

**SVKM’s NMIMS**

**Mukesh Patel School of Technology Management & Engineering**

**WALMART STORE’S SALES FORECASTING**

Project report



SUBMITTED BY:

ANSHOOL AGARWAL

SHUBHANSHU BHAGAT

ADITYA BILLORE

SATVIK CHOUDHARY

PRANAV KOLHE

MPSTME, SHIRPUR CAMPUS

JUNE 2021

TABLE OF CONTENTS

1. BACKGROUND
2. INTRODUCTION
3. AIM
4. DATA
5. TECHNOLOGIES
6. SYSTEM
   1. REQUIREMENTS
   2. IMPLEMENTATION
   3. VISUALS & OUTPUT
7. RESULT
8. CONCLUSION
9. FUTHER DEVELOPMENT
10. REFERENCES

**1. Background**

1.1 Introduction

Predicting future sales for a company is one of the most important aspects of strategic planning. Walmart is one of the largest retailers in the world and it is very important for them to have accurate forecasts for their sales in various departments. Every Departmental store chain like Walmart wants to predict the store sales in the nearby future so that inventory planning can be done. Along with that, sales prediction helps to increase/decrease store staff based on the rush (More sales can mean more customers are coming to the stores). Also, it is always a good idea to do sales and revenue forecasting to better understand the company's cash-flows and overall growth.

For inventory planning, you also need to know what products (or category of products aka department) will be utilised more. Under-stock some products and your sales are hit. Over-stock items like perishables and you run into losses if the product expires. That's why the sales prediction is done at a combination of store and department level (and sometimes even at product level for high-selling products).

Since there can be many factors that can affect the sales for every department, it becomes imperative that we identify the key factors that play a part in driving the sales and use them to develop a model that can help in forecasting the sales with some accuracy.

1.2 Aim

The objective of our project is to develop a working model which has the capability of predicting the total sales as well as giving an estimate figure of sales on 4 major holidays celebrated predominantly in the U.S. The statistical model should not only predict estimates accurately for the historical data provided by Walmart, but also on any recent data.

1.3 Data

The project uses the dataset given by the Walmart Inc. itself. The data collected ranges from 2010 to 2012, where 45 Walmart stores across the country were included in this analysis. Each store contains several departments, and we are tasked with predicting the department-wide sales for each store. It is important to note that we also have external data available like CPI, Unemployment Rate and Fuel Prices in the region of each store which, hopefully, helps us to make a more detailed analysis.

This data set is available on the kaggle website. These data sets contained information about the stores, departments, temperature, unemployment, CPI, is Holiday, and MarkDowns.

*Stores:*

Store: The store number. Range from 1–45.

Type: Three types of stores ‘A’, ‘B’ or ‘C’.

Size: Sets the size of a Store would be calculated by the no. of products available in the particular store ranging from 34,000 to 210,000.

*Features:*

Temperature: Temperature of the region during that week.

Fuel Price: Fuel Price in that region during that week.

MarkDown1:5: Represents the Type of markdown and what quantity was available during that week.

CPI: Consumer Price Index during that week.

Unemployment: The unemployment rate during that week in the region of the store.

*Sales:*

Date: The date of the week where this observation was taken.

Weekly Sales: The sales recorded during that Week.

Dept: One of 1–99 that shows the department.

IsHoliday: a Boolean value representing a holiday week or not.

1.4 Technologies/Concepts used:

*Machine Learning (ML):*  Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

*R (Programming Language):* R is a programming language and free software environment for statistical computing and graphics supported by the R Foundation for Statistical Computing. The R language is widely used among statisticians and data miners for developing statistical software and data analysis.

*RStudio:* RStudio is an Integrated Development Environment (IDE) for R.

*Predictive Analytics:* Predictive analytics is an area of statistics that deals with extracting information from data and using it to predict trends and behaviour patterns. It is the use of data, statistical algorithms and machine learning techniques to identify the likelihood of future outcomes based on historical data.

