STAT650\_Project

Mulati\_Patiguli

10/14/2018

**Prof. Eric A. Suess**

library(tidyverse)

## ── Attaching packages ──────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.0.0 ✔ purrr 0.2.5  
## ✔ tibble 1.4.2 ✔ dplyr 0.7.6  
## ✔ tidyr 0.8.1 ✔ stringr 1.3.1  
## ✔ readr 1.1.1 ✔ forcats 0.3.0

## ── Conflicts ─────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(mosaic)

## Loading required package: lattice

## Loading required package: ggformula

## Loading required package: ggstance

##   
## Attaching package: 'ggstance'

## The following objects are masked from 'package:ggplot2':  
##   
## geom\_errorbarh, GeomErrorbarh

##   
## New to ggformula? Try the tutorials:   
## learnr::run\_tutorial("introduction", package = "ggformula")  
## learnr::run\_tutorial("refining", package = "ggformula")

## Loading required package: mosaicData

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':  
##   
## expand

##   
## The 'mosaic' package masks several functions from core packages in order to add   
## additional features. The original behavior of these functions should not be affected by this.  
##   
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.

##   
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':  
##   
## mean

## The following objects are masked from 'package:dplyr':  
##   
## count, do, tally

## The following object is masked from 'package:purrr':  
##   
## cross

## The following object is masked from 'package:ggplot2':  
##   
## stat

## The following objects are masked from 'package:stats':  
##   
## binom.test, cor, cor.test, cov, fivenum, IQR, median,  
## prop.test, quantile, sd, t.test, var

## The following objects are masked from 'package:base':  
##   
## max, mean, min, prod, range, sample, sum

library(Lahman)  
library(nycflights13)  
library(skimr)

##   
## Attaching package: 'skimr'

## The following object is masked from 'package:mosaic':  
##   
## n\_missing

# Part I:Questions to answer

## 1.Find the US Government website where Airline On-Time Performance Data can be downloaded. What website is this and how can you download the data? Download the data for the available months in 2018 for the Bay Area. Can you do this? If not, what can you download?

**Answer:** I found Airline On-Time Performance Data from the Google and the URL link was actually: “<https://www.transtats.bts.gov/Tables.asp?DB_ID=120>” then click on “download” link for the set of data “Reporting Carrier On-Time Performance (1987-present)” not the “Marketing Carrier” one above it.

This database contains scheduled and actual departure and arrival times reported by certified U.S. air carriers that account for at least one percent of domestic scheduled passenger revenues. The data is collected by the Office of Airline Information, Bureau of Transportation Statistics (BTS).

“Reporting Carrier On-Time Performance (1987-present)”are required to report on-time data for flights they operate: on-time arrival and departure data for non-stop domestic flights by month and year, by carrier and by origin and destination airport. Includes scheduled and actual departure and arrival times, canceled and diverted flights, taxi-out and taxi-in times, causes of delay and cancellation, air time, and non-stop distance. For my project , I downloaded the available months data in Calfornia .First , I selected Calfornia from filter geography and filtered the first 7 months and 2018 year .second, I selected the same variables that are available in the full dataset with the variables in the nycflgihts13 data set and also selected 10 more new variables for my sfoflights18 data set .

Set up working directory

setwd("~/Desktop/project data set")

Read the.csv files :

Calfornia201801<- read\_csv(  
 file="~/Desktop/project data set/CA Jan.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201802<- read\_csv(  
 file="~/Desktop/project data set/CA Feb.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201803<- read\_csv(  
 file="~/Desktop/project data set/CA March.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201804<- read\_csv(  
 file="~/Desktop/project data set/CA April.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201805<- read\_csv(file="~/Desktop/project data set/CA May.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201806<- read\_csv(file="~/Desktop/project data set/CA JUNE.csv")

## Warning: Missing column names filled in: 'X24' [24]

Calfornia201807<- read\_csv(file="~/Desktop/project data set/CA JULY.csv")

## Warning: Missing column names filled in: 'X24' [24]

## 2.Once you have your data downloaded, develop your code for the first month of data. The last step will be to merge the data and perform an overall analysis for 2018. Extract the flights that departed from the Bay Area. Include all flights departing from San Francisco, Oakland, and San Jose. How many flight were there in January 2018?

