

STAT625_Project

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

2023-11-13

```
library(ggplot2)
rm(list = ls())
walmart <- read.csv("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
```

```

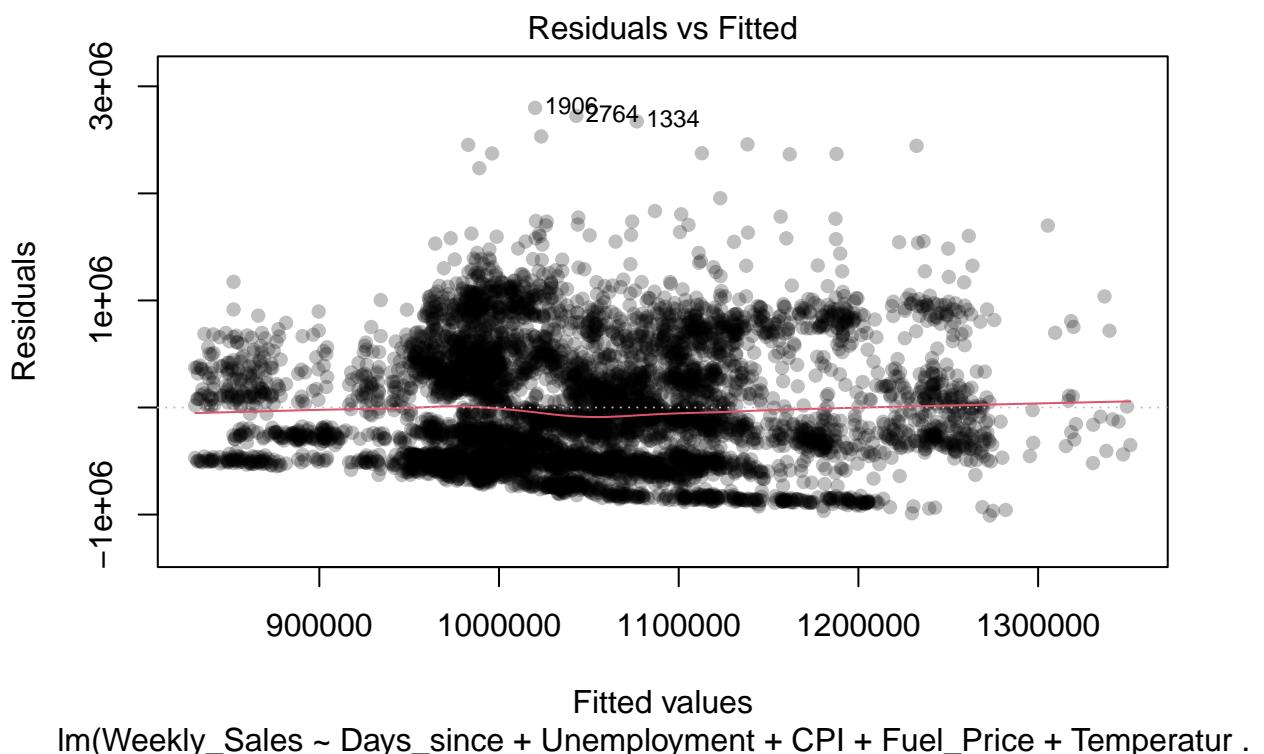
## 1      8.106    0 days
## 2      8.106    7 days
## 3      8.106   14 days
## 4      8.106   21 days
## 5      8.106   28 days
## 6      8.106   35 days

