STORE SALES PREDICTION: WALMART

Valipe Soundarya Lahari[1], Venkata Ramana Rohith Neralla[2], Anuradha Sahithi Padavala[3], Naga Govardhan Munagala[4]
College of Engineering and Computer Science University of Central Florida, Orlando, Florida – 32816
[1]soundaryalahari@knights.ucf.edu
[2]rohithneralla@knights.ucf.edu
[3]anuradhasahithi@Knights.ucf.edu
[4]naga.govardhan.munagala@knights.ucf.edu

***Abstract*- The sales forecast is a projected measure of how a market will respond to a company's go-to-market efforts. Walmart is one of the largest retail and wholesale businesses in the world, and they should have accurate forecasts for their sales in various departments. In this project, we experimented with different machine-learning models with the dataset provided by Kaggle for the "Walmart Recruiting - Store Sales Forecasting" competition. Our findings indicate that Random Forest and the Extra Trees models are the most promising techniques for this problem.**

I. INTRODUCTION

A. Background

Sales prediction is the process of estimating future revenue by predicting the amount of product or services a sales unit (which can be an individual salesperson, a sales team, or a company) will sell in the next week, month, quarter, or year. Walmart operates a chain of hypermarkets throughout the country, and it keeps a track of the sales of each of the stores of every year and uses the data to predict future sales. In the year 2010, net sales for the full year topped $405 billion, with international net sales exceeding $100 billion for the first time. In the year 2011, Net sales for the full year were $419 billion, an increase of 3.4 percent. In the year 2012, consolidated net sales for the full year were $443.9 billion, an increase of 5.9 percent. In 2021, Walmart's sales revenue was expected to contract 2.4 following the retailer's sales surge due to the impact of the coronavirus pandemic in the previous year. From 2022 Walmart was predicted to grow at a rate of 4.8 percent and continue the positive trend until 2026.

B. Problem

Store sales are hugely impacted by the holidays and other important days that occur every year. Thus, a good store sales prediction system in big businesses like Walmart can be used to estimate the future sales and helps the management in taking better decisions related to the store. It helps in overall business planning, budgeting, and risk management. Walmart runs several markdown (promotional) events throughout the year. And these promotional

events precede the important holidays which occur through the year. Sales prediction helps Walmart organize such events more efficiently. Our aim is to build a machine learning model that can accurately predict the sales in various departments under various circumstances. The machine learning model and the data analysis can be helpful in analyzing how internal and external factors can affect sales in the future.

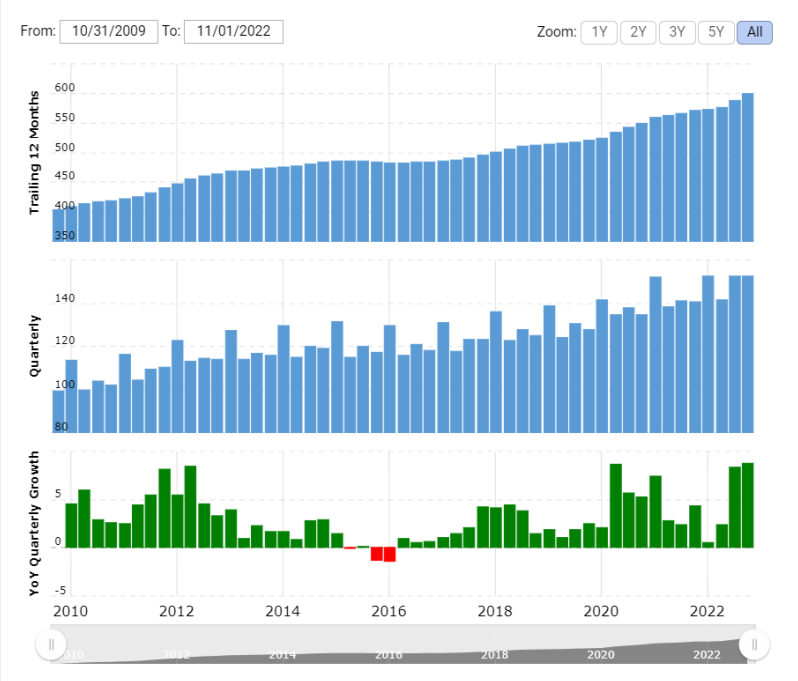


Figure 1: Walmart Revenue 2010-2022 (in billions)

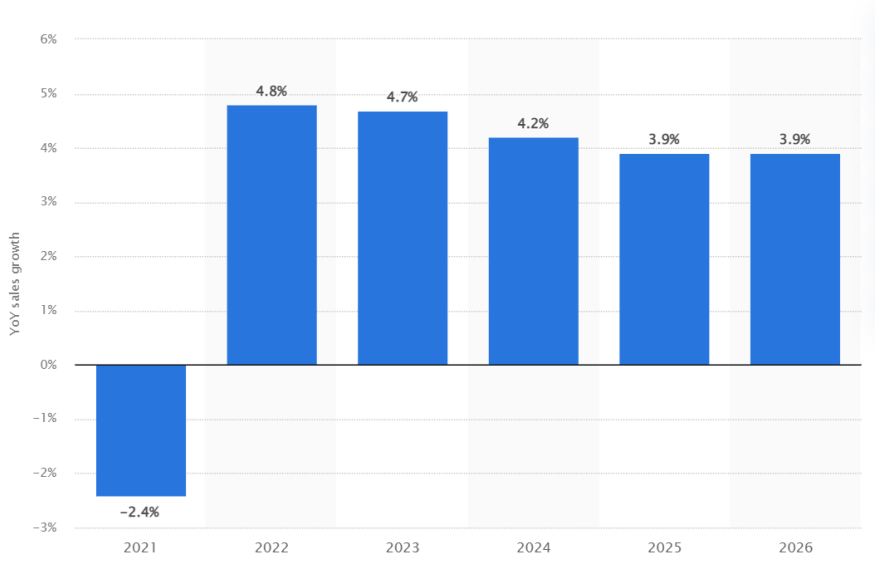


Figure 2: Forecast growth of Walmart net sales from 2021 to 2026

C. Importance

Store sales prediction is a vital task for any store or business. Businesses must forecast their sales to determine the demand for their products and decide how much inventory to refill in advance. Sales forecast assists the businesses and management in making informed decisions about staffing and prospective marketing attempts. It helps in managing the cash flow and identification of potential issues giving enough time for the store owners and management to avoid the issues or attenuate them. The future sales can be used to manage inventory around the special days in the year such as festival days.

D. Existing Literature

1) *Sales-forecasting of Retail Stores using Machine Learning Techniques:* In [6], the sales of a retail store were predicted using different machine learning algorithms. After using different models to predict sales, they found the best algorithm suited to their problem statement. They used regular regression techniques and boosting techniques in their methodology. It was found that the boosting algorithms have performed well and showed better results than the regular regression algorithms.

