Code ▾

Lab 2: Intro to Data

Julian Adames-Ng

2022-02-13

Hide

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

Exercise 1

Look carefully at these three histograms. How do they compare? Are features revealed in one that are obscured in another?

-As we increase the bin width, the number of bins shown decreases whereas decreasing the bin width more finely tunes the data on departure delay times. When increasing bin width, it seems to bunch up the data into bigger chunks showing less accuracy on a smaller scale.

Hide

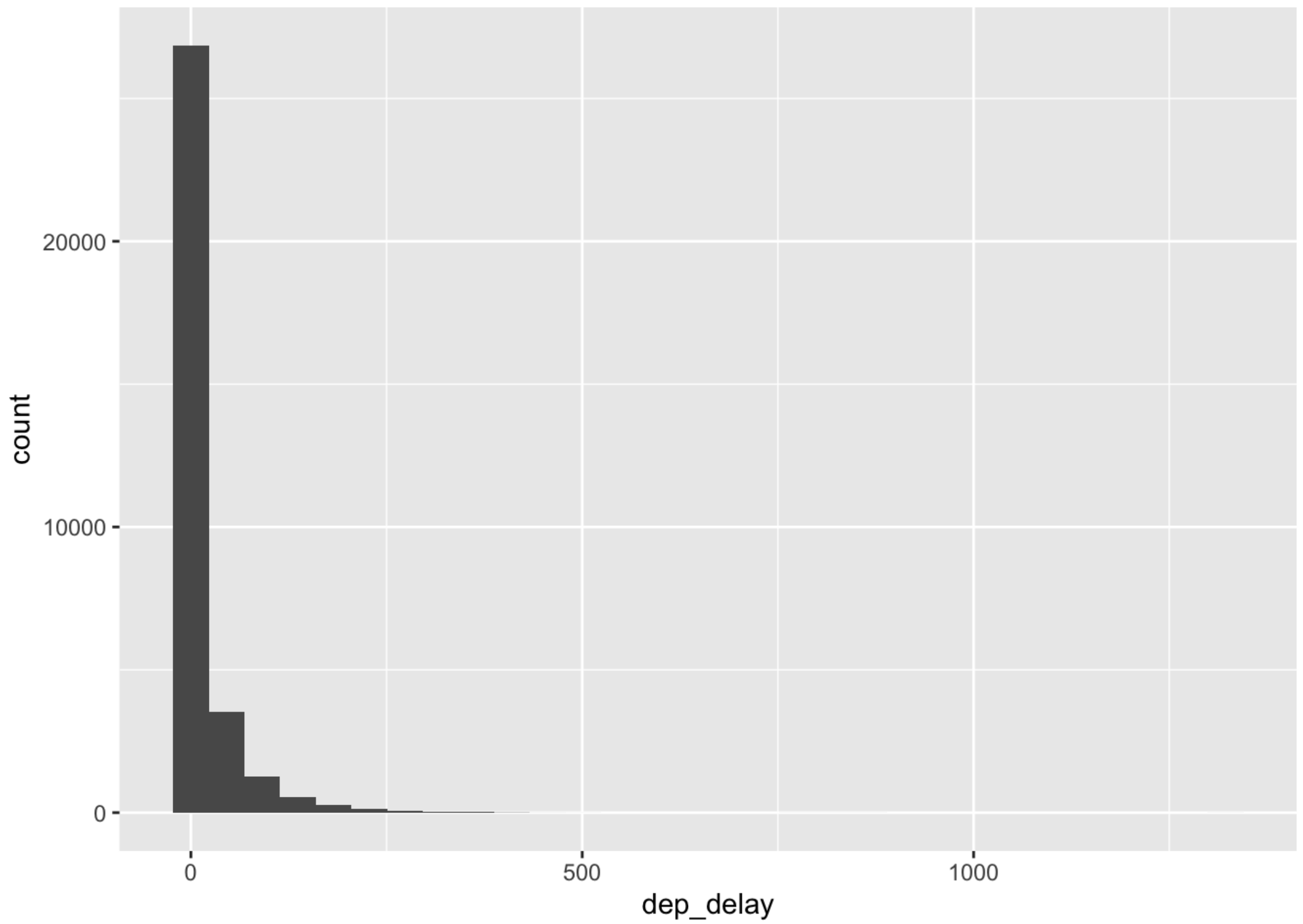
```
data(nycflights)
names(nycflights)
```

