Homerwork 1

Jesus Anderson Tuesta Soto

5/14/23

# Data Manipulation

## Problem 1: Use logical operators to find flights that:

- Had an arrival delay of two or more hours (\> 120 minutes)  
- Flew to Houston (IAH or HOU)  
- Were operated by United (`UA`), American (`AA`), or Delta (`DL`)  
- Departed in summer (July, August, and September)  
- Arrived more than two hours late, but didn't leave late  
- Were delayed by at least an hour, but made up over 30 minutes in flight

# Had an arrival delay of two or more hours (> 120 minutes)  
dplyr::filter(flights, arr\_delay >= 2) #check if arriva delay > 2 hours

# A tibble: 127,929 × 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 1 554 558 -4 740 728  
 5 2013 1 1 555 600 -5 913 854  
 6 2013 1 1 558 600 -2 753 745  
 7 2013 1 1 558 600 -2 924 917  
 8 2013 1 1 559 600 -1 941 910  
 9 2013 1 1 600 600 0 837 825  
10 2013 1 1 602 605 -3 821 805  
# ℹ 127,919 more rows  
# ℹ 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>

# Flew to Houston (IAH or HOU)  
dplyr::filter(flights,   
 dest == "IAH" | dest == "HOU") #Check if destiny is 'IAH' or 'HOU'

# A tibble: 9,313 × 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 1 623 627 -4 933 932  
 4 2013 1 1 728 732 -4 1041 1038  
 5 2013 1 1 739 739 0 1104 1038  
 6 2013 1 1 908 908 0 1228 1219  
 7 2013 1 1 1028 1026 2 1350 1339  
 8 2013 1 1 1044 1045 -1 1352 1351  
 9 2013 1 1 1114 900 134 1447 1222  
10 2013 1 1 1205 1200 5 1503 1505  
# ℹ 9,303 more rows  
# ℹ 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>

# Were operated by United (`UA`), American (`AA`), or Delta (`DL`)  
dplyr::filter(flights,   
 carrier == "UA" | carrier == "AA" | carrier == "DL")

# A tibble: 139,504 × 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 1 554 600 -6 812 837  
 5 2013 1 1 554 558 -4 740 728  
 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 1 559 600 -1 941 910  
10 2013 1 1 559 600 -1 854 902  
# ℹ 139,494 more rows  
# ℹ 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>

#Check if the carrier is UA, AA or DL

# Departed in summer (July, August, and September)  
dplyr::filter(flights,   
 month >=7 & month <=9) #Check if the month is between Jul and Sep

# A tibble: 86,326 × 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 7 1 1 2029 212 236 2359  
 2 2013 7 1 2 2359 3 344 344  
 3 2013 7 1 29 2245 104 151 1  
 4 2013 7 1 43 2130 193 322 14  
 5 2013 7 1 44 2150 174 300 100  
 6 2013 7 1 46 2051 235 304 2358  
 7 2013 7 1 48 2001 287 308 2305  
 8 2013 7 1 58 2155 183 335 43  
 9 2013 7 1 100 2146 194 327 30  
10 2013 7 1 100 2245 135 337 135  
# ℹ 86,316 more rows  
# ℹ 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>

# Arrived more than two hours late, but didn't leave late  
dplyr::filter(flights,   
 arr\_delay >= 2 & dep\_delay<=0) #Arrive 2 hours late (arr\_delay >= 2)

# A tibble: 37,527 × 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 1 554 558 -4 740 728  
 2 2013 1 1 555 600 -5 913 854  
 3 2013 1 1 558 600 -2 753 745  
 4 2013 1 1 558 600 -2 924 917  
 5 2013 1 1 559 600 -1 941 910  
 6 2013 1 1 600 600 0 837 825  
 7 2013 1 1 602 605 -3 821 805  
 8 2013 1 1 622 630 -8 1017 1014  
 9 2013 1 1 624 630 -6 909 840  
10 2013 1 1 624 630 -6 840 830  
# ℹ 37,517 more rows  
# ℹ 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>

#Didnt leave late (dep\_delay <= 0)

# Were delayed by at least an hour, but made up over 30 minutes in flight  
flights %>%   
 filter(dep\_delay>=1 & arr\_delay<=0.5) #Delay at least 1 hour dep\_delay>=1

