Walmart Sales Forecast

Final Project – Data Analysis for Data Science

Jack Fitzgibbons

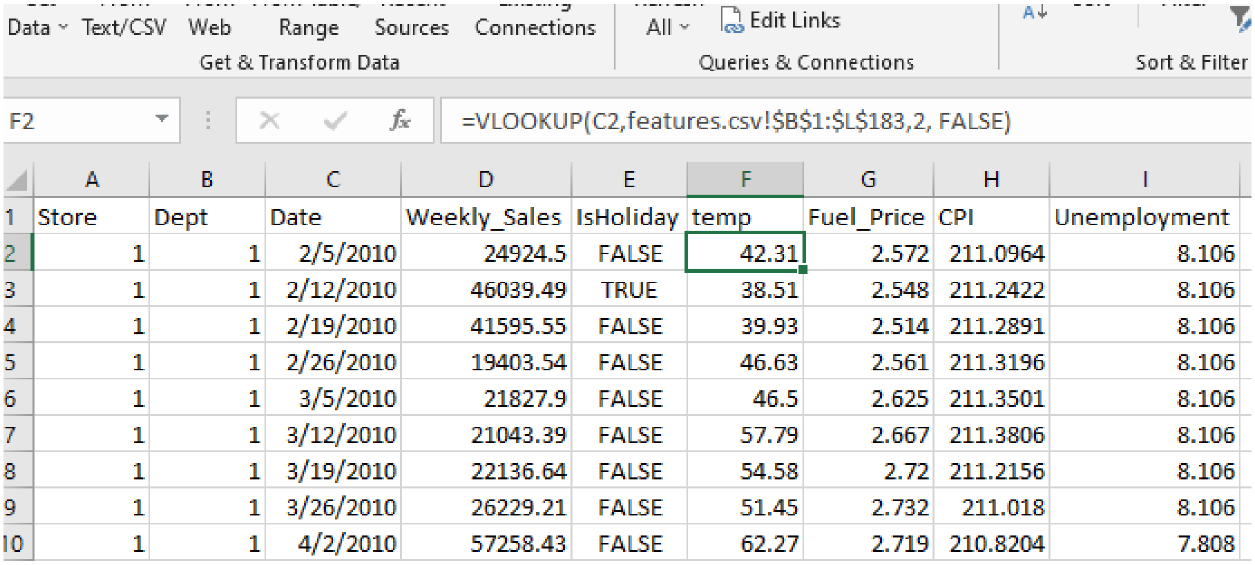
Introduction

I chose to analyze data sets about Walmart sales data. I found these data sets through one of the suggested links in the project description, which brought me to the website Kaggle where the downloadable spreadsheets are available. It looks like this data was posted by Walmart back in 2014 in order to identify potential talent for recruitment. I thought this data was interesting, and I wanted to see if the techniques that I learned with R this semester could be successful as providing some insight into forecasting sales numbers for different Walmart stores.

Link to data: https://www.kaggle.com/competitions/walmart-recruiting-store-sales-forecasting/data?select=features.csv.zip

Data Selection/Data Cleaning

The Kaggle website gave four spreadsheets in comma delimited format titled features, stores, test, and train. The features dataset gave information in the start date for the week of sales, the average temperature, the cost of fuel in the region, markdown data which has information on products that are on sale or coupons, the consumer price index, the unemployment rate, and whether that week is a special holiday week. The stores data set included information on the store number, the type of store, and the size of the store. The train dataset included the store number, the department number, the start date for the week, the weekly sales amount, and whether the week is a special holiday week. The test data set was identical to the train dataset so I chose to omit it. The data included 45 different stores, and 99 different departments. This meant that the original train dataset was 40,000 rows long, and the features dataset was about 8,000 rows long. I chose to analyze data only from store one. I made this decision using the stores spreadsheet, and finding the average store size which was 130,287.60. Store 1 had a store size of 151,315. Since store 1's size was close to the average of all stores, I chose to just focus on store 1. This significantly shrunk the data down, but it was still over 10,000 rows since there are 99 different departments in each store, and the sales data includes one row for every week from 2010 to 2013. Therefore, I chose to analyze store 1 data for departments 1 through 5. Next, I used the features dataset and the train dataset to combine the data into one csv file. I chose to ignore the markdown information since it would make things much more complicated. I needed to bring temp, fuel price, CPI, and Unemployment over from the features file into the train file. To do this I used a simple VLOOKUP which worked because the date variable is essentially a unique identifier for that specific week. So every department has the same values for these variables within that specific week. Below is a snapshot of the VLOOKUP formula that I used, and what the final csv file looked like.



