Lab2

Heleine Fouda

2023-09-17

Introduction to data

The Data

The data are from the Bureau of Transportation Statistics (BTS), an agency that is part of the Research and Innovative Technology Administration (RITA)

```
library(tidyverse)
library(openintro)
library(DATA606)

##
## Welcome to CUNY DATA606 Statistics and Probability for Data Analytics
## This package is designed to support this course. The text book used
## is OpenIntro Statistics, 4th Edition. You can read this by typing
## vignette('os4') or visit www.OpenIntro.org.
##
## The getLabs() function will return a list of the labs available.
##
## The demo(package='DATA606') will list the demos that are available.

library(ggplot2)

data("nycflights")

names(nycflights)

##  [1] "year"      "month"     "day"       "dep_time"  "dep_delay" "arr_time
"
##  [7] "arr_delay" "carrier"   "tailnum"   "flight"    "origin"    "dest"
## [13] "air_time"  "distance"  "hour"      "minute"

?nycflights

glimpse(nycflights)

## Rows: 32,735
## Columns: 16
## $ year      <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 20
13, …
## $ month     <int> 6, 5, 12, 5, 7, 1, 12, 8, 9, 4, 6, 11, 4, 3, 10, 1, 2, 8
```

```
, 10…
## $ day       <int> 30, 7, 8, 14, 21, 1, 9, 13, 26, 30, 17, 22, 26, 25, 21,
23, …
## $ dep_time  <int> 940, 1657, 859, 1841, 1102, 1817, 1259, 1920, 725, 1323,
940…
## $ dep_delay <dbl> 15, -3, -1, -4, -3, -3, 14, 85, -10, 62, 5, 5, -2, 115,
-4, …
## $ arr_time  <int> 1216, 2104, 1238, 2122, 1230, 2008, 1617, 2032, 1027, 15
49, …
## $ arr_delay <dbl> -4, 10, 11, -34, -8, 3, 22, 71, -8, 60, -4, -2, 22, 91,
-6, …
## $ carrier   <chr> "VX", "DL", "DL", "DL", "9E", "AA", "WN", "B6", "AA", "E
V", …
## $ tailnum   <chr> "N626VA", "N3760C", "N712TW", "N914DL", "N823AY", "N3AXA
A", …
## $ flight    <int> 407, 329, 422, 2391, 3652, 353, 1428, 1407, 2279, 4162,
20, …
## $ origin    <chr> "JFK", "JFK", "JFK", "JFK", "LGA", "LGA", "EWR", "JFK",
"LGA…
## $ dest      <chr> "LAX", "SJU", "LAX", "TPA", "ORF", "ORD", "HOU", "IAD",
"MIA…
## $ air_time  <dbl> 313, 216, 376, 135, 50, 138, 240, 48, 148, 110, 50, 161,
87,…
## $ distance  <dbl> 2475, 1598, 2475, 1005, 296, 733, 1411, 228, 1096, 820,
264,…
## $ hour      <dbl> 9, 16, 8, 18, 11, 18, 12, 19, 7, 13, 9, 13, 8, 20, 12, 2
0, 6…
## $ minute    <dbl> 40, 57, 59, 41, 2, 17, 59, 20, 25, 23, 40, 20, 9, 54, 17
, 24…
```

Questions we might want to answer with these data

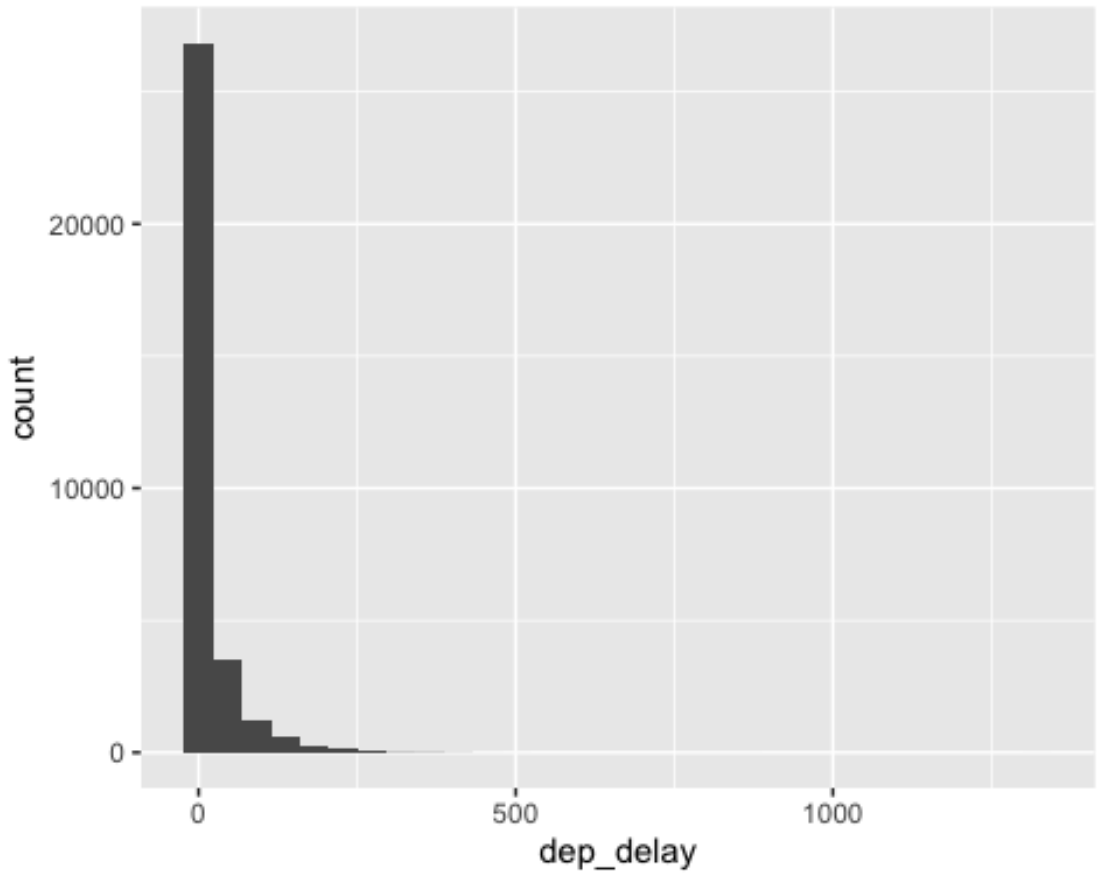
1. How delayed were flights that were headed to Los Angeles?
2. How do departure delays vary by month?
3. Which of the three major NYC airports has the bets on time percentage for departing flights?

