STAT625_Project

Brian Zhang, Vi Mai, Xinyu Zhou (Anna), Ziyan Zhao

2023-11-30

```
rm(list = ls())
walmart <- read.csv("~/Documents/Stat 625/Project/Walmart.csv")
head(walmart)
```

```
##   Store       Date Weekly_Sales Holiday_Flag Temperature Fuel_Price      CPI
## 1     1 05-02-2010      1643691            0       42.31      2.572 211.0964
## 2     1 12-02-2010      1641957            1       38.51      2.548 211.2422
## 3     1 19-02-2010      1611968            0       39.93      2.514 211.2891
## 4     1 26-02-2010      1409728            0       46.63      2.561 211.3196
## 5     1 05-03-2010      1554807            0       46.50      2.625 211.3501
## 6     1 12-03-2010      1439542            0       57.79      2.667 211.3806
##   Unemployment
## 1        8.106
## 2        8.106
## 3        8.106
## 4        8.106
## 5        8.106
## 6        8.106
```

Data Preprocessing

Since dates are strings, they must be converted to parsed and converted to days. Use days since the first day rather than the actual date to make computation easier.

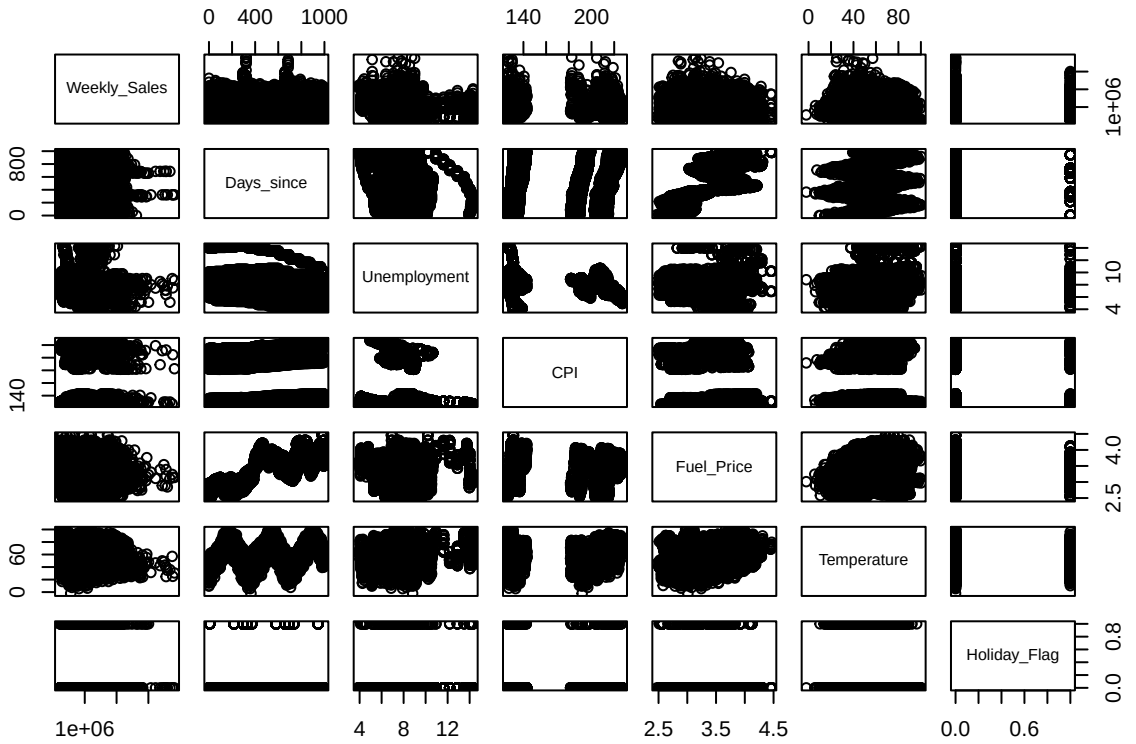
```
# Convert the dates from character strings into days since the first date
asDate_result <- as.Date(walmart$Date, "%d-%m-%Y")
first_date <- min(asDate_result)
days_elapsed <- asDate_result-first_date
walmart["Days_since"] <- days_elapsed
head(walmart)
```

```
##   Store       Date Weekly_Sales Holiday_Flag Temperature Fuel_Price      CPI
## 1     1 05-02-2010      1643691            0       42.31      2.572 211.0964
## 2     1 12-02-2010      1641957            1       38.51      2.548 211.2422
## 3     1 19-02-2010      1611968            0       39.93      2.514 211.2891
## 4     1 26-02-2010      1409728            0       46.63      2.561 211.3196
## 5     1 05-03-2010      1554807            0       46.50      2.625 211.3501
## 6     1 12-03-2010      1439542            0       57.79      2.667 211.3806
##   Unemployment Days_since
## 1        8.106     0 days
```

```
## 2        8.106     7 days
## 3        8.106    14 days
## 4        8.106    21 days
## 5        8.106    28 days
## 6        8.106    35 days
```

Scatterplot matrix - uninterprettable

```
pairs(Weekly_Sales ~ Days_since + Unemployment + CPI + Fuel_Price + Temperature + Holiday_Flag, data = w
```

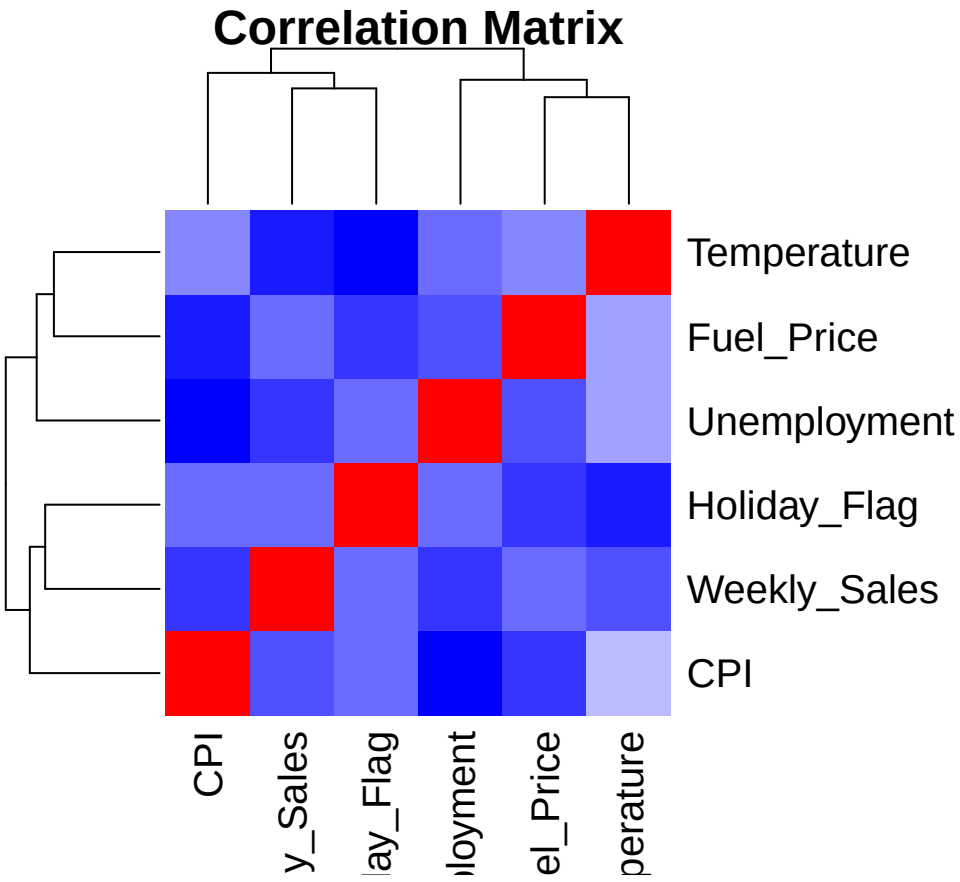



```
selected_columns <- c("Weekly_Sales", "Days_since", "Unemployment", "CPI", "Fuel_Price", "Temperature",
selected_data <- walmart[selected_columns]
correlation_matrix <- cor(selected_data[, sapply(selected_data, is.numeric)], use = "complete.obs")
correlation_matrix
```

```
##              Weekly_Sales Unemployment          CPI   Fuel_Price Temperature
## Weekly_Sales  1.000000000  -0.10617609 -0.072634162  0.009463786 -0.06381001
## Unemployment -0.106176090   1.00000000 -0.302020064 -0.034683745  0.10115786
## CPI          -0.072634162  -0.30202006  1.000000000 -0.170641795  0.17688768
## Fuel_Price    0.009463786  -0.03468374 -0.170641795  1.000000000  0.14498181
## Temperature  -0.063810013   0.10115786  0.176887676  0.144981806  1.00000000
## Holiday_Flag  0.036890968   0.01096028 -0.002162091 -0.078346518 -0.15509133
```

```
##              Holiday_Flag
## Weekly_Sales  0.036890968
## Unemployment  0.010960284
## CPI          -0.002162091
## Fuel_Price   -0.078346518
## Temperature  -0.155091329
## Holiday_Flag  1.000000000
```

