<Walmart Sales Forecasting>

<CIND820 Capstone Project

By: Hasib Chowdhury (500439135)

Supervisor: Dr. Uzair Ahmad >

**GitHub Link: https://github.com/Hasib147/CIND820-Project**

Table of Contents

[Revised Abstract 3](#_Toc452457337)-4

[Introduction](#_Toc452457340) 5

[Literature Review 6-8](#_Toc452457341)

[Data Description 9-11](#_Toc452457341)

[Exploratory Data Analysis 12](#_Toc452457337)-13

[Modelling & Evaluation 14-20](#_Toc452457337)

[K-Fold Cross Validation 21](#_Toc452457337)

[Random Forest for regression 22](#_Toc452457337)-23

[Random Forest for comparing 2 stores 24](#_Toc452457337)-25

[Effectiveness, Efficiency, Stability 26-29](#_Toc452457337)

[References 30](#_Toc452457337)

# Revised Abstract (CIND820 Capstone Project)

For this project, I will use the Walmart Sales Forecasting dataset from the Kaggle website (<https://www.kaggle.com/code/aslanahmedov/walmart-sales-forecasting/data>) to conduct the research to fulfill the “Data Analytics Project (CIND820)” requirements. The data that will be used are the clean\_data.csv file from the site, which takes into account many different open data sources from Kaggle involving Walmart sales forecast.

This dataset has 420,213 rows with 23 columns (attributes). In total, there are 45 different Walmart stores with about 100 departments in each of 3 different types (A,B,C) and different sizes which will be analyzed throughout the project.

The data is provided on a weekly basis from February 2010 to October 2012, over a 2.5-year period, and each week has its own set of features such as different CPI (consumer price index), unemployment rate as well as whether a holiday was a factor in its performance.

The main themes for this project will be stepwise regression and random forest (whether the Walmart was successful in sales or not or any other factor), as well time-series in terms of predictive analytics.

The 2 main questions that will be investigated for this project are how the sales have been in that week if there was a specific holiday during that week such as Christmas, Thanksgiving, New Years, etc. and also whether regression or random forest model that we can use to improve the sales for a better outcome for Walmart in terms of sales forecasting.

One other thing that will be examined is to predict the store’s sales at a particular week and if there are any impacts on it.

The regression methods will be employed to estimate the future sales of the store, and time series forecasting to see whether there is a trend in the data from February 2010 to October 2012 or specific time periods in between the years.

In addition, I will also compare the 2 models of random forest for regression between the 2 stores that have the highest & lowest total weekly sales from 2010 to 2012 (totalling all the departments in each store)

The software tool that will be used for this project will be R.

# Introduction

For this project, I decided to work on the Walmart Kaggle dataset for sales forecasting throughout its many stores in certain regions as there are 45 different stores and about 90-100 departments in each specific location. In addition, this dataset takes into account several different variables including weekly sales (which is the target variable) within the time frame from February 2010 to October 2012 and whether certain holidays (such as Christmas, Thanksgiving, etc.) have a factor in terms of sales forecasting for the big box corporation. In this project, I am going to analyze the dataset that is given in further detail and predict how the sales forecast will be like in the future years and also to see if the weekly sales at particular locations are impacted by either time-based (during different seasons such as spring, summer, winter) or space-based factors (how big a particular location is) affect the sales.

# Literature Review:

The literature review for the Walmart sales forecasting dataset will consist of regression and k-fold cross validation for sales forecasting to predict the outcomes for Walmart in the near future in terms of its weekly sales, it will also focus on random forest for time series. There are several seasons that affect sales in a given time of year that are significantly more or less than averages depending on a variety of factors. In this scenario, if Walmart is unaware about the seasons (eg Spring, Summer, Fall, Winter), it can lose too much money which in effect can cause low profit for the company. Predicting future sales is one of the most crucial plans for a big company such as Walmart since sales forecasting gives us an idea to the company for arranging stocks, calculating revenue, and deciding to make a new investment.