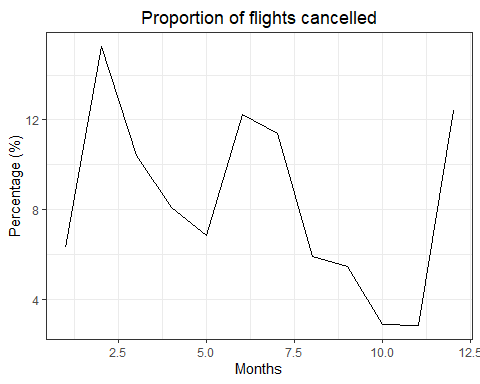
# A tibble: 35,442 × 19  
 year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
 <int> <int> <int> <int> <int> <dbl> <int> <int>  
 1 2013 1 1 601 600 1 844 850  
 2 2013 1 1 644 636 8 931 940  
 3 2013 1 1 646 645 1 910 916  
 4 2013 1 1 646 645 1 1023 1030  
 5 2013 1 1 701 700 1 1123 1154  
 6 2013 1 1 752 750 2 1025 1029  
 7 2013 1 1 803 800 3 1132 1144  
 8 2013 1 1 826 817 9 1145 1158  
 9 2013 1 1 846 845 1 1138 1205  
10 2013 1 1 856 855 1 1140 1203  
# ℹ 35,432 more rows  
# ℹ 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>

#Made up over 30 min in flight arr\_delay<=0.5

## Problem 2: What months had the highest and lowest proportion of cancelled flights? Interpret any seasonal patterns. To determine if a flight was cancelled use the following code

flights %>%   
 filter(is.na(dep\_time))

# What months had the highest and lowest % of cancelled flights?  
  
cancelled<-flights %>%   
 filter(is.na(dep\_time)) %>%   
 group\_by(month) %>%   
 summarise(cancelled\_flights=n()/nrow(.))  
  
cancelled %>%   
 ggplot(aes(x=month, y=cancelled\_flights\*100)) +  
 geom\_line() + theme\_bw() +  
 labs(x="Months", y="Percentage (%)", title='Proportion of flights cancelled') +  
 theme(plot.title = element\_text(hjust=0.5, vjust=0.5) )



The month with the highest proportion of cancelled flights is February, whereas November experiences the lowest number of cancellations. It appears that the majority of cancellations occur at the end and beginning of the year, gradually decreasing in the subsequent months.

## Problem 3: What plane (specified by the tailnum variable) traveled the most times from New York City airports in 2013? Please left\_join() the resulting table with the table planes (also included in the nycflights13 package).

For the plane with the greatest number of flights and that had more than 50 seats, please create a table where it flew to during 2013.

#Slice the plane dataset to find all the planes traveling from 2013 NYC  
plane1 <- flights %>%   
 filter(year==2013) %>% #Filter flights in 2013  
 filter(complete.cases(tailnum)) %>% #Non missing plane name  
 group\_by(tailnum) %>%   
 summarise(flightNY=n()) %>% #Calculate the flights done by plane  
 arrange(desc(flightNY)) #Sort it sorted descendingly  
  
#Show the plane with most flights from NYC in 2013  
plane1 %>%   
 slice(1)

# A tibble: 1 × 2  
 tailnum flightNY  
 <chr> <int>  
1 N725MQ 575

#Left joint the resulting table with planes dataset  
left\_join(planes, plane1, by="tailnum" )

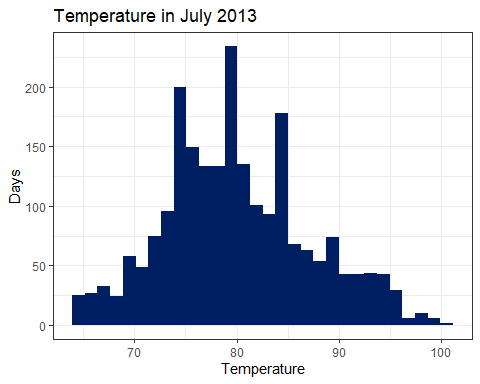
# A tibble: 3,322 × 10  
 tailnum year type manufacturer model engines seats speed engine flightNY  
 <chr> <int> <chr> <chr> <chr> <int> <int> <int> <chr> <int>  
 1 N10156 2004 Fixed w… EMBRAER EMB-… 2 55 NA Turbo… 153  
 2 N102UW 1998 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 48  
 3 N103US 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 46  
 4 N104UW 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 47  
 5 N10575 2002 Fixed w… EMBRAER EMB-… 2 55 NA Turbo… 289  
 6 N105UW 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 45  
 7 N107US 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 41  
 8 N108UW 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 60  
 9 N109UW 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 48  
10 N110UW 1999 Fixed w… AIRBUS INDU… A320… 2 182 NA Turbo… 40  
# ℹ 3,312 more rows

## Problem 4: The nycflights13 package includes a table (weather) that describes the weather during 2013. Use that table to answer the following questions:

- What is the distribution of temperature (`temp`) in July 2013? Identify any important outliers in terms of the `wind\_speed` variable.  
- What is the relationship between `dewp` and `humid`?  
- What is the relationship between `precip` and `visib`?