Analysis

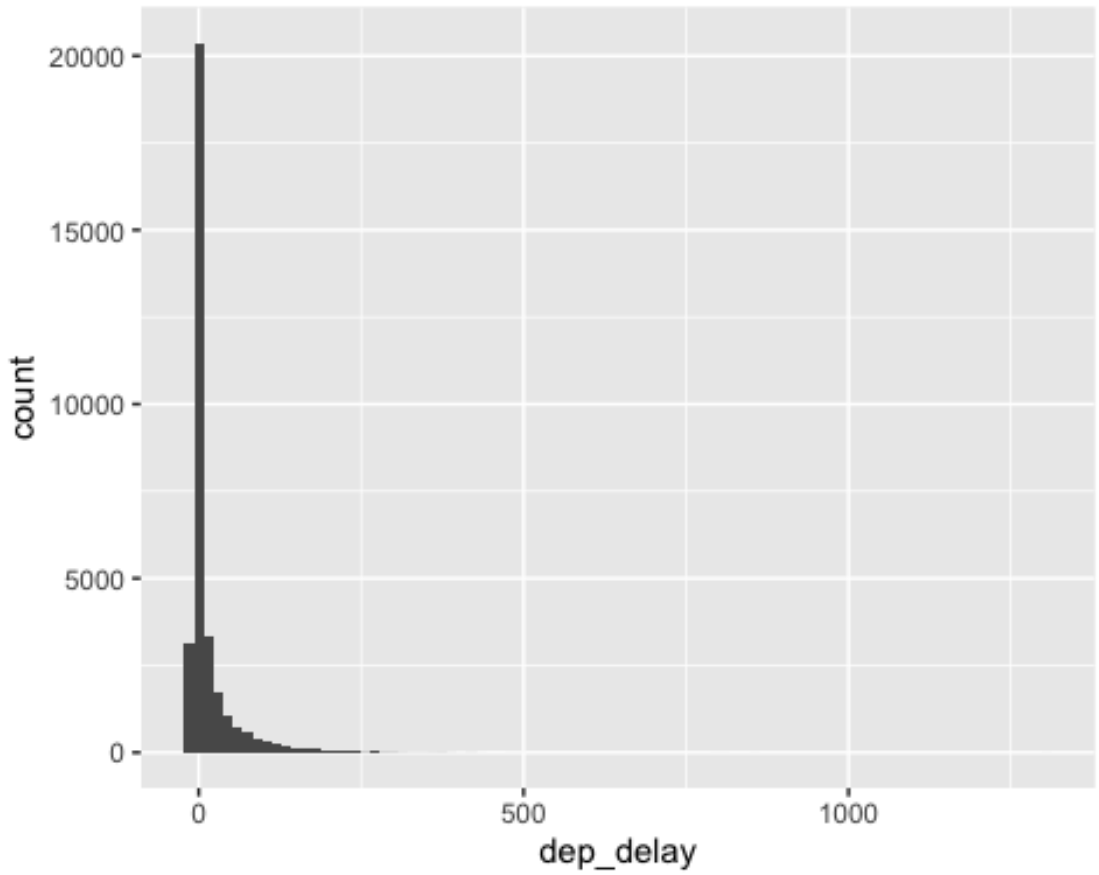
Let's start by examining the distribution of departure delays of all flights with a histogram

```
ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram()
```