**2. System**

2.1 Requirements:

*I. Hardware Requirements*

* Processor i5 or equivalent
* Hard Disk
* Memory
* Monitor

*II. Software Requirements*

* Windows 10 or higher
* R programming Language (version 4.0.5 or higher)
* R Studio
* MS Excel

2.2 Implementation (Source code):

rm(list=ls(all=TRUE))

gc(reset=TRUE)

##Loading Libraries

library(dplyr)

library(broom)

library(forecast)

library(readr)

library(tidyr)

library(lubridate)

library(glmnet)

library(gbm)

library(xts)

library(ggplot2)

library(tidyverse)

#Importing Data

#Importing data and changing data types to the desired types.

setwd("C://Users\BegyaptFayon//Desktop//IBM//wallmart sales forecast//intermediate")

store\_raw <- read\_csv("stores.csv")

feature\_raw <- read\_csv("features.csv")

train <- read\_csv("train.csv")

test\_final <- read\_csv("test.csv")

# glimpse(store\_raw)

# glimpse(feature\_raw)

# glimpse(train)

# glimpse(test\_final)

store\_raw$Store <- as.factor(store\_raw$Store)

train$Store <- as.factor(train$Store)

train$Dept <- as.factor(train$Dept)

test\_final$Store <- as.factor(test\_final$Store)

test\_final$Dept <- as.factor(test\_final$Dept)

feature\_raw$Store <- as.factor(feature\_raw$Store)

#Exploring Data

#1. How many stores are present in data?

train %>% summarize(Total\_stores = n\_distinct(Store))

#2. How many departments are present in data?

train %>% summarize(Total\_Dept = n\_distinct(Dept))

#3. How many store-department combinations have all weeks of sales data?

train %>% summarize(min\_date = min(Date), max\_date = max(Date), total\_weeks = difftime(min\_date,max\_date, unit = "weeks"))

train %>% group\_by(Store, Dept) %>% summarize(count\_wk = n\_distinct(Date)) %>% ggplot(aes(x = count\_wk)) + geom\_histogram()

#Distribution of stores by size and type Type ‘A’ is the larget format followed by Type ‘B’ and ‘C’. Also, most of the stores in data belong to ‘A’ and ‘B’ type.

ggplot(store\_raw, aes(x = Size)) + geom\_histogram(binwidth = 25000) + facet\_grid(Type~.

store\_raw %>% group\_by(Type) %>% summarize(n()) %>% rename(`Number of Stores` = `n()`)

#Distribution of sales by type Sales for type A and B is normally distributed with left tail being skewed. Hence outlier treatment has to be performed on sales.

train %>% left\_join(store\_raw, by = "Store") %>% ggplot(aes(x = Weekly\_Sales)) + geom\_histogram() + facet\_grid(Type~.) + scale\_x\_log10()

#Plotting sales by store type to check trend and important peaks/holidays seasonality.

train\_all\_factors <- train %>% left\_join(store\_raw, by = "Store") %>% left\_join(feature\_raw, by = c("Store", "Date")) %>% mutate(first\_wk = ifelse(mday(Date) <= 7, TRUE, FALSE))

train\_all\_factors %>% group\_by(Type, Date) %>% summarize(sales = sum(Weekly\_Sales)) %>% ggplot(aes(x = Date, y = sales, col = Type)) + geom\_line() + scale\_y\_log10() + scale\_x\_date(date\_labels = "%Y-%m-%d", date\_breaks = "4 weeks") + theme(axis.text.x = element\_text(angle = 90, hjust = 1))

train\_all\_factors %>% group\_by(Date,first\_wk) %>% summarize(sales = sum(Weekly\_Sales)) %>% ggplot(aes(x = first\_wk, y = sales)) + geom\_boxplot()

train\_all\_factors %>% ggplot(aes(x = CPI, y = Weekly\_Sales)) + geom\_point() + geom\_line(aes(y= fitted(lm(Weekly\_Sales~CPI, data = train\_all\_factors))), colour = "red")

train\_all\_factors %>% ggplot(aes(x = Unemployment, y = Weekly\_Sales)) + geom\_point() + geom\_line(aes(y= fitted(lm(Weekly\_Sales~Unemployment, data = train\_all\_factors))), colour = "red")

#Is there a relation between average temperature of the week and sales?

train\_all\_factors %>% ggplot(aes(x = Temperature, y = Weekly\_Sales)) + geom\_point() + geom\_line(aes(y= fitted(lm(Weekly\_Sales~Temperature, data = train\_all\_factors))), colour = "red")

#DATA CLEANING

#Missing CPI treatment CPI is missing for few dates in test period from 3rd May 2013 to 26th July 2013. Checking trend of CPI for few random stores.

feature\_raw %>% filter(Store == c(1,25,33,41)) %>% ggplot(aes(x = Date, y = CPI, col = Store)) + geom\_line()

#CPI seems to follow a good trend which can be forecasted using time series method. Thus, forecasting CPI using auto ARIMA for the missing weeks. The plot of actual along with forecast looks fine for random store.

