Retail Analysis with Walmart Data

1. Business Scenario:

One of the leading retail stores in the US, Walmart, would like to predict the sales and demand accurately. There are certain events and holidays which impact sales on each day. There are sales data available for 45 stores of Walmart. The business is facing a challenge due to unforeseen demands and runs out of stock sometimes, due to the inappropriate machine learning algorithm. An ideal ML algorithm will predict demand accurately and ingest factors like economic conditions including CPI, Unemployment Index, etc.

Walmart runs several promotional markdown events throughout the year. These markdowns precede prominent holidays, the four largest of all, which are the Super Bowl, Labour Day, Thanksgiving, and Christmas. The weeks including these holidays are weighted five times higher in the evaluation than non-holiday weeks. Part of the challenge presented by this competition is modelling the effects of markdowns on these holiday weeks in the absence of complete/ideal historical data. Historical sales data for 45 Walmart stores located in different regions are available.

2. Expectation /Goals / Objectives:

The primary objective is to create a model that can predict sales/demand by incorporating various data points and economic variables such as CPI, Fuel price and Unemployment rates. Also to identify whether these variables have a significant impact on sales. We wish to explore the data set and perform some basic analysis to answer few questions such as.

Basic Statistics tasks

- Which store has maximum sales
- Which store has maximum standard deviation i.e., the sales vary a lot. Also, find out the coefficient of mean to standard deviation
- Which store/s has good quarterly growth rate in Q3'2012
- Some holidays have a negative impact on sales. Find out holidays which have higher sales than the mean sales in non-holiday season for all stores together
- Provide a monthly and semester view of sales in units and give insights

Finally we wish to Model the Weekly sales based on the independent variables utilising the sales dataset to identify if we can successfully predict the demand/sales using these predictors in a reliable manner. As an example we will model the sales for Store 1 and also to understand if the independent variables have a significant impact on the Demand/Sales which is our target variable.

Statistical Model : For Store 1 – Build prediction models to forecast demand

- Linear Regression – Utilize variables like date and restructure dates as 1 for 5 Feb 2010 (starting from the earliest date in order). Hypothesize if CPI, unemployment, and fuel price have any impact on sales.
- Change dates into days by creating new variable.
- Select the model which gives best accuracy.

3. Dataset Description:

This is the historical data which covers sales from 2010-02-05 to 2012-11-01, in the file Walmart_Store_sales. Within this we find the following fields:

- Store - the store number
- Date - the week of sales
- Weekly_Sales - sales for the given store
- Holiday_Flag - whether the week is a special holiday week 1 – Holiday week 0 – Non-holiday week
- Temperature - Temperature on the day of sale
- Fuel_Price - Cost of fuel in the region
- CPI – Prevailing consumer price index
- Unemployment - Prevailing unemployment rate

Holiday Events

Super Bowl: 12-Feb-10, 11-Feb-11, 10-Feb-12, 8-Feb-13
Labour 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

Summary research notes/impressions about data:

- The data available is based on Weekly sales which were captured at the end of every week. The data captures the total weekly sales till Friday of the week.
- Another variables captured is a factor(dummy) variable to indicate whether it was one of the 4 major US holidays (Super Bowl, Labour Day, Thanksgiving, Christmas) under Holiday Flag. This is a factor variable with 2 levels where 1 – Holiday week 0 – Non-holiday week.
  This is important as the Holiday weeks have markdowns and sales as per Business definition. Which means prices are normally slashed and discounts provided to promote more units of goods and inventory sold compared to usual days. In the data however we have just the dollar value for the sales. So there is no indication on the actual sales volume in terms of units of goods sold.
- Store number identifies the particular store where the data is tagged. This is relevant as the index or Unique identifier for the particular store. Business could also use this variable to represent a particular region in case the sales in the region can be represented fairly by the store where the data is captured i.e. if the store is a fair and good representation of all the other stores within its region or neighbourhood.
- We also have data on other environment and social variables like – Temperature, Fuel Prices, CPI and Unemployment. These are hypothesized as having a influence on the sales. Hence the data has been captured to confirm if there is indeed a clear and statistically significant impact of these predictors on demand which can explain significantly variances of sales across the stores. These are numeric variables.
- When we view a summary of the data it is also seen that there a no missing values within the captured data for any field. This is good and we do not need to do much cleaning, replacement or omission of data for missing values.
- With regards to the time frame of the Data. We have nearly 3 years of continuous data . To be precise we have 2.75 years or 2 years and 3 Quarters worth of data.

  Additional Variables Created:
- Created variable for the month derived from the Date variable – To analyse month wise sales
- Created variable for Quarters derived from the Date variable – To compare sales in 2 quarters
- Created variable Semester derived from the Date variable – To analyse semester sales

4. Basic Statistics tasks : Data Exploration

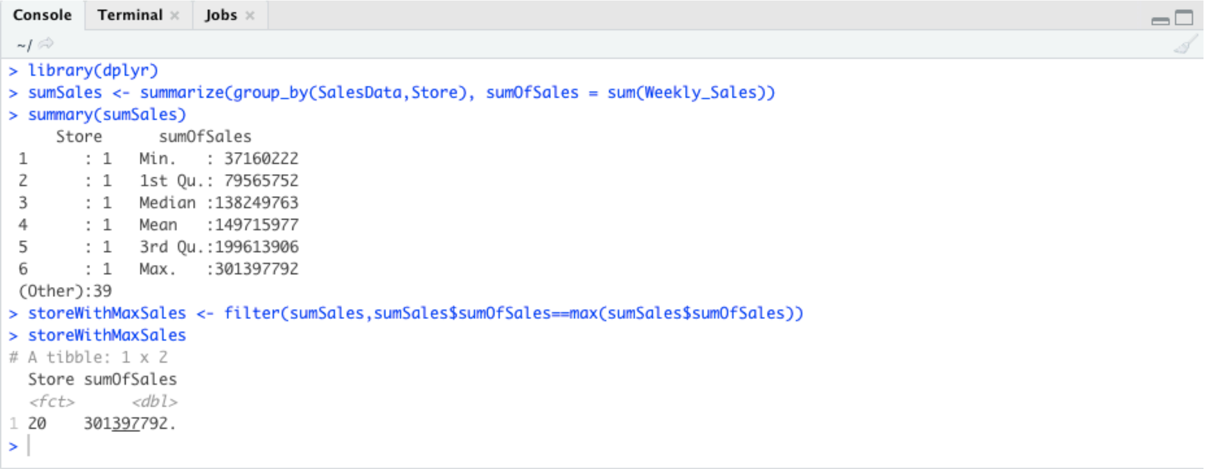
Q1. Which store has maximum sales :

Answer: Store #20 has max total sales at $301397792

Steps :

- Step 1: First take the Sum of Weekly_Sales data and group it by store. Saved this result in sumSales
- Step 2: Next we find the store with max sales by using filter and providing the logical check where the sumOfSales is equal to the max value.
- Solution : We get the answer is store 20 which has the max total sales.

Code:



```
> library(dplyr)
> sumSales <- summarize(group_by(SalesData,Store), sumOfSales = sum(Weekly_Sales))
> summary(sumSales)
     Store      sumOfSales
 1      : 1   Min.   : 37160222
 2      : 1   1st Qu.: 79565752
 3      : 1   Median :138249763
 4      : 1   Mean   :149715977
 5      : 1   3rd Qu.:199613906
 6      : 1   Max.   :301397792
 (Other):39
> storeWithMaxSales <- filter(sumSales,sumSales$sumOfSales==max(sumSales$sumOfSales))
> storeWithMaxSales
# A tibble: 1 x 2
  Store sumOfSales
  <fct>      <dbl>
1 20    301397792.
> 
```

Q2. Which store has maximum standard deviation i.e., the sales vary a lot. Also, find out the coefficient of mean to standard deviation

Answer : Here there are 2 solutions depending on the measure of Variability chosen.

a. If we go only by Standard deviation of the weekly sales the answer is Store 14 with the highest standard deviation of $317570
b. If we check the Coefficient of Standard deviation (i.e. SD/Mean) then the answer is Store 35 with a Maximum coefficient of SD : 0.23. The standard deviation for this store was $211243

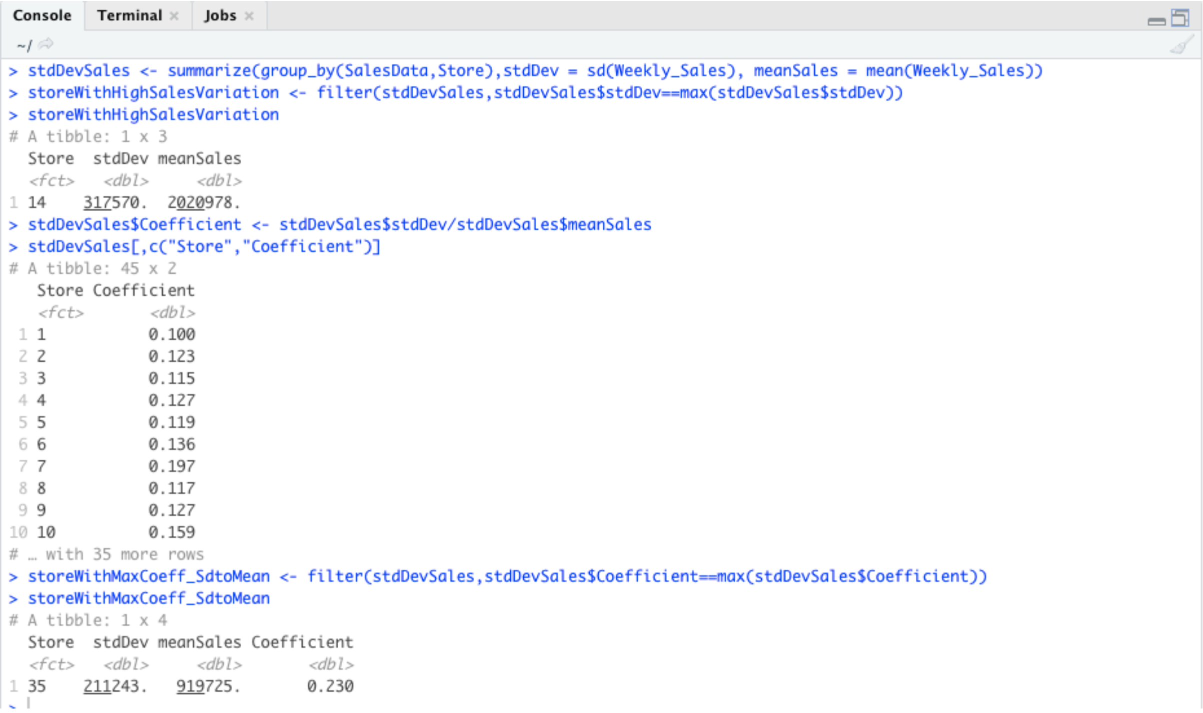
Conclusion: The coefficient of SD is a much better measure of Variability as opposed to just looking at which store had a max Standard deviation as these stores differ significantly in terms of the Average Sales. So a more robust measure is to compare the Standard deviation in relation to average sales. Based on this our conclusion is that Store 35 has the maximum variance in its sales.

Steps:

- Step 1: find the SD and mean on the sales data using summarize again group by store.
- Step 2: Now we find the store with max Std deviation of sales by using filter and applying condition for stdDev being equal to max among all stores
- Step 3: Find the coefficient of mean to std Deviation. Created a new variable Coefficient in the table and divided SD with mean
- Step 4: Filter out the store with the maximum coefficient of Standard Deviation.

Notes: As per the Coefficient of standard Deviation to Sales , We see that Store 35 is having a higher deviation by this metric.It creates a more standard scale to compare the deviation between the different stores.

Code:



```
> stdDevSales <- summarize(group_by(SalesData,Store),stdDev = sd(Weekly_Sales), meanSales = mean(Weekly_Sales))
> storeWithHighSalesVariation <- filter(stdDevSales,stdDevSales$stdDev==max(stdDevSales$stdDev))
> storeWithHighSalesVariation
# A tibble: 1 x 3
  Store  stdDev meanSales
  <fct>   <dbl>     <dbl>
1 14    317570.  2020978.
> stdDevSales$Coefficient <- stdDevSales$stdDev/stdDevSales$meanSales
> stdDevSales[,c("Store","Coefficient")]
# A tibble: 45 x 2
   Store Coefficient
   <fct>       <dbl>
 1 1           0.100
 2 2           0.123
 3 3           0.115
 4 4           0.127
 5 5           0.119
 6 6           0.136
 7 7           0.197
 8 8           0.117
 9 9           0.127
10 10          0.159
# … with 35 more rows
> storeWithMaxCoeff_SdtoMean <- filter(stdDevSales,stdDevSales$Coefficient==max(stdDevSales$Coefficient))
> storeWithMaxCoeff_SdtoMean
# A tibble: 1 x 4
  Store  stdDev meanSales Coefficient
  <fct>   <dbl>     <dbl>       <dbl>
1 35    211243.   919725.       0.230
>
```

Q3. Which store/s has good quarterly growth rate in Q3’2012

Answer : Here there are 2 solutions depending on the way we select the Quarters.