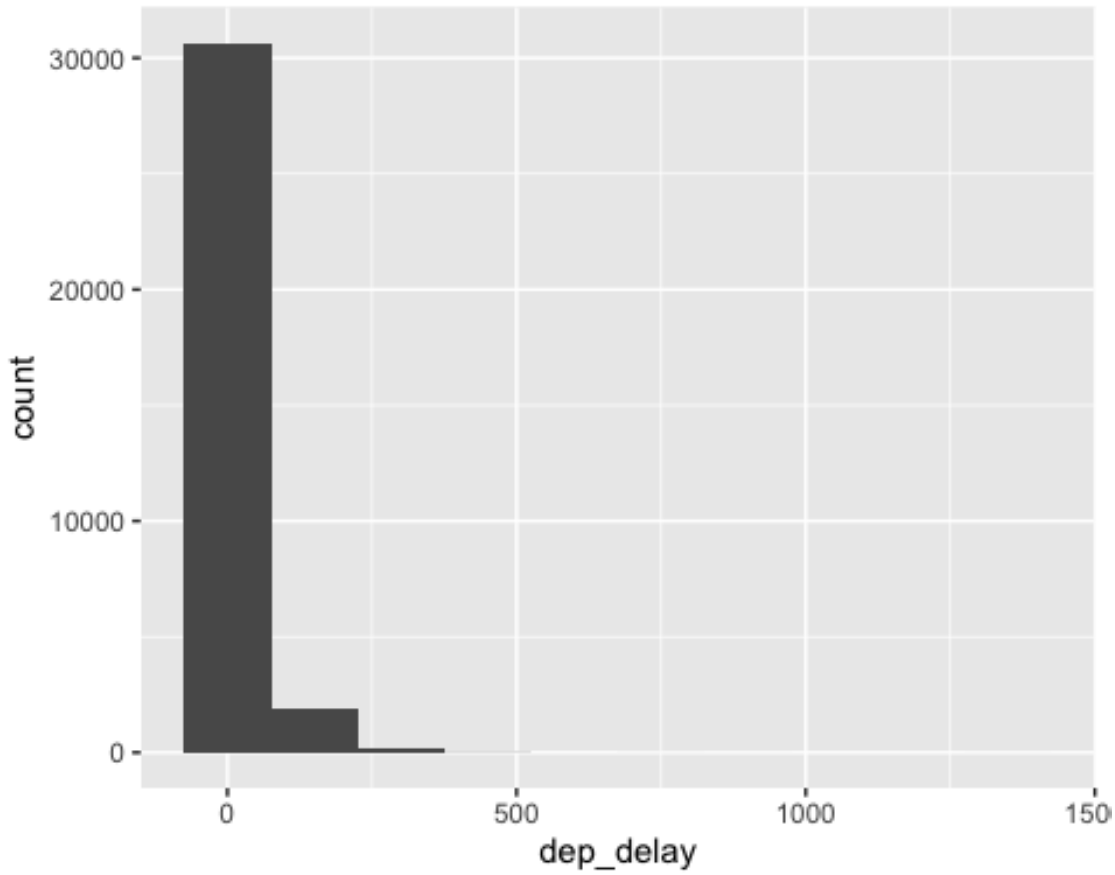
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram(binwidth = 15)
```

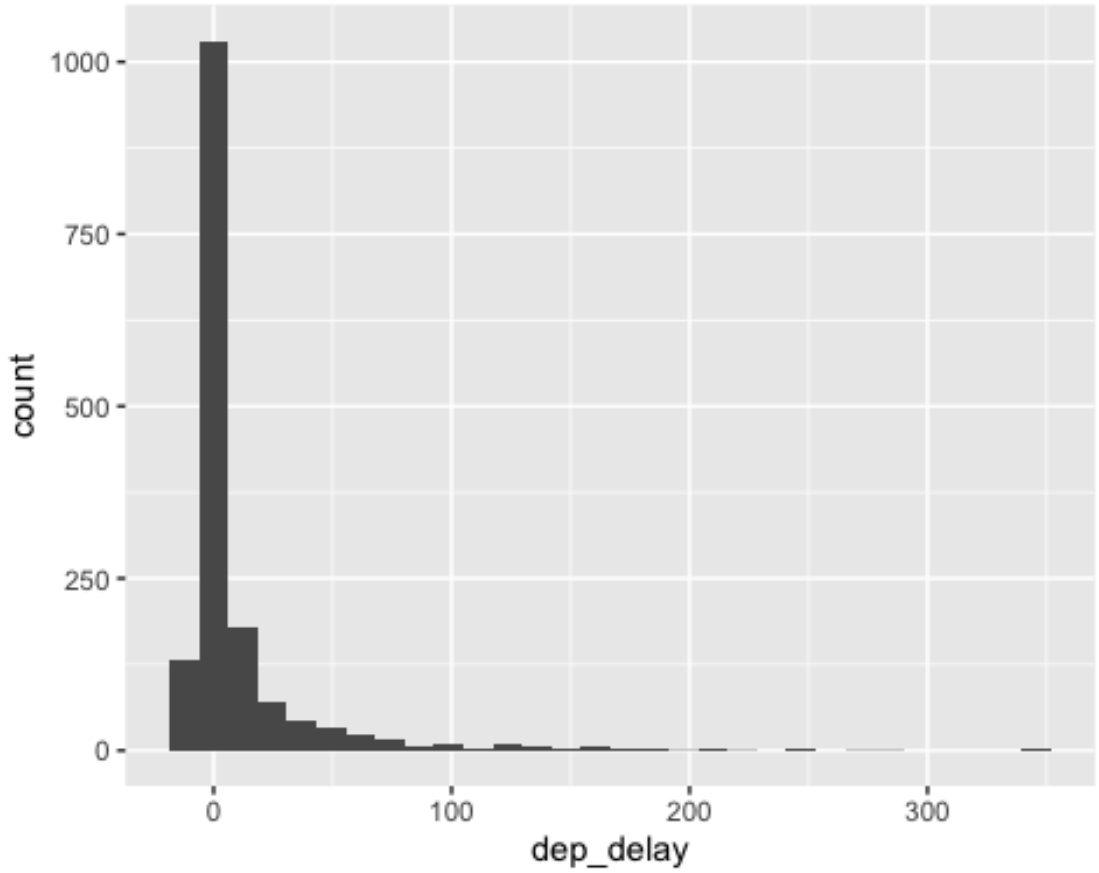



```
ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram(binwidth = 150)
```



To visualize only delays on flights headed to Los Angeles, you need to filter the data for flights with that destination (dest=="LAX") and then make a histogram of the departure delays of only those flights.

```
lax_flights <- nycflights %>%
  filter(dest =="LAX")
ggplot(data = lax_flights, aes(x = dep_delay))+
  geom_histogram()

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



Summary statistics:

You can also obtain numerical summaries for these flights:

```
lax_flights %>%
  summarise(mean_dd = mean(dep_delay),
            median_dd = median(dep_delay),
            n         = n())
```

```
## # A tibble: 1 × 3
##   mean_dd median_dd     n
##     <dbl>     <dbl> <int>
## 1    9.78        -1  1583
```

