Homework 1

Saagar Hemrajani

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)  
del\_2hr <- flights %>%   
 filter(arr\_delay > 120)  
print(del\_2hr)

# A tibble: 10,034 × 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 1 811 630 101 1047 830  
 2 2013 1 1 848 1835 853 1001 1950  
 3 2013 1 1 957 733 144 1056 853  
 4 2013 1 1 1114 900 134 1447 1222  
 5 2013 1 1 1505 1310 115 1638 1431  
 6 2013 1 1 1525 1340 105 1831 1626  
 7 2013 1 1 1549 1445 64 1912 1656  
 8 2013 1 1 1558 1359 119 1718 1515  
 9 2013 1 1 1732 1630 62 2028 1825  
10 2013 1 1 1803 1620 103 2008 1750  
# ℹ 10,024 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)  
dest\_hous\_iah <- flights %>%   
 filter(dest %in% c("HOU", "IAH"))  
print(dest\_hous\_iah)

# 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`)  
opr\_UA\_AA\_DL <- flights %>%   
 filter(carrier %in% c("UA", "AA", "DL"))  
print(opr\_UA\_AA\_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>

# Departed in summer (July, August, and September)  
dep\_sum <- flights %>%   
 filter(month %in% c(7, 8, 9))  
print(dep\_sum)

# 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  
arr\_hr\_late <- flights %>%   
 filter(arr\_delay > 120 & dep\_delay <= 0)  
print(arr\_hr\_late)

# A tibble: 29 × 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 27 1419 1420 -1 1754 1550  
 2 2013 10 7 1350 1350 0 1736 1526  
 3 2013 10 7 1357 1359 -2 1858 1654  
 4 2013 10 16 657 700 -3 1258 1056  
 5 2013 11 1 658 700 -2 1329 1015  
 6 2013 3 18 1844 1847 -3 39 2219  
 7 2013 4 17 1635 1640 -5 2049 1845  
 8 2013 4 18 558 600 -2 1149 850  
 9 2013 4 18 655 700 -5 1213 950  
10 2013 5 22 1827 1830 -3 2217 2010  
# ℹ 19 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 delayed by at least an hour, but made up over 30 minutes in flight  
made\_up30<- flights %>%   
 filter((dep\_delay-arr\_delay) > 30 & (dep\_delay >= 60))  
print(made\_up30)

# A tibble: 1,844 × 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 2205 1720 285 46 2040  
 2 2013 1 1 2326 2130 116 131 18  
 3 2013 1 3 1503 1221 162 1803 1555  
 4 2013 1 3 1839 1700 99 2056 1950  
 5 2013 1 3 1850 1745 65 2148 2120  
 6 2013 1 3 1941 1759 102 2246 2139  
 7 2013 1 3 1950 1845 65 2228 2227  
 8 2013 1 3 2015 1915 60 2135 2111  
 9 2013 1 3 2257 2000 177 45 2224  
10 2013 1 4 1917 1700 137 2135 1950  
# ℹ 1,834 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>

## 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

# What months had the highest and lowest % of cancelled flights?  
  
# Filter to all flights that were cancelled and count per month  
cancelled <- flights %>%   
 filter(is.na(dep\_time)) %>%   
 group\_by(month) %>%   
 summarise(count\_cancel = n() )  
  
# Filter for all flights  
all\_flights\_count <- flights %>%   
 group\_by(month) %>%   
 summarise(count\_total = n() )  
  
#  
all\_flights\_count %>%   
 left\_join(cancelled, by = "month") %>%  
 mutate(prop = count\_cancel/count\_total ) %>%   
 arrange(desc(month))

# A tibble: 12 × 4  
 month count\_total count\_cancel prop  
 <int> <int> <int> <dbl>  
 1 12 28135 1025 0.0364   
 2 11 27268 233 0.00854  
 3 10 28889 236 0.00817  
 4 9 27574 452 0.0164   
 5 8 29327 486 0.0166   
 6 7 29425 940 0.0319   
 7 6 28243 1009 0.0357   
 8 5 28796 563 0.0196   
 9 4 28330 668 0.0236   
10 3 28834 861 0.0299   
11 2 24951 1261 0.0505   
12 1 27004 521 0.0193

