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)
```

```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
```

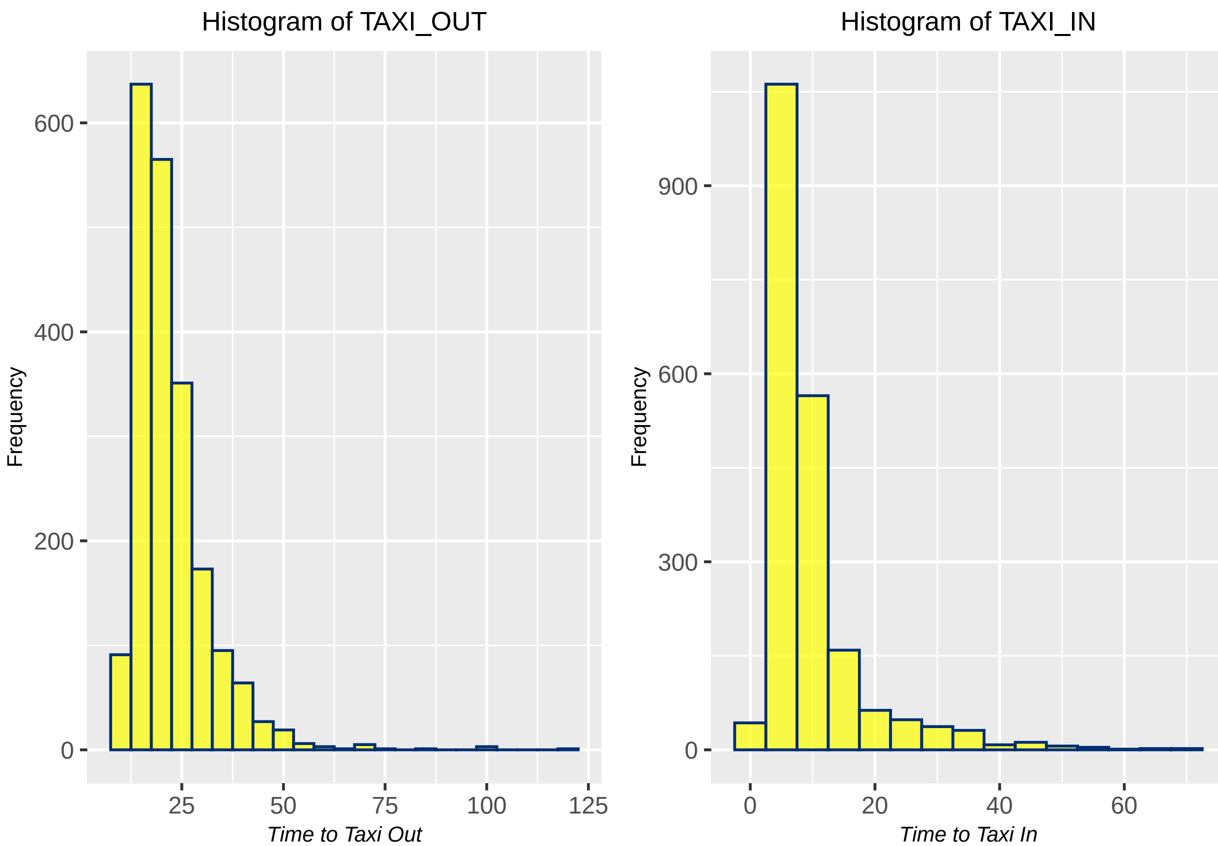
```
## # A tibble: 2,044 x 34
##     YEAR MONTH DAY_OF_MONTH DAY_OF_WEEK FL_DATE    OP_CARRIER TAIL_NUM
##    <dbl> <dbl>        <dbl>       <dbl> <date>     <chr>      <chr>
##  1  2020     1            1           3 2020-01-01 AA         N110AN
##  2  2020     1            2           4 2020-01-02 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

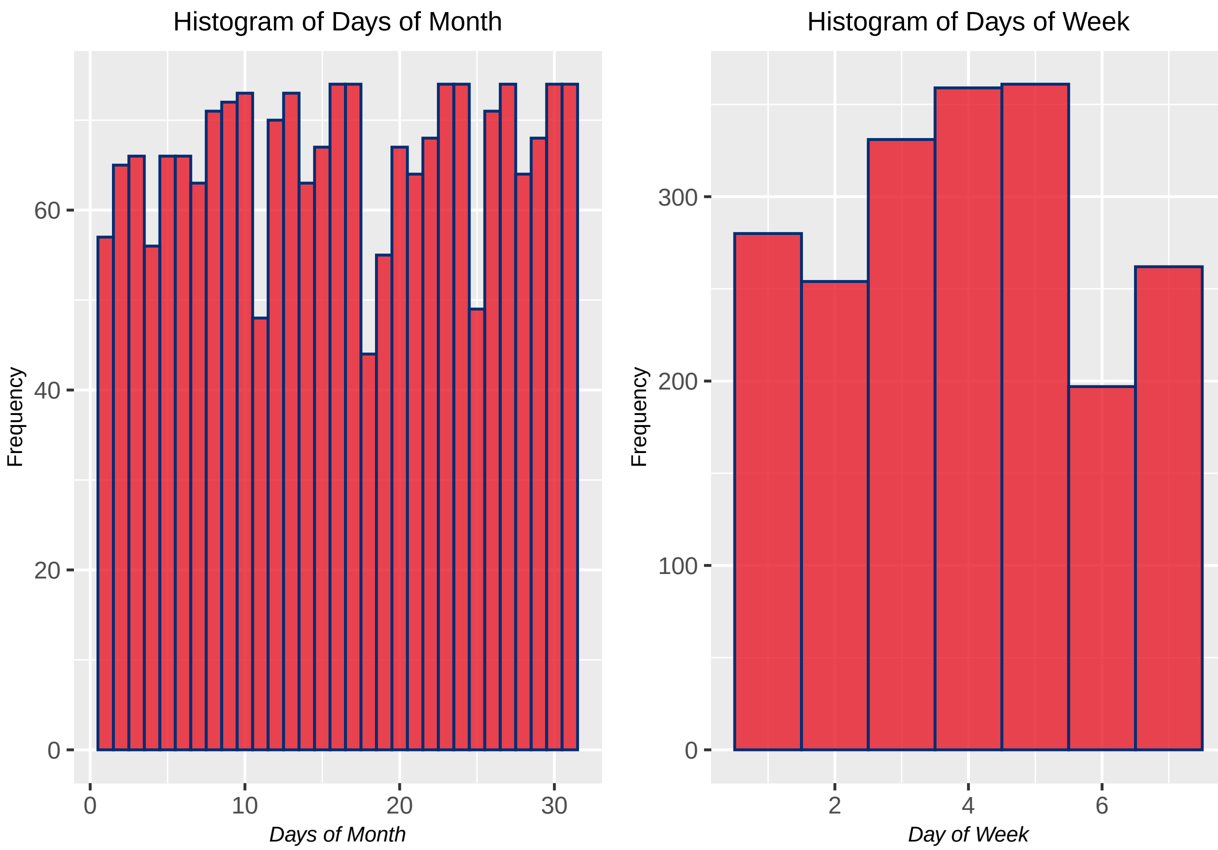
Taxi Histograms

```r
p00 <- 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))


p01 <- 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))


grid.arrange(p01, p00, nrow = 1)
```