Filter based on multiple criteria

```
sfo_feb_flights <- nycflights %>%
  filter(dest == "SFO", month == 2)
glimpse(sfo_feb_flights)
```

```
## Rows: 68
## Columns: 16
## $ year      <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 20
13, …
## $ month     <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
```

```
2, …
## $ day       <int> 18, 3, 15, 18, 24, 25, 7, 15, 13, 8, 11, 13, 25, 20, 12,
27,…
## $ dep_time  <int> 1527, 613, 955, 1928, 1340, 1415, 1032, 1805, 1056, 656,
191…
## $ dep_delay <dbl> 57, 14, -5, 15, 2, -10, 1, 20, -4, -4, 40, -2, -1, -6, -
7, 2…
## $ arr_time  <int> 1903, 1008, 1313, 2239, 1644, 1737, 1352, 2122, 1412, 10
39, …
## $ arr_delay <dbl> 48, 38, -28, -6, -21, -13, -10, 2, -13, -6, 2, -5, -30,
-22,…
## $ carrier   <chr> "DL", "UA", "DL", "UA", "UA", "UA", "B6", "AA", "UA", "D
L", …
## $ tailnum   <chr> "N711ZX", "N502UA", "N717TW", "N24212", "N76269", "N532U
A", …
## $ flight    <int> 1322, 691, 1765, 1214, 1111, 394, 641, 177, 642, 1865, 2
72, …
## $ origin    <chr> "JFK", "JFK", "JFK", "EWR", "EWR", "JFK", "JFK", "JFK",
"JFK…
## $ dest      <chr> "SFO", "SFO", "SFO", "SFO", "SFO", "SFO", "SFO", "SFO",
"SFO…
## $ air_time  <dbl> 358, 367, 338, 353, 341, 355, 359, 338, 347, 361, 332, 3
51, …
## $ distance  <dbl> 2586, 2586, 2586, 2565, 2565, 2586, 2586, 2586, 2586, 25
86, …
## $ hour      <dbl> 15, 6, 9, 19, 13, 14, 10, 18, 10, 6, 19, 8, 10, 18, 7, 1
7, 1…
## $ minute    <dbl> 27, 13, 55, 28, 40, 15, 32, 5, 56, 56, 10, 33, 48, 49, 2
3, 2…
```

```
sfo_feb_flights <- nycflights %>%
  filter(dest == "SFO"| month == 2)
```

Exercise 1

Answer: The three histograms all provide one with a quite different picture of the distribution.

The one with a large bindwidth (150) hides some features of the distribution. This can prevent one from getting a sense of the true nature of the data distribution. On the other hand, while the histogram with a much smaller bindwidth (15) provides a very detailed look of the data distribution, it may prevent one from getting a higher perspective of a phenomenon and capturing its overall “big picture”.

Exercise 2: Create a new data frame that includes flights headed to SFO in February, and save this data frame as `sfo_feb_flights`. How many flights meet these criteria?

```
sfo_feb_flights <- nycflights %>%
  filter(dest == "SFO", month == 2)
glimpse(sfo_feb_flights)
```

```
## Rows: 68
## Columns: 16
## $ year      <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 20
13, …
## $ month     <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, …
## $ day       <int> 18, 3, 15, 18, 24, 25, 7, 15, 13, 8, 11, 13, 25, 20, 12,
27,…
## $ dep_time  <int> 1527, 613, 955, 1928, 1340, 1415, 1032, 1805, 1056, 656,
191…
## $ dep_delay <dbl> 57, 14, -5, 15, 2, -10, 1, 20, -4, -4, 40, -2, -1, -6, -
7, 2…
## $ arr_time  <int> 1903, 1008, 1313, 2239, 1644, 1737, 1352, 2122, 1412, 10
39, …
## $ arr_delay <dbl> 48, 38, -28, -6, -21, -13, -10, 2, -13, -6, 2, -5, -30,
-22,…
## $ carrier   <chr> "DL", "UA", "DL", "UA", "UA", "UA", "B6", "AA", "UA", "D
L", …
## $ tailnum   <chr> "N711ZX", "N502UA", "N717TW", "N24212", "N76269", "N532U
A", …
## $ flight    <int> 1322, 691, 1765, 1214, 1111, 394, 641, 177, 642, 1865, 2
72, …
## $ origin    <chr> "JFK", "JFK", "JFK", "EWR", "EWR", "JFK", "JFK", "JFK",
"JFK…
## $ dest      <chr> "SFO", "SFO", "SFO", "SFO", "SFO", "SFO", "SFO", "SFO",
"SFO…
## $ air_time  <dbl> 358, 367, 338, 353, 341, 355, 359, 338, 347, 361, 332, 3
51, …
## $ distance  <dbl> 2586, 2586, 2586, 2565, 2565, 2586, 2586, 2586, 2586, 25
86, …
## $ hour      <dbl> 15, 6, 9, 19, 13, 14, 10, 18, 10, 6, 19, 8, 10, 18, 7, 1
7, 1…
## $ minute    <dbl> 27, 13, 55, 28, 40, 15, 32, 5, 56, 56, 10, 33, 48, 49, 2
3, 2…
```