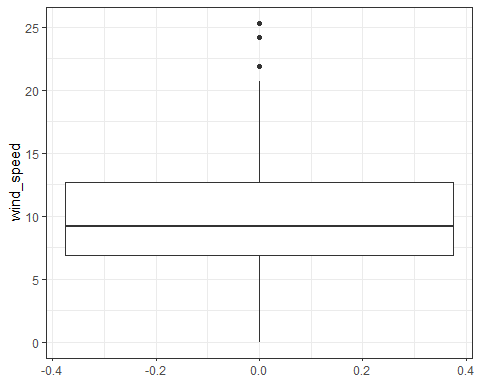
#Distribution of temperature in July 2013  
weather %>%   
 filter(year==2013 & month==7) %>% #Filter to year 2013 and July  
 ggplot(aes(x=temp)) + #Set the temperature distribution  
 geom\_histogram(fill='#001e62') + theme\_bw() + #Make a histogram  
 labs(title='Temperature in July 2013', y='Days', x='Temperature') #Add the labels

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



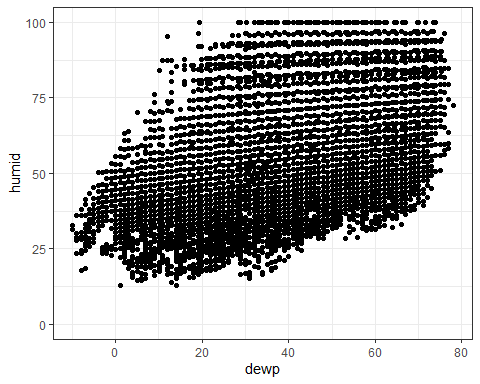
#We can observer 3 outliers from the wind variable  
weather %>%   
 filter(year==2013 & month==7) %>% #Filter to flights in July 2013  
 ggplot(aes(y=wind\_speed)) + #Analyse the wind dataset  
 geom\_boxplot() + theme\_bw() #Make a boxplot and set the theme

Warning: Removed 2 rows containing non-finite values (`stat\_boxplot()`).



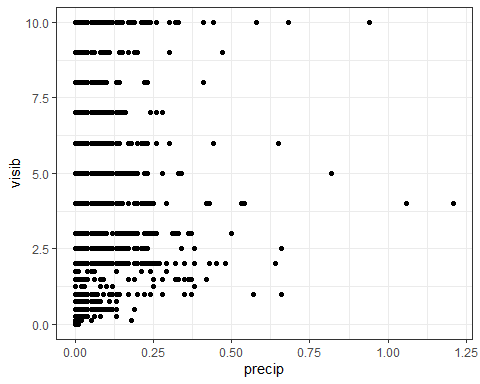
#What is the relationship between `dewp` and `humid`?  
  
#Relation between dewp and humid  
weather %>%   
 filter(year==2013) %>% #Filter for year 2013   
 ggplot(aes(x=dewp, y=humid)) + #Set the graph dewp vs humid  
 geom\_point() + theme\_bw() + #Make a scatter plot  
 ylim(0,100) #Set the limits in the y-scale

Warning: Removed 1 rows containing missing values (`geom\_point()`).



# We observe that there is no discernible relationship between dewp and humidity as indicated by the graph, which resemble a horizontal line.

#Relation between precip and visib  
weather %>%   
 filter(year==2013) %>%   
 ggplot(aes(x=precip, y=visib)) +  
 geom\_point() + theme\_bw()



# We observe that there is no discernible relationship between precipitation and visibility as indicated by the graph, which resemble a vertical line.

## Problem 5: Use the flights and planes tables to answer the following questions:

- How many planes have a missing date of manufacture?  
- What are the five most common manufacturers?  
- Has the distribution of manufacturer changed over time as reflected by the airplanes flying from NYC in 2013? (Hint: you may need to use case\_when() to recode the manufacturer name and collapse rare vendors into a category called Other.)

