

Visualization of flight routes

Packages needed

```
library(tidyverse)
library(usmap)
library(sf)
```

Dataset

The `flight_df_raw` data we have here is a complete record of all domestic flights for American, Delta, United, and Southwest in 2019 (`nycflights13::flights` contains only flights departing from NYC).

```
flight_df_raw <- read_csv("data/flight_df.csv") |> select(-DepTime, -ArrTime)
airports_raw <- read_csv("data/airports.csv")
us_map_sf <- us_map(regions = 'states') |> filter(!abbr %in% c("AK", "HI"))
```

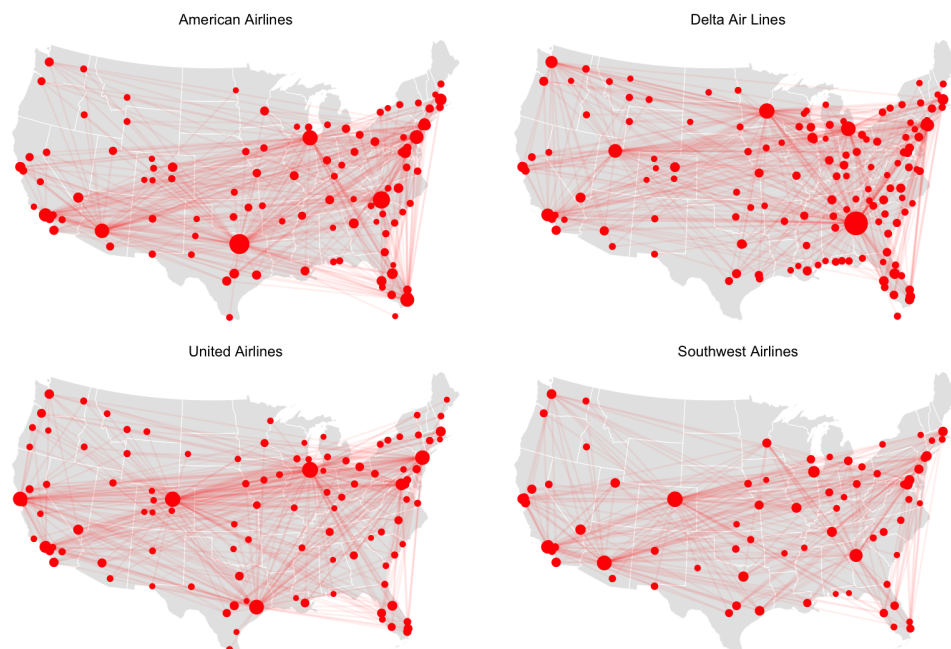
This is some wrangling to match the coordinate reference system (CRS) of the airports with the CRS of the US map (we will talk about CRS on week 5 Friday's class).

```
airports <- airports_raw |>
  # create an sf object
  st_as_sf(coords = c("x", "y"), crs = 4326) |>
  # transform to the same CRS as the us_map_sf
  st_transform(crs = st_crs(us_map_sf)) |>
  # extract the coordinates
  mutate(x = st_coordinates(geometry)[,1], y = st_coordinates(geometry)[,2]) |>
  # focus on airports on continental US
  filter(between(x, -3000000, 3000000), y < 700000) |>
  # clean up
  rename(airport = ident) |>
  as_tibble() |>
  select(airport, x, y)
```

airports

```
## # A tibble: 19,317 x 3
##   airport      x      y
##   <chr>    <dbl>  <dbl>
## 1 OJ0     1399199. -1366000.
## 2 2A8     1180494. -1107747.
## 3 AL03     1149352. -1265974.
## 4 EET     1230576. -1216678.
## 5 AL39     1226771. -1208917.
## 6 8A0     1262389. -1093420.
## 7 ALX     1310365. -1233325.
## 8 AIV     1100991. -1243840.
## 9 AL67     1138225. -1537664.
## 10 EKY     1215351. -1203918.
## # i 19,307 more rows
```

Your goal



Let's dissect the components of the plot. Our plot contains three layers:

- 1) The underlying map using the `us_map_sf` data. We will cover how to create a map in week 5 Friday.
- 2) The points for the airport: we want to show the location of all the airports, both origin and destination. The size of the points are proportional to the number of flights at the airport. All the points are red.
- 3) The lines for the routes: we want to show all the routes between two airports. Each route is represented by a line segment connecting the two airports. We make them dim (low alpha) so that the map is not too busy (also red).