2) *Sales Forecasting for Retail Chains:* In [7], data mining techniques are used for sales prediction of a European Pharmacy retailing company. The features that are used for prediction include retail competitors, school and state holidays, location, prior sales data, store promotions, time of the year and accessibility of the store. The Extreme Gradient Boosting algorithm was used as the model for prediction of the sales along with other traditional regression models. It was found that Xgboost outperformed the regression models and revealed the hidden facts in the data which helped in more accurate predictions of sales.

E. System Overview

Our project "Stores Sales Prediction: Walmart" is used to predict the sales of selected Walmart stores in the country for a particular time frame whose train and test data is provided by Walmart as a part of a Kaggle competition. The competition is hosted by Walmart to find the best suited processing and modelling to their problem statement. In this project, we first explored the data provided by Walmart to gain useful insights from the data, we performed data pre-processing and made it ready for modelling. Then, we trained models such as linear regression, ridge regression, decision trees, random forests and extra trees and calculated the training error, validation error using the training dataset. Based on the error values obtained, we chose the best models and using

the best models, we predicted the sales values of test data, combined the predictions of the two best models on test data and submitted it to the Kaggle competition. The figure below depicts our system overview.

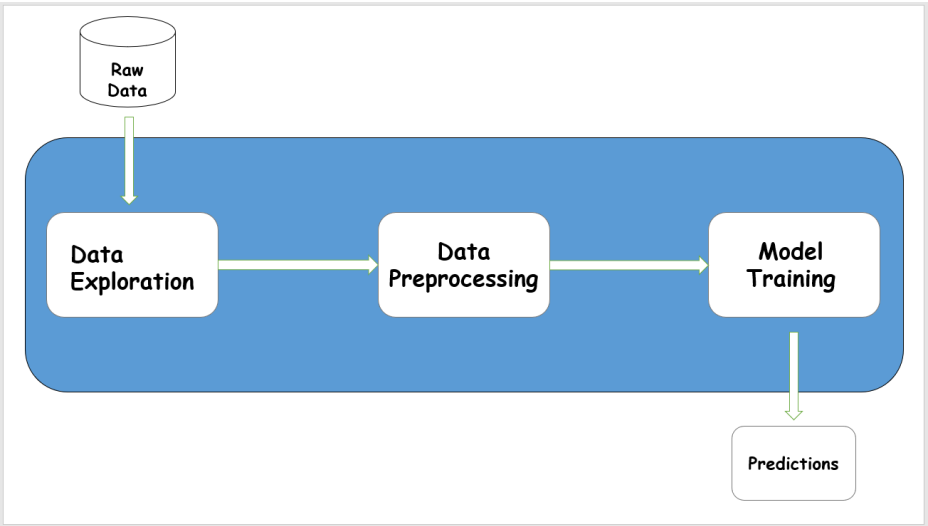


Figure 3: System Overview

F. Data Collection

Walmart provides us with historical sales data for 45 Walmart stores and the stores located in different regions throughout the country for the years 2010 - 2012. Each store contains multiple departments. We must project the weekly sales for each department in each store. Walmart also provides data related to Walmart's promotional/markdown events and factors governing the stores in different regions such as fuel price, temperature, etcetera. Walmart provided the data in the form of four datasets namely, *store.csv*, *features.csv*, *train.csv* and *test.csv*.

G. Components of ML System

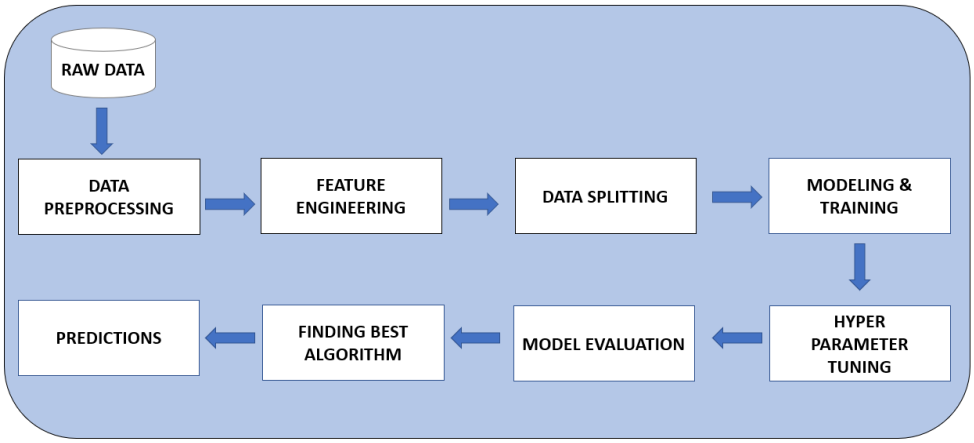


Figure 4: Components of ML system

1) *Data pre-processing*: Data pre-processing, is a part of the data preparation process. It is type of processing that is performed on raw data, which prepares it for another data processing procedure. It is the primary step in the data mining process. It transforms the data into a format that is easily processed in data mining and machine learning tasks. The data pre-processing techniques are generally used at the initial stages of the machine learning pipeline. In our project, as a part of data pre-processing, we performed steps such as *filling up missing values, encoding categorical data, and dropping irrelevant variables*.

2) *Feature Engineering*: In Feature Engineering, we use domain knowledge to choose and transform the most relevant and important variables from the data which can be used later for modelling in Machine Learning. Feature Engineering is performed to enhance the performance of Machine Learning models. In our project, we performed feature engineering on the variables *Date***,** introduced a new variable related to promotional events called *MarkdownsSum* from Markdown1-5 and introduced new variables such as *Days_to_Thanksgiving***,** *Days_to_Christmas*, *SuperBowlWeek*, *LaborDay*, *Thanksgiving*, *Christmas* based on the day and month of the year.

3) *Data Splitting*: The data which is ready for modelling is split into subsets of data for the purpose of training and validation. In our project, we used the training dataset after data pre-processing and feature engineering and split it into *training* and *validation subsets*.

4) *Modelling and Training*: In this step, we create a machine learning model and feed sufficient training data for the model to learn from. We created and trained the models - linear regression, ridge regression, decision trees, random forests, and extra trees.

5) *Hyperparameter Tuning*: In hyperparameter tuning, we choose a set of optimal parameters for the learning algorithm. Machine Learning algorithms have some parameters which control their learning. Thus, optimal values of these parameters can enhance the performance of the learning model. In our project, we used *Hyperparameter Tuning for Random Forest model* and found the optimal values for some of its parameters.