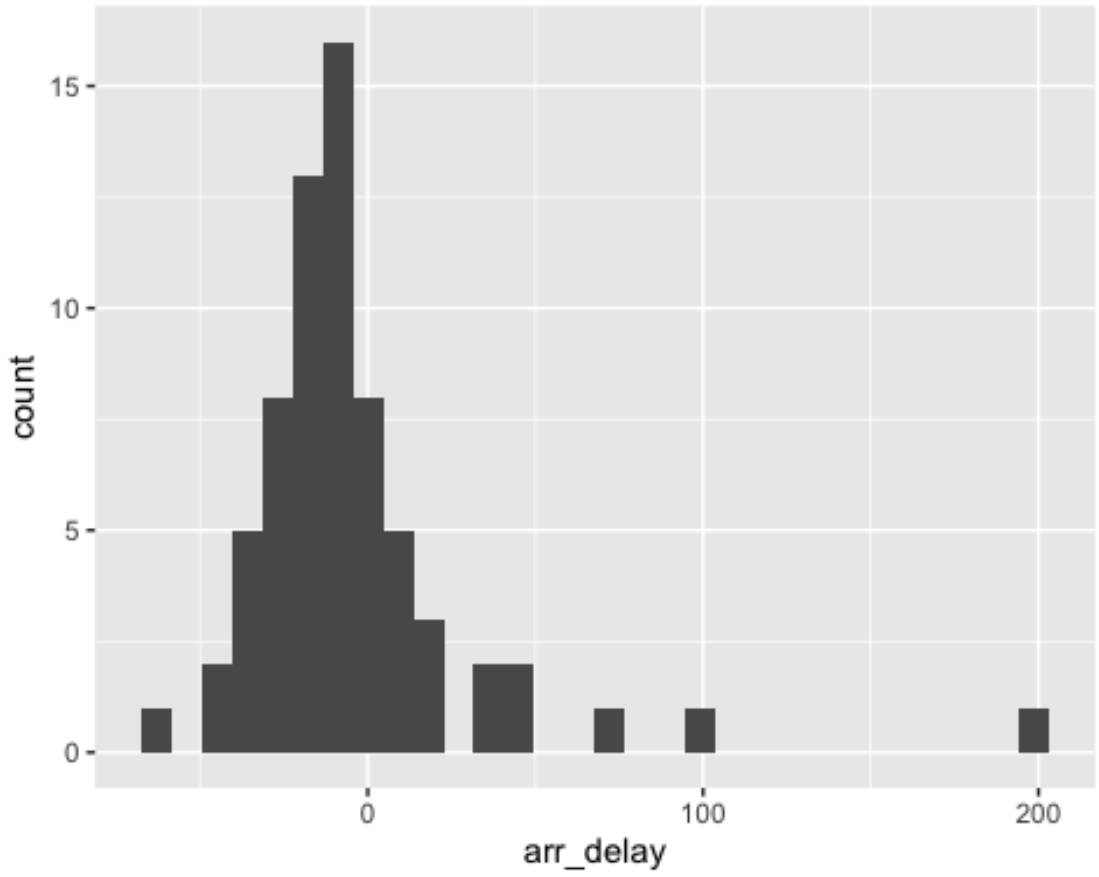
```
# |label: number-flights-meeting criteria
sfo_feb_flights %>%
  summarise(mean_dt = mean(flight),
            median_dt = median(flight),
            iqr_dt    = IQR(flight),
            n_flights = n())
```

```
## # A tibble: 1 × 4
##   mean_dt median_dt iqr_dt n_flights
##     <dbl>     <dbl>  <dbl>     <int>
## 1    795.       641  1402.        68
```

Exercise 3: Describe the distribution of the arrival delays of these flights using a histogram and appropriate summary statistics. Hint: The summary statistics you use should depend on the shape of the distribution.

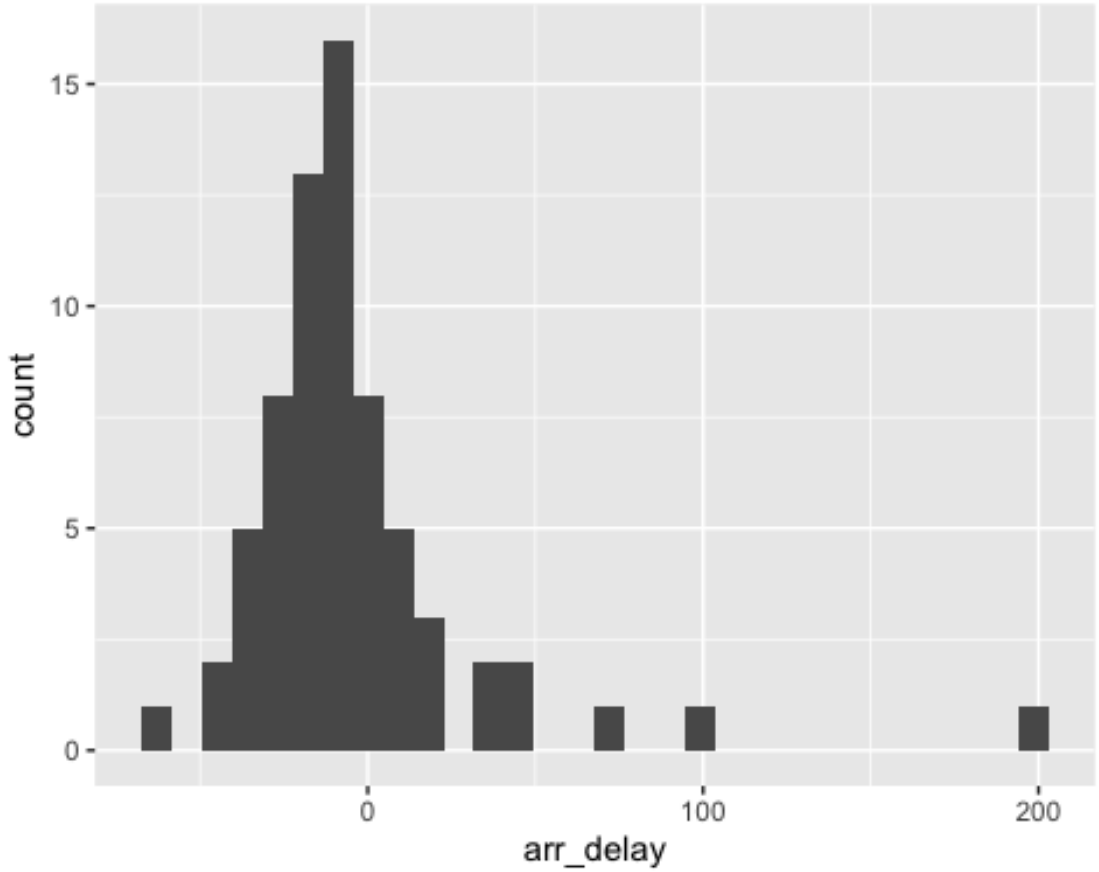
The histograms below reveal that the arrival delays of flights headed to SFO in February were normally distributed. One can therefore study the scope of the spread and the centrality of this distribution by examining its mean,median, standard deviation and IQR.

```
sfo_feb_flights_delay  <-  sfo_feb_flights %>%
  filter(flight == arr_delay)
     ggplot(data = sfo_feb_flights, aes(x =       arr_delay)) +
       geom_histogram()

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




