Sta 325 Final Project

Calleigh Smith, Hannah Bogomilsky, Hugh Esterson, Maria Henriquez, Mariana Izon

11/22/2020

```r
library(readr)
library(dplyr)
library(tidyverse)
library(gridExtra)
library(mgcv)
library(patchwork)
```

```r
flights <- read_csv("data/flights.csv")

unique(flights$OP_CARRIER)
```

```
## [1] "AA" "DL" "B6" "AS"
```

```r
unique(flights$DEST)
```

```
##  [1] "LAX" "SFO" "SJC" "SAN" "PSP" "SMF" "OAK" "LGB" "ONT" "BUR"
```

```r
class(flights$CARRIER_DELAY)
```

```
## [1] "numeric"
```

```r
flights <- flights %>%
  mutate(CARRIER_DELAY = case_when(CARRIER_DELAY > 0 ~ 1,
                                   TRUE ~ 0),
         WEATHER_DELAY = case_when(WEATHER_DELAY > 0 ~ 1,
                                   TRUE ~ 0),
         NAS_DELAY = case_when(NAS_DELAY > 0 ~ 1,
                               TRUE ~ 0),
         SECURITY_DELAY = case_when(SECURITY_DELAY > 0 ~ 1,
                                    TRUE ~ 0),
         LATE_AIRCRAFT_DELAY = case_when(LATE_AIRCRAFT_DELAY > 0 ~ 1,
                                         TRUE ~ 0))
```

```r
flights
```

```
## # A tibble: 2,044 x 34
##     YEAR MONTH DAY_OF_MONTH DAY_OF_WEEK FL_DATE    OP_CARRIER TAIL_NUM
##    <dbl> <dbl>        <dbl>       <dbl> <date>     <chr>      <chr>
##  1  2020     1            1           3 2020-01-01 AA         N110AN
##  2  2020     1            2           4 2020-01-02 AA         N111ZM
##  3  2020     1            3           5 2020-01-03 AA         N108NN
##  4  2020     1            4           6 2020-01-04 AA         N102NN
##  5  2020     1            5           7 2020-01-05 AA         N113AN
##  6  2020     1            6           1 2020-01-06 AA         N103NN
##  7  2020     1            7           2 2020-01-07 AA         N113AN
```

```
## 8 2020 1 8 3 2020-01-08 AA N106NN
## 9 2020 1 9 4 2020-01-09 AA N102NN
## 10 2020 1 10 5 2020-01-10 AA N117AN
## # ... with 2,034 more rows, and 27 more variables: OP_CARRIER_FL_NUM <dbl>,
## #   ORIGIN <chr>, ORIGIN_CITY_NAME <chr>, DEST <chr>, DEST_CITY_NAME <chr>,
## #   CRS_DEP_TIME <dbl>, DEP_TIME <dbl>, DEP_DELAY <dbl>, TAXI_OUT <dbl>,
## #   WHEELS_OFF <dbl>, WHEELS_ON <dbl>, TAXI_IN <dbl>, CRS_ARR_TIME <dbl>,
## #   ARR_TIME <dbl>, ARR_DELAY <dbl>, CANCELLED <dbl>, CANCELLATION_CODE <lgl>,
## #   DIVERTED <dbl>, CRS_ELAPSED_TIME <dbl>, ACTUAL_ELAPSED_TIME <dbl>,
## #   AIR_TIME <dbl>, DISTANCE <dbl>, CARRIER_DELAY <dbl>, WEATHER_DELAY <dbl>,
## #   NAS_DELAY <dbl>, SECURITY_DELAY <dbl>, LATE_AIRCRAFT_DELAY <dbl>
```

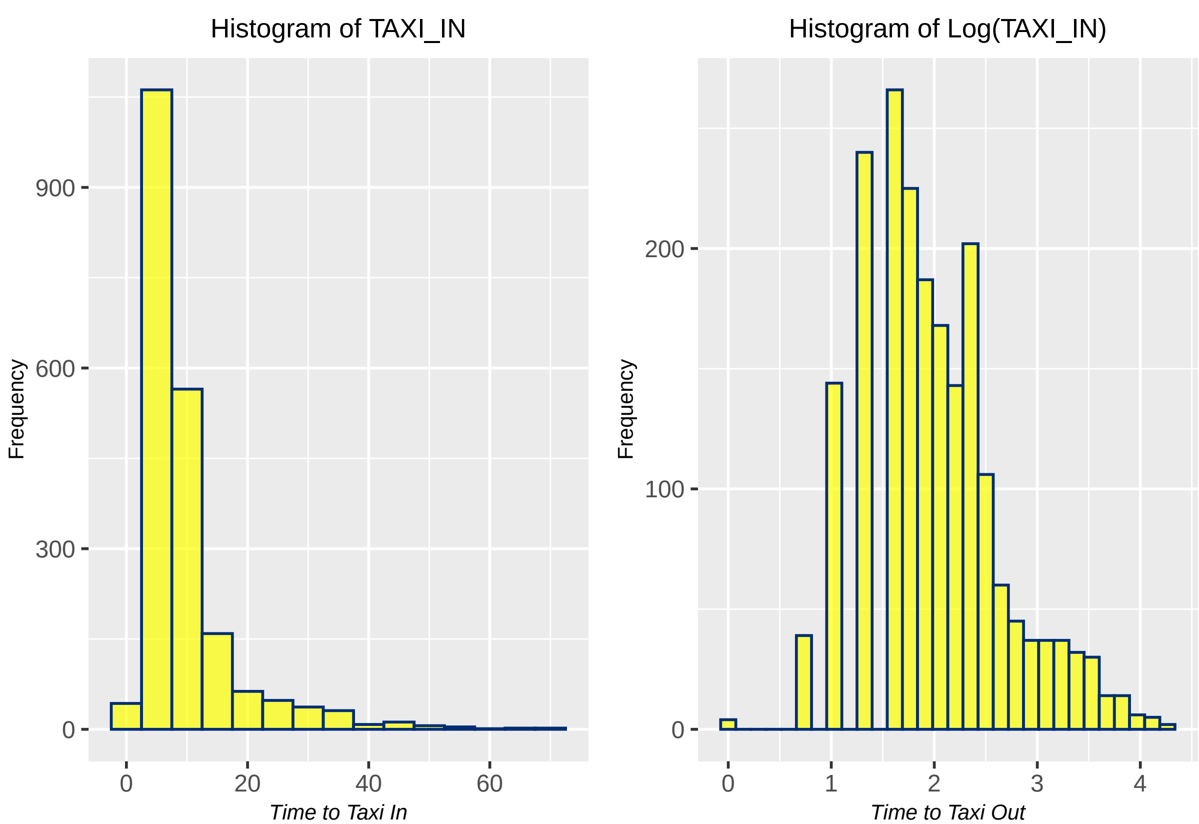
INDIVIDUAL PREDICTORS

Taxi Histograms

```
pTAXI_IN <- ggplot(data = flights, aes(x = TAXI_IN)) +
  geom_histogram(binwidth = 5, fill = "#FFFF00", color = "#002D72", alpha = .7) +
  labs(x = "Time to Taxi In",
       y = "Frequency",
       title = "Histogram of TAXI_IN") +
  theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))



 # ggplot(train_data, mapping = aes(x = St2)) +
 #   geom_histogram(binwidth =2.5, fill = "#FFFF00", color = "#002D72", alpha = .7) +
 #   labs(x = xlab(bquote('St^2')),
 #       # xlab(bquote('Assimilation ('*mu~ 'mol' ~CO[2]~ m^-2~s^-1*')')),
 #        y = "Frequency",
 #        title = "Histogram of Stokes Number, Squared") +
 #   theme(plot.title = element_text(size = 10,hjust = 0.5),
 #         plot.subtitle = element_text(hjust = 0.5),
 #         axis.title.x.bottom = element_text(size = 8, face = "italic"),
 #         axis.title.y.left = element_text(size = 8))


pTAXI_OUT <- ggplot(data = flights, aes(x = TAXI_OUT)) +
  geom_histogram(binwidth = 5, fill = "#FFFF00", color = "#002D72", alpha = .7) +
  labs(x = "Time to Taxi Out",
       y = "Frequency",
       title = "Histogram of TAXI_OUT") +
  theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))


flights$log_TAXI_OUT <- log(flights$TAXI_OUT)
flights$log_TAXI_IN <- log(flights$TAXI_IN)
```

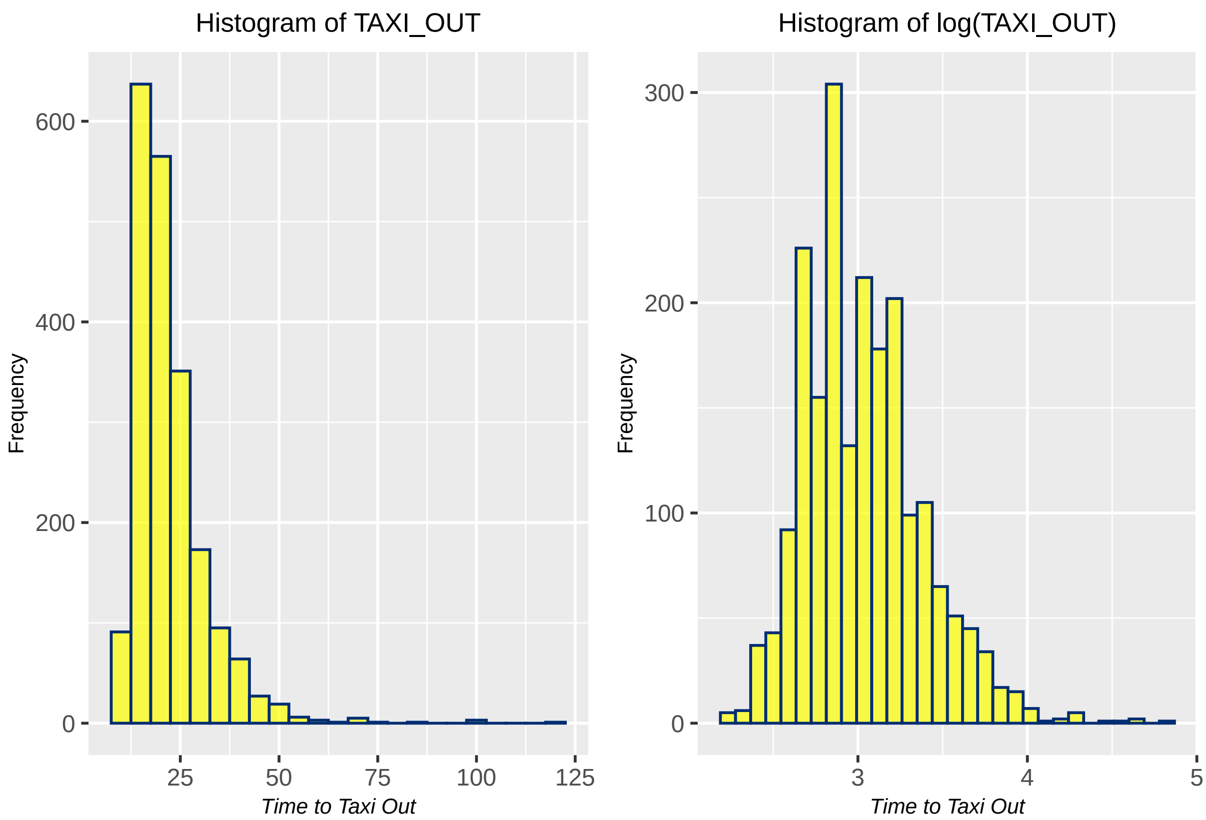
```r
plog_TAXI_OUT <- ggplot(data = flights, aes(x = log_TAXI_OUT)) +
  geom_histogram(fill = "#FFFF00", color = "#002D72", alpha = .7) +
  labs(x = "Time to Taxi Out",
       y = "Frequency",
       title = "Histogram of log(TAXI_OUT)") +
  theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))


plog_TAXI_IN <- ggplot(data = flights, aes(x = log_TAXI_IN)) +
  geom_histogram(fill = "#FFFF00", color = "#002D72", alpha = .7) +
  labs(x = "Time to Taxi Out",
       y = "Frequency",
       title = "Histogram of Log(TAXI_IN)") +
  theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))

pTAXI_IN + plog_TAXI_IN
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



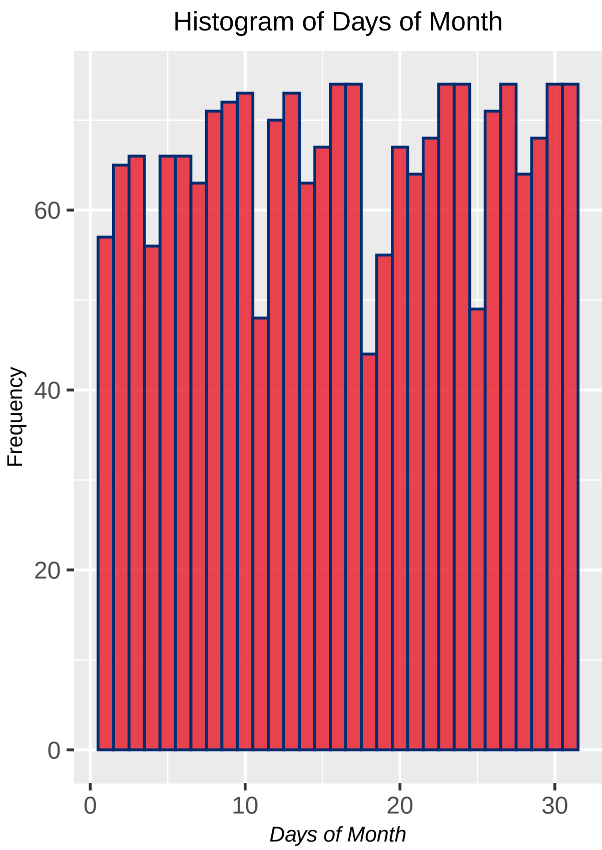