6) *Model Evaluation*: In this step, we use an evaluation metric in order to understand and analyse model performance. It also helps us in understanding the models' strengths and weaknesses. The evaluation metric we are using is *Weighted Mean Absolute Error (WMAE)* as it is the metric indicated by Kaggle to be used for this project.

7) *Finding the best algorithm*: After model evaluation, based on the values of evaluation metric, we select the best model which gave the most accurate results. In our project, after comparing the WMAE scores of different models we implemented, it was found that *Random Forest* performed the best with least value of WMAE. The second best model was Extra Trees.

8) *Predictions*: After selecting the best algorithms/models, we use the model for prediction of values of test data. In our project, we fed test data to the trained Random Forest model and Extra Trees model since they were the best models and obtained the weekly sales predictions of both the models. The prediction results which were obtained were combined. This data was submitted to Kaggle competition in order to get the WMAE (error) value in the public leader board.

II. IMPORTANT DEFINITIONS

Data

The data has been taken from the Kaggle "Walmart Recruiting - Store Sales Forecasting" competition. It offers historical sales data for 45 Walmart locations across the US. Each store has several departments.

The following files are used to forecast sales.

1) *stores.csv*: This file contains some more detailed information about the type and size of these 45 stores used in this study.

2) *train.csv*: It is the historical training data, which covers from 2010-02-05 to 2012-11-01. It also contains a column that suggests whether a particular date falls on a holiday or not. Within this file, we have the following fields:

   - Store - the store number
   - Dept - the department number
   - Date - the week

- Weekly_Sales - sales for the given department in the given store
- IsHoliday - whether the week is a special holiday week

3) *test.*csv: This file is identical to train.csv without the weekly sales field. We must predict the sales for each triplet of store, department, and date in this file.

4) *features.*csv: This file contains additional data related to the store, department, and regional activity for the given dates. It has the following fields:

- Store - the store number
- Date - the week
- Temperature - the average temperature in the region
- Fuel_Price - the cost of fuel in the region
- MarkDown1-5 - anonymized data related to promotional markdowns that Walmart is running. MarkDown data is only available after Nov 2011 and is not available for all stores all the time. Any missing value is marked with a NA.
- CPI - the consumer price index
- Unemployment - the unemployment rate
- IsHoliday - whether the week is a special holiday week

HOLIDAY NAME	**DATE 1**	**DATE 2**	**DATE 3**	**DATE 4**
Super Bowl	12-Feb-10	11-Feb-11	10-Feb-12	8-Feb-13
Labor Day	10-Sep-10	9-Sep-11	7-Sep-12	6-Sep-13
Thanksgiving	26-Nov-10	25-Nov-11	23-Nov-12	29-Nov-13
Christmas	31-Dec-10	30-Dec-11	28-Dec-12	27-Dec-13

TABLE 1
List of Holidays from the dataset

Prediction Target

For each row in the test dataset (store + department + date triplet), we should predict the weekly sales of that department. The Id column is formed by concatenating the Store, Dept, and Date with underscores (e.g. Store_Dept_2012-11-02).

Other Important Definitions

Variables: Variables are the individual observation units or data points for a training sample.

Independent Variables: Independent variables are the ones that you include in the model to explain or predict changes in the dependent variable.

Dependent Variables: The dependent variable is what you want to use the model to explain or predict.

Forward Pass: Step in the model training process that deals with calculating the predicted output for each training sample is called Forwards pass.

Supervised Learning: Supervised machine learning is when the algorithm (or model) is created using what's called a training dataset. The model is trained using many different examples of various inputs and outputs and thus learns how to classify any new input data it receives in the future.

Regression: Regression is a statistical method that attempts to determine the strength and character of the relationship between one dependent variable and a series of independent variables.

Linear Regression: Linear Regression is the supervised Machine Learning model in which the model finds the best fit linear line between the independent and dependent variable.

Ridge Regression: Ridge regression is a model tuning method used to analyse data that suffers from multicollinearity. This method performs L2 regularization.

Decision Tree: The decision tree predicts the class or value of the target variable by learning simple decision rules inferred from the training data.

Gini Index: Gini Index calculates the probability of a certain randomly selected feature that was classified incorrectly.

Random Forest: Random Forest is a powerful and versatile supervised machine learning algorithm that grows and combines multiple decision trees to create a "forest." It can be used for both classification and regression problems.

Ensemble: Ensemble modelling is a process where multiple diverse models are created to predict an outcome, either by using many different modelling algorithms or using different training data sets.

Bagging: It creates a different training subset from sample training data with replacement & the final output is based on majority voting. For example, Random Forest.

Boosting: It combines weak learners into strong learners by creating sequential models such that the final model has the highest accuracy. For example, ADA BOOST, XG BOOST.

Extra Trees: The extra trees algorithm, like the random forest algorithm, creates many decision trees, but the sampling for each tree is random, without replacement.

Constraints faced with Problem Statement

One of the biggest retailers in the world, Walmart needs precise estimates for its sales in a variety of divisions. A sales forecast helps every business make better business decisions. A forecast without rational analysis of historical data and sales forecasts could have severe repercussions for this store.

Our goal is to develop a machine-learning model that can properly forecast sales in various departments under varied conditions by analysing the weekly performance of each store over the prior year and examining the potential future effects of internal and external influences on sales. Also, the model needs to be computationally efficient and scalable.

The main problems we encountered in our research were caused by the large dataset, which presented various computational difficulties. Finding the appropriate variables to base the analysis was another difficult task. Also, the provided dataset has a large amount of missing data to be handled.

III. Overview Of Proposed Approach

Libraries

We first imported the following python Libraries- Pandas, NumPy, Matplotlib, Seaborn, Xgboost, Catboost, Lightgbm.

Data Loading

Walmart Sales dataset consists of five CSV files, namely, 'stores.csv', 'train.csv', 'test.csv' and 'features.csv'. We loaded these 4 files from their respective locations.

Data Merging

Having to deal with 4 different files would be a hectic operation. Hence, we merged all the 4 files to form 2 main data frames, namely, 'train_df' and 'test_df'. Under the name of 'feature_store', we firstly merged the features.csv and stores.csv based on Store field. Then, for 'train_df', we merged the 'feature_store' data frame and train.csv file based on Store, Date and IsHoliday fields. Now, for 'test_df', we again merged 'feature_store' data frame with test.csv file.

Exploratory Data Analysis

As we have sales data coming from 45 different Walmart stores all around the country, it is important to analyse the data we have at hand. After going through the store.csv file, it is evident that there are 3 types of stores, namely, A, B and C. There are 22 stores under type A, 17 stores under type B and 6 stores under type C.