```

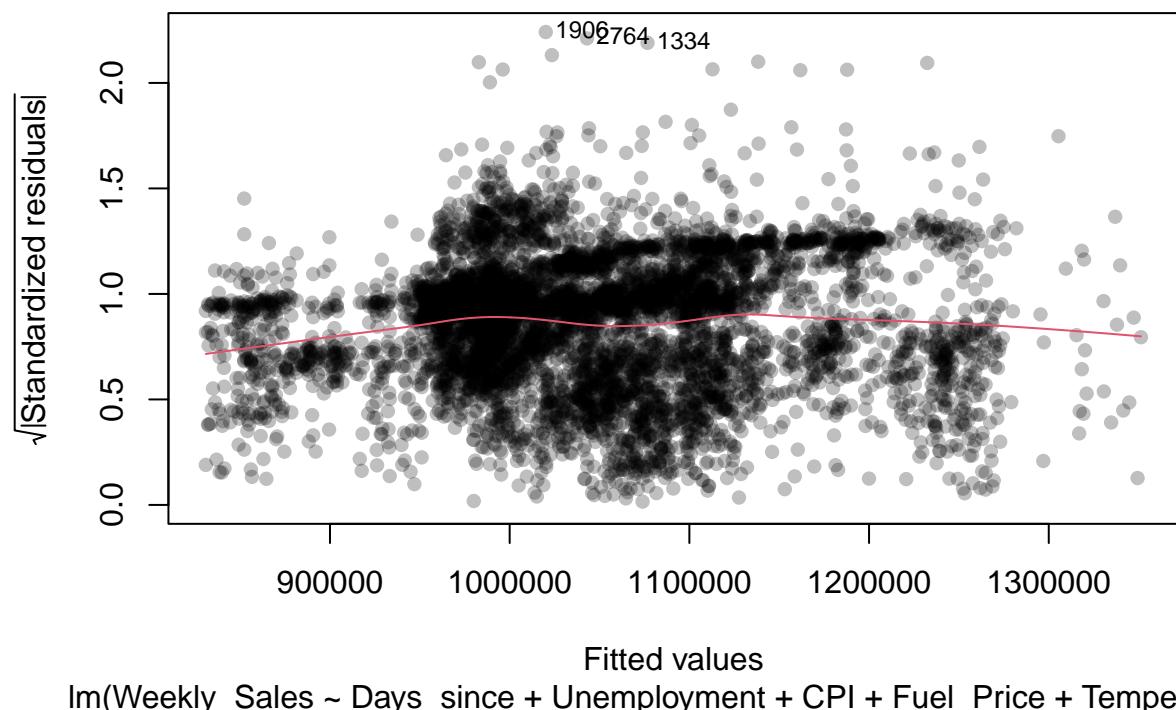
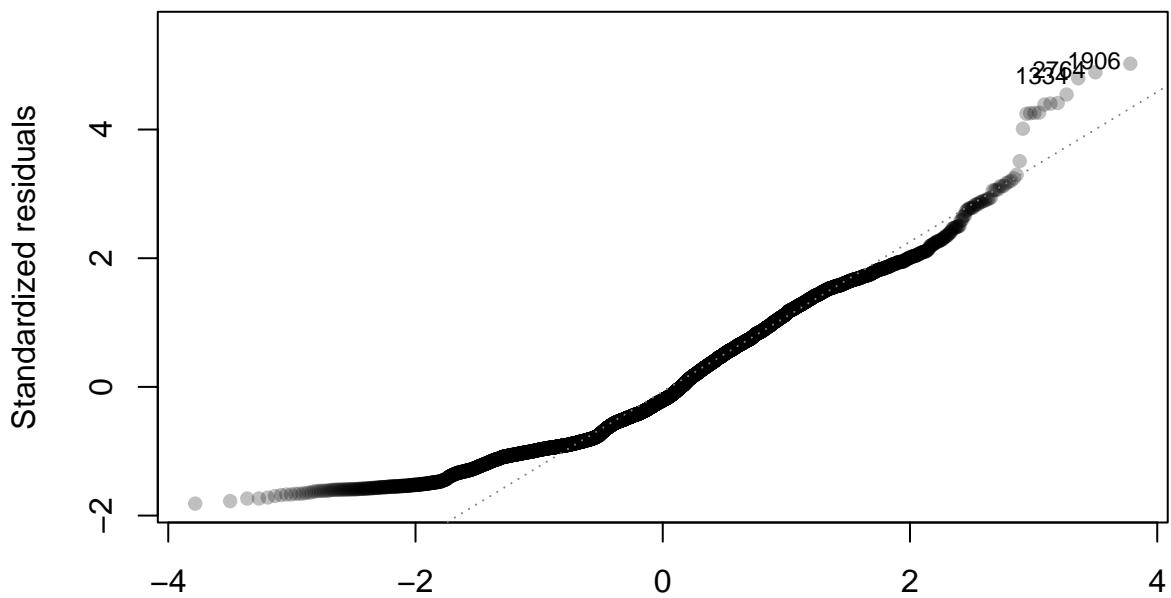
Initial Diagnostic Plots

```
# Comparing the need for a log transformation on Weekly Sales
```

