R for Data Science: Relational data

Monica Buczynski

8/22/2020

Note: The purpose of this document is to showcase a sample of skills that I learned in *R for Data Science* (chapter: Relational data) by Garrett Grolemund and Hadley Wickham. Particularly, I focus on relational data using dplyr. All scripts were taken from https://r4ds.had.co.nz/relational-data.html and https://jrnold.github.io/r4ds-exercise-solutions/index.html. The code for each exercise was studied carefully for understanding and then was retyped manually into R to maximize the learning experience; however, many of the original scripts were altered for further analysis and presentation aesthetics or I added my own code for further analysis.

The skills that I focused on in this chapter include:

- Keys
- Mutating joins
- Filtering joins
- Join problems
- Set operations

1) View and summerize data

```r
# airlines lets you look up the full carrier name from its abbreviated code:
airlines %>%
  print(n=5)
```

```
## # A tibble: 16 x 2
##   carrier name
##   <chr>   <chr>
## 1 9E      Endeavor Air Inc.
## 2 AA      American Airlines Inc.
## 3 AS      Alaska Airlines Inc.
## 4 B6      JetBlue Airways
## 5 DL      Delta Air Lines Inc.
## # ... with 11 more rows
```

```r
# airports gives information about each airport, identified by the faa airport code:
airports %>%
  print(n=5)
```

```
## # A tibble: 1,458 x 8
##   faa   name                             lat   lon   alt    tz dst   tzone
##   <chr> <chr>                          <dbl> <dbl> <dbl> <dbl> <chr> <chr>
## 1 04G   Lansdowne Airport               41.1 -80.6  1044    -5 A     America/New_Y~
## 2 06A   Moton Field Municipal Airp~  32.5 -85.7   264    -6 A     America/Chica~
## 3 06C   Schaumburg Regional             42.0 -88.1   801    -6 A     America/Chica~
## 4 06N   Randall Airport                 41.4 -74.4   523    -5 A     America/New_Y~
## 5 09J   Jekyll Island Airport           31.1 -81.4    11    -5 A     America/New_Y~
## # ... with 1,453 more rows
```

```r
# summary for all appropriate integer variables in airports dataset
summary(airports1 <- airports %>%
  select(-faa, -name))
```

```
##       lat              lon                alt                tz
##  Min.   :19.72   Min.   :-176.65   Min.   : -54.00   Min.   :-10.000
##  1st Qu.:34.26   1st Qu.:-119.19   1st Qu.:  70.25   1st Qu.: -8.000
##  Median :40.09   Median : -94.66   Median : 473.00   Median : -6.000
##  Mean   :41.65   Mean   :-103.39   Mean   :1001.42   Mean   : -6.519
##  3rd Qu.:45.07   3rd Qu.: -82.52   3rd Qu.:1062.50   3rd Qu.: -5.000
##  Max.   :72.27   Max.   : 174.11   Max.   :9078.00   Max.   :  8.000
##      dst               tzone
##  Length:1458        Length:1458
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

```r
# flights gives information about each flight, identified by carrier and flight number:
flights %>%
  print(n=5, width = Inf)
```

```
## # A tibble: 336,776 x 19
##    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
## 1  2013     1     1      517            515         2      830            819
```

```
## 2  2013     1     1      533            529         4      850            830
## 3  2013     1     1      542            540         2      923            850
## 4  2013     1     1      544            545        -1     1004           1022
## 5  2013     1     1      554            600        -6      812            837
##   arr_delay carrier flight tailnum origin dest  air_time distance  hour minute
##       <dbl> <chr>    <int> <chr>   <chr>  <chr>    <dbl>    <dbl> <dbl>  <dbl>
## 1        11 UA        1545 N14228  EWR    IAH        227     1400     5     15
## 2        20 UA        1714 N24211  LGA    IAH        227     1416     5     29
## 3        33 AA        1141 N619AA  JFK    MIA        160     1089     5     40
## 4       -18 B6         725 N804JB  JFK    BQN        183     1576     5     45
## 5       -25 DL         461 N668DN  LGA    ATL        116      762     6      0
##   time_hour
##   <dttm>
## 1 2013-01-01 05:00:00
## 2 2013-01-01 05:00:00
## 3 2013-01-01 05:00:00
## 4 2013-01-01 05:00:00
## 5 2013-01-01 06:00:00
## # ... with 336,771 more rows
```

```r
# summary for all appropriate integer variables in flights dataset
summary(flights1 <- flights %>%
          select(dep_delay, arr_delay, air_time, distance))
```

```
##    dep_delay          arr_delay           air_time        distance
##  Min.   : -43.00   Min.   : -86.000   Min.   : 20.0   Min.   :  17
##  1st Qu.:  -5.00   1st Qu.: -17.000   1st Qu.: 82.0   1st Qu.: 502
##  Median :  -2.00   Median :  -5.000   Median :129.0   Median : 872
##  Mean   :  12.64   Mean   :   6.895   Mean   :150.7   Mean   :1040
##  3rd Qu.:  11.00   3rd Qu.:  14.000   3rd Qu.:192.0   3rd Qu.:1389
##  Max.   :1301.00   Max.   :1272.000   Max.   :695.0   Max.   :4983
##  NA's   :8255      NA's   :9430       NA's   :9430
```

```r
# planes gives information about each plane, identified by its tailnum:
planes %>%
  print(n=5)
```

```
## # A tibble: 3,322 x 9
##   tailnum  year type           manufacturer   model  engines seats speed engine
##   <chr>   <int> <chr>          <chr>          <chr>    <int> <int> <int> <chr>
## 1 N10156   2004 Fixed wing mu~ EMBRAER        EMB-1~       2    55    NA Turbo-~
## 2 N102UW   1998 Fixed wing mu~ AIRBUS INDUST~ A320-~       2   182    NA Turbo-~
## 3 N103US   1999 Fixed wing mu~ AIRBUS INDUST~ A320-~       2   182    NA Turbo-~
## 4 N104UW   1999 Fixed wing mu~ AIRBUS INDUST~ A320-~       2   182    NA Turbo-~
## 5 N10575   2002 Fixed wing mu~ EMBRAER        EMB-1~       2    55    NA Turbo-~
## # ... with 3,317 more rows
```

```r
# summary for all appropriate integer variables in planes dataset
summary(planes1 <- planes %>%
          select(year, engines:speed))
```

```
##       year         engines          seats           speed
##  Min.   :1956   Min.   :1.000   Min.   :  2.0   Min.   : 90.0
##  1st Qu.:1997   1st Qu.:2.000   1st Qu.:140.0   1st Qu.:107.5
##  Median :2001   Median :2.000   Median :149.0   Median :162.0
##  Mean   :2000   Mean   :1.995   Mean   :154.3   Mean   :236.8
```

```
##  3rd Qu.:2005   3rd Qu.:2.000   3rd Qu.:182.0   3rd Qu.:432.0
##  Max.   :2013   Max.   :4.000   Max.   :450.0   Max.   :432.0
##  NA's   :70                                     NA's   :3299
```

```r
# weather gives the weather at each NYC airport for each hour:
weather %>%
  print(n=5)
```

```
## # A tibble: 26,115 x 15
##   origin  year month   day  hour  temp  dewp humid wind_dir wind_speed wind_gust
##   <chr>  <int> <int> <int> <int> <dbl> <dbl> <dbl>    <dbl>      <dbl>     <dbl>
## 1 EWR     2013     1     1     1  39.0  26.1  59.4      270      10.4         NA
## 2 EWR     2013     1     1     2  39.0  27.0  61.6      250       8.06        NA
## 3 EWR     2013     1     1     3  39.0  28.0  64.4      240      11.5         NA
## 4 EWR     2013     1     1     4  39.9  28.0  62.2      250      12.7         NA
## 5 EWR     2013     1     1     5  39.0  28.0  64.4      260      12.7         NA
## # ... with 26,110 more rows, and 4 more variables: precip <dbl>,
## #   pressure <dbl>, visib <dbl>, time_hour <dttm>
```