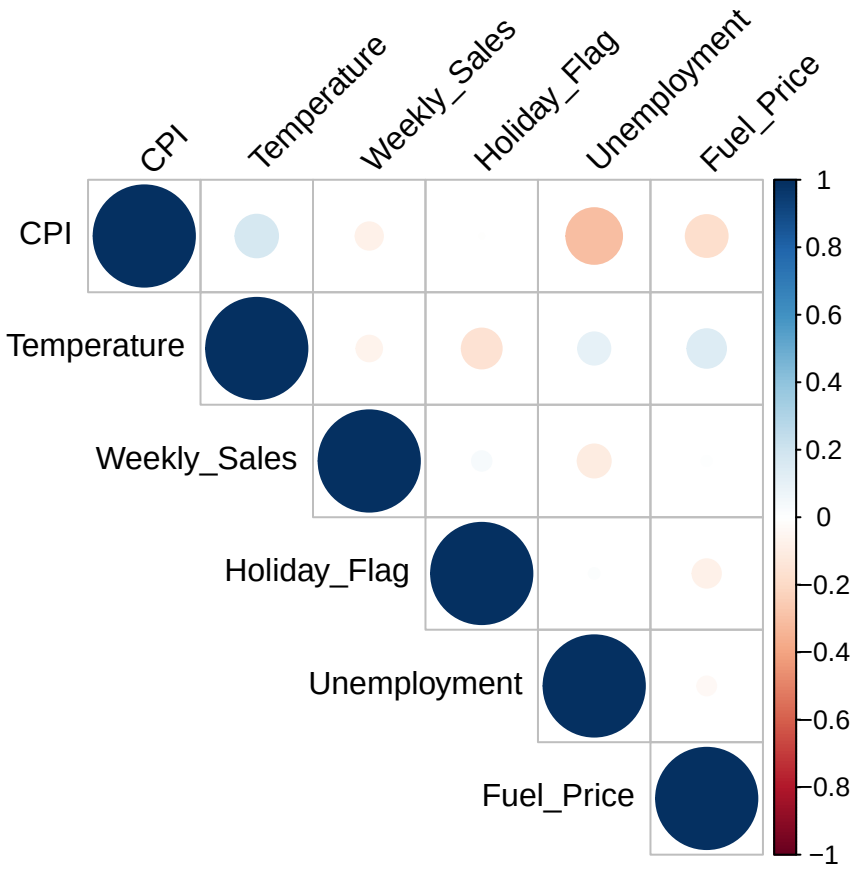
```
heatmap(correlation_matrix,
        col = colorRampPalette(c("blue", "white", "red"))(20),
        main = "Correlation Matrix")
```




```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
corrplot(correlation_matrix, type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45)
```