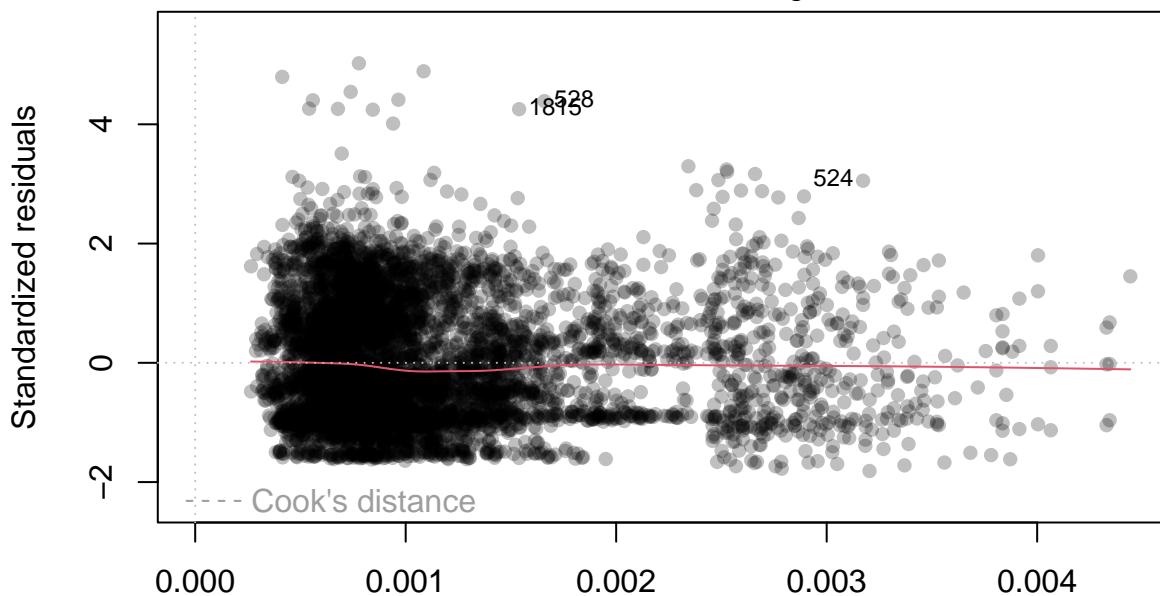
```
full_mod_no_log <- lm(Weekly_Sales ~ Days_since + Unemployment + CPI + Fuel_Price + Temperature + Holiday,
plot(full_mod_no_log,
      col = rgb(red = 0, green = 0, blue = 0, alpha = 0.25),
      pch = 16
)
```