Days of Month and Week

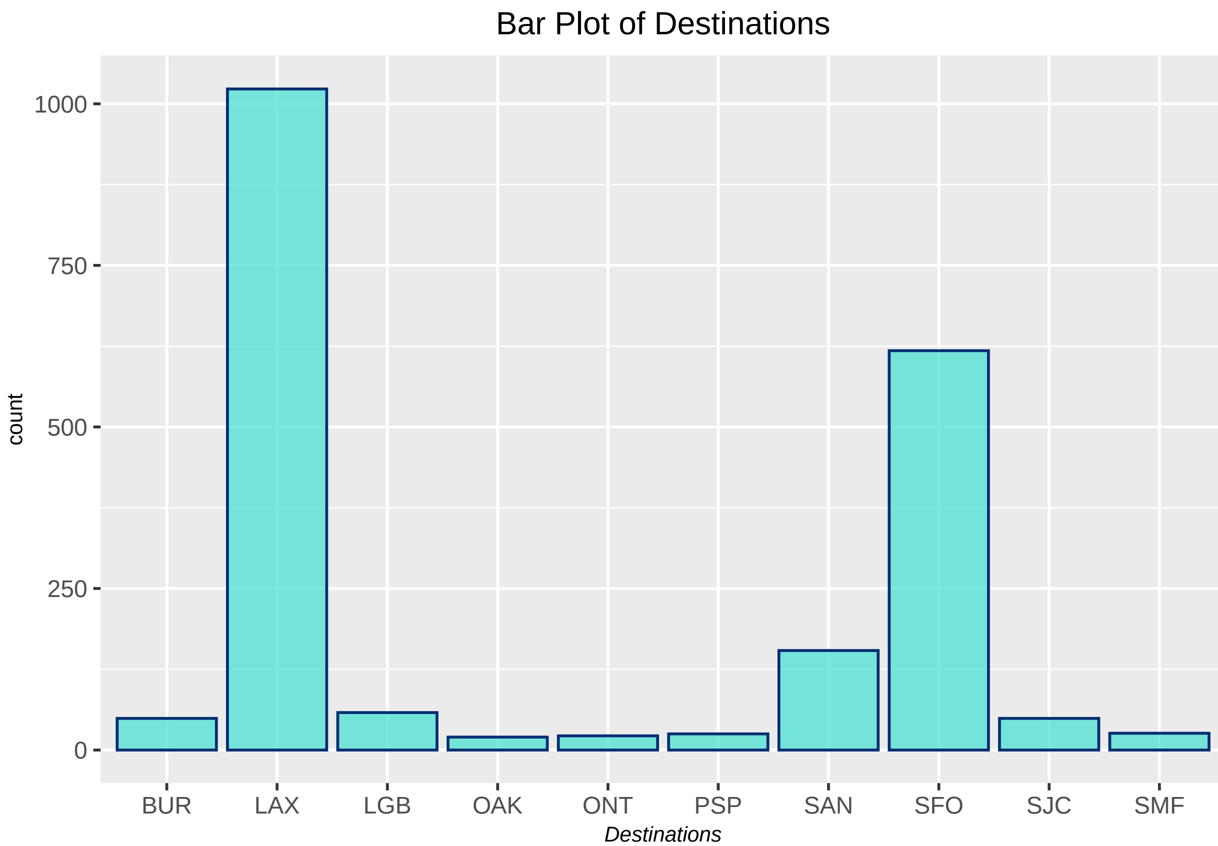
```r
p02 <- ggplot(data = flights, aes(x = DAY_OF_MONTH)) +
  geom_histogram(binwidth = 1, fill = "#E81828", color = "#002D72", alpha = .8) +
  labs(x = "Days of Month",
       y = "Frequency",
       title = "Histogram of Days of Month") +
    theme(plot.title = element_text(size = 10,hjust = 0.5),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

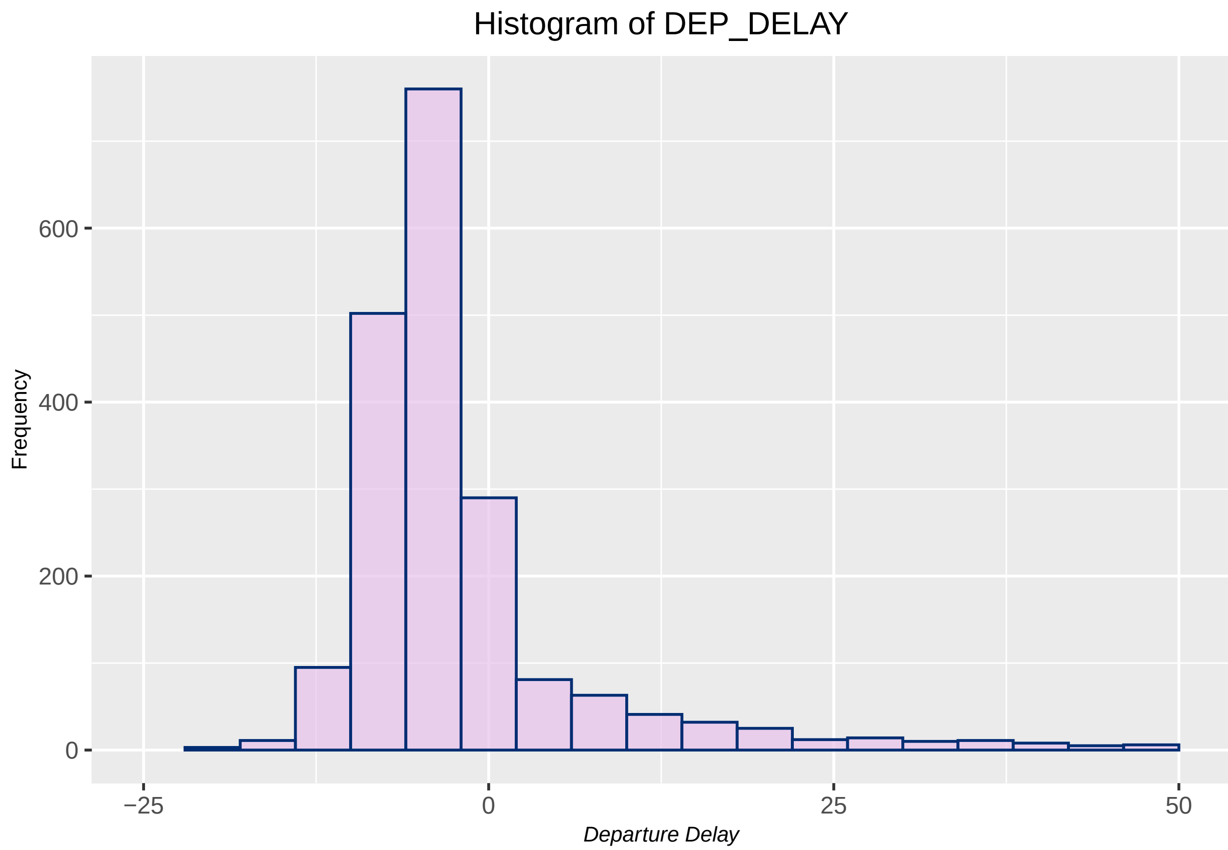
Origin is all JFK, but we could consider the different destination locations.