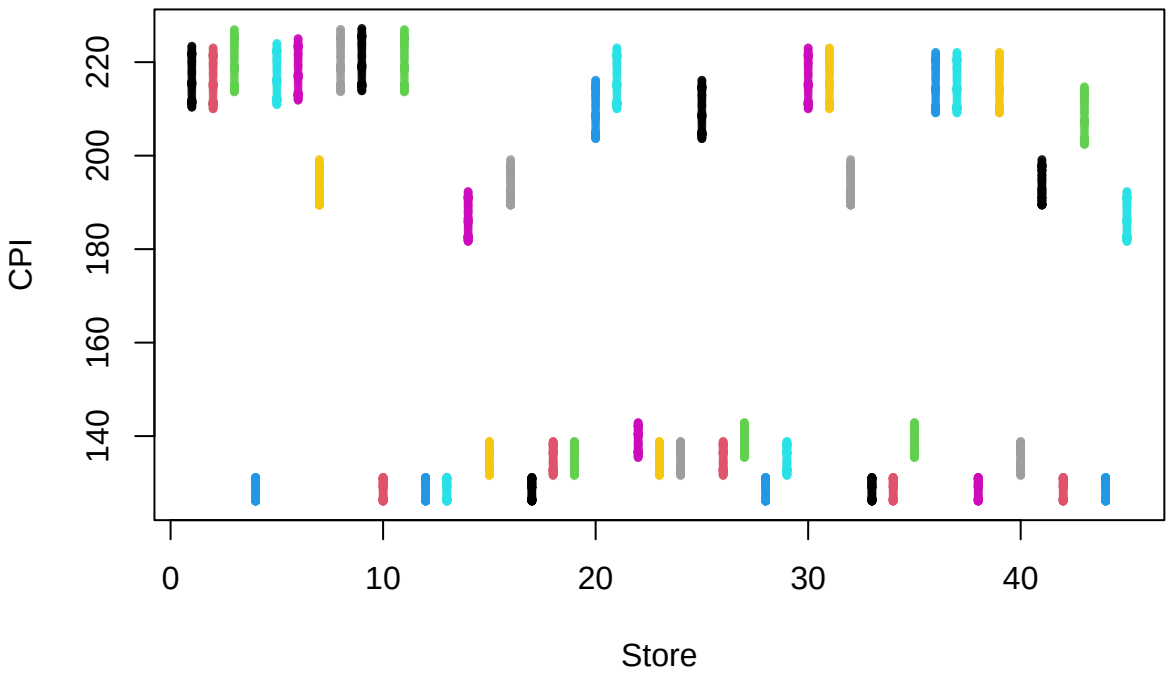


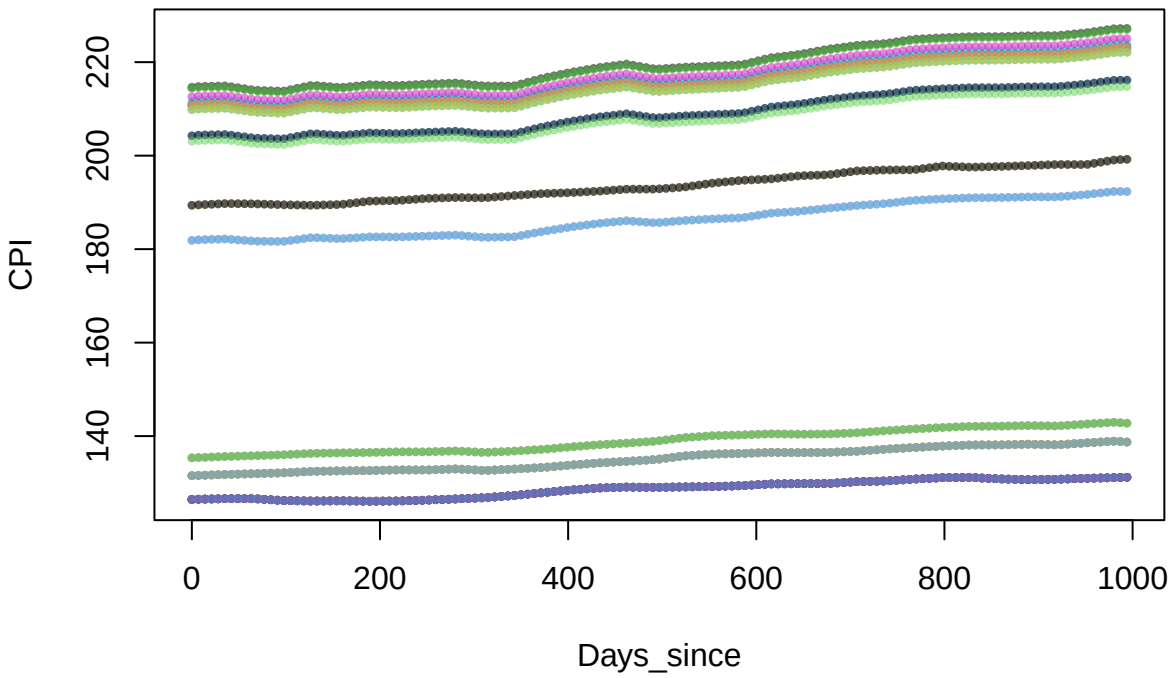
Scatterplot with CPI vs others

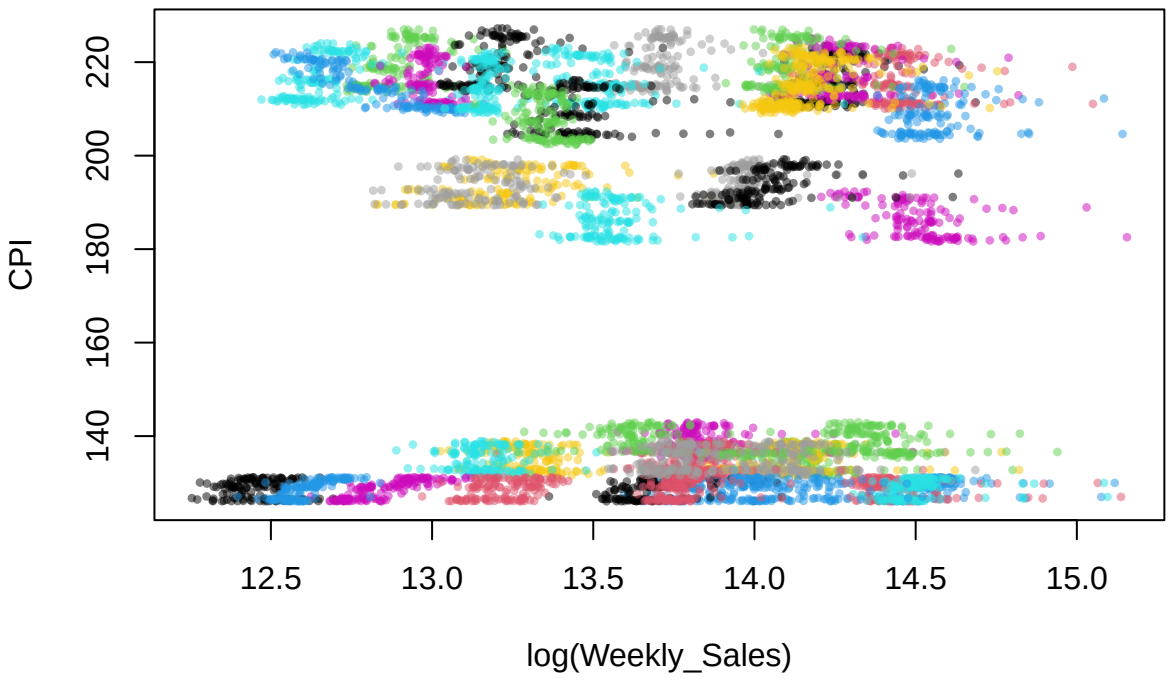
```
library(tidyverse)
```

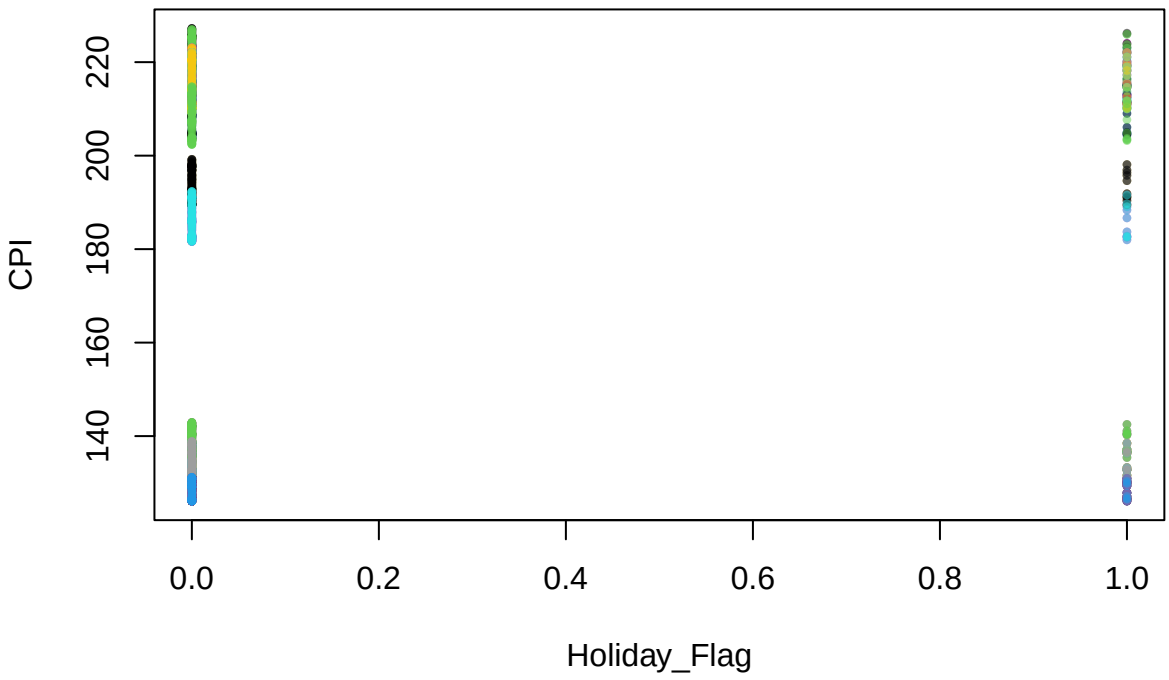
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

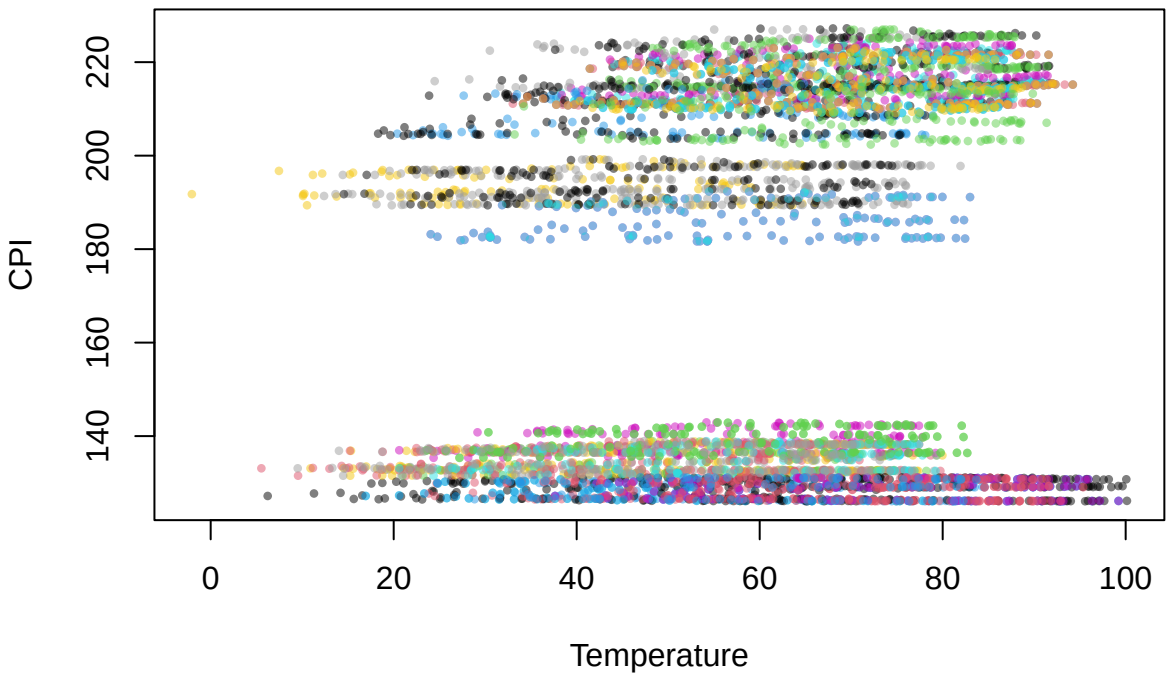
```
plot(CPI ~ Store + Days_since + log(Weekly_Sales) + Holiday_Flag + Temperature + Fuel_Price + Unemployme
     col = alpha(walmart$Store, 0.5),  # makes color dependent on store
     pch = 16,
     cex = 0.6)
```