In an article by Mentzer, J. T., & Cox, J. E.(1984), it states that “Sales Forecasting is quickly becoming one of the most crucial aspects of planning for companies and that a survey of 175 midwestern businesspeople indicated that 65% thought sales forecasting was very important to their company’s success and an additional 28% said forecasting was important, although not critical” (Mentzer et al, p. 27). This article from the 1980’s tells us that Walmart’s (being a big corporation) sales forecasting is one of the most important aspects in terms of sales, especially in the United States since there are more than 5,000+ locations in this country and over 6,000 locations worldwide.

“Forecasts have traditionally served as the basis for planning and executing supply chain  
activities. Forecasts drive supply chain decisions, and they have become critically important  
due to increasing customer expectations, shortening lead times, and the need to manage  
scarce resource” (Boone, T. et. al, p. 170)

In terms of this data, Walmart is a big corporation that sells several different products throughout North America and Worldwide. It also operates a chain of hypermarkets, in this scenario Walmart has provided data (in csv format) with a combination of 45 different stores (ranging from 3 different types – A,B,C as well as different sizes varying from 34,000 square feet all the way up to 200,000+ square feet) including store information and monthly sales. This data is provided on weekly basis, going back from February 2010 all the way to July 2013. Walmart tries to find the impact of holidays on the sales of store depending on the different time of year (eg. Spring, summer, fall, winter) eg. holidays such as Christmas, Thanksgiving, Super bowl, Labor Day, Family day, etc.

“Firms use product sales forecasting as a foundation to estimate sales revenue and make decisions regarding production, operation and marketing strategies” (Fan, Z. et. al, p. 90)

“Through product sales forecasting, firms can create a plan for marketing, sales management, production, procurement, logistics and so on to improve their economic benefits and reduce losses caused by weaknesses in the production plan” (Fan, Z. et al, p. 90).

In this scenario, Walmart can maintain its products in a yearly fashion and make good marginalized profit when it comes to their certain operations and marketing strategies in certain locations. They make good money off customers in things like electronics, furniture, etc. especially during the end of year during the Christmas holidays and black Friday/boxing day.

In August 2019 article study, it showed that “Walmart Inc. second-quarter sales came in ahead of market forecasts however, the corporation saw income fall by a third amid shrinking margins and rising costs. The world’s biggest retailer posted revenues of USD $138.6 billion for the quarter, 0.6% growth year-on-year and above analysts predictions of around USD $136.9 billion.” (Proactiveinvestors, Aug. 2021). This most likely occurred to inflation happening within the US economy as certain states have higher inflation than others but it still maintained to have over $100 billion in revenue over 3 months of the year in 2019. The sales more or less may have fluctuated afterwards in 2020 due to the COVID-19 pandemic as certain states were under restrictions.

In a challenge done on Kaggle in 2020, “The challenge was to predict future sales of Walmart products based on past sales. The competition was organised in two parallel challenges, one was a 28-day challenge and the other one was series of quantile estimates for the same period” (de Rezende, R., et al., p. 1) This study shows how the company did in that specific month in terms of the sales that it received from the consumers. Their “approach was conceived mainly to model product-store sales, as these are the most relevant for supply-chain decisions” (de Rezende et al, p. 1). This tells us how different Walmarts in different regions across the country do in terms of sales when it comes to supply-chain inquiries, the Walmart Kaggle dataset is consistent with this relation, based on the 3 years of data that is given.

# Data Description:

There are a total of 23 attributes in this dataset. The attributes in the Walmart sales forecast are as follows:

1. **Store:** This is the first column of the dataset and is numbered between 1-45, indicating the 45 different Walmart stores in the region (ranging from types A,B,C)
2. **Department:** This is the second column of the dataset and is numbered between 1-98, indicating the 98 different departments in each store.
3. **Date:** This is the third column of the dataset and is in the time span between February 2010 to October 2012 (Friday of every week).
4. **Weekly Sales:** This is the fourth column of the dataset, this is the target variable of the dataset since it tells us how much each store (along with the department made) in that particular week.
5. **IsHoliday:** This is the fifth column of the dataset and is based on True/False classification value based on the specific week in which there was a holiday (eg. Christmas, New Year’s).
6. **Temperature:** This is the sixth column of the dataset and represents the temperature on that specific day, the values range from -2 degrees to 100 degrees.
7. **Fuel Price:** This is the seventh column of the dataset and represents the fuel price on that specific day, the values range from 2.472 to 4.468.
8. **Markdowns 1-5:** These are columns 8 to 12 in the dataset, these 5 variables represents the promotional markdowns that Walmart is running during that specific week.
9. **CPI (Consumer Price Index):** This is column 13 of the dataset and values range from 126 to 227.
10. **Unemployment:** This is column 14 of the dataset and tells us the unemployment rate in that particular Walmart store, and the values range from 3.879 to 14.313
11. **Type:** This is column 15 of the dataset and tells us whether it’s a type A, B or C Walmart in that particular region.
12. **Size:** This is column 16 of the dataset and tells us how big that particular store is, values range from 34,875 to 219,622 .
13. The next 4 columns are the 4 different holidays or the biggest season for sales, being Super Bowl weekend, Labor Day, Thanksgiving, Christmas
14. **Week:** This is column 21 of the dataset and tell us the specific week of that year the Walmart made sales in, values start from early January (week 1) to late December (week 52)
15. **Month:** This is column 22 of the dataset and tells us the specific month, varying from month1 (January) to month 12(December) of that particular year.
16. **Year:** This is the last column(column 23) and tells us the year of the sales in the Walmart, this dataset spans 3 years (2010, 2011, and 2012)

The Summary of the dataset is as follows:

A close-up of a document

Description automatically generated with medium confidence

# Exploratory Data analysis:

For this Walmart Sales forecasting model, the data is consistent with the variables, the following table shows a summary of the values in the original dataset:

A close-up of a document

Description automatically generated with medium confidence

Out of the 23 variables in the dataset, about 8 of them are categorical and the rest are numerical including the target variable (weekly sales) which we will use to predict the sales of Walmart for that week. This dataset as mentioned earlier in the literature review & data description have over 420,000+ rows and 23 columns over the 2.5 year time frame (2010 to 2012).

The following is the summary of the normalized dataset (normalized only numerical values):

A close-up of a document

Description automatically generated

# Modelling:

The model that I have decided to use for this dataset is the stepwise regression model, which I think will be the right fit for this project. There are 2 approaches to this model, one is the forward selection and the other is the backward elimination algorithm. The forward selection approach starts with no variables and adds each new independent variable one by one depending on the values of the AIC (goes from large values to small values depending on the model’s input variables) testing for statistical significance. The backward elimination method begins with a full model loaded with multiple independent variables and then removes one independent variable at a time to test its statistical significance to overall results depending on the value of the AIC (goes from small to large, opposite to the forward selection).

**Screenshot of the original dataset model:**

Forward Selection:

Text, table

Description automatically generated with medium confidence

Backward Elimination:

Table

Description automatically generated

**Screenshot of the normalized dataset model:**

Forward Selection:

Text

Description automatically generated

Backward Elimination:

Text, table

Description automatically generated

**Evaluation:**

The stepwise regression model for the Walmart dataset is the following:

For forward selection: (original dataset – CIND820\_clean\_data):

Weekly\_Sales = 0.08945\*Size + 0.1275\*MarkDown3 - 19.66\*CPI + 30.57\*Temperature -240.1\*Unemployment + 0.07627\*MarkDown5 - 748.6\*Fuel\_Price + 0.02755\*MarkDown4 + 0.01738\*MarkDown2 + 9526

For Backward Elimination: (original dataset – CIND820\_clean\_data):

Weekly\_Sales = 30.57\*Temperature - 748.6\* Fuel\_Price - 19.66\*CPI -240.1\*Unemployment 0.08945\*Size + 0.01738\*MarkDown2 + 0.1275\*MarkDown3 + 0.02755\*MarkDown4 + 0.07627\*MarkDown5 + 9526