```r
# summary for all appropriate integer variables in weather dataset
summary(weather1 <- weather %>%
          select(-origin:-hour, -time_hour))
```

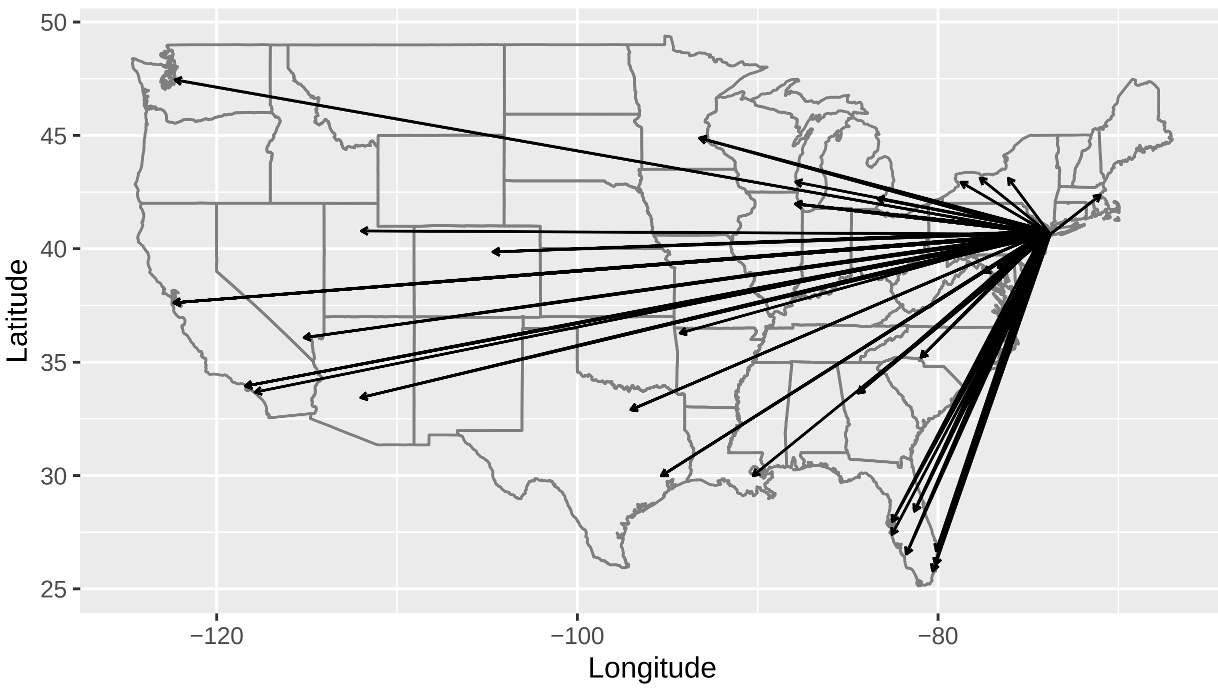
```
##       temp             dewp           humid           wind_dir    
##  Min.   : 10.94   Min.   :-9.94   Min.   : 12.74   Min.   :  0.0  
##  1st Qu.: 39.92   1st Qu.:26.06   1st Qu.: 47.05   1st Qu.:120.0  
##  Median : 55.40   Median :42.08   Median : 61.79   Median :220.0  
##  Mean   : 55.26   Mean   :41.44   Mean   : 62.53   Mean   :199.8  
##  3rd Qu.: 69.98   3rd Qu.:57.92   3rd Qu.: 78.79   3rd Qu.:290.0  
##  Max.   :100.04   Max.   :78.08   Max.   :100.00   Max.   :360.0  
##  NA's   :1        NA's   :1       NA's   :1        NA's   :460    
##    wind_speed         wind_gust         precip            pressure     
##  Min.   :   0.000   Min.   :16.11   Min.   :0.000000   Min.   : 983.8  
##  1st Qu.:   6.905   1st Qu.:20.71   1st Qu.:0.000000   1st Qu.:1012.9  
##  Median :  10.357   Median :24.17   Median :0.000000   Median :1017.6  
##  Mean   :  10.518   Mean   :25.49   Mean   :0.004469   Mean   :1017.9  
##  3rd Qu.:  13.809   3rd Qu.:28.77   3rd Qu.:0.000000   3rd Qu.:1023.0  
##  Max.   :1048.361   Max.   :66.75   Max.   :1.210000   Max.   :1042.1  
##  NA's   :4          NA's   :20778                      NA's   :2729    
##      visib       
##  Min.   : 0.000  
##  1st Qu.:10.000  
##  Median :10.000  
##  Mean   : 9.255  
##  3rd Qu.:10.000  
##  Max.   :10.000  
## 
```

2) Imagine you wanted to draw (approximately) the route each plane flies from its origin to its destination. What variables would you need? What tables would you need to combine?

Variables and tables needed:

- latitude and longitude of the origin and the destination airports of each flight
  - flights table contains origin (origin) destination (dest)
  - airport contain latitude (lat) and longitude (lon)
  - use inner join to drop canceled/missing flights

```
flights_latlon <- flights %>%
  inner_join(select(airports, origin = faa, origin_lat = lat, origin_lon = lon),
             by = "origin"
             ) %>%
  inner_join(select(airports, dest = faa, dest_lat = lat, dest_lon = lon),
             by = "dest"
             )

# plots the approximate flight paths of the first 100 flights in the flights dataset

flights_latlon %>%
  slice(1:100) %>%
  ggplot(aes(
    x = origin_lon, xend = dest_lon,
    y = origin_lat, yend = dest_lat)) +
  borders("state") +
  geom_segment(arrow = arrow(length = unit(0.1, "cm"))) +
  coord_quickmap() +
  labs(y = "Latitude", x = "Longitude")
```


50
45
40
35
30
25
Latitude
−120
−100
−80
Longitude

Mutating Joins

3) Experimenting with mutating joins

```r
# Create a narrower dataset

(flights2 <- flights %>%
   select(year:day, hour, origin, dest, tailnum, carrier))
```

```
## # A tibble: 336,776 x 8
##     year month   day  hour origin dest  tailnum carrier
##    <int> <int> <int> <dbl> <chr>  <chr> <chr>   <chr>
##  1  2013     1     1     5 EWR    IAH   N14228  UA
##  2  2013     1     1     5 LGA    IAH   N24211  UA
##  3  2013     1     1     5 JFK    MIA   N619AA  AA
##  4  2013     1     1     5 JFK    BQN   N804JB  B6
##  5  2013     1     1     6 LGA    ATL   N668DN  DL
##  6  2013     1     1     5 EWR    ORD   N39463  UA
##  7  2013     1     1     6 EWR    FLL   N516JB  B6
##  8  2013     1     1     6 LGA    IAD   N829AS  EV
##  9  2013     1     1     6 JFK    MCO   N593JB  B6
## 10  2013     1     1     6 LGA    ORD   N3ALAA  AA
## # ... with 336,766 more rows
```

```r
#  Add the full airline name to the flights2

(flights2 %>%
  select(-origin, -dest) %>%
  left_join(airlines, by = "carrier"))
```