The data has sales starting from Feb-2010 to Oct-2012. Since this data does not start from January like a typical calendar year. This creates two approaches. Approach 1 is better as data is uniformly split.

Approach 1: When we consider the Quarter’s based on the data start date (Feb-2010). i.e.

(Q1 = Feb, Mar, Apr   Q2 = May, Jun, Jul   Q3 = Aug, Sep, Oct   Q4 = Nov, Dec, Jan)

In this case we notice that all the 11 Quarters in the dataset have equal width (number of observations) = 585

In this approach the max sales growth is seen in Store 39 where Quarter over Quarter sales growth was 2.73%

Steps:

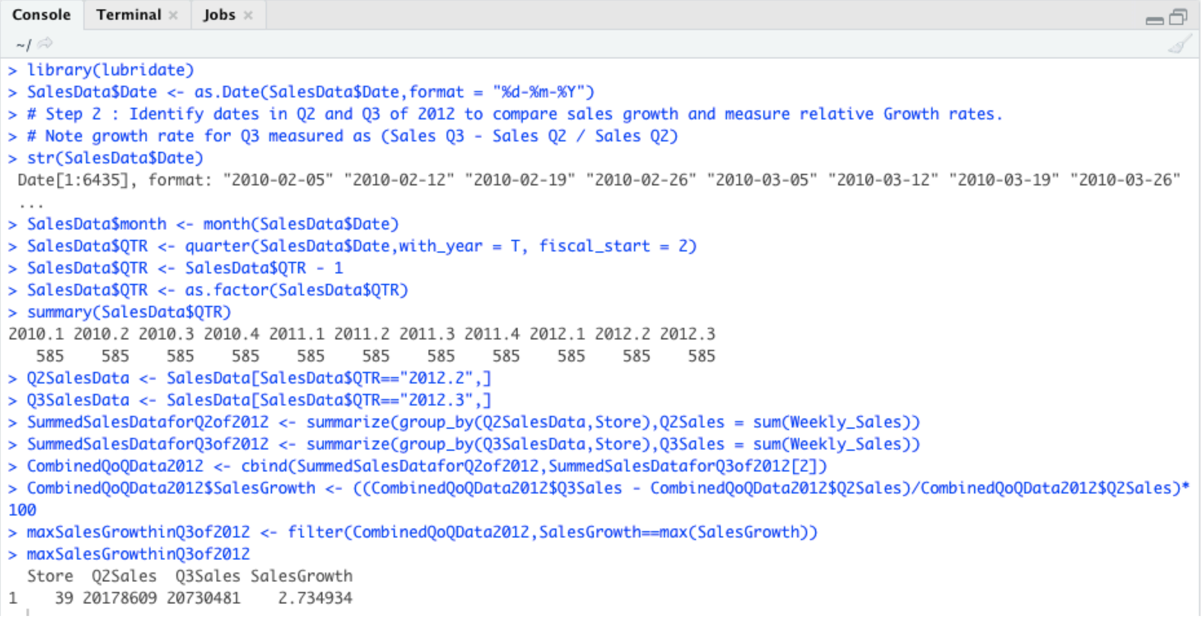
- Step 1: Convert the Date field in the SalesData frame to date type instead of char
- Step 2: Identify dates in Q2 and Q3 of 2012 to compare sales growth and measure relative Growth rates.

a) Select Quarter to start from the second month(Feb) since this is possible with the quarter function of lubridate package. Also month as the data starts from 2nd month i.e. Feb-2010.
b) Note the output of Quarter function has the next year tagged to the quarter since fiscal start was 2nd month. Hence the need to subtract 1 from the output
c) SalesData$QTR <- SalesData$QTR – 1. Next Convert QTR to factor variable.

- Step 3: Now extract the sales data for Q2 and Q3 of 2012 out of all the observations
- Step 4: Next summarize the Sales for the Quarters by store. We need the total sales for each store in the Quarter.
- Step 5: Using the total sales we can compare the sales between Q2’2012 and Q3’2012 for each store and obtain the growth rate. Combined the Q2 and Q3 sales against each store to a common table/data frame(CombinedQoQData2012) to calculate the growth rate.
  Note growth rate for Q3 measured as (Sales Q3 - Sales Q2 / Sales Q2)
- Step 6: Finally filter out the store with the maximum sales growth rate.

We note that the answer shows that most of the stores had a decline in sales in Q3 2012 when compared to Q2 2012. Finally store which had max Sales growth is Store 39

Code:



```
> library(lubridate)
> SalesData$Date <- as.Date(SalesData$Date,format = "%d-%m-%Y")
> # Step 2 : Identify dates in Q2 and Q3 of 2012 to compare sales growth and measure relative Growth rates.
> # Note growth rate for Q3 measured as (Sales Q3 - Sales Q2 / Sales Q2)
> str(SalesData$Date)
 Date[1:6435], format: "2010-02-05" "2010-02-12" "2010-02-19" "2010-02-26" "2010-03-05" "2010-03-12" "2010-03-19" "2010-03-26"
 ...
> SalesData$month <- month(SalesData$Date)
> SalesData$QTR <- quarter(SalesData$Date,with_year = T, fiscal_start = 2)
> SalesData$QTR <- SalesData$QTR - 1
> SalesData$QTR <- as.factor(SalesData$QTR)
> summary(SalesData$QTR)
2010.1 2010.2 2010.3 2010.4 2011.1 2011.2 2011.3 2011.4 2012.1 2012.2 2012.3
   585    585    585    585    585    585    585    585    585    585    585
> Q2SalesData <- SalesData[SalesData$QTR=="2012.2",]
> Q3SalesData <- SalesData[SalesData$QTR=="2012.3",]
> SummedSalesDataforQ2of2012 <- summarize(group_by(Q2SalesData,Store),Q2Sales = sum(Weekly_Sales))
> SummedSalesDataforQ3of2012 <- summarize(group_by(Q3SalesData,Store),Q3Sales = sum(Weekly_Sales))
> CombinedQoQData2012 <- cbind(SummedSalesDataforQ2of2012,SummedSalesDataforQ3of2012[2])
> CombinedQoQData2012$SalesGrowth <- ((CombinedQoQData2012$Q3Sales - CombinedQoQData2012$Q2Sales)/CombinedQoQData2012$Q2Sales)*
100
> maxSalesGrowthinQ3of2012 <- filter(CombinedQoQData2012,SalesGrowth==max(SalesGrowth))
> maxSalesGrowthinQ3of2012
  Store  Q2Sales  Q3Sales SalesGrowth
1    39 20178609 20730481    2.734934
```

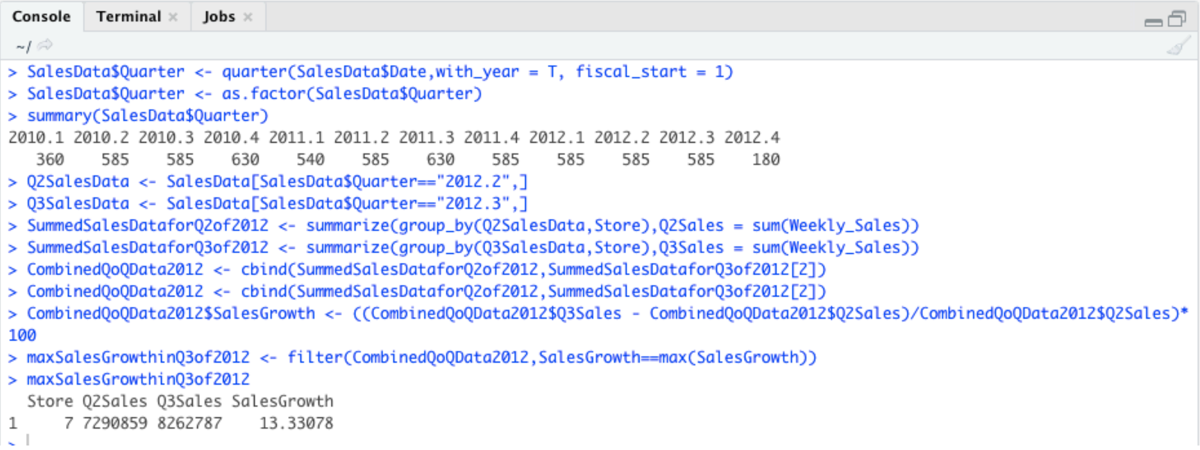
Approach 2: When we consider the Quarter’s based on the calendar year. i.e. Jan onwards
(Q1 = Jan, Feb, Mar, Q2 = Apr, May, Jun, Q3 = Jul, Aug, Sep, Q4 = Oct, Nov, Dec)
In this case we notice that all the 12 Quarters in the dataset have different widths (number of observations is not the same. But Q2 and Q3 of 2012 have same width = 585)

In this approach the max sales growth is seen in Store 7 where Quarter over Quarter sales growth was 13.33 %

Steps:

- Step 1: Convert the Date field in the SalesData frame to date type instead of char
- Step 2: Identify dates in Q2 and Q3 of 2012 to compare sales growth and measure relative Growth rates. Select Quarter to start from the first month(Jan) for this approach.
- Step 3: Now extract the sales data for Q2 and Q3 of 2012 out of all the observations
- Step 4: Next summarize the Sales for the Quarters by store. We need the total sales for each store in the Quarter.
- Step 5: Using the total sales we can compare the sales between Q2’2012 and Q3’2012 for each store and obtain the growth rate. Combined the Q2 and Q3 sales against each store to a common table/data frame (CombinedQoQData2012) to calculate the growth rate.
  Note growth rate for Q3 measured as (Sales Q3 - Sales Q2 / Sales Q2)
- Step 6: Finally filter out the store with the maximum sales growth rate.

Code:



```
> SalesData$Quarter <- quarter(SalesData$Date,with_year = T, fiscal_start = 1)
> SalesData$Quarter <- as.factor(SalesData$Quarter)
> summary(SalesData$Quarter)
2010.1 2010.2 2010.3 2010.4 2011.1 2011.2 2011.3 2011.4 2012.1 2012.2 2012.3 2012.4
   360    585    585    630    540    585    630    585    585    585    585    180
> Q2SalesData <- SalesData[SalesData$Quarter=="2012.2",]
> Q3SalesData <- SalesData[SalesData$Quarter=="2012.3",]
> SummedSalesDataforQ2of2012 <- summarize(group_by(Q2SalesData,Store),Q2Sales = sum(Weekly_Sales))
> SummedSalesDataforQ3of2012 <- summarize(group_by(Q3SalesData,Store),Q3Sales = sum(Weekly_Sales))
> CombinedQoQData2012 <- cbind(SummedSalesDataforQ2of2012,SummedSalesDataforQ3of2012[2])
> CombinedQoQData2012 <- cbind(SummedSalesDataforQ2of2012,SummedSalesDataforQ3of2012[2])
> CombinedQoQData2012$SalesGrowth <- ((CombinedQoQData2012$Q3Sales - CombinedQoQData2012$Q2Sales)/CombinedQoQData2012$Q2Sales)*
100
> maxSalesGrowthinQ3of2012 <- filter(CombinedQoQData2012,SalesGrowth==max(SalesGrowth))
> maxSalesGrowthinQ3of2012
  Store Q2Sales Q3Sales SalesGrowth
1     7 7290859 8262787    13.33078
```

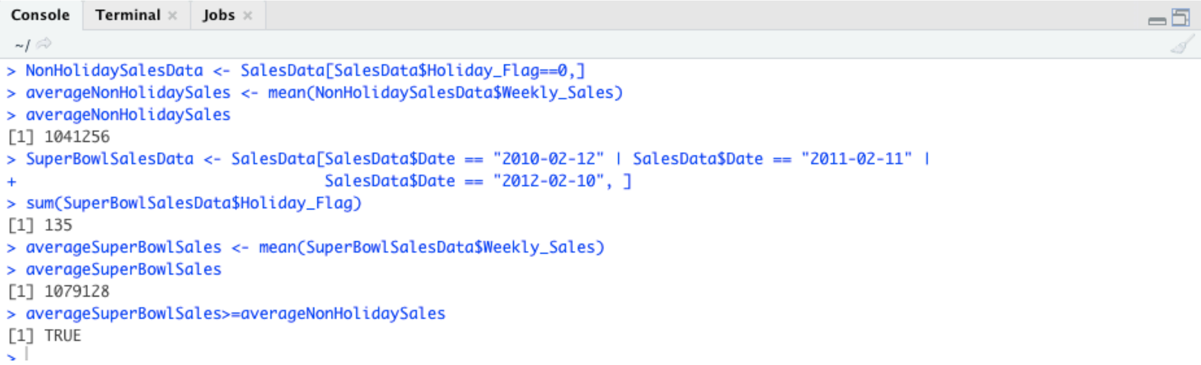
Q4. Some holidays have a negative impact on sales. Find out holidays which have higher sales than the mean sales in non-holiday season for all stores together

Answer: Out of all the Holidays Thanksgiving has the max Average Holiday sales. We see that only for Christmas the Sales are below the Non holiday average.