For forward selection: (normalized dataset – CIND820\_clean\_data2):

Weekly\_Sales = 0.08945\*Size + 0.1275\*MarkDown3 - 19.66\*CPI + 30.57\*Temperature -240.1\*Unemployment + 0.07627\*MarkDown5 - 748.6\*Fuel\_Price + 0.02755\*MarkDown4 + 0.01738\*MarkDown2 + 0.388

For Backward Elimination: (normalized dataset – CIND820\_clean\_data2):

Weekly\_Sales = 30.57\*Temperature - 748.6\* Fuel\_Price - 19.66\*CPI -240.1\*Unemployment 0.08945\*Size + 0.01738\*MarkDown2 + 0.1275\*MarkDown3 + 0.02755\*MarkDown4 + 0.07627\*MarkDown5 + 0.3888

Markdown1 is redundant (not considered in the model) and markdowns 2-5 are more relevant and observed in the intercept as well as the whole dataset. Size is the most important variable due to the values and might have more departments in specific locations to attract more clients and also the products/services that location may have.

The following is the barplot between the weekly sales and the isholiday, Christmas, Thanksgiving, Super Bowl, and Labor Day

Chart

Description automatically generated

A picture containing square

Description automatically generated

Square

Description automatically generated with medium confidence

Square

Description automatically generated with medium confidence

Square

Description automatically generated with medium confidence

By looking at the above barplots, this tells us that during the holidays (such as thanksgiving, Christmas, labor day) there are more or less sales compared to the other days of the year. Out of the entire dataset of 420,000+ rows, there were approximately 30,000 rows that had a holiday in it (approximately 7% of the dataset).

# K-Fold Cross Validation:

**Cross-validation for the original dataset:**

Graphical user interface, text, email

Description automatically generated

From the R-squared value above, this tells us the error value is low and also 378,000 rows out of the 420,000 were taken into account when doing the 10-fold cross validation (9:1 train-test split)

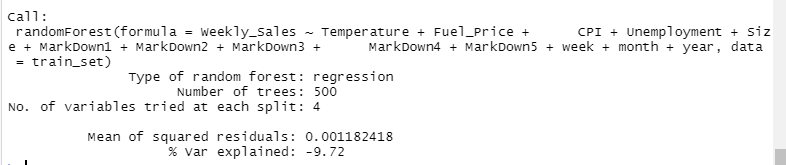
**Cross-validation for the normalized dataset:**

Text

Description automatically generated

The cross-validation for the normalized dataset used the same number of sample sizes compared to the original and what was observed was that the R-squared value (measure of the correlation between the predictions made by the model and the actual observations) remained the same, however the RMSE (root mean square error) has dropped in value which tells us that the model can be more closely related to the actual observation (this was done through trial & error using k-fold and the accuracy of the result may be off from the predicted values)

# Random forest for regression:



The value of the mean of squared residuals is 0.001182418, which tells us the data is consistent with all the other variables (for 7000 training set and 3000 testing set). However when the training data changes to 8000 for training set and 2000 for testing set, it changes to:

Text

Description automatically generated

The mean of squared residuals is about the same however the % var explained decreases.

The following is the code used to get the random forest for regression:

# 2 models to predict weekly sales for the best and worst Walmart stores in the region using random forest regression

store\_sales <- aggregate(train\_set$Weekly\_Sales, list(train\_set$Store),sum)

names(store\_sales)<-c("store","Total\_Sales")

str(store\_sales) # gives us 45 different stores of the total sales, combining all the departments

min(store\_sales$Total\_Sales) # 0.8388226

max(store\_sales$Total\_Sales) # 7.677282

which(store\_sales$Total\_Sales==min(store\_sales$Total\_Sales))

#Store 44 has the lowest total weekly sales out of the 45 stores at 0.8388226

which(store\_sales$Total\_Sales==max(store\_sales$Total\_Sales))