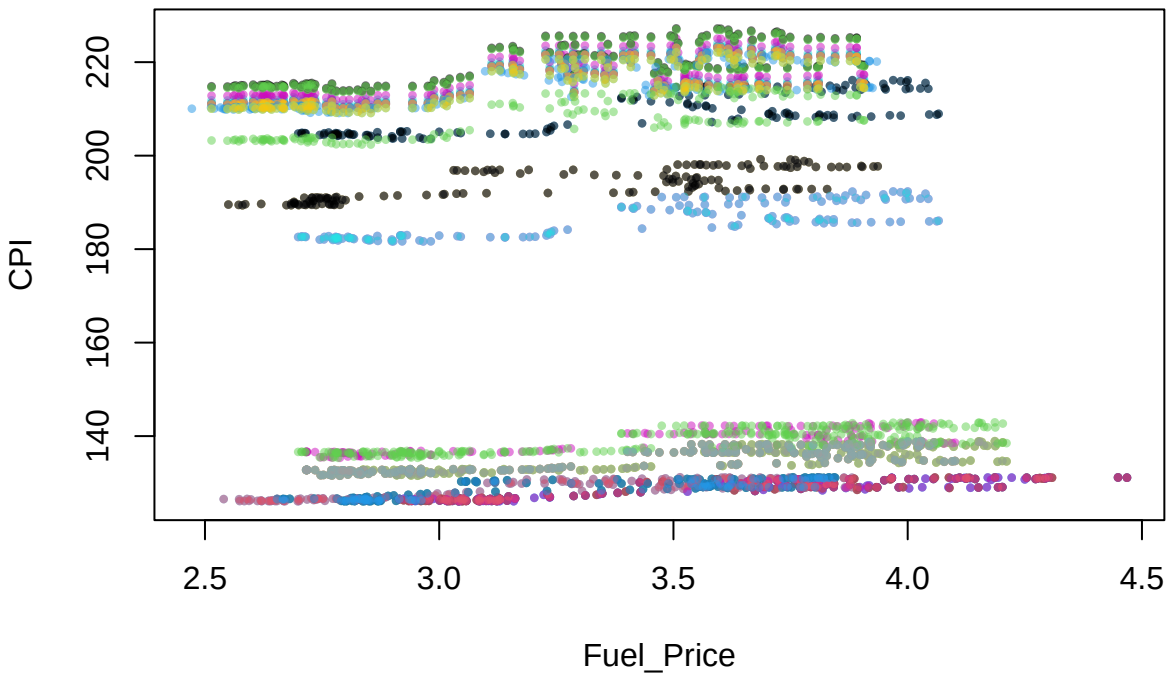

CPI
140
160
180
200
220
0
10
20
30
40
Store

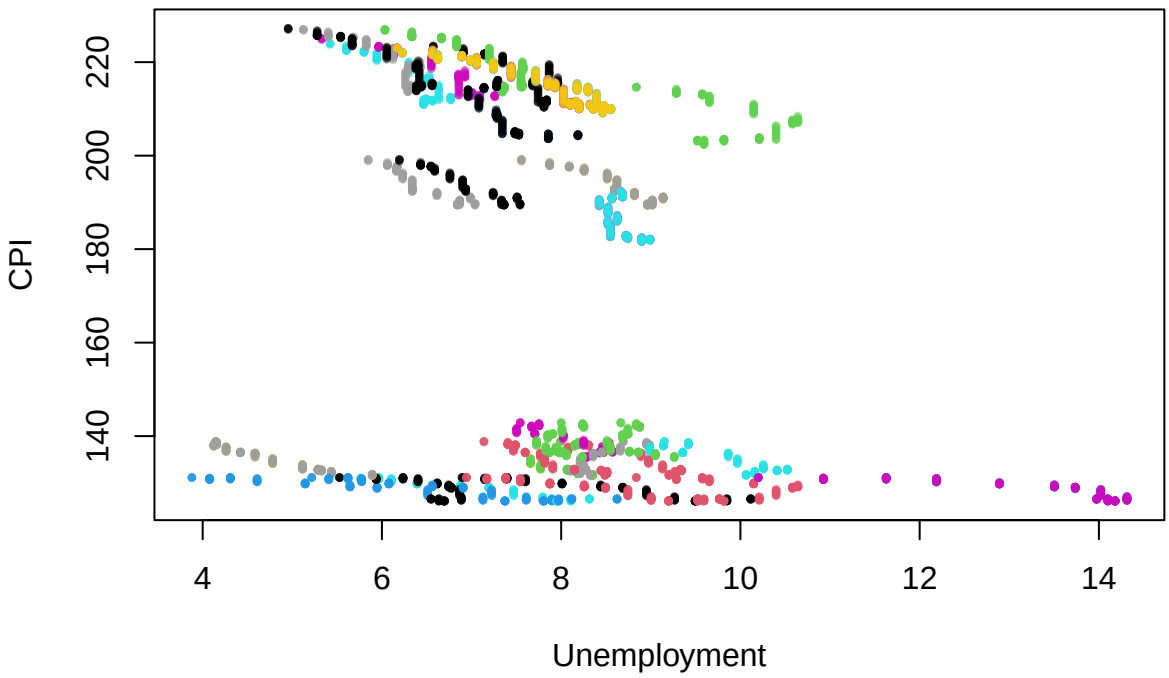

CPI
140
160
180
200
220
0
200
400
600
800
1000
Days_since


CPI
220
200
180
160
140
12.5
13.0
13.5
14.0
14.5
15.0
log(Weekly_Sales)


