Patricia Reynoso

October 11th, 2018

Project scope:

Flights data used in this analysis:

a. Domestic commercial flights
b. Origin: OAK, SFO or SJC.
c. Operated from January 1st to July 31st 2018

Project content

Part I.

Answers to the questions in exercises 4.2, 4.3, 4.4, 4.5, 4.6, 4.7 from *Modern Data Science with R.*

Part II.

A. Merged Table from sfoflights18 and new weather data

B. Answers to questions in exercises 4.6, 4.7 from *Modern Data Science with R.*

Part III. Appendices

Appendix I. **Tables**

Appendix II. **Code**

Part I.

Answers to the questions in exercises 4.2, 4.3, 4.4, 4.5, 4.6, 4.7 from *Modern Data Science with R*.

4.2

What month had the highest proportion of cancelled flights?

Answer: March had the highest proportion of cancelled flights with 2% of 23,055 flights total.

What month had the lowest?

Answer: February had the lowest proportion of cancelled flights with 0.7% of 20,142 flights total

---

Interpret any seasonal patterns.

Answer: The results are consistent with the precipitation data. March was the rainiest month (16 days) of all which may explain the highest cancellation rate. On the other hand, February had only 4 days of rain and the lowest flights volume which explains the lowest cancellation rate.

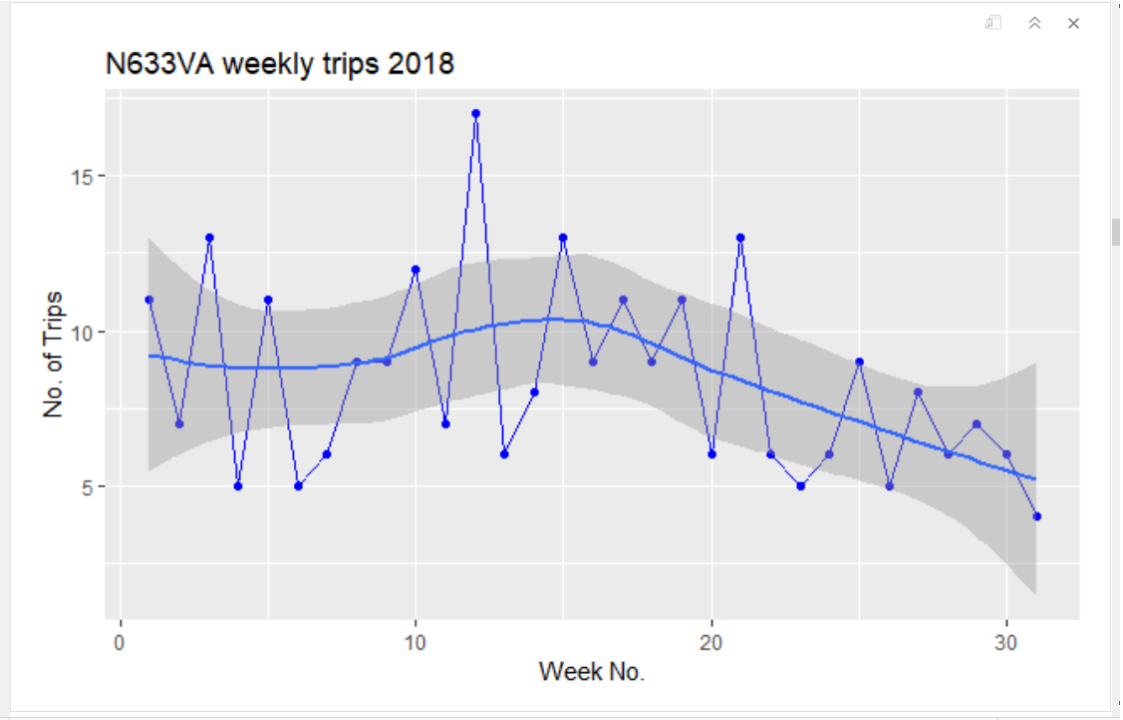
4.3

What plane (specified by the tailnum variable) traveled the most times from Bay Area airports from January 1st to July 31st 2018?

Answer: The plane that traveled the most times was N633VA with a total of 260 trips from January 1st to July 31st 2018.

Plot the number of trips per week over the year.

Answer: The average number of trips per week for this aircraft is about 9. There was a peak week (week 12) when this plane made about 20 trips. As the year progressed, the number of trips trended downwards.



4.4

What is the oldest plane that flew from Bay Area airports from from January 1st to July 31st 2018?

Answer: The oldest plane that flew from January 1st to July 31st 2018 is N990JB, manufactured in 1977.

How many planes that flew from any of the three airports in the Bay Area are included in the airplanes table?

Answer: 3660 planes that flew from the Bay Area are included in the airplanes table. Only 64 planes were not included.

4.5

How many planes have a missing date of manufacture?

Answer: 86 planes have a missing date of manufacture.

What are the five most common manufacturers?

Answer: The five most common manufacturers are:

MANUFACTURER	PLANES
BOEING	2169
AIRBUS	1095
BOMBARDIER	212
EMBRAER	113
MCDONNEL DOUGLAS	45

Has the distribution of manufacturer changed over time as reflected by the airplanes flying from the Bay Area in 2018?

Answer: The newest planes are coming from the biggest manufacturers. The smallest manufacturers have planes which are older. That indicates that the industry is trending towards buying the planes from the big companies and probably not buying from small companies anymore.

MANUFACTURER	Avg. Year of MFR.
EMBRAER	2016
AIRBUS	2007
BOEING	2006
BOMBARDIER	2004
OTHER	2003
MCDONNEL DOUGLAS	1996

Part II.

A. Merged Table from sfoflights18 and new weather data

The strategy used to merge flight data with weather data, was first creating one table of each type(flight and weather) per origin, and manipulating some of the existing variables to create one variable that could be used to link each pair of tables.

After the resulting three tables with weather data and flights by origin were generated, they were binded into one table with weather and flight data for the three origins (OAK,SJC,SFO).

The merged table has the same number of observations as the original sfoflights18 table.

In the original Wheather table, there were multiple weather data points per hour, some with missing values in most of the varibles. The data was wrangled to keep one observation per hour with values in at least most of the relevant variables.

SJC and OAK had their hourly weather observations recorded at the minute 53 of every hour, whereas SFO had them at the 56'.

The variable *time_minute* was created in both tables to serve as the link for the joining operation. the right weather data gets merged to the flights data.

For the creation of the *time_minute* variable, some string manipulation and date manipulation functions were used.

The resulting table *BAflightWeather* contains 33 variables:

a. 21 flight variables (names are in upper case)

b. 10 weather variables (names are in lower case)

c. 2 new variables (*hour* and *time_minute*) served as bridges for the tables merging.

Note that the new weather-flights merged table BAflightWeather* has 163,056 observations, exactly the same as the original *sfoflights18* flights table.*

```
glimpse(BAflightWeather)
```

