Wk 2 - Lab 2

Tyler Frankenberg

2021-02-14

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

SFO in February

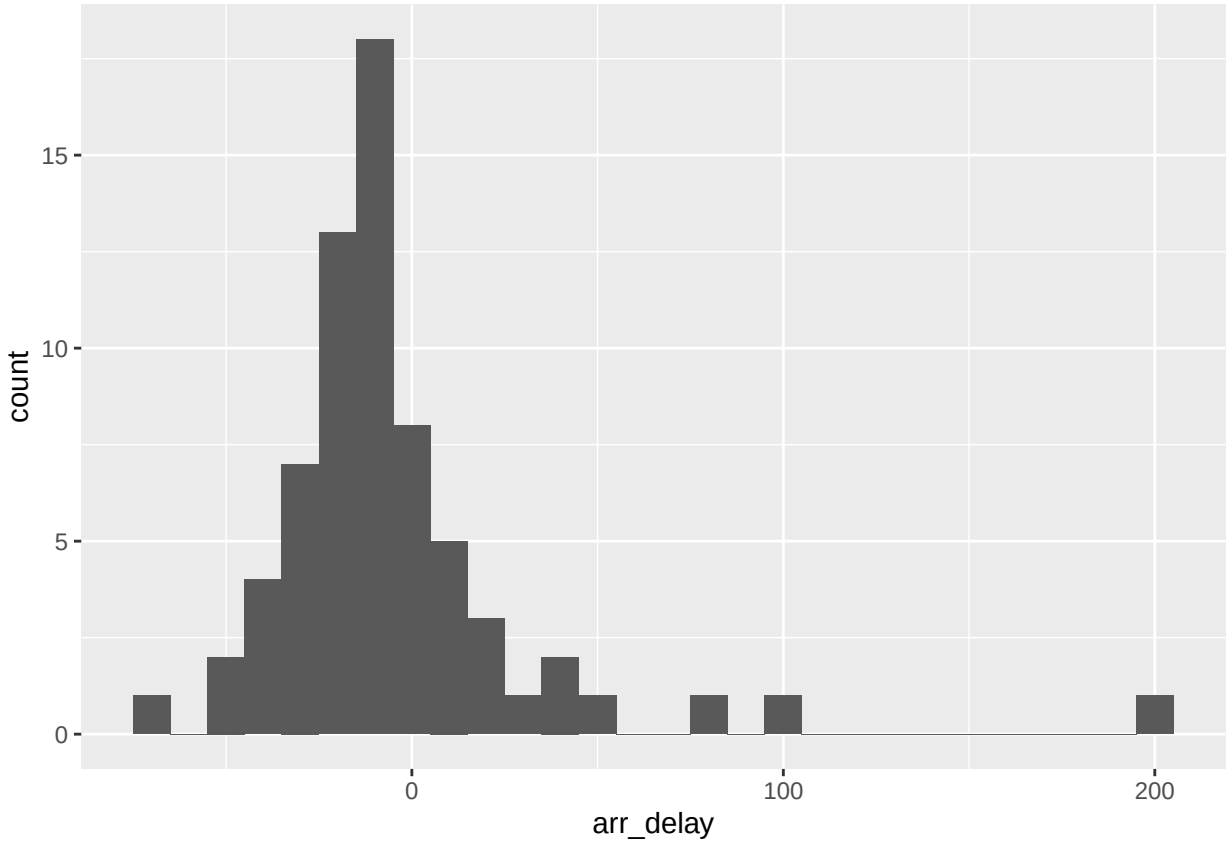
1. Create a new data frame that includes flights headed to SFO in February, and save this data frame as `sfo_feb_flights`. How many flights meet these criteria?

There are 68 flights that meet this criteria

2. 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 mean arrival time of flights to SFO in February was: 4.5 minutes early. The data is skewed rightward.

```
##create data frame

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

##histogram of arrival delays

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




```
##summary statistics

sfo_feb_flights %>%
  summarise(mean_ad   = mean(arr_delay),
            median_ad = median(arr_delay),
            sd_ad     = sd(arr_delay),
            min_ad    = min(arr_delay),
            max_ad    = max(arr_delay),
            n         = n())
```

```
## # A tibble: 1 x 6
##   mean_ad median_ad sd_ad min_ad max_ad     n
##     <dbl>     <dbl> <dbl>  <dbl>  <dbl> <int>
## 1    -4.5       -11  36.3    -66    196    68
```

Median & Interquartile Range

3. Calculate the median and interquartile range for `arr_delay`s of flights in in the `sfo_feb_flights` data frame, grouped by carrier. Which carrier has the most variable arrival delays?

With a IQR of 22.00, Delta and United are tied for the most variable arrival delays on flights to SFO in February

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