#Store 20 has the highest total weekly sales out of the 45 stores at 7.677282

nrow(CIND820\_clean\_data2[CIND820\_clean\_data2==44,]) # 18463 total rows

nrow(CIND820\_clean\_data2[CIND820\_clean\_data2==20,]) # 24965 total rows

# Converting store 44 categorical variables into factors

store44<-CIND820\_clean\_data2[CIND820\_clean\_data2==44,]

store44$week<-as.factor(store44$week)

store44$month<-as.factor(store44$month)

store44$year<-as.factor(store44$year)

str(store44)

# Converting store 20 categorical variables into factors

store20<-CIND820\_clean\_data2[CIND820\_clean\_data2==20,]

store20$week<-as.factor(store20$week)

store20$month<-as.factor(store20$month)

store20$year<-as.factor(store20$year)

str(store20)

# Random forest for comparing the 2 stores (best and worst in terms of weekly sales):

The 2 stores with the highest and lowest total weekly sales were store 44 (lowest weekly sales) and store 20 (highest weekly sales). We first had to remove the ‘NA’ values from both the datasets as an error kept appearing. Store 20 had a total of 10,173 total rows taken into account when considering this model from its 81 different departments (listed from 1 to 98 with some missing values) from February 2010 to October 2012, whereas Store 44 also had 81 departments from February 2010 to October 2012 however 7,142 total rows were considered (this may have thrown off the results). However when we broke this up into further when considering 70% training and 30% testing, this took into account 5,000 rows of the store 44 train set and 7,121 rows of the store 20 train set. The following is the output:

Text

Description automatically generated

As for results, they both gave very similar results as they both have regression as the type of random forest and both have the same number of tree as well as the # of variables at each split. However the % var explained is a bit lower in store 44 train set compared to store 20 train set, and the mean of squared residuals is slightly higher in store 20 compared to store 44.

# Effectiveness, Efficiency, Stability:

**Effectiveness:**

The model that performed well was the 7000 training data and 3000 testing data as the value of the RMSE is small for both (0.1512367 and 0.1512731). If its small there is a smaller percentage error and the model is accurate for both models. This was taken into account when the training data was 7000 rows out of the total 420,000 rows and 3000 rows for testing data out of the 420,000.

**Efficiency:**

Comparing the 2 best and worst stores (stores 44 and 20), when using the Sys.time() command to compare the time it took to run the random forest regression model, it took about 45 seconds for store 44 train set (the worst store) for the model to run on R. As for store 20 (the best store), it took about 2 minutest for store 20 train set (the best store) for the model to run on R. Please note: this was done by trial and error as it depended on the computers memory.

However when it came to the individual stores (best and worst stores – stores 20 & 44). The random forest for store 44 when there was 7000 training data entries, it took 2.5 minutes for the model to run, however when the value increased to 8000 training data entries, it took a bit longer at approximately 3.5 minutes as well as store 33 being considered for the lowest weekly sales. Same thing happened for store 20, the model originally took about 4.5 minutes for running 7000 data entries, this increased to 5.5 minutes when it increased to 8000 data entries, this is due to the fact that more data was collected to build this model.

**Stability:**

When the training data changes from 7000 data entries to 8000 data entries and testing data changes from 3000 to 2000 data entries (keeping the total amount to 10000 data entries) the output then changes when it comes to the best and worst stores, originally store 44 had the lowest weekly sales out of all the stores but this changes to store 33 when 8000 values are considered instead of 7000, this may be due to the fact that more data is being collected. This results in a slight change in the accuracy since a different store has the lowest weekly sales.

* 10 RMSE values for samples (different training data) – different samples 10 times

|  |  |  |  |
| --- | --- | --- | --- |
| **Train data** | **Test Data** | **Train RMSE** | **Test RMSE** |
| 1000 | 9000 | 0.1496976 | 0.151304 |
| 2000 | 8000 | 0.1516118 | 0.1512477 |
| 3000 | 7000 | 0.1539014 | 0.1513073 |
| 4000 | 6000 | 0.1515423 | 0.1513018 |
| 5000 | 5000 | 0.1503784 | 0.15128 |
| 6000 | 4000 | 0.1491193 | 0.1513039 |
| **7000** | **3000** | **0.1512367** | **0.1512731** |
| 8000 | 2000 | 0.1528445 | 0.1512889 |
| 9000 | 1000 | 0.151213 | 0.1512795 |
| 10000 | 0 | 0.1492315 | NaN |