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

Hide

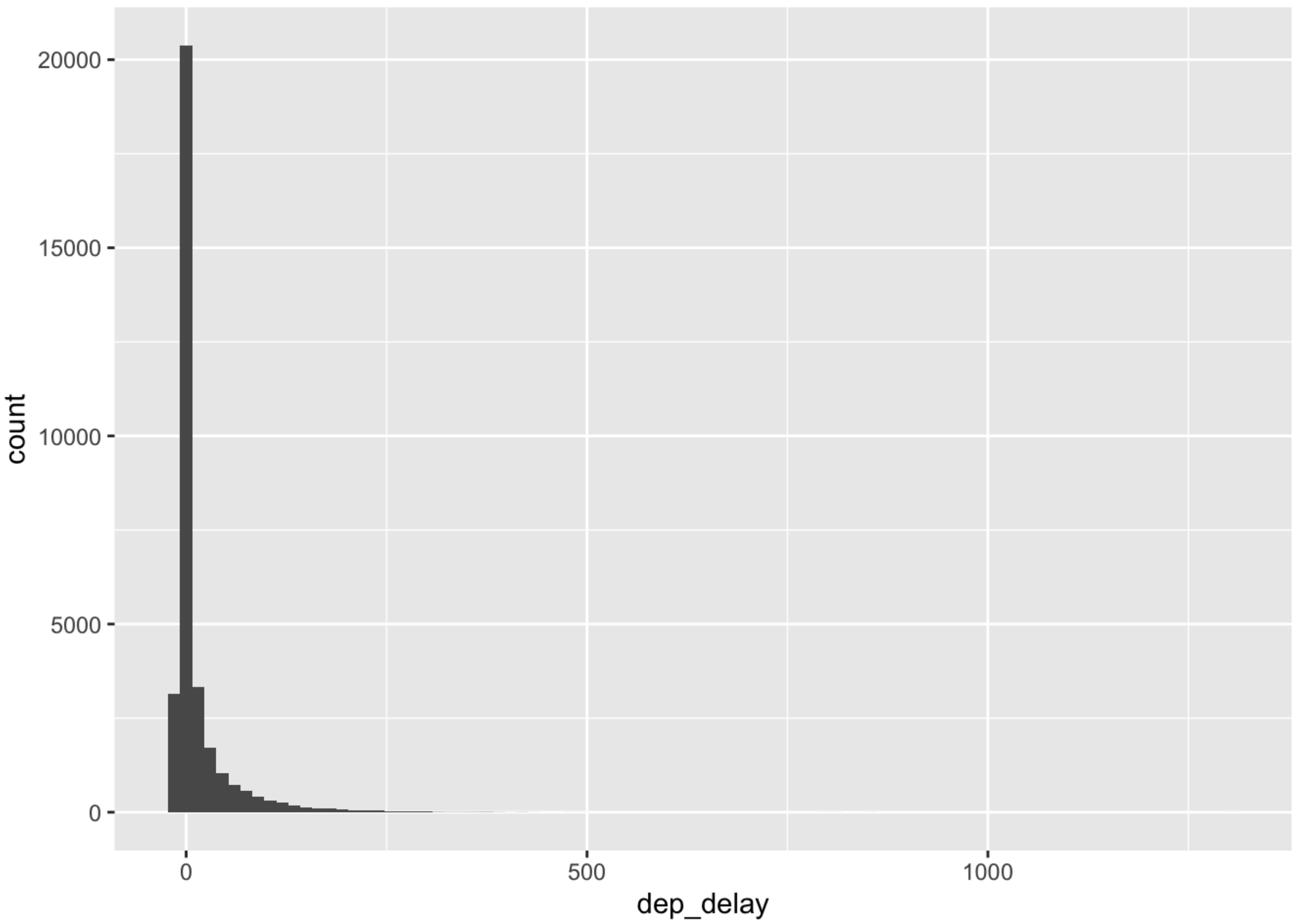
```
ggplot(data = nycflights, aes(x = dep_delay)) + geom_histogram()
```