```
sfo_feb_flights_delay <- sfo_feb_flights %>%
  select(arr_delay)
```

```
    ggplot(data = sfo_feb_flights, aes(x =       arr_delay)) +
      geom_histogram()

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




```
 summary(nycflights$arr_delay, na.rm = TRUE)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  -73.000  -17.000   -5.000    7.101   14.000 1272.000

sfo_feb_flights %>%
  select(arr_delay) %>%
  summarize(mean_ad = mean(arr_delay, na.rm = TRUE),
            median_ad = median(arr_delay, na.rm = TRUE),
            sd_ad     = sd(arr_delay, na.rm = TRUE),
            iqr_ad    = IQR(arr_delay, na.rm = TRUE),
                    n_flights  = n())

## # A tibble: 1 × 5
##   mean_ad median_ad sd_ad iqr_ad n_flights
##     <dbl>     <dbl> <dbl>  <dbl>     <int>
## 1    -4.5       -11  36.3   23.2        68
```

Exercise 4 Calculate the median and interquartile range for arr_delays of flights in in the sfo_feb_flights data frame, grouped by carrier. Which carrier has the most variable arrival delays?

Below is the summary statistics of arrival delays in February. The summary reveals UA as the carrier that had the most variable arrival delays in February: 21.

```
sfo_feb_flights|>
  group_by(carrier)|>
  summarise(mean_ad = mean(arr_delay),
            iqr_ad  = IQR(arr_delay),
          n_flights = n())
```

```
## # A tibble: 5 × 4
##   carrier mean_ad iqr_ad n_flights
##   <chr>     <dbl>  <dbl>     <int>
## 1 AA        11.5    17.5        10
## 2 B6        -6.33   12.2         6
## 3 DL       -13.5    22          19
## 4 UA         1.81   22          21
## 5 VX       -13.8    21.2        12
```

Exercise 5 Suppose you really dislike departure delays and you want to schedule your travel in a month that minimizes your potential departure delay leaving NYC. One option is to choose the month with the lowest mean departure delay. Another option is to choose the month with the lowest median departure delay. What are the pros and cons of these two choices?

```
 nycflights %>%
  group_by(month) %>%
  summarise(mean_dd = mean(dep_delay)) %>%
  arrange(desc(mean_dd))
```

```
## # A tibble: 12 × 2
##    month mean_dd
##    <int>   <dbl>
##  1     7   20.8
##  2     6   20.4
##  3    12   17.4
##  4     4   14.6
##  5     3   13.5
##  6     5   13.3
##  7     8   12.6
##  8     2   10.7
##  9     1   10.2
## 10     9    6.87
## 11    11    6.10
## 12    10    5.88
```

```
nycflights %>%
  group_by(month) %>%
  summarise(median_dd = median(dep_delay)) %>%
  arrange(desc(median_dd))
```

```
## # A tibble: 12 × 2
##    month median_dd
##    <int>     <dbl>
##  1    12         1
##  2     6         0
##  3     7         0
##  4     3        -1
##  5     5        -1
##  6     8        -1
##  7     1        -2
##  8     2        -2
##  9     4        -2
## 10    11        -2
## 11     9        -3
## 12    10        -3
```

I will rely more on the median than on the mean to select the best month that minimizes travel delays. The median is not as sensitive to outliers as is the mean. It is therefore a stable and accurate measurement of centrality that provides one with more certainty than the mean. The mean, on the contrary, is sensitive to outliers and to a moving reality.In the case of air travels for instance, the same mean of departure delays will vary in understanding according to whether the volume of the airport activities is high or low, among many other things. Therefore the mean, although a good measurement of average in theory, lacks in certainty and in the stability because of its sensitivity to outliers.

Exercise 6 If you were selecting an airport simply based on on time departure percentage, which NYC airport would you choose to fly out of?

Based on the summary statistics and graphic below Laguardia airport (LGA) is the airport I would like to fly out of as it shows the best on-time departure rate in NYC.

1. First, let's classify the flights as "on-time" or "delayed"

```
nycflights <-nycflights|>
  mutate(dep_type = ifelse(dep_delay < 5, "on time", "delayed"))
head(nycflights)
```

```
## # A tibble: 6 × 17
##    year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
flight
##   <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>
<int>
## 1  2013     6    30      940        15     1216        -4 VX      N626VA
407
## 2  2013     5     7     1657        -3     2104        10 DL      N3760C
```

```
329
## 3  2013    12     8      859        -1     1238        11 DL      N712TW
422
## 4  2013     5    14     1841        -4     2122       -34 DL      N914DL
2391
## 5  2013     7    21     1102        -3     1230        -8 9E      N823AY
3652
## 6  2013     1     1     1817        -3     2008         3 AA      N3AXAA
353
## # ℹ 7 more variables: origin <chr>, dest <chr>, air_time <dbl>, distance <
dbl>,
## #   hour <dbl>, minute <dbl>, dep_type <chr>
```

2. Let's find the best on -time airport in NY

```
nycflights %>%
  group_by(origin) %>%
  summarise(ot_dep_rate = sum(dep_type == "on time") / n()) %>%
  arrange(desc(ot_dep_rate))
```

```
## # A tibble: 3 × 2
##   origin ot_dep_rate
##   <chr>        <dbl>
## 1 LGA          0.728
## 2 JFK          0.694
## 3 EWR          0.637
```

```
ggplot(data = nycflights, aes(x = origin, fill = dep_type)) +
geom_bar()
```

More practice

Exercise 7 Mutate the data frame so that it includes a new variable that contains the average speed, avg_speed traveled by the plane for each flight (in mph). Hint: Average speed can be calculated as distance divided by number of hours of travel, and note that air_time is given in minutes.