```
## # A tibble: 5 x 4
##   carrier median_ad iqr_ad n_flights
## * <chr>       <dbl>  <dbl>     <int>
## 1 AA            5     17.5        10
## 2 B6          -10.5   12.2         6
## 3 DL          -15     22          19
## 4 UA          -10     22          21
## 5 VX          -22.5   21.2        12
```

...

Mean vs. Median

4. 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?

When you choose the month with the lowest median departure delay, you can have confidence that you are equally as likely to experience either a shorter or longer departure delay than the median. However, your median does not reflect the relative influence of outliers in either direction, or tell you anything about how widely the data varies. When you choose the mean, you know equally little about the shape of the data, and you know nothing about the relative likelihood of experiencing shorter or longer delays than the mean. Your small average could be the combined result of a relative few 'way-way-better-than-average' departures, and many, many more 'worse-than-average'delays.'

```
nycflights %>%
  group_by(month) %>%
  summarise(mean_dd = mean(dep_delay),
            median_dd = median(dep_delay)) %>%
  arrange(mean_dd)
```

```
## # A tibble: 12 x 3
##    month mean_dd median_dd
##    <int>   <dbl>     <dbl>
##  1    10    5.88        -3
##  2    11    6.10        -2
##  3     9    6.87        -3
##  4     1   10.2         -2
##  5     2   10.7         -2
##  6     8   12.6         -1
##  7     5   13.3         -1
##  8     3   13.5         -1
##  9     4   14.6         -2
## 10    12   17.4          1
## 11     6   20.4          0
## 12     7   20.8          0
```

```
nycflights %>%
  group_by(month) %>%
  summarise(median_dd = median(dep_delay),
            mean_dd = mean(dep_delay)) %>%
  arrange(median_dd)
```

```
## # A tibble: 12 x 3
##    month median_dd mean_dd
##    <int>     <dbl>   <dbl>
##  1     9        -3    6.87
##  2    10        -3    5.88
##  3     1        -2   10.2
##  4     2        -2   10.7
##  5     4        -2   14.6
##  6    11        -2    6.10
##  7     3        -1   13.5
##  8     5        -1   13.3
##  9     8        -1   12.6
## 10     6         0   20.4
## 11     7         0   20.8
## 12    12         1   17.4
```

On Time Departure Rate

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

LGA all the way, baby!

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

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

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

More Practice

1. 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.

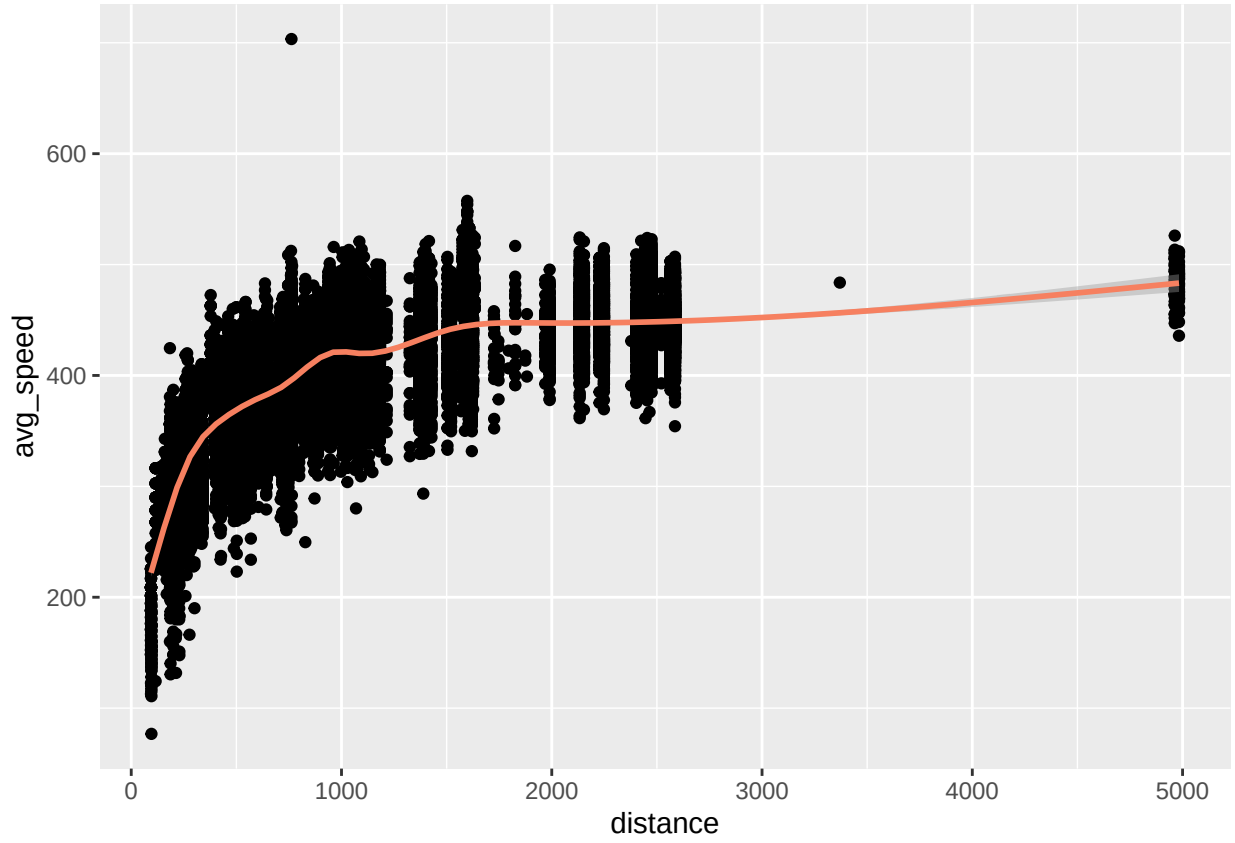
```
nycflights <- nycflights %>%
    mutate(at_in_hours = air_time / 60)