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



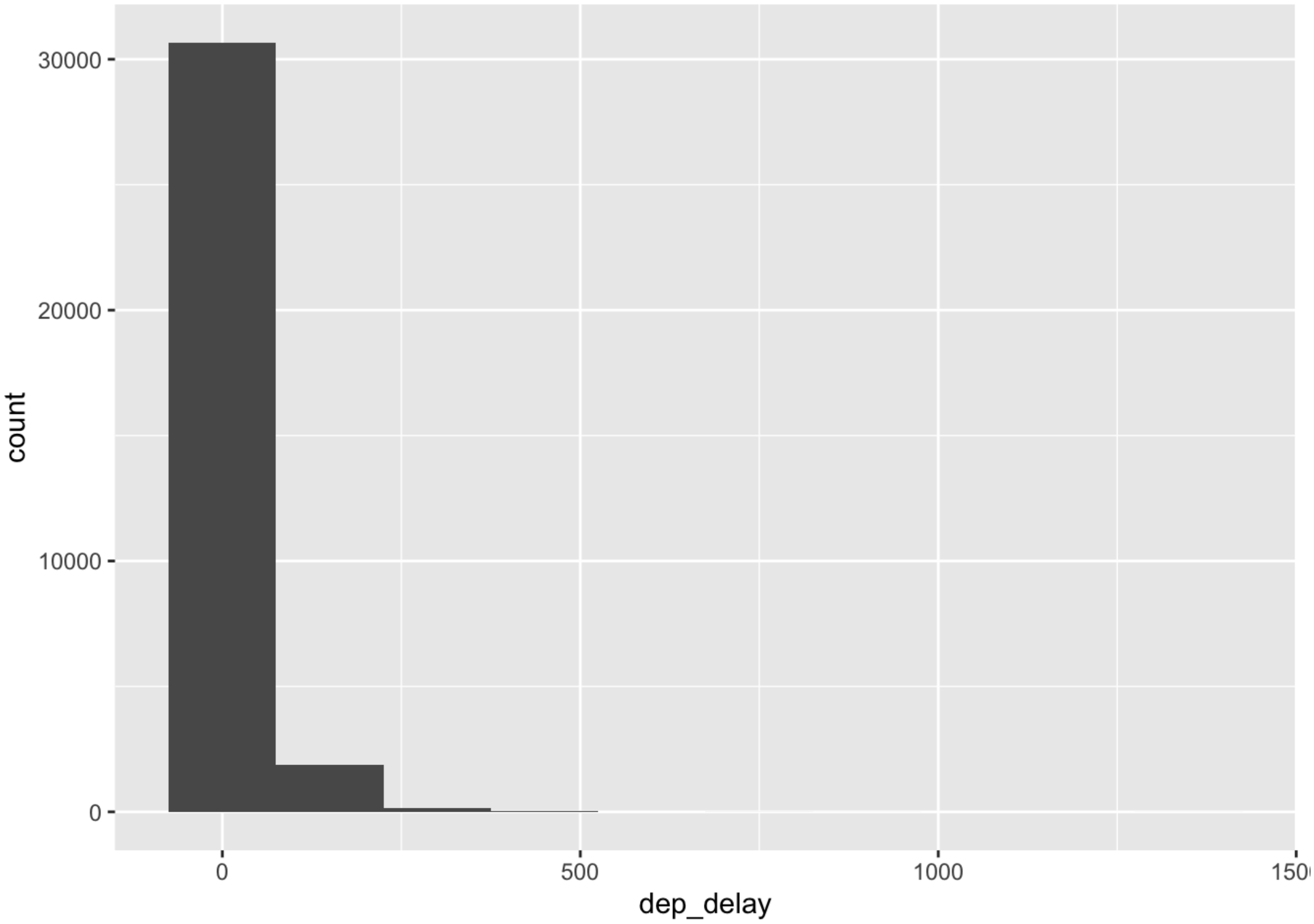
Hide

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



Hide

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



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?

-There are 68 flights that meet this criteria, based on the number of rows in the summary table. The sample size confirms this.

Hide

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

#shows 68 rows
sfo_feb_flights
```

```
## # A tibble: 68 × 16
##     year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
##    <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>
##  1  2013     2    18     1527        57     1903        48 DL      N711ZX
##  2  2013     2     3      613        14     1008        38 UA      N502UA
##  3  2013     2    15      955        -5     1313       -28 DL      N717TW
##  4  2013     2    18     1928        15     2239        -6 UA      N24212
##  5  2013     2    24     1340         2     1644       -21 UA      N76269
##  6  2013     2    25     1415       -10     1737       -13 UA      N532UA
##  7  2013     2     7     1032         1     1352       -10 B6      N627JB
##  8  2013     2    15     1805        20     2122         2 AA      N335AA
##  9  2013     2    13     1056        -4     1412       -13 UA      N532UA
## 10  2013     2     8      656        -4     1039        -6 DL      N710TW
## # … with 58 more rows, and 7 more variables: flight <int>, origin <chr>,
## #   dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>
```

Hide