#How many planes have a missing date of manufacture?  
#Missing date of manufacture  
planes %>%   
 filter(is.na(year)) %>% #Find the planes with missings   
 summarise(no\_date=n()) #Count them

# A tibble: 1 × 1  
 no\_date  
 <int>  
1 70

#What are the five most common manufacturers?  
planes %>%   
 group\_by(manufacturer) %>% #Sorted by manufacturer  
 summarise(planes\_produce=n()) %>% #Estimate planes produced by manufacturer  
 top\_n(5, planes\_produce) %>% #Take the five highest  
 arrange(desc(planes\_produce)) #Sorted by planes produced

# A tibble: 5 × 2  
 manufacturer planes\_produce  
 <chr> <int>  
1 BOEING 1630  
2 AIRBUS INDUSTRIE 400  
3 BOMBARDIER INC 368  
4 AIRBUS 336  
5 EMBRAER 299

#Distribution of the total airplanes  
planes %>%   
 group\_by(manufacturer) %>% #Sor by manufacturer  
 summarise(planes\_produce=n()/nrow(.)) %>% #Percentage of planes produced   
 top\_n(5, planes\_produce) %>% #Take the top 5  
 arrange(desc(planes\_produce)) #Sort it

# A tibble: 5 × 2  
 manufacturer planes\_produce  
 <chr> <dbl>  
1 BOEING 0.491   
2 AIRBUS INDUSTRIE 0.120   
3 BOMBARDIER INC 0.111   
4 AIRBUS 0.101   
5 EMBRAER 0.0900

#New manufacturer distribution  
#The most common manufacters  
cat\_to\_keep=c('BOEING', 'AIRBUS INDUSTRIE', 'BOMBARDIER INC', 'AIRBUS', 'EMBRAER')  
  
#Mkae a column recoding the manufacturers to show the monst commons ones  
planes2<-planes %>%  
 mutate(manufac2=ifelse(manufacturer %in% cat\_to\_keep, manufacturer,'OTHER') )  
  
#Merge it with the fligths dataset  
flights2<-left\_join(flights, planes2, by='tailnum')  
  
#Slice the new dataset to find the new proportion of manufacturers  
flights2 %>%  
 filter(complete.cases(manufac2)) %>% #Only show cases where manufacturer is complete  
 group\_by(manufac2) %>%   
 summarise(planes\_produce=n()/nrow(.)) %>% #Find the proportion of planes produce by manufacturer   
 arrange(desc(planes\_produce))

# A tibble: 6 × 2  
 manufac2 planes\_produce  
 <chr> <dbl>  
1 BOEING 0.292   
2 EMBRAER 0.232   
3 AIRBUS 0.166   
4 AIRBUS INDUSTRIE 0.144   
5 BOMBARDIER INC 0.0995  
6 OTHER 0.0659

#The distribution has undergone a significant change, with Boeing's share of total production dropping from nearly 50% to only 30%. On the other hand, Embraer's share increased from 11% to 23%, placing them in the lead. Overall, the market has become more competitive, with market share being more evenly distributed among manufacturers.

## Problem 6: Use the flights and planes tables to answer the following questions:

- What is the oldest plane (specified by the tailnum variable) that flew from New York City airports in 2013?  
- How many airplanes that flew from New York City are included in the planes table?

#What is the oldest plane (specified by the tailnum variable) that flew from New York City airports in 2013?  
flights2 %>%  
 arrange(year.y) %>% #Sort it by year of production   
 select(tailnum) %>% #Select the plane names  
 slice\_head(n=1) #Take the highest observation

# A tibble: 1 × 1  
 tailnum  
 <chr>   
1 N381AA

#How many airplanes that flew from New York City are included in the planes table?  
flights2 %>%   
 filter(!is.na(manufac2)) %>% #Filter for flights that have a mannufacturer in plane dataset  
 summarise(airplanes\_included=length(unique(tailnum))) #Count the uniques observations

# A tibble: 1 × 1  
 airplanes\_included  
 <int>  
1 3322

## Problem 7: Use the nycflights13 to answer the following questions:

- What is the median arrival delay on a month-by-month basis in each airport?  
- For each airline, plot the median arrival delay for each month and origin airport.