F2

```
=VLOOKUP(C2,features.csv!$B$1:$L$183,2, FALSE)
```

	A	B	C	D	E	F	G	H	I
1	Store	Dept	Date	Weekly_Sales	IsHoliday	temp	Fuel_Price	CPI	Unemployment
2	1	1	2/5/2010	24924.5	FALSE	42.31	2.572	211.0964	8.106
3	1	1	2/12/2010	46039.49	TRUE	38.51	2.548	211.2422	8.106
4	1	1	2/19/2010	41595.55	FALSE	39.93	2.514	211.2891	8.106
5	1	1	2/26/2010	19403.54	FALSE	46.63	2.561	211.3196	8.106
6	1	1	3/5/2010	21827.9	FALSE	46.5	2.625	211.3501	8.106
7	1	1	3/12/2010	21043.39	FALSE	57.79	2.667	211.3806	8.106
8	1	1	3/19/2010	22136.64	FALSE	54.58	2.72	211.2156	8.106
9	1	1	3/26/2010	26229.21	FALSE	51.45	2.732	211.018	8.106
10	1	1	4/2/2010	57258.43	FALSE	62.27	2.719	210.8204	7.808

Questions

I decided to focus firstly on the differences in weekly sales between the 5 departments, and to get a general understanding of the data. Furthermore, the specific questions that I wanted to answer are:

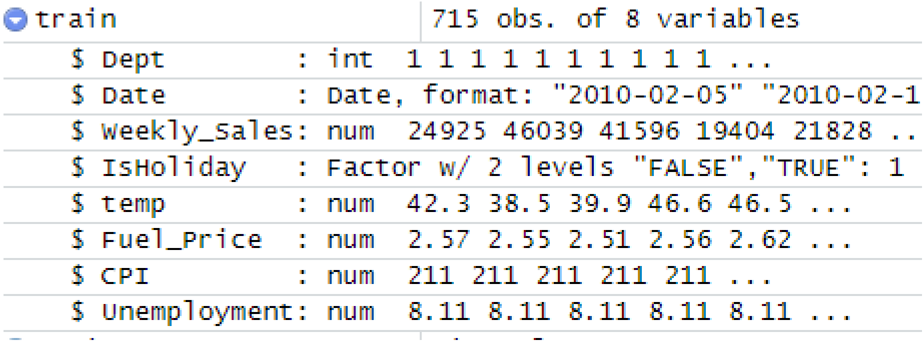
What does a prediction for weekly sales look like for the next two years?

Which variable from the dataset is the best predictor, or has the best relationship with weekly sales?

Which holiday period shows the best sales spikes?

Data Structures

Below is a list of the data structure for each variable. I had to change Date from character format to a Date format which can be seen in the R code under the first read csv line. I also had to change IsHoliday to a factor of 1 for TRUE and 2 for False.

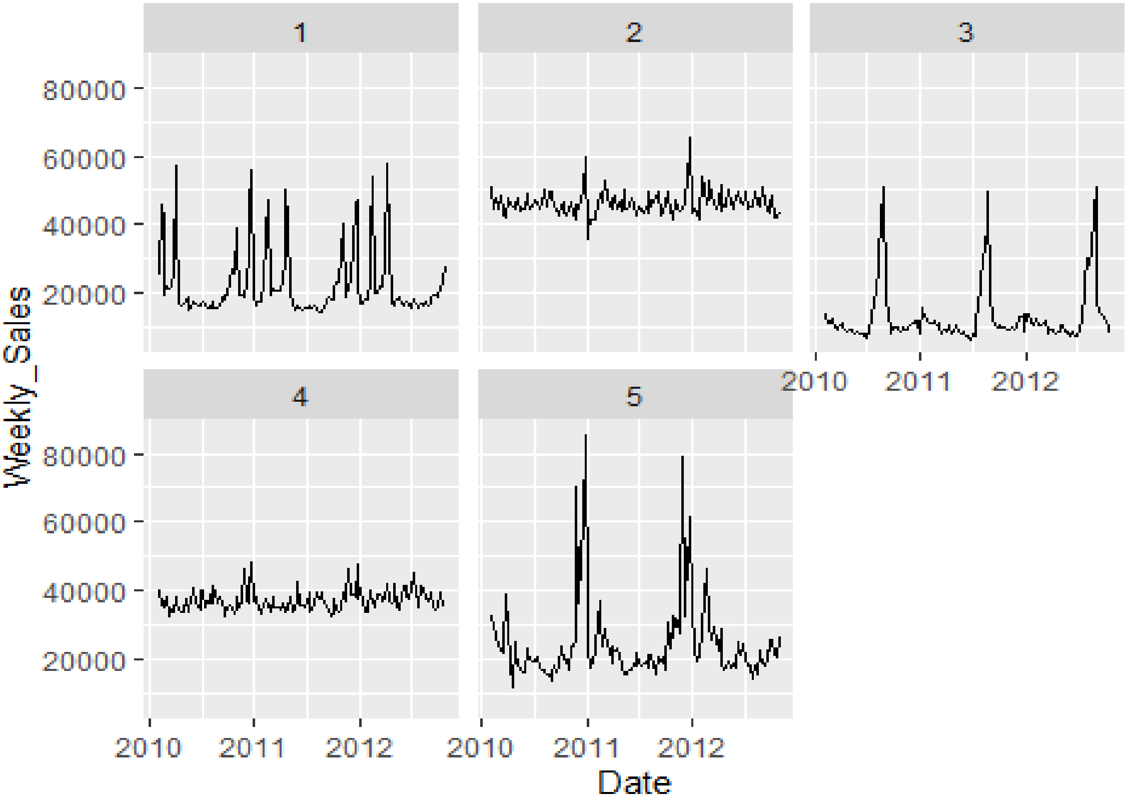


```
train                          715 obs. of 8 variables
    $ Dept        : int  1 1 1 1 1 1 1 1 1 1 ...
    $ Date        : Date, format: "2010-02-05" "2010-02-1
    $ Weekly_Sales: num  24925 46039 41596 19404 21828 ..
    $ IsHoliday   : Factor w/ 2 levels "FALSE","TRUE": 1
    $ temp        : 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 ...
```

```
library(ggplot2)

library(rpart)
train <- read.csv("C:\\Users\\Fitzg\\OneDrive\\Documents\\train.csv", header=
TRUE)

train$Date <- as.Date(train$Date)
```