**Answer:** After extracting all the flights departing from SFO ,OAK,SJC , there were total of 22606 flights in January 2018 . There were 14135 flights from “SFO” , 4277 flights from “SJC” and 4194 flights from “OAK” .

Calfornia201801

## # A tibble: 107,006 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## 2 2018 1 27 2018-01-27 N16217 366  
## 3 2018 1 27 2018-01-27 N37255 361  
## 4 2018 1 27 2018-01-27 N424UA 360  
## 5 2018 1 27 2018-01-27 N36469 358  
## 6 2018 1 27 2018-01-27 N835UA 346  
## 7 2018 1 27 2018-01-27 N76503 341  
## 8 2018 1 27 2018-01-27 N69810 340  
## 9 2018 1 27 2018-01-27 N76288 338  
## 10 2018 1 27 2018-01-27 N67846 325  
## # ... with 106,996 more rows, and 18 more variables:  
## # ORIGIN\_AIRPORT\_ID <int>, ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>,  
## # ORIGIN\_CITY\_NAME <chr>, ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

Calfornia201801 %>% filter(ORIGIN == "SFO" | ORIGIN == "SJC" | ORIGIN == "OAK" ) %>%  
 count()

## # A tibble: 1 x 1  
## n  
## <int>  
## 1 22606

14135 flights from “SFO”

Calfornia201801

## # A tibble: 107,006 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## 2 2018 1 27 2018-01-27 N16217 366  
## 3 2018 1 27 2018-01-27 N37255 361  
## 4 2018 1 27 2018-01-27 N424UA 360  
## 5 2018 1 27 2018-01-27 N36469 358  
## 6 2018 1 27 2018-01-27 N835UA 346  
## 7 2018 1 27 2018-01-27 N76503 341  
## 8 2018 1 27 2018-01-27 N69810 340  
## 9 2018 1 27 2018-01-27 N76288 338  
## 10 2018 1 27 2018-01-27 N67846 325  
## # ... with 106,996 more rows, and 18 more variables:  
## # ORIGIN\_AIRPORT\_ID <int>, ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>,  
## # ORIGIN\_CITY\_NAME <chr>, ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

Calfornia201801 %>% filter (ORIGIN == "SFO" )%>%  
 count()

## # A tibble: 1 x 1  
## n  
## <int>  
## 1 14135

4277 flights from “SJC”

Calfornia201801

## # A tibble: 107,006 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## 2 2018 1 27 2018-01-27 N16217 366  
## 3 2018 1 27 2018-01-27 N37255 361  
## 4 2018 1 27 2018-01-27 N424UA 360  
## 5 2018 1 27 2018-01-27 N36469 358  
## 6 2018 1 27 2018-01-27 N835UA 346  
## 7 2018 1 27 2018-01-27 N76503 341  
## 8 2018 1 27 2018-01-27 N69810 340  
## 9 2018 1 27 2018-01-27 N76288 338  
## 10 2018 1 27 2018-01-27 N67846 325  
## # ... with 106,996 more rows, and 18 more variables:  
## # ORIGIN\_AIRPORT\_ID <int>, ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>,  
## # ORIGIN\_CITY\_NAME <chr>, ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

Calfornia201801 %>% filter (ORIGIN == "SJC" )%>%  
 count()

## # A tibble: 1 x 1  
## n  
## <int>  
## 1 4277

4194 flights from “OAK” .

Calfornia201801

## # A tibble: 107,006 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## 2 2018 1 27 2018-01-27 N16217 366  
## 3 2018 1 27 2018-01-27 N37255 361  
## 4 2018 1 27 2018-01-27 N424UA 360  
## 5 2018 1 27 2018-01-27 N36469 358  
## 6 2018 1 27 2018-01-27 N835UA 346  
## 7 2018 1 27 2018-01-27 N76503 341  
## 8 2018 1 27 2018-01-27 N69810 340  
## 9 2018 1 27 2018-01-27 N76288 338  
## 10 2018 1 27 2018-01-27 N67846 325  
## # ... with 106,996 more rows, and 18 more variables:  
## # ORIGIN\_AIRPORT\_ID <int>, ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>,  
## # ORIGIN\_CITY\_NAME <chr>, ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

Calfornia201801 %>% filter (ORIGIN == "OAK" )%>%  
 count()

## # A tibble: 1 x 1  
## n  
## <int>  
## 1 4194

## 3.Compare the variables that are available in the full dataset with the variables in the nycflgihts13 data set. Make a table of the variables that are in both datasets, with a description of each variable. Hint: In RStudio see Help > RMarkdown Quick Reference > Tables. Report the intersection of the variables.