```
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))
```

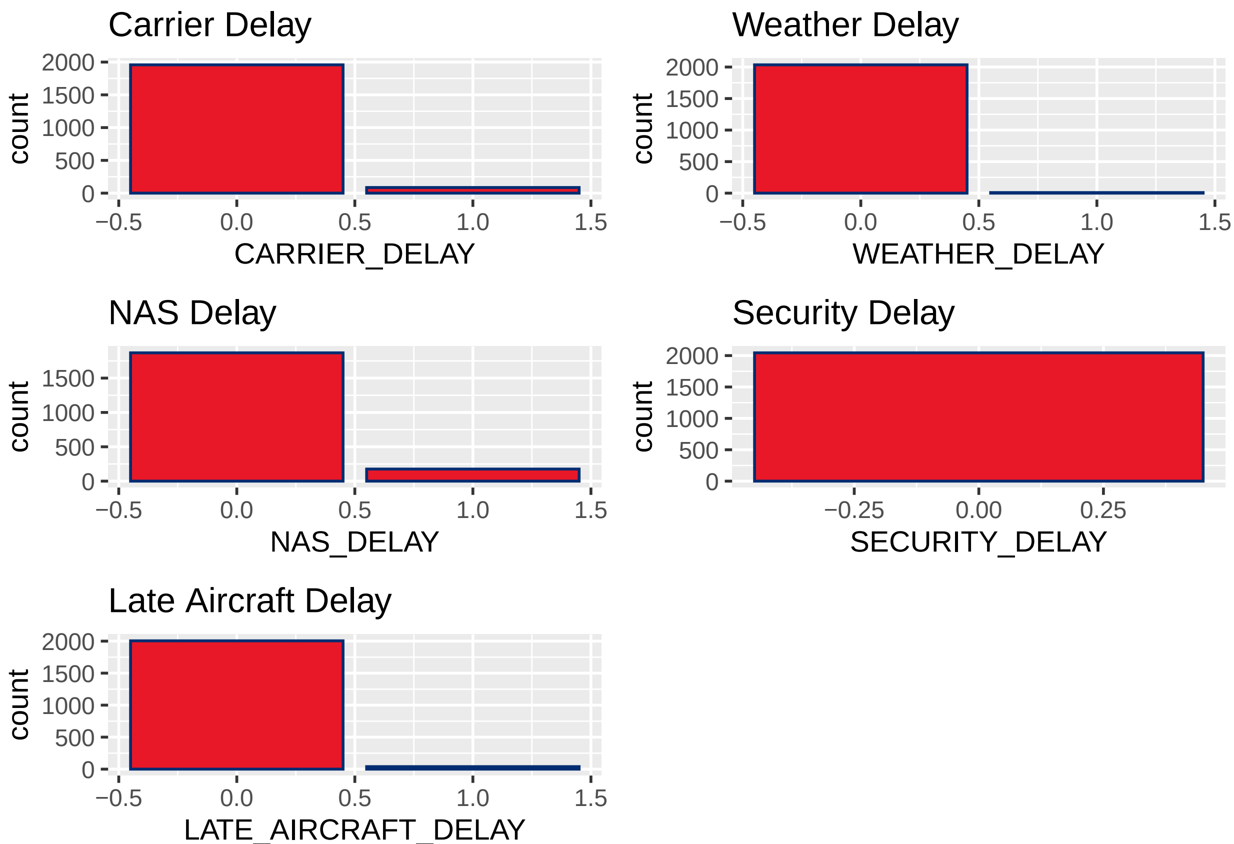


Depart Delay Histogram

```
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.

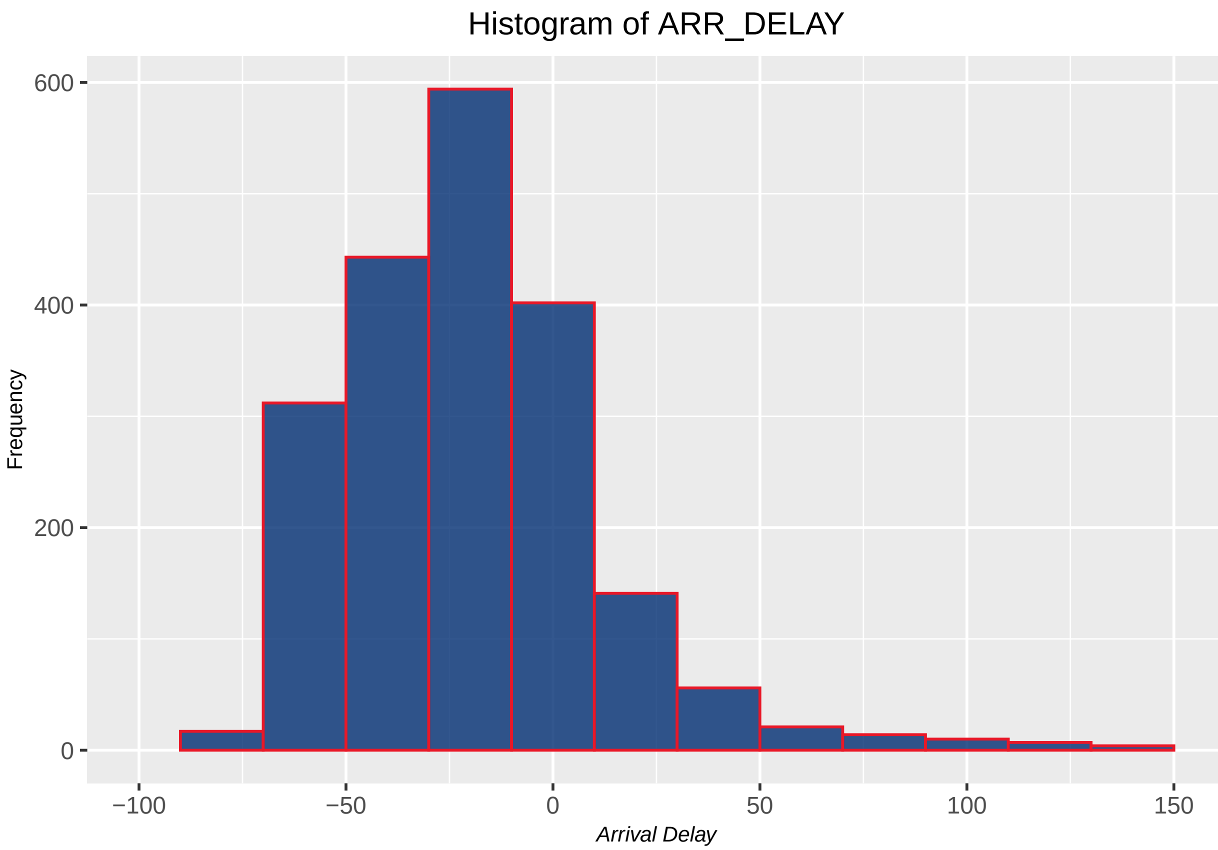
Overall, the categorical delay predictors I would think we could use are: Carrier Delay, NAS Delay, and Late Aircraft Delay