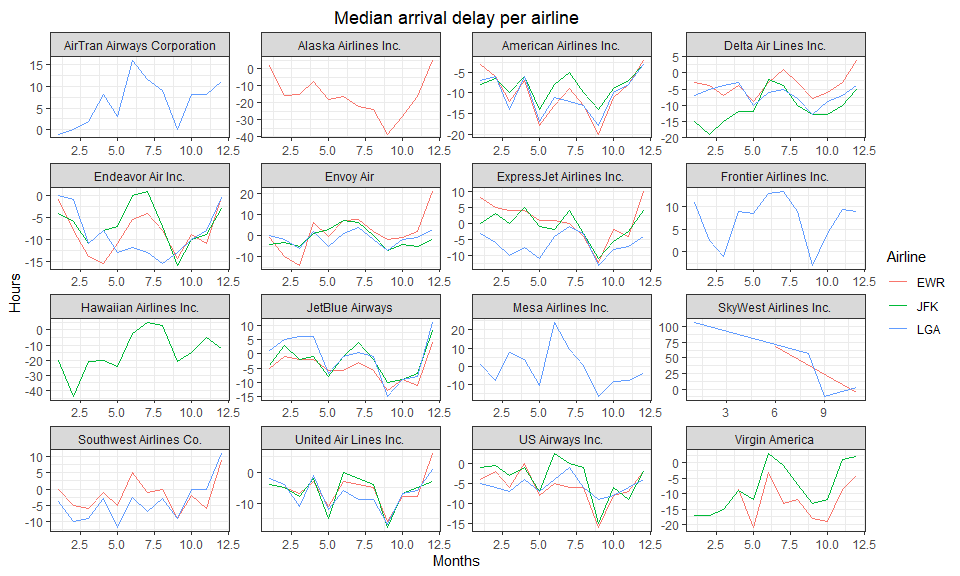
#What is the median arrival delay on a month-by-month basis in each airport?  
flights %>%   
 filter(!is.na(arr\_delay)) %>% #Filter only rows with arr\_dealy values  
 group\_by(origin, month) %>%   
 summarise(me\_delay=median(arr\_delay)) #Find the median arr\_delay by month and origin airport

`summarise()` has grouped output by 'origin'. You can override using the  
`.groups` argument.

# A tibble: 36 × 3  
# Groups: origin [3]  
 origin month me\_delay  
 <chr> <int> <dbl>  
 1 EWR 1 0  
 2 EWR 2 -2  
 3 EWR 3 -4  
 4 EWR 4 -1  
 5 EWR 5 -6  
 6 EWR 6 -1  
 7 EWR 7 -2  
 8 EWR 8 -5  
 9 EWR 9 -13  
10 EWR 10 -6  
# ℹ 26 more rows

#For each airline, plot the median arrival delay for each month and origin airport.  
  
flights3<-left\_join(flights, airlines, by='carrier') #Merge flights and airlines datasets  
  
flights3 %>%   
 filter(!is.na(arr\_delay)) %>% #Filter the observations with no value in arr\_dealy  
 filter(!is.na(carrier)) %>% #Filter obs with no value in carrier  
 group\_by(name, origin, month) %>%   
 summarise(me\_delay=median(arr\_delay)) %>% #Calculate the median delay by origin, airline name and month  
 ggplot(aes(x=month, y=me\_delay,color=origin)) + #set the aesthetics  
 geom\_line() + #Make the line graph  
 facet\_wrap(~name, scales = "free") + #Repeat it by airline name  
 theme\_bw() + #Theme black and white  
 labs(title='Median arrival delay per airline', x='Months', y='Hours',color='Airline') + #Add the lables  
 theme(plot.title = element\_text(hjust=0.5, vjust=0.5) ) #Put the title in the middle

`summarise()` has grouped output by 'name', 'origin'. You can override using  
the `.groups` argument.



## Problem 8: Let’s take a closer look at what carriers service the route to San Francisco International (SFO). Join the flights and airlines tables and count which airlines flew the most to SFO. Produce a new dataframe, fly\_into\_sfo that contains three variables: the name of the airline, e.g., United Air Lines Inc. not UA, the count (number) of times it flew to SFO, and the percent of the trips that that particular airline flew to SFO.