```r
pTAXI_OUT + plog_TAXI_OUT
```

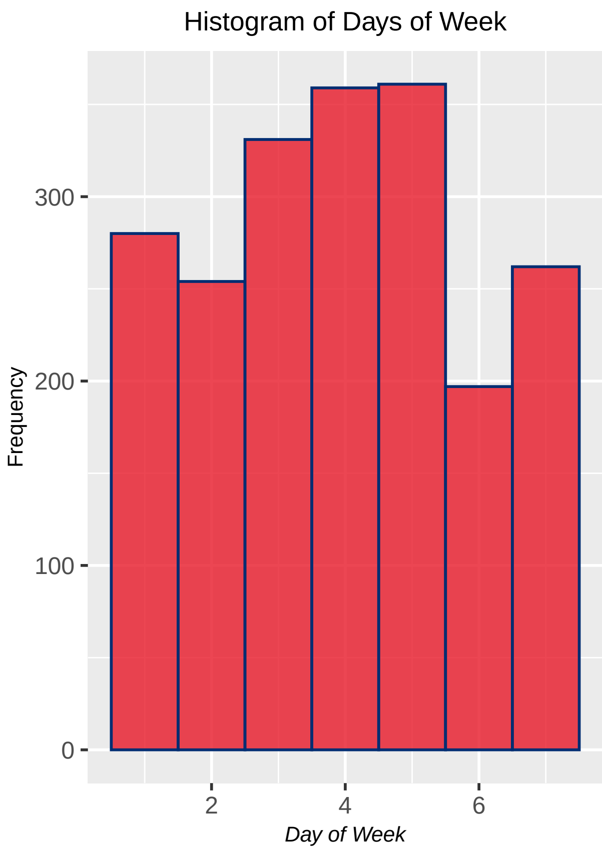
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Days of Month and Week

```r
p02 <- ggplot(data = flights, aes(x = DAY_OF_MONTH)) +
  geom_histogram(binwidth = 1, fill = "#E81828", color = "#002D72", alpha = .8) +
  labs(x = "Days of Month",
       y = "Frequency",
       title = "Histogram of Days of Month") +
    theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))

p03 <- ggplot(data = flights, aes(x = DAY_OF_WEEK)) +
  geom_histogram(binwidth = 1, fill = "#E81828", color = "#002D72", alpha = .8) +
  labs(x = "Day of Week",
       y = "Frequency",
       title = "Histogram of Days of Week") +
    theme(plot.title = element_text(size = 10,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))

grid.arrange(p02, p03, nrow = 1)
```

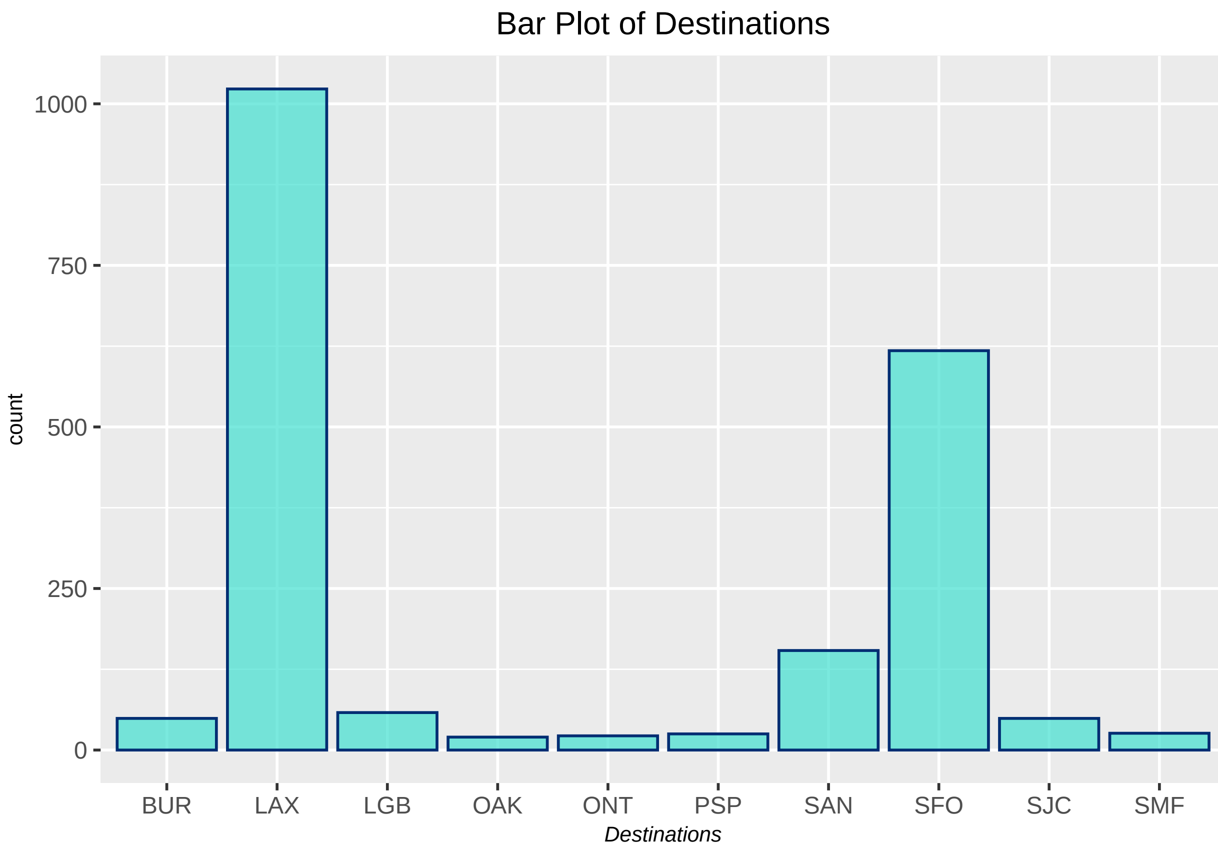




Destination Locations

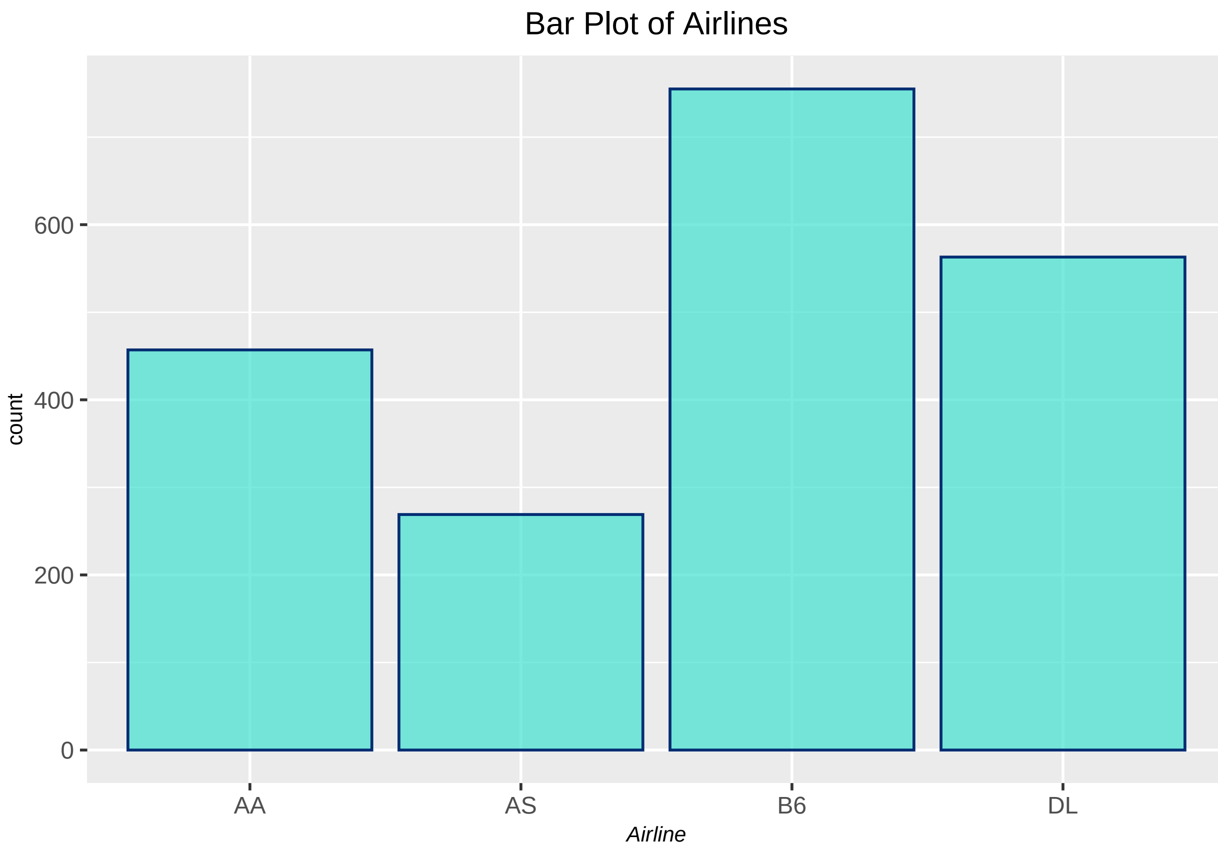
Origin is all JFK, but we could consider the different destination locations.

```
ggplot(data = flights, aes(x = DEST)) +
  geom_bar(fill = "#40E0D0", color = "#002D72", alpha = .7) +
  labs(x = "Destinations",
      title = "Bar Plot of Destinations") +
  theme(plot.title = element_text(size = 12,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))
```

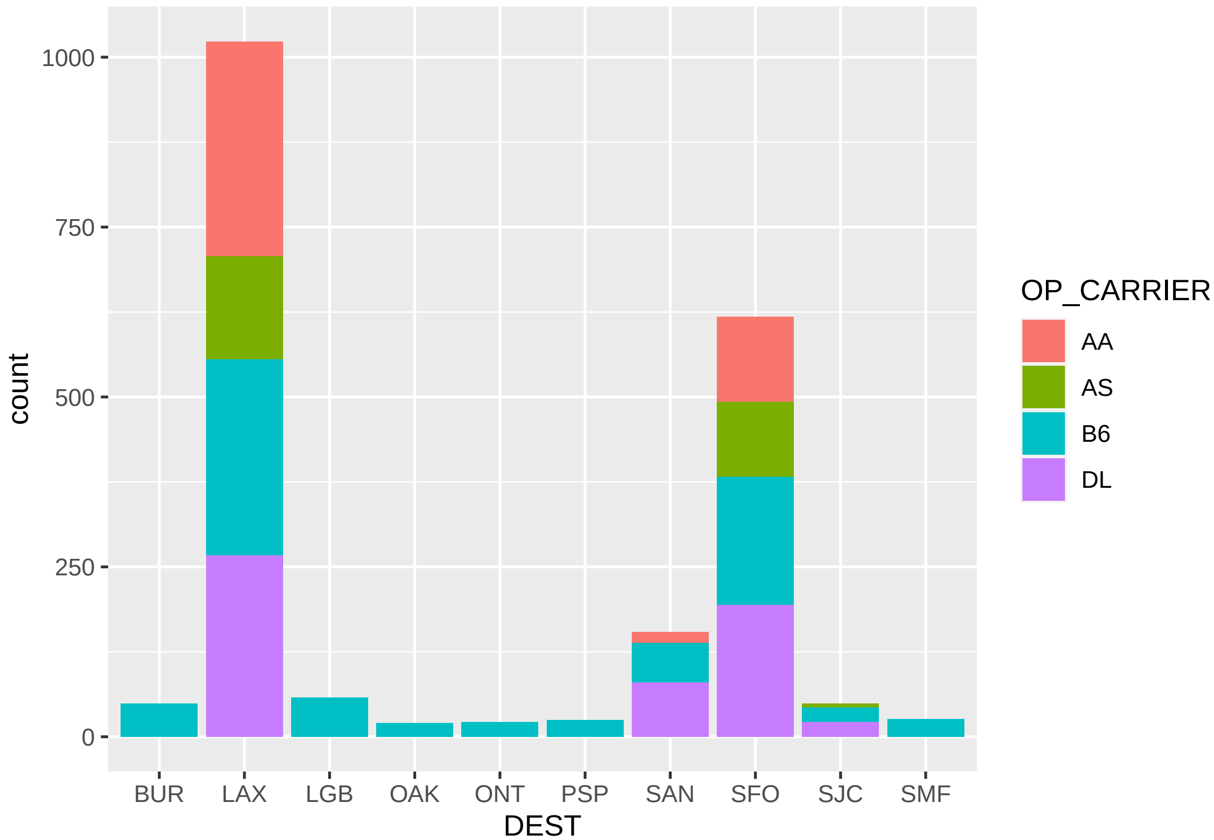


Airlines