```
## # A tibble: 336,776 x 7
##     year month   day  hour tailnum carrier name
##    <int> <int> <int> <dbl> <chr>   <chr>   <chr>
##  1  2013     1     1     5 N14228  UA      United Air Lines Inc.
##  2  2013     1     1     5 N24211  UA      United Air Lines Inc.
##  3  2013     1     1     5 N619AA  AA      American Airlines Inc.
##  4  2013     1     1     5 N804JB  B6      JetBlue Airways
##  5  2013     1     1     6 N668DN  DL      Delta Air Lines Inc.
##  6  2013     1     1     5 N39463  UA      United Air Lines Inc.
##  7  2013     1     1     6 N516JB  B6      JetBlue Airways
##  8  2013     1     1     6 N829AS  EV      ExpressJet Airlines Inc.
##  9  2013     1     1     6 N593JB  B6      JetBlue Airways
## 10  2013     1     1     6 N3ALAA  AA      American Airlines Inc.
## # ... with 336,766 more rows
```

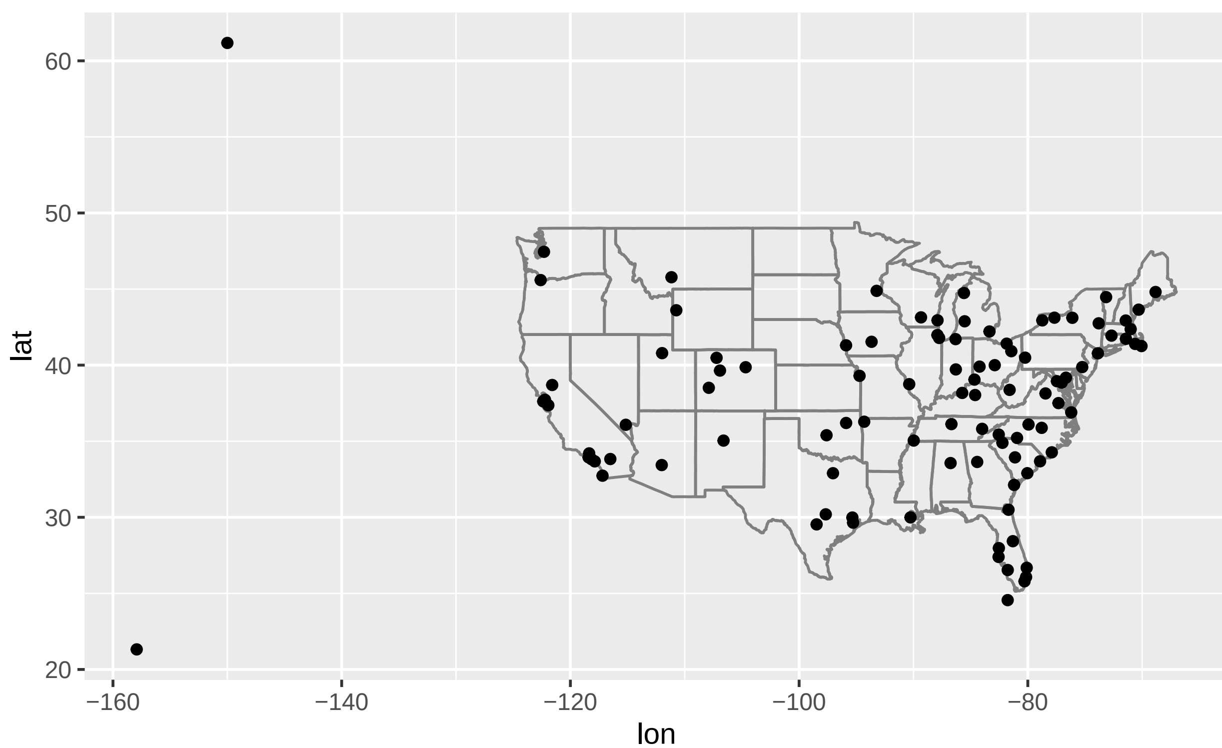
```r
# Produces same output as above, but uses mutate function

flights2 %>%
  select(-origin, -dest ) %>%
  mutate(name = airlines$name[match(carrier, airlines$carrier)])
```

```
## # A tibble: 336,776 x 7
##     year month   day  hour tailnum carrier name
##    <int> <int> <int> <dbl> <chr>   <chr>   <chr>
##  1  2013     1     1     5 N14228  UA      United Air Lines Inc.
##  2  2013     1     1     5 N24211  UA      United Air Lines Inc.
```

```
##  3  2013     1     1     5 N619AA  AA      American Airlines Inc.
##  4  2013     1     1     5 N804JB  B6      JetBlue Airways
##  5  2013     1     1     6 N668DN  DL      Delta Air Lines Inc.
##  6  2013     1     1     5 N39463  UA      United Air Lines Inc.
##  7  2013     1     1     6 N516JB  B6      JetBlue Airways
##  8  2013     1     1     6 N829AS  EV      ExpressJet Airlines Inc.
##  9  2013     1     1     6 N593JB  B6      JetBlue Airways
## 10  2013     1     1     6 N3ALAA  AA      American Airlines Inc.
## # ... with 336,766 more rows
```

4) Compute the average delay by destination, then join on the airports data frame so you can show the spatial distribution of delays.

```
# Base graph

airports %>%
  semi_join(flights, c("faa" = "dest")) %>%
  ggplot(aes(lon, lat)) +
  borders("state") +
  geom_point() +
  coord_quickmap()
```

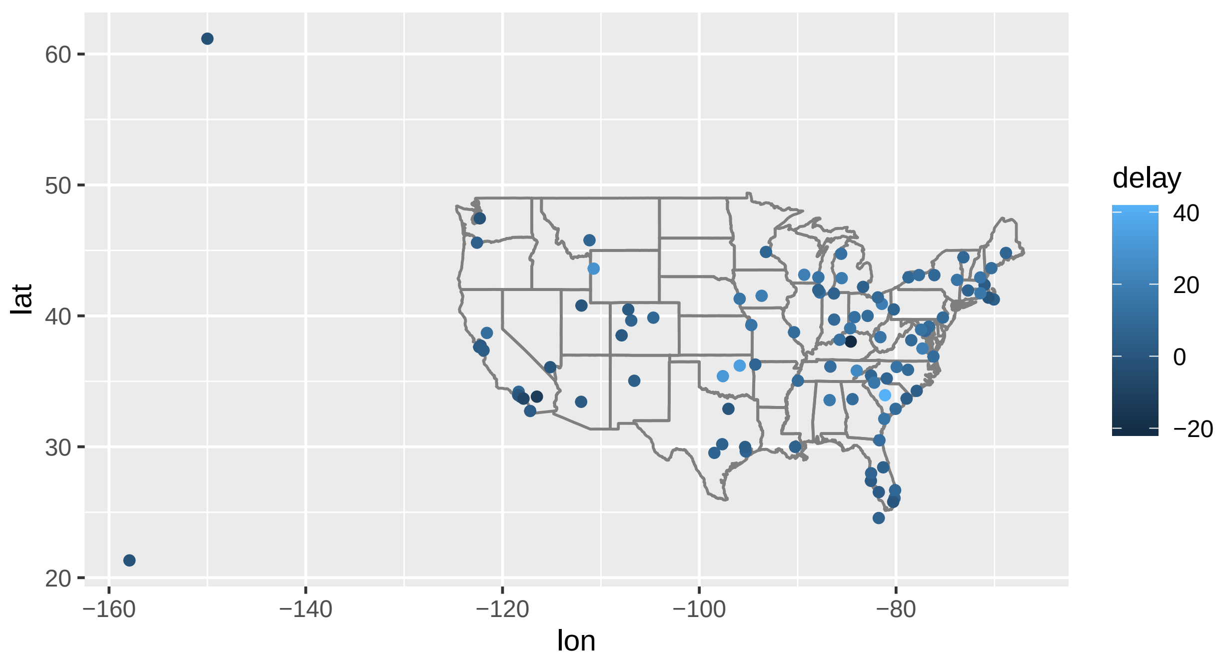



```
# Use color of the points to display the average delay for each airport.