**Answer:** I have 13 shared variables in my sfoflights18 data set .

|  |  |
| --- | --- |
| Shared Variables Names | Variable Description |
| YEAR | Year of flight |
| MONTH | month of flight |
| DAY | day of the flight |
| TAIL\_NUM | tail number which identifies the plane |
| ORIGIN | Origin Airport |
| DEST | Destination Airport |
| DEPTIME | Actual Departure Time (local time: hhmm) |
| DEPDELAY | Difference in minutes between scheduled and actual departure time. Early departures show negative numbers. |
| ARRTIME | Actual Arrival Time (local time: hhmm) |
| ARRDELAY | Difference in minutes between scheduled and actual arrival time. Early arrivals show negative numbers. |
| AIRTIME | Flight Time, in Minutes |
| FLIGHTS | Number of Flights |
| DISTANCE | Distance between airports (miles) |

## 4.What new variables do you now also have. Make a table of the variables that are in the new dataset (that could also be downloaded from the website), with a description of each variable. (Ok, this is kind of long. Make a table for 10 other variables you consider important.)

**Answer:** I have another 10 more new variables ,that were different from nycflights13 ,in my new sfoflights18 data set .

|  |  |
| --- | --- |
| New Variables Name | Variable Description |
| FLIGHTDATE | Flight Date (yyyymmdd) |
| Flight\_Number\_Reporting\_Airline | Flight Number |
| OriginAirportID | Origin Airport, Airport ID. An identification number assigned by US DOT to identify a unique airport. Use this field for airport analysis across a range of years because an airport can change its airport code and airport codes can be reused. |
| OriginAirportSeqID | Origin Airport, Airport Sequence ID. An identification number assigned by US DOT to identify a unique airport at a given point of time. Airport attributes, such as airport name or coordinates, may change over time. |
| OriginCityName | Origin Airport, City Name |
| OriginStateName | Origin Airport, State Name |
| DestAirportID | Destination Airport, Airport ID. An identification number assigned by US DOT to identify a unique airport. Use this field for airport analysis across a range of years because an airport can change its airport code and airport codes can be reused. |
| DestAirportSeqID | Destination Airport, Airport Sequence ID. An identification number assigned by US DOT to identify a unique airport at a given point of time. Airport attributes, such as airport name or coordinates, may change over time. |
| DestCityName | Destination Airport, City Name |
| DestStateName | Destination Airport, State Name |

## 5.Answer Exercises 4.2, 4.3, 4.4 (you can only answer the second part of 4.4) on page 89 of the book, changing nycflights13 to sfoflights18. Answer all of the questions for the SF Bay Area in 2018.

**Answer to 4.2:** before doing an overall analysis for 2018 data , we need to merge the data . we use bind\_rows()function to bind the available months table .

Calfornia2018 <- bind\_rows( Calfornia201801 , Calfornia201802, Calfornia201803, Calfornia201804,Calfornia201805, Calfornia201806, Calfornia201807)

The month with the highest proportion of cancelled flights is month == 3, March . March has a highest number of cancelations with 560 .  
month == 2,February had the lowest cancellation rate with 184, since we have the data only from 1 to 7 months , there is no obvious seasonal pattern .

Calfornia2018 %>% filter(ORIGIN == "SFO" | ORIGIN == "SJC" | ORIGIN == "OAK" ) %>%  
 select(MONTH, ARR\_DELAY) %>%  
 group\_by(MONTH) %>%  
 skim()