Normal Q–Q



Residuals vs Leverage

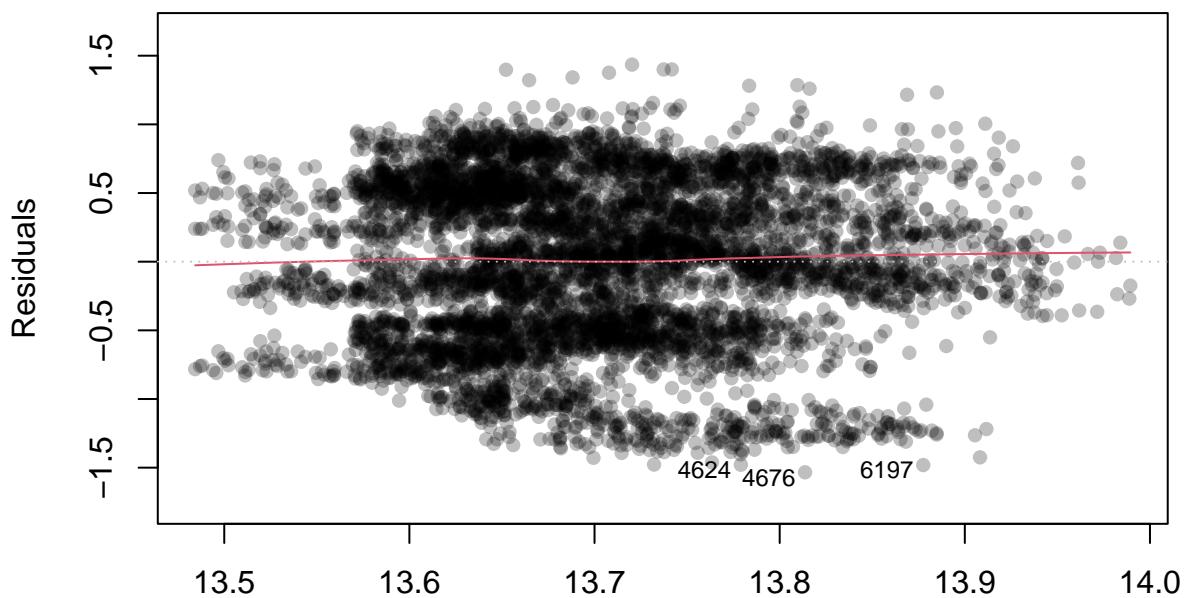


Leverage

`lm(Weekly_Sales ~ Days_since + Unemployment + CPI + Fuel_Price + Temperatur .`

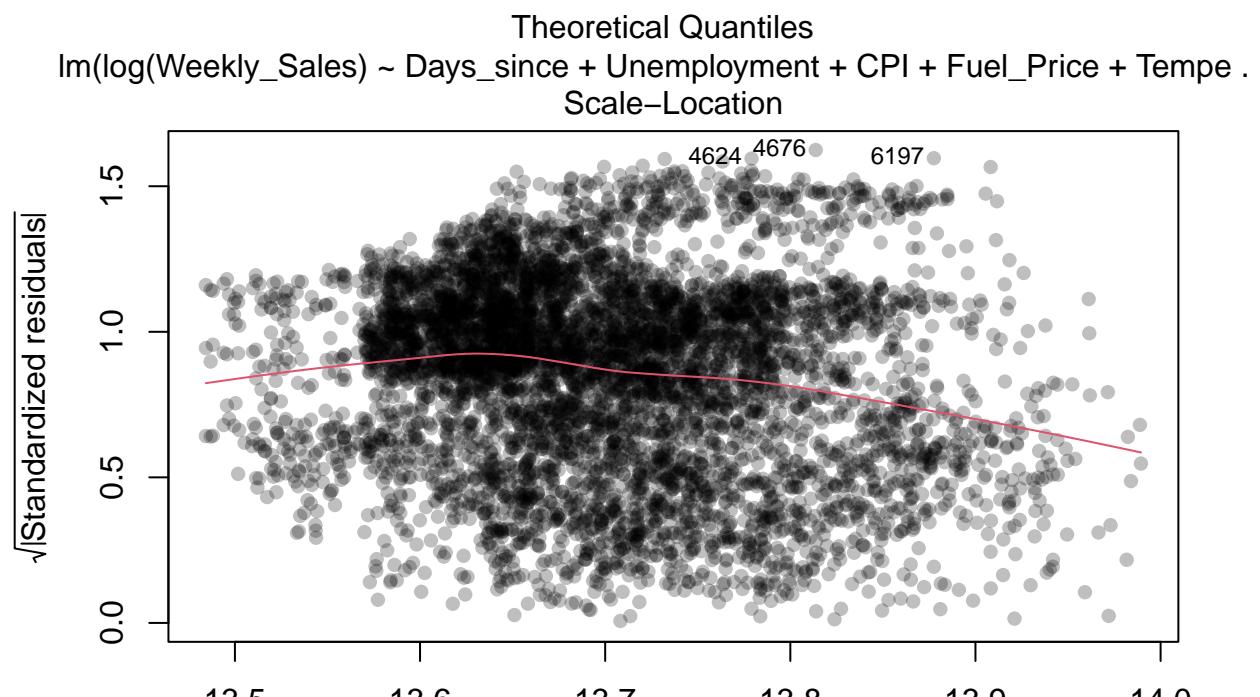
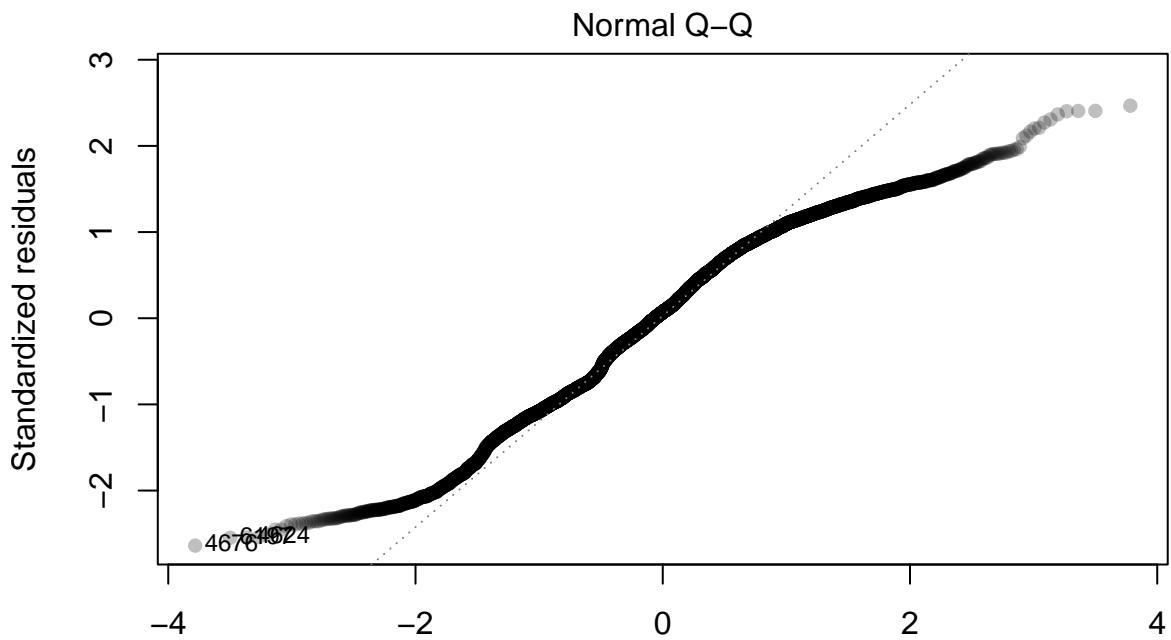
```
full_mod_w_log <- lm(log(Weekly_Sales) ~ Days_since + Unemployment + CPI + Fuel_Price + Temperature + H
plot(full_mod_w_log,
      col = rgb(red = 0, green = 0, blue = 0, alpha = 0.25),
      pch = 16
)
```