- One must note that the total dollar-value of the sales is not a fair comparison between Non-Holiday v/s Holidays. Since Holidays normally have a markdown(discount). Due to the markdown and discounts during holiday the dollar($) amount (Price) of any given product is lesser.
- This means that while there could be a possibility that actual units sold were higher during holidays .The final dollar value of the Sales can actually be lesser or equal the Non discounted sales value during the normal period. So this comparison is dependent upon the amount of Holiday discount which is not specified in this analysis/dataset.
- A better measure to compare the impact of Holiday discounts on the Sales compared to Non-Holiday weeks would be to compare actual units of goods sold across categories between the two periods(Holiday v/s Non-Holiday). But this information is not currently available in the data-set provided for this analysis.

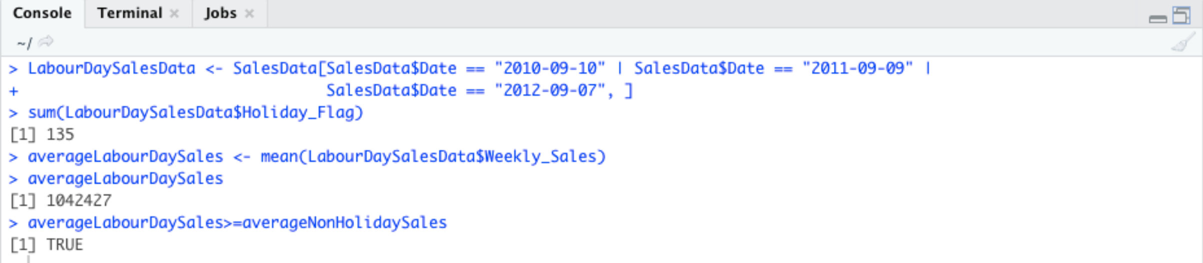
Steps and Codes:
Compare holiday sales average to non holiday sales average . First we split to get non-holiday sales average
Step 1: Exclude the holiday data for the stores(i.e. Select rows where Holiday_Flag=0). Then find the average sales for non-holiday dates
We see average NonHolidaySales is 1041256.
Step 2: Next we pick only the holiday observations for the stores(i.e. Select rows where Holiday_Flag=1) and compare with average sales. We do this for each of the 4 holidays.
Note: Excluded the holiday dates for 2013 during filter as these are not contained in the data-set.

Superbowl: Feb-10, 11-Feb-11, 10-Feb-12, 8-Feb-13. **The average sales during Superbowl week was $1079128 which is higher than during the non-holiday period.**



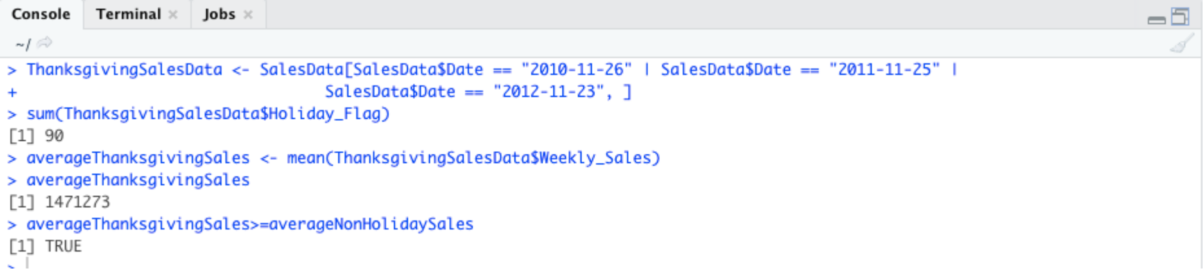
```
> NonHolidaySalesData <- SalesData[SalesData$Holiday_Flag==0,]
> averageNonHolidaySales <- mean(NonHolidaySalesData$Weekly_Sales)
> averageNonHolidaySales
[1] 1041256
> SuperBowlSalesData <- SalesData[SalesData$Date == "2010-02-12" | SalesData$Date == "2011-02-11" |
+                                 SalesData$Date == "2012-02-10", ]
> sum(SuperBowlSalesData$Holiday_Flag)
[1] 135
> averageSuperBowlSales <- mean(SuperBowlSalesData$Weekly_Sales)
> averageSuperBowlSales
[1] 1079128
> averageSuperBowlSales>=averageNonHolidaySales
[1] TRUE
```

Labour Day: 10-Sep-10, 9-Sep-11, 7-Sep-12, 6-Sep-13. **The average sales during Labour day week was $1042427 which is marginally higher than during the non-holiday period.**



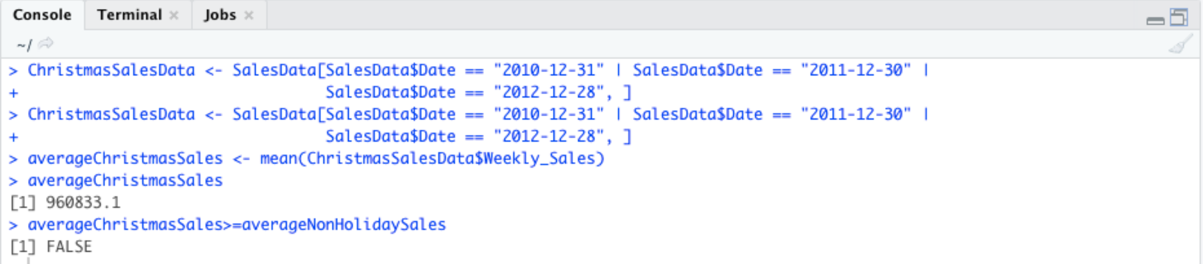
```
> LabourDaySalesData <- SalesData[SalesData$Date == "2010-09-10" | SalesData$Date == "2011-09-09" |
+                                 SalesData$Date == "2012-09-07", ]
> sum(LabourDaySalesData$Holiday_Flag)
[1] 135
> averageLabourDaySales <- mean(LabourDaySalesData$Weekly_Sales)
> averageLabourDaySales
[1] 1042427
> averageLabourDaySales>=averageNonHolidaySales
[1] TRUE
```

Thanksgiving: 26-Nov-10, 25-Nov-11, 23-Nov-12, 29-Nov-13. **The average sales during Thanksgiving week was $ 1471273 which is higher than during the non-holiday period.**



```
> ThanksgivingSalesData <- SalesData[SalesData$Date == "2010-11-26" | SalesData$Date == "2011-11-25" |
+                                 SalesData$Date == "2012-11-23", ]
> sum(ThanksgivingSalesData$Holiday_Flag)
[1] 90
> averageThanksgivingSales <- mean(ThanksgivingSalesData$Weekly_Sales)
> averageThanksgivingSales
[1] 1471273
> averageThanksgivingSales>=averageNonHolidaySales
[1] TRUE
```

Christmas: 31-Dec-10, 30-Dec-11, 28-Dec-12, 27-Dec-13. **The average sales during Christmas week was $ 960833.1 was lesser than during the non-holiday period.**



```
> ChristmasSalesData <- SalesData[SalesData$Date == "2010-12-31" | SalesData$Date == "2011-12-30" |
+                                 SalesData$Date == "2012-12-28", ]
> ChristmasSalesData <- SalesData[SalesData$Date == "2010-12-31" | SalesData$Date == "2011-12-30" |
+                                 SalesData$Date == "2012-12-28", ]
> averageChristmasSales <- mean(ChristmasSalesData$Weekly_Sales)
> averageChristmasSales
[1] 960833.1
> averageChristmasSales>=averageNonHolidaySales
[1] FALSE
```

Q5. Provide a monthly and semester view of sales in units and give insights

Answer:

Part 1 - Monthly Sales analysis insights:

- The average sales are highest in last part (i.e. months 11 and 12) of the year. January has lowest average sales (923884.6) and December average sales(1281863.6) is the highest.
- Also note that December has a higher Standard Deviation in the sales. The box plot also confirms this as there are plenty of outlier points on the higher side of the scale in December.
- Also there appears to be a dip in sales during September and October where the average sales is below 1000000

Part 2 – Semester Sales analysis insights:

- On Average the 2nd Semester sales are higher than 1st Semester. Except for in the year 2012. This is logically conclusive as well since the monthly analysis validates that sales on average is higher in Nov and Dec months which fall in 2nd Semester.
- Note that for 2012- 2nd Semester data is incomplete/less as data was available only till October 2012. And as noted in monthly analysis the November and December month average sales tend to be the highest of all months in the year.
- But there is not a large difference/disparity in the average weekly sales between the semesters. Range of Average weekly sales is 1002080 to 1087128.The lowest average weekly sales was recorded in the 1st Semester of 2011 and The highest was recorded in the 2nd Semester of 2011.
- Again we see a high variance of weekly sales in the 2nd Semester with a large number of outliers. This again follows from our conclusion in Monthly sales analysis which revealed high number of outliers for weekly sales figures in Nov and Dec.

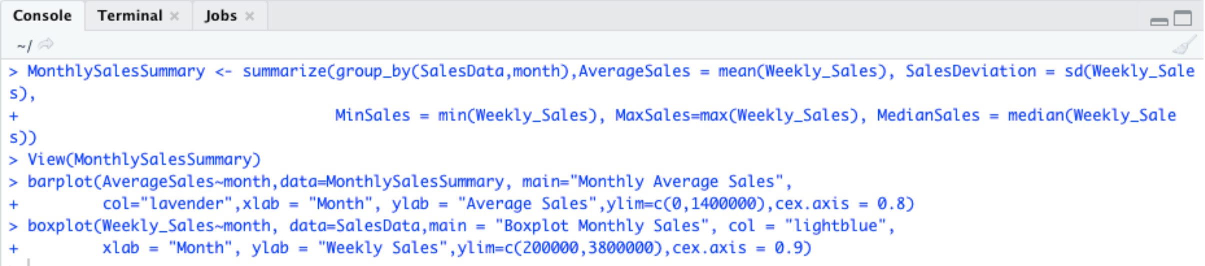
Steps and Code:

Part 1 - Monthly Sales analysis insights:

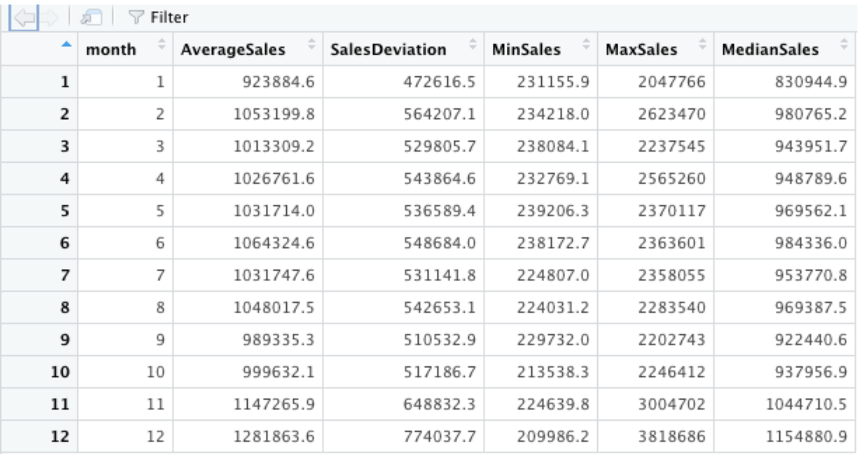
Step1: We already have the month information in the table which was derived during the exercise of classifying the data into Quarters under Question 3. So the next step is to get the summary of the average sales , SD, Min, Max and Median values grouped by the months for our analysis.

Step 2: Now plot graph for the average monthly sales for comparison

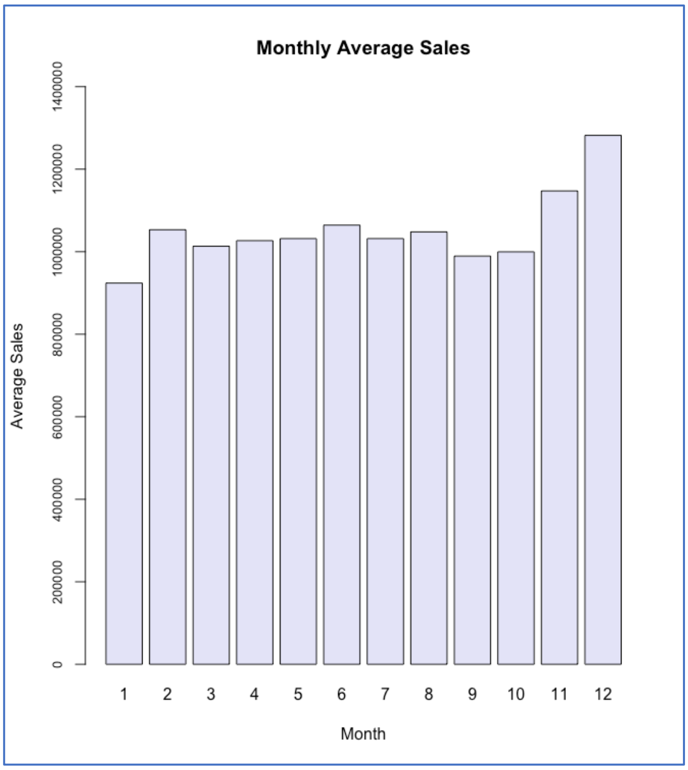
Step 3: Created box plot to visualize the ranges and spread of the data across months.