```
ggplot(data = flights, aes(x = OP_CARRIER)) +
  geom_bar(fill = "#40E0D0", color = "#002D72", alpha = .7) +
  labs(x = "Airline",
       title = "Bar Plot of Airlines") +
  theme(plot.title = element_text(size = 12,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))
```

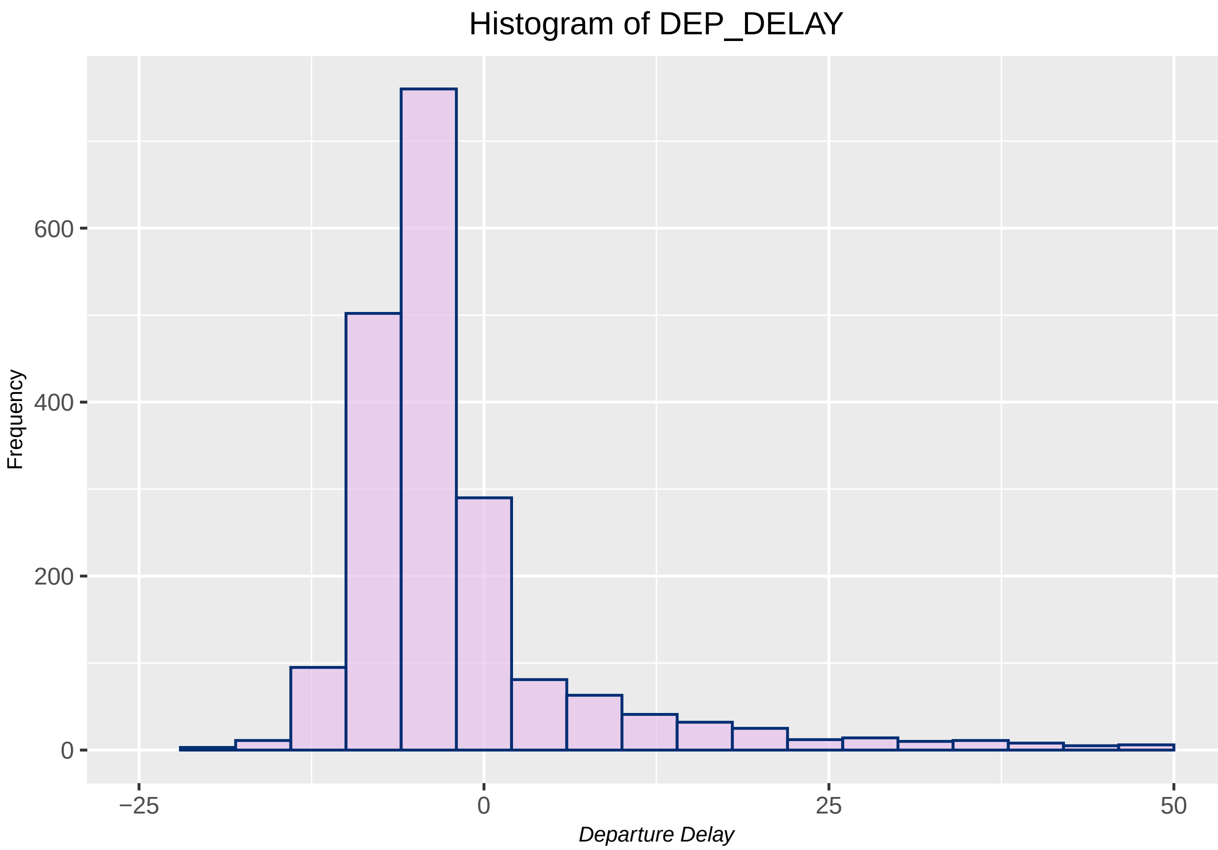



```r
ggplot(data = flights, aes(x = DEST, fill = OP_CARRIER)) +
  geom_bar()
```

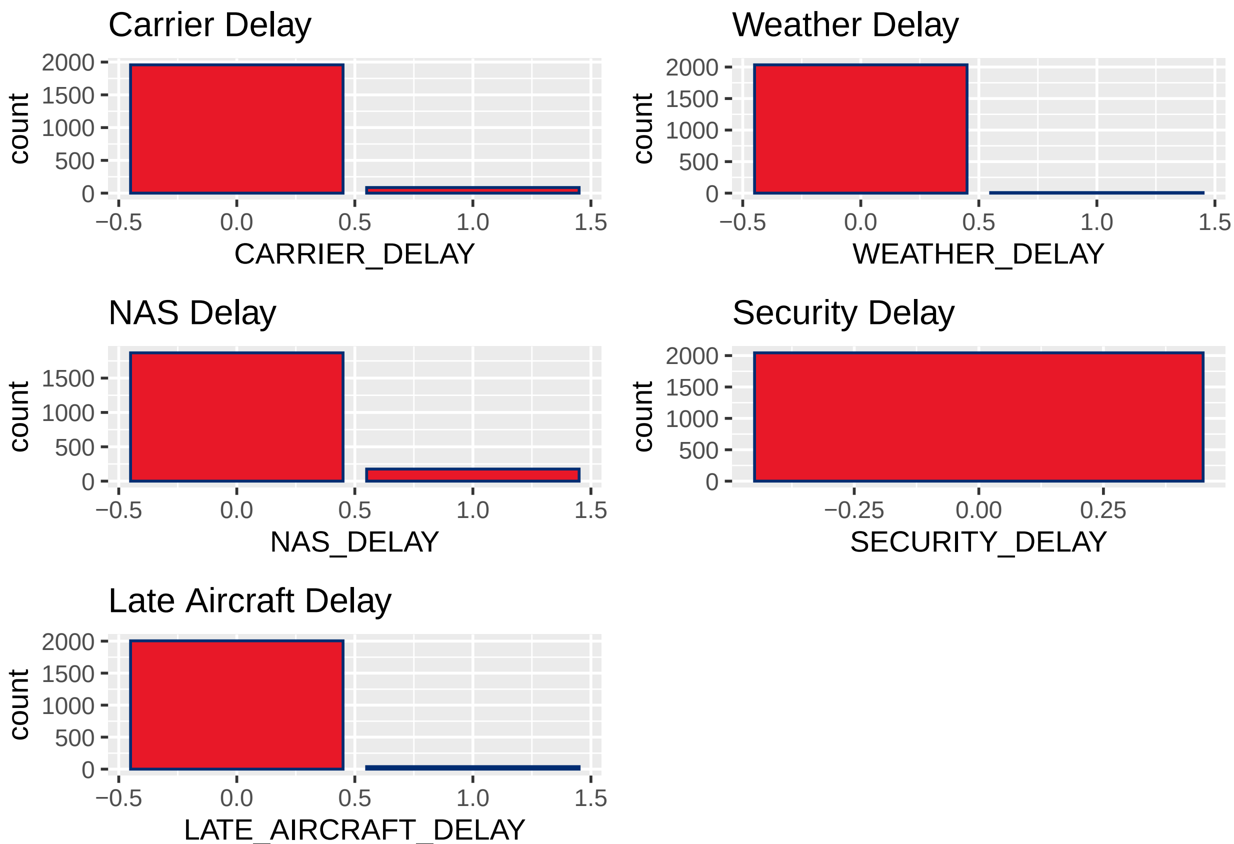


Depart Delay Histogram

```
ggplot(data = flights, aes(x = DEP_DELAY)) +
  geom_histogram(binwidth = 4, fill = "#e9c2ed", color = "#002D72", alpha = 0.7) +
  xlim(-25, 50) +
  labs(x = "Departure Delay",
       y = "Frequency",
       title = "Histogram of DEP_DELAY") +
  theme(plot.title = element_text(size = 12,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))
```




```
p1 <- ggplot(data = flights, aes(x = CARRIER_DELAY)) +
  geom_bar(fill = "#E81828", color = "#002D72") +
  labs(title = "Carrier Delay")

p2 <- ggplot(data = flights, aes(x = WEATHER_DELAY)) +
  geom_bar(fill = "#E81828", color = "#002D72") +
  labs(title = "Weather Delay")

p3 <- ggplot(data = flights, aes(x = NAS_DELAY)) +
  geom_bar(fill = "#E81828", color = "#002D72") +
  labs(title = "NAS Delay")

p4 <- ggplot(data = flights, aes(x = SECURITY_DELAY)) +
  geom_bar(fill = "#E81828", color = "#002D72") +
  labs(title = "Security Delay")

p5 <- ggplot(data = flights, aes(x = LATE_AIRCRAFT_DELAY)) +
  geom_bar(fill = "#E81828", color = "#002D72") +
  labs(title = "Late Aircraft Delay")

grid.arrange(p1,p2,p3,p4,p5, nrow = 3)
```



From this EDA of the categorical variables, we probably should not perform analysis with `SECURITY_DELAY` since all of them are classified as 0.

```
flights %>%
  count(WEATHER_DELAY)
```

```
## # A tibble: 2 x 2
##   WEATHER_DELAY     n
##           <dbl> <int>
## 1             0  2035
## 2             1     9
```

Furthermore, only 9 flights are classified with a weather delay, so it may not be good for our model to include this as a variable for right now.

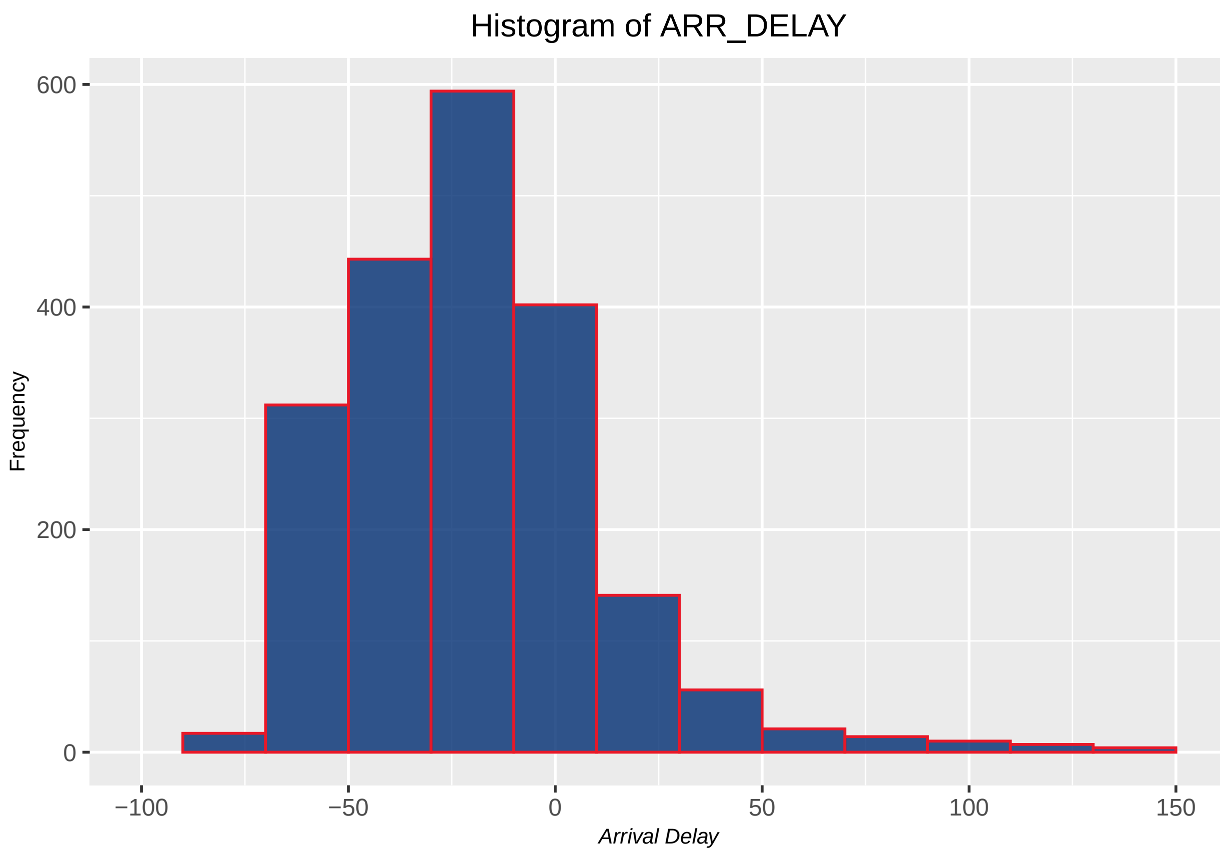
Overall, the categorical delay predictors I would think we could use are: Carrier Delay, NAS Delay, and Late Aircraft Delay

RESPONSE VARIABLE: ARRIVAL DELAY TIME

I just made it a different color so that when I scroll up to look at distributions I can easily tell the response from predictors (definitely can change at the end).