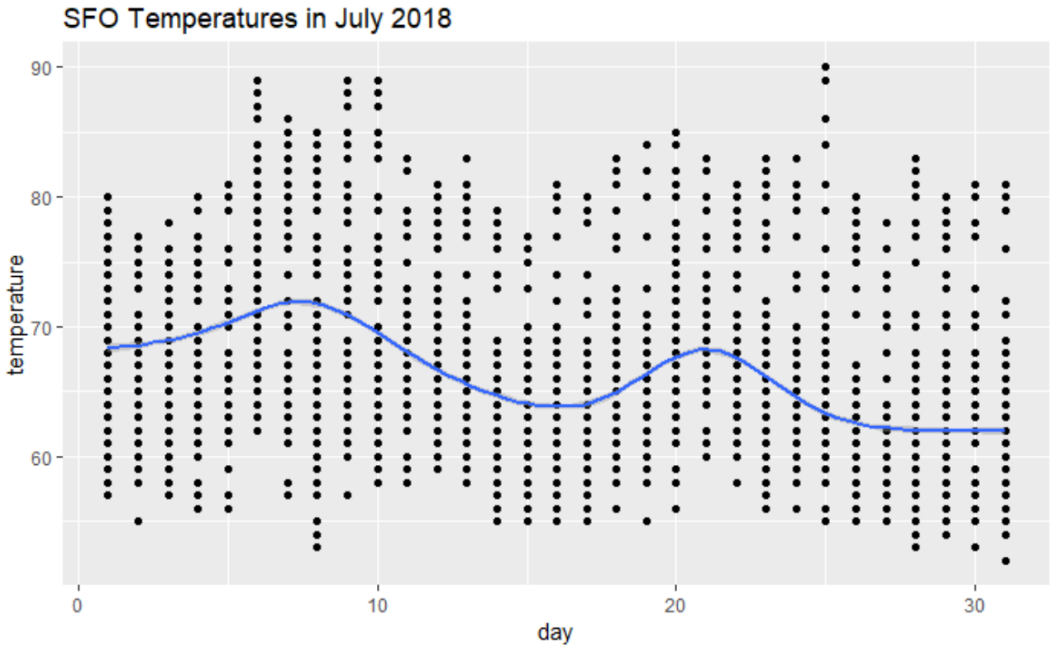
```
## Observations: 163,056
## Variables: 33
## $ YEAR               <int> 2018, 2018, 2018, 2018, 2018, 2018, 2018, 20...
## $ MONTH              <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ DAY_OF_MONTH       <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2,...
## $ DAY_OF_WEEK        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2,...
## $ FL_DATE            <chr> "1/1/2018", "1/1/2018", "1/1/2018", "1/1/201...
## $ OP_UNIQUE_CARRIER  <chr> "AS", "AS", "AS", "AS", "AS", "AS", "AS", "A...
## $ TAIL_NUM           <chr> "N273AK", "N565AS", "N274AK", "N526AS", "N58...
## $ OP_CARRIER_FL_NUM  <int> 335, 345, 353, 569, 811, 845, 877, 915, 917,...
## $ ORIGIN_AIRPORT_ID  <int> 13796, 13796, 13796, 13796, 13796, 13796, 13...
## $ ORIGIN             <chr> "OAK", "OAK", "OAK", "OAK", "OAK", "OAK", "O...
## $ DEST_AIRPORT_ID    <int> 14747, 14747, 14747, 14747, 13830, 12758, 12...
## $ DEST               <chr> "SEA", "SEA", "SEA", "SEA", "OGG", "KOA", "L...
## $ CRS_DEP_TIME       <chr> "1237", "605", "905", "1847", "700", "815", ...
## $ DEP_TIME           <int> 1229, 708, 858, 1909, 649, 830, 733, 927, 19...
## $ DEP_DELAY          <int> -8, 63, -7, 22, -11, 15, 14, -10, -9, 1, -6,...
## $ CRS_ARR_TIME       <int> 1436, 815, 1110, 2052, 1044, 1158, 1110, 113...
## $ ARR_TIME           <int> 1426, 911, 1101, 2104, 1004, 1152, 1101, 110...
## $ ARR_DELAY          <int> -10, 56, -9, 12, -40, -6, -9, -26, -37, -16,...
## $ CANCELLED          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ AIR_TIME           <int> 97, 101, 94, 92, 296, 296, 312, 82, 77, 87, ...
## $ DISTANCE           <int> 672, 672, 672, 672, 2349, 2378, 2457, 543, 5...
## $ hour               <chr> "12", "6", "9", "18", "7", "8", "7", "9", "2...
## $ time_minute        <dttm> 2018-01-01 12:53:00, 2018-01-01 06:53:00, 2...
## $ origin             <chr> "OAK", "OAK", "OAK", "OAK", "OAK", "OAK", "O...
## $ visib              <dbl> 6, 5, 4, 10, 5, 5, 5, 4, 10, 8, 10, 8, 7, 8,...
## $ temp               <int> 54, 41, 50, 51, 40, 45, 40, 50, 51, 57, 47, ...
## $ dewp               <int> 48, 41, 50, 49, 40, 45, 40, 50, 48, 49, 44, ...
## $ humid              <int> 80, 100, 100, 92, 100, 100, 100, 100, 89, 74...
## $ wind_speed         <int> 6, 0, 5, 0, 6, 0, 6, 5, 0, 3, 0, 0, 3, 0, 0,...
## $ wind_dir           <int> 270, 0, 200, 0, 30, 0, 30, 200, 0, 340, 0, 0...
## $ pressure           <dbl> 30.11, 30.15, 30.18, 30.07, 30.15, 30.16, 30...
## $ precip             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ wind_gust          <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
```

B. Answers to questions in exercises 4.6, 4.7 from *Modern Data Science with R.*

4.6

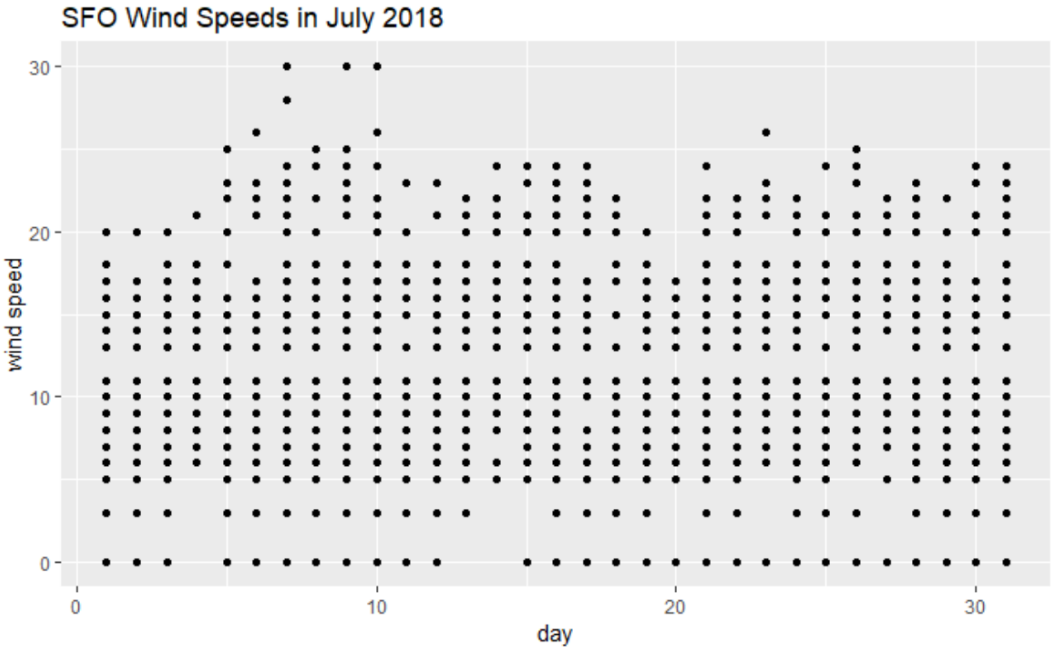
What is the distribution of temperature in July, 2018?

Answer: The average temperature in the Bay Area was stable and within the range between 60F and 72F throughout July 2018.



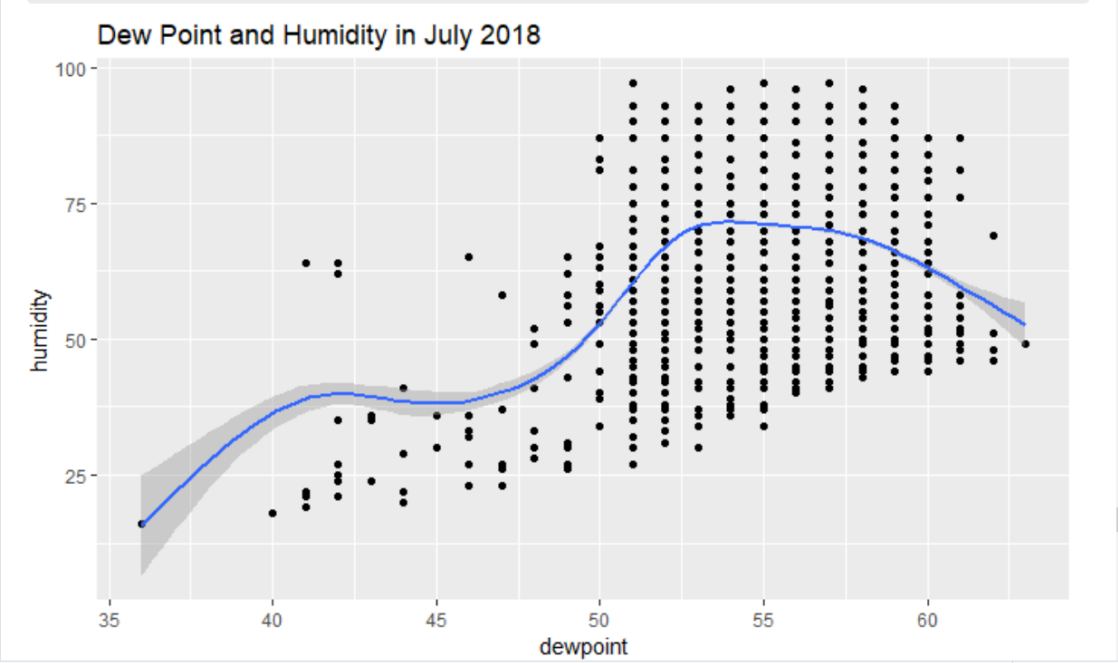
Identify any important outliers in terms of the wind speed variable.