## Skim summary statistics  
## n obs: 163056   
## n variables: 2   
## group variables: MONTH   
##   
## ── Variable type:numeric ────────────────────────────────────────────  
## MONTH variable missing complete n mean sd p0 p25 p50 p75 p100  
## 1 ARR\_DELAY 391 22215 22606 1 35.23 -62 -15 -7 5 1110  
## 2 ARR\_DELAY 184 19958 20142 -0.47 35.01 -69 -15 -7 5 1216  
## 3 ARR\_DELAY 560 22495 23055 5.83 40.97 -66 -14 -5 11 956  
## 4 ARR\_DELAY 304 22786 23090 2.87 40.95 -68 -14 -6 7 1284  
## 5 ARR\_DELAY 357 23808 24165 9.24 41.51 -57 -11 -2 14 1157  
## 6 ARR\_DELAY 306 24220 24526 7.96 43.21 -56 -11 -2 12 1431  
## 7 ARR\_DELAY 350 25122 25472 7.18 38.72 -54 -11 -2 12 1096  
## hist  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁  
## ▇▁▁▁▁▁▁▁

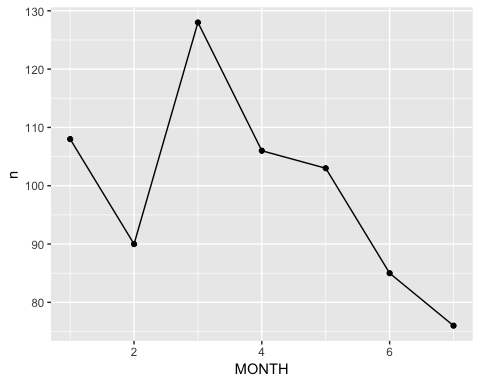
**Answer to 4.3:**

Tail number N633VA plane travelled the most times from SFO in 2018 ,Tail number 337NV plane travelled the most times from OAK in 2018 and Tail number N188SY plane travelled the most times from SSJC in 2018.

Calfornia2018 %>% filter ( ORIGIN == "SFO") %>%  
 group\_by(TAIL\_NUM) %>%  
 count()%>%  
 arrange(desc(n))

## # A tibble: 3,254 x 2  
## # Groups: TAIL\_NUM [3,254]  
## TAIL\_NUM n  
## <chr> <int>  
## 1 N633VA 254  
## 2 N281VA 248  
## 3 N630VA 248  
## 4 N286VA 245  
## 5 N640VA 245  
## 6 N284VA 244  
## 7 N285VA 243  
## 8 N629VA 243  
## 9 N835VA 232  
## 10 N847VA 232  
## # ... with 3,244 more rows

Calfornia2018 %>% filter( TAIL\_NUM == "N633VA") %>%  
 group\_by(MONTH) %>%  
 tally() %>%  
 ggplot(aes(x = MONTH, y = n)) +  
 geom\_point() +  
 geom\_line()

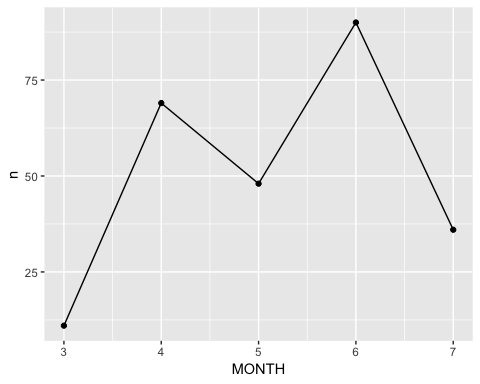


Tail number 337NV plane travelled the most times from OAK in 2018.

Calfornia2018 %>% filter ( ORIGIN == "OAK") %>%  
 group\_by(TAIL\_NUM) %>%  
 count()%>%  
 arrange(desc(n))

## # A tibble: 1,669 x 2  
## # Groups: TAIL\_NUM [1,669]  
## TAIL\_NUM n  
## <chr> <int>  
## 1 337NV 90  
## 2 N202HA 70  
## 3 N288WN 61  
## 4 N930WN 61  
## 5 N273WN 60  
## 6 N7742B 60  
## 7 N260WN 57  
## 8 N551WN 56  
## 9 N204HA 55  
## 10 N7738A 55  
## # ... with 1,659 more rows

Calfornia2018 %>% filter( TAIL\_NUM == "337NV") %>%  
 group\_by(MONTH) %>%  
 tally() %>%  
 ggplot(aes(x = MONTH, y = n)) +  
 geom\_point() +  
 geom\_line()

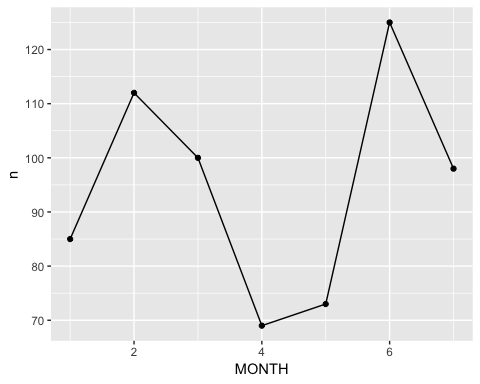


Tail number N188SY plane travelled the most times from SSJC in 2018.