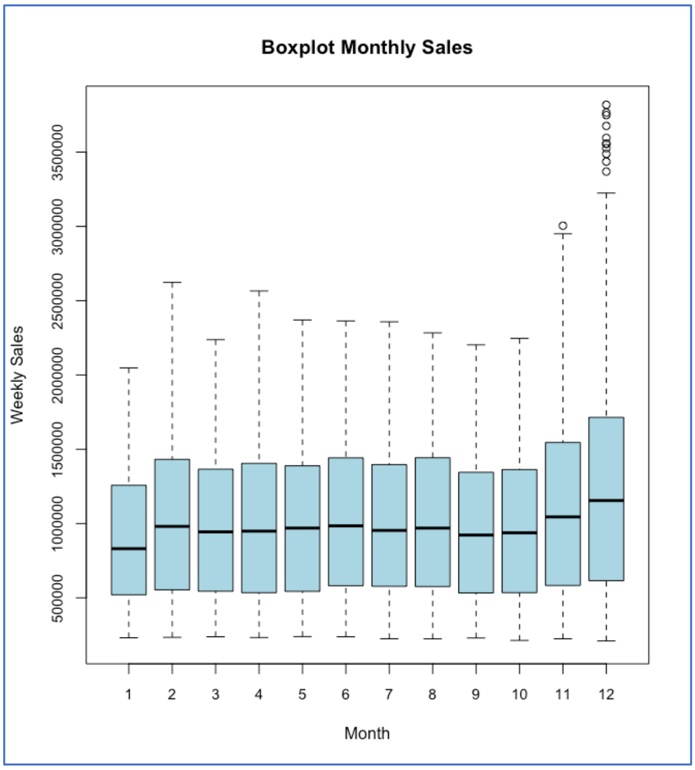


```
> MonthlySalesSummary <- summarize(group_by(SalesData,month),AverageSales = mean(Weekly_Sales), SalesDeviation = sd(Weekly_Sale
s),
+                               MinSales = min(Weekly_Sales), MaxSales=max(Weekly_Sales), MedianSales = median(Weekly_Sale
s))
> View(MonthlySalesSummary)
> barplot(AverageSales~month,data=MonthlySalesSummary, main="Monthly Average Sales",
+         col="lavender",xlab = "Month", ylab = "Average Sales",ylim=c(0,1400000),cex.axis = 0.8)
> boxplot(Weekly_Sales~month, data=SalesData,main = "Boxplot Monthly Sales", col = "lightblue",
+         xlab = "Month", ylab = "Weekly Sales",ylim=c(200000,3800000),cex.axis = 0.9)
```

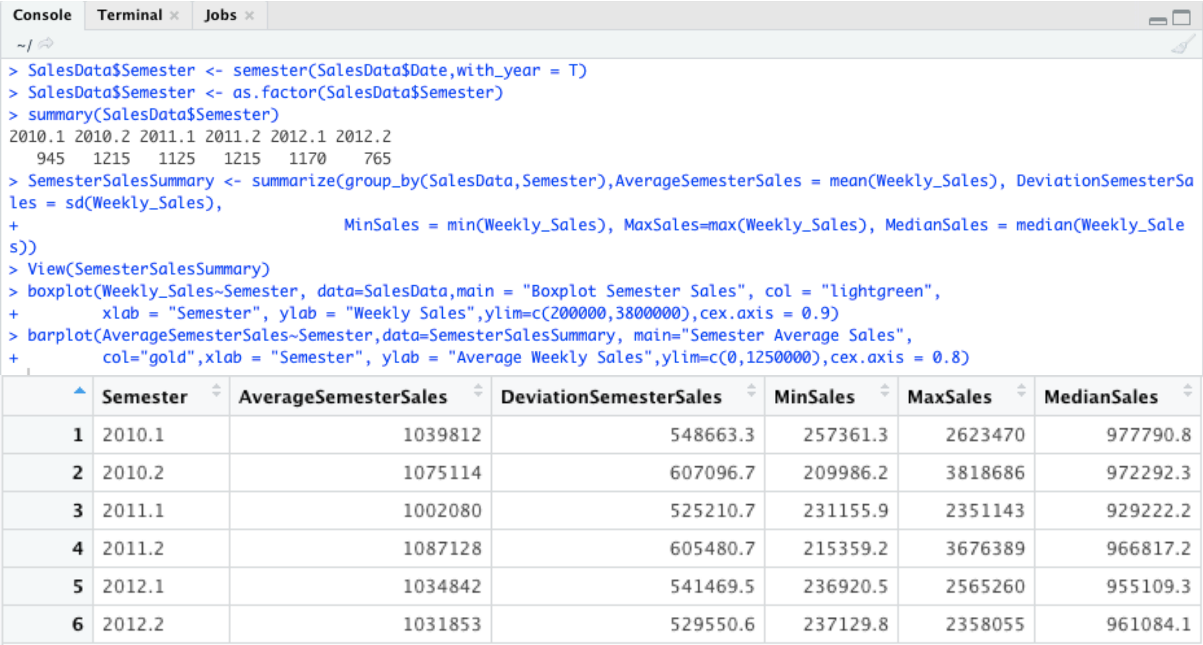


	month	AverageSales	SalesDeviation	MinSales	MaxSales	MedianSales
1	1	923884.6	472616.5	231155.9	2047766	830944.9
2	2	1053199.8	564207.1	234218.0	2623470	980765.2
3	3	1013309.2	529805.7	238084.1	2237545	943951.7
4	4	1026761.6	543864.6	232769.1	2565260	948789.6
5	5	1031714.0	536589.4	239206.3	2370117	969562.1
6	6	1064324.6	548684.0	238172.7	2363601	984336.0
7	7	1031747.6	531141.8	224807.0	2358055	953770.8
8	8	1048017.5	542653.1	224031.2	2283540	969387.5
9	9	989335.3	510532.9	229732.0	2202743	922440.6
10	10	999632.1	517186.7	213538.3	2246412	937956.9
11	11	1147265.9	648832.3	224639.8	3004702	1044710.5
12	12	1281863.6	774037.7	209986.2	3818686	1154880.9



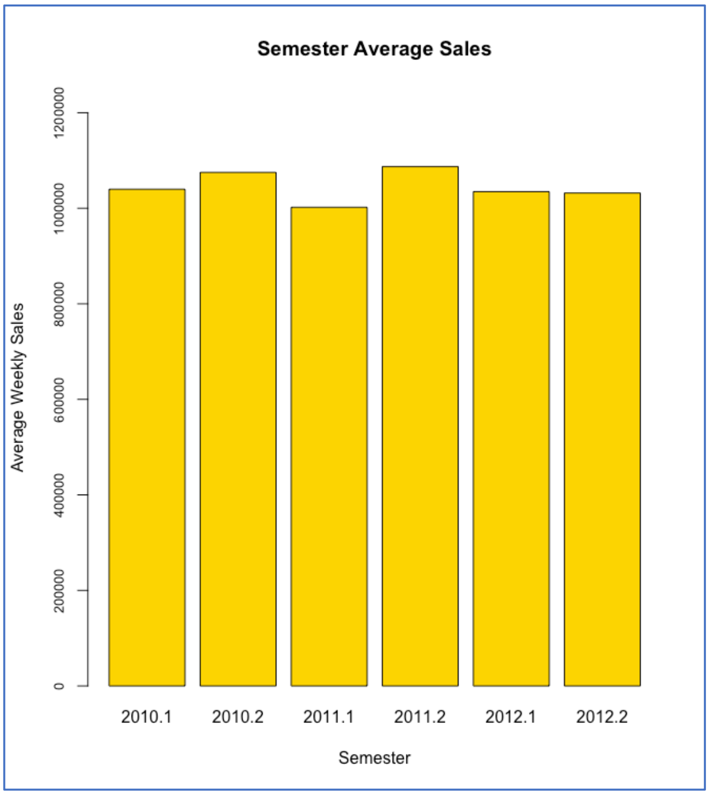


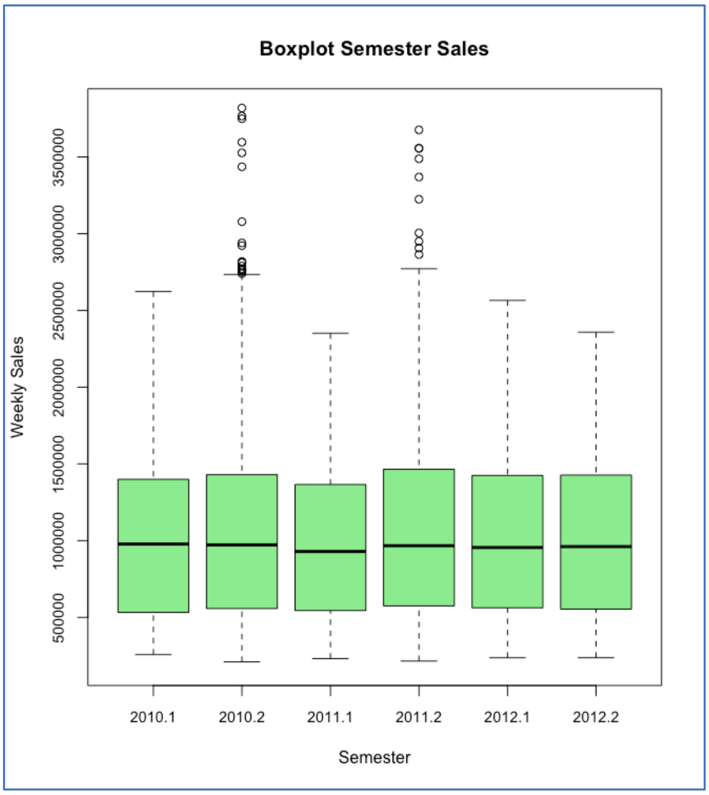
Part 2 - Semester Sales analysis insights:
Step1: Classify sales data into semesters. Adding semester variable to data.
Step2: Get the summary of the average sales , SD, Min, Max and Median values grouped by the Semester for our analysis.
Step 2: Now plot graph for the average Semester sales for comparison
Step 3: Created box plot to visualize the ranges and spread of the data across Semester.



```
> SalesData$Semester <- semester(SalesData$Date,with_year = T)
> SalesData$Semester <- as.factor(SalesData$Semester)
> summary(SalesData$Semester)
2010.1 2010.2 2011.1 2011.2 2012.1 2012.2 
   945   1215   1125   1215   1170    765 
> SemesterSalesSummary <- summarize(group_by(SalesData,Semester),AverageSemesterSales = mean(Weekly_Sales), DeviationSemesterSa
les = sd(Weekly_Sales),
+                                   MinSales = min(Weekly_Sales), MaxSales=max(Weekly_Sales), MedianSales = median(Weekly_Sale
s))
> View(SemesterSalesSummary)
> boxplot(Weekly_Sales~Semester, data=SalesData,main = "Boxplot Semester Sales", col = "lightgreen",
+         xlab = "Semester", ylab = "Weekly Sales",ylim=c(200000,3800000),cex.axis = 0.9)
> barplot(AverageSemesterSales~Semester,data=SemesterSalesSummary, main="Semester Average Sales",
+         col="gold",xlab = "Semester", ylab = "Average Weekly Sales",ylim=c(0,1250000),cex.axis = 0.8)
```

	Semester	AverageSemesterSales	DeviationSemesterSales	MinSales	MaxSales	MedianSales
1	2010.1	1039812	548663.3	257361.3	2623470	977790.8
2	2010.2	1075114	607096.7	209986.2	3818686	972292.3
3	2011.1	1002080	525210.7	231155.9	2351143	929222.2
4	2011.2	1087128	605480.7	215359.2	3676389	966817.2
5	2012.1	1034842	541469.5	236920.5	2565260	955109.3
6	2012.2	1031853	529550.6	237129.8	2358055	961084.1





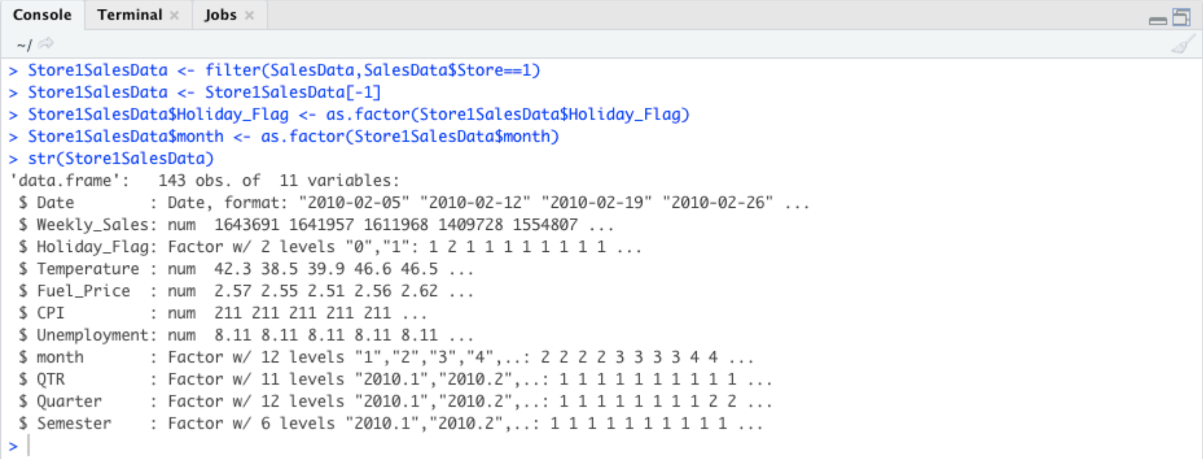
5. Linear Regression Models:

For Store 1 – Build prediction models to forecast demand

- Linear Regression – Utilize variables like date and restructure dates as 1 for 5 Feb 2010 (starting from the earliest date in order). Hypothesize if CPI, unemployment, and fuel price have any impact on sales.
- Change dates into days by creating new variable.
- Select the model which gives best accuracy.