Analysis

```
ggplot(train, aes(Date, Weekly_Sales)) + geom_line() +facet_wrap(~Dept)
```



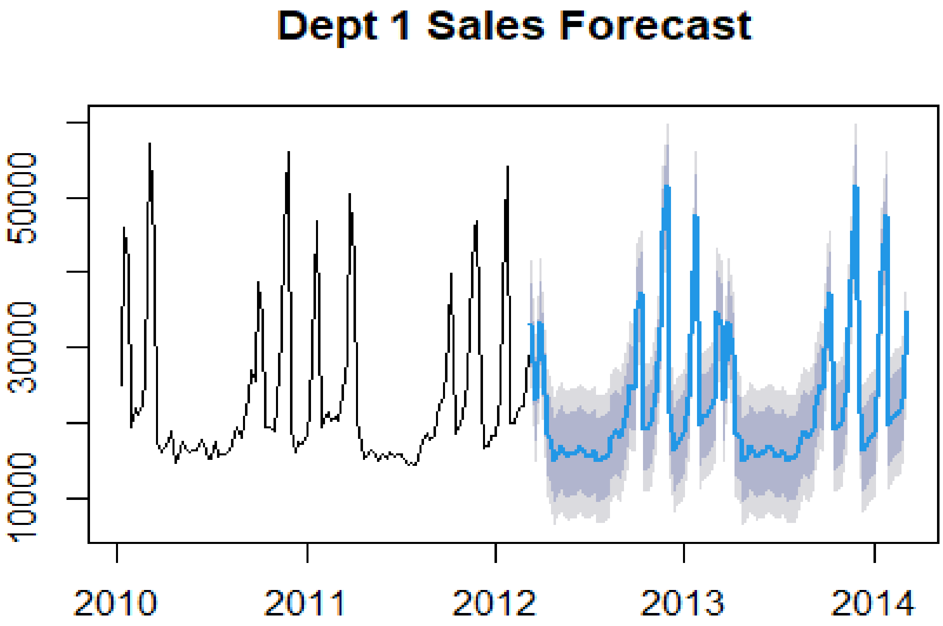
Using the R package ggplot2, I first graphed each department's weekly sales date on a line graph as shown above. You can clearly see that each graph is much different than the others. Graph 2 and 4 seem to have much smaller spikes in sales thought the three years, while 1, 3, and 5 have significantly large spikes. Graph 5 in particular seems to spike towards the end of a year, and quickly drops at the start of the next year. Unfortunately, the data does not specify which department is which, they are only labeled as numbers. But one could hypothesize that department 5 is popular around that time of year due to holidays like Christmas, Hanukkah, and Thanksgiving. This initial graphing of the weekly sales data between departments gave me a solid understanding of the differences in sales quantities before I did further analysis.

```r
train$IsHoliday <- as.factor(train$IsHoliday)
model <- glm(Weekly_Sales ~+IsHoliday+temp+Fuel_Price+CPI+Unemployment, data
= train)
summary(model)
```

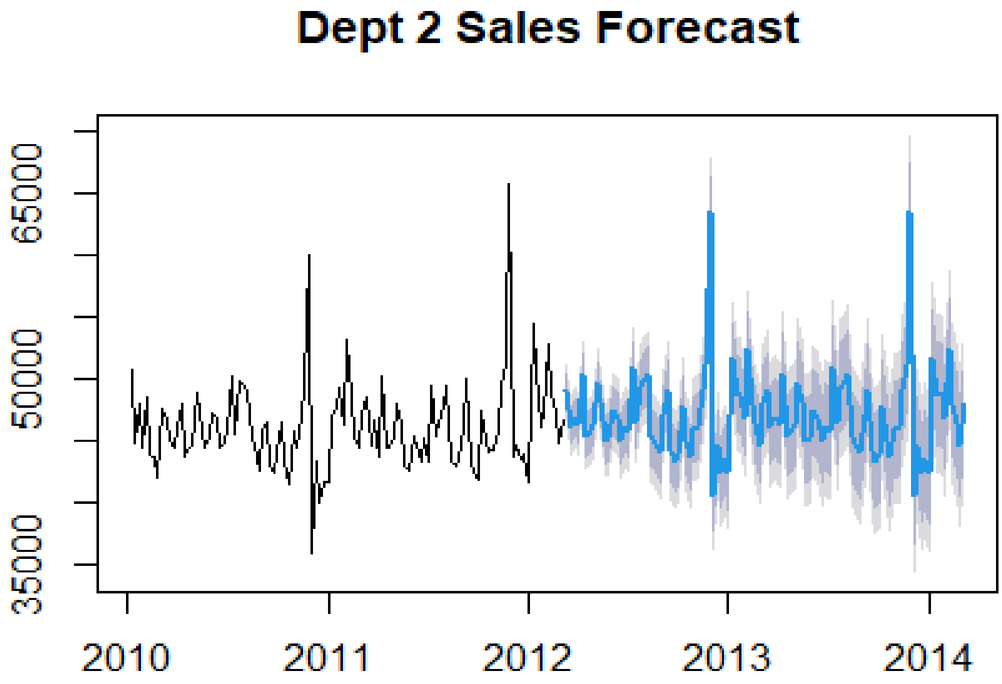
```
## 
## Call:
## glm(formula = Weekly_Sales ~ +IsHoliday + temp + Fuel_Price + 
##     CPI + Unemployment, data = train)
## 
## Deviance Residuals: 
##    Min      1Q  Median      3Q     Max  
## -26386  -11058   -2397   11874   56606  
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)  
## (Intercept)   -91122.74   75001.63  -1.215   0.2248  
## IsHolidayTRUE   1928.45    2110.97   0.914   0.3613  
## temp             -85.00      39.45  -2.155   0.0315 *
## Fuel_Price     -2311.75    2025.28  -1.141   0.2541  
## CPI              497.18     290.35   1.712   0.0873 .
## Unemployment    3343.38    2512.66   1.331   0.1837  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 196321174)
## 
##     Null deviance: 1.4162e+11  on 714  degrees of freedom
## Residual deviance: 1.3919e+11  on 709  degrees of freedom
## AIC: 15690
## 
## Number of Fisher Scoring iterations: 2
```