Now, we tried to understand how the Type field effects the weekly sales of a store. We plotted a bar graph showing the average weekly sales for a store under each type – A, B and C, as shown below. We observe that a store of Type A tends to have more weekly sales.

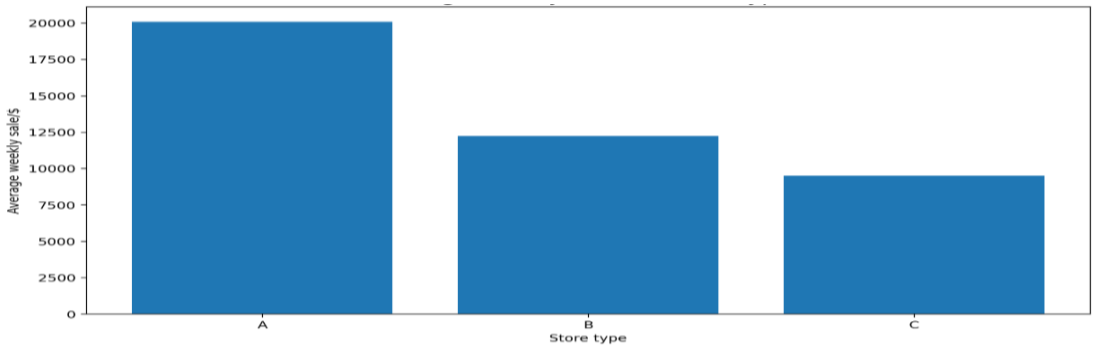


Figure 6: Average Weekly Sales of Store Types

We then tried to analyse the effect of a holiday on the weekly sales. 'IsHoliday' field signifies whether that week has a notable holiday or not. If it does, the value is True and the value is False if it is otherwise. Now, firstly we need to understand how many records have the value True for that field. We found out that there are 29,661 true values out of 4,21,470 records. That is considerably a small number. Now, we came up with the average weekly sales whenever there was a holiday in a week and when there was not one. We observe that when there is a holiday, the average weekly sales tend to be a little higher than when there is no holiday. The graphs below represent the same information.

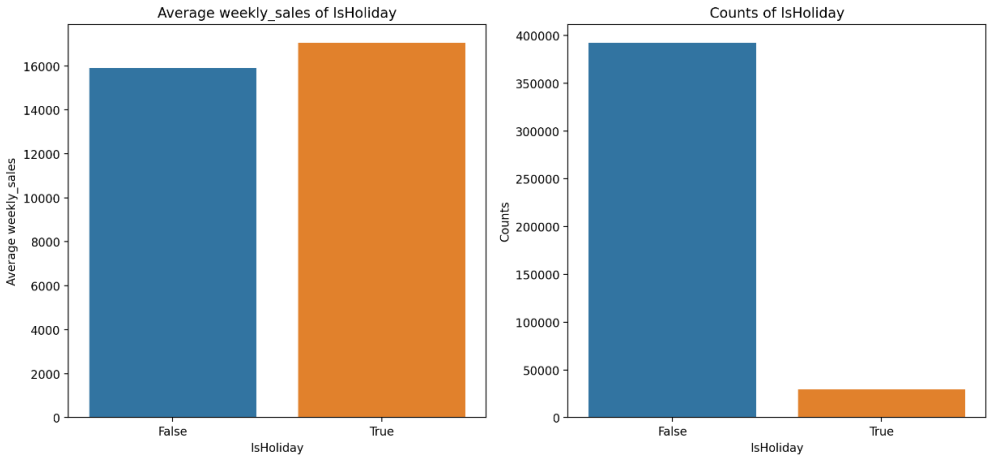


Figure 7: Weekly Sales – IsHoliday

For the next step, we wanted to understand how the week of a year impacts the weekly sales. For this, we had to first split the Date field into 'Day', 'Month' 'Week' and 'Year'. The 'Week' attribute signifies the week number of that day in that year. For example, the date: 02/05/2011, falls on week 6 of the year 2011. After splitting, we plotted a graph for weekly sales for each week of 2010, 2011 and 2012 respectively.

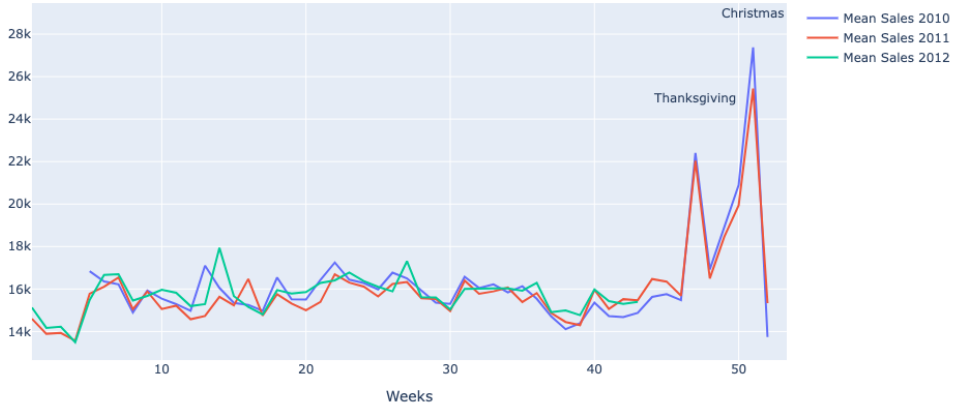


Figure 8: Weekly sales of Years 2010, 2011, 2012

We observe that weekly sales for most of the year remain stable, except for a few ups and downs. But at around week 46, which marks the start of the Thanksgiving season, the weekly sales soar up high and will reach an all-time high with the start of Christmas season. Thus, we conclude that festival seasons which bring deals, also bring huge rise in sales.

Now, we try to understand how store size impacts the sales. For that, we plot a graph with Sales on Y-axis and Size on X-axis, as shown below. We realize that, for the most part, Sales increase with increase in size of the Store. Hence, Size is an important feature.

Figure 9: Store Size and Sales

We then find out that Type A have bigger sized stores, followed by Types B and C.