Answer: There are a few outliers present in the days 5 to 10 in July. There were seven measurements that indicated wind speeds greater than 25MPH and up to 30MPH.



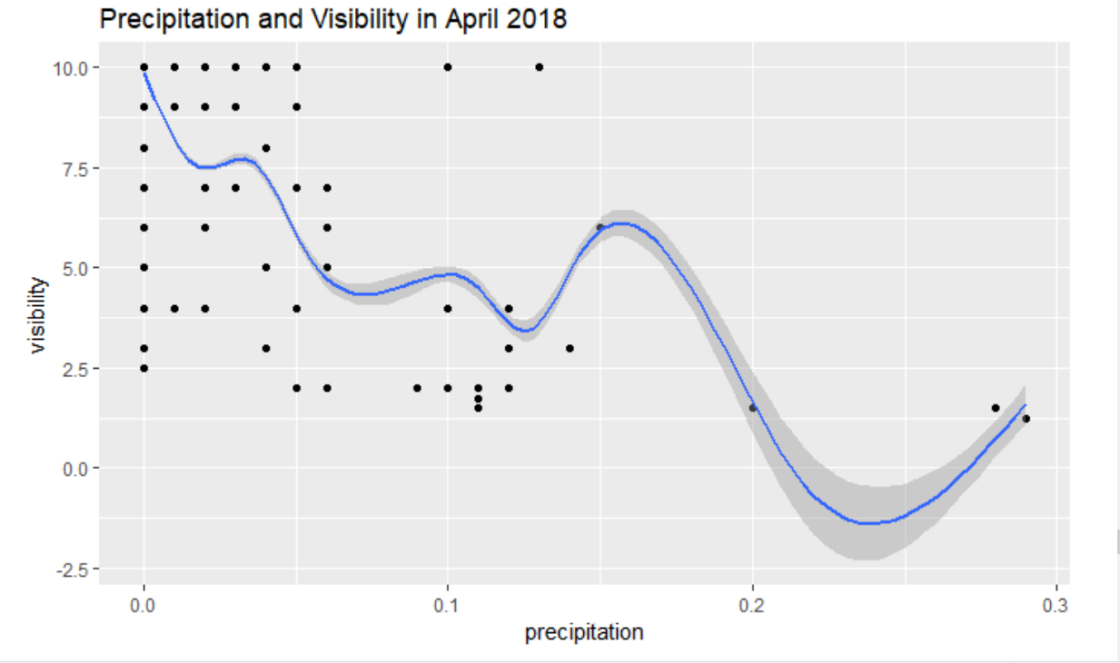
What is the relationship between dewp and humid?

Answer: There is a strong positive correlation between dew point and humidity up to a point and then it turns to a negative correlation. The inflection point occurs when the humidity reaches about 70%.



What is the relationship between precip and visib?

Answer: For this question, data from the month of April was used because there was no precipitation in July. For the month of April 2018, precipitation and visibility had a strong negative correlation which makes sense since visibility is hindered by rain.



4.7

On how many days was there precipitation in the San Francisco bay area in 2018?

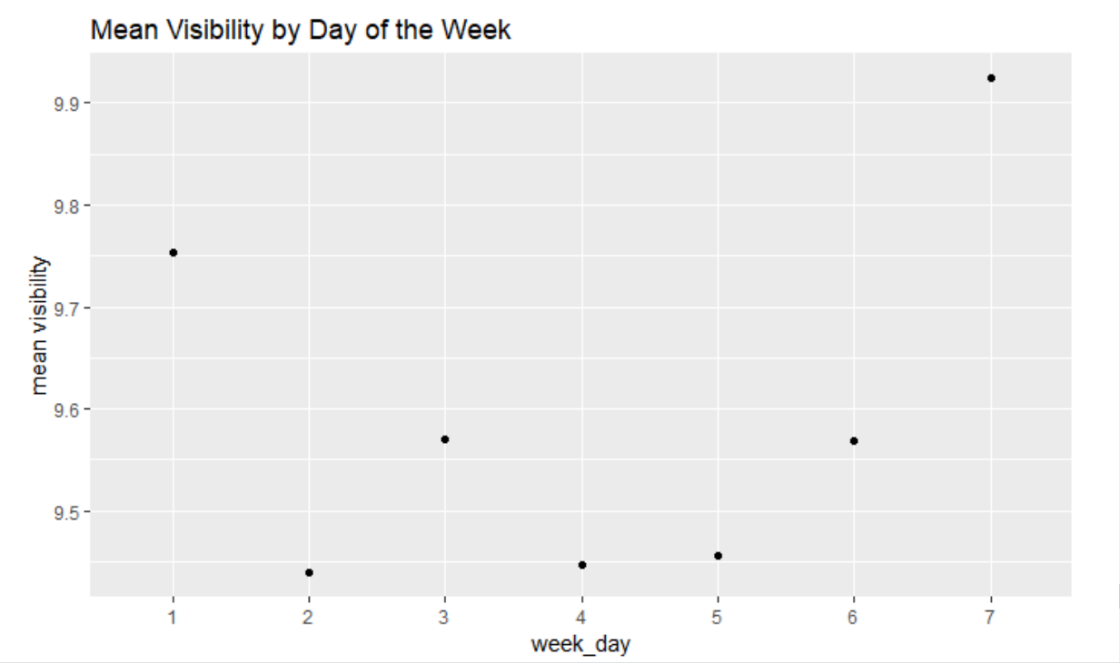
Answer: There was precipitation on 36 days from January 1st to July 31st 2018.

Were there differences in the mean visibility (visib) based on the day of the week?

Answer: There were no significant differences in the mean visibility values by day of the week.

week_day <chr>	mean <dbl>	sd <dbl>	N <int>
1	9.752717	1.1805645	22577
2	9.439483	1.6356283	24788
3	9.569760	1.1939650	24120
4	9.446800	1.5672294	23608
5	9.455470	1.5637999	23963
6	9.568969	1.4413644	24112
7	9.924333	0.4950504	18981

7 rows

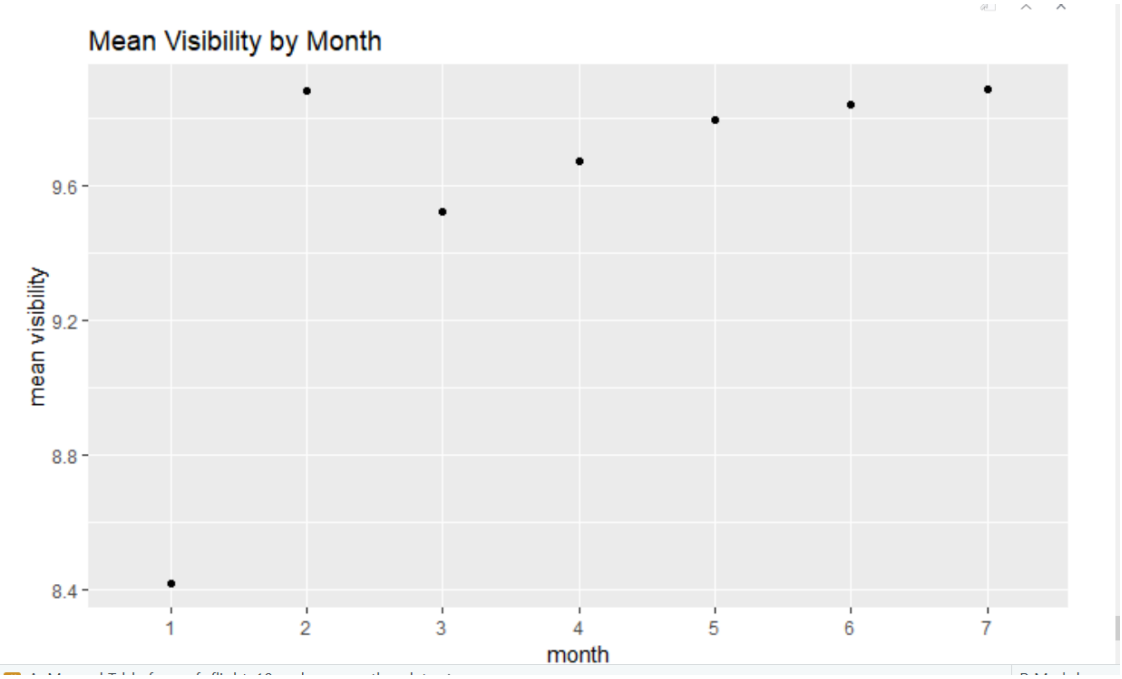


Were there differences in the mean visibility (visib) based on the month of the year?