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") +
```

```
  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))
```

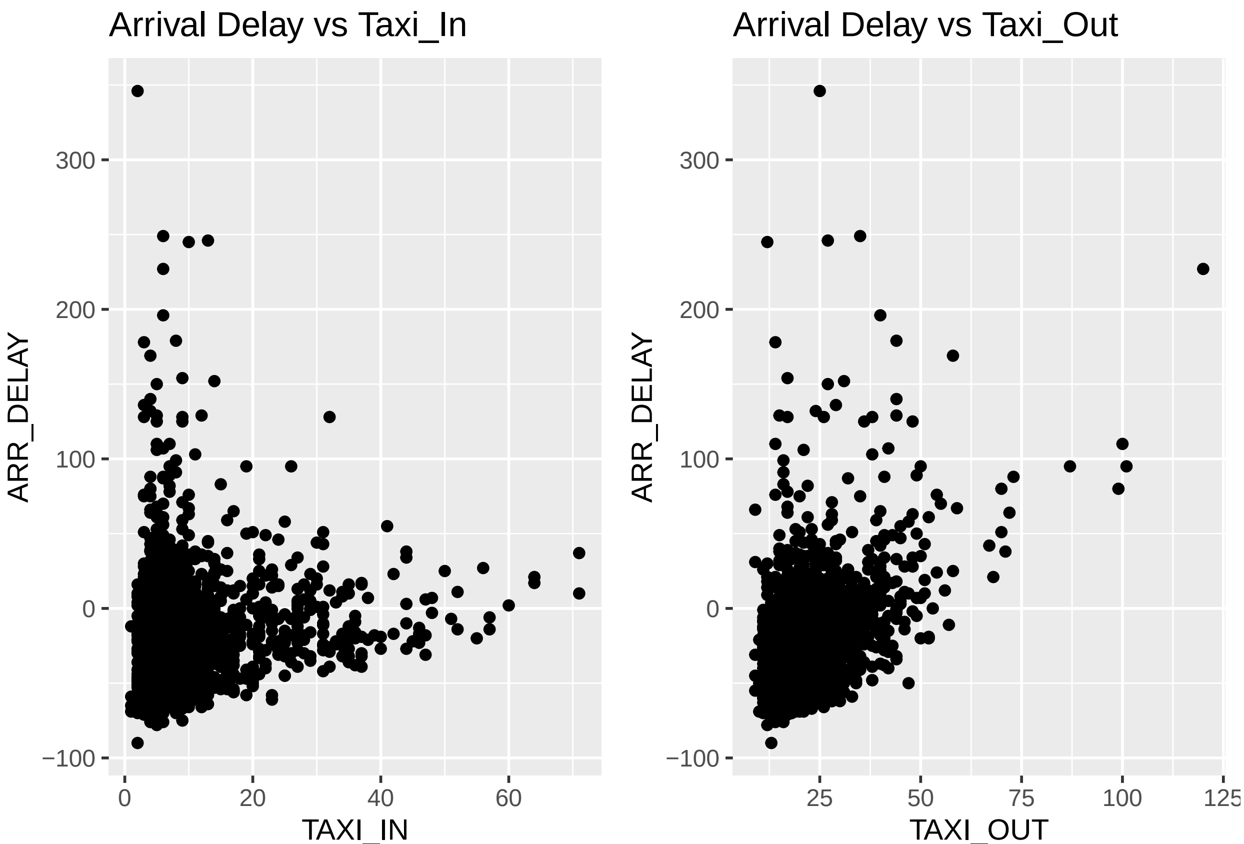


PREDICTORS VS RESPONSE

ARR_DELAY and TAXI_IN / TAXI_OUT

```
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)
```

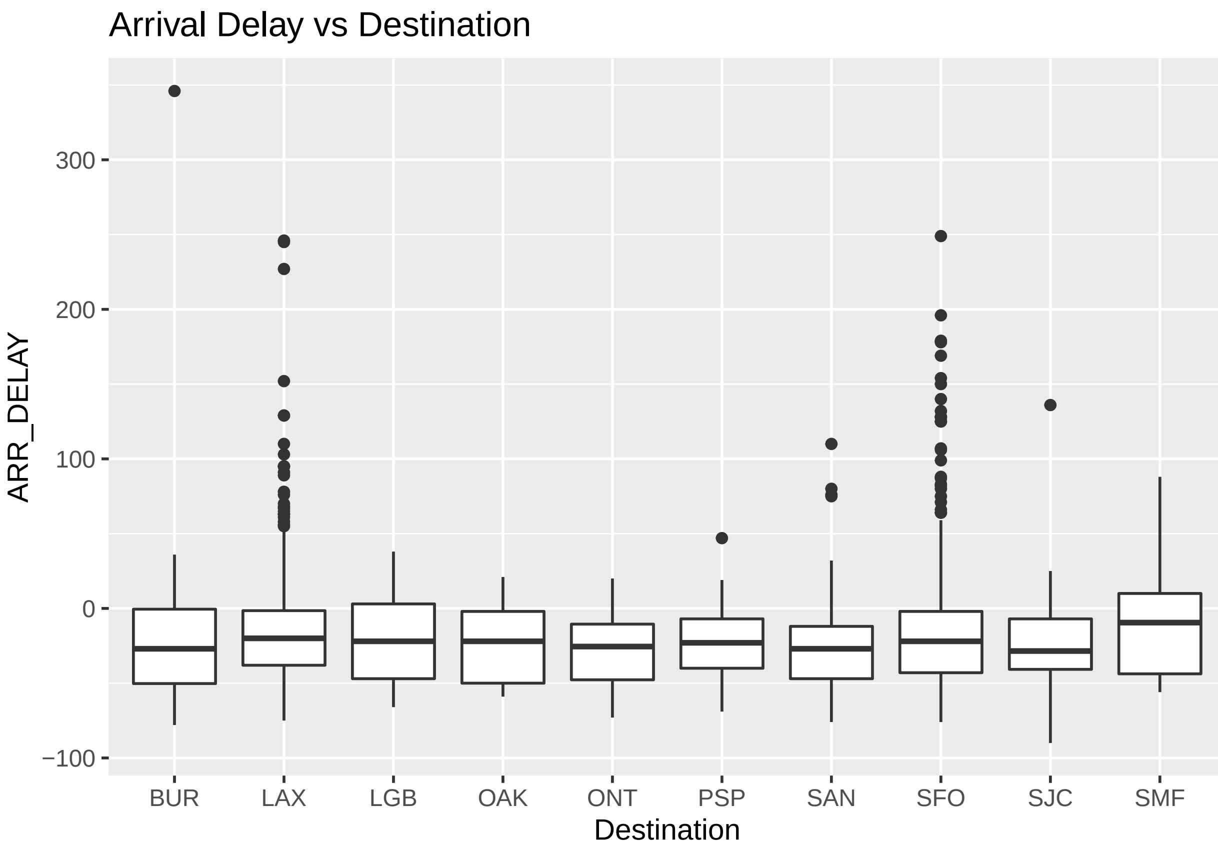
```
## Warning: Removed 11 rows containing missing values (geom_point).

