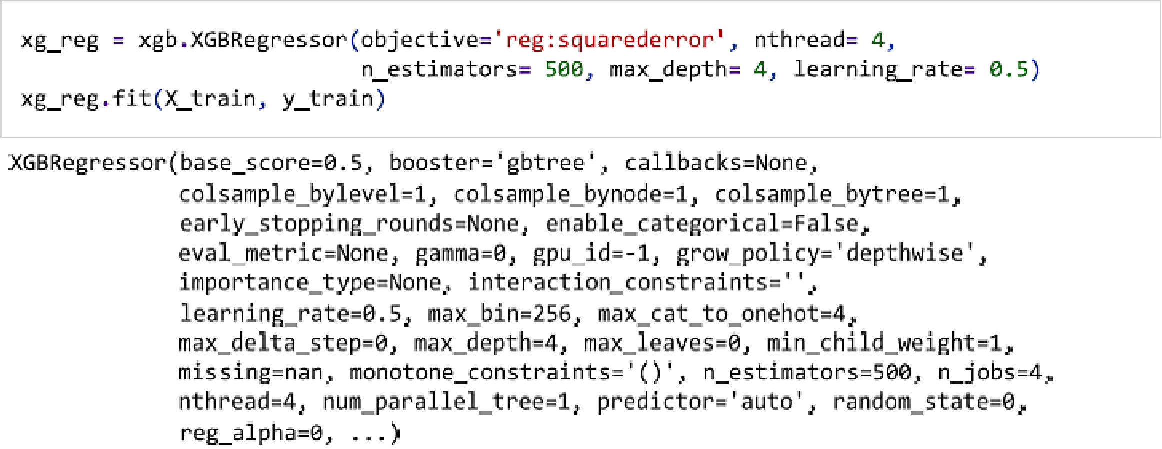


```
print('Training Accuracy:',rf.score(X_train, y_train.ravel())*100,'%')
```

```
Training Accuracy: 99.07266320161324 %
```



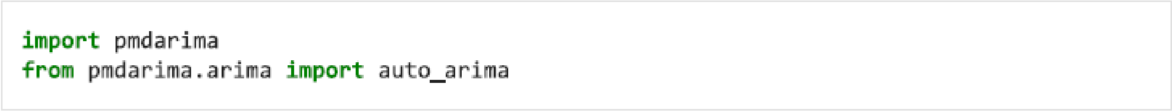
```
import xgboost as xgb
import warnings
```



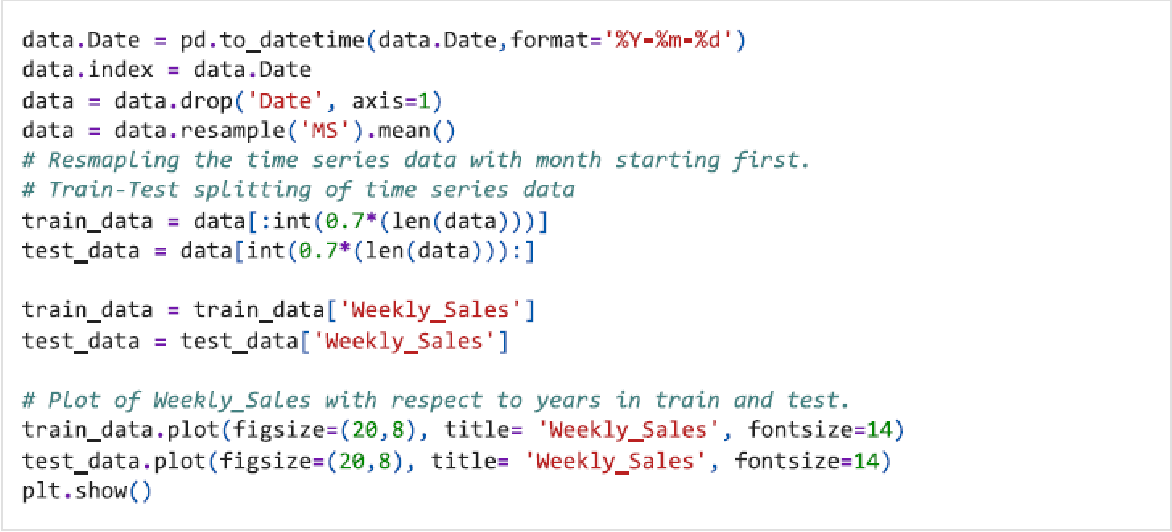
```
xg_reg = xgb.XGBRegressor(objective='reg:squarederror', nthread= 4,
                          n_estimators= 500, max_depth= 4, learning_rate= 0.5)
xg_reg.fit(X_train, y_train)
```

```
XGBRegressor(base_score=0.5, booster='gbtree', callbacks=None,
             colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
             early_stopping_rounds=None, enable_categorical=False,
             eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
             importance_type=None, interaction_constraints='',
             learning_rate=0.5, max_bin=256, max_cat_to_onehot=4,
             max_delta_step=0, max_depth=4, max_leaves=0, min_child_weight=1,
             missing=nan, monotone_constraints='()', n_estimators=500, n_jobs=4,
             nthread=4, num_parallel_tree=1, predictor='auto', random_state=0,
             reg_alpha=0, ...)
```