Next, I decided to run a generalized linear model to try to identify which variable has the best relationship with weekly sales. The results of the regression, shown above, depict all 5 variables, and their relationship with weekly sales. The P values aren’t terribly large which is a good sign. According to R, temperature and CPI are the best variables is relation to weekly sales when looking at the p values. The deviance residuals are somewhat closely mirrored which is a good sign. Running another glm model with just temp and CPI returned better results, but not a big enough difference versus the original model. I explored these variables relationships further with a bagging model depicted later in my analysis.

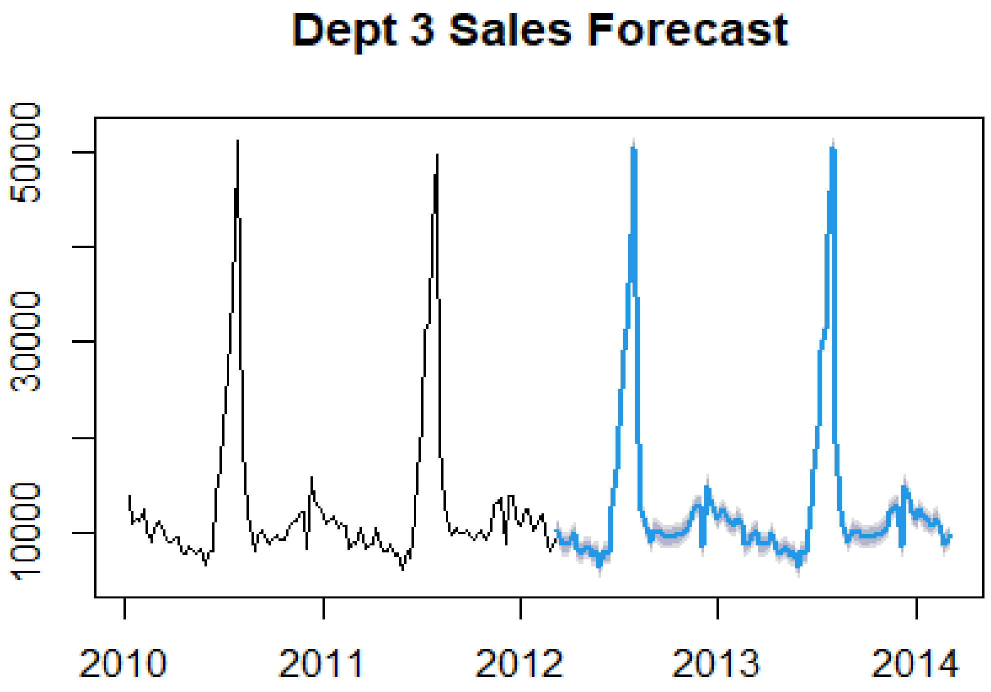
```
dept_1 <- train %>%
  filter(train$Dept == 1)

train.ts <- ts(dept_1$Weekly_Sales, frequency = 52, start = c(2010, 2), end =
c(2012, 10))
train.forecast <- forecast(train.ts)
plot(train.forecast, main = "Dept 1 Sales Forecast")
```