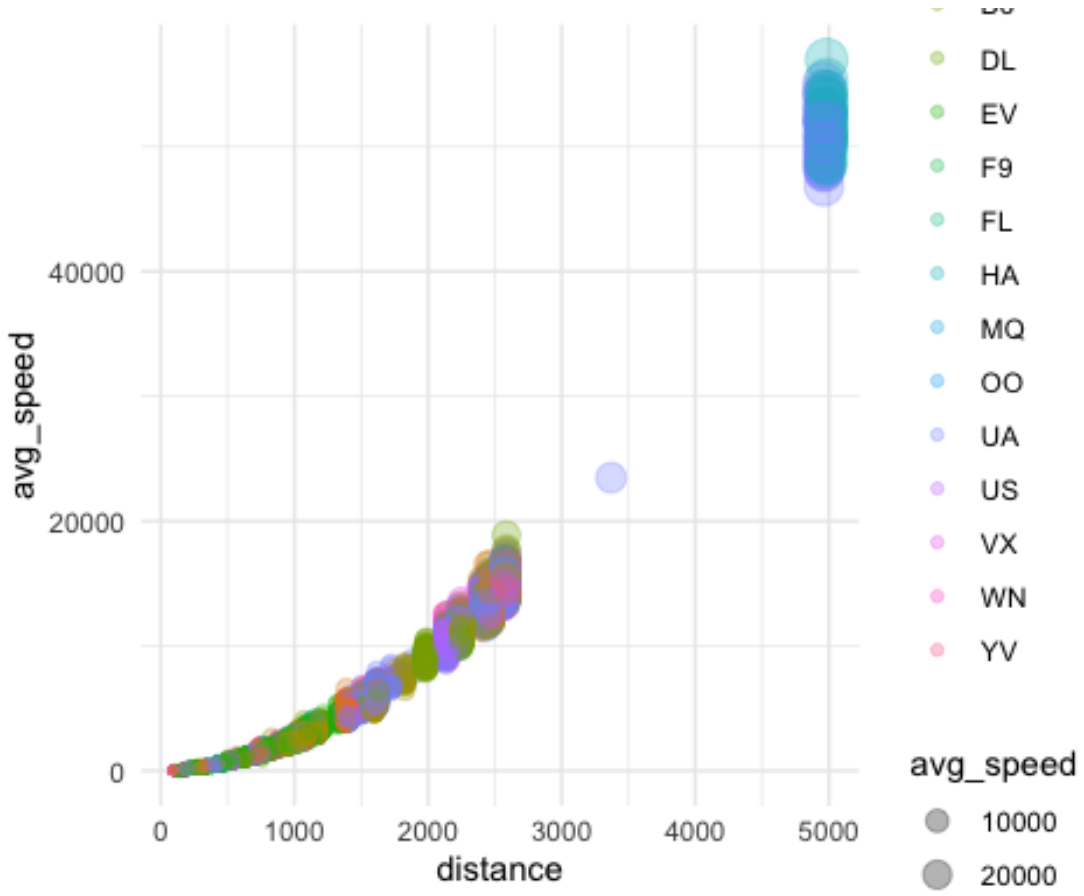
```
nycflights <- nycflights|>
  mutate(avg_speed = distance/60 * air_time)
head(nycflights)
```

```
## # A tibble: 6 × 18
##    year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
flight
##   <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>
<int>
## 1  2013     6    30      940        15     1216        -4 VX      N626VA
407
## 2  2013     5     7     1657        -3     2104        10 DL      N3760C
```

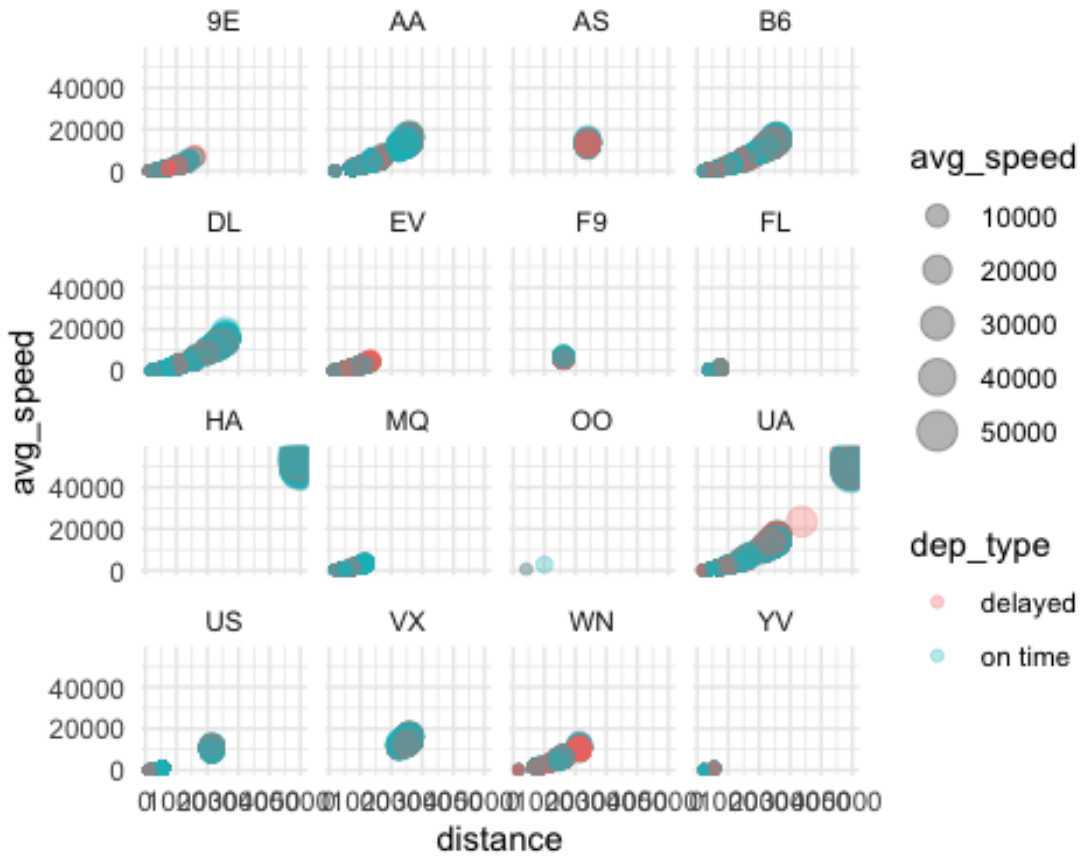
```
329
## 3  2013    12     8      859         -1     1238         11 DL       N712TW
422
## 4  2013     5    14     1841         -4     2122        -34 DL       N914DL
2391
## 5  2013     7    21     1102         -3     1230         -8 9E       N823AY
3652
## 6  2013     1     1     1817         -3     2008          3 AA       N3AXAA
353
## # ℹ 8 more variables: origin <chr>, dest <chr>, air_time <dbl>, distance <
dbl>,
## #   hour <dbl>, minute <dbl>, dep_type <chr>, avg_speed <dbl>
```