Points representing the airports

You may want your data to look like this :

airline	airport	long	lat	n
American Airlines	DFW	-97.038	32.896	500
Delta Air Lines	ATL	-84.427	33.640	600

```
nodes_df <- flight_df_raw |>
  # this function standardizes the variable names to lowercase with underscores
  janitor::clean_names()
  # write some code to pivot the origin and dest airport into one coloumn (week 6 Monday)

  # write some code to group by airline and airport, then count the number of flights at each airport

  # write some code to join with the airports data to get the coordinates of each airport
```

This is my first few rows of the data (I divide the count by 100,000 to make the size of the points reasonable):

reporting_airline	airport	n	x	y
<fct>	<chr>	<dbl>	<dbl>	<dbl>

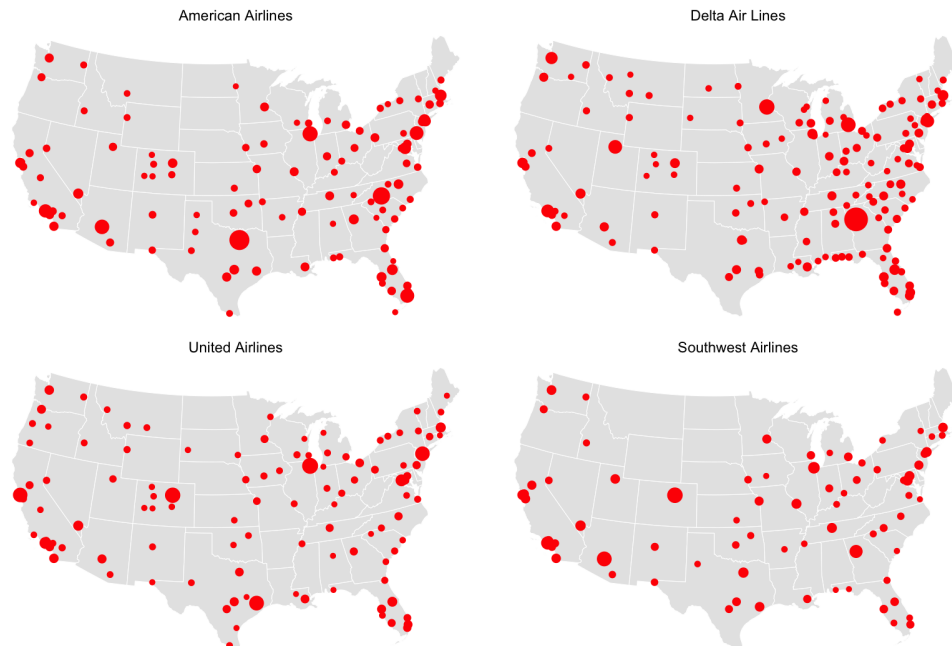
```

1 American Airlines ABQ      0.0525 -603105. -1082686.
2 American Airlines AGS      0.0001 1668088. -1113377.
3 American Airlines ALB      0.0168 2093897.   90840.
4 American Airlines AMA      0.00446 -155480. -1084678.
5 American Airlines ANC      0.00538      NA      NA

```

You can see there are airports that don't get the coordinates after the join - Let's worry about those later.

After these steps, your plot may look like this:



Lines representing the routes

You may want your data to look like this:

airline	id	type	airport	long	lat
American Airlines	1	dep	DFW	-97.038	32.896
American Airlines	1	arr	LAX	-118.408	33.942
Delta Air Lines	2	dep	ATL	-84.427	33.640
Delta Air Lines	2	arr	JFK	-73.778	40.641

```

routes_df <- flight_df_raw |>
  janitor::clean_names()
# here you don't need to plot the repeated routes
# what would you do to find the unique routes for each airline?

# Again you may want to do the pivoting here (week 6 Monday)

# and the join

```

My first few rows of the `routes_df` data look like this:

```

reporting_airline  id type  airport      x      y
<fct>             <int> <chr> <chr>      <dbl> <dbl>

```

1	American Airlines	1	origin	LAX	-1689593.	-1043457.
2	American Airlines	1	dest	DFW	278083.	-1338465.
3	American Airlines	2	origin	CLT	1718766.	-891789.
4	American Airlines	2	dest	PBI	1982322.	-1802086.
5	American Airlines	3	origin	OGG	NA	NA

Final plot - Assemble all the components

Now you have the dataset for each component - can you build the final plot?

```
# initialize the ggplot
```

```
# the us map layer
```

```
# the points layer
```

```
# the routes layer
```

```
# themes, facets, etc
```

ggplot2 produces some warning about

- 1: Removed 32 rows containing missing values or values outside the scale range (``geom_point()``).
- 2: Removed 204 rows containing missing values or values outside the scale range (``geom_line()``).

This signals something about the missing values. What would you do to suppress these warnings?