```
#confirming sample size
sfo_feb_flights %>% summarise(n = n())
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1    68
```

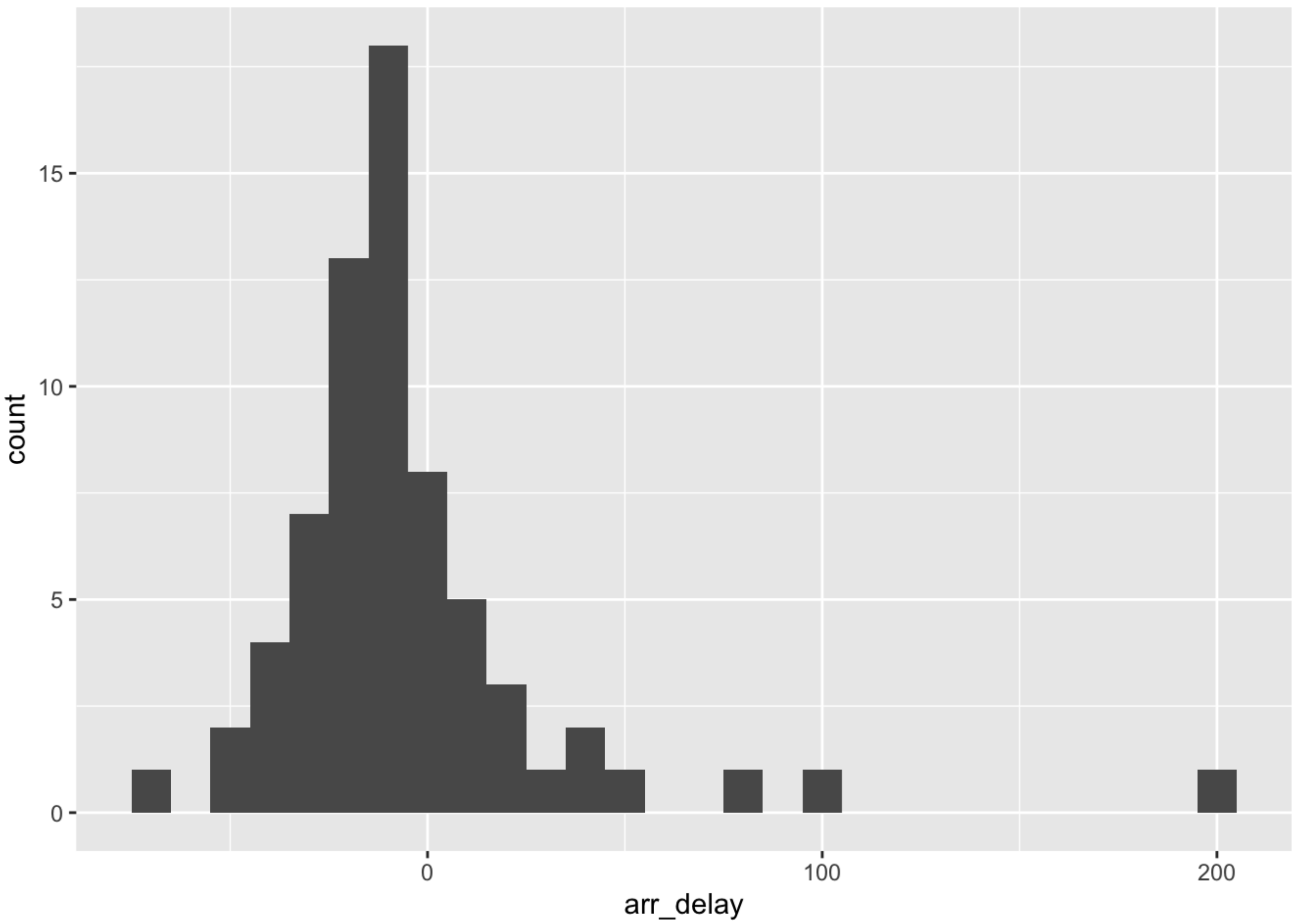
Exercise 3

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 shape of the histogram seems to be skewed right or somewhat normal