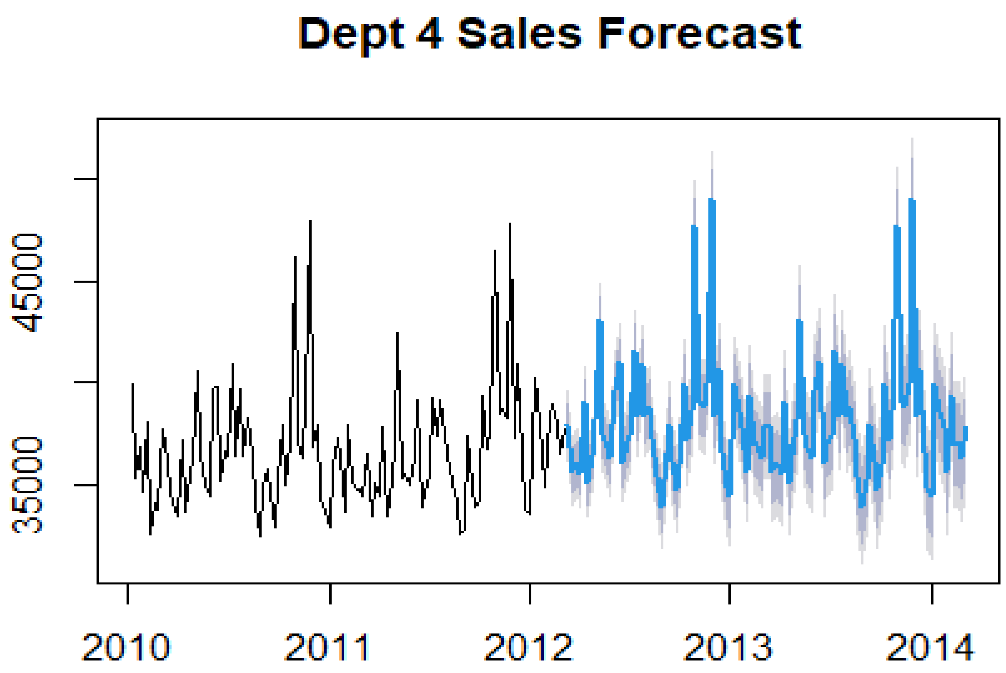
```r
dept_2 <- train %>%
  filter(train$Dept == 2)

train.ts2 <- ts(dept_2$Weekly_Sales, frequency = 52, start = c(2010, 2), end
= c(2012, 10))
train.forecast2 <- forecast(train.ts2)
plot(train.forecast2, main = "Dept 2 Sales Forecast")
```



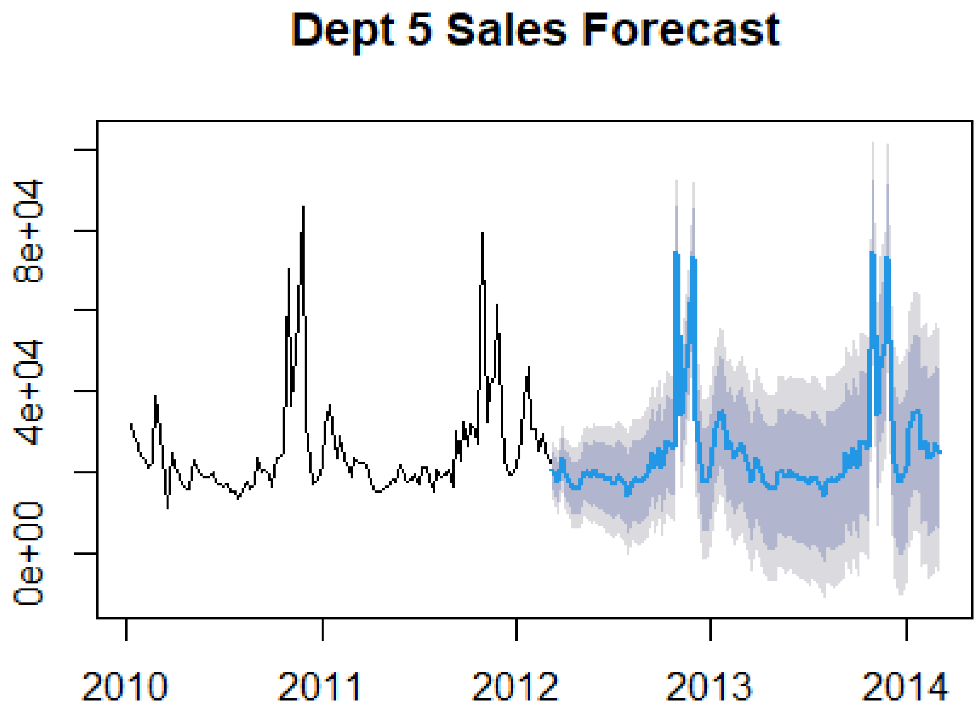
```r
dept_3 <- train %>%
  filter(train$Dept == 3)

train.ts3 <- ts(dept_3$Weekly_Sales, frequency = 52, start = c(2010, 2), end
= c(2012, 10))
train.forecast3 <- forecast(train.ts3)
plot(train.forecast3, main = "Dept 3 Sales Forecast")
```



```
dept_4 <- train %>%
  filter(train$Dept == 4)

train.ts4 <- ts(dept_4$Weekly_Sales, frequency = 52, start = c(2010, 2), end
= c(2012, 10))
train.forecast4 <- forecast(train.ts4)
plot(train.forecast4, main = "Dept 4 Sales Forecast")
```