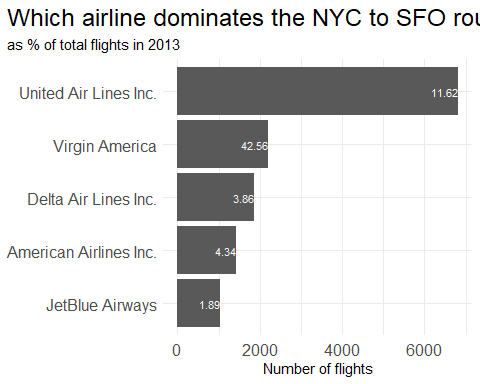
#Find which airline flight the most to SFO  
fly\_into\_sfo<-flights3 %>%   
 filter(dest=='SFO') %>% #Flights to SFO  
 group\_by(name) %>%   
 summarise(count=n()) #Count the flights to SFO by airlines

#Pecentage of the trips that particular airline flew to SFO  
fly\_into\_sfo2<-flights3 %>%   
 mutate(SFO=ifelse(dest=='SFO',1,0)) %>% #New column with 1 if flight to SFO  
 group\_by(name) %>% #Group by airline  
 summarise(percent=100\*sum(SFO)/n()) %>% #Fing the percentage in which airlines flight to SFO   
 mutate(percent=format(percent, digits = 2, nsmall=2)) #Improve the format

#Produce a new dataframe, fly\_into\_sfo that contains three variables: the name of the airline, e.g., United Air Lines Inc.   
fly\_into\_sfo<-left\_join(fly\_into\_sfo, fly\_into\_sfo2, by='name')

And here is some bonus ggplot code to plot your dataframe

fly\_into\_sfo %>%   
   
 # sort 'name' of airline by the numbers it times to flew to SFO  
 mutate(name = fct\_reorder(name, count)) %>%   
   
 ggplot() +  
   
 aes(x = count,   
 y = name) +  
   
 # a simple bar/column plot  
 geom\_col() +  
   
 # add labels, so each bar shows the % of total flights   
 geom\_text(aes(label = percent),  
 hjust = 1,   
 colour = "white",   
 size = 3)+  
   
 # add labels to help our audience   
 labs(title="Which airline dominates the NYC to SFO route?",   
 subtitle = "as % of total flights in 2013",  
 x= "Number of flights",  
 y= NULL) +  
   
 theme\_minimal() +   
   
 # change the theme-- i just googled those , but you can use the ggThemeAssist add-in  
 # https://cran.r-project.org/web/packages/ggThemeAssist/index.html  
   
 theme(#  
 # so title is left-aligned  
 plot.title.position = "plot",  
   
 # text in axes appears larger   
 axis.text = element\_text(size=12),  
   
 # title text is bigger  
 plot.title = element\_text(size=18)  
 ) +  
  
 # add one final layer of NULL, so if you comment out any lines  
 # you never end up with a hanging `+` that awaits another ggplot layer  
 NULL