CPI
140
160
180
200
220
0.0
0.2
0.4
0.6
0.8
1.0
Holiday_Flag


CPI
140
160
180
200
220
0
20
40
60
80
100
Temperature

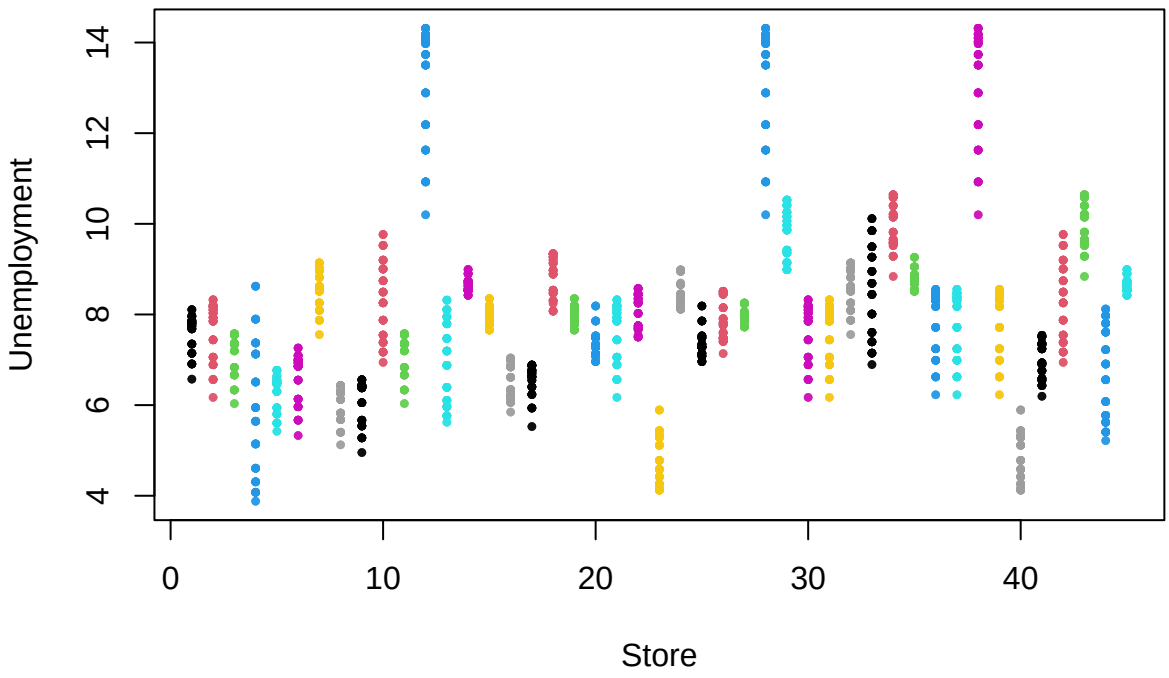

CPI
140
160
180
200
220
2.5
3.0
3.5
4.0
4.5
Fuel_Price

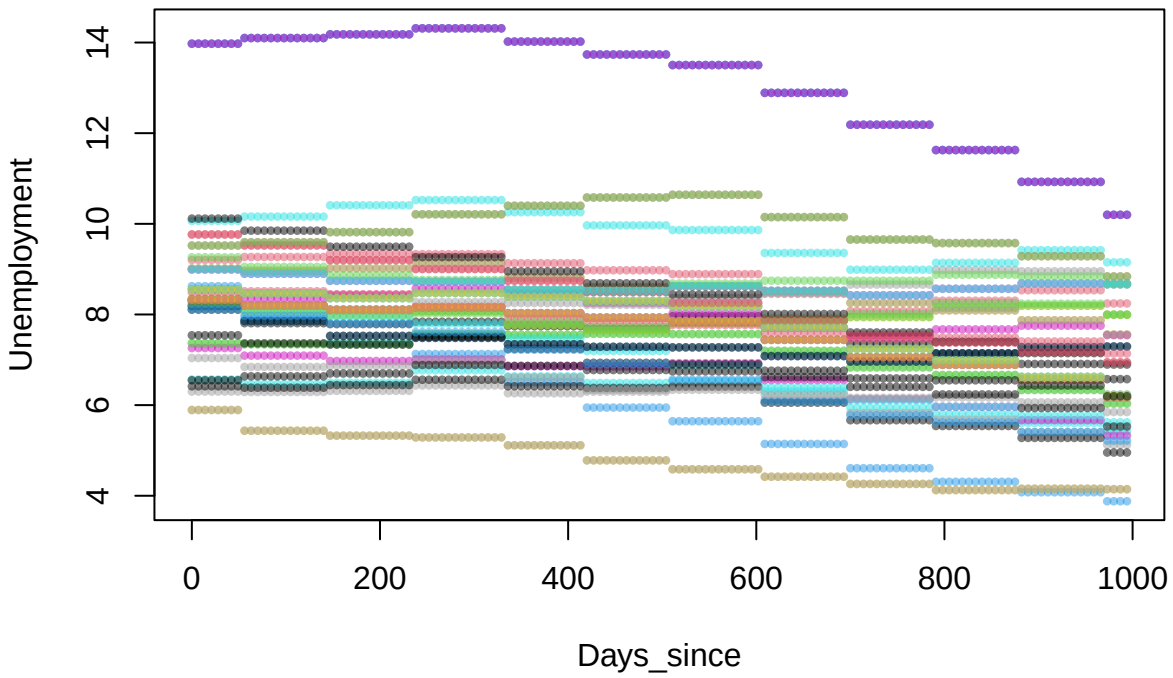


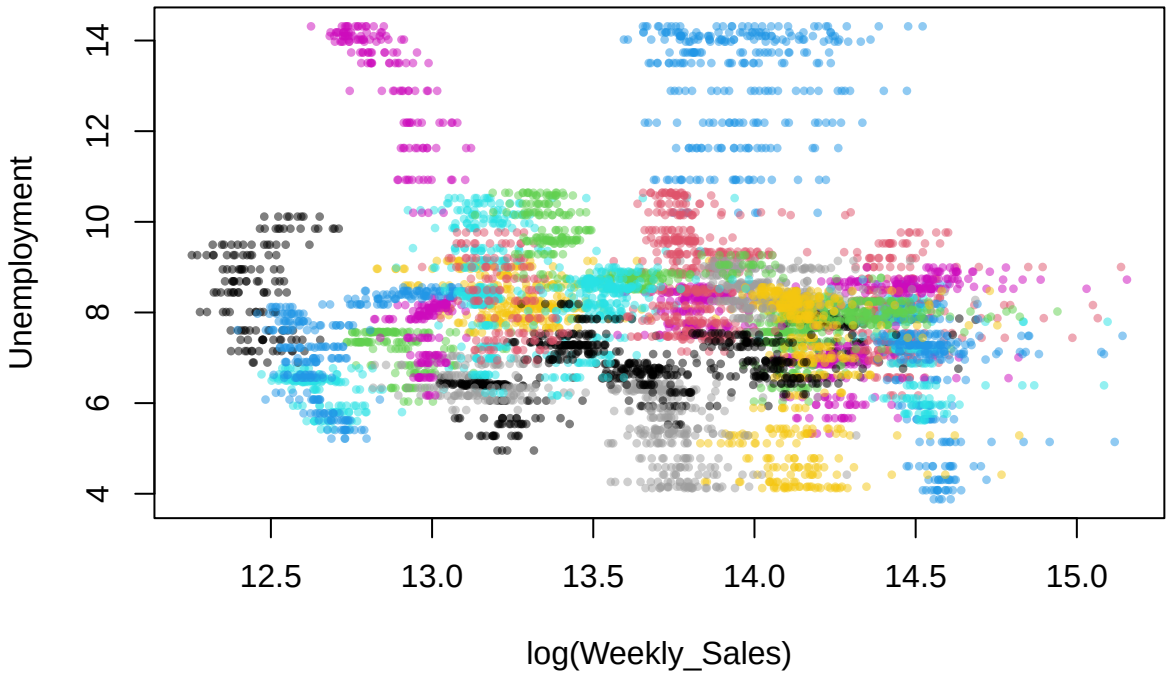
CPI increases over time. CPI does not affect sales.

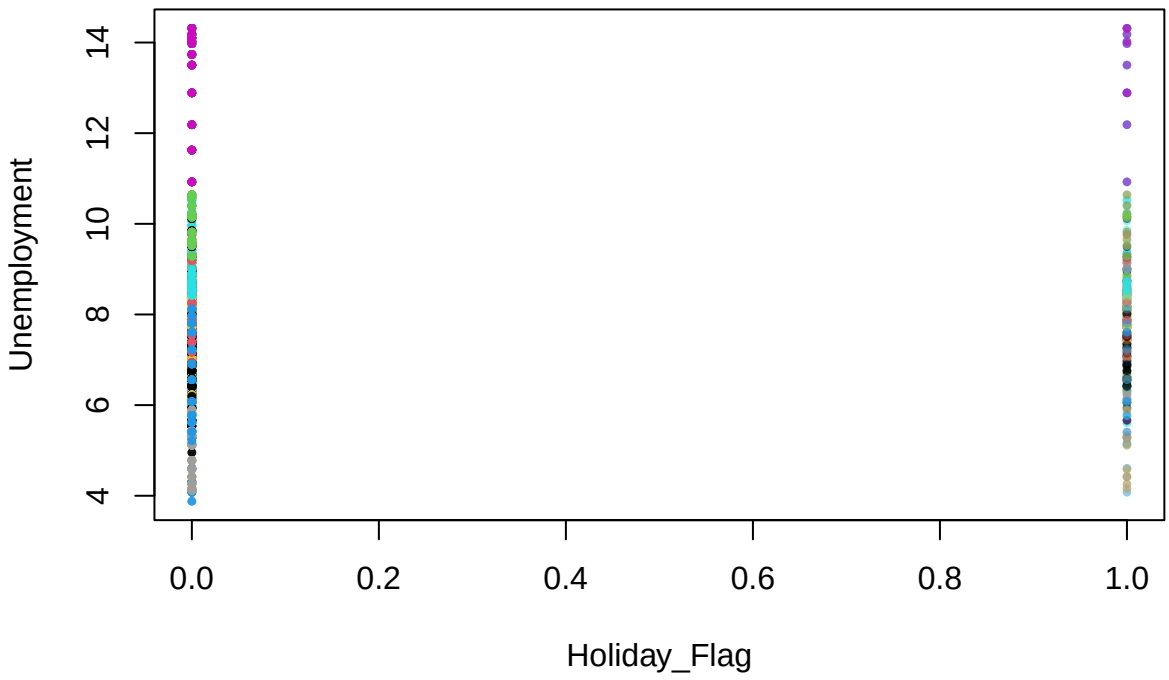
Scatterplot with Unemployment vs others

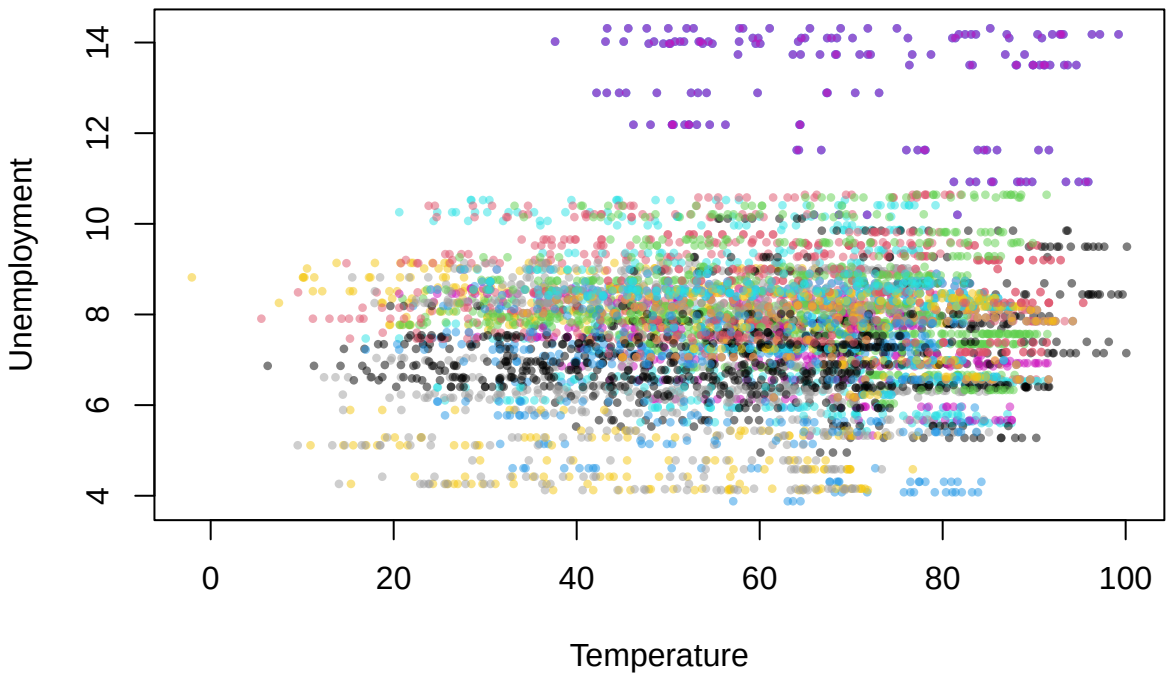
```
plot(Unemployment ~ Store + Days_since + log(Weekly_Sales) + Holiday_Flag + Temperature + Fuel_Price + C
     col = alpha(walmart$Store, 0.5),  # makes color dependent on store
     pch = 16,
     cex = 0.6)
```