with a few outliers toward the upper end of the data. The data is centered just below 0. Summarizing the statistics, we see that the measures of central tendency are -4.5 for the mean and -11 for the median which are consistent with the plotted distribution.

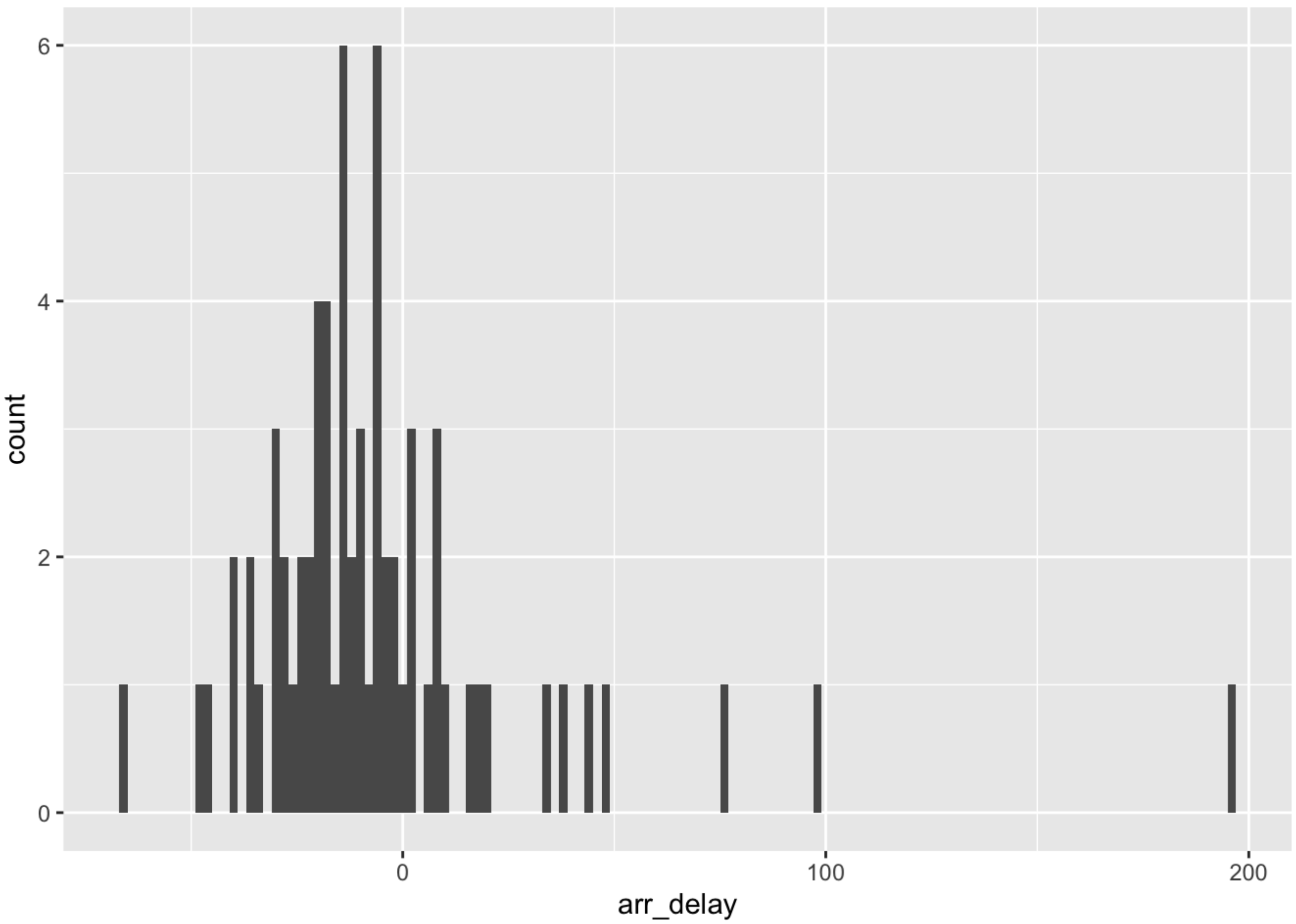
Hide