## 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.

#Assuming that only JFK and LaGuardia qualify as new york airports  
  
flights %>%   
 # Filter out cancelled flghts  
 filter(!is.na(dep\_time) & year == 2013) %>%   
 # Filter NYC Takeoffs  
 filter(origin %in% c("JFK", "LGA")) %>%   
 # Left join on tailnum  
 left\_join(planes, by = "tailnum") %>%   
 # Group by tailnum  
 group\_by(tailnum) %>%   
 # Summarize by counting by plane  
 summarise(count\_nyc\_takeoff = n()) %>%   
 # Arrange in descending order  
 arrange(desc(count\_nyc\_takeoff)) %>%   
 slice(1)

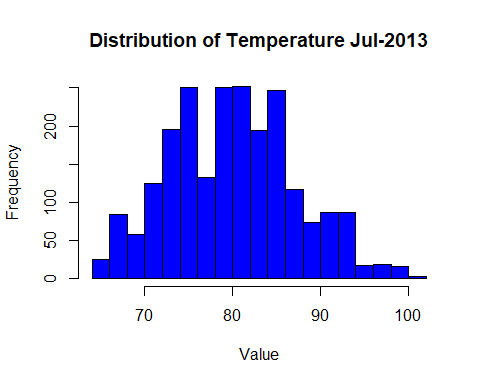
# A tibble: 1 × 2  
 tailnum count\_nyc\_takeoff  
 <chr> <int>  
1 N725MQ 546

# N725MQ traveled the most times from New York City airports in 2013

## 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`?

# Temperature in Jul  
temp\_jul\_2013 <- weather %>%   
 filter(month == 7) %>%   
 pull(temp)  
  
# Get histogram of temperature in july  
temp\_jul\_hist <- hist(temp\_jul\_2013, breaks = 15, col = "blue", xlab = "Value", ylab = "Frequency", main = "Distribution of Temperature Jul-2013")



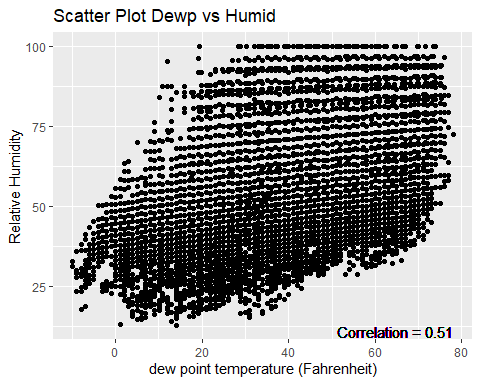
# Get all data where windspeed is not NA  
weather\_wind\_speed <-   
 weather %>%   
 filter(!is.na(wind\_speed))  
  
# Give me all wind speeds more than 3 standard deviations from the mean  
outlier\_wind\_speeds <- weather\_wind\_speed %>%   
 mutate(z = scale(wind\_speed)) %>%   
 filter(abs(z) > 3) %>%   
 select(origin, year, month, day, hour, wind\_speed)

Warning: Using one column matrices in `filter()` was deprecated in dplyr 1.1.0.  
ℹ Please use one dimensional logical vectors instead.

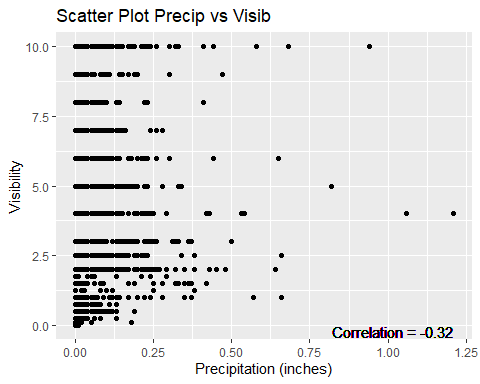
print(outlier\_wind\_speeds)