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



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).
```

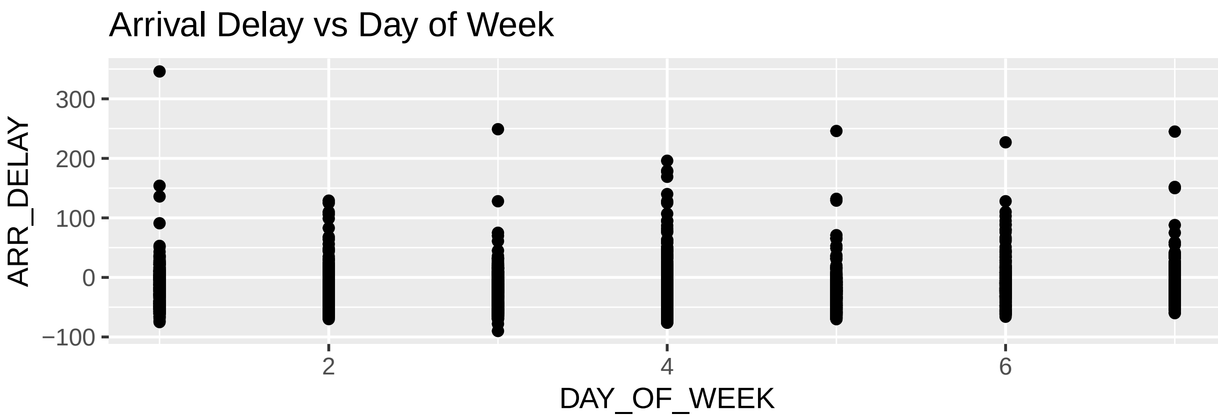


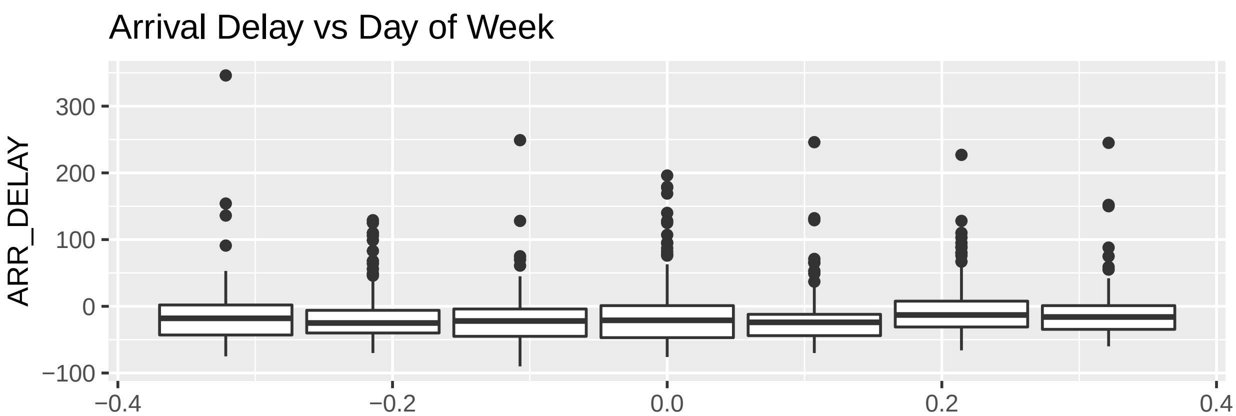
ARR_DELAY and DAY_OF_WEEK

```
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).
```



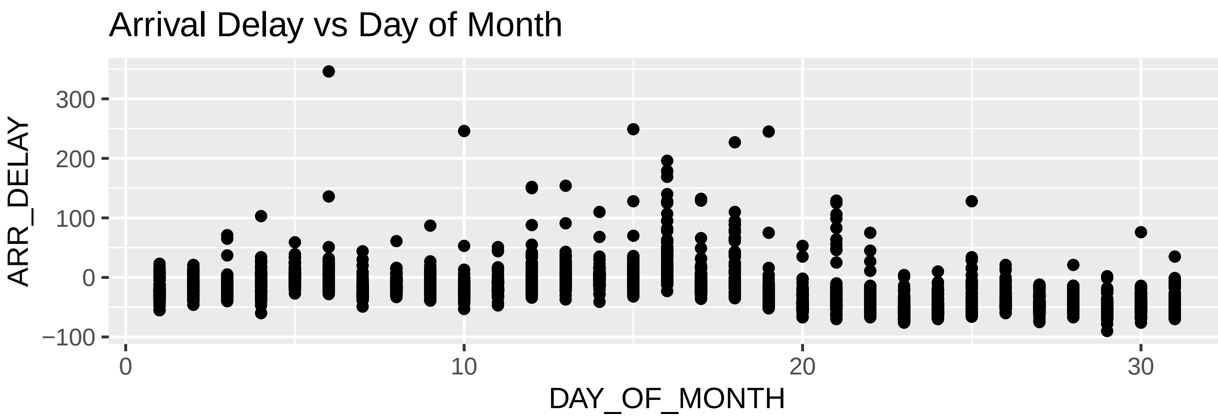


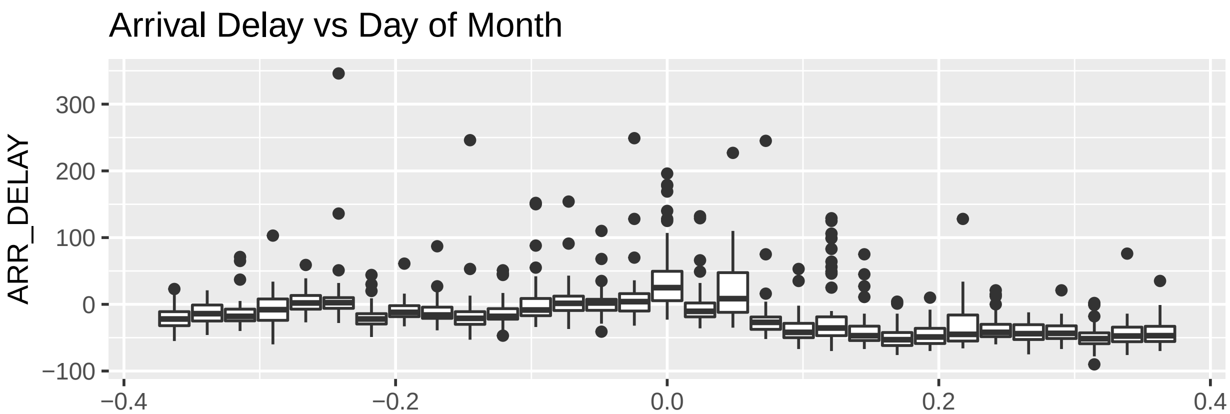
ARR_DELAY and DAY_OF_MONTH

```
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).
```





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

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

```
full_model <- lm(ARR_DELAY ~ DAY_OF_MONTH +
                   DAY_OF_WEEK +
                   TAXI_IN +
                   TAXI_OUT +
                   DEST +
                   DEP_DELAY +
                   CARRIER_DELAY +
                   NAS_DELAY +
```