#Creating Exhaustive table with all weeks and store combination and joining feature to get CPI and Unemployement exhaustive data set to implement time series forecast for missing 2013 data

CPI\_Unemp\_exh <- feature\_raw %>% select(Store, Date) %>% complete(Store, Date) %>% left\_join(feature\_raw, by = c("Store","Date")) %>% select(Store, Date, CPI, Unemployment)

# Store list to put in loop

store\_lst <- distinct(CPI\_Unemp\_exh,Store)

# Dates to be forecasted

date\_fore <- CPI\_Unemp\_exh %>% filter(is.na(CPI)) %>% select(Date) %>% distinct() %>% arrange()

# Function to time series forecast CPI by store

forecast <- NULL

for (i in unique(CPI\_Unemp\_exh$Store)) {

CPI\_ts <- CPI\_Unemp\_exh %>% filter(Date<= "2013-04-26", Store == i) %>% select(CPI) %>% ts(start = c(2010,2,5), end = c(2013,4,26), frequency = 52)

fit <- auto.arima(CPI\_ts)

frcst <- as.data.frame(forecast(fit,13)) %>% select(`Point Forecast`) %>% rename(CPI = `Point Forecast`)

fore <- merge(i,date\_fore, all = T) %>% cbind(frcst)

forecast <- rbind(forecast, fore)

}

plot(forecast(fit,13))

#Missing Unemployment treatment Unemployment is missing for few dates in test period from 3rd May 2013 to 26th July 2013. Checking trend of Unemployment for few random stores.

feature\_raw %>% filter(Store == c(1,39,33,44)) %>% ggplot(aes(x = Date, y = Unemployment, col = Store)) + geom\_line()

# Function to forecast Unemployment

forecast\_Un <- NULL

for (i in unique(CPI\_Unemp\_exh$Store)) {

Unemp\_ts <- CPI\_Unemp\_exh %>% filter(Date<= "2013-04-26", Store == i) %>% select(Unemployment) %>% ts(start = c(2010,2,5), end = c(2013,4,26), frequency = 52)

fit <- auto.arima(Unemp\_ts)

frcst <- as.data.frame(forecast(fit,13)) %>% select(`Point Forecast`) %>% rename(Unemployment = `Point Forecast`)

fore <- merge(i,date\_fore, all = T) %>% cbind(frcst)

forecast\_Un <- rbind(forecast\_Un, fore)

}

plot(forecast(fit,13))

# Combining both the forecast

forecast\_CPI\_Un <- inner\_join(forecast, forecast\_Un, by = c("x", "Date")) %>% rename(Store = x)

# Joining it with feature\_raw and substituting the missing values with forecast

feature\_raw1 <- feature\_raw %>% left\_join(forecast\_CPI\_Un, by = c("Store","Date")) %>% mutate(CPI = ifelse(is.na(CPI.x),CPI.y,CPI.x),Unemployment = ifelse(is.na(Unemployment.x),Unemployment.y,Unemployment.x)) %>%

select(-Unemployment.x, -Unemployment.y, -CPI.x, -CPI.y)

#Weekly Sales Outlier Treatment

##The weekly sales outlier treatment is done for non-holiday non-markdown weeks as the peak in salaes due to events should not be capped. The outlier treatment is done using IQR method i.e. any sales below or above range defined by 25th Percentile - 1.5XIQR and 75th Percentile + 1.5XIQR is capped. 2.19% of the observations were capped in the training dataset.

feature\_fi <- feature\_raw1 %>% mutate\_at(c("MarkDown1","MarkDown2","MarkDown3","MarkDown4","MarkDown5"),funs(ifelse(.<0 | is.na(.),0,.))) %>% mutate(mkdn\_flag = (MarkDown1 !=0 | MarkDown2 !=0 | MarkDown3 !=0 | MarkDown4 !=0 | MarkDown5 !=0 ), mkdn\_hol = IsHoliday | mkdn\_flag)

train\_all\_fac <- train %>% left\_join(feature\_fi, by = c("Store","Date"))

# Treating outliers outside 1.5\*IQR range at store-dept level whenever its not holiday or no markdown given

train\_outlier\_treated <- train\_all\_fac %>% group\_by(Store, Dept) %>% mutate(perc\_25 = quantile(Weekly\_Sales,0.25), perc\_75 = quantile(Weekly\_Sales,0.75), iqr\_sales = IQR(Weekly\_Sales), Wkly\_sales\_treated = ifelse(Weekly\_Sales<perc\_25 - 1.5\* iqr\_sales & !mkdn\_hol, perc\_25 - 1.5\* iqr\_sales, ifelse(Weekly\_Sales > perc\_75 + 1.5\* iqr\_sales & !mkdn\_hol, perc\_75 + 1.5\* iqr\_sales,Weekly\_Sales)))