(avg_dest_delays <-
  flights %>%
  group_by(dest) %>%
  summarise(delay = mean(arr_delay, na.rm = TRUE)) %>%
  inner_join(airports, by = c(dest ="faa")))
```

```
## # A tibble: 101 x 9
##    dest  delay name                        lat    lon   alt    tz dst   tzone
##    <chr> <dbl> <chr>                     <dbl>  <dbl> <dbl> <dbl> <chr> <chr>
##  1 ABQ    4.38 Albuquerque Internati~  35.0 -107.    5355    -7 A     America/De~
##  2 ACK    4.85 Nantucket Mem           41.3  -70.1     48    -5 A     America/Ne~
##  3 ALB   14.4  Albany Intl             42.7  -73.8    285    -5 A     America/Ne~
##  4 ANC   -2.5  Ted Stevens Anchorage~  61.2 -150.     152    -9 A     America/An~
```

```
##  5 ATL    11.3  Hartsfield Jackson At~  33.6  -84.4  1026    -5 A     America/Ne~
##  6 AUS     6.02 Austin Bergstrom Intl   30.2  -97.7   542    -6 A     America/Ch~
##  7 AVL     8.00 Asheville Regional Ai~  35.4  -82.5  2165    -5 A     America/Ne~
##  8 BDL     7.05 Bradley Intl            41.9  -72.7   173    -5 A     America/Ne~
##  9 BGR     8.03 Bangor Intl             44.8  -68.8   192    -5 A     America/Ne~
## 10 BHM    16.9  Birmingham Intl         33.6  -86.8   644    -6 A     America/Ch~
## # ... with 91 more rows
```

```
avg_dest_delays %>%
  ggplot(aes(lon, lat, colour = delay)) +
  borders("state") +
  geom_point() +
  coord_quickmap()
```



5) Add the location of the origin and destination (i.e. the lat and lon) to flights.

```
slice_head(airport_locations <- airports %>%
  select(faa, lat, lon))
```

```
## # A tibble: 1 x 3
##   faa     lat   lon
##   <chr> <dbl> <dbl>
## 1 04G    41.1 -80.6
```

```
flights %>%
  head(5) %>%
  select(year:day, hour, origin, dest) %>%
  left_join(
    airport_locations,
    by = c("origin" = "faa")
  ) %>%
  left_join(
    airport_locations,
    by = c("dest" = "faa"),
    suffix = c("_origin", "_dest") # if I do not add the suffix,  dplyr will distinguish
    #the two by adding .x, and .y to the ends of the variable names to solve naming conflicts
)
```

```
## # A tibble: 5 x 10
##    year month   day  hour origin dest  lat_origin lon_origin lat_dest lon_dest
##   <int> <int> <int> <dbl> <chr>  <chr>      <dbl>      <dbl>    <dbl>    <dbl>
## 1  2013     1     1     5 EWR    IAH         40.7      -74.2     30.0    -95.3
## 2  2013     1     1     5 LGA    IAH         40.8      -73.9     30.0    -95.3
## 3  2013     1     1     5 JFK    MIA         40.6      -73.8     25.8    -80.3
## 4  2013     1     1     5 JFK    BQN         40.6      -73.8     NA       NA
## 5  2013     1     1     6 LGA    ATL         40.8      -73.9     33.6    -84.4
```

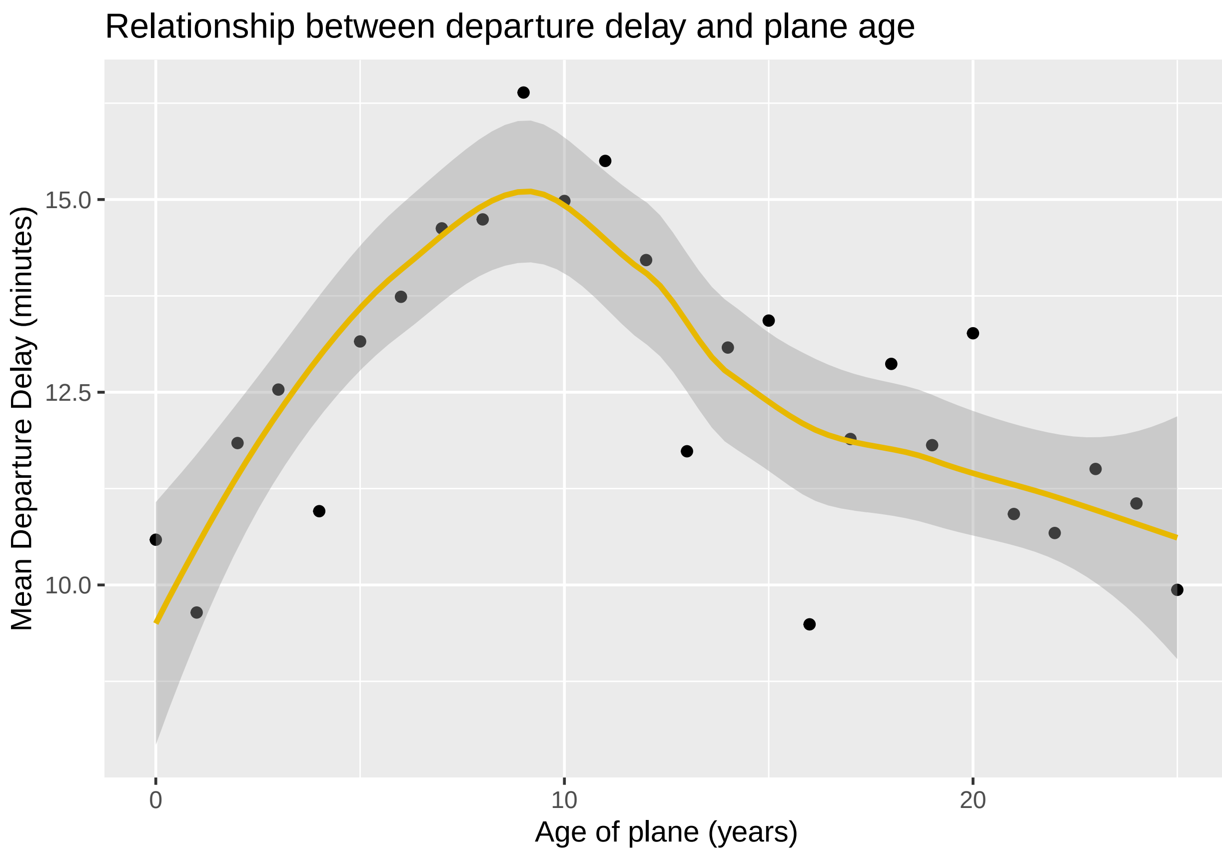
6) Is there a relationship between the age of a plane and its delays (departure and arrival)?

There is an inverted "U" relationship between the age of the plane and its delays. From the plane's manufacturing to about 10 years, delays increase, as expected. After about 10 years, delays tend to decrease; flight times may already include a "buffer" for older planes as it is more likely that an older plane may need more or unexpected maintenance. Hence, there is less of a likelihood that the flight will be actually delayed.

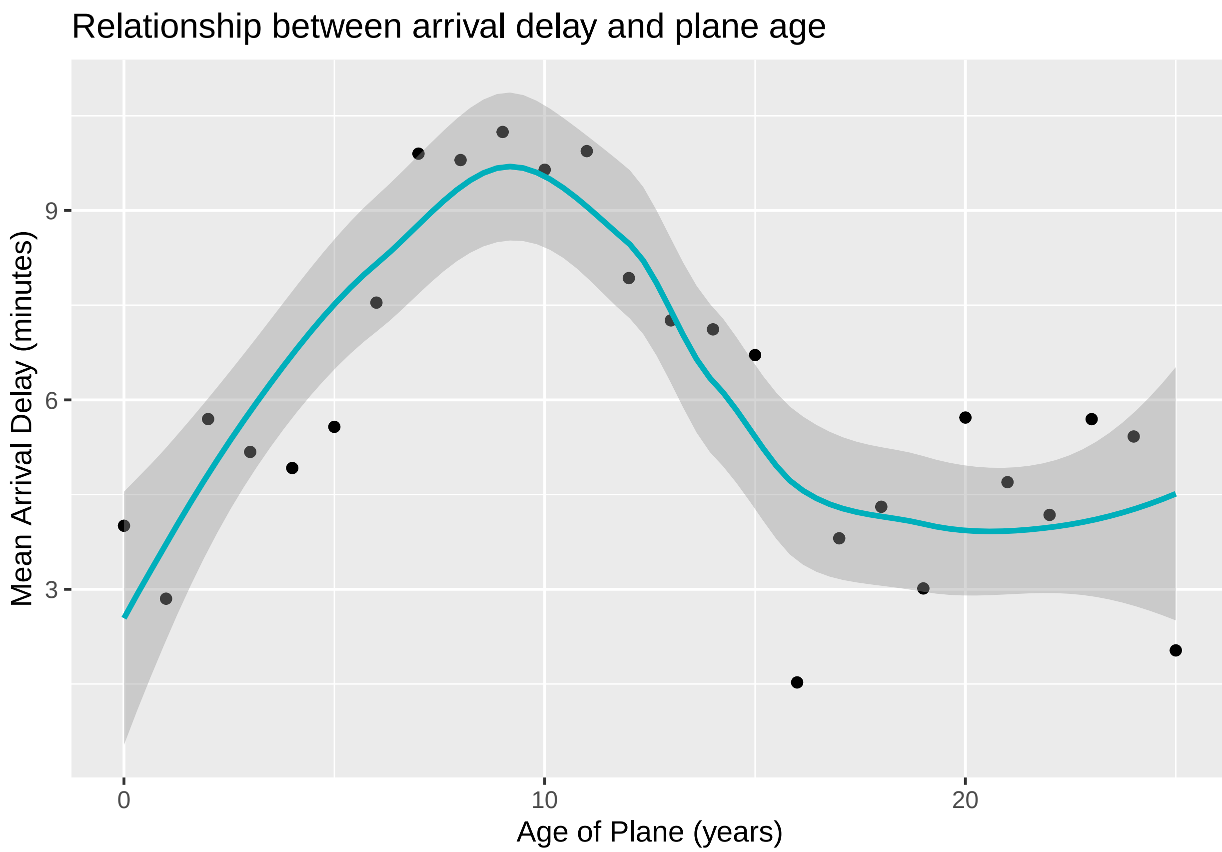
```
plane_cohorts <- inner_join(flights,
                            select(planes, tailnum, plane_year = year),
                            by = "tailnum"
                            ) %>%
  mutate(age = year - plane_year) %>%
  filter(!is.na(age)) %>%
  mutate(age = if_else(age > 25, 25L, age)) %>%
  group_by(age) %>%
  summarise(dep_delay_mean = mean(dep_delay, na.rm = TRUE),
            dep_delay_sd = sd(dep_delay, na.rm = TRUE),
            arr_delay_mean = mean(arr_delay, na.rm = TRUE),
            arr_delay_sd = sd(arr_delay, na.rm = TRUE),
n_arr_delay = sum(!is.na(arr_delay)), # sum of all of the values that are not NA in the dataset
n_dep_delay = sum(!is.na(dep_delay))
)