```
ggplot(data = flights, aes(x = ARR_DELAY)) +
  geom_histogram(binwidth = 20, fill = "#002D72", color = "#E81828", alpha = 0.8) +
  xlim(-100, 150) +
  labs(x = "Arrival Delay",
       y = "Frequency",
       title = "Histogram of ARR_DELAY") +
```

```r
  theme(plot.title = element_text(size = 12,hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.title.x.bottom = element_text(size = 8, face = "italic"),
        axis.title.y.left = element_text(size = 8))
```

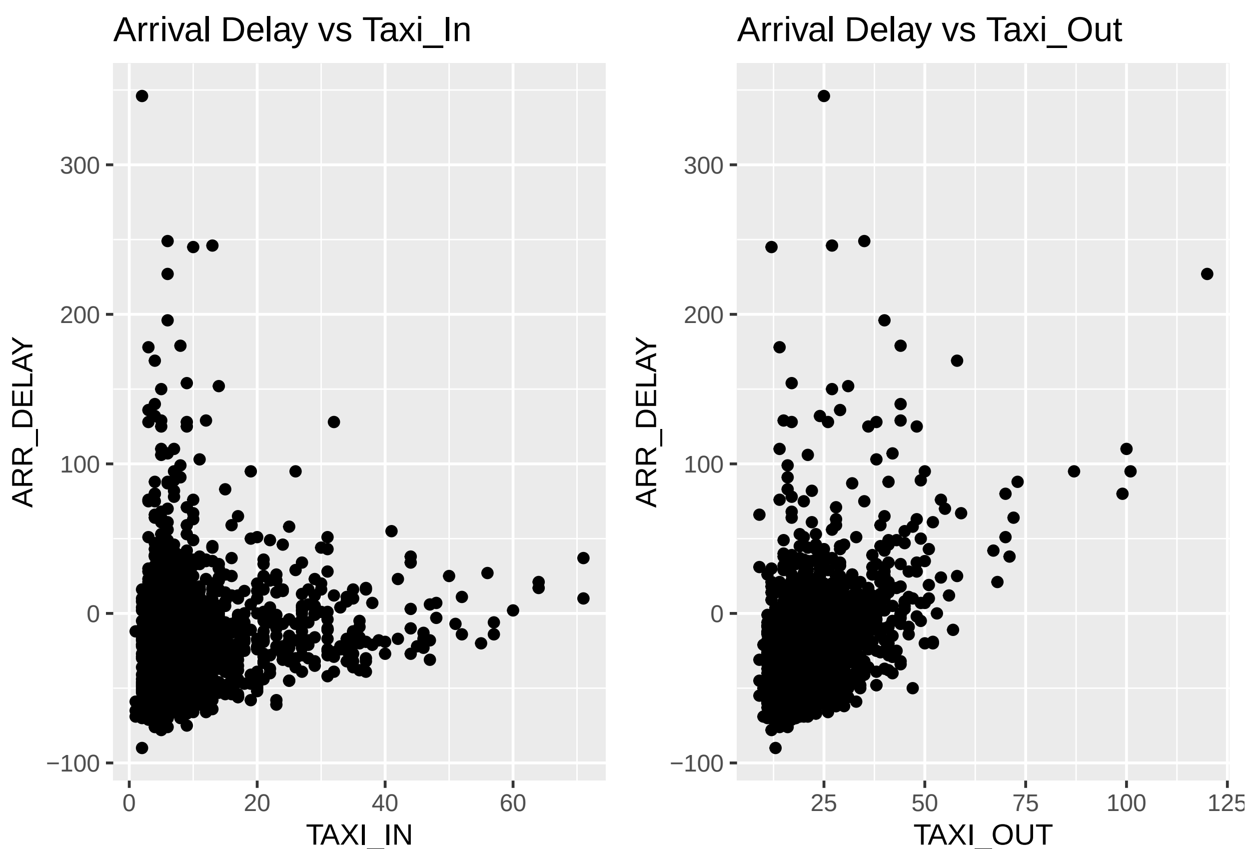


PREDICTORS VS RESPONSE

ARR_DELAY and TAXI_IN / TAXI_OUT

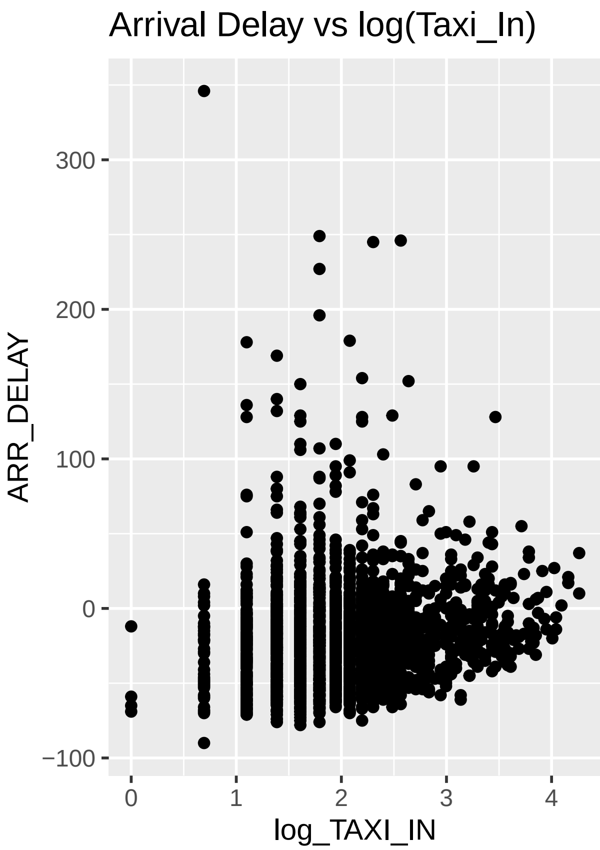
```r
p6 <- ggplot(data = flights, aes(y = ARR_DELAY, x = TAXI_IN)) +
  geom_point() +
  labs(title = "Arrival Delay vs Taxi_In")

p7 <- ggplot(data = flights, aes(y = ARR_DELAY, x = TAXI_OUT)) +
  geom_point() +
  labs(title = "Arrival Delay vs Taxi_Out")

grid.arrange(p6,p7, nrow = 1)
```

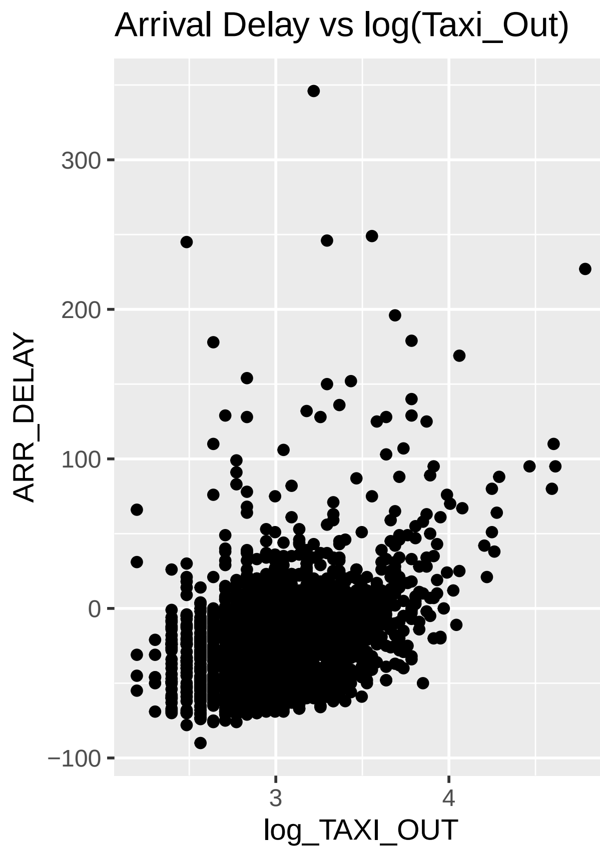
```
## Warning: Removed 11 rows containing missing values (geom_point).

## Warning: Removed 11 rows containing missing values (geom_point).
```




```
plog6 <- ggplot(data = flights, aes(y = ARR_DELAY, x = log_TAXI_IN)) +
  geom_point() +
  labs(title = "Arrival Delay vs log(Taxi_In)")

plog7 <- ggplot(data = flights, aes(y = ARR_DELAY, x = log_TAXI_OUT)) +
  geom_point() +
  labs(title = "Arrival Delay vs log(Taxi_Out)")
grid.arrange(plog6,plog7, nrow = 1)
```

```
## Warning: Removed 11 rows containing missing values (geom_point).

## Warning: Removed 11 rows containing missing values (geom_point).
```

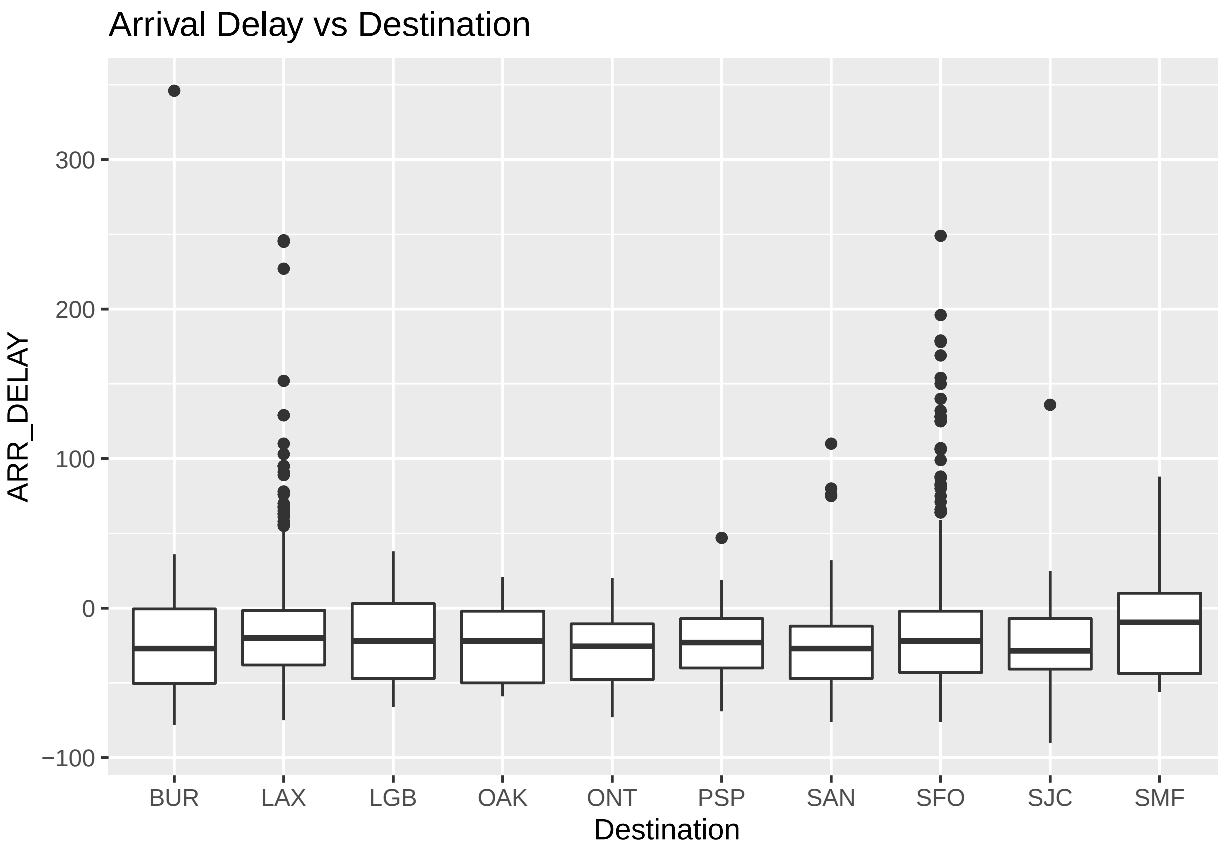




These plots above suggest that we may want to transform the variables at some point.

```
ggplot(data = flights, aes(y = ARR_DELAY, x = DEST)) +
  geom_boxplot() +
  labs(x = "Destination",
       title = "Arrival Delay vs Destination")
```