# A tibble: 10 × 6  
 origin year month day hour wind\_speed  
 <chr> <int> <int> <int> <int> <dbl>  
 1 EWR 2013 1 31 4 40.3  
 2 EWR 2013 1 31 6 42.6  
 3 EWR 2013 1 31 8 39.1  
 4 EWR 2013 2 12 3 1048.   
 5 JFK 2013 1 31 3 36.8  
 6 JFK 2013 1 31 4 42.6  
 7 JFK 2013 1 31 7 36.8  
 8 JFK 2013 3 6 14 38.0  
 9 JFK 2013 11 24 10 36.8  
10 LGA 2013 1 31 4 40.3

# Dataframe with dewp and humidity  
weather\_dewp\_humid <- weather %>%   
 filter(!is.na(dewp) & !is.na(humid)) %>%   
 select(dewp, humid)   
  
# Create scatter plot with correlation coefficient  
scatter\_dewp\_humid <- ggplot(weather\_dewp\_humid, aes(dewp, humid)) +  
 geom\_point() +  
 geom\_text(label = paste("Correlation =", round(cor(weather\_dewp\_humid$dewp, weather\_dewp\_humid$humid), 2)), x = max(weather\_dewp\_humid$dewp), y = min(weather\_dewp\_humid$humid), hjust = 1, vjust = 1) +  
 labs(title = "Scatter Plot Dewp vs Humid", x = "dew point temperature (Fahrenheit)", y = "Relative Humidity")  
  
plot(scatter\_dewp\_humid)



## Moderate positve correlation (0.51) between Dewp and Humidity  
  
# Dataframe with precip and visib  
weather\_precip\_visib <- weather %>%   
 filter(!is.na(precip) & !is.na(visib)) %>%   
 select(precip, visib)   
  
# Create scatter plot with correlation coefficient  
scatter\_precip\_visib <- ggplot(weather\_precip\_visib, aes(precip, visib)) +  
 geom\_point() +  
 geom\_text(label = paste("Correlation =", round(cor(weather\_precip\_visib$precip, weather\_precip\_visib$visib), 2)), x = max(weather\_precip\_visib$precip), y = min(weather\_precip\_visib$visib), hjust = 1, vjust = 1) +  
 labs(title = "Scatter Plot Precip vs Visib", x = "Precipitation (inches)", y = "Visibility")  
  
plot(scatter\_precip\_visib)



## No real correlation between preciptation and visibility

## 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.)

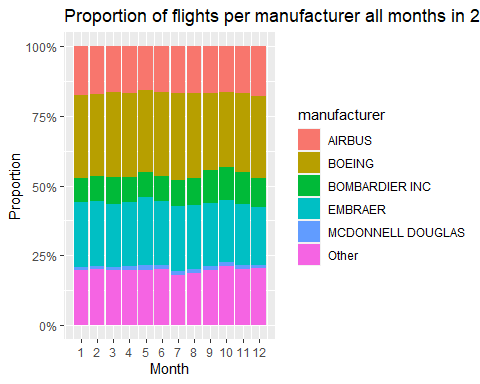
# Filter all planes where year of manufacture is NA   
missing\_manu\_plane <- filter(planes, is.na(year))  
nrow(missing\_manu\_plane) #70 planes have no manufacture date

[1] 70

# Group the planes data frame by manufacturer and count the occurrences  
manufacturer\_counts <- planes %>%   
 # Replace AIRBUS INDUSTRIE with AIRBUS as these refer to the same company  
 mutate(manufacturer = recode(manufacturer,   
 "AIRBUS INDUSTRIE" = "AIRBUS",  
 "MCDONNELL DOUGLAS AIRCRAFT CO" = "MCDONNELL DOUGLAS",  
 "MCDONNELL DOUGLAS CORPORATION" = "MCDONNELL DOUGLAS"   
 )) %>%  
 group\_by(manufacturer) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count))  
  