Calfornia2018 %>% filter ( ORIGIN == "SJC") %>%  
 group\_by(TAIL\_NUM) %>%  
 count()%>%  
 arrange(desc(n))

## # A tibble: 2,394 x 2  
## # Groups: TAIL\_NUM [2,394]  
## TAIL\_NUM n  
## <chr> <int>  
## 1 N188SY 179  
## 2 N186SY 175  
## 3 N193SY 163  
## 4 N184SY 144  
## 5 N189SY 141  
## 6 N176SY 140  
## 7 N192SY 138  
## 8 N194SY 129  
## 9 N177SY 122  
## 10 N178SY 122  
## # ... with 2,384 more rows

Calfornia2018 %>% filter( TAIL\_NUM == "N188SY") %>%  
 group\_by(MONTH) %>%  
 tally() %>%  
 ggplot(aes(x = MONTH, y = n)) +  
 geom\_point() +  
 geom\_line()

 **Answer to 4.4:**

Total of 4,114 airplanes that flew from SFO ,OAK ,SJC are included in plane table . 1701 of airplanes from SFO , 1388 of airplanes from SJC and 1025 of airplanes from OAK seperately .

Calfornia2018 %>% head(1)

## # A tibble: 1 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## # ... with 18 more variables: ORIGIN\_AIRPORT\_ID <int>,  
## # ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>, ORIGIN\_CITY\_NAME <chr>,  
## # ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

planes %>% head(1)

## # A tibble: 1 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… EMBRAER EMB-1… 2 55 NA Turbo…

planes

## # 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 wi… EMBRAER EMB-1… 2 55 NA Turbo…  
## 2 N102UW 1998 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 3 N103US 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 4 N104UW 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 5 N10575 2002 Fixed wi… EMBRAER EMB-1… 2 55 NA Turbo…  
## 6 N105UW 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 7 N107US 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 8 N108UW 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 9 N109UW 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## 10 N110UW 1999 Fixed wi… AIRBUS INDUS… A320-… 2 182 NA Turbo…  
## # ... with 3,312 more rows

planes2 <- planes %>% rename( YEAR = year )%>%   
 rename( TAIL\_NUM =tailnum )  
  
planes2

## # A tibble: 3,322 x 9  
## TAIL\_NUM YEAR type manufacturer model engines seats speed engine  
## <chr> <int> <chr> <chr> <chr> <int> <int> <int> <chr>   
## 1 N10156 2004 Fixed wi… EMBRAER EMB-… 2 55 NA Turbo…  
## 2 N102UW 1998 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 3 N103US 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 4 N104UW 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 5 N10575 2002 Fixed wi… EMBRAER EMB-… 2 55 NA Turbo…  
## 6 N105UW 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 7 N107US 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 8 N108UW 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 9 N109UW 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## 10 N110UW 1999 Fixed wi… AIRBUS INDUS… A320… 2 182 NA Turbo…  
## # ... with 3,312 more rows

planes2 %>% left\_join(Calfornia2018, by = c("TAIL\_NUM" = "TAIL\_NUM")) %>%  
 filter(ORIGIN == "SFO" | ORIGIN == "SJC" | ORIGIN == "OAK" )%>%  
 group\_by(TAIL\_NUM,ORIGIN) %>%  
 count()

## # A tibble: 4,114 x 3  
## # Groups: TAIL\_NUM, ORIGIN [4,114]  
## TAIL\_NUM ORIGIN n  
## <chr> <chr> <int>  
## 1 N102UW SFO 1  
## 2 N103US OAK 3  
## 3 N104UW OAK 2  
## 4 N107US OAK 4  
## 5 N107US SJC 1  
## 6 N109UW OAK 1  
## 7 N110UW OAK 1  
## 8 N111US OAK 2  
## 9 N11206 SFO 34  
## 10 N11206 SJC 2  
## # ... with 4,104 more rows

1701 of airplanes from SFO are included in the planes table.