Answer: January has a significant lower visibility value than other months. This makes sense since January was the second rainiest month of the season. But it seems that there might have been additional climatological factors, possibly fog which is not included in this analysis, that negatively affected the visibility in january 2018.

month <chr>	mean <dbl>	sd <dbl>	N <int>
1	8.420439	2.4872244	22470
2	9.884279	0.6863389	20011
3	9.525474	1.4756960	22952
4	9.675978	1.2040047	22929
5	9.796200	0.8341097	24019
6	9.841795	0.6972597	24403
7	9.886487	0.6228402	25365

7 rows



Part III. Appendices

Appendix I. Tables

Table 1. Source of data for the construction of the data set *sfoflights18* which contains all the domestic flights operated between January 1st and July 31st 2018 in the SFO Bay Area: Oakland, San Francisco and San Jose International Airports.

flights	Description
DB	Monthly reporting carrier on-time performance
Source	Bureau of Transportation Statistics (BTS)
Link	https://www.transtats.bts.gov/Tables.asp?DB_ID=120&DB_Name=Airline%20On-Time%20Performance%20Data&DB_Short_Name=On-Time

Table 2. Source of data for the construction of the data set *airplanes* which contains all registered aircrafts with the Federal Aviation Administration up to October 3rd 2018.

planes	Description
DB	Aircraft Registration Database.Tables: *MASTER*, *ACFTREF*
Source	Federal Aviation Administration (FAA)
Link	https://www.faa.gov/licenses_certificates/aircraft_certification/aircraft_registry/releasable_aircraft_download/

Table 3. Source of data for the construction of the data set *weatherBA* which contains weather information for the three stations corresponding to the three airports subject to analysis (OAK, SFO, SJC).

weather	Description
DB	Local Climatological Data (LCD)
Source	National Oceanic and Atmospheric Administration (NOAA)
Link	https://www.ncdc.noaa.gov/cdo-web/datatools/lcd?prior=N

Table 4. Variables in both data sets *nycflights13* and *sfoflights18*

nycflights13	sfoflights18	Description
year	YEAR	Year of departure date
month	MONTH	Month of departure date
day	DAY_OF_MONTH	Day of departure date
dep_time	DEP_TIME	Actual time an aircraft lifts off from the origin airport.
sched_dep_time	CRS_DEP_TIME	The scheduled time that an aircraft should lift off from the origin airport.
dep_delay	DEP_DELAY	The difference between the scheduled departure time and the actual departure time from the origin airport gate.
arr_time	ARR_TIME	Actual time of arrival
sched_arr_time	CRS_ARR_TIME	The scheduled time that an aircraft should cross a certain point (landing or metering fix)
arr_delay	ARR_DELAY	The difference of the actual arrival time minus the scheduled arrival time. A flight is considered on-time when it arrives less than 15 minutes after its published arrival time.
carrier	OP_UNIQUE_CARRIER	Unique carrier code. It is the carrier code most recently used by a carrier.
flight	OP_CARRIER_FL_NUM	A one to four-character alpha-numeric code for a particular flight
tailnum	TAIL_NUM	Aircraft id number
origin	ORIGIN	A three-character alpha-numeric code issued by the U.S. Department of Transportation which is the official designation of the airport. In this case, the airport of origin.
dest	DEST	A three-character alpha-numeric code issued by

		the U.S. Department of Transportation which is the official designation of the airport. In this case, the airport of destination.
air_time	AIR_TIME	The total time an aircraft is in the air between an origin-destination airport pair, i.e. from wheels-off at the origin airport to wheels-down at the destination airport
distance	DISTANCE	Distance between airport of origin and airport of destination (in miles)
time_hour	FL_TIME	Day, month and year of the flight. It differs from the variable in nycflights13 because it does not include the time.
tailnum	TAIL_NUM	Identification number assigned to aircraft. Note: However, the FAA table uses a different notation for this number using digits only. The character "N" had to be added to the string using R code in order to be able to join the tables.
year	YEAR MFR	Year the airplane was manufactured
manufacturer	MFR MDL CODE	A code assigned to the aircraft manufacturer. Note: This is a code not the name of the manufacturer. I had to use a third table (ACFTREF) to get the manufacturer's name (MFR). This third table was joined to the second one by manufacture's code.

Table 5. New relevant variable

New Variable	Description
CANCELLED	A flight that was listed in a carrier's computer reservation system during the seven calendar days prior to scheduled departure but was not operated

Appendix II. Code

```
library(tidyverse)
library(lubridate)
library(nycflights13)
```

```
#Read data from the following sources: Seven monthly tables with California
flights data, one table with airplanes data,one table with plane
```

```
manufacturers data, and one table with weather data from three stations:
Oakland International Airport, San Francisco International Airport, and San
Jose California.

January18  <- read_csv(file="./Reporting Carrier Data/January2018.csv")
February18 <- read_csv(file="./Reporting Carrier Data/February2018.csv")
March18    <- read_csv(file="./Reporting Carrier Data/March2018.csv")
April18    <- read_csv(file="./Reporting Carrier Data/April2018.csv")
May18      <- read_csv(file="./Reporting Carrier Data/May2018.csv")
June18     <- read_csv(file="./Reporting Carrier Data/June2018.csv")
July18     <- read_csv(file="./Reporting Carrier Data/July2018.csv")
airplanes  <- read_csv(file="./Marketing Carrier Data/MASTER.csv")
manufNames <- read_csv(file="./Marketing Carrier Data/ACFTREF.csv")
weatherBA  <- read_csv(file="./Reporting Carrier Data/bayAreaWeather.csv")
```

```
#Merge data from the seven monthly tables into one table containing every
month between January and July 2018 for all flights departing from or
arriving to any airport in the State of California.

CA2018 <- bind_rows(January18, February18, March18, April18, May18, June18,
July18)
```

```
# Extract OAK, SFO and SJC originated flights data from the CA2018 flight
table.

sfoflights18 <- CA2018%>%
                    #mutate(time_hour)
                    filter(ORIGIN%in%c("OAK", "SFO","SJC"))
```

Programming block for 4.2

What month had the highest proportion of cancelled flights?

What month had the lowest?

```
# Calculate the proportion of cancelled flights for each month

cancelRate<- sfoflights18%>%
                group_by(MONTH)%>%
                summarize(N = n(), Num_of_Cancelled = sum(CANCELLED, na.rm =
TRUE), Cancel_rate = mean(CANCELLED, na.rm = TRUE))%>%
                arrange(desc(Cancel_rate))
```

```r
#identify the month with the highest cancellation rate.
head(cancelRate,1)
```

```
## # A tibble: 1 x 4
##   MONTH     N Num_of_Cancelled Cancel_rate
##   <int> <int>            <int>       <dbl>
## 1     3 23055              519      0.0225
```

```r
#identify the month with the lowest cancellation rate.
tail(cancelRate,1)
```

```
## # A tibble: 1 x 4
##   MONTH     N Num_of_Cancelled Cancel_rate
##   <int> <int>            <int>       <dbl>
## 1     2 20142              144     0.00715
```

Programming block for 4.3.a

What plane (specified by the tailnum variable) traveled the most times from Bay Area airports from January to July 2018?

```r
# Find the plane with the most trips made from January to July 2018

head(sfoflights18%>%
     select(TAIL_NUM, OP_UNIQUE_CARRIER)%>%
     group_by(TAIL_NUM)%>%
     summarize(N = n())%>%
     arrange(desc(N)),1)
```