nycflights <- nycflights %>%
    mutate(avg_speed = distance / at_in_hours)
```

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

Average speed rises steeply as distance increases up to a limit of approximately 425 mph at 1000 miles traveled, then rises very gradually toward 500 mph as distance traveled approaches 5000. To examine the relationship more clearly at distances < 1000, we'll show the graph again using a logarithmic scale on the x-axis.

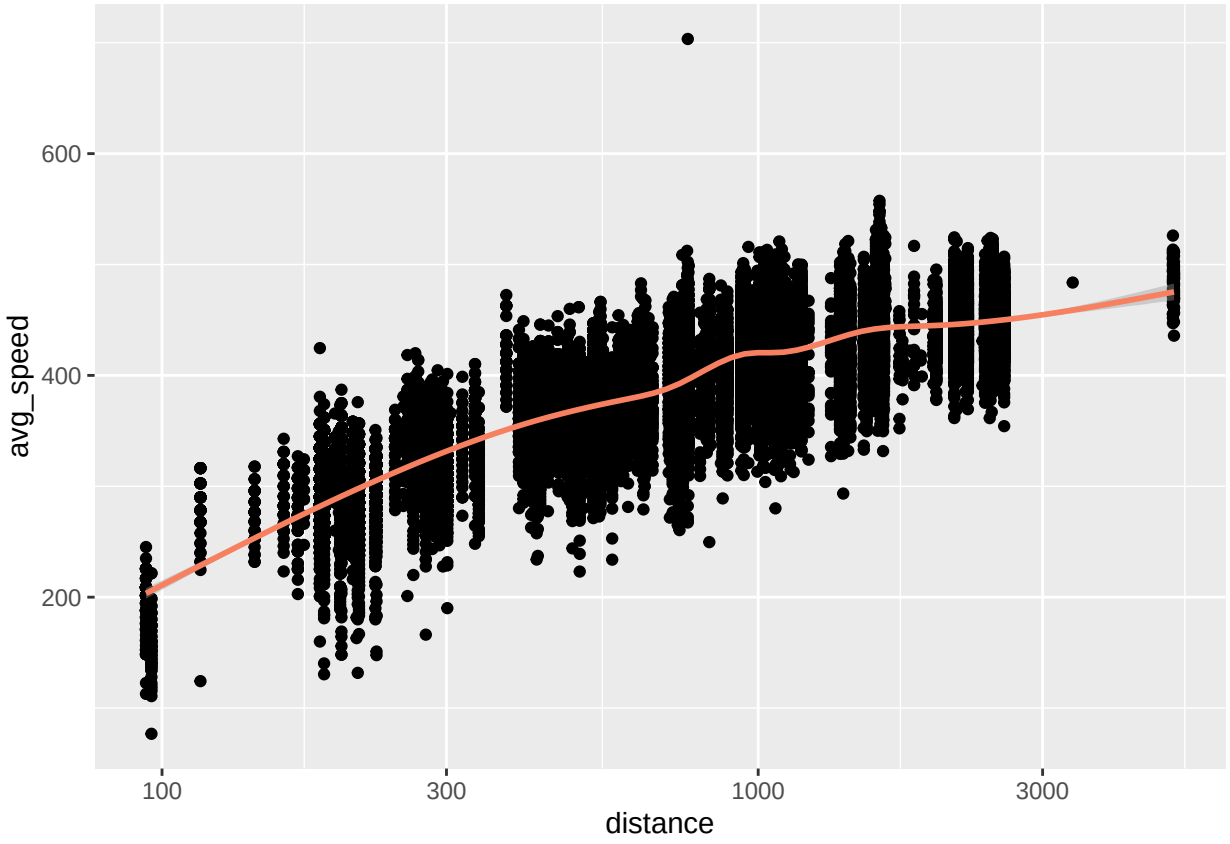
```
ggplot(nycflights, aes(distance, avg_speed)) +
  geom_point() +
  geom_smooth(color="#f68060")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
ggplot(nycflights, aes(distance, avg_speed)) +
  geom_point() +
  geom_smooth(color="#f68060")+