Calfornia2018 %>% head(1)

## # A tibble: 1 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## # ... with 18 more variables: ORIGIN\_AIRPORT\_ID <int>,  
## # ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>, ORIGIN\_CITY\_NAME <chr>,  
## # ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

planes %>% head(1)

## # A tibble: 1 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… EMBRAER EMB-1… 2 55 NA Turbo…

planes2 <- planes %>% rename( YEAR = year )%>%   
 rename( TAIL\_NUM =tailnum )  
  
planes2 %>% left\_join(Calfornia2018, by = c("TAIL\_NUM" = "TAIL\_NUM")) %>%  
 filter(ORIGIN == "SFO" )%>%  
 group\_by(TAIL\_NUM,ORIGIN) %>%  
 count()

## # A tibble: 1,701 x 3  
## # Groups: TAIL\_NUM, ORIGIN [1,701]  
## TAIL\_NUM ORIGIN n  
## <chr> <chr> <int>  
## 1 N102UW SFO 1  
## 2 N11206 SFO 34  
## 3 N1201P SFO 9  
## 4 N12109 SFO 48  
## 5 N12114 SFO 52  
## 6 N12116 SFO 61  
## 7 N12125 SFO 61  
## 8 N12216 SFO 38  
## 9 N12218 SFO 62  
## 10 N12221 SFO 44  
## # ... with 1,691 more rows

1388 of airplanes from SJC are included in the planes table.

Calfornia2018 %>% head(1)

## # A tibble: 1 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## # ... with 18 more variables: ORIGIN\_AIRPORT\_ID <int>,  
## # ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>, ORIGIN\_CITY\_NAME <chr>,  
## # ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

planes %>% head(1)

## # A tibble: 1 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… EMBRAER EMB-1… 2 55 NA Turbo…

planes2 <- planes %>% rename( YEAR = year )%>%   
 rename( TAIL\_NUM =tailnum )  
  
planes2 %>% left\_join(Calfornia2018, by = c("TAIL\_NUM" = "TAIL\_NUM")) %>%  
 filter(ORIGIN == "SJC" )%>%  
 group\_by(TAIL\_NUM,ORIGIN) %>%  
 count()

## # A tibble: 1,388 x 3  
## # Groups: TAIL\_NUM, ORIGIN [1,388]  
## TAIL\_NUM ORIGIN n  
## <chr> <chr> <int>  
## 1 N107US SJC 1  
## 2 N11206 SJC 2  
## 3 N12216 SJC 3  
## 4 N12221 SJC 1  
## 5 N12225 SJC 1  
## 6 N12238 SJC 4  
## 7 N13248 SJC 2  
## 8 N13716 SJC 3  
## 9 N13750 SJC 4  
## 10 N14214 SJC 5  
## # ... with 1,378 more rows

1025 of airplanes from OAK are included in the planes table.

Calfornia2018 %>% head(1)

## # A tibble: 1 x 24  
## YEAR MONTH DAY\_OF\_MONTH FL\_DATE TAIL\_NUM OP\_CARRIER\_FL\_N…  
## <int> <int> <int> <date> <chr> <int>  
## 1 2018 1 27 2018-01-27 N477UA 368  
## # ... with 18 more variables: ORIGIN\_AIRPORT\_ID <int>,  
## # ORIGIN\_AIRPORT\_SEQ\_ID <int>, ORIGIN <chr>, ORIGIN\_CITY\_NAME <chr>,  
## # ORIGIN\_STATE\_NM <chr>, DEST\_AIRPORT\_ID <int>,  
## # DEST\_AIRPORT\_SEQ\_ID <int>, DEST <chr>, DEST\_CITY\_NAME <chr>,  
## # DEST\_STATE\_NM <chr>, DEP\_TIME <chr>, DEP\_DELAY <dbl>, ARR\_TIME <chr>,  
## # ARR\_DELAY <dbl>, AIR\_TIME <dbl>, FLIGHTS <dbl>, DISTANCE <dbl>,  
## # X24 <chr>

planes %>% head(1)