Prerequisite Steps:

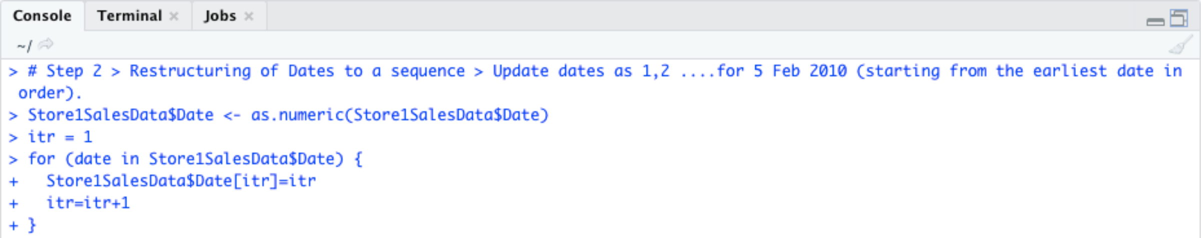
- Since we are creating model only for Store#1 . We will first filter the data for Store 1 observations only.
- Once filter is done remove first column which is Store number and as all values are referring to store number 1 itself.
- Ensuring that Holiday Flag and month are treated as factors to assess them based on the factor levels rather than as numeric data which would be incorrect. Since these are not Physical or meaningful as numeric quantities. For e.g. Variable like fuel price or temperature is a quantity but Month and Holiday_Flag are categorical in nature.



```
> Store1SalesData <- filter(SalesData,SalesData$Store==1)
> Store1SalesData <- Store1SalesData[-1]
> Store1SalesData$Holiday_Flag <- as.factor(Store1SalesData$Holiday_Flag)
> Store1SalesData$month <- as.factor(Store1SalesData$month)
> str(Store1SalesData)
'data.frame':	143 obs. of  11 variables:
 $ Date        : Date, format: "2010-02-05" "2010-02-12" "2010-02-19" "2010-02-26" ...
 $ Weekly_Sales: num  1643691 1641957 1611968 1409728 1554807 ...
 $ Holiday_Flag: Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 1 1 ...
 $ Temperature : num  42.3 38.5 39.9 46.6 46.5 ...
 $ Fuel_Price  : num  2.57 2.55 2.51 2.56 2.62 ...
 $ CPI         : num  211 211 211 211 211 ...
 $ Unemployment: num  8.11 8.11 8.11 8.11 8.11 ...
 $ month       : Factor w/ 12 levels "1","2","3","4",..: 2 2 2 2 3 3 3 3 4 4 ...
 $ QTR         : Factor w/ 11 levels "2010.1","2010.2",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Quarter     : Factor w/ 12 levels "2010.1","2010.2",..: 1 1 1 1 1 1 1 1 2 2 ...
 $ Semester    : Factor w/ 6 levels "2010.1","2010.2",..: 1 1 1 1 1 1 1 1 1 1 ...
> 
```

Task 1: Utilize variables like date and restructure dates as 1 for 5 Feb 2010 (starting from the earliest date in order).

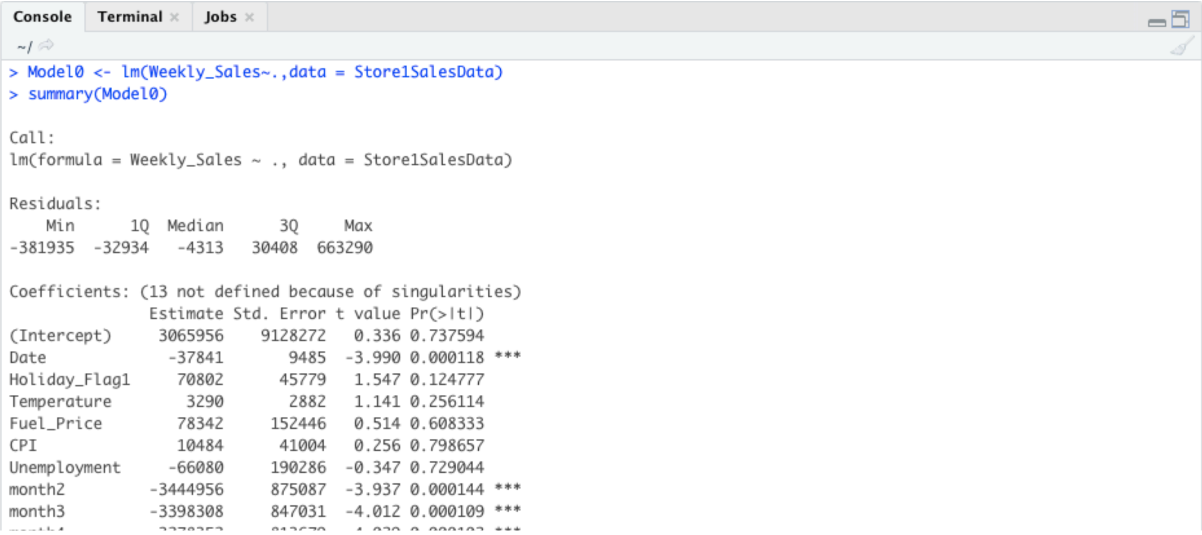
- First convert the Date field to numeric type.
- Then created iterator/counter for the for loop and initialized at value itr = 1. This will both iterate through the entire set of dates/rows in the Store1 data set and help updating the date value as per the count.
- Finally run the for loop for all dates in the Store1 data under the Date variable to update them as integers where the first date 5 Feb 2010 is 1 . Next date 12 Feb 2010 is 2 and so on.



```
> # Step 2 > Restructuring of Dates to a sequence > Update dates as 1,2 ....for 5 Feb 2010 (starting from the earliest date in
 order).
> Store1SalesData$Date <- as.numeric(Store1SalesData$Date)
> itr = 1
> for (date in Store1SalesData$Date) {
+   Store1SalesData$Date[itr]=itr
+   itr=itr+1
+ }
```

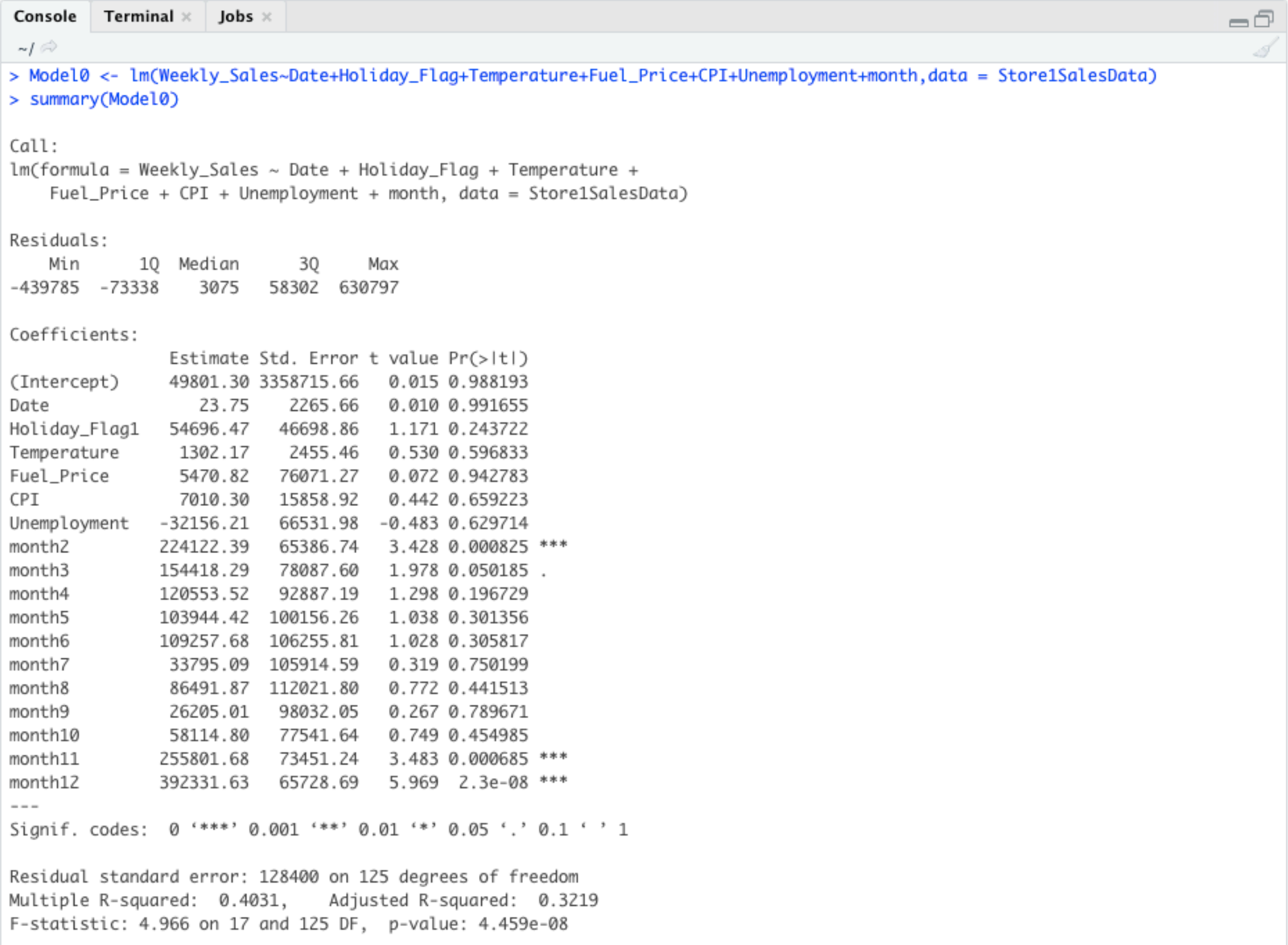
Task 2: Building Linear Regression models and analyse if the models can accurately predict Sales. Steps:

- First build a general model for demand - with all of the variables first and then test various methods to improve models. Created Model0 and checked the summary.