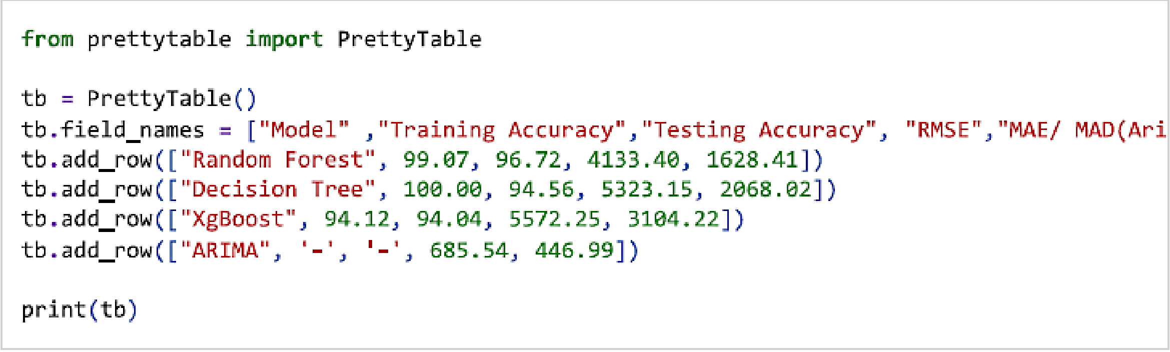
```
pred=xg_reg.predict(X_train)
y_pred=xg_reg.predict(X_test)
```



```
import pmdarima
from pmdarima.arima import auto_arima
```



```
data.Date = pd.to_datetime(data.Date,format='%Y-%m-%d')
data.index = data.Date
data = data.drop('Date', axis=1)
data = data.resample('MS').mean()
# Resmapling the time series data with month starting first.
# Train-Test splitting of time series data
train_data = data[:int(0.7*(len(data)))]
test_data = data[int(0.7*(len(data))):]

train_data = train_data['Weekly_Sales']
test_data = test_data['Weekly_Sales']

# Plot of Weekly_Sales with respect to years in train and test.
train_data.plot(figsize=(20,8), title= 'Weekly_Sales', fontsize=14)
test_data.plot(figsize=(20,8), title= 'Weekly_Sales', fontsize=14)
plt.show()
```



```
from prettytable import PrettyTable

tb = PrettyTable()
tb.field_names = ["Model" ,"Training Accuracy","Testing Accuracy", "RMSE","MAE/ MAD(Ari
tb.add_row(["Random Forest", 99.07, 96.72, 4133.40, 1628.41])
tb.add_row(["Decision Tree", 100.00, 94.56, 5323.15, 2068.02])
tb.add_row(["XgBoost", 94.12, 94.04, 5572.25, 3104.22])
tb.add_row(["ARIMA", '-', '-', 685.54, 446.99])

print(tb)
```