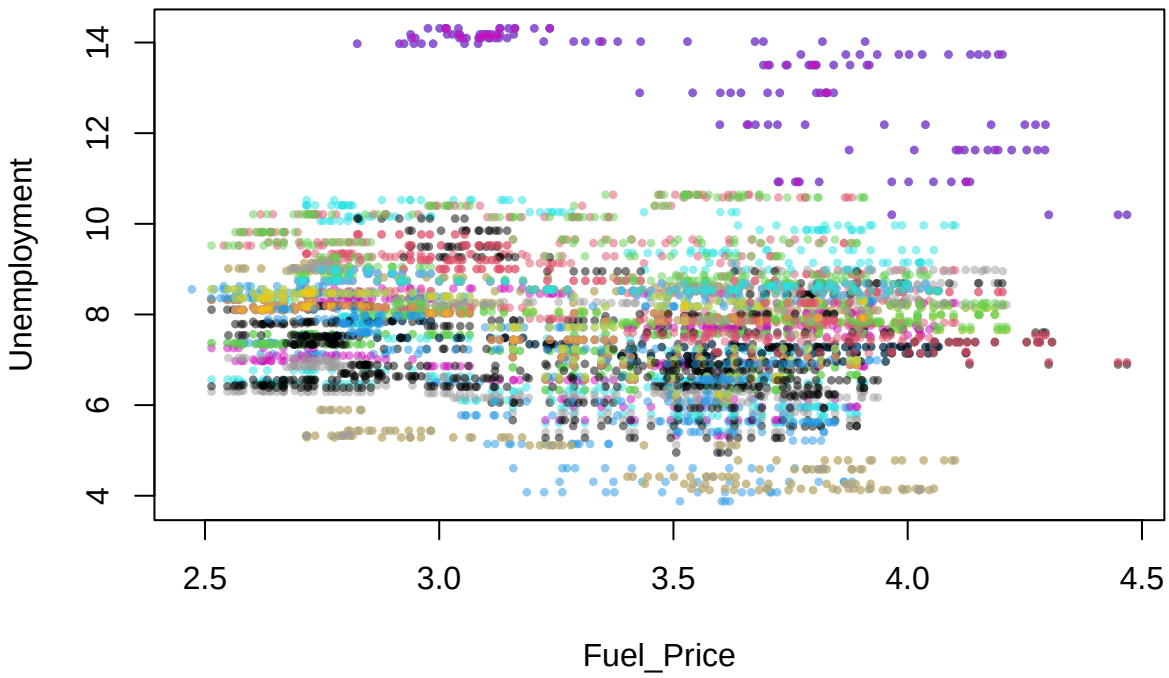

Unemployment
4
6
8
10
12
14
0
10
20
30
40
Store

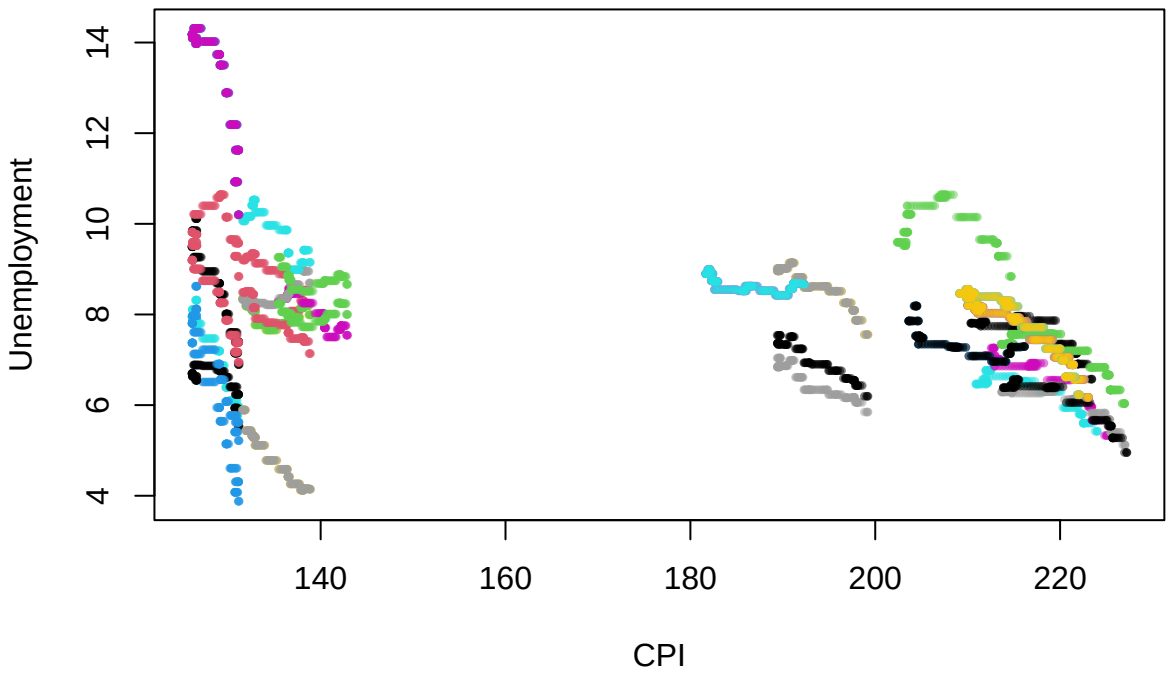

Unemployment
4
6
8
10
12
14
0
200
400
600
800
1000
Days_since


Unemployment
4
6
8
10
12
14
12.5
13.0
13.5
14.0
14.5
15.0
log(Weekly_Sales)