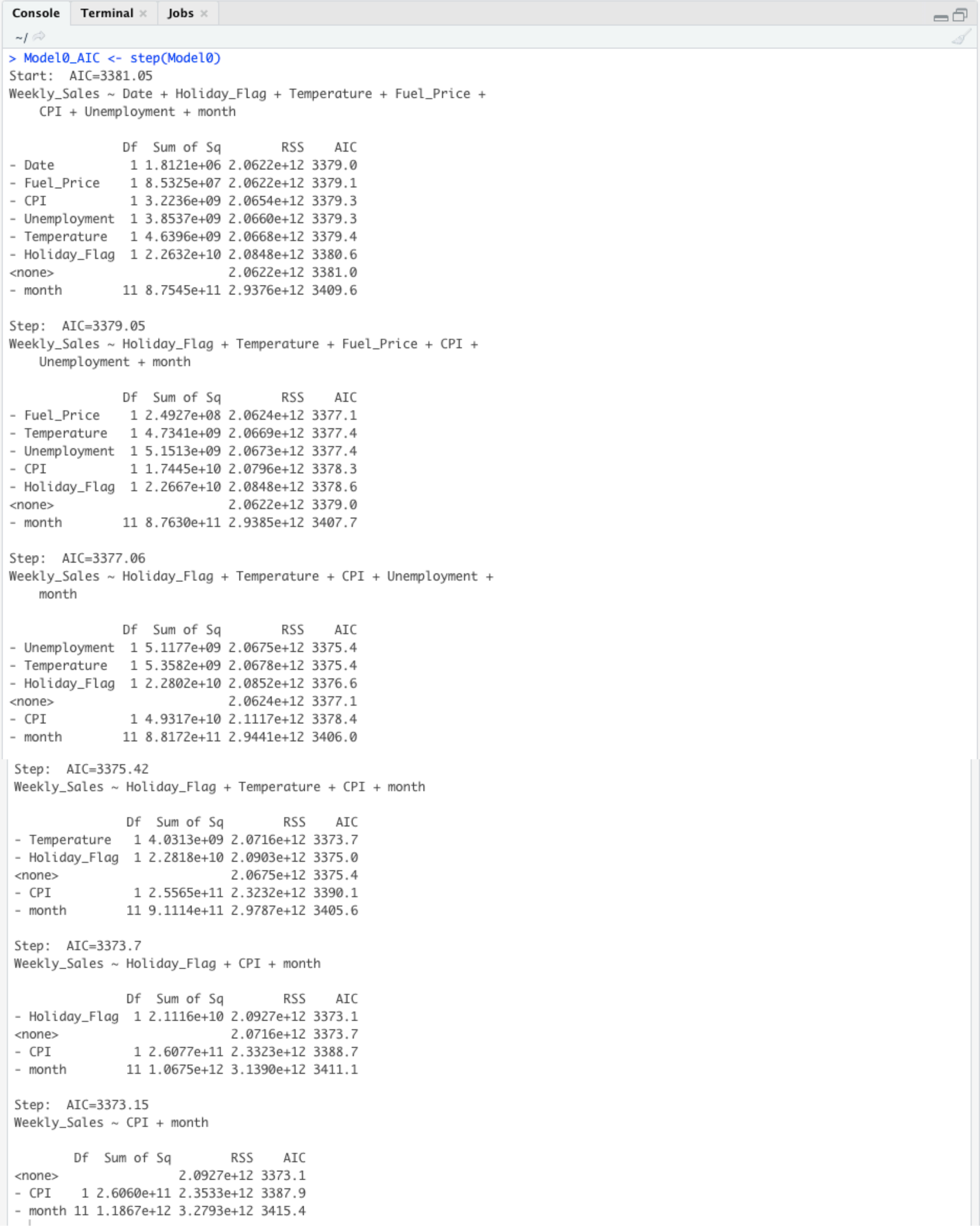
```
> Model0 <- lm(Weekly_Sales~.,data = Store1SalesData)
> summary(Model0)

Call:
lm(formula = Weekly_Sales ~ ., data = Store1SalesData)

Residuals:
    Min      1Q  Median      3Q     Max 
-381935  -32934   -4313   30408  663290 

Coefficients: (13 not defined because of singularities)
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)     3065956    9128272   0.336 0.737594    
Date             -37841       9485  -3.990 0.000118 ***
Holiday_Flag1     70802      45779   1.547 0.124777    
Temperature        3290       2882   1.141 0.256114    
Fuel_Price        78342     152446   0.514 0.608333    
CPI               10484      41004   0.256 0.798657    
Unemployment     -66080     190286  -0.347 0.729044    
month2         -3444956     875087  -3.937 0.000144 ***
month3         -3398308     847031  -4.012 0.000109 ***
```

- As the levels of Semester and Quarter and QTR factor variables depend entirely on Past dates (Quarter/Semesters for years 2010 to 2012 based on the given past data).Further the values of these variables are derived based on the Date variable.
- So for future dates which we would like to predict demand for would never fall in the categories of factor variables Semester, Quarter and QTR which are currently defined. Predictor factor variables (Semester, Quarter and QTR) do not make sense to help predict demand on future data hence are removed from the next revision of the model.
- Revised the model (Model0) . Here formula used was Weekly_Sales ~ Date + Holiday_Flag + Temperature + Fuel_Price + CPI + Unemployment + month. After this Model0 has Adjusted R-squared: 0.3219 but still has variables that are statistically insignificant.



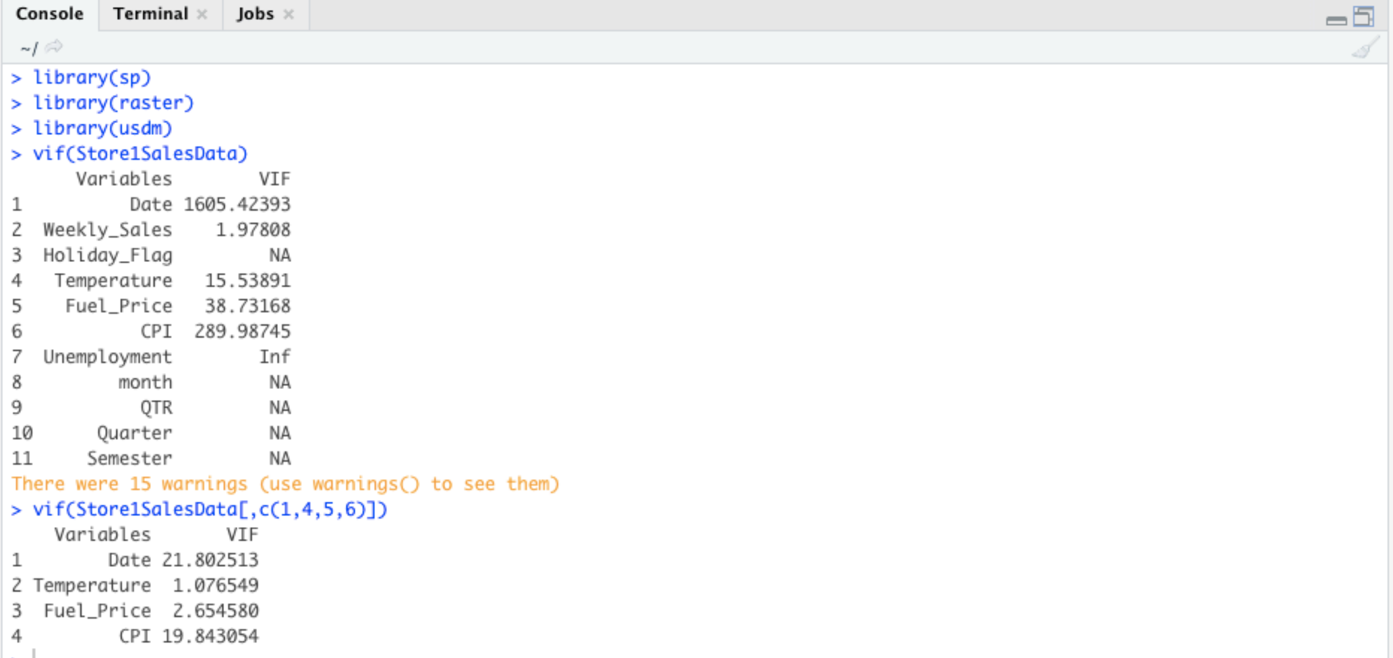
```
> Model0 <- lm(Weekly_Sales~Date+Holiday_Flag+Temperature+Fuel_Price+CPI+Unemployment+month,data = Store1SalesData)
> summary(Model0)

Call:
lm(formula = Weekly_Sales ~ Date + Holiday_Flag + Temperature + 
    Fuel_Price + CPI + Unemployment + month, data = Store1SalesData)

Residuals:
    Min      1Q  Median      3Q     Max 
-439785  -73338    3075   58302  630797 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)     49801.30 3358715.66   0.015 0.988193    
Date               23.75    2265.66   0.010 0.991655    
Holiday_Flag1   54696.47   46698.86   1.171 0.243722    
Temperature      1302.17    2455.46   0.530 0.596833    
Fuel_Price       5470.82   76071.27   0.072 0.942783    
CPI              7010.30   15858.92   0.442 0.659223    
Unemployment   -32156.21   66531.98  -0.483 0.629714    
month2         224122.39   65386.74   3.428 0.000825 ***
month3         154418.29   78087.60   1.978 0.050185 .  
month4         120553.52   92887.19   1.298 0.196729    
month5         103944.42  100156.26   1.038 0.301356    
month6         109257.68  106255.81   1.028 0.305817    
month7          33795.09  105914.59   0.319 0.750199    
month8          86491.87  112021.80   0.772 0.441513    
month9          26205.01   98032.05   0.267 0.789671    
month10         58114.80   77541.64   0.749 0.454985    
month11        255801.68   73451.24   3.483 0.000685 ***
month12        392331.63   65728.69   5.969  2.3e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 128400 on 125 degrees of freedom
Multiple R-squared:  0.4031,	Adjusted R-squared:  0.3219 
F-statistic: 4.966 on 17 and 125 DF,  p-value: 4.459e-08
```

- Next revised the Model using Akaike information criterion (AIC) Statistical algorithm present in R.
- Final model with AIC is Model0_AIC : Weekly_Sales ~ CPI + month. This improved Adjusted R-squared to 0.3384 but model has statistically insignificant levels(month7,month9,month10) of the Month factor at 5% Level of significance.
- However we may consider this a better predictive model than the initial Model0 as other months do significantly determine weekly sales (i.e. The coefficients are Statistically significant at 5% LoS).
- But the overall prediction accuracy for the model is still low(Low R-squared) and we cannot rely on this for making predictions as the predicted values would have high range of variations(Residual errors).

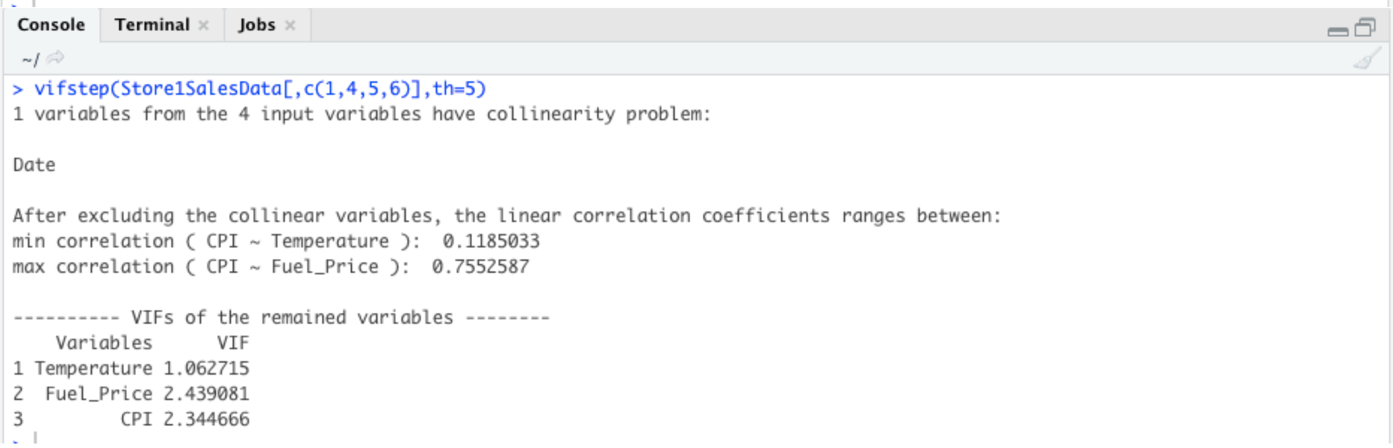


```
> Model0_AIC <- step(Model0)
Start:  AIC=3381.05
Weekly_Sales ~ Date + Holiday_Flag + Temperature + Fuel_Price + 
    CPI + Unemployment + month

               Df  Sum of Sq        RSS    AIC
- Date          1 1.8121e+06 2.0622e+12 3379.0
- Fuel_Price    1 8.5325e+07 2.0622e+12 3379.1
- CPI           1 3.2236e+09 2.0654e+12 3379.3
- Unemployment  1 3.8537e+09 2.0660e+12 3379.3
- Temperature   1 4.6396e+09 2.0668e+12 3379.4
- Holiday_Flag  1 2.2632e+10 2.0848e+12 3380.6
<none>                       2.0622e+12 3381.0
- month        11 8.7545e+11 2.9376e+12 3409.6

Step:  AIC=3379.05
Weekly_Sales ~ Holiday_Flag + Temperature + Fuel_Price + CPI + 
    Unemployment + month

               Df  Sum of Sq        RSS    AIC
- Fuel_Price    1 2.4927e+08 2.0624e+12 3377.1
- Temperature   1 4.7341e+09 2.0669e+12 3377.4
- Unemployment  1 5.1513e+09 2.0673e+12 3377.4
- CPI           1 1.7445e+10 2.0796e+12 3378.3
- Holiday_Flag  1 2.2667e+10 2.0848e+12 3378.6
<none>                       2.0622e+12 3379.0
- month        11 8.7630e+11 2.9385e+12 3407.7

Step:  AIC=3377.06
Weekly_Sales ~ Holiday_Flag + Temperature + CPI + Unemployment + 
    month

               Df  Sum of Sq        RSS    AIC
- Unemployment  1 5.1177e+09 2.0675e+12 3375.4
- Temperature   1 5.3582e+09 2.0678e+12 3375.4
- Holiday_Flag  1 2.2802e+10 2.0852e+12 3376.6
<none>                       2.0624e+12 3377.1
- CPI           1 4.9317e+10 2.1117e+12 3378.4
- month        11 8.8172e+11 2.9441e+12 3406.0

Step:  AIC=3375.42
Weekly_Sales ~ Holiday_Flag + Temperature + CPI + month

               Df  Sum of Sq        RSS    AIC
- Temperature   1 4.0313e+09 2.0716e+12 3373.7
- Holiday_Flag  1 2.2818e+10 2.0903e+12 3375.0
<none>                       2.0675e+12 3375.4
- CPI           1 2.5565e+11 2.3232e+12 3390.1
- month        11 9.1114e+11 2.9787e+12 3405.6

Step:  AIC=3373.7
Weekly_Sales ~ Holiday_Flag + CPI + month

               Df  Sum of Sq        RSS    AIC
- Holiday_Flag  1 2.1116e+10 2.0927e+12 3373.1
<none>                       2.0716e+12 3373.7
- CPI           1 2.6077e+11 2.3323e+12 3388.7
- month        11 1.0675e+12 3.1390e+12 3411.1

Step:  AIC=3373.15
Weekly_Sales ~ CPI + month

        Df  Sum of Sq        RSS    AIC
<none>                2.0927e+12 3373.1
- CPI    1 2.6060e+11 2.3533e+12 3387.9
- month 11 1.1867e+12 3.2793e+12 3415.4
```

- Next we can try and analyse the models for multicollinearity between the variables using VIF. Since the categorical variables have no VIF data(NA). Checking for the remaining variables.
- Next run VIF step algorithm and it eliminates Date which has a collinearity problem using VIF threshold of 5.