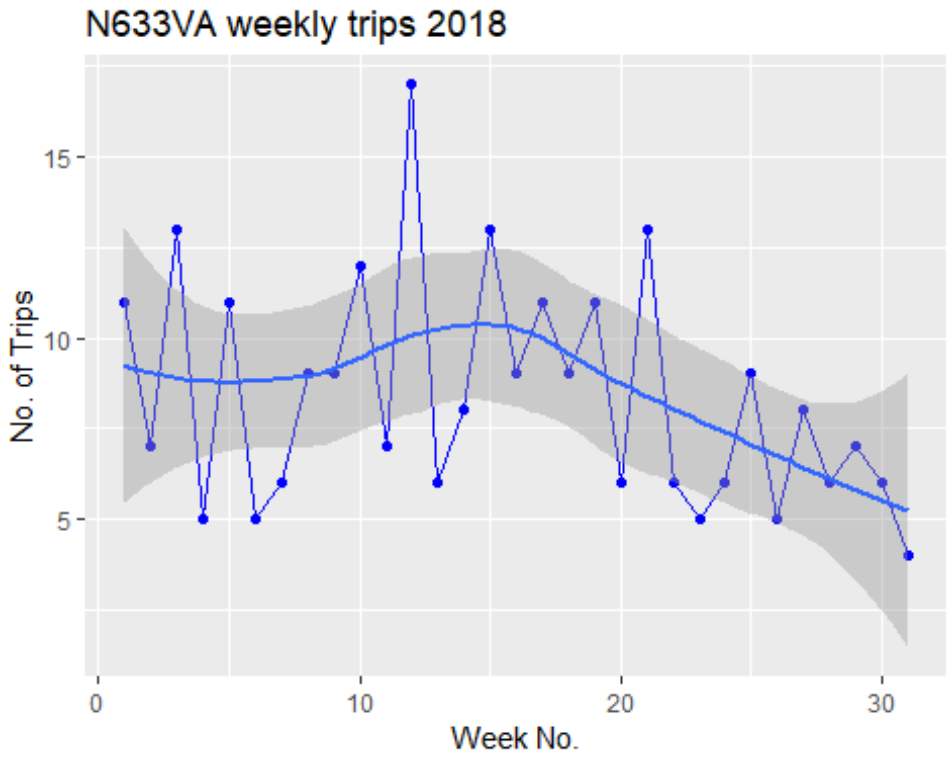
```
## # A tibble: 1 x 2
##   TAIL_NUM     N
##   <chr>    <int>
## 1 N633VA     260
```

Programming block for 4.3.b

Plot the number of trips per week over the year

```r
# Change the variable "date" from character to date type

sfoflights18 <- sfoflights18%>%
                    mutate(FL_DATE = as.Date(FL_DATE, format = "%m/%d/%Y"))
```

```
# Create a time series plot for the number of trips per week made by the most
used plane identified above.

library(ggplot2)
sfoflights18%>%
    mutate(week_num = week(FL_DATE))%>%
    filter(TAIL_NUM == "N633VA")%>%
    group_by(week_num)%>%
    summarize(N = n())%>%
    ggplot(aes(y = N, x = week_num))+ geom_point(color = "blue") +
geom_line(color="blue") + geom_smooth() + ggtitle("N633VA weekly trips
2018")+ xlab("Week No.") + ylab("No. of Trips")

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Programming block for 4.4.a

What is the oldest plane (specified by the tailnum variable) that flew from the Bay Area airports from January to July 2018?

```
# The airplanes id numbers on the original airplanes database from the FAA
don't have the "N" character at the begining of the  ID string. To be able to
join the airplane table to the flights table, the "N" has to be concatenated
to the id number to make it equal to the tailnum string on the flights
database. This chunk performs thit task.
```

```
 airplanes <-airplanes%>%
            mutate(N = paste0("N", airplanes$"N-NUMBER"))%>%
            select(`N-NUMBER`, N, `MFR MDL CODE`, `YEAR MFR`)
```

```
# Join the table containing manufacturer data to the flights table

sfoplanes <- sfoflights18%>%
                inner_join(airplanes, by = c("TAIL_NUM" = "N" ))
```

```
# List each plane and its manufacturing year to identify the oldest plane
that flew from the Bay Area from January to July 2018.

mfr <- distinct(sfoplanes%>%
            select(TAIL_NUM,`YEAR MFR`)%>%
            arrange(`YEAR MFR`))

head(mfr,1)

## # A tibble: 1 x 2
##   TAIL_NUM `YEAR MFR`
##   <chr>         <int>
## 1 N990JB         1977
```

Programming block for 4.4.b

How many airplanes that flew from the Bay Area are included in the planes table from the FAA?

Bonus question: How many are NOT included?

```
# Create a table of planes that flew from BA airports and join it to the
planes master from the FAA to identify how many of the flying planes were
included in the planes master.


planesSFO <- distinct(sfoflights18%>%
                        group_by(TAIL_NUM)%>%
                        select(TAIL_NUM))

planesMaster <- airplanes%>%
                  select("N")

planesJoined <-planesSFO%>%
```

```
                      inner_join(planesMaster, by = c("TAIL_NUM" = "N"))
nrow(planesJoined)

## [1] 3660

#Calculate the number of planes flying from the Bay Area.


nrow(planesSFO)

## [1] 3724
```

Note: As a result of this we learn that 64 planes that flew from the BA were not included in the planes table.

Programming block for 4.5.a

How many planes have a missing date of manufacture?

```
#  Calculate how many of the planes that flew from the BA are missing
manufacture date.

miss_mfr <- mfr %>%
              filter(is.na(`YEAR MFR`))

nrow(miss_mfr)

## [1] 86
```

Programming block for 4.5.b

What are the five most common manufacturers?

```
# Create a table containing flights, planes, manufacturer names and dates of
manufacture.

Manufacturers <- airplanes%>%
                    inner_join(manufNames, by = c("MFR MDL CODE" =
"CODE"))%>%
                    select(N,`MFR MDL CODE`, MFR, `YEAR MFR`)
```

```
# Calulate the number of planes flying from the BA per manufacturer.

common_manuf <- sfoflights18%>%
                    inner_join(Manufacturers, by = c("TAIL_NUM" = "N"))%>%
```

```
                    select(TAIL_NUM, MFR)%>%
                    unique()%>%
                    group_by(MFR)%>%
                    summarize(aircrafts = n())%>%
                    arrange(desc(aircrafts))

common_manuf
```

```
## # A tibble: 25 x 2
##    MFR                aircrafts
##    <chr>                  <int>
##  1 BOEING                  2169
##  2 AIRBUS                   705
##  3 AIRBUS INDUSTRIE         390
##  4 BOMBARDIER INC           212
##  5 EMBRAER S A              104
##  6 MCDONNELL DOUGLAS         45
##  7 EMBRAER                    9
##  8 CESSNA                     4
##  9 AIR TRACTOR INC            3
## 10 CIRRUS DESIGN CORP         3
## # ... with 15 more rows
```

```
#Fix the duplicate manufacturer names and group the manufacturers with small
amount of planes into one category called "other"


fix_dupl_mfr <- sfoflights18%>%
                    inner_join(Manufacturers, by = c("TAIL_NUM" = "N"))%>%
                    mutate(shrtName = strtrim(MFR,6), Mfrer =
ifelse(shrtName%in%c("AIRBUS","BOEING","BOMBAR","EMBRAE","MCDONN"),
shrtName,"OTHER"))%>%
                    select(Mfrer, TAIL_NUM, `YEAR MFR`)%>%
                    unique()%>%
                    group_by(Mfrer)%>%
                    summarize(aircrafts = n())%>%
                    arrange(desc(aircrafts))



fix_dupl_mfr
```

```
## # A tibble: 6 x 2
##   Mfrer  aircrafts
##   <chr>      <int>
## 1 BOEING      2169
## 2 AIRBUS      1095
## 3 BOMBAR       212
```

```
## 4 EMBRAE        113
## 5 MCDONN         45
## 6 OTHER          26
```

Programming block for 4.5.c

Has the distribution of Manufacturer changed over time as reflected by the planes flying from the Bay Area from January to July 2018?