```r
                   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
## -44.970 -10.430  -1.266   9.387  45.226
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -24.24577    2.63925  -9.187   <2e-16 ***
## DAY_OF_MONTH         -1.32356    0.04011 -32.999   <2e-16 ***
## DAY_OF_WEEK          -0.21541    0.19217  -1.121   0.2625
## TAXI_IN               0.57861    0.04587  12.615   <2e-16 ***
## TAXI_OUT              0.77139    0.04245  18.171   <2e-16 ***
## DESTLAX               1.20551    2.32706   0.518   0.6045
## DESTLGB               2.97956    3.05864   0.974   0.3301
## DESTOAK               1.86463    4.15748   0.448   0.6539
## DESTONT              -4.52792    4.07365  -1.112   0.2665
## DESTPSP              -0.31847    3.86654  -0.082   0.9344
## DESTSAN              -2.46808    2.60663  -0.947   0.3439
## DESTSFO               0.79911    2.34731   0.340   0.7336
## DESTSJC              -7.12166    3.43070  -2.076   0.0381 *
## DESTSMF               6.58721    4.00370   1.645   0.1001
## DEP_DELAY             0.91942    0.01821  50.496   <2e-16 ***
## CARRIER_DELAY         4.73917    2.09069   2.267   0.0235 *
## NAS_DELAY            32.50028    1.45516  22.335   <2e-16 ***
## LATE_AIRCRAFT_DELAY  -2.74307    2.97881  -0.921   0.3573
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.39 on 1607 degrees of freedom
##   (10 observations deleted due to missingness)
## Multiple R-squared:  0.8404, Adjusted R-squared:  0.8387
## F-statistic: 497.6 on 17 and 1607 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 +
##     DEST + DEP_DELAY + CARRIER_DELAY + NAS_DELAY, data = train)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -47.787 -10.321  -1.338   9.256  45.408
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -25.01721    2.53099  -9.884   <2e-16 ***
## DAY_OF_MONTH   -1.32297    0.04008 -33.009   <2e-16 ***
## TAXI_IN         0.57627    0.04584  12.572   <2e-16 ***
## TAXI_OUT        0.76822    0.04236  18.134   <2e-16 ***
## DESTLAX         1.19063    2.32489   0.512   0.6086
## DESTLGB         2.96546    3.05813   0.970   0.3323
## DESTOAK         1.75968    4.15706   0.423   0.6721
## DESTONT        -4.65443    4.07297  -1.143   0.2533
## DESTPSP        -0.45469    3.86545  -0.118   0.9064
## DESTSAN        -2.49858    2.60422  -0.959   0.3375
## DESTSFO         0.80759    2.34558   0.344   0.7307
## DESTSJC        -7.11782    3.43083  -2.075   0.0382 *
## DESTSMF         6.43808    4.00106   1.609   0.1078
## DEP_DELAY       0.91300    0.01677  54.458   <2e-16 ***
## CARRIER_DELAY   4.73531    2.09024   2.265   0.0236 *
## NAS_DELAY      32.40451    1.45106  22.332   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.39 on 1609 degrees of freedom
##   (10 observations deleted due to missingness)
## Multiple R-squared:  0.8401, Adjusted R-squared:  0.8387
## F-statistic: 563.8 on 15 and 1609 DF,  p-value: < 2.2e-16
```

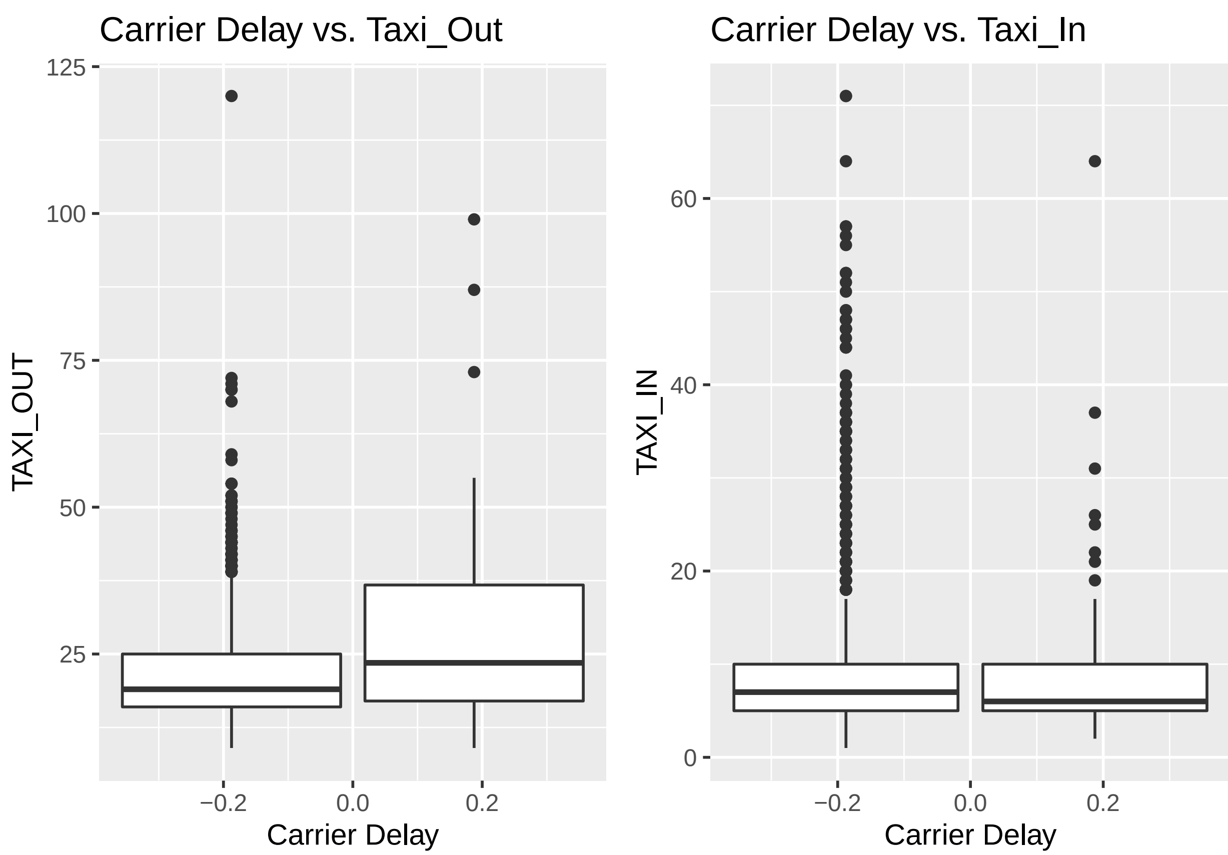
The only variables that were removed were DAY_OF_WEEK and LATE_AIRCRAFT_DELAY. Let's continue using the step_model then.

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.

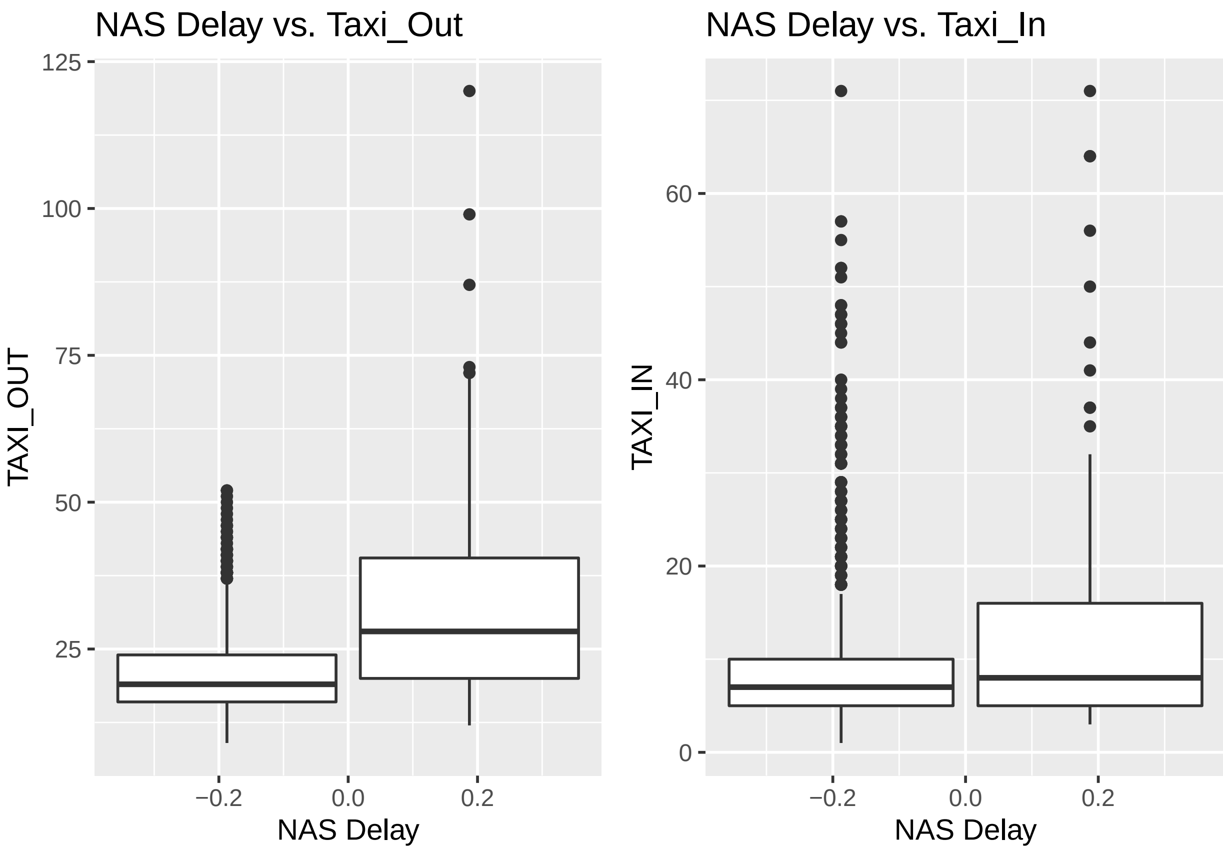
```
p12 <- ggplot(data = train, aes(group = CARRIER_DELAY, y = TAXI_OUT)) +
  geom_boxplot() +
  labs(title = "Carrier Delay vs. Taxi_Out",
       x = "Carrier Delay")