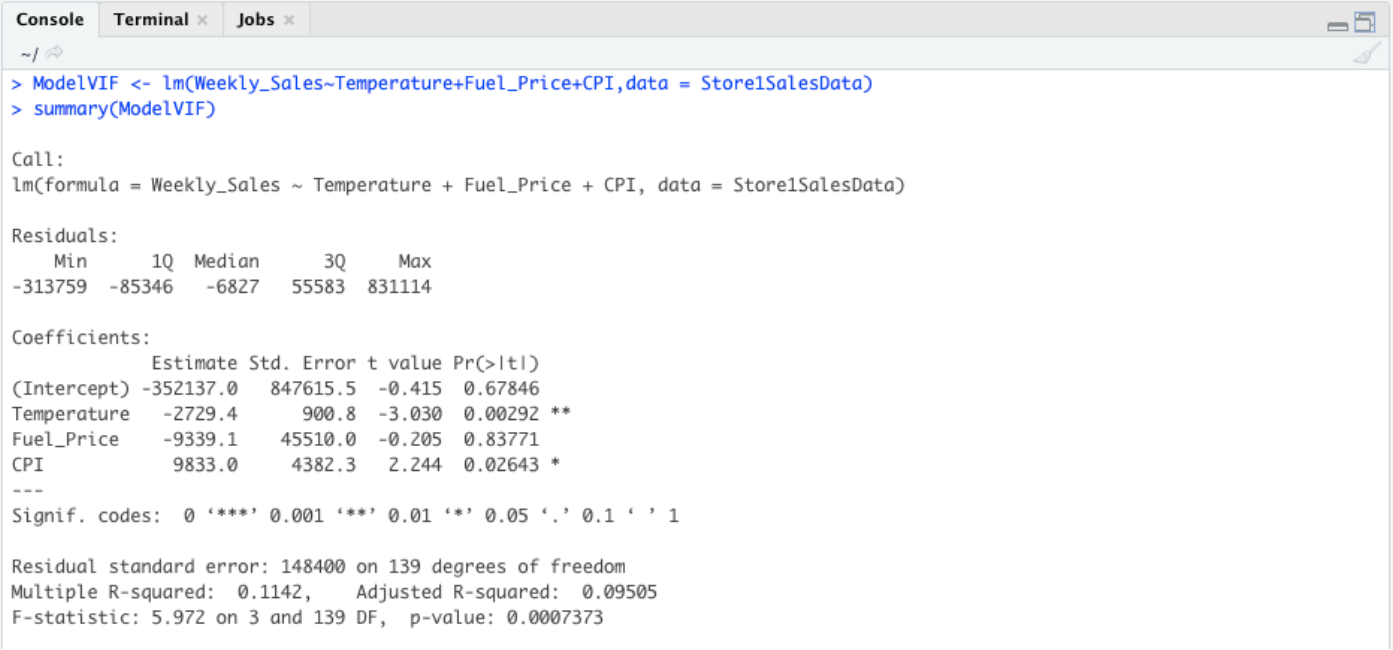


```
> library(sp)
> library(raster)
> library(usdm)
> vif(Store1SalesData)
      Variables        VIF
1          Date 1605.42393
2  Weekly_Sales    1.97808
3  Holiday_Flag         NA
4   Temperature   15.53891
5    Fuel_Price   38.73168
6           CPI  289.98745
7  Unemployment        Inf
8         month         NA
9           QTR         NA
10      Quarter         NA
11     Semester         NA
There were 15 warnings (use warnings() to see them)
> vif(Store1SalesData[,c(1,4,5,6)])
    Variables       VIF
1        Date 21.802513
2 Temperature  1.076549
3  Fuel_Price  2.654580
4         CPI 19.843054
```



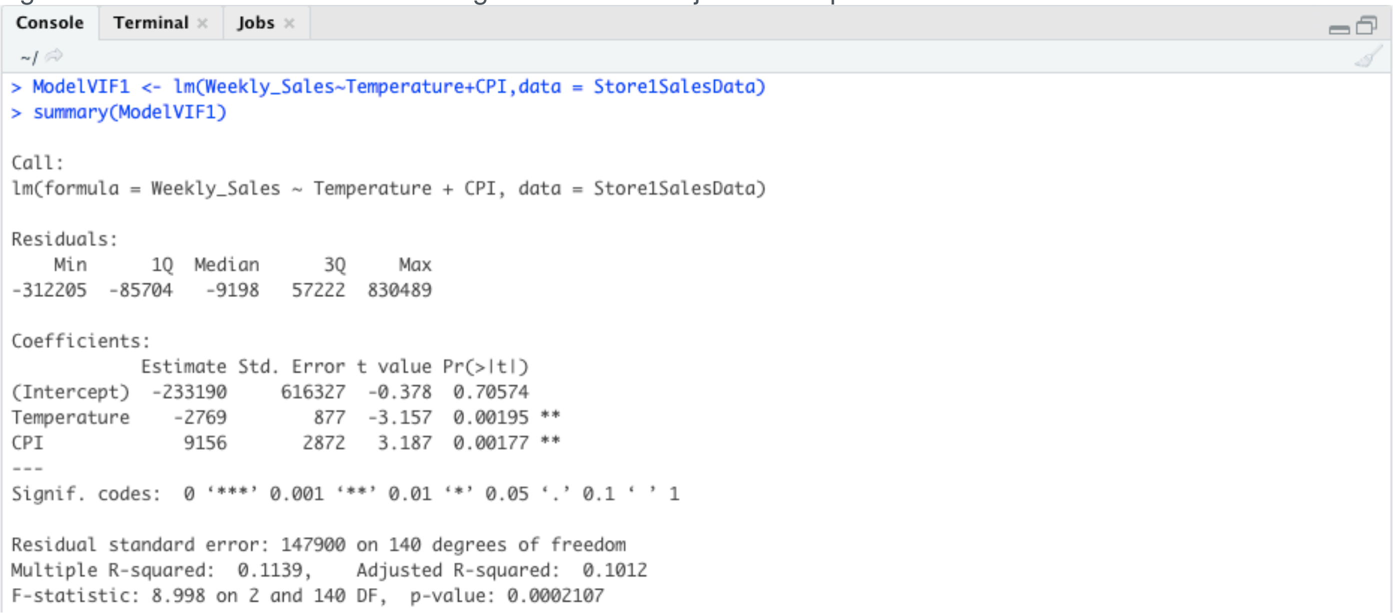
```
> vifstep(Store1SalesData[,c(1,4,5,6)],th=5)
1 variables from the 4 input variables have collinearity problem:

Date

After excluding the collinear variables, the linear correlation coefficients ranges between:
min correlation ( CPI ~ Temperature ):  0.1185033
max correlation ( CPI ~ Fuel_Price ):  0.7552587

---------- VIFs of the remained variables --------
    Variables      VIF
1 Temperature 1.062715
2  Fuel_Price 2.439081
3         CPI 2.344666
```

- This leaves us with 3 variables Temperature, Fuel_Price, CPI. Built model using these variables.
- Fuel price is not statistically significant in the model ModelVIF. Model has an Adjusted R-squared: 0.09505. Which means these variables do not explain most of the variation in Weekly_Sales. So this model needs to be revised



```
> ModelVIF <- lm(Weekly_Sales~Temperature+Fuel_Price+CPI,data = Store1SalesData)
> summary(ModelVIF)

Call:
lm(formula = Weekly_Sales ~ Temperature + Fuel_Price + CPI, data = Store1SalesData)

Residuals:
    Min      1Q  Median      3Q     Max 
-313759  -85346   -6827   55583  831114 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)   
(Intercept) -352137.0   847615.5  -0.415  0.67846   
Temperature   -2729.4      900.8  -3.030  0.00292 **
Fuel_Price    -9339.1    45510.0  -0.205  0.83771   
CPI            9833.0     4382.3   2.244  0.02643 * 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 148400 on 139 degrees of freedom
Multiple R-squared:  0.1142,	Adjusted R-squared:  0.09505 
F-statistic: 5.972 on 3 and 139 DF,  p-value: 0.0007373
```

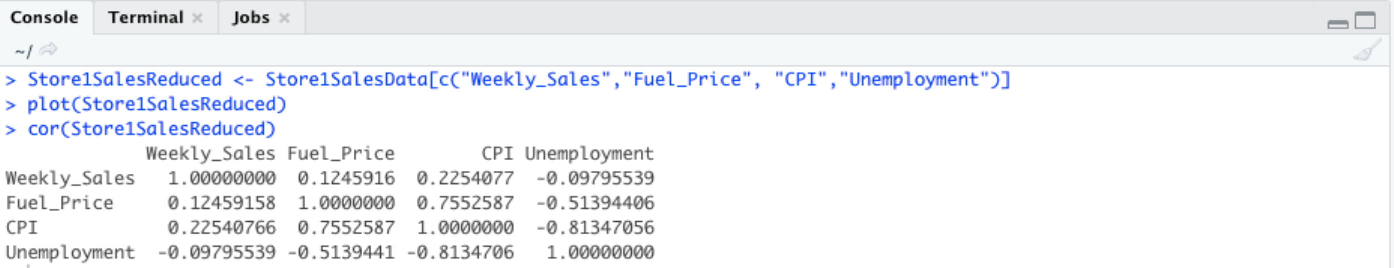
- Building a revised model based using remaining variables Temperature and CPI.
- Removing Fuel price yields ModelVIF1 with where the remaining variable coefficients are statistically significant at 1% LoS. But the model again has a low Adjusted R-squared: 0.1012



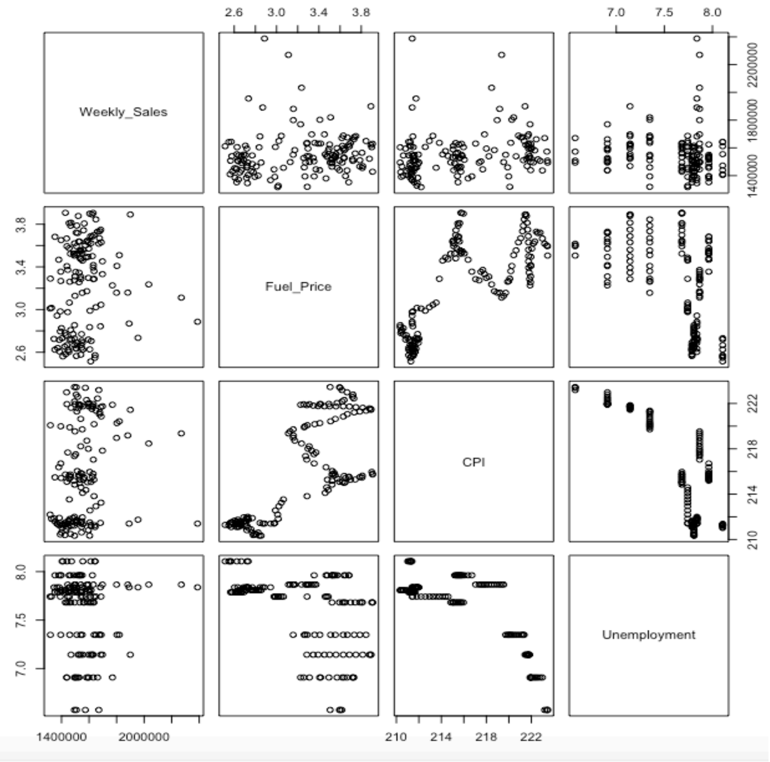
```
> ModelVIF1 <- lm(Weekly_Sales~Temperature+CPI,data = Store1SalesData)
> summary(ModelVIF1)

Call:
lm(formula = Weekly_Sales ~ Temperature + CPI, data = Store1SalesData)

Residuals:
    Min      1Q  Median      3Q     Max 
-312205  -85704   -9198   57222  830489 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept)  -233190     616327  -0.378  0.70574   
Temperature    -2769        877  -3.157  0.00195 **
CPI             9156       2872   3.187  0.00177 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 147900 on 140 degrees of freedom
Multiple R-squared:  0.1139,	Adjusted R-squared:  0.1012 
F-statistic: 8.998 on 2 and 140 DF,  p-value: 0.0002107
```