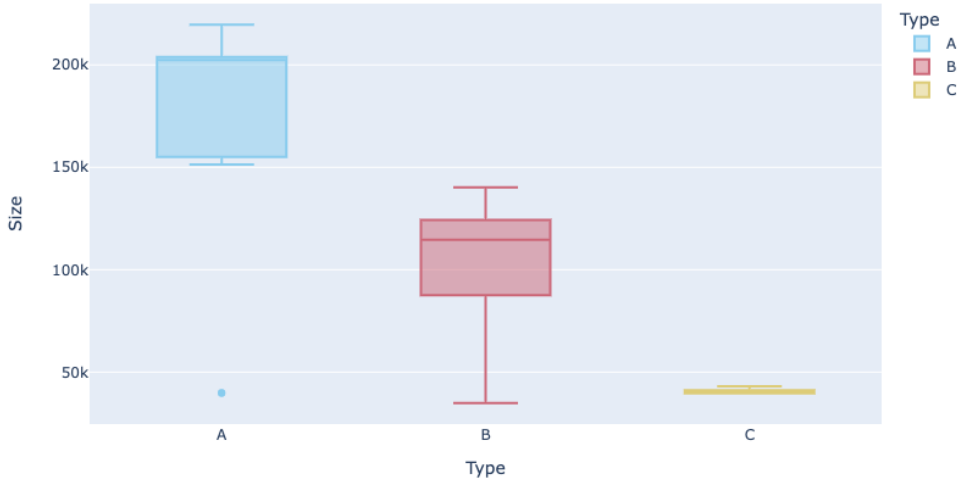


Figure 10: Store Size and Store Type

We also find out that even though Type C stores have the smallest size and are least in, they have the highest median weekly sales at 21k, followed by Type A at 18.2k and Type B at 17.8k. We plotted the following graph for the same.

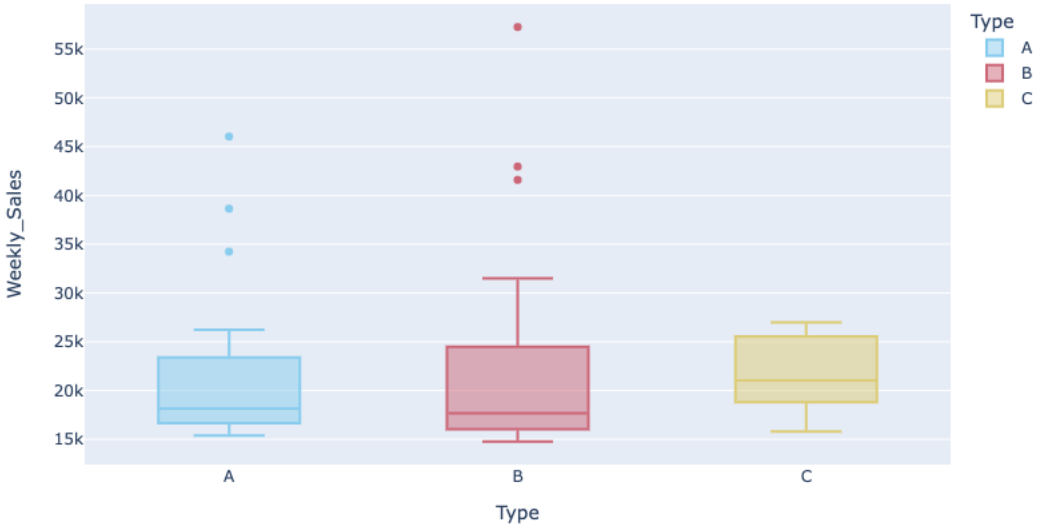


Figure 11: Store Type and Sales

From train.csv, we know that there are 99 different departments in each store. We now try to analyze how the sales are for different departments. We plotted a graph between Weekly Sales and Department fields to understand the same. From the graph, it is evident that some departments generate more sales compared to others and hence it is an important attribute.

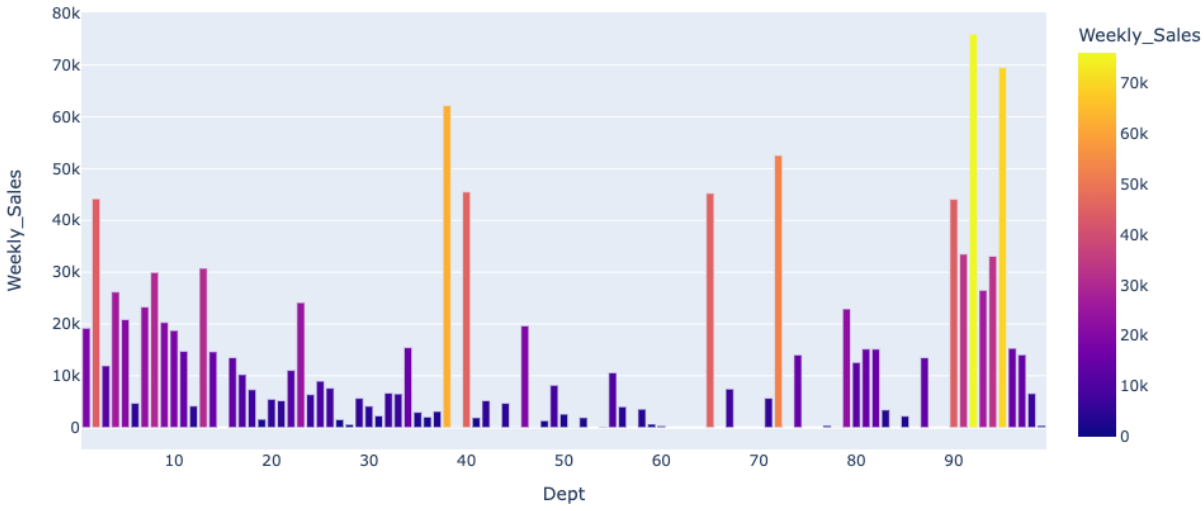


Figure 12: Department and Sales

Now, after analysing all these features, we try to find the correlation between features, and we plot a heatmap which clearly depicts the correlation of each feature with every other feature. After that, we will plot the correlation of each feature with Weekly Sales and try to analyse which one impacts Weekly Sales the most.