```
# Calculate the average year of manufacture of the planes flying from the BA
per manufacturer.


sfoflights18%>%
                    inner_join(Manufacturers, by = c("TAIL_NUM" = "N"))%>%
                    mutate(shrtName = strtrim(MFR,6), Mfrer =
ifelse(shrtName%in%c("AIRBUS","BOEING","BOMBAR","EMBRAE","MCDONN"),
shrtName,"OTHER"))%>%
                    select(Mfrer, TAIL_NUM, `YEAR MFR`)%>%
                    group_by(Mfrer) %>%
                    summarize(avgyear = round(mean(`YEAR MFR`, na.rm =
TRUE))) %>%
                    arrange(desc(avgyear))

## # A tibble: 6 x 2
##   Mfrer  avgyear
##   <chr>    <dbl>
## 1 EMBRAE    2016
## 2 AIRBUS    2007
## 3 BOEING    2006
## 4 BOMBAR    2004
## 5 OTHER     2003
## 6 MCDONN    1996
```

Programming block for Merging New Weather Data with sfoflights18

```
# Wrangle the data before the merge.
# Create a date variable to the minute by converting the time_hour character
variable into POSIXct format.
# Correct the formats of some of the weather variables
# Rename the long names of the weather stations to make them consistent with
the airport codes.
#  Select the relevant weather variables for the first 7 months of 2018

weatherBA <- weatherBA%>%
          mutate(time_minute = as.POSIXct(time_hour, format = "%m/%d/%Y
%H:%M"), wind_dir = as.integer(wind_dir), visib = as.double(visib), precip =
```

```
as.double(precip), origin =
            case_when(
              origin == "OAKLAND METROPOLITAN INTERNATIONAL AIRPORT CA US" ~
"OAK",
              origin == "SAN FRANCISCO INTERNATIONAL AIRPORT CA US" ~ "SFO",
              origin == "SAN JOSE CA US" ~ "SJC",
              TRUE ~ origin))%>%
            filter(month(time_minute)<8)%>%
                select(origin, visib, temp, dewp, humid, wind_speed,
wind_dir, pressure, precip, wind_gust, time_minute)%>%
                arrange(time_minute)
```

Create an HOURLY WEATHER table for each origin

Hourly Weather table for OAK airport

```
OAKStation <- weatherBA%>%
  filter(origin == "OAK", minute(time_minute)==53)
```

Hourly Weather table for SJC airport

```
SJCStation <- weatherBA%>%
  filter(origin == "SJC", minute(time_minute)==53)
```

Hourly Weather table for SFO airport

```
SFOStation <- weatherBA%>%
  filter(origin == "SFO", minute(time_minute)==56)
```

Flights from Oakland California (OAK)

```
# Filter the flights from Oakland
# Change the scheduled departure time variable (CRS_DEP_TIME) to character
variable
# Substract the hour part of the scheduled departure time and assign it to a
variable called "hour"
# Paste the character variables FL_DATE and hour to the string"53" to create
a unique date format variable with year, month, day, hour and the minute 53
as it is every hourly observation in the weather table.
sfoflights18Oak <- CA2018%>%
                    filter(ORIGIN%in%c("OAK"))%>%
                    mutate(CRS_DEP_TIME = as.character(CRS_DEP_TIME),  hour =
ifelse(nchar(CRS_DEP_TIME)==3,substr(CRS_DEP_TIME,1,1),substr(CRS_DEP_TIME,1,
2)), time_minute = as.POSIXct(paste(paste(FL_DATE, hour, sep="
"),"53",sep=":"),format="%m/%d/%Y %H:%M"))
```

Flights from San Jose California (SJC)

```
sfoflights18SJC <- CA2018%>%
                    filter(ORIGIN%in%c("SJC"))%>%
                    mutate(CRS_DEP_TIME = as.character(CRS_DEP_TIME),  hour =
ifelse(nchar(CRS_DEP_TIME)==3,substr(CRS_DEP_TIME,1,1),substr(CRS_DEP_TIME,1,
2)), time_minute = as.POSIXct(paste(paste(FL_DATE, hour, sep="
"),"53",sep=":"),format="%m/%d/%Y %H:%M"))
```

Flights from San Francisco California (SFO)

```
# Use the string "56" instead of "53" to create the linking variable. (SFO
uses this minute for its weather hourly observations).
sfoflights18SFO <- CA2018%>%
                    filter(ORIGIN%in%c("SFO"))%>%
                    mutate(CRS_DEP_TIME = as.character(CRS_DEP_TIME),  hour =
ifelse(nchar(CRS_DEP_TIME)==3,substr(CRS_DEP_TIME,1,1),substr(CRS_DEP_TIME,1,
2)), time_minute = as.POSIXct(paste(paste(FL_DATE, hour, sep="
"),"56",sep=":"),format="%m/%d/%Y %H:%M"))
```

Flight&Weather data set for Oakland

```
#Merge the flights table to the weather table for OAK origin

OAKflightWeath <- sfoflights18Oak%>%
        left_join(OAKStation, by = "time_minute")
```

Flight&Weather data set for San Jose

```
#Merge the flights table to the weather table for SJC origin

SJCflightWeath <- sfoflights18SJC%>%
                    left_join(SJCStation, by = "time_minute")
```

Flight&Weather data set for San Francisco

```
#Merge the flights table to the weather table for SFO origin

SFOflightWeath <- sfoflights18SFO%>%
                    left_join(SFOStation, by = "time_minute")
```

Merged table Weather and Flights for the three origins in the Bay Area.

```
#Bind the three single origin merged tables into BAflightWeather

BAflightWeather <- bind_rows(OAKflightWeath, SJCflightWeath, SFOflightWeath)
```

A glimpse of the merged table "BAflightWeather" which contains all flights flying from the Bay Area with their respective weather indicators per Airport per hour.

Note that this merged table has 163,056 observations, exactly the same as the original sfoflights18 table.