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)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).

## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```




```r
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)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).

## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```



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 + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_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   1609 333202
## 2   1608 333015  1    186.44 0.9002 0.3429
```

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

```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_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   1609 333202
## 2   1608 333108  1    93.461 0.4512 0.5019
```

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

```
## Analysis of Variance Table
##
## Model 1: ARR_DELAY ~ DAY_OF_MONTH + TAXI_IN + TAXI_OUT + DEST + DEP_DELAY +
##     CARRIER_DELAY + NAS_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   1609 333202
## 2   1608 330298  1    2904.1 14.138 0.000176 ***
## ---
## 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:

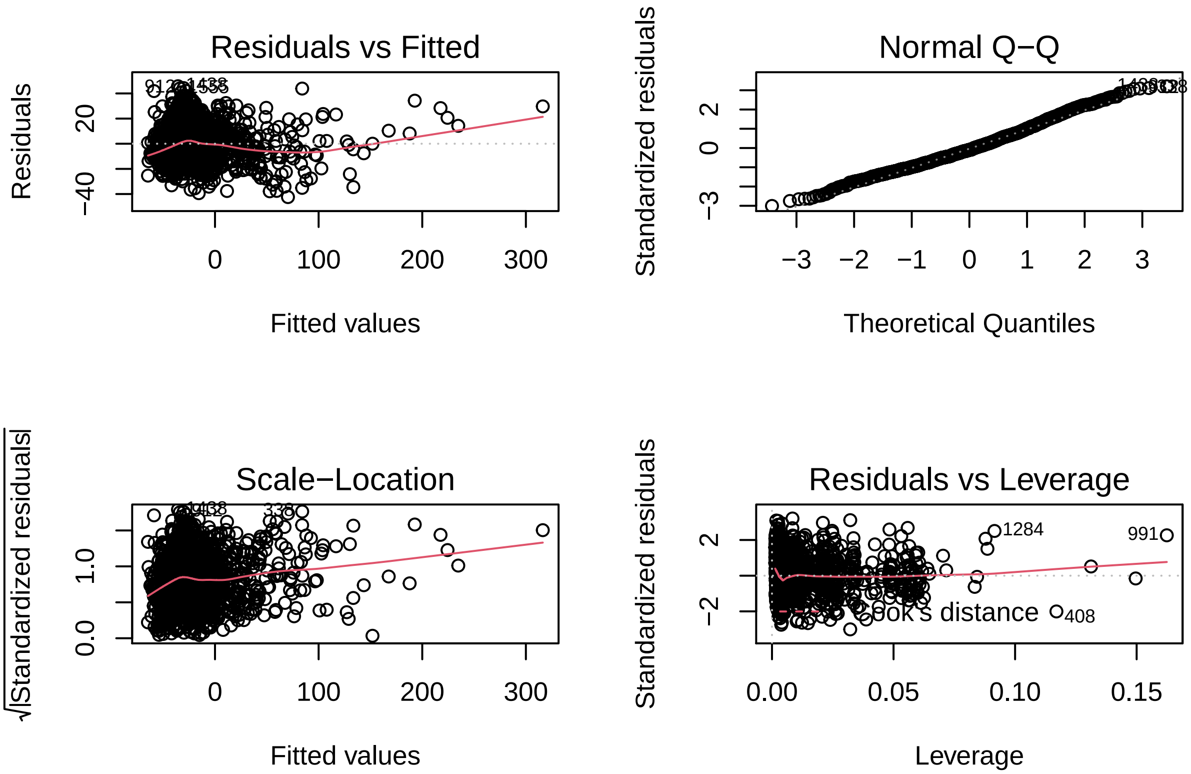
Final Linear Model

```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
## -42.417 -10.143  -1.367   9.125  45.718
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -25.09770    2.52082  -9.956  < 2e-16 ***
## DAY_OF_MONTH       -1.33461    0.04004 -33.335  < 2e-16 ***
## TAXI_IN             0.66008    0.05080  12.993  < 2e-16 ***
## TAXI_OUT            0.75919    0.04226  17.965  < 2e-16 ***
## DESTLAX             0.86503    2.31708   0.373 0.708953
## DESTLGB             2.60523    3.04723   0.855 0.392705
## DESTOAK             1.59986    4.14041   0.386 0.699251
## DESTONT            -4.56546    4.05651  -1.125 0.260560
## DESTPSP            -0.52212    3.84981  -0.136 0.892137
## DESTSAN            -2.56349    2.59371  -0.988 0.323132
## DESTSFO             0.42116    2.33833   0.180 0.857087
## DESTSJC            -7.50347    3.41844  -2.195 0.028306 *
## DESTSMF             5.97867    3.98670   1.500 0.133900
## DEP_DELAY           0.90830    0.01674  54.246  < 2e-16 ***
## CARRIER_DELAY       4.86486    2.08205   2.337 0.019584 *
## NAS_DELAY          37.53106    1.98682  18.890  < 2e-16 ***
## TAXI_IN:NAS_DELAY  -0.39716    0.10563  -3.760 0.000176 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.33 on 1608 degrees of freedom
##   (10 observations deleted due to missingness)
## Multiple R-squared:  0.8415, Adjusted R-squared:   0.84
## F-statistic: 533.7 on 16 and 1608 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.