# Select the top five manufacturers by count  
top\_manufacturers <- top\_n(manufacturer\_counts, n = 5, wt = count)  
print(top\_manufacturers) #BOEING, AIRBUS, BOMBARDIER INC, EMBRAER,MCDONNELL DOUGLAS

# A tibble: 5 × 2  
 manufacturer count  
 <chr> <int>  
1 BOEING 1630  
2 AIRBUS 736  
3 BOMBARDIER INC 368  
4 EMBRAER 299  
5 MCDONNELL DOUGLAS 237

# Get top manufacturer as a list  
top\_manufacturer\_vector <- top\_manufacturers$manufacturer  
  
# Get all flights that werent cancelled  
# Get only month, flight\_no and tailnum  
not\_na\_flights <- flights %>%   
 filter(!is.na(dep\_time)) %>%   
 select (month, flight, tailnum)  
  
# Left join unclassed flights with all planes data  
# Join by tail number, filter out manufacturers that are NA  
flights\_with\_manufacturer <- left\_join(not\_na\_flights,   
 planes %>%   
 select (tailnum, manufacturer), by = "tailnum") %>%   
 filter(!is.na(manufacturer))  
  
# Mutate the manufacture column to be Other when not present  
# in the list of top 5 manufacuters  
# Group by month (in 2013) and manufacturer, count n in each group  
collapsed\_flights <- flights\_with\_manufacturer %>%  
 mutate(manufacturer = case\_when(  
 manufacturer %in% top\_manufacturer\_vector ~ as.character(manufacturer),  
 TRUE ~ "Other"  
 )) %>%  
 group\_by(month, manufacturer) %>%  
 summarise(count = n())  
  
# Calculate the total number of uncancelled flights per month  
flights\_per\_month <- flights\_with\_manufacturer %>%   
 group\_by(month) %>%  
 summarise(total\_flights\_month = n())  
  
# Join the two, divide flights per manufacturer by total flights  
# This gives us percentage og flights per month per manufacturer  
flights\_by\_manufacuter <- left\_join(collapsed\_flights,   
 flights\_per\_month, by = "month") %>%   
 mutate(manufacturer\_percentage = count/total\_flights\_month)  
  
# Create an stacked column chart  
stacked\_column\_manufacturer\_prop <- ggplot(flights\_by\_manufacuter, aes(x = month, y = manufacturer\_percentage, fill = manufacturer)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Proportion of flights per manufacturer all months in 2013", x = "Month", y = "Proportion") +  
 scale\_x\_continuous(breaks = seq(1,12, by = 1)) + #scale x axis as months 1-12  
 scale\_y\_continuous(labels = scales::percent) # Format y-axis labels as percentages  
  
plot(stacked\_column\_manufacturer\_prop)



## 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?

#Assuming that only JFK and LaGuardia qualify as new york airports  
  
# Get all planes that have taken off from NYC in 2013  
planes\_from\_nyc\_2013 <- flights %>%   
 # Filter out cancelled flghts  
 filter(!is.na(dep\_time) & year == 2013) %>%   
 # Filter NYC Takeoffs  
 filter(origin %in% c("JFK", "LGA")) %>%   
 # Drop duplicate tailnum  
 distinct(tailnum) %>%   
 # Select only tailnum column  
 pull(tailnum)  
  
# Get oldest planes  
oldest\_planes <- planes %>%   
 # Filter only planes that took off from nyc in 2013  
 # and where there exists a manufacture date  
 filter(tailnum %in% planes\_from\_nyc\_2013 & !is.na(year)) %>%   
 # Give me only ones with the lowest year  
 top\_n(n = 1, wt = -year) %>%   
 # Select only tailnum, manufacturer and year column  
 select(tailnum, manufacturer, year)  
  
print(oldest\_planes) # N381AA DOUGLAS 1956

# A tibble: 1 × 3  
 tailnum manufacturer year  
 <chr> <chr> <int>  
1 N381AA DOUGLAS 1956

# How many planes that flew from NYC in 2013 are in planes table  
num\_planes\_nyc\_2013 <- planes %>%   
 # Filter only planes that took off from nyc in 2013  
 filter(tailnum %in% planes\_from\_nyc\_2013) %>%   
 summarise(count = n())  
  