Unemployment
4
6
8
10
12
14
0.0
0.2
0.4
0.6
0.8
1.0
Holiday_Flag


Unemployment
4
6
8
10
12
14
0
20
40
60
80
100
Temperature

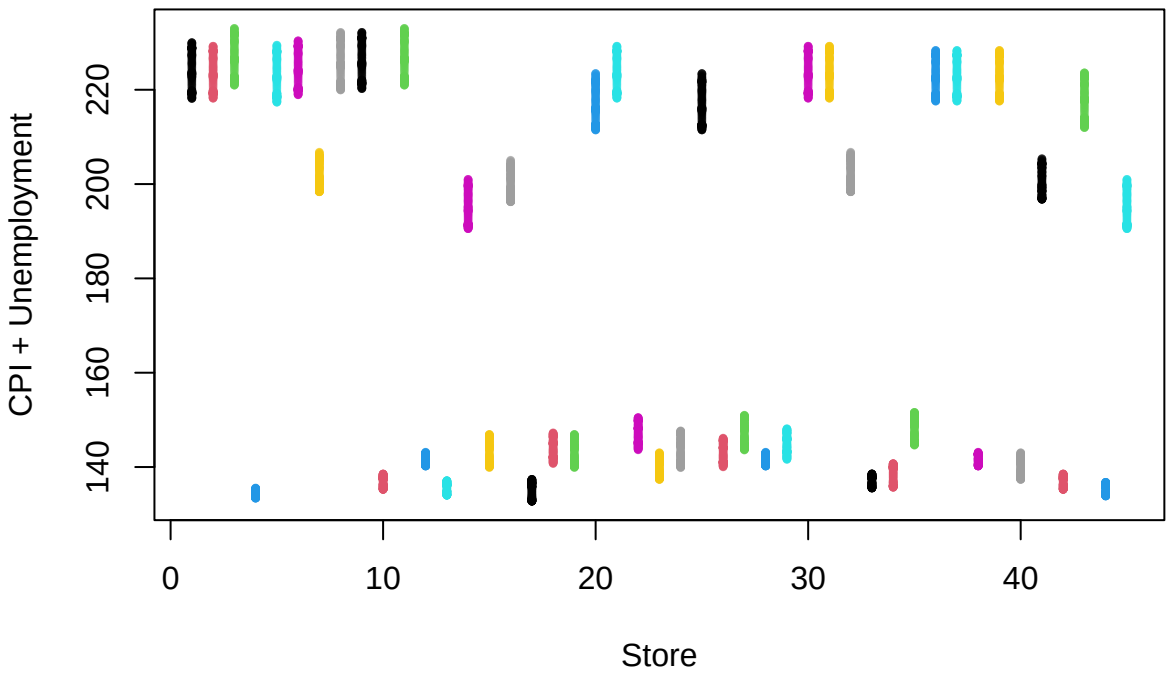

Unemployment
4
6
8
10
12
14
2.5
3.0
3.5
4.0
4.5
Fuel_Price

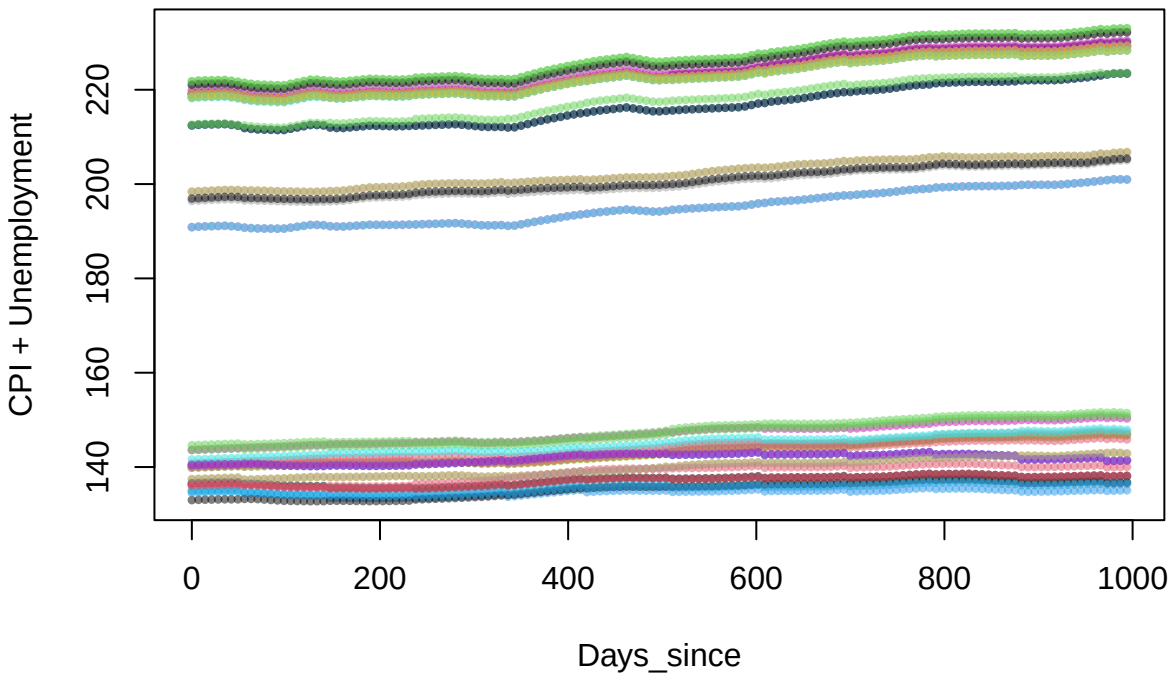


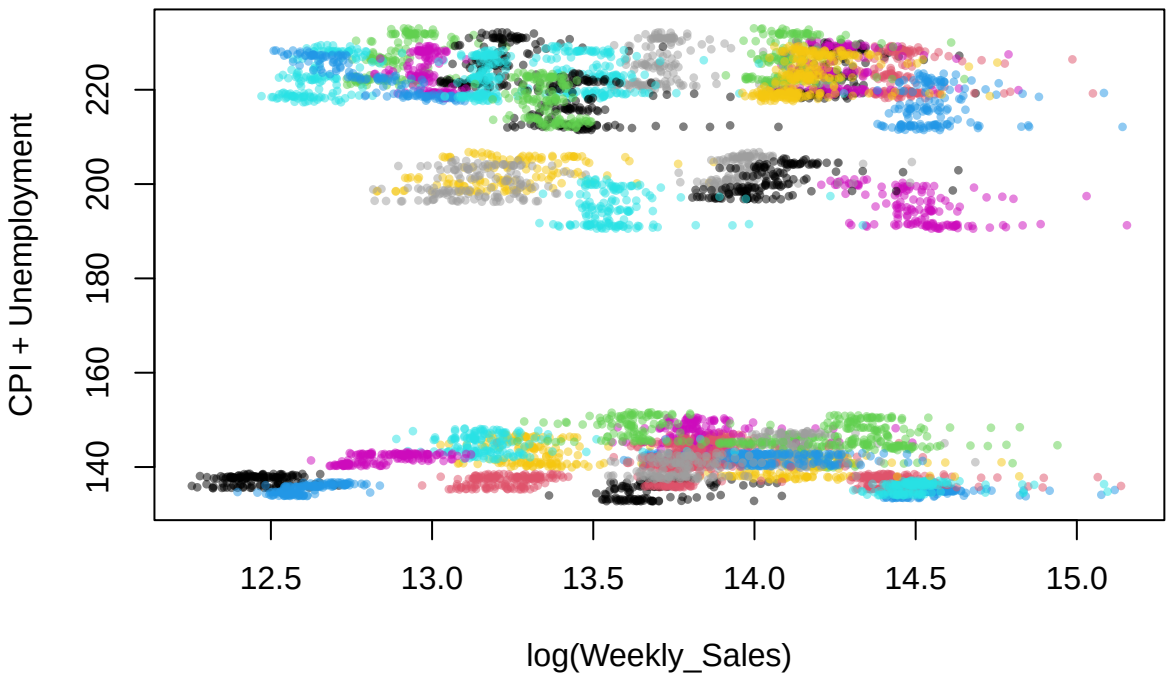
The unemployment rate decreases over time. Sales are affected by the unemployment rate, so the higher the unemployment rate, the lower the sales. Unemployment seems like a good predictor of the weekly sales, the higher the unemployment rate, the lower the weekly sales.

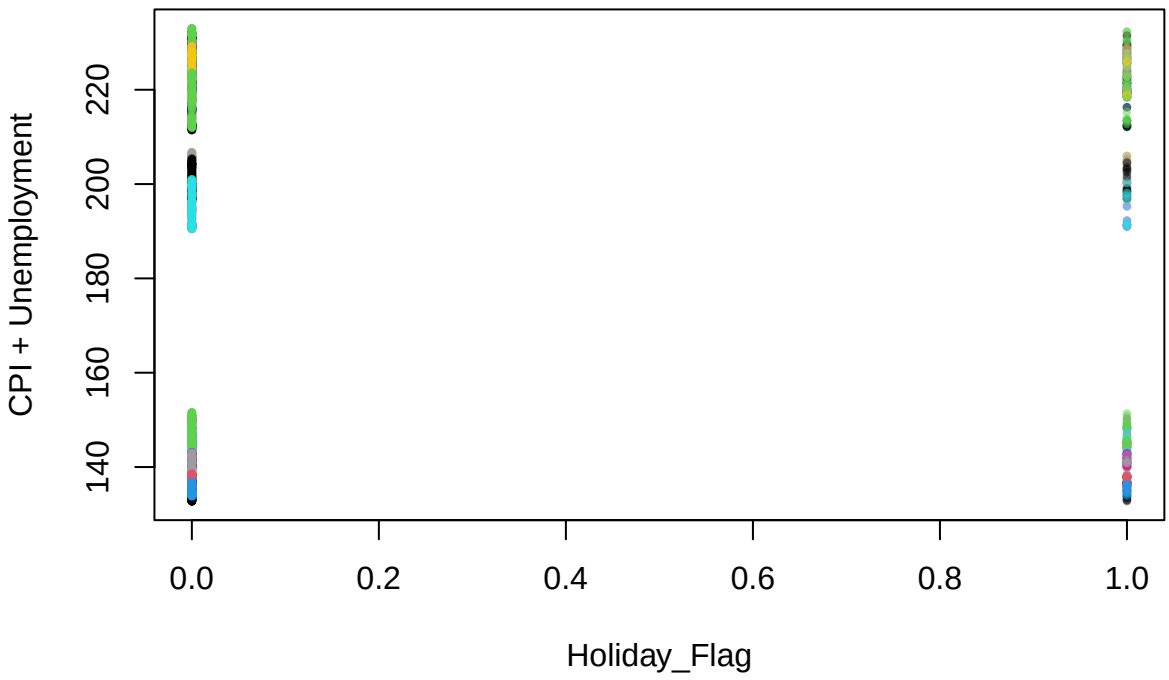
Scatterplot with CPI + Unemployment vs others

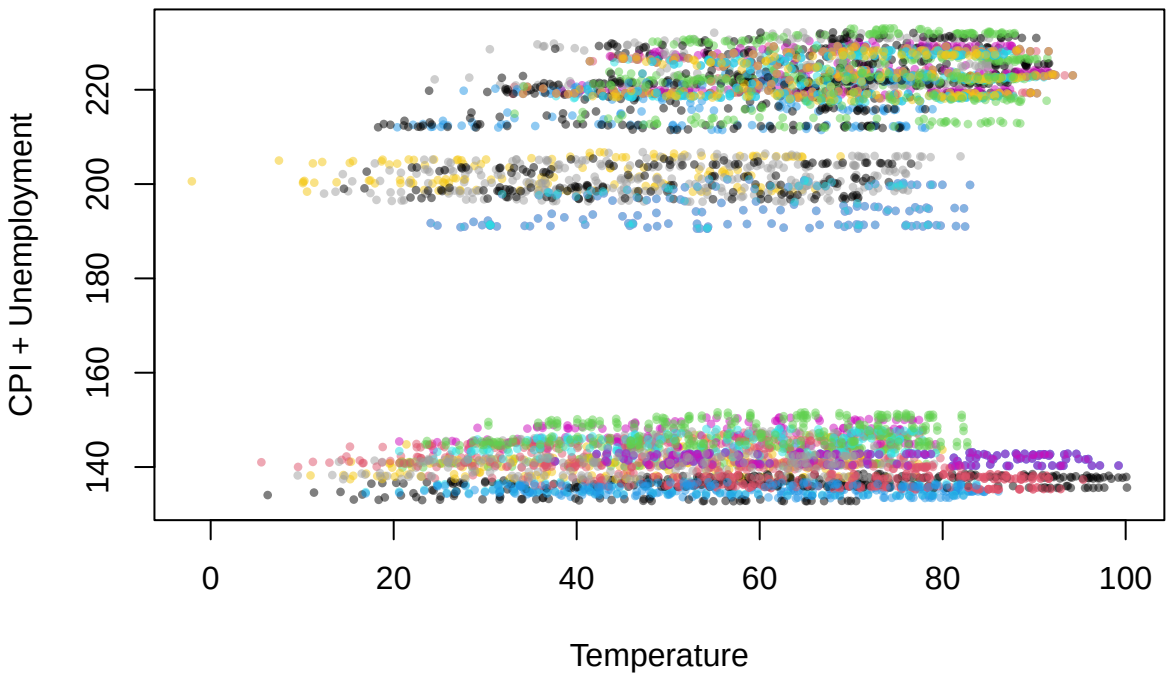
```
plot(CPI + Unemployment ~ Store + Days_since + log(Weekly_Sales) + Holiday_Flag + Temperature + Fuel_Pri
     col = alpha(walmart$Store, 0.5),  # makes color dependent on store
     pch = 16,
     cex = 0.6)
```