```
ggplot(data = sfo_feb_flights, aes(x = arr_delay)) + geom_histogram(binwidth = 10)
```



Hide

```
#using smaller binwidth
ggplot(data = sfo_feb_flights, aes(x = arr_delay)) + geom_histogram(binwidth = 2)
```



Hide

```
sfo_feb_flights %>% summarise(mean_ad = mean(arr_delay),
                              median_ad = median(arr_delay),
                              n = n())
```

```
## # A tibble: 1 × 3
##   mean_ad median_ad     n
##     <dbl>     <dbl> <int>
## 1    -4.5       -11    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?

-Using just the IQR, there seems to be a tie between DL and UA at 22. However, when using standard deviation, it’s clear that UA has the most variable arrival delays.

Hide

```
sfo_feb_flights %>%
  group_by(carrier) %>%
  summarise(median_ad = median(arr_delay),
            iqr_ad = IQR(arr_delay),
            sD_ad = sd(arr_delay),
            n_flights = n())
```

```
## # A tibble: 5 × 5
##   carrier median_ad iqr_ad sD_ad n_flights
##   <chr>       <dbl>  <dbl> <dbl>     <int>
## 1 AA            5     17.5  29.5        10
## 2 B6          -10.5   12.2  11.0         6
## 3 DL          -15     22    22.0        19
## 4 UA          -10     22    48.3        21
## 5 VX          -22.5   21.2  40.8        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?

-The pros of using the mean when determining when to depart is that it uses all of the information in the data set

-A con to using the mean is that extremely high or low values may affect the mean by a lot, giving a somewhat inaccurate interpretation of the data.

-An upside to using the median is that it is not affected by extreme values as is the mean. This gives a truer interpretation of the middle of the data in a wide range of data.

-A downside to using the median is that it disregards data that is not in the middle of the data set.

Hide

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

```
## # A tibble: 12 × 2
##    month mean_ad
##    <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
```

Hide

```
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
```

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 departure percentage alone, I would select LGA as the airport to fly out of.

Hide

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

nycflights
```

```
## # A tibble: 32,735 × 17
##     year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
##    <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>  
##  1  2013     6    30      940        15     1216        -4 VX      N626VA 
##  2  2013     5     7     1657        -3     2104        10 DL      N3760C 
##  3  2013    12     8      859        -1     1238        11 DL      N712TW 
##  4  2013     5    14     1841        -4     2122       -34 DL      N914DL 
##  5  2013     7    21     1102        -3     1230        -8 9E      N823AY 
##  6  2013     1     1     1817        -3     2008         3 AA      N3AXAA 
##  7  2013    12     9     1259        14     1617        22 WN      N218WN 
##  8  2013     8    13     1920        85     2032        71 B6      N284JB 
##  9  2013     9    26      725       -10     1027        -8 AA      N3FSAA 
## 10  2013     4    30     1323        62     1549        60 EV      N12163 
## # … with 32,725 more rows, and 8 more variables: flight <int>, origin <chr>,
## #   dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   dep_type <chr>
```

Hide

```
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
```

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.

Hide

```
nycflights <- nycflights %>%
  #divide air time by 60 to get time in hours
  #avg_speed is in miles per hour
  mutate(avg_speed = distance / (air_time / 60))

nycflights
```

```
## # A tibble: 32,735 × 18
##     year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
##    <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>  
##  1  2013     6    30      940        15     1216        -4 VX      N626VA 
##  2  2013     5     7     1657        -3     2104        10 DL      N3760C 
##  3  2013    12     8      859        -1     1238        11 DL      N712TW 
##  4  2013     5    14     1841        -4     2122       -34 DL      N914DL 
##  5  2013     7    21     1102        -3     1230        -8 9E      N823AY 
##  6  2013     1     1     1817        -3     2008         3 AA      N3AXAA 
##  7  2013    12     9     1259        14     1617        22 WN      N218WN 
##  8  2013     8    13     1920        85     2032        71 B6      N284JB 
##  9  2013     9    26      725       -10     1027        -8 AA      N3FSAA 
## 10  2013     4    30     1323        62     1549        60 EV      N12163 
## # … with 32,725 more rows, and 9 more variables: flight <int>, origin <chr>,
## #   dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   dep_type <chr>, avg_speed <dbl>
```