print(num\_planes\_nyc\_2013) # 2880 Planes in planes db, flew from NYC

# A tibble: 1 × 1  
 count  
 <int>  
1 2880

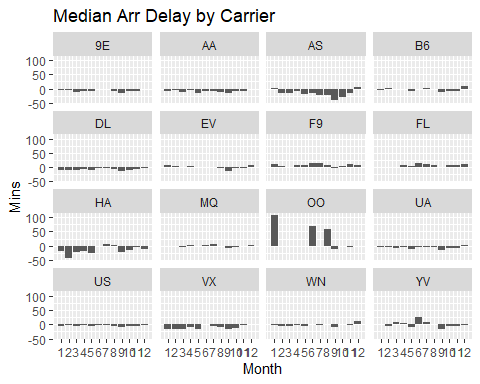
## Problem 7: Use the nycflights13 to answer the following questions:

- 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.

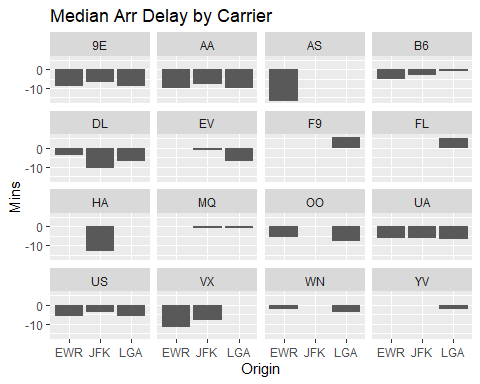
# Median Arrival Delay   
med\_arr\_delay\_by\_mon <- flights %>%   
 # Filter out cancelled flights  
 filter(!is.na(dep\_time) & year == 2013) %>%   
 # Group by destinato airport and month  
 group\_by(dest, month) %>%   
 # Calculate median arr\_delay  
 summarise(med\_arr\_delay = median(arr\_delay, na.rm=TRUE)) %>%   
 # Pivot table to make month in columns  
 pivot\_wider(names\_from = month, values\_from = med\_arr\_delay)  
  
print(med\_arr\_delay\_by\_mon)

# A tibble: 104 × 13  
# Groups: dest [104]  
 dest `4` `5` `6` `7` `8` `9` `10` `11` `12` `1` `2` `3`  
 <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 ABQ 14 -19 -2.5 -6 -14 -16 -10 -6 27 NA NA NA   
 2 ACK NA -8 -2 0.5 -6 -4 -1 NA NA NA NA NA   
 3 ALB 9 0 -14 -13 -19 -19 13 -1 0 6 -3 0.5  
 4 ANC NA NA NA 1.5 3 NA NA NA NA NA NA NA   
 5 ATL 2 -4 3 7 1 -6 -4 -1 3 -2 -1 -4   
 6 AUS -1.5 -9 -5.5 1 -5.5 -21 -9 -11 5 -2 2 -4   
 7 AVL -22 -10 -2 2.5 -2 -3 7 2.5 -2 23.5 NA NA   
 8 BDL -3 -11 -2 49 -14 -17 -11 8 5.5 -10 -11 -1   
 9 BGR 12 -21 -7 11 25 -12 -12.5 -9.5 -3 NA NA -22   
10 BHM -3 -23 11 21.5 -3 -17 4.5 5 31 -11 -10 -1   
# ℹ 94 more rows

med\_arr\_delay\_by\_carrier\_mon <- flights %>%   
 filter(!is.na(dep\_time) & year == 2013) %>%   
 group\_by(carrier, month) %>%  
 summarise(med\_arr\_delay = median(arr\_delay, na.rm=TRUE))   
  