plane_cohorts %>%
  print(width = Inf)
```

```
## # A tibble: 26 x 7
##      age dep_delay_mean dep_delay_sd arr_delay_mean arr_delay_sd n_arr_delay
##    <int>          <dbl>        <dbl>          <dbl>        <dbl>       <int>
##  1     0          10.6          34.4           4.01         38.5        4611
##  2     1           9.64         31.9           2.85         37.4        7196
##  3     2          11.8          41.8           5.70         46.8        6008
##  4     3          12.5          37.5           5.18         41.9        3771
##  5     4          11.0          35.5           4.92         39.7        6572
##  6     5          13.2          39.6           5.57         43.9       17731
##  7     6          13.7          41.4           7.54         45.2       15142
##  8     7          14.6          41.3           9.90         45.1       12998
##  9     8          14.7          41.5           9.80         45.4       14064
## 10     9          16.4          44.2          10.2          48.0       15273
##    n_dep_delay
##          <int>
##  1        4621
##  2        7214
##  3        6017
##  4        3777
##  5        6584
##  6       17809
##  7       15207
##  8       13030
##  9       14112
## 10       15339
## # ... with 16 more rows
```

```r
# Graph relationship between departure delay and plane age

ggplot(plane_cohorts, aes(x = age, y = dep_delay_mean)) +
  geom_point() +
  geom_smooth(color = "#E7B800") +
  scale_x_continuous("Age of plane (years)", breaks = seq(0,30, by = 10)) +
  scale_y_continuous("Mean Departure Delay (minutes)") +
  ggtitle("Relationship between departure delay and plane age")
```



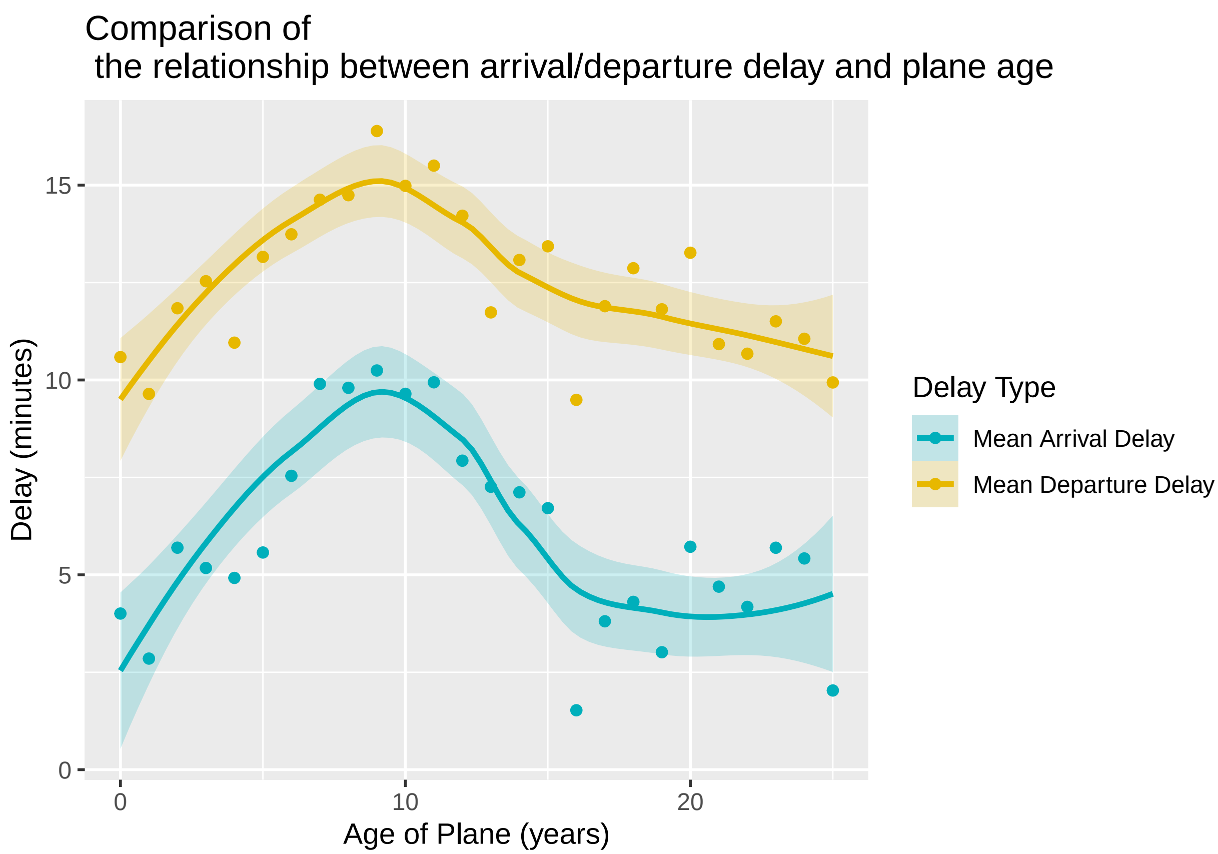
```r
# Graph relationship between arrival delay and plane age

ggplot(plane_cohorts, aes(age, arr_delay_mean)) +
  geom_point() +
  geom_smooth(color = "#00AFBB") +
  scale_x_continuous("Age of Plane (years)", breaks = seq(0,30, by = 10)) +
  scale_y_continuous("Mean Arrival Delay (minutes)") +
  ggtitle("Relationship between arrival delay and plane age")
```