# Percentage of outliers

paste(round(mean(train\_outlier\_treated$Weekly\_Sales != train\_outlier\_treated$Wkly\_sales\_treated) \* 100,2),"%")

# Building Models

# Creating exhaustive set of store-dept-date combination

store\_dept\_date\_exh <- train %>% select(Store, Dept, Date, IsHoliday) %>% rbind(test\_final) %>% select(-IsHoliday) %>% complete(nesting(Store,Dept),Date)

# Checking number of data points for each store-dept combinations

cnt\_obs\_store\_dept <- train %>% group\_by(Store,Dept) %>% summarize(n())

# Filtering store-Dept combinations that will be forecasted using Time Series i.e. n\_obs >= 104

time\_series\_store\_dept <- cnt\_obs\_store\_dept %>% filter(`n()` >= 104) %>% inner\_join(store\_dept\_date\_exh, by = c("Store", "Dept")) %>% select(-`n()`) %>% left\_join(train, by = c("Store","Dept","Date")) %>% select(-IsHoliday) %>% mutate(id = paste(Store,Dept, sep = "\_")) %>% ungroup()

# Treating missing weeks data in between normal weeks by taking avg of previous and next week

time\_series\_store\_dept\_mistreat <- time\_series\_store\_dept %>% arrange(Store, Dept, Date) %>% group\_by(Store, Dept) %>%

mutate(lead1 = lead(Weekly\_Sales), lag1 = lag(Weekly\_Sales)) %>% ungroup() %>%

rowwise() %>% mutate(wkly\_sales\_treated = ifelse(is.na(Weekly\_Sales) & Date < "2012-11-02", mean(c(lead1,lag1),na.rm = T),Weekly\_Sales))

date\_train <- seq(as.Date("2010-02-05"), as.Date("2012-10-26"), by = 7)

sales\_ts <- time\_series\_store\_dept\_mistreat %>% filter(Date < "2012-11-02", id == "45\_98") %>% mutate(wkly\_sales\_treated = ifelse(is.na(wkly\_sales\_treated)|is.nan(wkly\_sales\_treated),0,wkly\_sales\_treated)) %>% select(wkly\_sales\_treated) %>% xts(order.by = date\_train)

#1. Identifying the order of differencing.

sales.ts.df1 <- diff(sales\_ts,1,1)

sales.ts.df1\_v1 <- ifelse(is.na(sales.ts.df1),0,sales.ts.df1)

#adf.test(sales.ts.df1\_v1, alternative = "stationary", k = 0)

plot.ts(sales.ts.df1\_v1)

#2. Identifying MA and AR terms.

acf(sales.ts.df1\_v1, lag.max = 10)

##Cut off of PACF at 3 indicates MA term i.e. q = 3.

pacf(sales.ts.df1\_v1, lag.max = 10)

##Building model with abobve parameters and plotting actuals + forecast for a sample store-dept combination

fit <- arima(sales\_ts, c(1, 1, 2),seasonal = list(order = c(1, 1, 3), period = 52))

pred <- forecast(fit,39)

plot(pred)

#Regression

##Forming an analytical dataset with all the predictors like CPI, Unemployment, Temperature, Markdowns and Holiday Flags.

hol\_flag <- tibble(Date = c("2010-02-12","2011-02-11", "2012-02-10","2013-02-08","2010-09-10","2011-09-09","2012-09-07","2013-09-06","2010-11-26","2011-11-25","2012-11-23","2013-11-29","2010-12-31","2011-12-30","2012-12-28","2013-12-27"), hol = c(rep("SB",4), rep("LD",4), rep("TG",4), rep("CH",4)))

hol\_flag$Date <- as.Date(hol\_flag$Date)

train\_reg\_data <- train\_outlier\_treated %>% select(-4,-5,-13,-c(16:20)) %>% left\_join(hol\_flag, by = "Date") %>% mutate(hol = ifelse(is.na(hol), "Other", hol)) %>% select(-3)

train\_reg\_data$hol <- as.factor(train\_reg\_data$hol)

x <- model.matrix(Wkly\_sales\_treated~.,train\_reg\_data)

y <- train\_reg\_data$Wkly\_sales\_treated

# Dividing data into test and train

train = sample(1:nrow(x),2\*nrow(x)/3)

test = (-train)

y.test = y[test]

##Cross validation to identify best tuning parameter. The tuning parameter for lowest cross validation error is 823.10.

cv.out = cv.glmnet(x[train,], y[train],alpha = 0)

bestlam = cv.out$lambda.min

plot(cv.out)