```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```

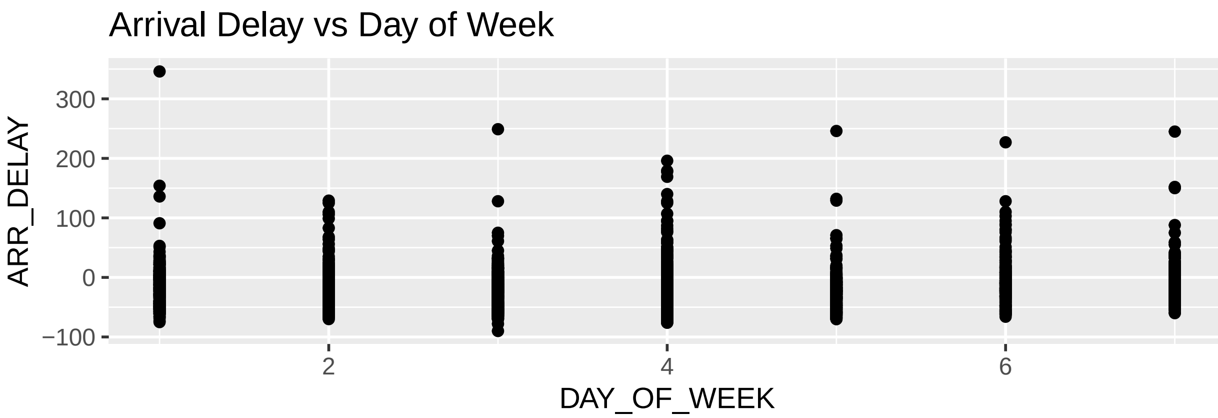


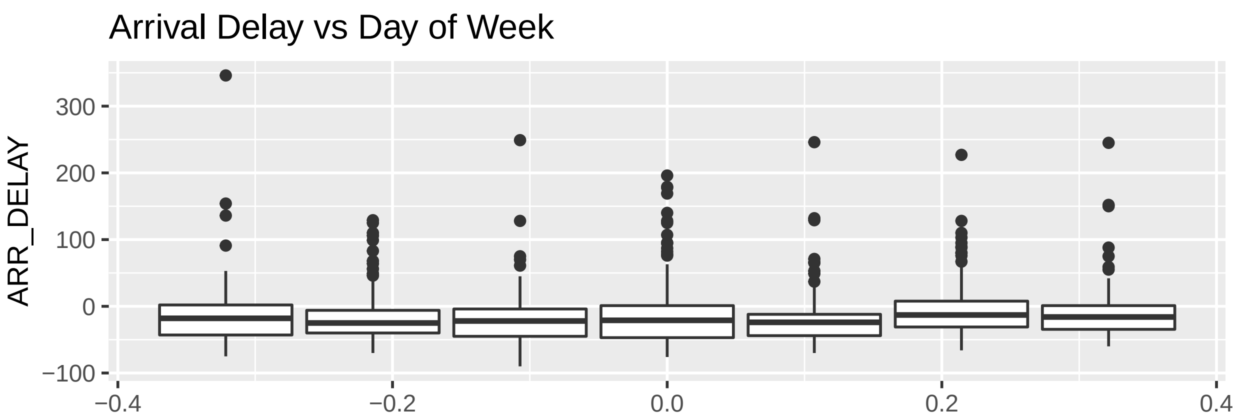
ARR_DELAY and DAY_OF_WEEK

```
p8 <- ggplot(data = flights, aes(y = ARR_DELAY, x = DAY_OF_WEEK)) +
  geom_point() +
  labs(title = "Arrival Delay vs Day of Week")

p9 <- ggplot(data = flights, aes(y = ARR_DELAY, group = DAY_OF_WEEK)) +
  geom_boxplot() +
  labs(title = "Arrival Delay vs Day of Week")

grid.arrange(p8,p9, nrow = 2)
```

```
## Warning: Removed 11 rows containing missing values (geom_point).
```

```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```



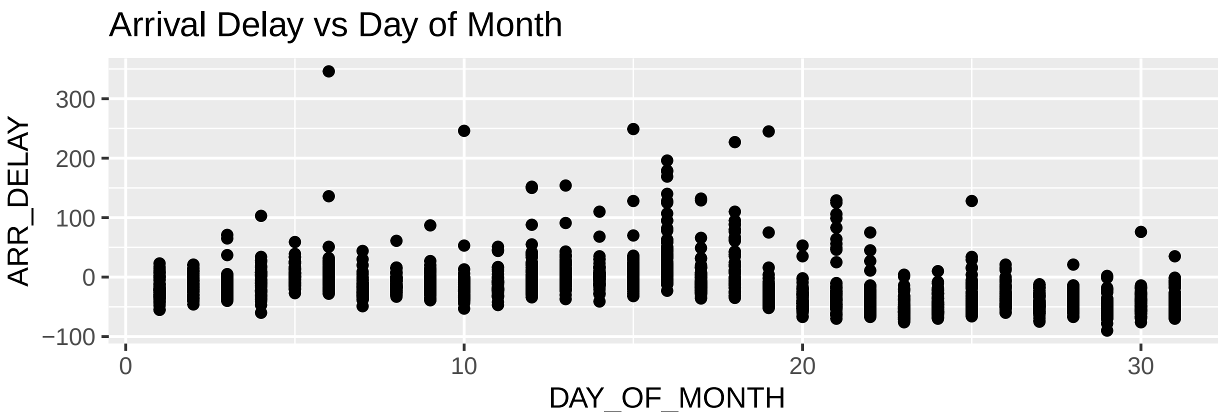


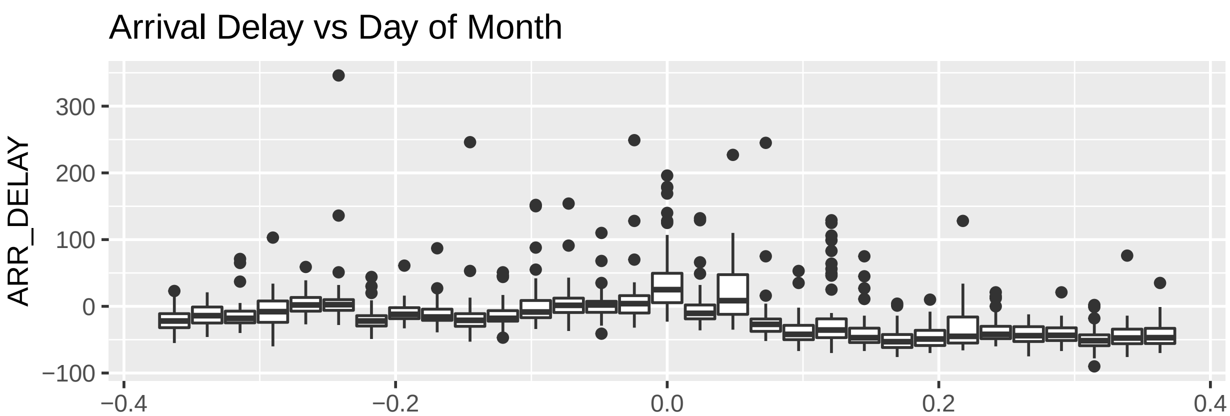
ARR_DELAY and DAY_OF_MONTH

```
p10 <- ggplot(data = flights, aes(y = ARR_DELAY, x = DAY_OF_MONTH)) +
  geom_point() +
  labs(title = "Arrival Delay vs Day of Month")

p11 <- ggplot(data = flights, aes(y = ARR_DELAY, group = DAY_OF_MONTH)) +
  geom_boxplot() +
  labs(title = "Arrival Delay vs Day of Month")

grid.arrange(p10, p11, nrow = 2)
```

```
## Warning: Removed 11 rows containing missing values (geom_point).
```

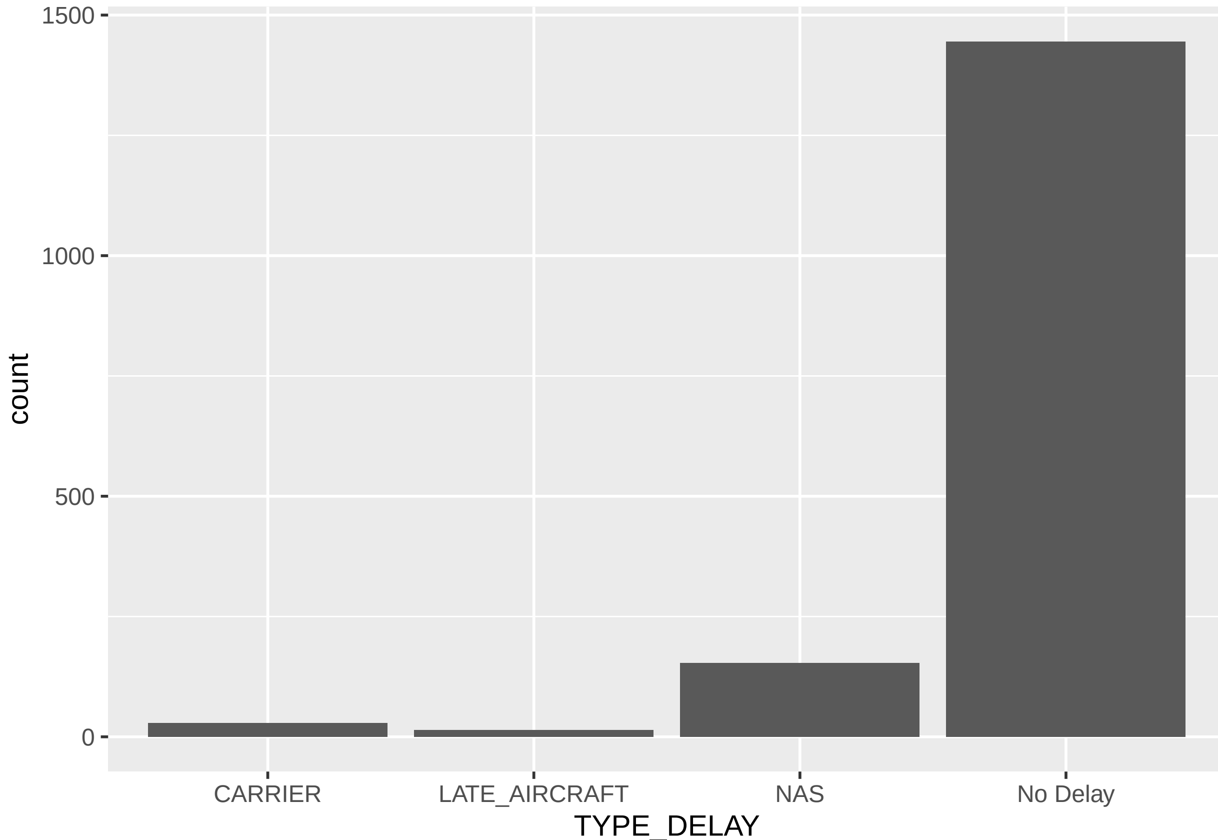
```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```





Further Data Cleaning

```
# take only SFO/LAX since all 4 carriers fly there
flights <- flights %>%
  filter(DEST == "SFO" | DEST == "LAX") %>%
  mutate(TYPE_DELAY = case_when(NAS_DELAY == 1 ~ "NAS",
                                CARRIER_DELAY == 1 ~ "CARRIER",
                                LATE_AIRCRAFT_DELAY == 1 ~ "LATE_AIRCRAFT",
                                TRUE ~ "No Delay"))

ggplot(data = flights, aes(x = TYPE_DELAY)) +
  geom_bar()
```




```
unique(flights$TYPE_DELAY)
```

```
## [1] "No Delay"      "NAS"           "LATE_AIRCRAFT" "CARRIER"
```

SPLITTING DATA

```
set.seed(1234)
flights <- flights %>%
  mutate(id = row_number())
train <- flights %>%
  sample_frac(0.8)
test <- anti_join(flights, train, by = "id")
```

LINEAR MODELS

Variables that I think we could explore: department delay time, days of month, days of week, taxi-in, taxi-out, destination, Carrier Delay, NAS Delay, and Late Aircraft Delay.

Full Model

```
lm.01 <- lm(ARR_DELAY ~ DEP_DELAY + DAY_OF_WEEK + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME + log_

#plot(lm.01)
#summary(lm.01)
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## The following object is masked from 'package:patchwork':
##
##     area

## The following objects are masked from 'package:openintro':
##
##     housing, mammals
```

```r
step_model <- stepAIC(lm.01, direction = "backward", trace = FALSE)
#summary(step_model)

lm.02 <- lm(ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME + log_TAXI_OUT + log
#summary(lm.02)
#anova(step_model, lm.02)

lm.03 <- lm(ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME + log_TAXI_OUT + log

#anova(lm.02, lm.03)


lm.04 <- lm(ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME + log_TAXI_OUT + log

#anova(lm.03, lm.04)

final_model <- lm(ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME + log_TAXI_OU

anova(lm.04, final_model)
```