```r
# Graph relationship between arrival/departure delay comparison and plane age

plane_cohorts1 <- plane_cohorts %>%
  select(arr_delay_mean, dep_delay_mean, age) %>%
  pivot_longer(c(arr_delay_mean, dep_delay_mean), names_to = "delay_type" , values_to = "delay_time")

plane_cohorts1 %>%
  head(5)
```

```
## # A tibble: 5 x 3
##     age delay_type     delay_time
##   <int> <chr>               <dbl>
## 1     0 arr_delay_mean       4.01
## 2     0 dep_delay_mean      10.6 
## 3     1 arr_delay_mean       2.85
## 4     1 dep_delay_mean       9.64
## 5     2 arr_delay_mean       5.70
```

```r
b <- ggplot(plane_cohorts1, aes(x = age, y = delay_time))

b + geom_point(aes(color = delay_type)) +
  geom_smooth(aes(color = delay_type, fill = delay_type), alpha = 0.2) +

  scale_color_manual(name = "Delay Type", labels = c("Mean Arrival Delay",
                                                      "Mean Departure Delay"),
                     values = c("#00AFBB", "#E7B800")) +
  scale_fill_manual(name = "Delay Type", labels = c("Mean Arrival Delay",
                                                     "Mean Departure Delay"),
                    values = c("#00AFBB", "#E7B800")) +

    scale_x_continuous("Age of Plane (years)", breaks = seq(0,30, by = 10)) +
      scale_y_continuous("Delay (minutes)") +
   ggtitle("Comparison of\n the relationship between arrival/departure delay and plane age")
```



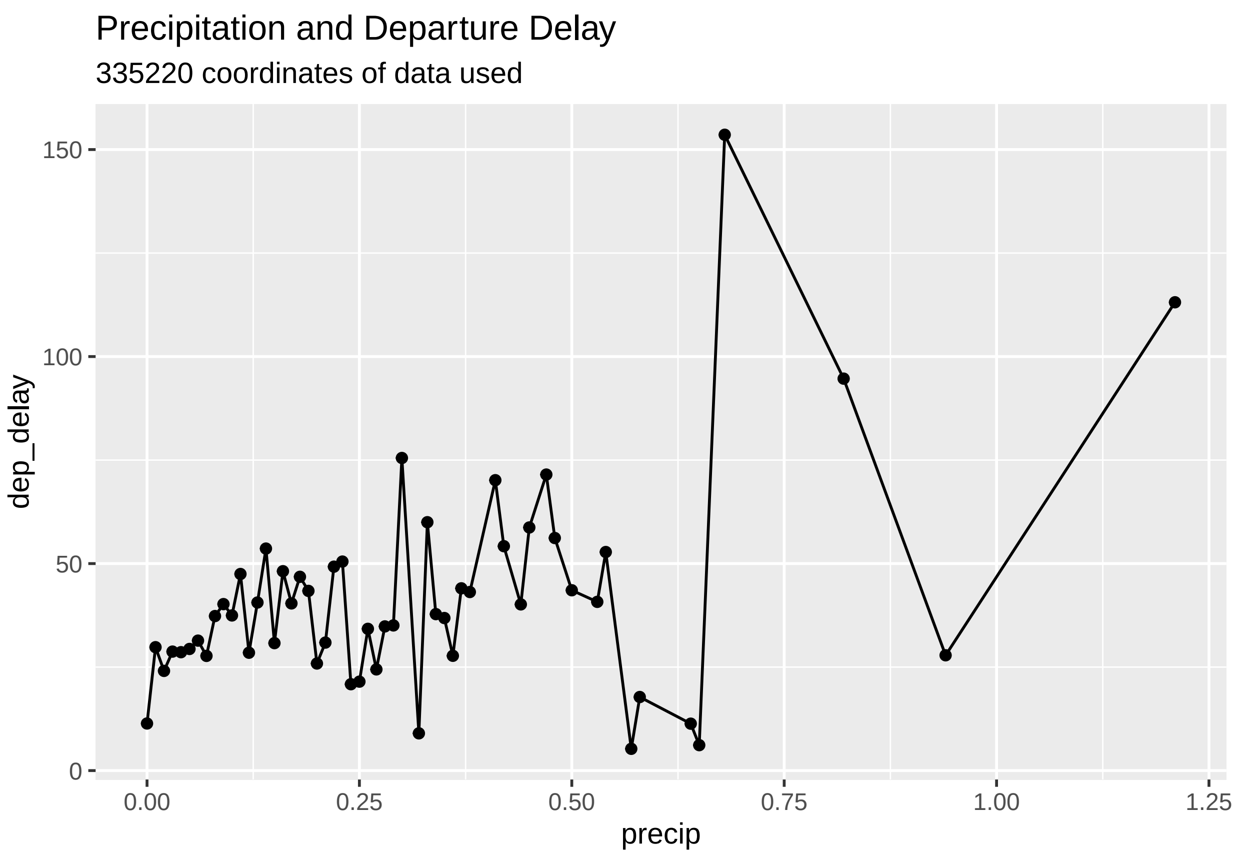
7) What weather conditions make it more likely to see a departure delay?

Visually, it seems like there is evidence that there is a strong negative linear correlation between departure delay and visibility than with precipitation. However, when calculating the correlation, the variables "precip" and "visib" have a Pearson correlation coefficent of 0.09 and -0.09, respectively. Looking at the heat map, the weather conditions listed is this study suggest a, if any, weak correlation with /departure delays.

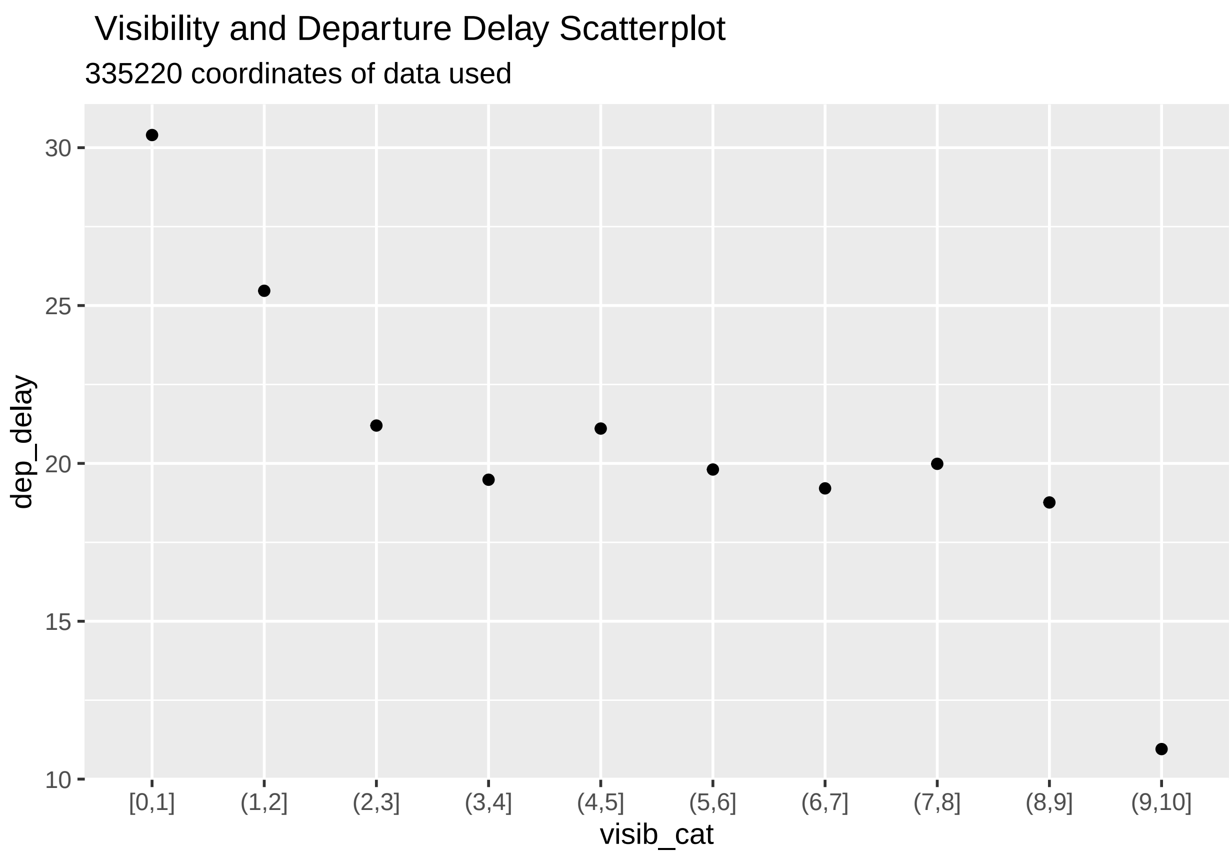
```
flight_weather <-
  flights %>%
  inner_join(weather, by = c("origin", "year", "month", "day", "hour"))