## # A tibble: 1 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… EMBRAER EMB-1… 2 55 NA Turbo…

planes2 <- planes %>% rename( YEAR = year )%>%   
 rename( TAIL\_NUM =tailnum )  
  
planes2 %>% left\_join(Calfornia2018, by = c("TAIL\_NUM" = "TAIL\_NUM")) %>%  
 filter(ORIGIN == "OAK" )%>%  
 group\_by(TAIL\_NUM,ORIGIN) %>%  
 count()

## # A tibble: 1,025 x 3  
## # Groups: TAIL\_NUM, ORIGIN [1,025]  
## TAIL\_NUM ORIGIN n  
## <chr> <chr> <int>  
## 1 N103US OAK 3  
## 2 N104UW OAK 2  
## 3 N107US OAK 4  
## 4 N109UW OAK 1  
## 5 N110UW OAK 1  
## 6 N111US OAK 2  
## 7 N112US OAK 2  
## 8 N117UW OAK 3  
## 9 N118US OAK 1  
## 10 N119US OAK 2  
## # ... with 1,015 more rows

# Part II.

Choice II: Find a topic of interest to you and find a data source where you can download relevant data. The data should not be too larger, it should be manageable. Develop a list of questions to be explored with your data. Your analysis should be similar to the analysis suggested in Choice 1.

**Answer :** for my second part of project , i used the Violations data set in the mdsr package contains information regarding the out- come of health inspections of restaurants in New York City. It is a data frame with 480,621 observations on the following 16 variables.

library(mdsr)

Violations

## # A tibble: 480,621 x 16  
## camis dba boro building street zipcode phone inspection\_date   
## <int> <chr> <chr> <int> <chr> <int> <dbl> <dttm>   
## 1 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2015-02-09 00:00:00  
## 2 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2014-03-03 00:00:00  
## 3 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-10-10 00:00:00  
## 4 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-09-11 00:00:00  
## 5 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-09-11 00:00:00  
## 6 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-08-14 00:00:00  
## 7 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-08-14 00:00:00  
## 8 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-08-14 00:00:00  
## 9 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-08-14 00:00:00  
## 10 3.01e7 MORR… BRONX 1007 MORRI… 10462 7.19e9 2013-08-14 00:00:00  
## # ... with 480,611 more rows, and 8 more variables: action <chr>,  
## # violation\_code <chr>, score <int>, grade <chr>, grade\_date <dttm>,  
## # record\_date <dttm>, inspection\_type <chr>, cuisine\_code <dbl>

## 2.What new variables do you now also have. Make a table of the variables that are in the new dataset with a description of each variable.

**Answer :**In Violation data set I have 16 variables , below is variable name with its description .

|  |  |
| --- | --- |
| Variable name | variable discription |
| camis | unique identifier |
| dba | full name doing business as |
| boro | borough of New York |
| building | building name |
| street | street address |
| zipcode | zipcode |
| phone | phone number |
| inspection\_date | inspection date |
| action | action taken |
| violation\_code | violation code |
| score | inspection score |
| grade | inspection grade |
| grade\_date | grade date |
| record\_date | recording date |
| inspection\_type | inspect type |
| cuisine\_code | cuisine code |

## 3. How many cuisine are graded with A ,B ,C ?

**Answer :** There were 162362 cuisines graded with A , 40121 cuisines graded with B and 11501 graded with C .

Violations %>%filter(grade=="A") %>%  
 group\_by(grade)%>%  
 count()

## # A tibble: 1 x 2  
## # Groups: grade [1]  
## grade n  
## <chr> <int>  
## 1 A 162362

Violations %>%filter(grade=="C") %>%  
 group\_by(grade)%>%  
 count()

## # A tibble: 1 x 2  
## # Groups: grade [1]  
## grade n  
## <chr> <int>  
## 1 C 11501

Violations %>%filter(grade=="B") %>%  
 group\_by(grade)%>%  
 count()

## # A tibble: 1 x 2  
## # Groups: grade [1]  
## grade n  
## <chr> <int>  
## 1 B 40121

## 4. which zipcode had the highest inspection score ?and which zipcode had the lowest inspection score ?

zipcode number 10022 had the highest inspection score that is 156 and zipcode number 10165 had the lowest inspection score that is -2 .