Residuals vs Fitted

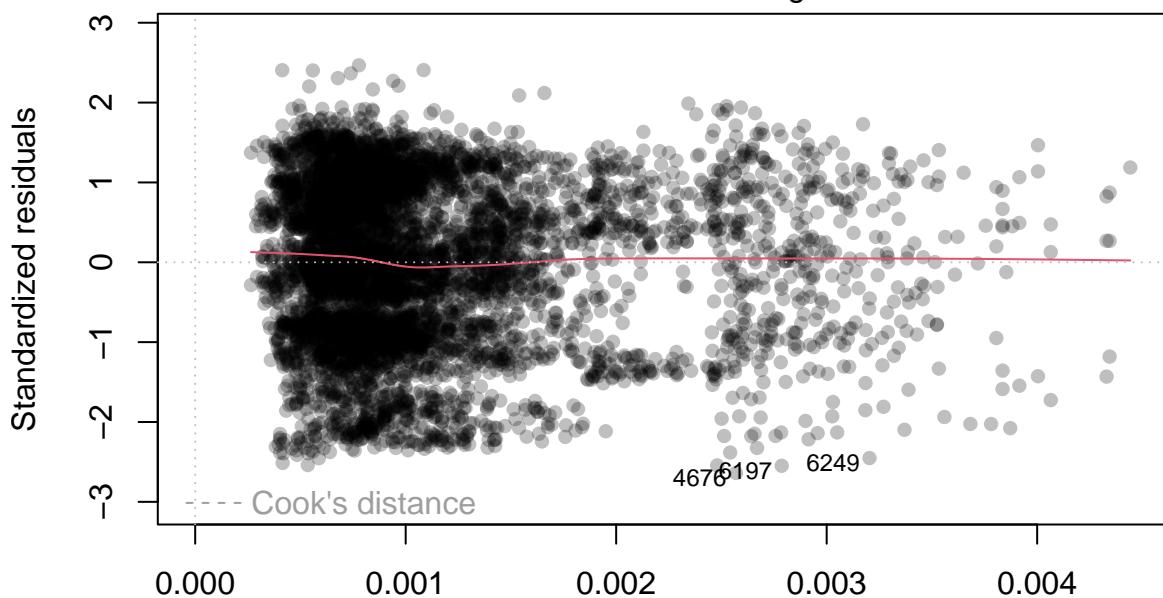


Fitted values

`lm(log(Weekly_Sales) ~ Days_since + Unemployment + CPI + Fuel_Price + Tempe .`



Residuals vs Leverage



```
# Looking at effect of Days and Store on log(Weekly Sales)
plot(log(Weekly_Sales) ~ Days_since + Store + Temperature, data = walmart,
  col = alpha(walmart$Store, 0.5), # makes color dependent on store
  pch = 16,
  cex = 0.6)
```