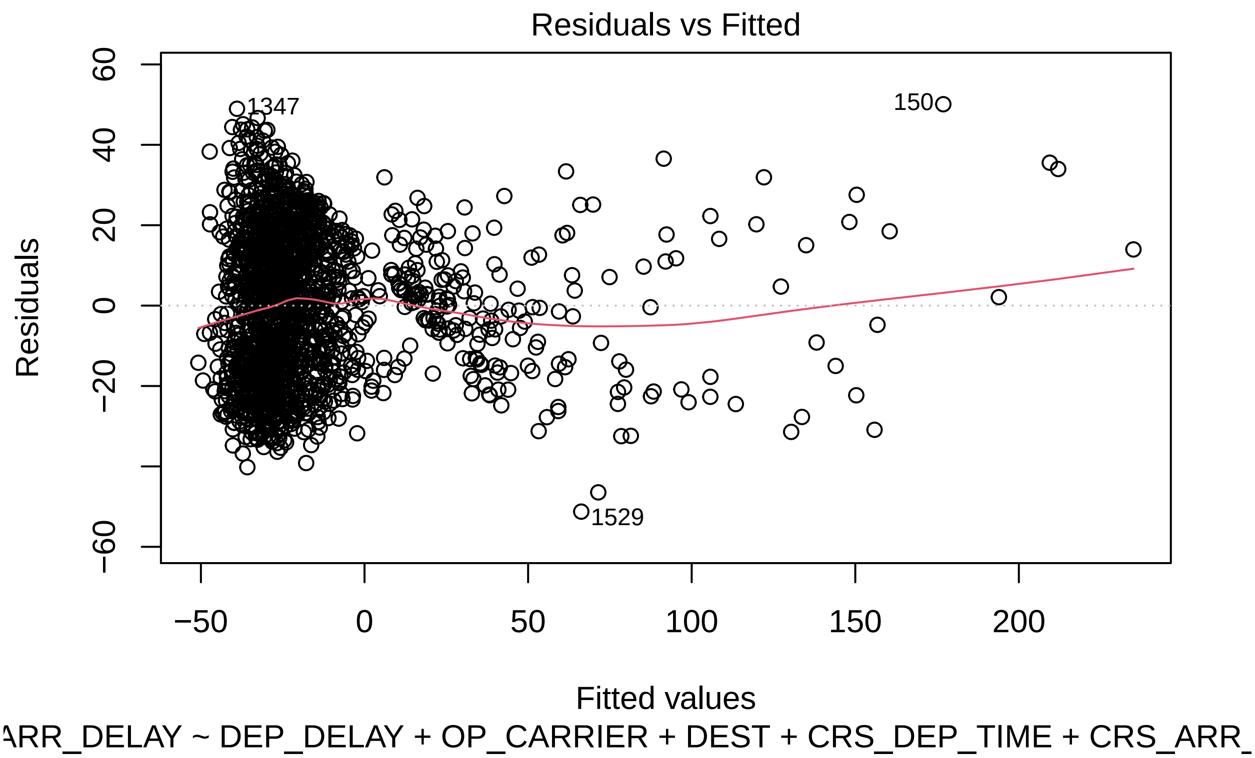
```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME +
##     log_TAXI_OUT + log_TAXI_IN + TYPE_DELAY + OP_CARRIER:DEST +
##     DEST:log_TAXI_IN + CRS_ARR_TIME:log_TAXI_IN
## Model 2: ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_TIME +
##     log_TAXI_OUT + log_TAXI_IN + TYPE_DELAY + OP_CARRIER:DEST +
##     DEST:log_TAXI_IN + CRS_ARR_TIME:log_TAXI_IN + log_TAXI_OUT:DEP_DELAY
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1   1620 532909
## 2   1619 530347  1    2561.6 7.8199 0.005229 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

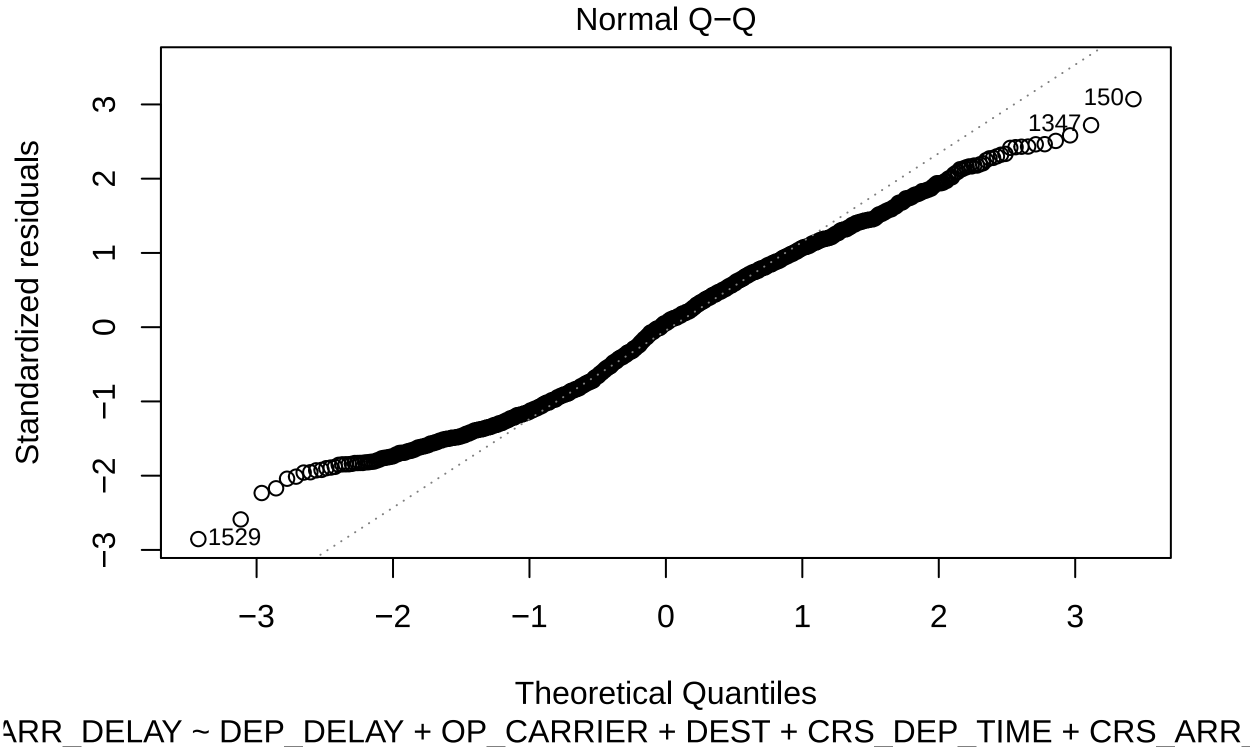
```r
summary(final_model)
```

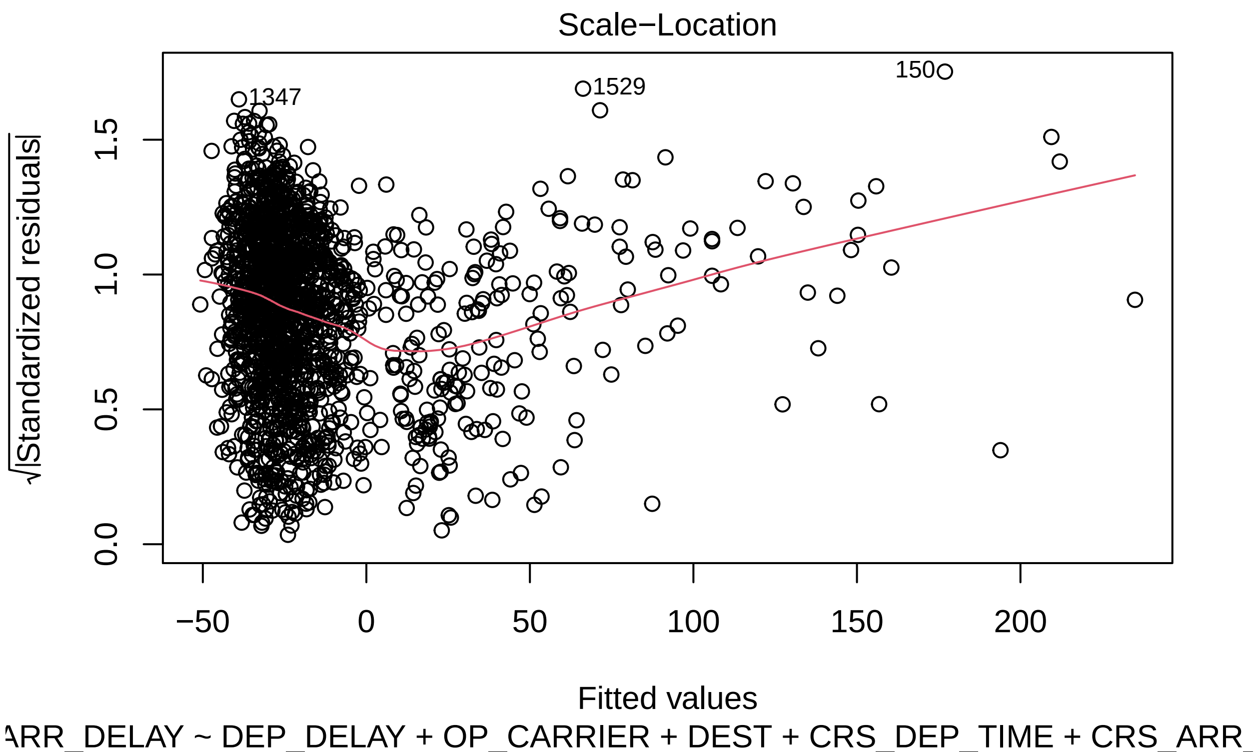
```
##
## Call:
## lm(formula = ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME +
```

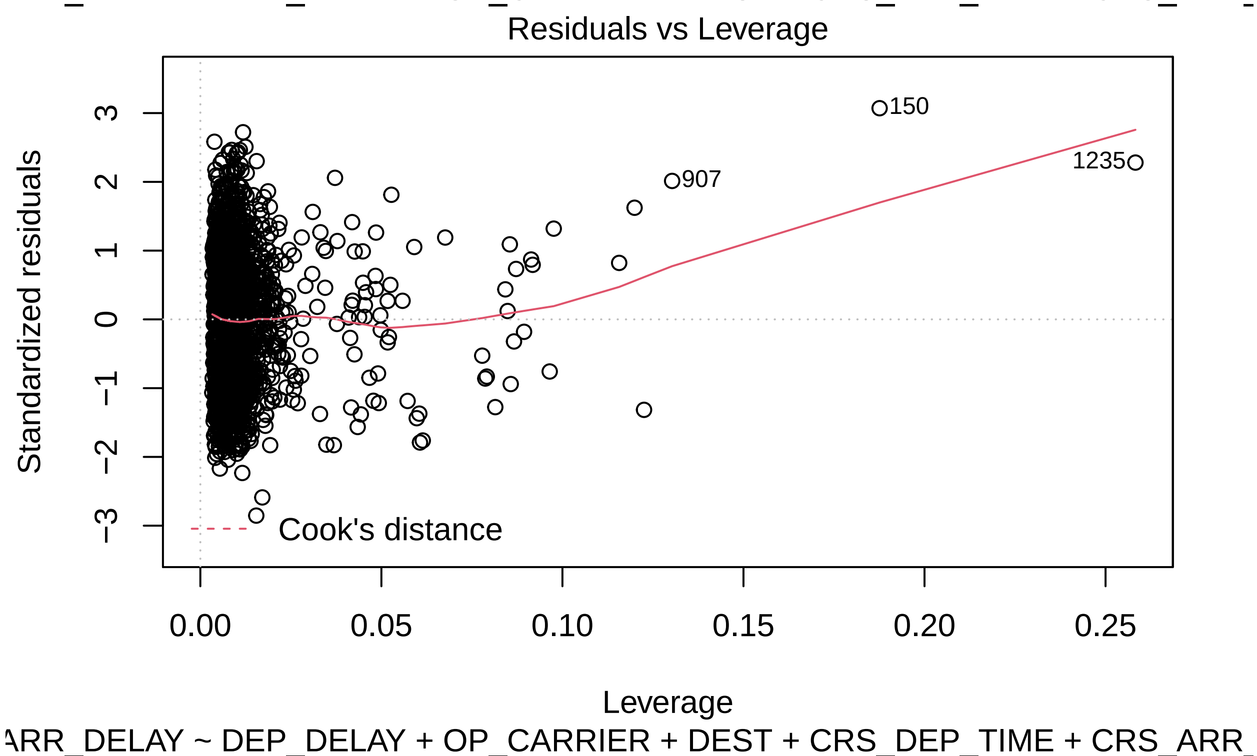
```
##     CRS_ARR_TIME + log_TAXI_OUT + log_TAXI_IN + TYPE_DELAY +
##     OP_CARRIER:DEST + DEST:log_TAXI_IN + CRS_ARR_TIME:log_TAXI_IN +
##     log_TAXI_OUT:DEP_DELAY, data = flights)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.248 -15.188   0.999  13.708  50.098
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -7.118e+01  7.864e+00  -9.052  < 2e-16 ***
## DEP_DELAY                 5.279e-01  1.303e-01   4.052 5.31e-05 ***
## OP_CARRIERAS             -4.897e+00  1.841e+00  -2.659  0.00791 **
## OP_CARRIERB6              4.484e+00  1.488e+00   3.014  0.00262 **
## OP_CARRIERDL             -2.239e+00  1.519e+00  -1.474  0.14066
## DESTSFO                   5.662e+00  4.017e+00   1.410  0.15882
## CRS_DEP_TIME             -4.224e-03  9.788e-04  -4.316 1.69e-05 ***
## CRS_ARR_TIME             -7.569e-03  2.794e-03  -2.709  0.00682 **
## log_TAXI_OUT              2.169e+01  1.458e+00  14.873  < 2e-16 ***
## log_TAXI_IN               3.738e+00  2.214e+00   1.688  0.09156 .
## TYPE_DELAYLATE_AIRCRAFT  -7.634e+00  6.013e+00  -1.270  0.20443
## TYPE_DELAYNAS             2.343e+01  4.159e+00   5.633 2.08e-08 ***
## TYPE_DELAYNo Delay       -1.676e+01  4.105e+00  -4.083 4.67e-05 ***
## OP_CARRIERAS:DESTSFO      4.650e+00  2.989e+00   1.556  0.11990
## OP_CARRIERB6:DESTSFO     -4.672e+00  2.581e+00  -1.810  0.07046 .
## OP_CARRIERDL:DESTSFO      1.700e-01  2.611e+00   0.065  0.94810
## DESTSFO:log_TAXI_IN      -3.400e+00  1.790e+00  -1.899  0.05768 .
## CRS_ARR_TIME:log_TAXI_IN  2.740e-03  1.243e-03   2.204  0.02764 *
## DEP_DELAY:log_TAXI_OUT    1.117e-01  3.993e-02   2.796  0.00523 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.1 on 1619 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.7425, Adjusted R-squared:  0.7397
## F-statistic: 259.4 on 18 and 1619 DF,  p-value: < 2.2e-16
```

```r
plot(final_model)
```


Residuals vs Fitted
Residuals
Fitted values
ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_
1347
150
1529
60
40
20
0
−20
−60
−50
0
50
100
150
200


Normal Q−Q
Standardized residuals
Theoretical Quantiles
ARR_DELAY ~ DEP_DELAY + OP_CARRIER + DEST + CRS_DEP_TIME + CRS_ARR_
150
1347
1529
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3






```
## SIGNIFICANT INTERACTIONS
#OP_CARRIER:DEST
#DEST:log_TAXI_IN
#CRS_DEP_TIME:DEST (***** makes zero intuitive sense - might not wanna do this)
#CRS_ARR_TIME:log_TAXI_IN
#log_TAXI_OUT:DEP_DELAY
```

```r
#log_TAXI_OUT:CRS_DEP_TIME (verrrrrry close to 0.05)
```

First, let's just fit a full linear model with all the variables we would like to explore.