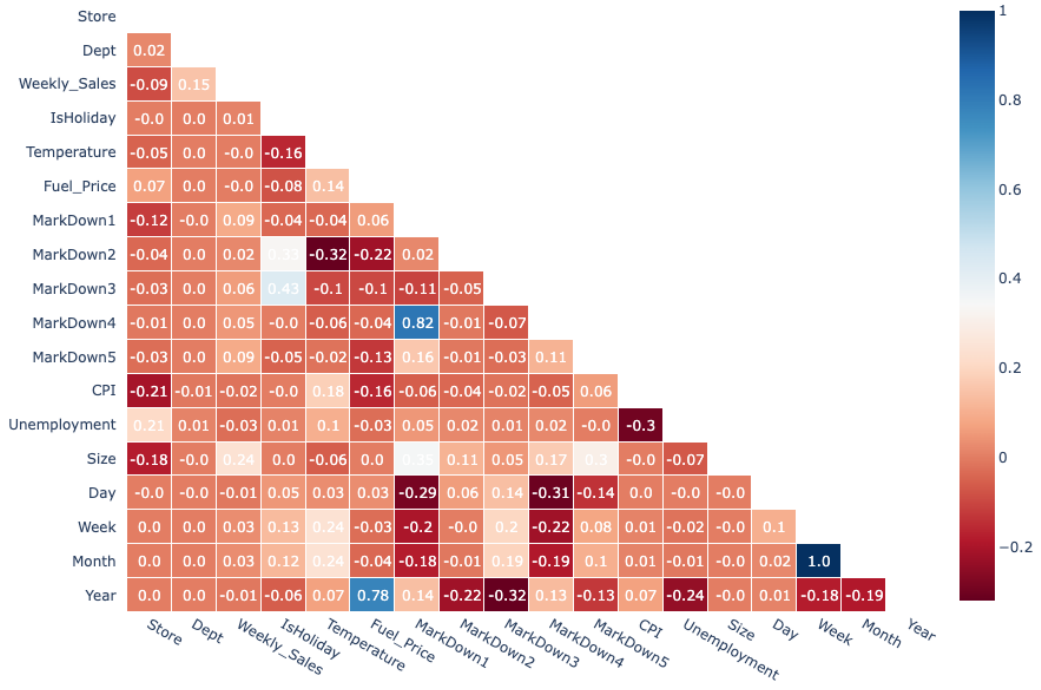


Figure 13: Heat Map of Correlations

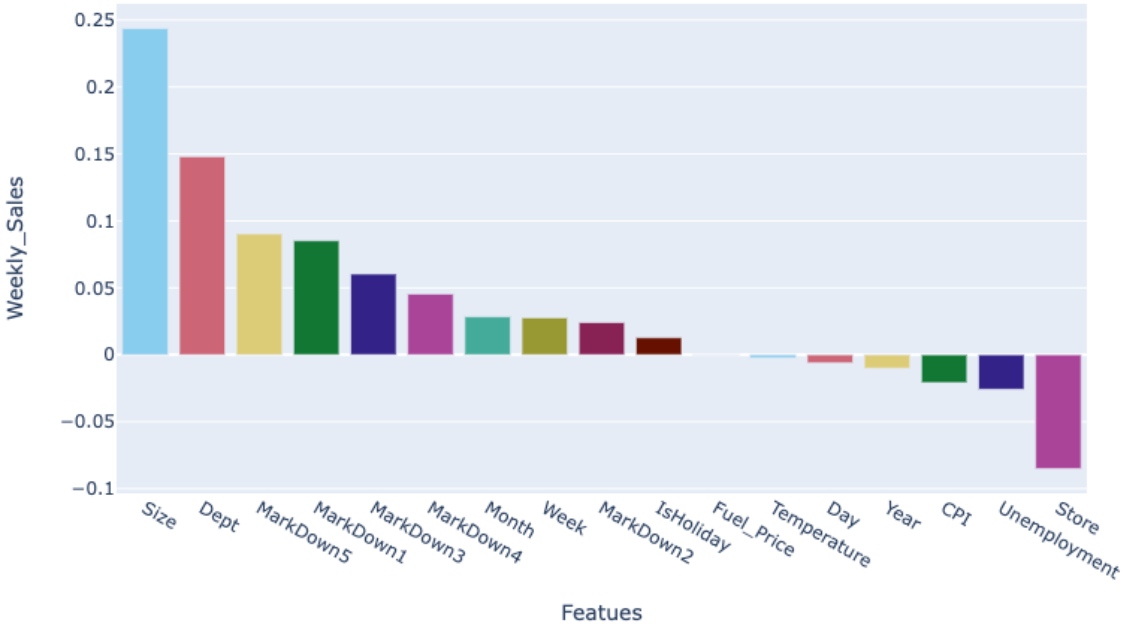


Figure 14: Feature Correlation with Weekly Sales

We see that Size have the highest correlation with Weekly Sales. Other features like Department, Month, Week, IsHoliday and Markdown fields have a positive correlation with Weekly Sales. Unemployment and CPI have negative correlations with Weekly Sales. This means that as Unemployment value increases, Weekly Sales decrease.

Feature Engineering

1) *Holidays Fields*: We have established that Week of the year and major holiday seasons like Thanksgiving and Christmas have a huge impact on Sales. Hence, we now tried to extract new features like,
    - Days_to_Thanksgiving – the number of days remaining for Thanksgiving.
    - Days_to_Christmas – the number of days remaining for Christmas
    - SuperBowlWeek – signifies whether that week has Super Bowl event
    - LaborDay – signifies whether that week has Labor Day
    - ThanksGiving – Signifies whether that week has Thanksgiving Thursday.
    - Christmas – Signifies whether Christmas falls in that week.

2) *Markdown Fields*: In the previous section, we found out that all the Markdown fields, i.e., Markdown1-5, have a positive, yet negligible correlation with Weekly Sales. We could ignore these fields if they must be considered individually. But all of them when summed up will have a considerable impact on the weekly sales. Hence, we add up all the Markdown values for that record into a new feature called 'MarkdownsSum'.

	Store	Date	MarkdownsSum	Days_to_Thanksgiving	Days_to_Christmas	SuperBowlWeek	LaborDay	Thanksgiving	Christmas
421565	45	2012-09-28	9468.01	57	87	0	0	0	0
421566	45	2012-10-05	NaN	50	80	0	0	0	0
421567	45	2012-10-12	NaN	43	73	0	0	0	0
421568	45	2012-10-19	NaN	36	66	0	0	0	0
421569	45	2012-10-26	5247.26	29	59	0	0	0	0

Figure 15: Features Extracted

Pre-processing

1) *Filling the Missing Values:*

After appropriate evaluation, we find out that there are 75 percent of entries for Markdown fields have NA values. Hence, we replaced all the NA values with 0s in both the 'train_df' and 'test_df' data frames. CPI and Unemployment fields also have missing values. For these fields, we replace the NA values with the mean of all values in that respective column.

2) *Encoding the Categorical Data:*

We know that 'IsHoliday' is a Boolean attribute which takes True and False values. Now, to make this field ready to be fed to the model, we need to convert the categorical values in this field into numerical values. Here, we replace 'True' with 1 and 'False' with 0. Also, the 'Type' attribute takes categorical values, i.e., 'A', 'B' and 'C'. We replace those with 1, 2 and 3 respectively.

IV. TECHNICAL DETAILS

Feature Selection

For the selection of appropriate features to be fed to the models, we use the function 'PermutationImportance' to calculate the weighted importance of each feature in the data frame.

We calculated the weighted-importance's and arranged them in descending order as shown in the figure below.