```
glimpse(BAflightWeather)
```

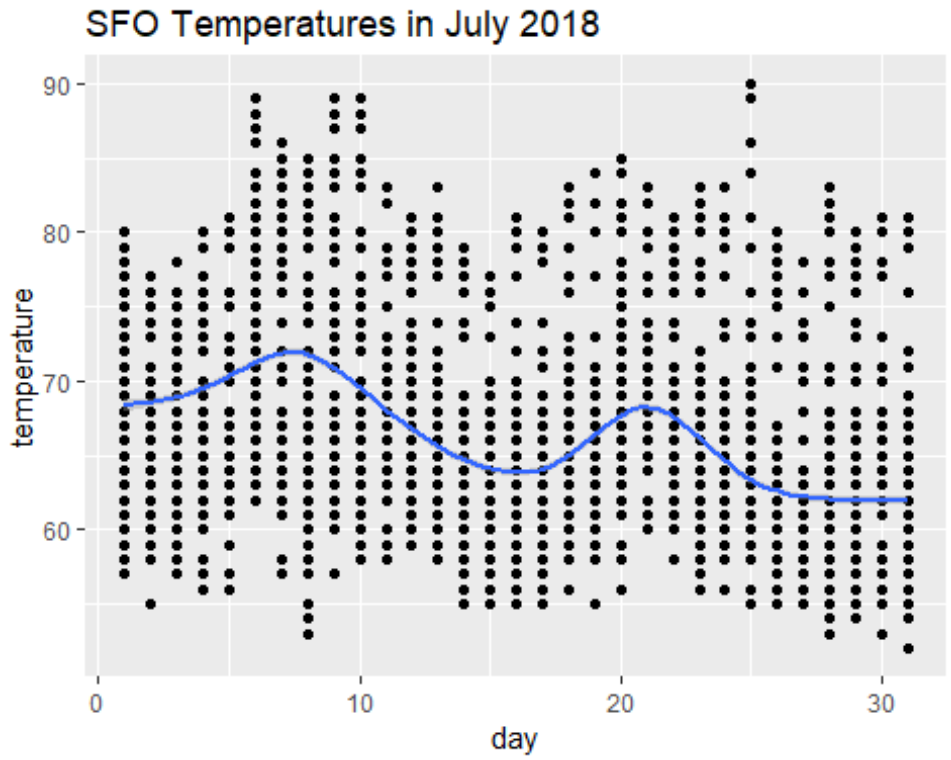
```
## Observations: 163,056
## Variables: 33
## $ YEAR              <int> 2018, 2018, 2018, 2018, 2018, 2018, 2018, 20...
## $ MONTH             <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ DAY_OF_MONTH      <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2,...
## $ DAY_OF_WEEK       <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2,...
## $ FL_DATE           <chr> "1/1/2018", "1/1/2018", "1/1/2018", "1/1/201...
## $ OP_UNIQUE_CARRIER <chr> "AS", "AS", "AS", "AS", "AS", "AS", "AS", "A...
## $ TAIL_NUM          <chr> "N273AK", "N565AS", "N274AK", "N526AS", "N58...
## $ OP_CARRIER_FL_NUM <int> 335, 345, 353, 569, 811, 845, 877, 915, 917,...
## $ ORIGIN_AIRPORT_ID <int> 13796, 13796, 13796, 13796, 13796, 13796, 13...
## $ ORIGIN            <chr> "OAK", "OAK", "OAK", "OAK", "OAK", "OAK", "O...
## $ DEST_AIRPORT_ID   <int> 14747, 14747, 14747, 14747, 13830, 12758, 12...
## $ DEST              <chr> "SEA", "SEA", "SEA", "SEA", "OGG", "KOA", "L...
## $ CRS_DEP_TIME      <chr> "1237", "605", "905", "1847", "700", "815", ...
## $ DEP_TIME          <int> 1229, 708, 858, 1909, 649, 830, 733, 927, 19...
## $ DEP_DELAY         <int> -8, 63, -7, 22, -11, 15, 14, -10, -9, 1, -6,...
## $ CRS_ARR_TIME      <int> 1436, 815, 1110, 2052, 1044, 1158, 1110, 113...
## $ ARR_TIME          <int> 1426, 911, 1101, 2104, 1004, 1152, 1101, 110...
## $ ARR_DELAY         <int> -10, 56, -9, 12, -40, -6, -9, -26, -37, -16,...
## $ CANCELLED         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ AIR_TIME          <int> 97, 101, 94, 92, 296, 296, 312, 82, 77, 87, ...
## $ DISTANCE          <int> 672, 672, 672, 672, 2349, 2378, 2457, 543, 5...
## $ hour              <chr> "12", "6", "9", "18", "7", "8", "7", "9", "2...
## $ time_minute       <dttm> 2018-01-01 12:53:00, 2018-01-01 06:53:00, 2...
## $ origin            <chr> "OAK", "OAK", "OAK", "OAK", "OAK", "OAK", "O...
## $ visib             <dbl> 6, 5, 4, 10, 5, 5, 5, 4, 10, 8, 10, 8, 7, 8,...
## $ temp              <int> 54, 41, 50, 51, 40, 45, 40, 50, 51, 57, 47, ...
## $ dewp              <int> 48, 41, 50, 49, 40, 45, 40, 50, 48, 49, 44, ...
## $ humid             <int> 80, 100, 100, 92, 100, 100, 100, 100, 89, 74...
## $ wind_speed        <int> 6, 0, 5, 0, 6, 0, 6, 5, 0, 3, 0, 0, 3, 0, 0,...
## $ wind_dir          <int> 270, 0, 200, 0, 30, 0, 30, 200, 0, 340, 0, 0...
## $ pressure          <dbl> 30.11, 30.15, 30.18, 30.07, 30.15, 30.16, 30...
## $ precip            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ wind_gust         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
```

B. Answers to questions in exercises 4.6, 4.7 from *Modern Data Science with R.*

Programming block for 4.6.a

What is the distribution of temperature in July 2018?

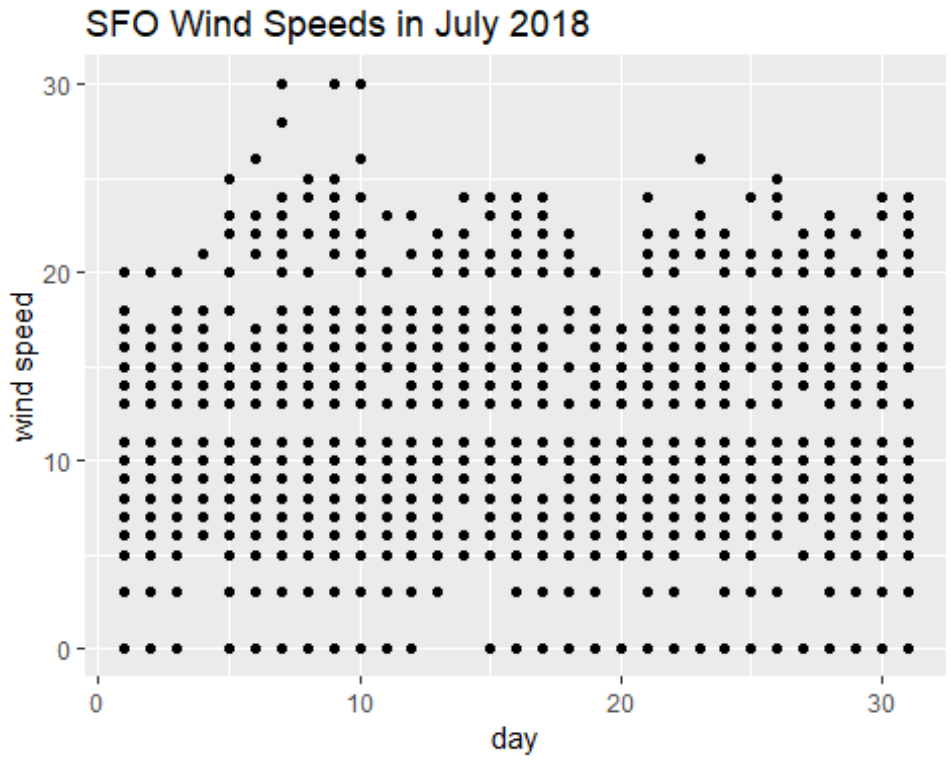
```
# Create a dotplot for the temperature data in july 2018 at the three bay
area airports under study (OAK, SFO, SJC)

BAflightWeather%>%
          filter(month(time_minute)== 7)%>%
          mutate(day = day(time_minute))%>%
          select(day,temp)%>%
          ggplot(aes(x =day, y = temp)) + geom_point() + labs(title = "SFO
Temperatures in July 2018", y = "temperature") + geom_smooth()

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



Programming block for 4.6.b

Identify any important outliers in terms of the wind_speed variable

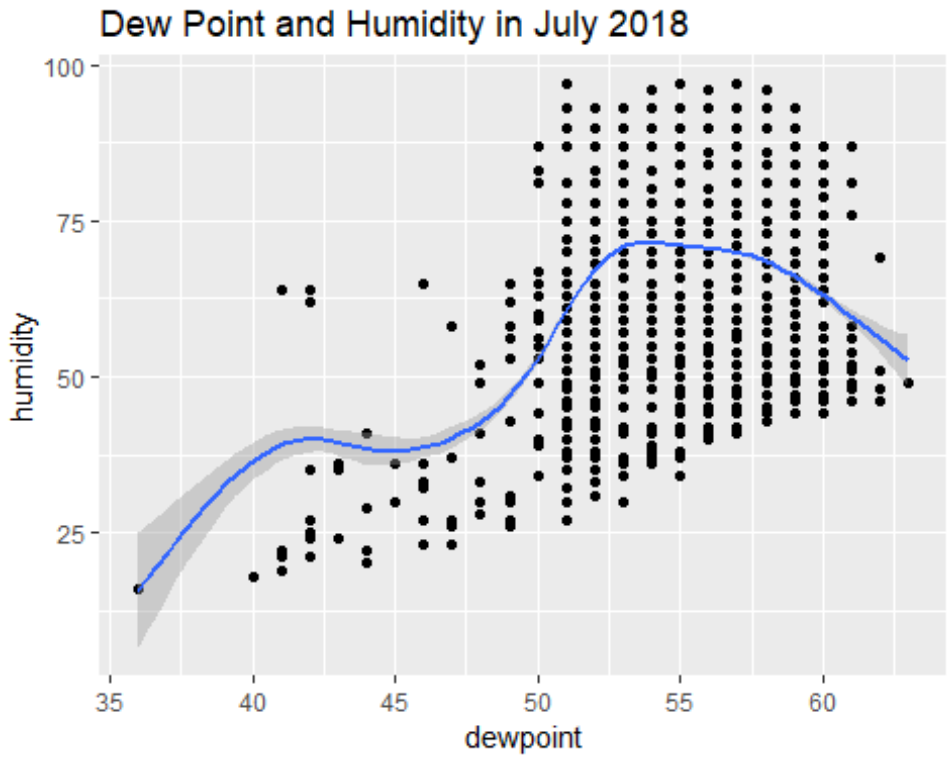
```
# Create a dotplot for the wind speed data in july 2018 at the three bay area
airports under study (OAK, SFO, SJC)

BAflightWeather%>%
          filter(month(time_minute)== 7)%>%
          mutate(day = day(time_minute))%>%
          select(day,wind_speed)%>%
          ggplot(aes(x =day, y = wind_speed)) + geom_point() + labs(title =
"SFO Wind Speeds in July 2018", y = "wind speed")
```



Programming block for 4.6.c

What is the relationship between *dewp* and *humid*?