```r
full_model <- lm(ARR_DELAY ~ DAY_OF_MONTH +
                   DAY_OF_WEEK +
                   TAXI_IN +
                   TAXI_OUT +
                   DEST +
                   DEP_DELAY +
                   CARRIER_DELAY +
                   NAS_DELAY +
                   LATE_AIRCRAFT_DELAY, data = train)

summary(full_model)
```

```
##
## Call:
## lm(formula = ARR_DELAY ~ DAY_OF_MONTH + DAY_OF_WEEK + TAXI_IN +
##     TAXI_OUT + DEST + DEP_DELAY + CARRIER_DELAY + NAS_DELAY +
##     LATE_AIRCRAFT_DELAY, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.659  -9.913  -1.229   9.243  46.780
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -22.18852    1.58821 -13.971   <2e-16 ***
## DAY_OF_MONTH         -1.28951    0.04418 -29.187   <2e-16 ***
## DAY_OF_WEEK          -0.28103    0.20758  -1.354   0.1760
## TAXI_IN               0.55575    0.04785  11.615   <2e-16 ***
## TAXI_OUT              0.73768    0.04368  16.887   <2e-16 ***
## DESTSFO              -0.33517    0.82901  -0.404   0.6861
## DEP_DELAY             0.89165    0.02221  40.145   <2e-16 ***
## CARRIER_DELAY         2.30229    2.30029   1.001   0.3171
## NAS_DELAY            32.68992    1.54500  21.159   <2e-16 ***
## LATE_AIRCRAFT_DELAY   5.54853    3.24643   1.709   0.0877 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.11 on 1301 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8259, Adjusted R-squared:  0.8247
## F-statistic: 685.8 on 9 and 1301 DF,  p-value: < 2.2e-16
```

Select Model with AIC

```r
library(MASS)
step_model <- stepAIC(full_model, trace = FALSE)
summary(step_model)
```

```
##
```

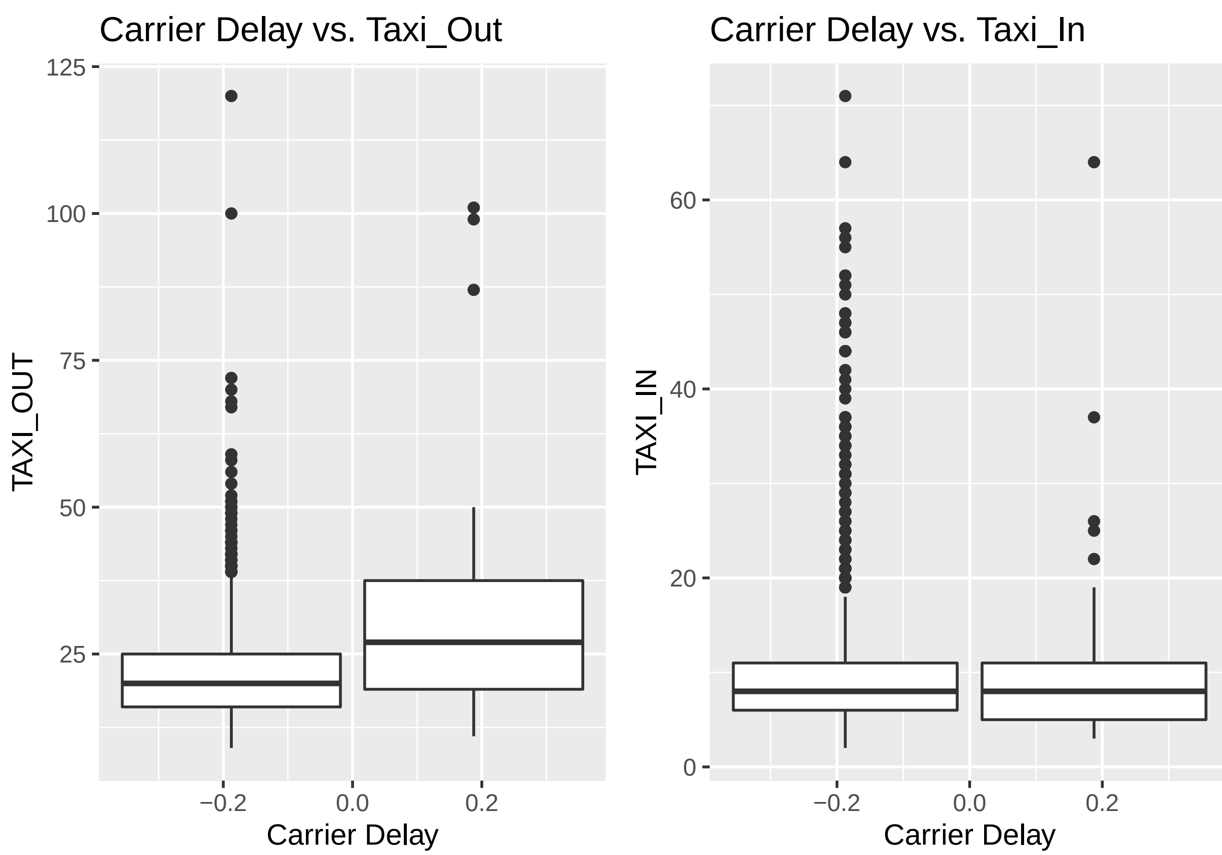
```
## Call:
## lm(formula = ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT +
##     DEP_DELAY + NAS_DELAY + LATE_AIRCRAFT_DELAY, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.702 -10.034  -1.314   9.034  46.852
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -23.31594    1.34840  -17.29   <2e-16 ***
## DAY_OF_MONTH         -1.28947    0.04400  -29.30   <2e-16 ***
## TAXI_IN               0.55710    0.04637   12.01   <2e-16 ***
## TAXI_OUT              0.73506    0.04347   16.91   <2e-16 ***
## DEP_DELAY             0.89777    0.02100   42.76   <2e-16 ***
## NAS_DELAY            33.03098    1.50853   21.90   <2e-16 ***
## LATE_AIRCRAFT_DELAY   5.44580    3.24199    1.68   0.0932 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.12 on 1304 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8255, Adjusted R-squared:  0.8247
## F-statistic:  1028 on 6 and 1304 DF,  p-value: < 2.2e-16
```

The only variables that were removed were DAY_OF_WEEK and LATE_AIRCRAFT_DELAY. Let's continue using the step_model then.

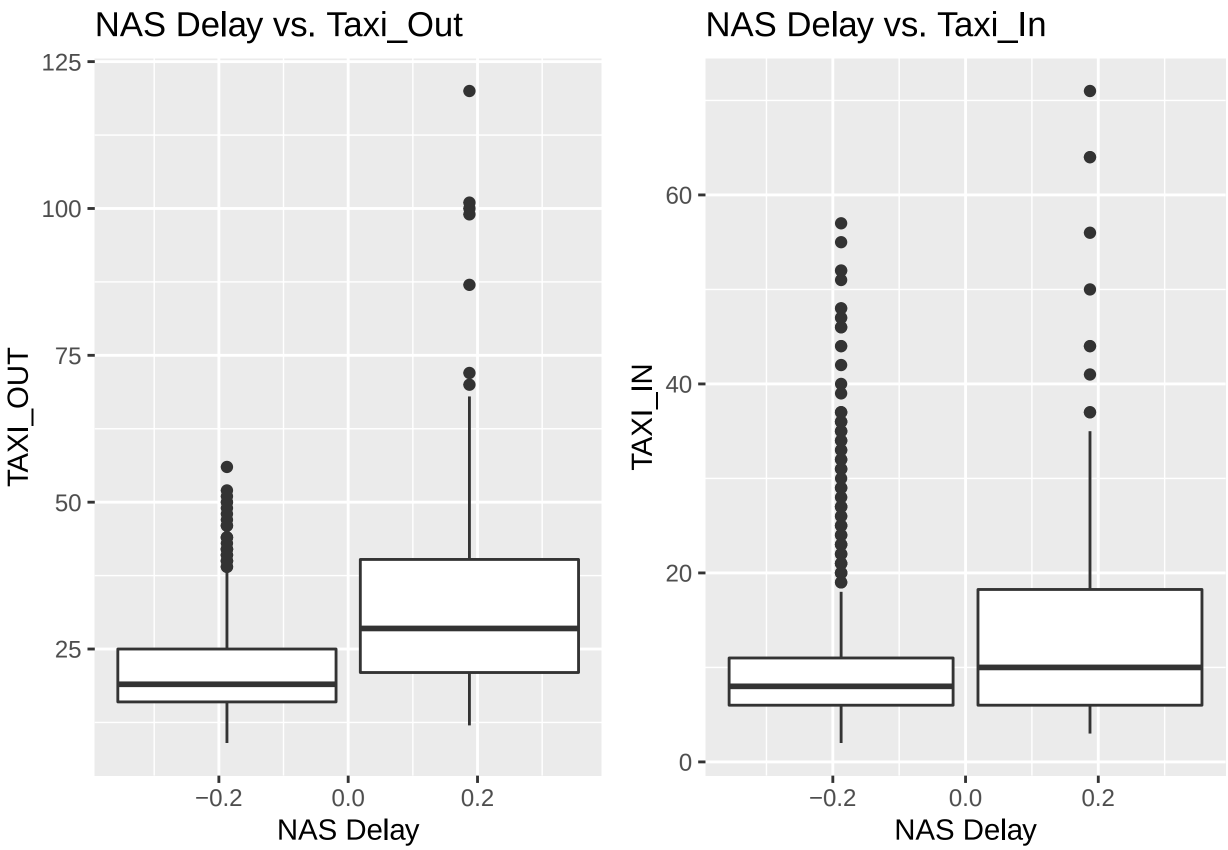
Interactions

Because there are so many levels to Destination, I don't know if we should necessarily include an interaction with this categorical variable. My suggestion would be to find interactions with carrier_delay and nas_delay.

```r
p12 <- ggplot(data = train, aes(group = CARRIER_DELAY, y = TAXI_OUT)) +
  geom_boxplot() +
  labs(title = "Carrier Delay vs. Taxi_Out",
       x = "Carrier Delay")

p13 <- ggplot(data = train, aes(group = CARRIER_DELAY, y = TAXI_IN)) +
  geom_boxplot() +
  labs(title = "Carrier Delay vs. Taxi_In",
       x = "Carrier Delay")

grid.arrange(p12, p13, nrow = 1)
```




```
p14 <- ggplot(data = train, aes(group = NAS_DELAY, y = TAXI_OUT)) +
  geom_boxplot() +
  labs(title = "NAS Delay vs. Taxi_Out",
       x = "NAS Delay")

p15 <- ggplot(data = train, aes(group = NAS_DELAY, y = TAXI_IN)) +
  geom_boxplot() +
  labs(title = "NAS Delay vs. Taxi_In",
       x = "NAS Delay")

grid.arrange(p14, p15, nrow = 1)
```



From what I'm seeing in the plots above, there could be an interaction between taxi_out and carrier_delay. There also seems to be an interaction between NAS delay and taxi_out as well as a possible one between NAS delay and taxi_in. Let's test these three interactions below.

```
# carrier vs taxi out
interaction1 <- lm(ARR_DELAY ~ DAY_OF_MONTH +
                   TAXI_IN +
                   TAXI_OUT +
                   DEST +
                   DEP_DELAY +
                   CARRIER_DELAY +
                   NAS_DELAY +
                  CARRIER_DELAY*TAXI_OUT, data = train)
# nas vs taxi out
interaction2 <- lm(ARR_DELAY ~ DAY_OF_MONTH +
                   TAXI_IN +
                   TAXI_OUT +
                   DEST +
                   DEP_DELAY +
                   CARRIER_DELAY +
                   NAS_DELAY +
                  NAS_DELAY*TAXI_OUT, data = train)

# nas vs taxi in
interaction3 <- lm(ARR_DELAY ~ DAY_OF_MONTH +
                   TAXI_IN +
                   TAXI_OUT +
```

```r
                   DEST +
                   DEP_DELAY +
                   CARRIER_DELAY +
                   NAS_DELAY +
                  NAS_DELAY*TAXI_IN, data = train)
```

```r
anova(step_model, interaction1)
```

```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEP_DELAY + NAS_DELAY +
##     LATE_AIRCRAFT_DELAY
## Model 2: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_DELAY + CARRIER_DELAY * TAXI_OUT
##   Res.Df    RSS Df Sum of Sq F Pr(>F)
## 1   1304 259813
## 2   1302 260081  2   -268.59
```

```r
anova(step_model, interaction2)
```

```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEP_DELAY + NAS_DELAY +
##     LATE_AIRCRAFT_DELAY
## Model 2: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_DELAY + NAS_DELAY * TAXI_OUT
##   Res.Df    RSS Df Sum of Sq F Pr(>F)
## 1   1304 259813
## 2   1302 260101  2   -288.77
```

```r
anova(step_model, interaction3)
```

```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEP_DELAY + NAS_DELAY +
##     LATE_AIRCRAFT_DELAY
## Model 2: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_DELAY + NAS_DELAY * TAXI_IN
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1   1304 259813
## 2   1302 258380  2    1432.9 3.6103 0.02732 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It actually seems that interaction3: NAS_DELAY and TAXI_IN is the only interaction that is statistically significant in predicting ARR_DELAY. Let's make this model our current model:

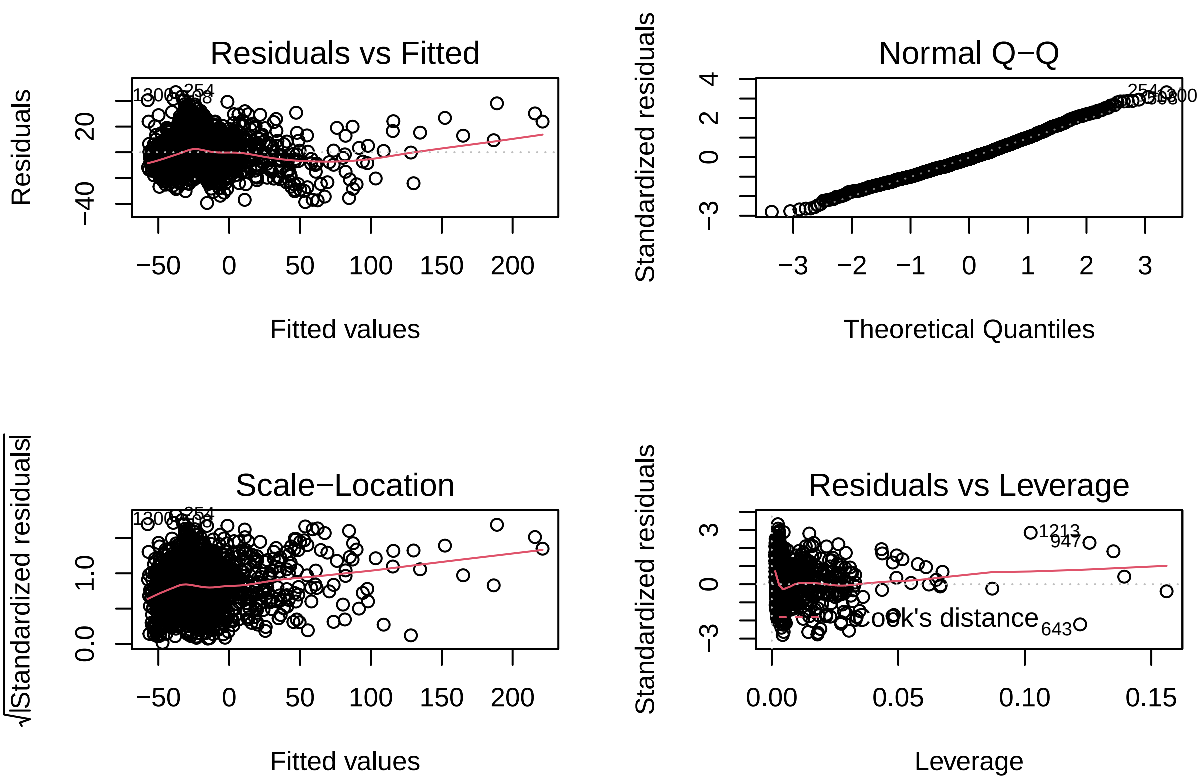
Final Linear Model

```r
current_model <- interaction3

summary(current_model)
```

```
##
## Call:
## lm(formula = ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT +
```

```
##     DEST + DEP_DELAY + CARRIER_DELAY + NAS_DELAY + NAS_DELAY *
##     TAXI_IN, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -39.388  -9.698  -1.216   8.983  46.729
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -23.48832    1.41476 -16.602  < 2e-16 ***
## DAY_OF_MONTH       -1.29353    0.04411 -29.322  < 2e-16 ***
## TAXI_IN             0.62691    0.05364  11.688  < 2e-16 ***
## TAXI_OUT            0.72130    0.04348  16.587  < 2e-16 ***
## DESTSFO            -0.39829    0.82704  -0.482  0.63019
## DEP_DELAY           0.90193    0.01975  45.666  < 2e-16 ***
## CARRIER_DELAY       2.63404    2.29653   1.147  0.25161
## NAS_DELAY          37.29871    2.17272  17.167  < 2e-16 ***
## TAXI_IN:NAS_DELAY  -0.32214    0.10933  -2.946  0.00327 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.09 on 1302 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8265, Adjusted R-squared:  0.8254
## F-statistic: 775.1 on 8 and 1302 DF,  p-value: < 2.2e-16
```

```r
par(mfrow = c(2,2))
plot(current_model)
```



The diagnostic plots above suggest that this model decently satisfies the necessary conditions to assume a linear regression.

Response (Box-Cox) Transformation

```
## AFTER SELECTED MODEL
library(EnvStats)
```

```
##
## Attaching package: 'EnvStats'

## The following object is masked from 'package:MASS':
##
##     boxcox

## The following objects are masked from 'package:stats':
##
##     predict, predict.lm

## The following object is masked from 'package:base':
##
##     print.default
```

```
# bc_model <- boxcox(final_model, optimize = TRUE)
# bc_lambda <- bc_model$lambda
# bc_lambda
# plot(bc_model)
```

```
# add Box-Cox transform to data
# train_data <- train_data %>%
#   mutate(bc_R_moment_1 = ((R_moment_1^bc_lambda) - 1)/bc_lambda)
#
# hist(train_data$bc_R_moment_1)
```

Test Error

```
lm_preds <- predict(current_model, test)
#mean((test$ARR_DELAY - lm_preds)^2)
```

***when all of the (test$ARR_DELAY - lm_preds)^2 are added up we get NA so not sure what to do abt that

GAM MODEL

Initial Model

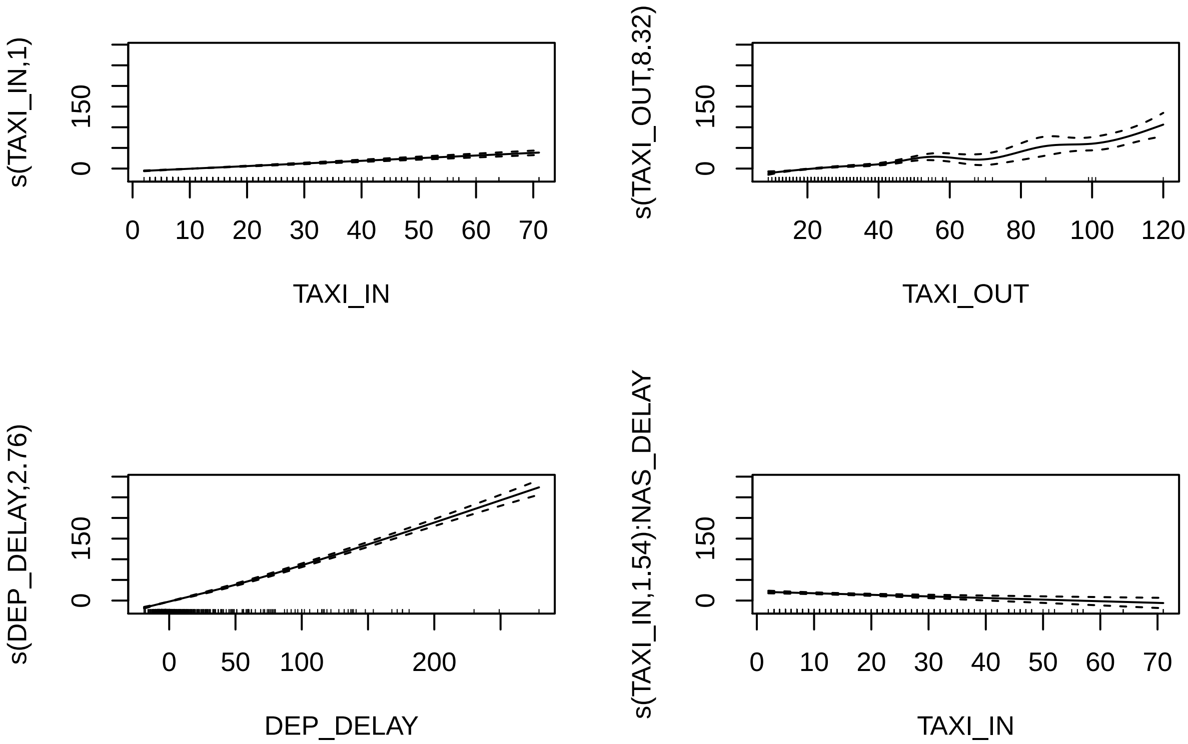
fit a gam model with numerical variables on a smoothing spline and including the interaction between NAS_DELAY and TAXI_IN

```
gam00 <- gam(ARR_DELAY ~ DAY_OF_MONTH +
                   s(TAXI_IN) +
                   s(TAXI_OUT) +
                   DEST +
                   s(DEP_DELAY) +
                   CARRIER_DELAY +
                   NAS_DELAY +
                   s(TAXI_IN, by = NAS_DELAY), data = flights)
```

```r
summary(gam00)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## ARR_DELAY ~ DAY_OF_MONTH + s(TAXI_IN) + s(TAXI_OUT) + DEST +
##     s(DEP_DELAY) + CARRIER_DELAY + NAS_DELAY + s(TAXI_IN, by = NAS_DELAY)
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.98256    0.80542   2.462   0.0139 *
## DAY_OF_MONTH  -1.33450    0.03996 -33.394   <2e-16 ***
## DESTSFO       -0.09686    0.74742  -0.130   0.8969
## CARRIER_DELAY  4.00763    2.09712   1.911   0.0562 .
## NAS_DELAY     17.66208    1.04364  16.924   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                        edf Ref.df      F p-value
## s(TAXI_IN)           1.000  1.000 192.04  <2e-16 ***
## s(TAXI_OUT)          8.319  8.812  40.18  <2e-16 ***
## s(DEP_DELAY)         2.756  3.454 826.39  <2e-16 ***
## s(TAXI_IN):NAS_DELAY 1.539  1.576 104.88  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 41/42
## R-sq.(adj) =  0.842   Deviance explained = 84.4%
## GCV = 201.01  Scale est. = 198.79    n = 1638
```

```r
par(mfrow = c(2,2))
plot.gam(gam00, se=TRUE)
```



Checking Lineartiy

TAXI_IN and the interaction between NAS_DELAY and TAXI_IN may be linear

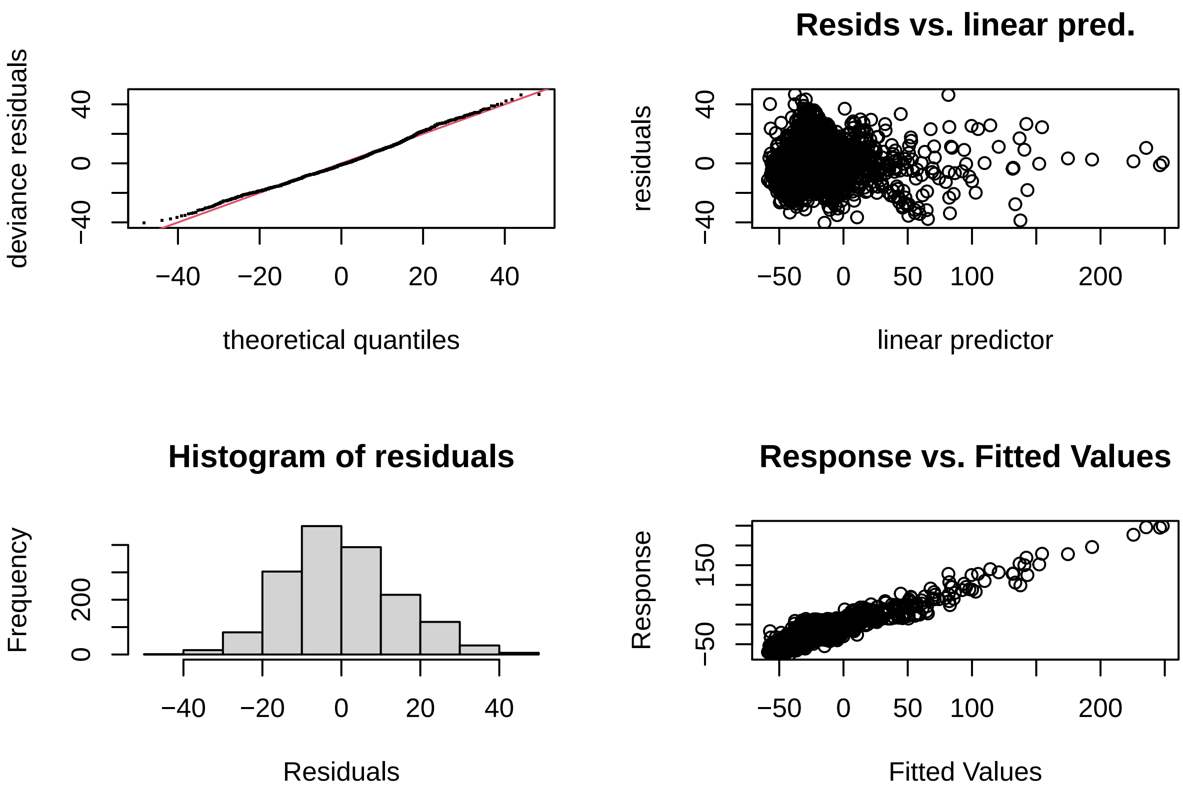
```
gam01 <- gam(ARR_DELAY ~ DAY_OF_MONTH +
                   TAXI_IN +
                   s(TAXI_OUT) +
                   DEST +
                   s(DEP_DELAY) +
                   CARRIER_DELAY +
                   NAS_DELAY +
                   TAXI_IN*NAS_DELAY, data = flights)

anova(gam00, gam01, test = "F")
```

```
## Analysis of Deviance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + s(TAXI_IN) + s(TAXI_OUT) + DEST +
##     s(DEP_DELAY) + CARRIER_DELAY + NAS_DELAY + s(TAXI_IN, by = NAS_DELAY)
## Model 2: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + s(TAXI_OUT) + DEST + s(DEP_DELAY) +
##     CARRIER_DELAY + NAS_DELAY + TAXI_IN * NAS_DELAY
##   Resid. Df Resid. Dev      Df Deviance      F  Pr(>F)
## 1    1618.7     322010
## 2    1618.5     321883 0.15046   127.57 4.2659 0.06531 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

based on anova test, the model with smoothing splines on TAXI_IN and the interaction term is a better fit

Model Diagnostics

```r
par(mfrow = c(2,2))
gam.check(gam00)
```




```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 14 iterations.
## The RMS GCV score gradient at convergence was 1.188084e-05 .
## The Hessian was positive definite.
## Model rank =  41 / 42
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                          k'   edf k-index p-value
## s(TAXI_IN)             9.00  1.00    0.82  <2e-16 ***
## s(TAXI_OUT)            9.00  8.32    0.90  <2e-16 ***
## s(DEP_DELAY)           9.00  2.76    0.86  <2e-16 ***
## s(TAXI_IN):NAS_DELAY 10.00  1.54    0.82  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Test Error

```r
gam_preds <- predict.gam(gam00, newdata = test)
#mean((test$ARR_DELAY - gam_preds)^2)
```