Violations %>% select( zipcode , score ) %>%  
 group\_by(zipcode)%>%  
 summarize( n=n(), score\_max=max(score, na.rm = TRUE) )%>%  
 arrange(desc(score\_max))

## # A tibble: 229 x 3  
## zipcode n score\_max  
## <int> <int> <int>  
## 1 10022 8853 156  
## 2 11372 7434 132  
## 3 10019 12007 131  
## 4 10002 9113 122  
## 5 10018 6414 121  
## 6 11354 7827 121  
## 7 10036 11226 117  
## 8 11226 3994 115  
## 9 11413 721 115  
## 10 10006 977 111  
## # ... with 219 more rows

Violations %>% select( zipcode , score ) %>%  
 group\_by(zipcode)%>%  
 summarize( n=n(), score\_min=min(score, na.rm = TRUE) )%>%  
 arrange((score\_min))

## # A tibble: 229 x 3  
## zipcode n score\_min  
## <int> <int> <int>  
## 1 11218 2941 -2  
## 2 10001 8420 -1  
## 3 10002 9113 -1  
## 4 10010 4658 -1  
## 5 10011 8790 -1  
## 6 10012 8048 -1  
## 7 10013 10152 -1  
## 8 10014 8440 -1  
## 9 10016 9083 -1  
## 10 10017 6730 -1  
## # ... with 219 more rows

## 5. #. Of the cuisines with grade of A and B, what proportion of these cuisines are between the score of 20 or more and less than 40 ?

**Answer:** 11.31 % of score which is from 20 to 40

Violations%>% filter(grade == "A" |grade == "B" ) %>%  
 summarize(n=n(), mean( score >= 20& score < 40 ))

## # A tibble: 1 x 2  
## n `mean(score >= 20 & score < 40)`  
## <int> <dbl>  
## 1 202483 0.113

## 6. Which dba(full name doing business as) had the highest number of grade with A ?

**Answer:** DUNKIN’ DONUTS is the highest number of grade with A which is 2936

Violations %>%   
 group\_by(dba,grade) %>%   
 count() %>%   
 arrange(desc(n))

## # A tibble: 48,108 x 3  
## # Groups: dba, grade [48,108]  
## dba grade n  
## <chr> <chr> <int>  
## 1 DUNKIN' DONUTS A 2936  
## 2 SUBWAY A 2638  
## 3 SUBWAY <NA> 2070  
## 4 MCDONALD'S <NA> 1966  
## 5 MCDONALD'S A 1877  
## 6 DUNKIN' DONUTS <NA> 1831  
## 7 STARBUCKS A 1709  
## 8 DUNKIN' DONUTS, BASKIN ROBBINS A 1004  
## 9 CROWN FRIED CHICKEN <NA> 923  
## 10 DUNKIN' DONUTS, BASKIN ROBBINS <NA> 811  
## # ... with 48,098 more rows

## 7.What was the mean score of the cuisine? What was the mean score of the cuisine that is graded with A? What was the mean score of the cuisine that is graded with B?What was the mean score of the cuisine that is graded with C ?

**Answer:** The mean score of cuisines was 20.042, the mean score of cuisines that was graded by A is 9.799 ,the mean score of cuisines that was graded by B is 20.569 and the mean score of cuisines that was graded by C is 36.987 .

Violations %>% select(score) %>%  
 summarize(n=n(), score\_mean=mean(score, na.rm=TRUE))

## # A tibble: 1 x 2  
## n score\_mean  
## <int> <dbl>  
## 1 480621 20.0

Violations%>% select(grade, score) %>%   
 group\_by(grade) %>%  
 summarize(score\_mean=mean(score, na.rm=TRUE))

## # A tibble: 7 x 2  
## grade score\_mean  
## <chr> <dbl>  
## 1 A 9.80  
## 2 B 20.6   
## 3 C 37.0   
## 4 Not Yet Graded 22.7   
## 5 P 6.97  
## 6 Z 25.3   
## 7 <NA> 26.4

## 8.What Type of grade more?

**Answer:** grade type with A more than other grade type .

Violations%>%   
 group\_by(grade) %>%  
 summarise(n=n())

## # A tibble: 7 x 2  
## grade n  
## <chr> <int>  
## 1 A 162362  
## 2 B 40121  
## 3 C 11501  
## 4 Not Yet Graded 1217  
## 5 P 1803  
## 6 Z 3329  
## 7 <NA> 260288