	Weight	Feature
0	1.6708 ± 0.0492	Dept
1	0.4636 ± 0.0111	Size
2	0.1097 ± 0.0030	Store
3	0.0454 ± 0.0026	CPI
4	0.0344 ± 0.0059	Week
5	0.0201 ± 0.0177	Thanksgiving
6	0.0159 ± 0.0010	Type
7	0.0109 ± 0.0019	Days_to_Thansksgiving
8	0.0085 ± 0.0014	Day
9	0.0064 ± 0.0099	MarkDown3
10	0.0063 ± 0.0009	Temperature
11	0.0061 ± 0.0005	Unemployment
12	0.0025 ± 0.0005	IsHoliday
13	0.0019 ± 0.0004	Fuel_Price
14	0.0013 ± 0.0002	Month
15	0.0009 ± 0.0005	MarkDown4
16	0.0006 ± 0.0001	MarkDown5
17	0.0003 ± 0.0001	MarkDown2
18	0.0001 ± 0.0003	MarkdownsSum
19	0.0001 ± 0.0000	Year
20	0.0001 ± 0.0000	LaborDay
21	0 ± 0.0000	Christmas
22	0 ± 0.0000	Days_to_Christmas
23	-0.0000 ± 0.0000	SuperBowlWeek
24	-0.0000 ± 0.0001	MarkDown1

Figure 16: Weighted Importance of features

We then chose the X-baseline features for the modelling, and those included Department, Size, Store, CPI, Week, Thanksgiving, Type, Day, Year and IsHoliday which are the top features.

Data Splitting

We then split the train data into Training Set and Validation Set. They are split into Training and Validation in the ratio of 9:1 respectively with some random state.

Machine Learning Models

Linear Regression

Predicting the value of a dependent variable based on an independent variable is the aim of a simple linear regression. The prediction is more accurate the stronger the linear relationship between the independent and dependent variables. We made use of the imported LinearRegression() function. Training the model with the Linear Regression model gave us a WMAE score of 14808.15 on the training dataset and 14834.85 on the validation dataset, which is not that great.

```
Training dataset WMAE is 14808.15
Validation dataset WMAE is 14834.85
```

Ridge Regression

The technique used to analyse multicollinearity in multiple regression data is called ridge regression. Ridge regression uses a type of shrinkage estimator called a ridge estimator. It uses the L2 regularization. We made use of the imported Ridge () function. Training the model with the Ridge Regression model also gave us a WMAE score of 14808.15 on the training dataset and 14834.85 on the validation dataset, which is not that great.

```
Training dataset WMAE is 14808.15
Validation dataset WMAE is 14834.85
```

Decision Tree

Any tree in which each internal (non-leaf) node is labelled with an input feature is referred to as a decision tree. Each leaf of the tree has a category or probability distribution across the classes labelled on it, indicating that the data set has been classified by the tree into a particular class or probability distribution. weekly_sales have

been used in our project as the criteria to forecast future sales. We made use of the imported DecisionTreeRegressor() function. Training the model with the Decision Tree model gave us a WMAE score of 14808.15 on the training dataset and 1709.88 on the validation dataset. Using the decision tree greatly improved the WMAE score on the validation dataset.

```
Training dataset WMAE is 14808.15
Validation dataset WMAE is 1709.88
```

Random Forest

We used the Random Forest because it is not easily prone to overfitting and has less influence to the outlier data. a random forest is made up of numerous independent decision trees that work together as an ensemble. Every tree in the random forest spits out a class forecast, and the classification that receives the most votes become the prediction made by our model. We made use of the imported RandomForestRegressor() function. When we trained the model using the default parameters the results were similar to the Decision tree predictions. Then we introduced the hyper parameter tuning which greatly helped us to enhance the WMEA values for both the training and validation dataset.

We did tuning for the following parameters: n_estimators, min_samples_split, min_samples_leaf, max_samples. Training the model with the Random Forest model gave us a WMAE score of 542.21 on the training dataset and 1348.48 on the validation dataset.

```
Training dataset WMAE is 542.21
Validation dataset WMAE is 1348.38
```

Extra Trees

The additional trees approach, like the random forests technique, generates a large number of decision trees, but the sampling for each tree is random and without replacement. With distinct samples, this generates a dataset for each tree. Each tree additionally receives a predetermined number of randomly chosen features from the entire set of features. The choice of a splitting value for a feature is made randomly, which is both the most significant and distinctive feature of additional trees. The approach randomly chooses a split value for the data instead of finding a locally optimal value using Gini or entropy. The trees become diverse and uncorrelated as a result. We

made use of the imported ExtraTreesRegressor () function with 100 estimators. Training the model with the Extra Tree model gave us a WMAE score of 609.17 on the training dataset and 1562.77 on the validation dataset. Using the decision tree greatly improved the WMAE score on the validation dataset.

```
Training dataset WMAE is 609.17
Validation dataset WMAE is 1562.77
```

Evaluation Metric

The evaluation metric for model performance used in this project as specified by Kaggle is Weighted Mean Absolute Error (WMAE)**.** It is a measure used to evaluate the performance of regression or forecasting models.

Below is the formula to calculate the WMAE is –

$$\text{WMAE} = \frac{1}{\sum w_i} \sum_{i=1}^{n} w_i |y_i - \hat{y}_i|$$

- n is the number of rows
- ŷ, is the predicted sales
- y, is the actual sales
- w, are weights. w = 5 if the week is a holiday week, 1 otherwise

Weighted Importance calculation for Feature Selection

We use the function 'PermutationImportance' to calculate the weighted importance of each feature in the data frame. The basic algorithmic idea behind this function is that it calculates the weighted importance of each feature by calculating how much the score reduces if that feature is removed. We get this function from the "eli5" module.

V. EXPERIMENTAL RESULTS

Case Study: Hyperparamter Tuning

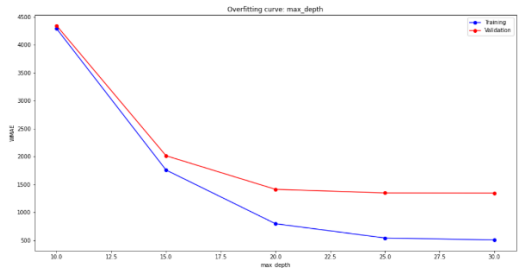
This is the method used to explore a range of possibilities to build a model architecture.

Also, the parameters which define the model architecture are referred to as hyperparameters and the process of searching for the ideal model architecture is called hyperparameter tuning.

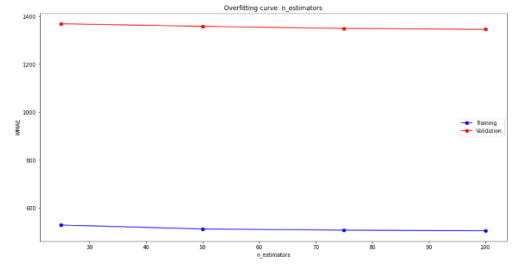
Hyperparameters are not model parameters, and they cannot be directly trained from the data. They are learned during training when we optimize a loss function using gradient descent.