  scale_x_log10()
```

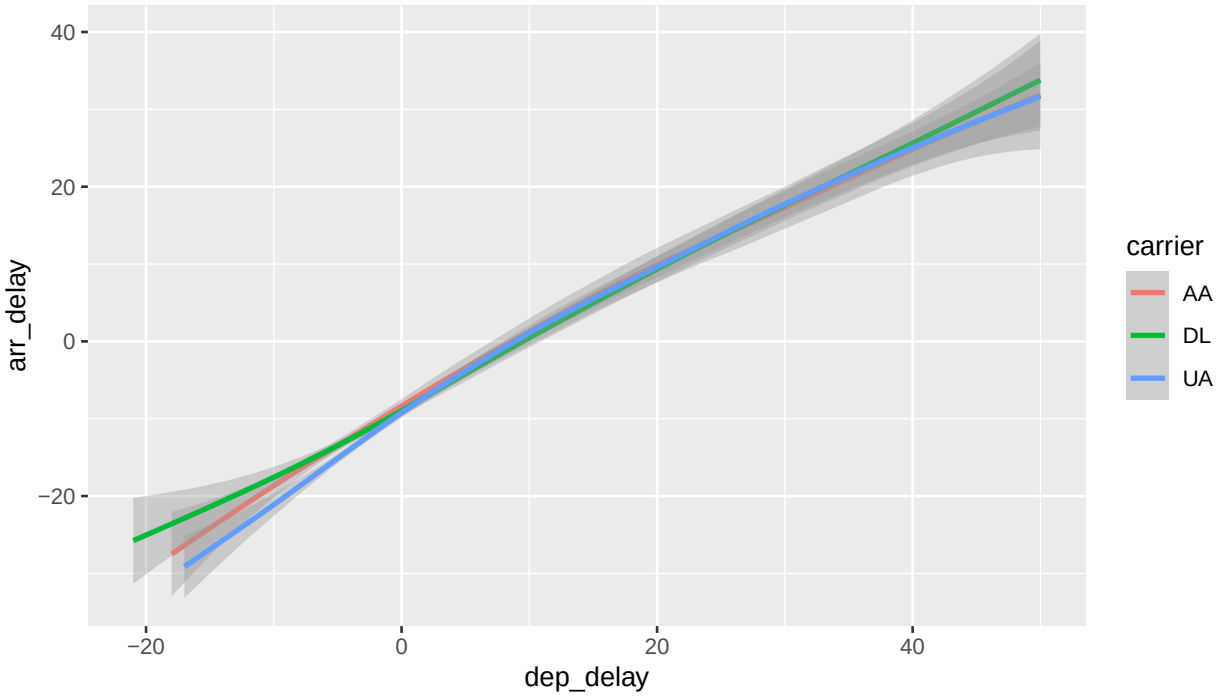
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



3. 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.

By subsetting the data to examine only cases where arrival and departure time are both less than 50 minutes, and changing the plot to show the regression line for each carrier, we can see there is on average a linear relationship between minutes delayed on departure, and minutes delayed on arrival. It appears from the graph that we can depart up to approximately 15 minutes late, and still arrive on average less than 5 minutes late.

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```


40
20
0
−20
arr_delay
−20
0
20
40
dep_delay
carrier
AA
DL
UA