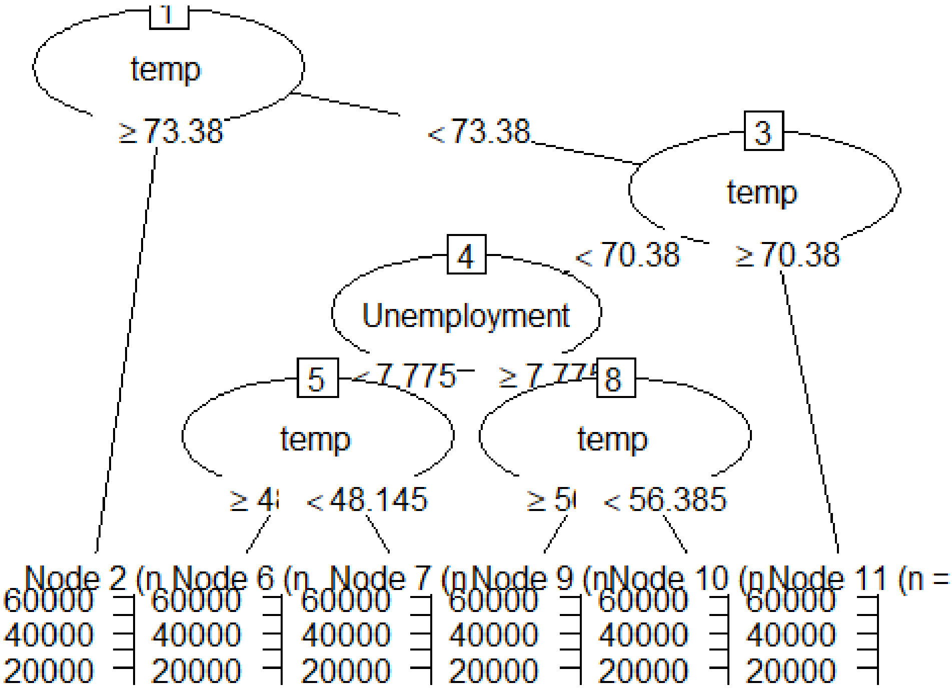
```
dept_5 <- train %>%
  filter(train$Dept == 5)

train.ts5 <- ts(dept_5$Weekly_Sales, frequency = 52, start = c(2010, 2), end
= c(2012, 10))
train.forecast5 <- forecast(train.ts5)
plot(train.forecast5, main = "Dept 5 Sales Forecast")
```



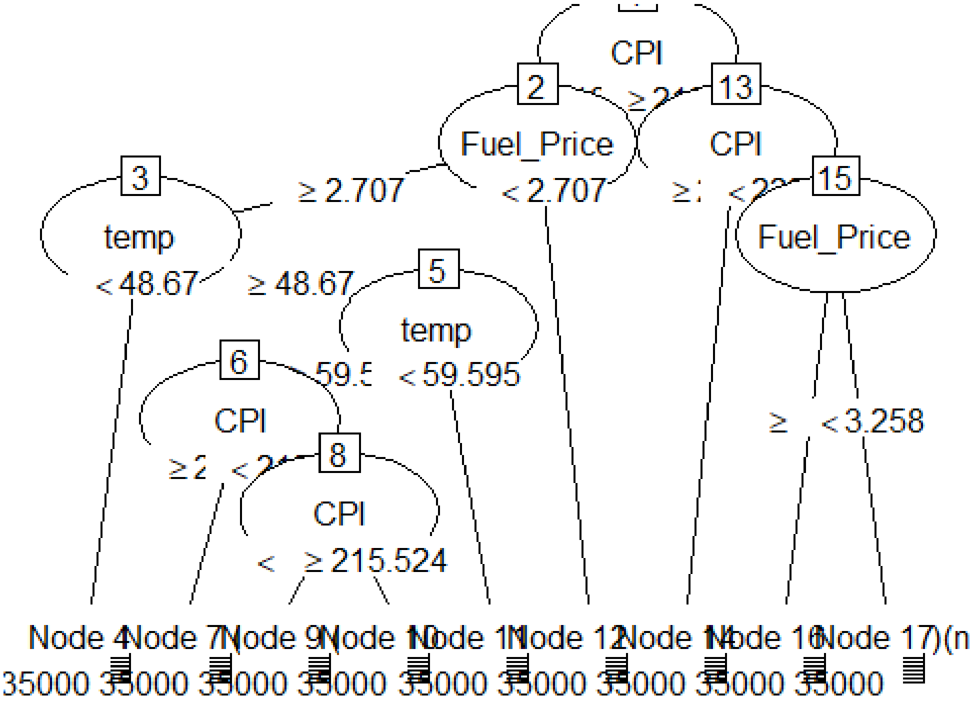
Next, I chose to use the time series model to try to predict the upcoming sales for the years 2013 to 2014 for each department. The results from R are shown above. I was a little hesitant about these results as each graph seems to just copy the same or similar results from the previous years, with the room for error highlighted in gray. However, they are almost mirrored perfectly with each model which makes me question their accuracy. It does answer my question about how sales might look in the future, and the results do make some sense since the years 2010 through 2012 look very similar with the data that we do have. For example, department 3 shows almost the exact same sales numbers each year, and when a spike will occur.

Bagging

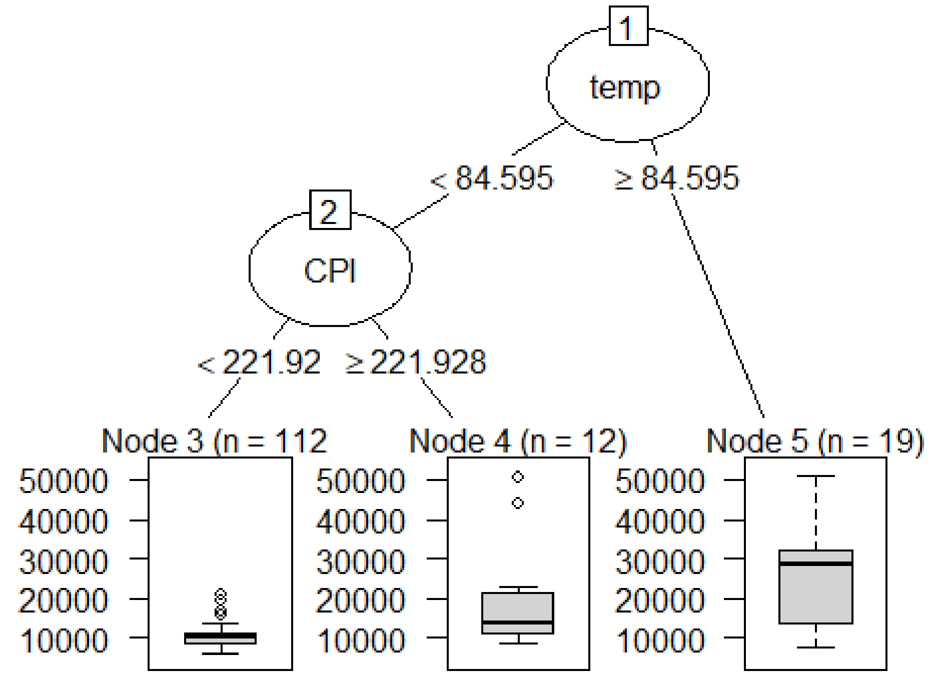
```
set.seed(8675309)
train.ct <- rpart (Weekly_Sales ~ +IsHoliday+temp+Fuel_Price+CPI+Unemployment
, data = dept_1)
train.ct.party <- as.party(train.ct)
plot (train.ct.party)
```