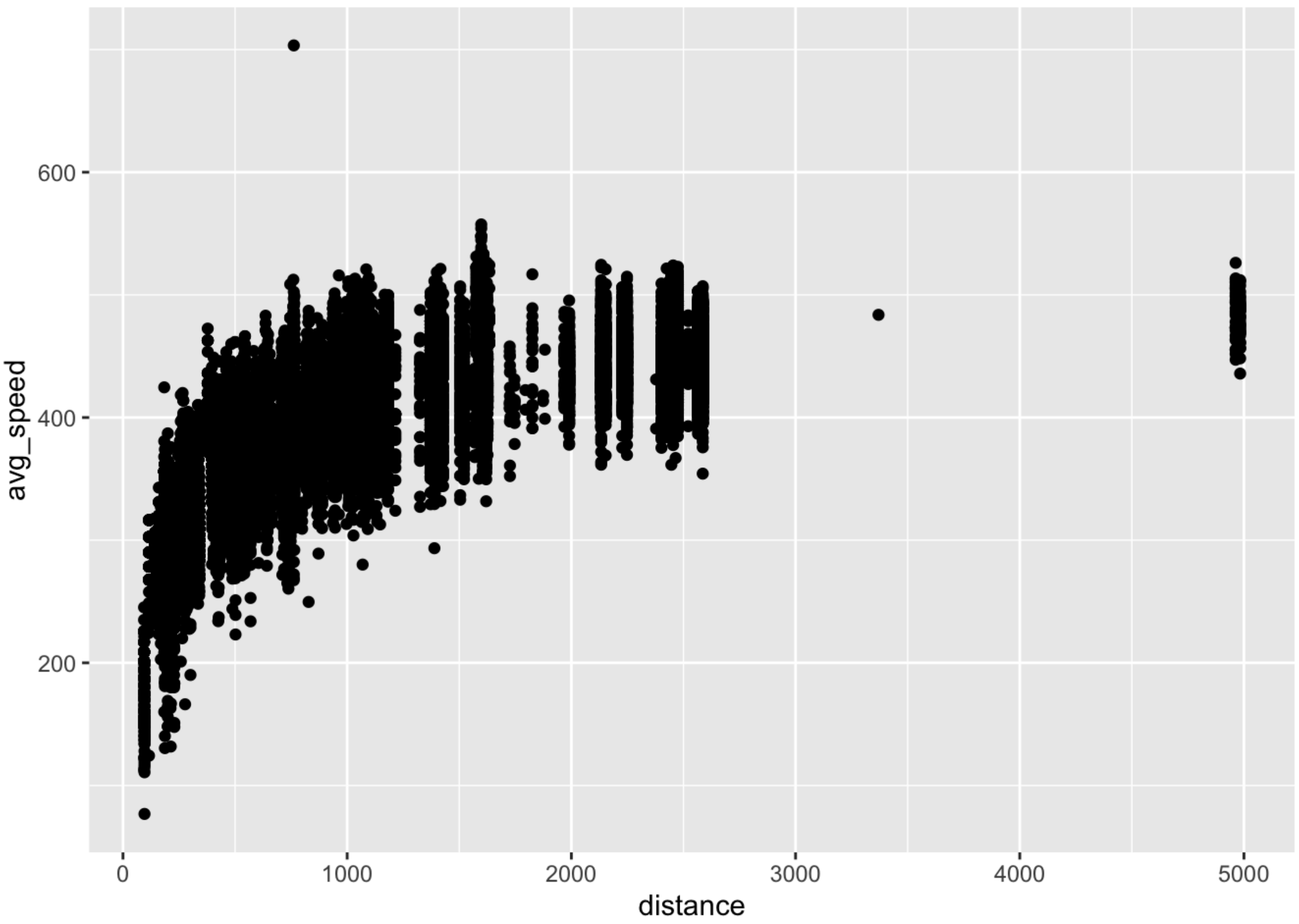
Exercise 8

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

-There seems to be a nonlinear relationship between distance and avg_speed. There's a small curvature in the data toward lower distances. The data shows that for shorter distances, the average speed is lower also. As the distance traveled increases, so does the average speed, but it seems to plateau with increasing distance.

Hide

```
ggplot(data = nycflights, aes(x = distance, y = avg_speed)) + geom_point()
```



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 colored 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.

-There seems to be a strong linear relationship between departure delays and arrival delays. Most flights that arrived on time had a departure delay of no more than 8-10 minutes or so as an extreme. In general, any departure delays below 5 minutes resulted in an on time arrival.