## Problem 9: Let’s take a look at cancellations of flights to SFO. We create a new dataframe cancellations as follows

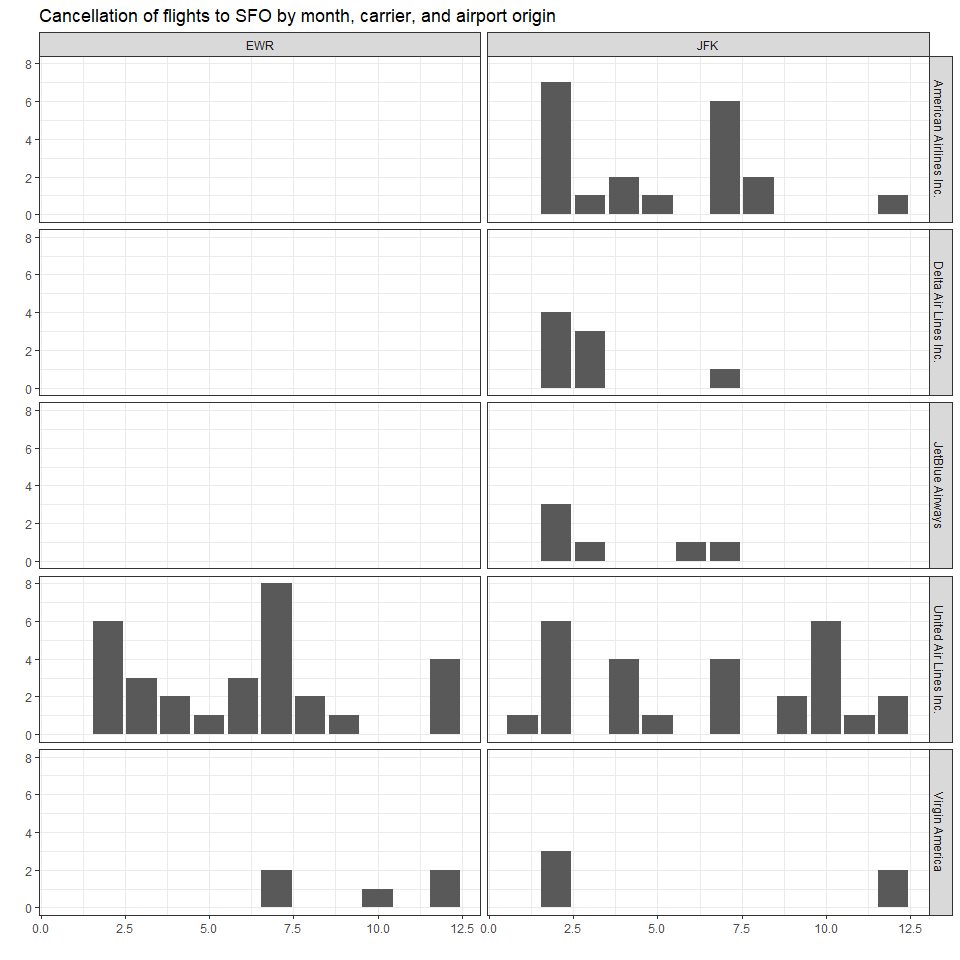
cancellations <- flights %>%   
   
 # just filter for destination == 'SFO'  
 filter(dest == 'SFO') %>%   
   
 # a cancelled flight is one with no `dep\_time`   
 filter(is.na(dep\_time))

I want you to think how we would organise our data manipulation to create the following plot. No need to write the code, just explain in words how you would go about it.



#Replicating the graph above  
#We need to merge the cancellations dataset and airlines to get the airlines names  
cancellations<-left\_join(cancellations, airlines, by='carrier')  
  
#Make the graph  
cancellations %>%   
 filter(origin=='EWR' | origin=='JFK') %>% #Only save the EWR and JFK flights   
 group\_by(name, month, origin) %>%   
 summarise(flight\_can=n()) %>% #Calculated the number of flights per airline, month and airport origin   
 ggplot(aes(x=month, y=flight\_can)) + #Set the graph   
 geom\_col() + #Graph a column table  
 facet\_wrap(origin ~ name, scales='free') + #Repeated by name and origin airport  
 facet\_grid(name ~ origin) +  
 theme\_bw() +labs(x='', y='', title='Cancellation of flights to SFO by month, carrier, and airport origin') #Set the theme and eliminate the lables

`summarise()` has grouped output by 'name', 'month'. You can override using the  
`.groups` argument.



#The only thing missing is to set the format for the x-axis

## Problem 10: On your own – Hollywood Age Gap

age\_gaps <- readr::read\_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-02-14/age\_gaps.csv')

Rows: 1155 Columns: 13  
── Column specification ────────────────────────────────────────────────────────  
Delimiter: ","  
chr (6): movie\_name, director, actor\_1\_name, actor\_2\_name, character\_1\_gend...  
dbl (5): release\_year, age\_difference, couple\_number, actor\_1\_age, actor\_2\_age  
date (2): actor\_1\_birthdate, actor\_2\_birthdate  
  
ℹ Use `spec()` to retrieve the full column specification for this data.  
ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

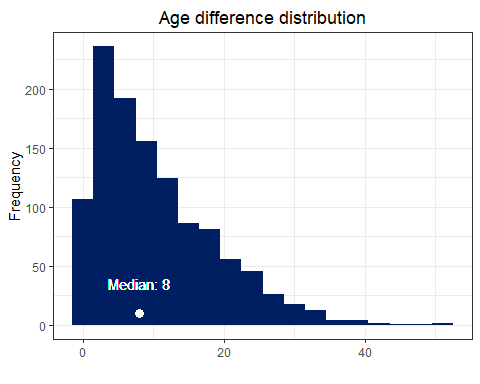
How would you explore this data set? Here are some ideas of tables/ graphs to help you with your analysis

* How is age\_difference distributed? What’s the ‘typical’ age\_difference in movies?
* The half plus seven\ rule. Large age disparities in relationships carry certain stigmas. One popular rule of thumb is the [half-your-age-plus-seven](https://en.wikipedia.org/wiki/Age_disparity_in_sexual_relationships#The_.22half-your-age-plus-seven.22_rule) rule. This rule states you should never date anyone under half your age plus seven, establishing a minimum boundary on whom one can date. In order for a dating relationship to be acceptable under this rule, your partner’s age must be:

How frequently does this rule apply in this dataset?