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

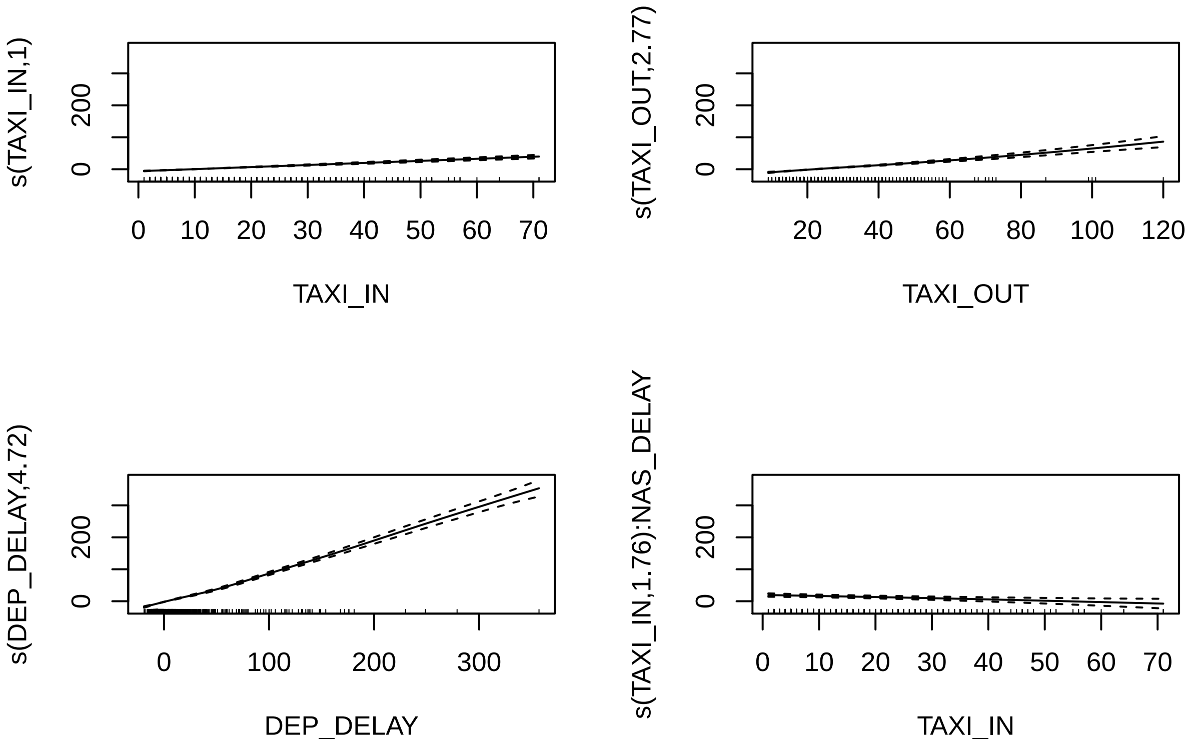
fit a gam model with numerical variables on a smoothing spline and including the interaction between NAS_DELAY and TAXI_IN

```
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)

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)    0.45836    2.17068   0.211   0.8328
## DAY_OF_MONTH  -1.35659    0.03584 -37.849   <2e-16 ***
## DESTLAX        1.03218    2.12413   0.486   0.6271
## DESTLGB        3.07292    2.78800   1.102   0.2705
## DESTOAK        1.01916    3.84584   0.265   0.7910
## DESTONT       -2.20746    3.65232  -0.604   0.5456
## DESTPSP       -2.39529    3.50810  -0.683   0.4948
## DESTSAN       -1.29401    2.35685  -0.549   0.5830
## DESTSFO        1.00012    2.14412   0.466   0.6409
## DESTSJC       -6.54423    2.97443  -2.200   0.0279 *
## DESTSMF        6.10605    3.46017   1.765   0.0778 .
## CARRIER_DELAY  4.39089    1.87974   2.336   0.0196 *
## NAS_DELAY     18.01532    2.10636   8.553   <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  1.000 208.56  < 2e-16 ***
## s(TAXI_OUT)           2.766  3.475 123.65  < 2e-16 ***
## s(DEP_DELAY)          4.719  5.741 646.27  < 2e-16 ***
## s(TAXI_IN):NAS_DELAY 1.762  1.977  37.48 5.73e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 49/50
## R-sq.(adj) =  0.842   Deviance explained = 84.4%
## GCV = 201.99  Scale est. = 199.73    n = 2033
```

```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

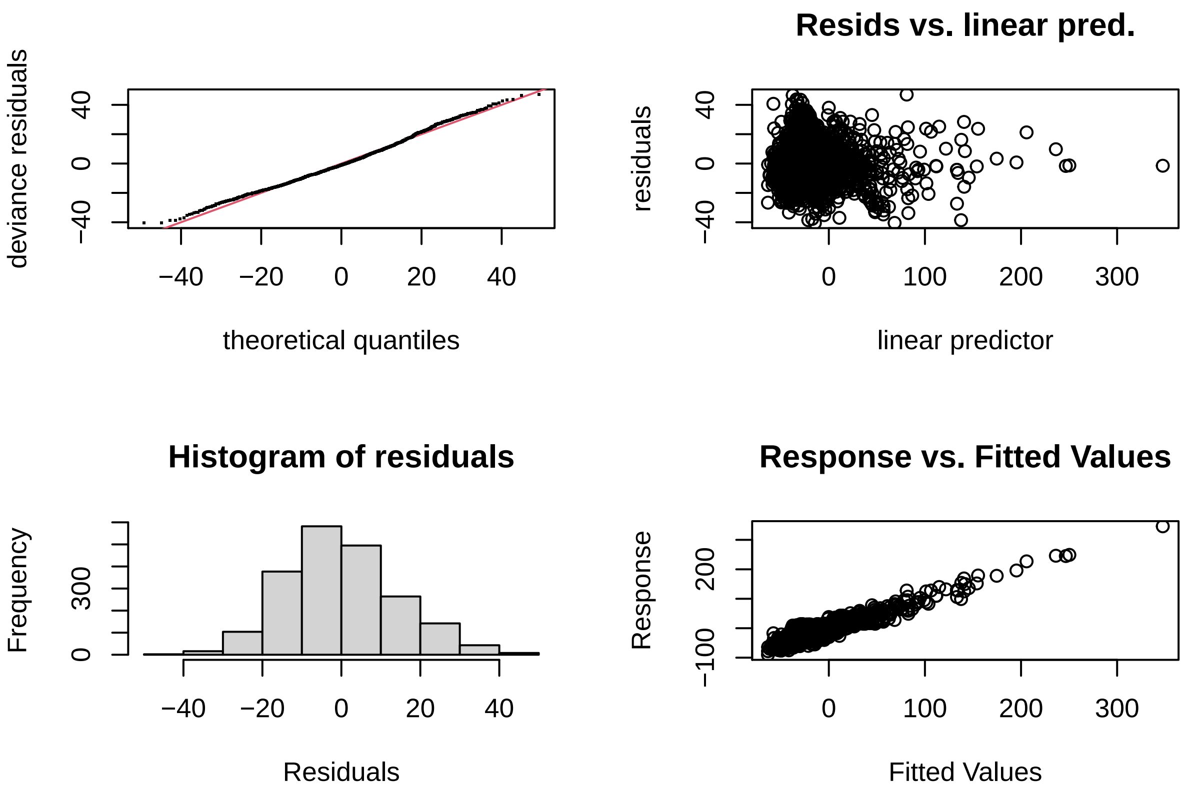
```
gam01 <- gam(ARR_DELAY ~ DAY_OF_MONTH +
                   TAXI_IN +
                   s(TAXI_OUT) +
                   DEST +
                   s(DEP_DELAY) +
                   CARRIER_DELAY +
                   NAS_DELAY +
                   TAXI_IN*NAS_DELAY, data = flights)

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    2008.3     401501
## 2    2008.9     401648 -0.55356   -147.5 1.3341 0.2189
```

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

Model Diagnostics

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




```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 12 iterations.
## The RMS GCV score gradient at convergence was 8.548148e-06 .
## The Hessian was positive definite.
## Model rank =  49 / 50
##
## 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    0.83  <2e-16 ***
## s(TAXI_OUT)            9.00  2.77    0.90  <2e-16 ***
## s(DEP_DELAY)           9.00  4.72    0.86  <2e-16 ***
## s(TAXI_IN):NAS_DELAY 10.00  1.76    0.83  <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)
```