##Building model using this parameter.

ridge.mod = glmnet(x,y,alpha = 0, lambda = bestlam, thresh = 1e-12)

#Boosting

##Building boosting model using same predictors as used in reidge regression.

sales.boost <- gbm(Wkly\_sales\_treated~., data = train\_reg\_data, distribution = "gaussian", n.trees = 5000, interaction.depth = 4, shrinkage = 0.2, verbose = F)

summary(sales.boost)

2.3 Plots & Output:

**Chart, histogram

Description automatically generated**

**A picture containing graphical user interface

Description automatically generated**

**Chart, histogram

Description automatically generated**

**A picture containing text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

**Chart, scatter chart

Description automatically generated**

**Chart, scatter chart

Description automatically generated**

**Chart, scatter chart

Description automatically generated**

**Chart, line chart

Description automatically generated**

**Chart

Description automatically generatedLine chart

Description automatically generated with medium confidenceChart

Description automatically generatedChart

Description automatically generated with medium confidenceChart, box and whisker chart

Description automatically generatedChart

Description automatically generated**

**Chart, histogram

Description automatically generated**

**Diagram, histogram

Description automatically generated**

Chart

Description automatically generated

Text, table

Description automatically generated

**3. Result/Insights:**

When plotting sales by week of store types we can get the actual duration when sales where greater as compared to average. By referring the same graph, we can conclude that Thanksgiving and Christmas periods are important and hence generating the biggest peaks for sales. Other smaller peaks correspond to Labour Day and Easter Day. Also, we can see that this trend is store-type independent i.e., all stores nearly follow the same pattern.

Sales trend across month: Sales seems to be high at 1st week of the month and then falls slowly in other weeks as indicated by boxplot. Outliers in non-first week resemble the Thanksgiving and Christmas holiday sales.

Is there a relation between CPI and sales? The relation between sales and consumer price index (CPI) is very weak. Sales decreases negligibly with increase in price index.

Is there a relation between Unemployment and sales? Unemployment does not affect sales significantly. Sales reduces slightly at high values of unemployment.

Is there a relation between average temperature of the week and sales? There is no evident relation between the two.

Interestingly, 22nd week of the year is the 5th best sales. It is end of May and the time when schools are closed.

Christmas holiday introduces as the last days of the year. But people generally shop at 51st week. So, when we look at the total sales of holidays, Thanksgiving has higher sales between them which was assigned by Walmart. But when we look at the data, we can understand it is not a good idea to assign Christmas sales in data to last days of the year. It must assign 51st week.

January sales are significantly less than other months. This is the result of November and December high sales. After two high sales months, people prefer to pay less in January.

The predicted values are plotted in the graph one can refer than for better understanding of stats.

**4. Conclusion:**

Walmart, being one of the biggest names in the retailer & FMCG sector, has an approximate 245 million customers visiting across its 10000+ stores all over the globe. With such a huge number of interactions, the data(unstructured) collected by hi-tech systems of Walmart are enormous. According to a research it is estimated that Walmart produces 2.5 petabytes of data every hour and which is increasing with every passing hour. With such data it is important to use it efficiently, so that Walmart’s dominance can be maintained and growth can be achieved. By using data analytics and Predictive analytics Walmart is achieving the same. This project implemented by us is just one small part of the big picture. The significance of our project can be determined by the accuracy it provides to our client-'WALMART'. As predicted, it is showing that the sales numbers are going to increase in the future and hence it is important for Walmart to check the effectiveness of the inventory in its branches. So that they can meet the demand of the customers and provide better services. Periods of events such as Christmas and Thanksgiving, drive many people in the shops, hence relevant offers should be given and items needs to be present, to give a satisfiable experience to its customers.

**5. Further development**

As a modeling perspective, data will be made more stationary with different techniques, feature selection and feature engineering can be added to model also.

As a data quality perspective, to improve the results and make it more accurate, more data is needed (which also includes some from the recent times). Also, many more events can be taken into consideration such as Halloween, Memorial Day, etc.

There are markdowns and store sales for some special seasons. The effect of markdowns and discounts on departments can be added to the model.

From the data it is observed that some stores and department have higher effect on some special occasions. So, different models can be built and implemented for such items.

Market basket analysis can also be added to the model, to find goods that have a higher demand that can be both intra/inter department and relationship between goods can be identified.

**6. References:**

https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting

https://data-flair.training/blogs/data-analytics-tutorial/

https://www.walmart.com/

https://data-flair.training/blogs/r-predictive-and-descriptive-analytics/

https://data-flair.training/blogs/machine-learning-for-r-programming/