* Which movie has the greatest number of love interests?
* Which actors/ actresses have the greatest number of love interests in this dataset?
* Is the mean/median age difference staying constant over the years (1935 - 2022)?
* How frequently does Hollywood depict same-gender love interests?

#How is age\_difference distributed? What's the 'typical' age\_difference in movies?  
age\_gaps %>%   
ggplot(aes(x = age\_difference)) + #Set the aesthetics  
 geom\_histogram(binwidth = 3, fill = "#001e62") + #Make a histogram  
 #Add a point in the median  
 geom\_point(aes(x = median(age\_difference), y = 10), color = "white", size = 3)+  
 #Add a text for the median value  
 geom\_text(aes(x = median(age\_difference), y = 20, label = paste0("Median: ", round(median(age\_difference), 2))),  
 color = "white", vjust = -1) +  
 labs(x = "", y = "Frequency", title = "Age difference distribution") +  
 theme\_bw() + #Add the labels  
 theme(plot.title = element\_text(hjust=0.5, vjust=0.5) ) #Put the title in the middle



#The half plus seven rule  
#The inequality given above is simetric showing that if it works for an actor 1, then it also works for actor 2. So, we only need to compute it once.   
#How frequently does this rule apply in this dataset?  
age\_gaps %>%   
 mutate(lim\_inf=(actor\_1\_age/2) + 7, #Set the lower limit  
 lim\_sup=2\*(actor\_1\_age-7), #Set the upper limit  
 #Check if the rule holds in the movie  
 half\_plus\_seven=ifelse(actor\_2\_age>lim\_inf & actor\_2\_age<lim\_sup,1,0)) %>% summarise(rule\_applied=sum(half\_plus\_seven)) #Sum all the cases in which the rule holds

# A tibble: 1 × 1  
 rule\_applied  
 <dbl>  
1 795

#Which movie has the greatest number of love interests?  
age\_gaps %>%   
 group\_by(movie\_name) %>%   
 summarise(love\_interest=n()) %>% #Number of love interest per movie  
 arrange(desc(love\_interest)) %>% #Sort it descendingly  
 top\_n(1, love\_interest) #Take the first observation

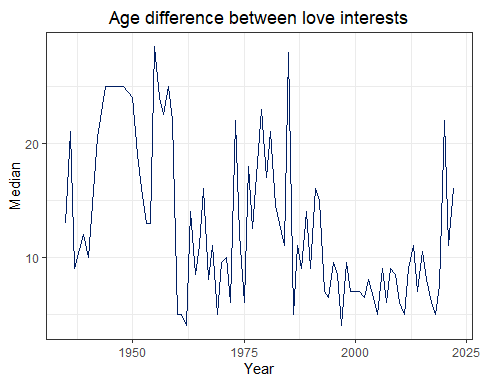
# A tibble: 1 × 2  
 movie\_name love\_interest  
 <chr> <int>  
1 Love Actually 7

#Which actors/ actresses have the greatest number of love interests in this dataset?  
age\_gaps %>%   
 #Pass from wide to long by actor  
 pivot\_longer(cols=ends\_with('name'), names\_to='type', values\_to='actor') %>%   
 #Count the number of romantic interest by actor  
 group\_by(actor) %>%   
 summarise(count=n()) %>%   
 #Sort it descendingly  
 arrange(desc(count)) %>%  
 #Take the higher observation  
 top\_n(1, count)

# A tibble: 1 × 2  
 actor count  
 <chr> <int>  
1 Keanu Reeves 27

#Is the mean/median age difference staying constant over the years (1935 - 2022)?  
age\_gaps %>%   
 #Find the media age difference by movie release year  
 group\_by(release\_year) %>% #   
 summarise(median\_dif=median(age\_difference)) %>%  
 #Make the graph  
 ggplot(aes(x=release\_year, y=median\_dif)) +  
 #Set the line graph  
 geom\_line(color='#001e62', size=0.7) +  
 #Set the theme  
 theme\_bw() +  
 #Add the labels  
 labs(x='Year', y='Median', title='Age difference between love interests') +  
 theme(plot.title = element\_text(hjust=0.5, vjust=0.5) )

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
ℹ Please use `linewidth` instead.



# Details

* Who did you collaborate with: Jesus Tuesta
* Approximately how much time did you spend on this problem set: 6 hours
* What, if anything, gave you the most trouble: First part