dim(flight_weather)[1] #number of observations
```

```
## [1] 335220
```

```
flight_weather %>%
  group_by(precip) %>%
  summarise(dep_delay = mean(dep_delay, na.rm = TRUE)) %>%
  ggplot(aes(x = precip, y = dep_delay)) +
  geom_line() +
  geom_point() +
  labs(title = "Precipitation and Departure Delay", subtitle = "335220 coordinates of data used")
```



```r
flight_weather %>%
  ungroup() %>%
  mutate(visib_cat = cut_interval(visib, n = 10)) %>%
  group_by(visib_cat) %>%
  summarise(dep_delay = mean(dep_delay, na.rm = TRUE)) %>%
  ggplot(aes(x = visib_cat, y = dep_delay)) +
   geom_point() +
  labs(title = " Visibility and Departure Delay Scatterplot" , subtitle = "335220 coordinates of data us
```



```r
# Pearson correlation test for dep_delay and visib

cor(flight_weather$visib, flight_weather$dep_delay,
    use = "complete.obs",
         method = "pearson")
```

```
## [1] -0.09411769
```

```r
# Save visib and dep_delay as separate dataframe to df1.
# Then, counts the numbers of rows for comparison.  Data frame contains all values including NA.

nrow(df1 <- flight_weather %>%
  select(visib, dep_delay))
```

```
## [1] 335220
```

```r
# Remove na in r - remove rows - na.omit function.
# Then, counts the number of rows, i.e. how many pairs are used in the Pearson correlation test.
nrow(na.omit(df1))
```

```
## [1] 326993
```

```r
# Pearson correlation test for dep_delay and precip

cor(flight_weather$precip, flight_weather$dep_delay,
    use = "complete.obs",
         method = "pearson")
```

```
## [1] 0.09040014
```

```r
# Save precip and dep_delay as separate dataframe to df1.
# Then, counts the numbers of rows for comparison.  Data frame contains all values including NA.

nrow(df2 <- flight_weather %>%
  select(precip, dep_delay))
```

```
## [1] 335220
```

```r
# Remove na in r - remove rows - na.omit function.
# Then, counts the number of rows, i.e. how many pairs are used in the Pearson correlation test.
nrow(na.omit(df2))
```

```
## [1] 326993
```

```r
library(reshape2)

# number of observations - eliminate rows with NA
nrow(na.omit(flight_weather %>%
               select(precip, dep_delay)))
```

```
## [1] 326993
```

```r
flight_weather_corrmap <- na.omit(flight_weather %>%
  select(-year:-sched_dep_time, -arr_time:-time_hour.x, -time_hour.y))

# correlation matrix - table
cormat <- round(cor(flight_weather_corrmap), 2)

head(cormat)
```

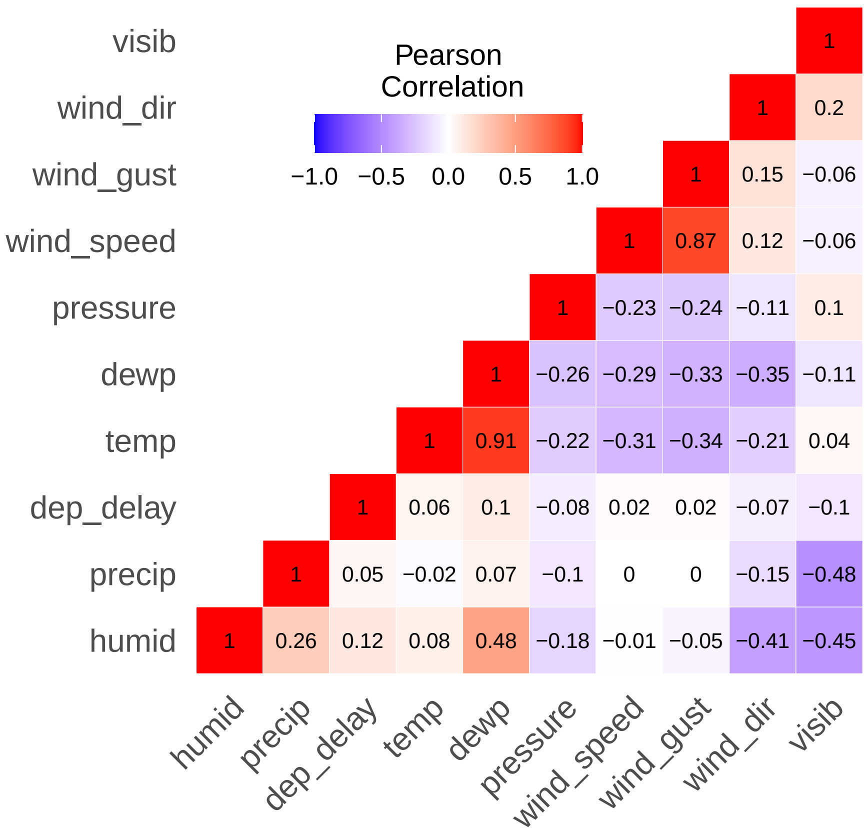
```
##            dep_delay  temp  dewp humid wind_dir wind_speed wind_gust precip
## dep_delay       1.00  0.06  0.10  0.12    -0.07       0.02      0.02   0.05
## temp            0.06  1.00  0.91  0.08    -0.21      -0.31     -0.34  -0.02
## dewp            0.10  0.91  1.00  0.48    -0.35      -0.29     -0.33   0.07
## humid           0.12  0.08  0.48  1.00    -0.41      -0.01     -0.05   0.26
## wind_dir       -0.07 -0.21 -0.35 -0.41     1.00       0.12      0.15  -0.15
## wind_speed      0.02 -0.31 -0.29 -0.01     0.12       1.00      0.87   0.00
##            pressure visib
## dep_delay     -0.08 -0.10
## temp          -0.22  0.04
## dewp          -0.26 -0.11
## humid         -0.18 -0.45
## wind_dir      -0.11  0.20
## wind_speed    -0.23 -0.06
```