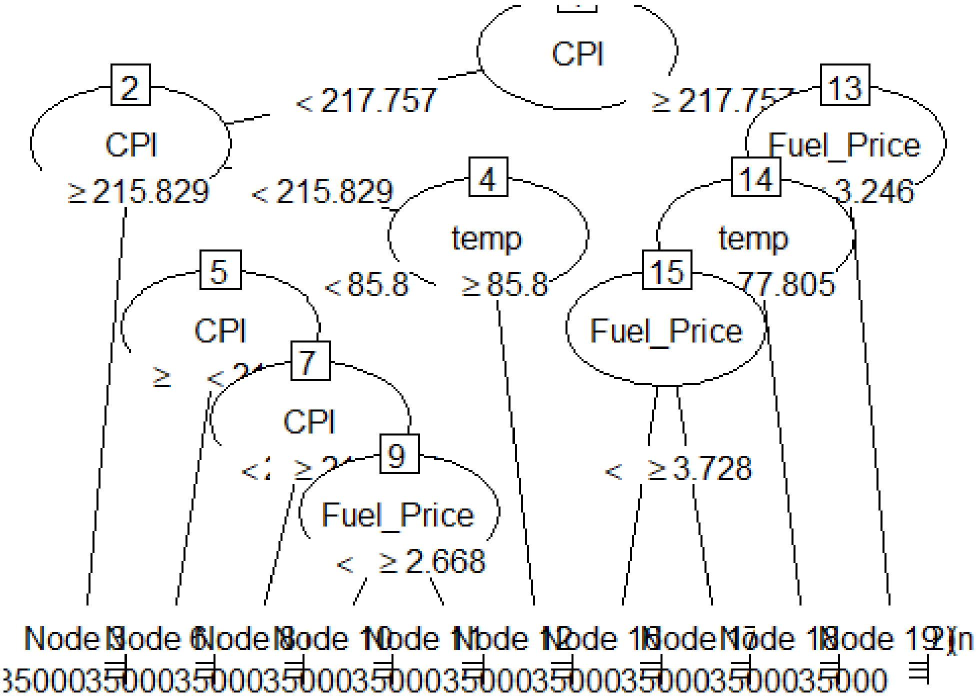
```
set.seed(8675309)
train.ct <- rpart (Weekly_Sales ~ +IsHoliday+temp+Fuel_Price+CPI+Unemployment
, data = dept_2)
train.ct.party <- as.party(train.ct)
plot (train.ct.party)
```



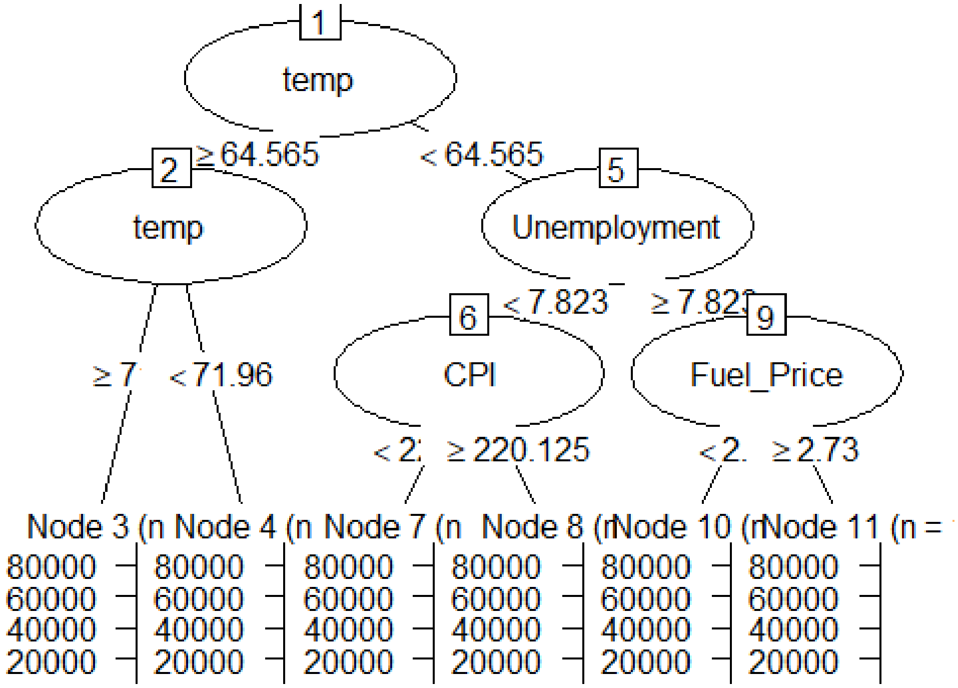
```
set.seed(8675309)
train.ct <- rpart (Weekly_Sales ~ +IsHoliday+temp+Fuel_Price+CPI+Unemployment
, data = dept_3)
train.ct.party <- as.party(train.ct)
plot (train.ct.party)
```



```
set.seed(8675309)
train.ct <- rpart (Weekly_Sales ~ +IsHoliday+temp+Fuel_Price+CPI+Unemployment
, data = dept_4)
train.ct.party <- as.party(train.ct)
plot (train.ct.party)
```