Exercise 8 Make a scatterplot of avg_speed vs. distance. Describe the relationship between average speed and distance. Hint: Use geom_point().

```
nycflights <- nycflights|>
  mutate(avg_speed = distance/60 * air_time)
ggplot(nycflights, aes (x= distance, y= avg_speed, color = carrier, size = av
g_speed ))+
theme_minimal()+
  geom_point(alpha = 0.3)
```



```
nycflights <- nycflights|>
  mutate(avg_speed = distance/60 * air_time)

ggplot(nycflights, aes (x= distance, y= avg_speed, color = dep_type, size = a
vg_speed ))+
theme_minimal()+
  geom_point(alpha = 0.3) +
  facet_wrap(~carrier)
```

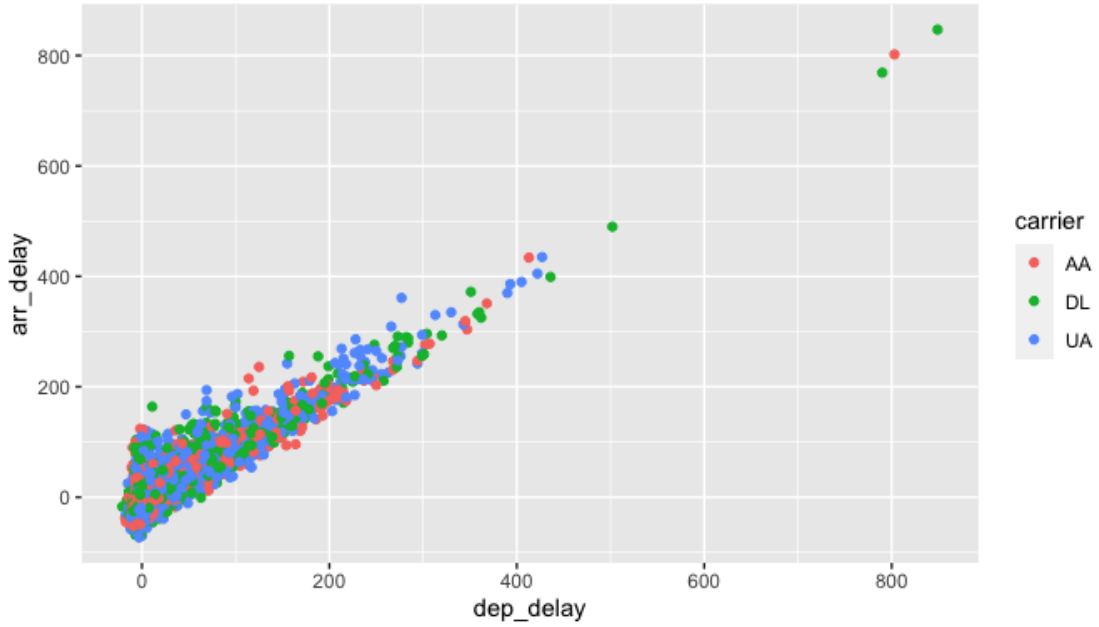


The correlation test below shows a strong positive correlation of 0.9506363 between average speed and distance within an interval confidence of 0.9495823 0.9516688

```
cor.test(nycflights$avg_speed, nycflights$distance)

##
##  Pearson's product-moment correlation
##
## data:  nycflights$avg_speed and nycflights$distance
## t = 554.26, df = 32733, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9495823 0.9516688
## sample estimates:
```

```
##       cor
## 0.9506363
```

Exercise 9 Replicate the following plot. Hint: The data frame plotted only contains flights from American Airlines, Delta Airlines, and United Airlines, and the points are `color`ed by `carrier`. Once you replicate the plot, determine (roughly) what the cutoff point is for departure delays where you can still expect to get to your destination on time.



The scatterplot below shows no correlation between departure delays of AA, DL, UA carriers and their arrival time, when experiencing a 30 minutes departure delay. Based on this fact I consider a 30 minutes to be the the cut to still arrive on time with the listed carriers.

```
acceptable_dpdelay <- dl_aa_ua %>%
  filter(minute == "30")
head(acceptable_dpdelay )
```

```
## # A tibble: 6 × 18
##    year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
flight
##   <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>
<int>
## 1  2013    12    18      730         0     1038        -2 AA      N3KCAA
29
## 2  2013     9    16     1730        -5     2008        -2 UA      N27421
1580
## 3  2013     2    20     1730         1     2044        -5 AA      N382AA
543
## 4  2013    11    16     1030        -5     1225         5 AA      N470AA
```

```
325
## 5  2013     9    25     1130        -7     1335        -9 DL      N915DE
2219
## 6  2013     7     3      630        -3      911       -17 UA      N37462
1701
## # ℹ 8 more variables: origin <chr>, dest <chr>, air_time <dbl>, distance <
dbl>,
## #   hour <dbl>, minute <dbl>, dep_type <chr>, avg_speed <dbl>
```

```
ggplot(data = acceptable_dpdelay, aes(x = dep_delay, y = arr_time, color= car
rier)) +
  geom_point()
```

```
library(latexpdf)
```