Hide

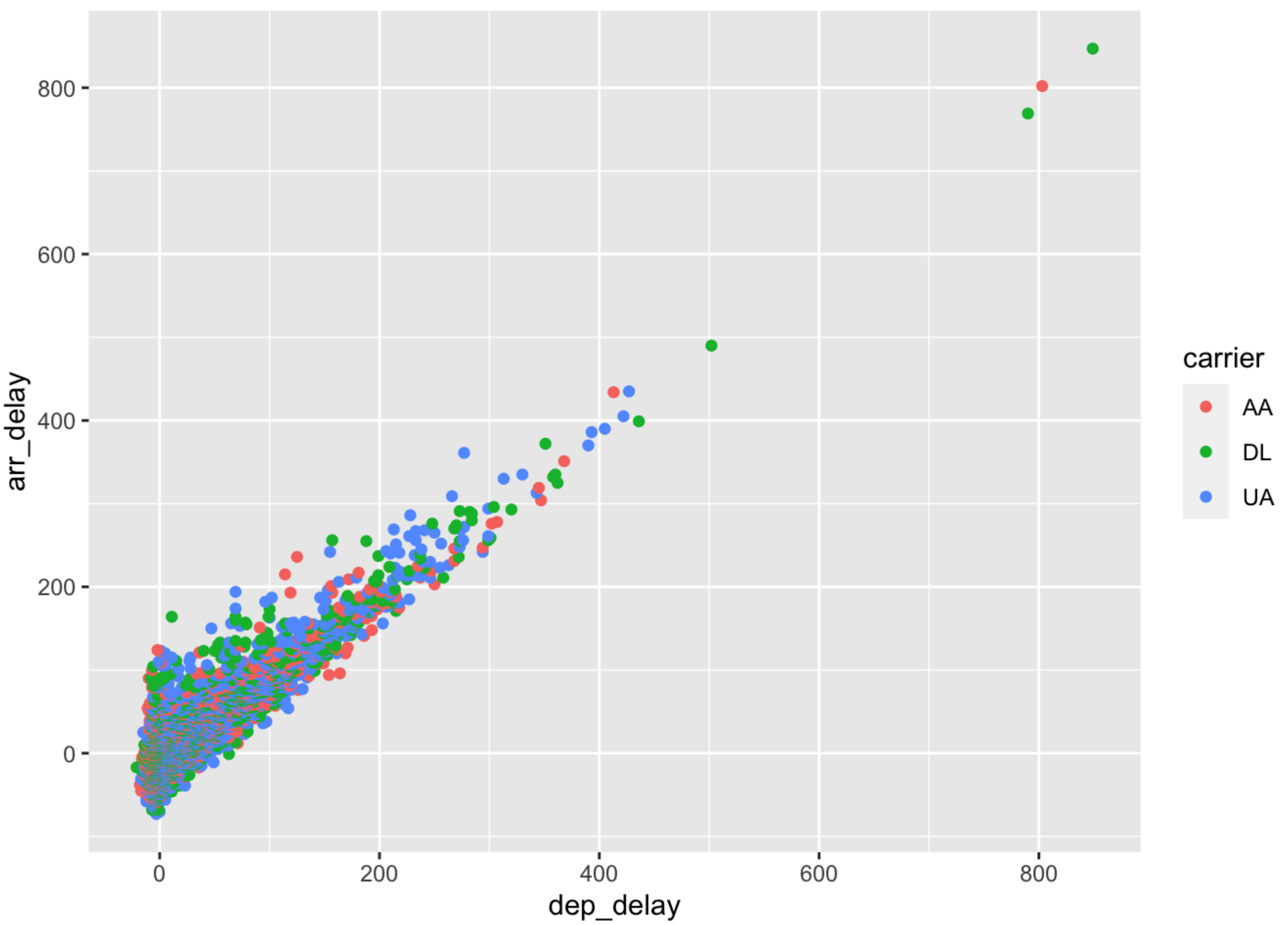
```
#use OR symbol...."|" to make a new data frame with only flights from AA, DL, or UA
flights_3 <- nycflights %>% filter(carrier == "UA"| carrier == "AA"| carrier == "D
        L") %>%
  arrange(arr_delay)

flights_3
```

```
## # A tibble: 13,709 × 18
##     year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
##    <int> <int> <int>    <int>     <dbl>    <int>     <dbl> <chr>   <chr>  
##  1  2013     5     2     1926        -3     2157       -73 UA      N24212 
##  2  2013     2    26     1335         0     1819       -70 UA      N76065 
##  3  2013     5     6     1924        -1     2145       -68 DL      N654DL 
##  4  2013     2    26     1918        -7     2155       -68 DL      N3768  
##  5  2013     5    13     1819        -6     2041       -65 UA      N24702 
##  6  2013     5     3     1556        -4     1847       -64 DL      N707TW 
##  7  2013     1     3     1228        -7     1503       -63 DL      N389DA 
##  8  2013     3    25     1723        -2     1958       -62 DL      N705TW 
##  9  2013     9     6     1439        -6     1656       -62 UA      N560UA 
## 10  2013     9    30     1423        -6     1626       -62 UA      N435UA 
## # … with 13,699 more rows, and 9 more variables: flight <int>, origin <chr>,
## #   dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   dep_type <chr>, avg_speed <dbl>
```

Hide

```
ggplot(data = flights_3, aes(x = dep_delay, y = arr_delay, color = carrier)) + geom
        _point()
```


arr_delay
dep_delay
carrier
● AA
● DL
● UA