Hyperparameter tunning depends on the results of different method used to determine a optimal values in order to evaluate the performance of each model. Hence, we used different methods.

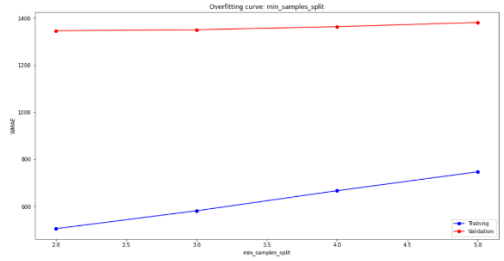
1. test_param_and_plot('max_depth', [10, 15, 20, 25, 30])



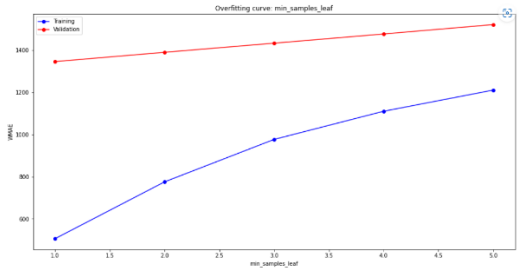
2. test_param_and_plot('n_estimators', [25, 50, 75, 100])



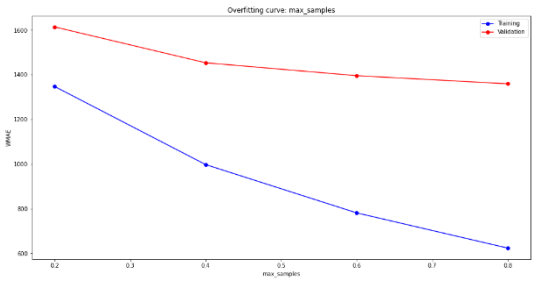
3. test_param_and_plot('min_samples_split', [2, 3, 4, 5])



4. test_param_and_plot('min_samples_leaf', [1, 2, 3, 4, 5])



5. test_param_and_plot('max_samples', [0.2, 0.4, 0.6, 0.8])



OVERALL PERFORMANCES

MODEL	TRAINING WMAE	TESTING WMAE
LINEAR REGRESSION	**14808.15**	**14834.85**
RIDGE REGRESSION	**14808.15**	**14834.85**
DECISION TREE	**14808.15**	**1709.88**
RANDOM FOREST	**542.21**	**1348.38**
EXTRA TREES	**609.17**	**1562.77**

Table 2

Overall Performances of all the models

Random Forest model displays the least WMAE values, followed by Extra Trees, making them the best feasible models for our problem case. Regression models and Decision Tree show very high WMAE, and hence we proceed to discard those models for out final submission.

Baseline Methods for Comparison

In this project we used various ML models such as, Linear regression, Ridge regression, Decision tree, Random Forest, and Extra trees. Among which Random Forest had the best performance. Here is why?

1. Random Forest compared to Linear regression: There are fewer number of parameters in a linear model when compared to random forest. Hence, the results are much more accurate compared to linear regression.

2. Random Forest compared to Decision Tree: The random forest can generalize the data in a better way. Hence, the results are much more accurate compared to decision tree.

3. Random Forest compared to Ridge regression: Random Forest works well when it comes to categorical data compared to Regression models.

4. Random Forest compared to Extra trees: Random Forest does an optimal split whereas extra tree does it randomly.

Strategy for Final Submission:

We have so far found out that Random Forest and Extra Trees are the 2 best methods in our case. Hence, we are going to blend these 2 models to get the best baseline predictions. We have made an average of prediction values from both these models and made the final submission. This has given us a pretty good accuracy score and has placed us in the top 5 percent of the competition.

Submission and Description	Private Score	Public Score	Selected
submission_rf_etr.csv Complete (after deadline) · 4h ago	2761.91997	2671.90774	☐

Related Work

In [6] and [7], regression models and gradient boosting models are being used for the sales prediction of stores. In our project, we implemented regression models, random forests, and extra trees in order to bring novelty to the project.

VI. Conclusion

In this project, the experiments which were performed are useful for stores sales prediction of Walmart. It can be used to predict sales based on the unemployment rate, the average temperature in the store region, fuel price in the store region, prices during holidays, and markdowns by using the data of the years 2010, 2011, and 2012. This can be used by retailers to forecast sales for the upcoming years.

As shown in the Results, after experimenting with Linear Regression, Ridge Regression, Decision Tree, Random Forest, and Extra Trees models, we observe that the Random Forest and Extra trees are the most effective methods for sales prediction. The predictions obtained by combining both the Random Forest and Extra Trees gave much better results. The Random Forest model achieved the lowest WMAE value of 542.21 on the training dataset, 1348.38 on the validation dataset, and 2902.45 for the final submission score on the Kaggle. However, the combined predictions of Random Forest and Extra Trees gave us a score of 2761.90 for the final submission score on the Kaggle.

VII. FUTURE SCOPE

We can implement deep learning models to further enhance our system because they will raise our accuracy. Neural networks are useful because they can cluster and classify raw data, find hidden patterns and correlations in it, and continuously learn and get better over time.

REFERENCES

[1] A. Krishna, A. V, A. Aich and C. Hegde, "Sales-forecasting of Retail Stores using Machine Learning Techniques," 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), 2018, pp. 160-166, doi: 10.1109/CSITSS.2018.8768765.

[2] Jain, Ankur, Manghat Nitish Menon, and Saurabh Chandra. “Sales Forecasting for Retail Chains.” (2015).

[3] https://corporate.walmart.com/newsroom/2010/02/18/walmart-reports-fourth-quarter-and-fiscal-year-2010-results#:~:text=With%20fiscal%20year%202010%20sales,than%202.0%20million%20associates%20worldwide.

[4] https://corporate.walmart.com/newsroom/2011/02/22/walmart-reports-fourth-quarter-eps-from-continuing-operations-of-1-41-underlying-eps-from-continuing-operations-of-1-34-exceeds-consensus-and-company-guidance

[5] https://corporate.walmart.com/newsroom/2012/02/21/walmart-reports-q4-eps-from-continuing-operations-of-1-51-walmart-u-s-delivers-positive-traffic-and-positive-comp-sales-in-q4

[6] https://www.macrotrends.net/stocks/charts/WMT/walmart/revenue

[7] https://www.statista.com/statistics/1172941/walmart-year-over-year-sales-growth/#:~:text=From%202022%20Walmart%20was%20predicted,the%20positive%20trend%20until%202026.

GITHUB CODE LINK: https://github.com/vardhanmunagala/ML_Project