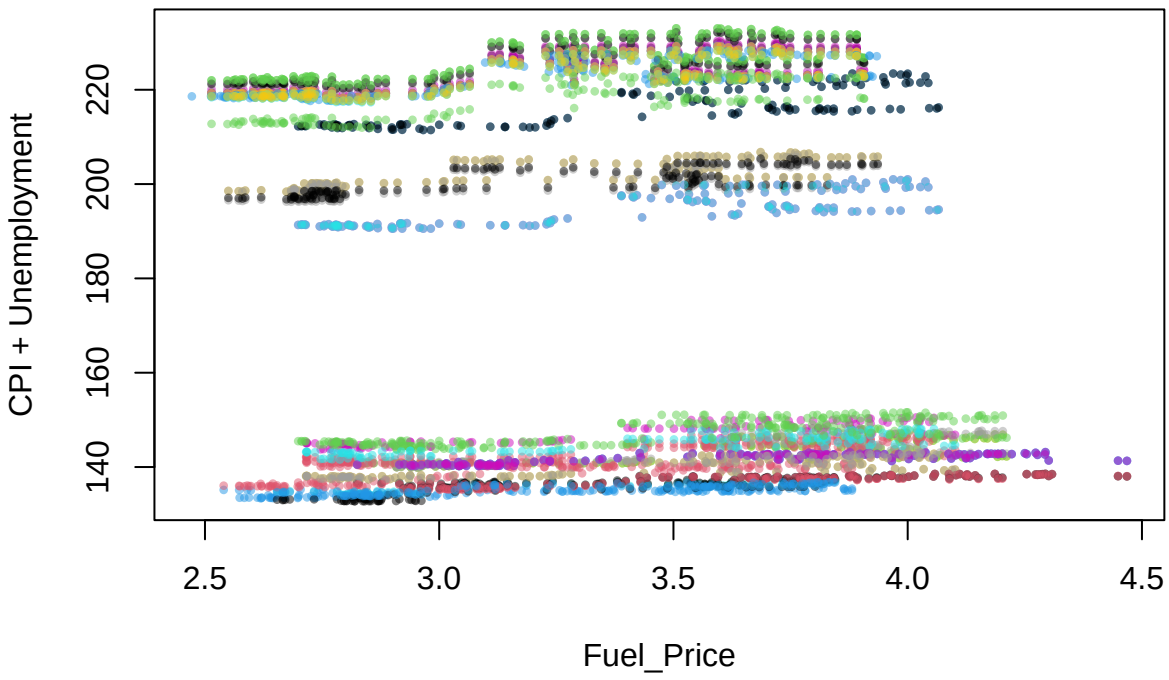

CPI + Unemployment
140
160
180
200
220
0
10
20
30
40
Store


CPI + Unemployment
140
160
180
200
220
0
200
400
600
800
1000
Days_since


CPI + Unemployment
140
160
180
200
220
12.5
13.0
13.5
14.0
14.5
15.0
log(Weekly_Sales)


CPI + Unemployment
140
160
180
200
220
0.0
0.2
0.4
0.6
0.8
1.0
Holiday_Flag


CPI + Unemployment
140
160
180
200
220
0
20
40
60
80
100
Temperature



CPI + Unemployment increases over time.

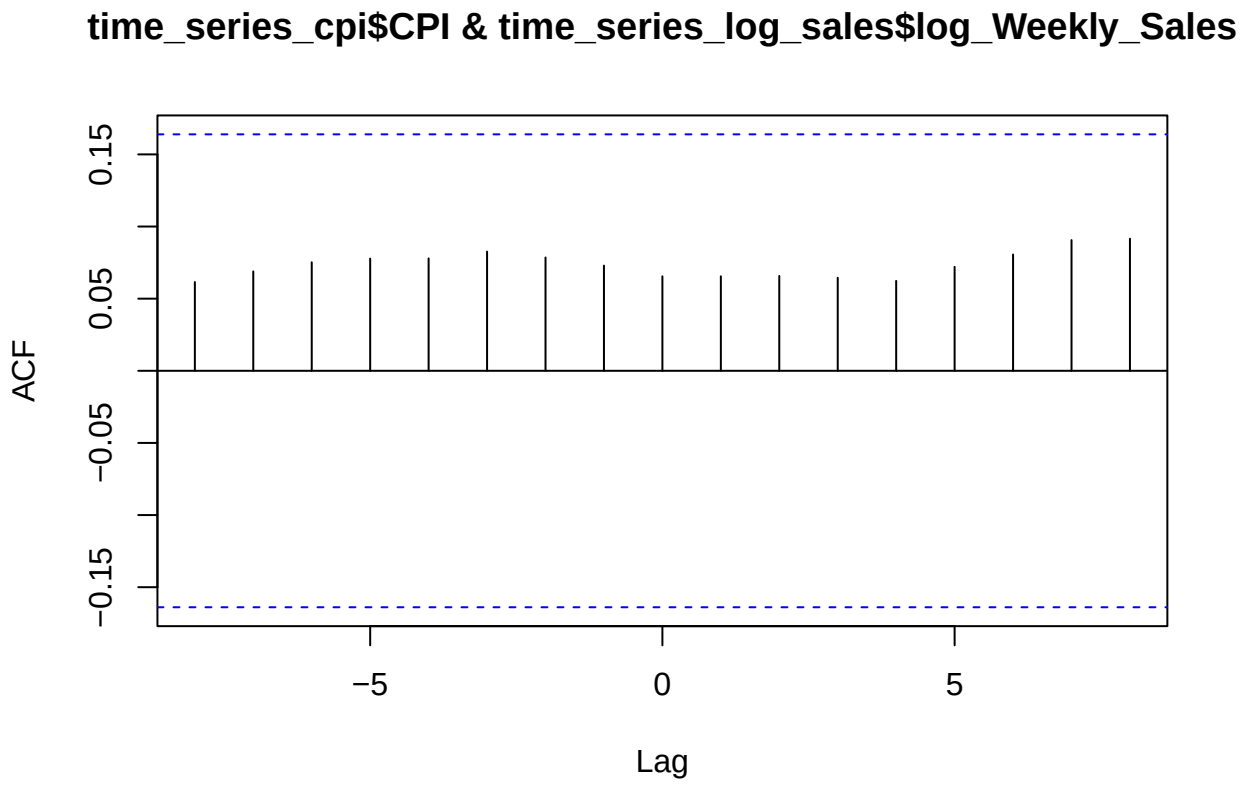
lagged model

```
# Assuming CPI is in a column called "CPI" and Weekly_Sales in "log_Weekly_Sales"
walmart$Date <- as.Date(walmart$Date, format="%d-%m-%Y")
walmart$Days_since <- as.numeric(difftime(walmart$Date, min(walmart$Date), units = "days"))

# Extract time series data for CPI
time_series_cpi <- data.frame(Days_since = unique(walmart$Days_since), CPI = numeric(length(unique(walma
for (day in unique(walmart$Days_since)) {
  time_series_cpi$CPI[time_series_cpi$Days_since == day] <- walmart$CPI[walmart$Days_since == day][1]
}

# Extract time series data for log_Weekly_Sales
time_series_log_sales <- data.frame(Days_since = unique(walmart$Days_since), log_Weekly_Sales = numeric
for (day in unique(walmart$Days_since)) {
  time_series_log_sales$log_Weekly_Sales[time_series_log_sales$Days_since == day] <- mean(log(walmart$W
}

ccf_result <- ccf(time_series_cpi$CPI, time_series_log_sales$log_Weekly_Sales, lag.max = 8)
```




```
ccf_result
```

```
##
## Autocorrelations of series 'X', by lag
##
##    -8    -7    -6    -5    -4    -3    -2    -1     0     1     2     3     4
## 0.062 0.069 0.075 0.078 0.078 0.083 0.079 0.073 0.065 0.066 0.066 0.064 0.062
##     5     6     7     8
## 0.072 0.081 0.091 0.092
```

The highest positive autocorrelation is at lag 10, with a value of 0.101.
The positive autocorrelation values at lags 7 to 10 (0.072 to 0.101) suggest a positive relationship between the CPI and log(Weekly_Sales) with a lag of 7 to 10 time points. This means that high values of CPI at a particular time point are associated with high values of log(Weekly_Sales) at a lag of 7 to 10 time points.
The positive autocorrelation values at lags 1 to 6 (0.064 to 0.078) also indicate a positive relationship, but with a shorter lag. Thus, there is some degree of correlation between the CPI and log(Weekly_Sales) at various lags.