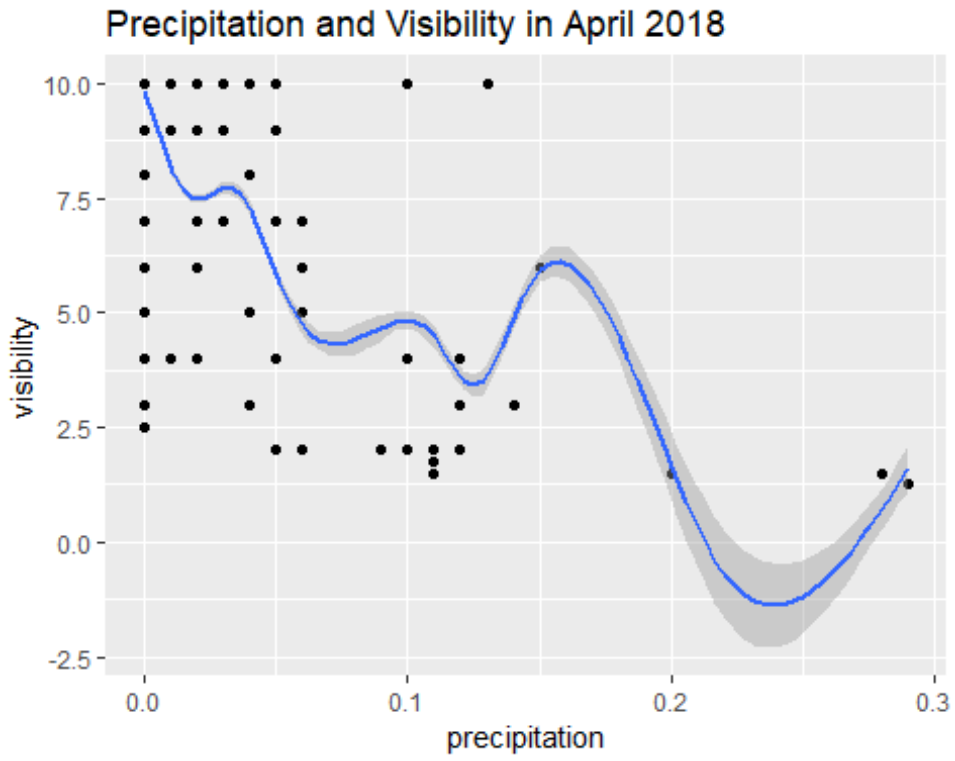
```
# Create a data plot to show the relationship between dew point and humidity
in July 2018 at the three airports of study (OAK, SFO, SJC)

BAflightWeather%>%
          filter(month(time_minute)== 7)%>%
          select(dewp,humid)%>%
          ggplot(aes(x = dewp, y = humid)) + geom_point() + labs(title = "Dew
Point and Humidity in July 2018", x = "dewpoint", y = "humidity") +
geom_smooth()

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



Programming block for 4.6.d

What is the relationship between *precip* and *visib*?

```
# Create a data plot to show the relationship between precipitation and
visibility in April 2018 at the three airports of study (OAK, SFO, SJC)

BAflightWeather%>%
    filter(month(time_minute)== 4)%>%
    select(precip,visib)%>%
    ggplot(aes(x = precip, y = visib)) + geom_point() + labs(title =
"Precipitation and Visibility in April 2018", x = "precipitation", y =
"visibility") + geom_smooth()

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



Programming block for 4.7.a

On how many days was there precipitation in the Bay Area from January to July 2018?

Bonus: Days of precipitation by month

```
# Calculate the total days of precipitation (Jan-Jul 2018) in OAK, SFO, SJC.

days_precip <- BAflightWeather%>%
                  filter(precip != 0)%>%
                  mutate(day = day(time_minute), month =
month(time_minute))%>%
                  select(month, day, precip)%>%
                  distinct(month, day)

nrow(days_precip)

## [1] 36
```

```
# Calculate the number of days when there was precipitation in the three BA
airports under study (OAK, SFO, SJC)
```

```
BAflightWeather%>%
   filter(precip != 0)%>%
   mutate(day = day(time_minute), month = month(time_minute))%>%
   select(month, day, precip)%>%
   distinct(month,day)%>%
   group_by(month)%>%
   summarize(n=n())
```

```
## # A tibble: 5 x 2
##   month     n
##   <dbl> <int>
## 1     1     9
## 2     2     4
## 3     3    16
## 4     4     6
## 5     5     1
```

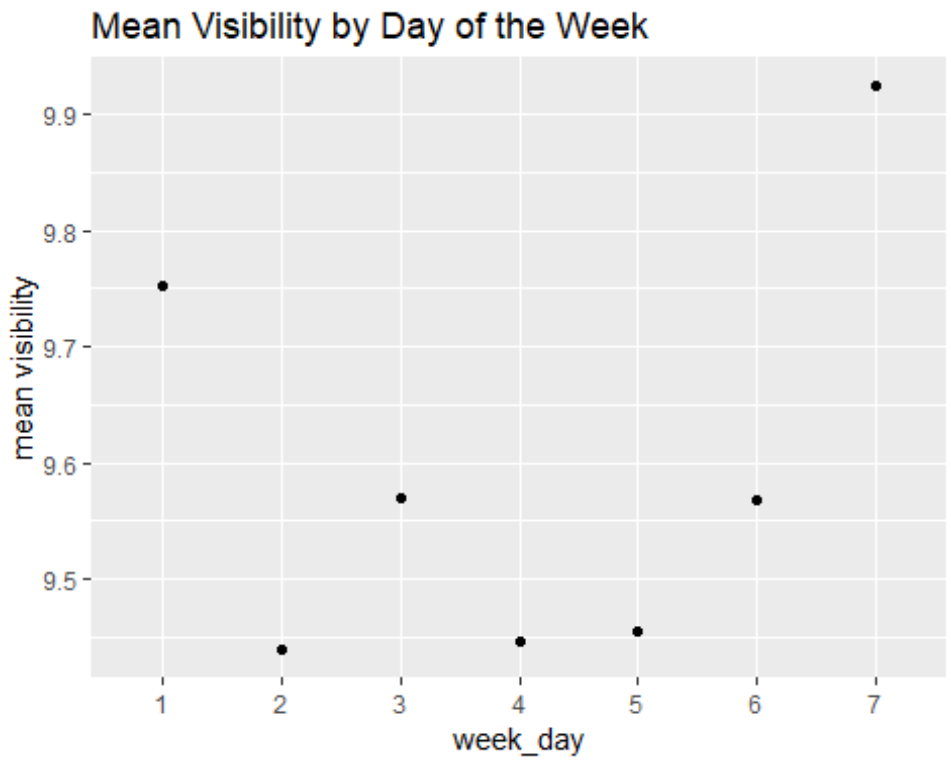
Programming block for 4.7.b

Were there any differences in the mean visibility based on the day of the week?

```
# Calculate the average visibility by day of the week (include std and N)

meanVisibDay <- na.omit(BAflightWeather%>%
                    mutate(week_day = as.character(wday(time_minute)))%>%
                    group_by(week_day)%>%
                    summarize(mean = mean(visib, na.rm = TRUE), sd =
sd(visib, na.rm = TRUE), N = n()))

meanVisibDay
```

```
## # A tibble: 7 x 4
##   week_day  mean    sd     N
##   <chr>    <dbl> <dbl> <int>
## 1 1         9.75 1.18  22577
## 2 2         9.44 1.64  24788
## 3 3         9.57 1.19  24120
## 4 4         9.45 1.57  23608
## 5 5         9.46 1.56  23963
## 6 6         9.57 1.44  24112
## 7 7         9.92 0.495 18981
```

```
# Create a plot with the average values of visibility by day of the week

ggplot(data = meanVisibDay, aes(x = week_day, y = mean)) + geom_point() +
ggtitle("Mean Visibility by Day of the Week") + labs(y = "mean visibility")
```



Programming block for 4.7.c

Were there any differences in the mean visibility based on the month of the year?

```
# Calculate the average visibility by month (include std and N)

meanVisibMonth <- na.omit(BAflightWeather%>%
                      mutate(month = as.character(month(time_minute)))%>%
                      group_by(month)%>%
                      summarize(mean = mean(visib, na.rm = TRUE), sd =
sd(visib, na.rm = TRUE), N = n()))

meanVisibMonth
```

```
## # A tibble: 7 x 4
##   month  mean    sd     N
##   <chr> <dbl> <dbl> <int>
## 1 1      8.42 2.49  22470
## 2 2      9.88 0.686 20011
## 3 3      9.53 1.48  22952
## 4 4      9.68 1.20  22929
## 5 5      9.80 0.834 24019
## 6 6      9.84 0.697 24403
## 7 7      9.89 0.623 25365
```

```r
# Create a plot with the average values of visibility by month

ggplot(data = meanVisibMonth, aes(x = month, y = mean)) + geom_point() +
ggtitle("Mean Visibility by Month") + labs(y = "mean visibility")
```