```r
# correlation matrix - table triangle
get_upper_tri <- function(cormat){
  cormat[lower.tri(cormat)] <- NA
  return(cormat)
}

upper_tri <- get_upper_tri(cormat)

upper_tri
```

```
##            dep_delay temp dewp humid wind_dir wind_speed wind_gust precip
## dep_delay          1 0.06 0.10  0.12    -0.07       0.02      0.02   0.05
## temp              NA 1.00 0.91  0.08    -0.21      -0.31     -0.34  -0.02
## dewp              NA   NA 1.00  0.48    -0.35      -0.29     -0.33   0.07
## humid             NA   NA   NA  1.00    -0.41      -0.01     -0.05   0.26
## wind_dir          NA   NA   NA    NA     1.00       0.12      0.15  -0.15
## wind_speed        NA   NA   NA    NA       NA       1.00      0.87   0.00
## wind_gust         NA   NA   NA    NA       NA         NA      1.00   0.00
## precip            NA   NA   NA    NA       NA         NA        NA   1.00
## pressure          NA   NA   NA    NA       NA         NA        NA     NA
## visib             NA   NA   NA    NA       NA         NA        NA     NA
##            pressure visib
## dep_delay     -0.08 -0.10
## temp          -0.22  0.04
```

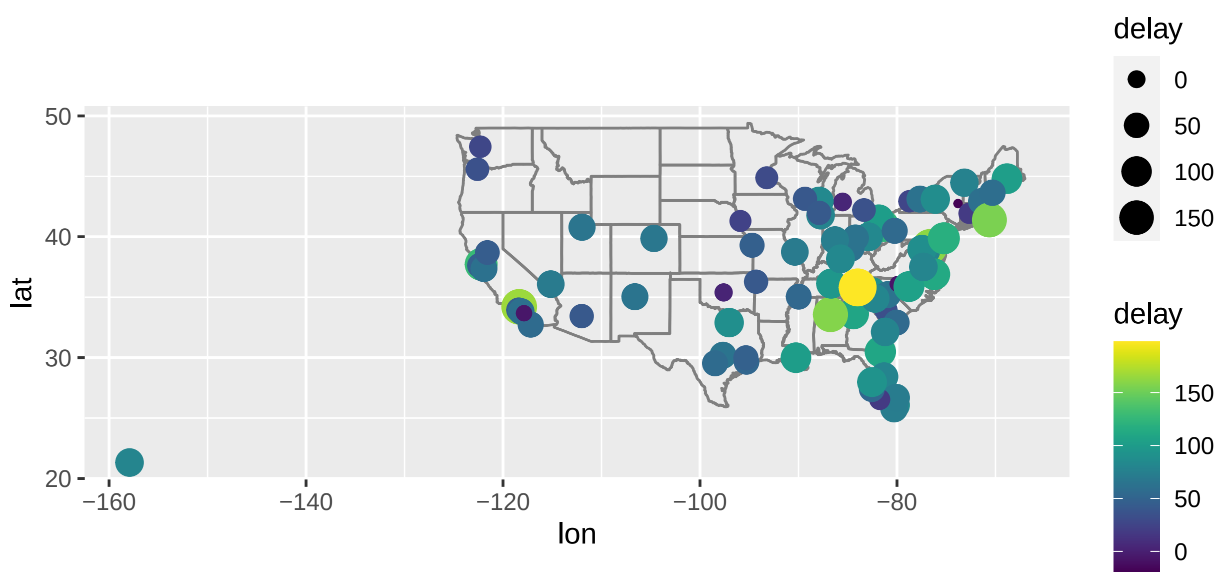
```
## dewp          -0.26 -0.11
## humid         -0.18 -0.45
## wind_dir      -0.11  0.20
## wind_speed    -0.23 -0.06
## wind_gust     -0.24 -0.06
## precip        -0.10 -0.48
## pressure       1.00  0.10
## visib            NA  1.00
```

```r
# correlation matrix

reorder_cormat <- function(cormat){
  dd <- as.dist((1-cormat)/2)
  hc <- hclust(dd)
  cormat <- cormat[hc$order, hc$order]
}

cormat <- reorder_cormat(cormat)
upper_tri <- get_upper_tri(cormat)

melted_cormat <- melt(upper_tri, na.rm =TRUE)

#melted_cormat

ggheatmap <- ggplot(melted_cormat, aes(Var2, Var1, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0,
                       limit = c(-1,1), space = "Lab", name = "Pearson\n Correlation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12, hjust = 1),
        axis.text.y = element_text(size = 12)) +
  coord_fixed()

ggheatmap +
  geom_text(aes(Var2, Var1, label =value), color = "black", size = 2.75) +
  theme(
    axis.title = element_blank(),
    panel.grid.major = element_blank(),
    panel.border = element_blank(),
    panel.background = element_blank(),
    axis.ticks = element_blank(),
    legend.justification = c(1,0),
    legend.position = c(0.6, 0.7),
    legend.direction = "horizontal") +
  guides(fill = guide_colorbar(barwidth = 7, barheight = 1, title.position = "top",
                               title.hjust = 0.5))
```


Pearson
Correlation
−1.0
−0.5
0.0
0.5
1.0
visib
wind_dir
wind_gust
wind_speed
pressure
dewp
temp
dep_delay
precip
humid
1
1 0.2
1 0.15 −0.06
1 0.87 0.12 −0.06
1 −0.23 −0.24 −0.11 0.1
1 −0.26 −0.29 −0.33 −0.35 −0.11
1 0.91 −0.22 −0.31 −0.34 −0.21 0.04
1 0.06 0.1 −0.08 0.02 0.02 −0.07 −0.1
1 0.05 −0.02 0.07 −0.1 0 0 −0.15 −0.48
1 0.26 0.12 0.08 0.48 −0.18 −0.01 −0.05 −0.41 −0.45
humid
precip
dep_delay
temp
dewp
pressure
wind_speed
wind_gust
wind_dir
visib

8) What happened on June 13 2013? Display the spatial pattern of delays, and then use Google to cross-reference with the weather. Large storms called *derechos* occurred in the Southeast and Midwest on June 13, 2013.

```
library(viridis)
flights %>%
  filter(year == 2013, month == 6, day == 13) %>%
  group_by(dest) %>%
  summarise(delay = mean(arr_delay, na.rm = TRUE)) %>%
  inner_join(airports, by = c("dest" = "faa")) %>%
  ggplot(aes(y = lat, x = lon, size = delay, colour = delay)) +
  borders("state") +
  geom_point() +
  coord_quickmap() +
  scale_colour_viridis()
```



Filtering Joins

9) Practice problem with semi_join.

```
# Sample data: top ten most popular destinations

(top_dest <- flights %>%
  count(dest, sort = TRUE) %>%
  head(10))
```

```
## # A tibble: 10 x 2
##    dest      n
##    <chr> <int>
##  1 ORD   17283
##  2 ATL   17215
##  3 LAX   16174
##  4 BOS   15508
##  5 MCO   14082
##  6 CLT   14064
##  7 SFO   13331
##  8 FLL   12055
##  9 MIA   11728
## 10 DCA    9705
```

```
# semi-join connects the two tables like a mutating join,
# but instead of adding new columns,
# only keeps the rows in x that have a match in y

flights %>%
  semi_join(top_dest)
```

```
## # A tibble: 141,145 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
##  1  2013     1     1      542            540         2      923            850
##  2  2013     1     1      554            600        -6      812            837
##  3  2013     1     1      554            558        -4      740            728
##  4  2013     1     1      555            600        -5      913            854
##  5  2013     1     1      557            600        -3      838            846
##  6  2013     1     1      558            600        -2      753            745
##  7  2013     1     1      558            600        -2      924            917
##  8  2013     1     1      558            600        -2      923            937
##  9  2013     1     1      559            559         0      702            706
## 10  2013     1     1      600            600         0      851            858
## # ... with 141,135 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```

10) Filter flights to only show flights with planes that have flown at least 100 flights.

```
# First, I find all planes that have flown at least 100 flights. I need to filter flights that are miss

planes_gte100 <- flights %>%
  filter(!is.na(tailnum)) %>%
  group_by(tailnum) %>%
```

```r
  count() %>%
  filter(n>100)

# Now, I will semi join the data frame of planes that have flown at least 100 flights to the data frame

flights %>%
  semi_join(planes_gte100, by = "tailnum")
```

```
## # A tibble: 226,690 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
##  1  2013     1     1      517            515         2      830            819
##  2  2013     1     1      533            529         4      850            830
##  3  2013     1     1      544            545        -1     1004           1022
##  4  2013     1     1      554            558        -4      740            728
##  5  2013     1     1      555            600        -5      913            854
##  6  2013     1     1      557            600        -3      709            723
##  7  2013     1     1      557            600        -3      838            846
##  8  2013     1     1      558            600        -2      849            851
##  9  2013     1     1      558            600        -2      853            856
## 10  2013     1     1      558            600        -2      923            937
## # ... with 226,680 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```