Task 3: Testing the Hypothesis if CPI, unemployment, and fuel price have any impact on sales.
Steps:

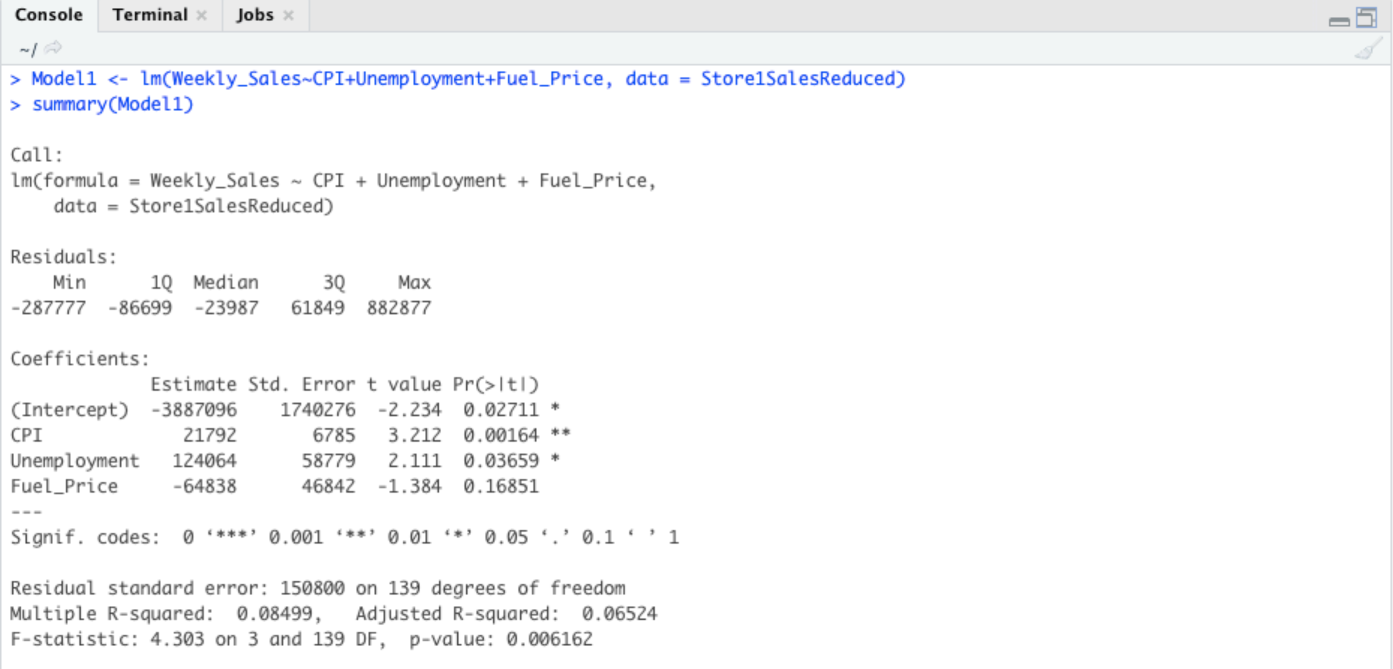
- For this our Null Hypothesis : Weekly_Sales is not affected by CPI, unemployment, and fuel price.
- Removing the variables other than Weekly_Sales, CPI, unemployment, and fuel price.
- Doing some basic statistical exploration on the reduced data by checking the scatter plot and correlations
- Plots revealed no specific trend/pattern for Weekly sales based on the predictor variables. The plots do not highlight a clear relation. Further the correlation matrix also confirms a weak correlation between Weekly sales and independent variables CPI, unemployment, and fuel price.



```
> Store1SalesReduced <- Store1SalesData[c("Weekly_Sales","Fuel_Price", "CPI","Unemployment")]
> plot(Store1SalesReduced)
> cor(Store1SalesReduced)
             Weekly_Sales Fuel_Price        CPI Unemployment
Weekly_Sales   1.00000000  0.1245916  0.2254077  -0.09795539
Fuel_Price     0.12459158  1.0000000  0.7552587  -0.51394406
CPI            0.22540766  0.7552587  1.0000000  -0.81347056
Unemployment  -0.09795539 -0.5139441 -0.8134706   1.00000000
```



- So as per this initial analysis there is a low chance of these variables (CPI, unemployment, and fuel price) being accurate predictors for the target variable (Weekly_Sales)
- Build the model(Model1) for Weekly sales using predictors CPI, unemployment, and fuel price. Review the model summary.



```
> Model1 <- lm(Weekly_Sales~CPI+Unemployment+Fuel_Price, data = Store1SalesReduced)
> summary(Model1)

Call:
lm(formula = Weekly_Sales ~ CPI + Unemployment + Fuel_Price,
    data = Store1SalesReduced)

Residuals:
    Min      1Q  Median      3Q     Max
-287777  -86699  -23987   61849  882877

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3887096    1740276  -2.234  0.02711 *
CPI             21792       6785   3.212  0.00164 **
Unemployment   124064      58779   2.111  0.03659 *
Fuel_Price     -64838      46842  -1.384  0.16851
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 150800 on 139 degrees of freedom
Multiple R-squared:  0.08499,	Adjusted R-squared:  0.06524
F-statistic: 4.303 on 3 and 139 DF,  p-value: 0.006162
```

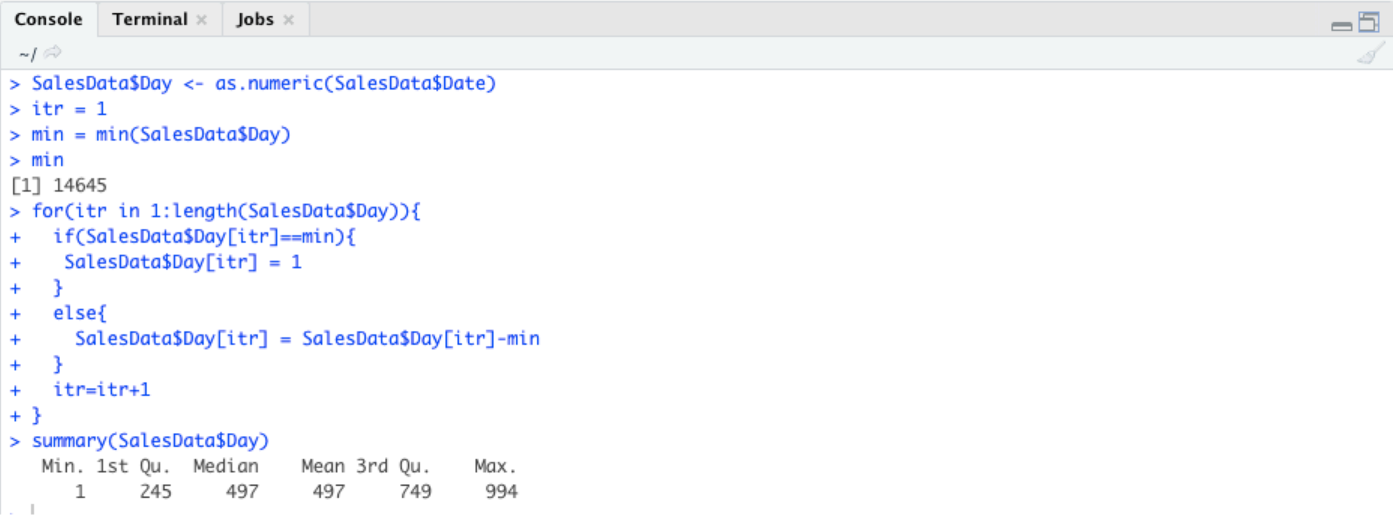
- Summary on Model1 reveals Weekly sales cannot be well explained by CPI, unemployment, and fuel price
- Model has Adjusted R-squared: 0.06524. The Adjusted R-Squared is just 6% which implies only 6% of the variation of weekly sales is effectively explained by CPI, unemployment, and fuel price. Further Fuel price is highly insignificant to the model with a p-value of 0.16851.Unemployment and CPI are statistically significant in terms of their impact on Weekly_Sales
- Since the R-squared value for this model is very low, it would not be recommended for making Sales/Demand predictions.

Note: Revising the model by dropping Fuel Price only decreases the R-squared even more.

Conclusion: Out of CPI, unemployment, and fuel price. Fuel price does not have a significant impact or effect on the target variable (Weekly_Sales).
CPI and Unemployment have statistical significance and their coefficients have a low p-value. However with the model having a very low adjusted R-squared (0.06524) this means they can explain very little of the variation in weekly sales. So while they do have an impact on sales which is explained by the low p-value of coefficient. The impact is very weak or low and cannot successfully help predict the target variable.

Task 4: Change dates into days by creating new variable.
Converting the Dates into Days. Since there is no definition if we are looking for weekday conversion or day of the year. We will convert the dates into days based on the number of days for which the data was observed in the dataset.
I.e. Mapping the first date 2010-02-05 to 1 and then successive observation days are tagged based on number of days after the first day. So the next Date 2010-02-12 will be day 7 and so on.
Steps:

- First create the new Day variable and assign it as numeric values of the Date.
- Next create an iterator for the For loop . itr = 1
- Set the min date to be compare or adjust all dates with . The first date 2010-02-05 of the data set is our min and will be tagged as 1.
- Next build the for loop with logic. If the Date is the first date then this will be tagged as Day = 1. Else for all other dates we want Day = Current Date – First Date(2010-02-05)
- The first observation date 2010-02-05 is now tagged as 1 and the last date in the observation 2012-10-26 is tagged as Day number 994.



```
> SalesData$Day <- as.numeric(SalesData$Date)
> itr = 1
> min = min(SalesData$Day)
> min
[1] 14645
> for(itr in 1:length(SalesData$Day)){
+   if(SalesData$Day[itr]==min){
+     SalesData$Day[itr] = 1
+   }
+   else{
+     SalesData$Day[itr] = SalesData$Day[itr]-min
+   }
+   itr=itr+1
+ }
> summary(SalesData$Day)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
      1     245     497     497     749     994 
```

6. Conclusion:

1. Based on the analysis and models built for Store #1 we see that none of the variables in the data captured reflect as accurate predictors of Sales/Demand.
2. Suggest that business could try to build and test models for other stores randomly to again verify if there is no strong predictors for Sales. As the analysis of sales data store by store revealed that the stores have some differences in terms of average sales volume as well as Variability of sales. Also stores located in different regions are not always fully homogeneous.
3. We also see that month appears to impact the sales quite well as there were some trends seen in the month-wise sales analysis.
4. The hypothesis tested for CPI, Unemployment and Fuel price revealed again that which economic factors like CPI and unemployment do show some impact on sales. They cannot accurately help predict the sales.
5. Holiday v/s non-holiday sales comparison showed that for most Holidays except Christmas the Average Sales are higher. While this was a good study if the overall revenue was under consideration. As suggested this comparison is limited when trying to analyse units sold due to the fact that the prices of goods during the holidays. Since the date available is only in amount sold in dollar value. This doesn't help identify the increase in number of units of stock/goods that were sold due to the discounts provided in the Holidays. So from an inventory perspective to predict the number of units which Walmart would need to keep in stock this does not help provide any clear insight. Although this could be assessed if the discount rates were known for various categories.
6. Finally to conclude. For Walmart to better predict the sales/demand and create a model to predict the inventory requirements to prevent stockouts. The model would need to consider or incorporate other variables which may have more significant impact on sales. These could be for example. Season of the year, Median neighbourhood incomes, Inflation rates etc. which could affect sales of some or many products. Also analysis of products based on categories is recommended as not all products in a supermarket have the same sales/demand requirements all throughout the year.
   Walmart could try to focus on specific major product categories and then try to see if they can create more accurate predictions for the demand for the categories alone. This could help in ensuring that demand for such products can be predicted well and stock can be maintained according to the anticipated demand.