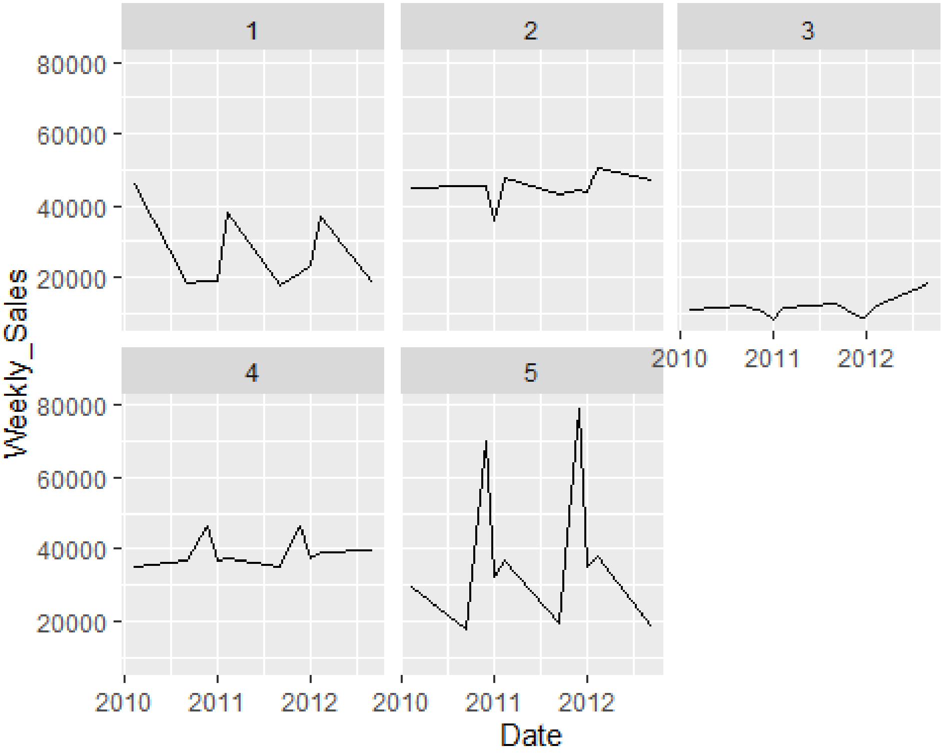


```r
set.seed(8675309)
train.ct <- rpart (Weekly_Sales ~ +IsHoliday+temp+Fuel_Price+CPI+Unemployment
, data = dept_5)
train.ct.party <- as.party(train.ct)
plot (train.ct.party)
```



As I mentioned earlier, I decided to run a bagging model for each department to try to further identify the variables that have the best relationship with weekly sales. Each model is slightly different, which makes sense, but Temp and CPI are considered two of the best variables which supports our regression model from earlier. Fuel price also came up as a top variable in departments 2 and 4, while unemployment showed up as the second best behind temperature in department 5.

```
by_date <- train %>%
  filter(train$IsHoliday == TRUE)

ggplot(by_date, aes(Date, Weekly_Sales)) + geom_line() +facet_wrap(~Dept)
```



In order to answer the question about which holiday season shows the best spikes for each department, I decided to use a ggplot again. This time, I filtered for only weeks where IsHoliday is True. Again, the results differ heavily between departments. 1, 4, and 5 seem to spike around the change of the year, while department 3 dips before the start of the new year. Similar case with department two as well, where sales seem to dip slightly before the new year, and spike once the new year begins.

Conclusion

I felt I did a decent job answering the questions I initially had regarding Walmart sales data. The sales forecasting that I did for each department was probably the question that I could have explored in more detail. Implementing more departments could have resulted in better forecasting graphs. In my defense, the sales data does look similar for each year, which would explain the prediction graph results being so closely patterned to the original data. The best variable for predicting weekly sales was not what I was expecting. I initially thought the IsHoliday variable would be the best predictor, but this was not the case for any of the departments that I analyzed. However, temperature and CPI do make sense as being valid predictors as more people are out and about in warmer weather, and changes in CPI indicates there are adjustments to cost of living which could lead to more spending. As for identifying which holiday period shows the best sales spikes, the graphs that I created are a good example of spending increasing in the end of the year holiday season. Again, I would have liked to see other departments results, and I would also like to know which department is which, instead of just department 1, 2, etc. Overall, I found my results to be interesting, and somewhat satisfying. The data that Walmart shared is much larger than the data I was working with, so it would be interesting to see the results of analysis on all of the departments.