From the above table, the data for both train and test RMSE is consistent with one another, the values range from 0.149 to 0.153 for the train RMSE and its about the same for all the testing values for the test RMSE (with the exception if 0 values were tested which indicated that its not possible to have a RMSE). The best model when it came to comparing the 2 RMSE values would be the 7000 training data and 3000 testing data, because they are very close to one another (0.1512367 compared with 0.1512731, a difference of 0.0000364) which tells us that this is the best model on the list in terms of the accuracy.

In terms of predictions and how long it takes to do that, the evaluation metric I used was the Mean Absolute Percentage Error (or MAPE). The lower the MAPE value, the better the forecasting model. when using the command for the 2 stores, the following commands were used:

library(MLmetrics)

# Predicted values for store 20

predictions\_train\_set\_store20 = predict(random\_forest\_store20, data = store20\_train\_set)

MAPE(store20\_train\_set$Weekly\_Sales, predictions\_train\_set\_store20)

predictions\_test\_set\_store20 = predict(random\_forest\_store20, data = store20\_test\_set)

MAPE(store20\_test\_set$Weekly\_Sales, predictions\_test\_set\_store20[1:nrow(store20\_test\_set)])

# Predicted values for store 44

predictions\_train\_set\_store44 = predict(random\_forest\_store44, data = store44\_train\_set)

MAPE(store44\_train\_set$Weekly\_Sales, predictions\_train\_set\_store44)

predictions\_test\_set\_store44 = predict(random\_forest\_store44, data = store44\_test\_set)

MAPE(store44\_test\_set$Weekly\_Sales, predictions\_test\_set\_store44[1:nrow(store44\_test\_set)])

The outputs are 0.8895105, 0.8524784, 1.222545, and 1.141424 respectively. For store 20, the training and testing are very close to one another which tells us the model is quite accurate (difference of approximately 0.04), however for store 44 the values are above 1, and it has a difference of 0.08 which tells us that the data may not be as accurate due to the normalized values.

This tells us that the similarity in results over the training and testing data sets for both stores is one of the indicators to suggest that the random forest model is robust and generates significant results.

# References:

*Walmart Sales Forecasting.* Kaggle. Retrieved July 4, 2022 from <https://www.kaggle.com/code/aslanahmedov/walmart-sales-forecasting> <-- where the dataset is obtained

Proactiveinvestors. (2021). *Proactive Investors: Walmart sales beat forecasts but profits drop by a third*. Newstex.

Boone, T., Ganeshan, R., Jain, A., & Sanders, N. R. (01/2019). *Forecasting sales in the supply chain: Consumer analytics in the big data era* Elsevier. doi:10.1016/j.ijforecast.2018.09.003

Fan, Z., Che, Y., & Chen, Z. (05/2017). *Product sales forecasting using online reviews and historical sales data: A method combining the bass model and sentiment analysis* Elsevier. doi:10.1016/j.jbusres.2017.01.010

Mentzer, J. T., & Cox, J. E. (01/1984). *Familiarity, application, and performance of sales forecasting techniques* Elsevier. doi:10.1002/for.3980030104

de Rezende, R., Egert, K., Marin, I., & Thompson, G. (12/2021). *A white-boxed ISSM approach to estimate uncertainty distributions of walmart sales* Elsevier. doi:10.1016/j.ijforecast.2021.11.006

*K-Fold Cross Validation in R (Step-by-Step).* Statology (November 4, 2020). Retrieved June 27, 2022 from <https://www.statology.org/k-fold-cross-validation-in-r/> .

*Machine Learning for Time Series Data in R.* Decision Science Analytics. (May 12, 2020). Retrieved July 4, 2022 from https://www.pluralsight.com/guides/machine-learning-for-time-series-data-in-r