plot\_med\_arr\_delay\_carrier\_mon <- ggplot(med\_arr\_delay\_by\_carrier\_mon,   
 aes(x = month, y = med\_arr\_delay)) +  
 geom\_bar(stat = "identity") +  
 facet\_wrap(. ~ carrier) +  
 scale\_x\_continuous(breaks = seq(1,12, by = 1)) + #scale x axis as months 1-12  
 labs(title = "Median Arr Delay by Carrier", x = "Month", y = "Mins")  
  
plot(plot\_med\_arr\_delay\_carrier\_mon)



med\_arr\_delay\_by\_carrier\_origin <- flights %>%   
 filter(!is.na(dep\_time) & year == 2013) %>%   
 group\_by(carrier, origin) %>%  
 summarise(med\_arr\_delay = median(arr\_delay, na.rm=TRUE))   
  
plot\_med\_arr\_delay\_carrier\_origin <- ggplot(med\_arr\_delay\_by\_carrier\_origin,   
 aes(x = origin, y = med\_arr\_delay)) +  
 geom\_bar(stat = "identity") +  
 facet\_wrap(. ~ carrier) +  
 labs(title = "Median Arr Delay by Carrier", x = "Origin", y = "Mins")  
  
plot(plot\_med\_arr\_delay\_carrier\_origin)

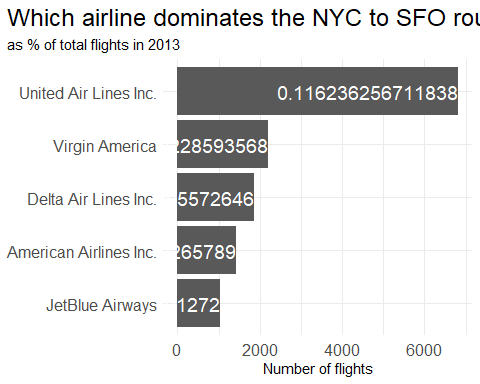


## Problem 8: Let’s take a closer look at what carriers service the route to San Francisco International (SFO). Join the flights and airlines tables and count which airlines flew the most to SFO. Produce a new dataframe, fly\_into\_sfo that contains three variables: the name of the airline, e.g., United 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.

# Join flights and airlines tables  
total\_flights\_by\_carriers <- flights %>%  
 left\_join(airlines, by = "carrier") %>%  
 group\_by(name) %>%  
 summarise(total\_flights = n())  
  
# Join flights and airlines tables  
fly\_into\_sfo <- flights %>%  
 left\_join(airlines, by = "carrier") %>%  
 filter(dest == "SFO") %>%  
 group\_by(name) %>%  
 summarise(count = n()) %>%   
 left\_join(total\_flights\_by\_carriers, by = "name") %>%   
 mutate(percent = count/total\_flights) %>%   
 select(name, count, percent) %>%   
 arrange(desc(percent))

And here is some bonus ggplot code to plot your dataframe

#| label: ggplot-flights-toSFO  
 #| message: false  
 #| warning: false  
  
 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 = 5)+  
   
 # 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))



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

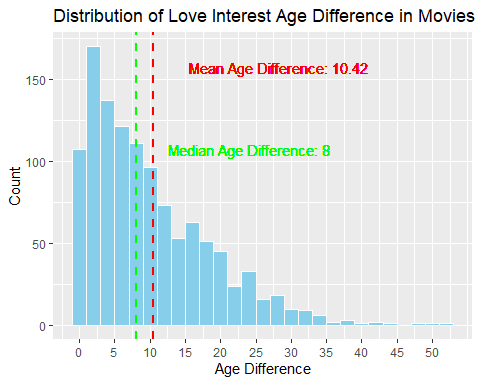
age\_gaps <- readr::read\_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-02-14/age\_gaps.csv', show\_col\_types = FALSE)  
  
# Adding a columns for half\_plus\_seven\_rule  
age\_gaps <- age\_gaps %>%  
 mutate(half\_plus\_seven = actor\_2\_age > ((actor\_1\_age / 2) + 7) & actor\_2\_age < ((actor\_1\_age - 7) \* 2))  
  
# How is `age\_difference` distributed? What's the 'typical' `age\_difference` in movies?  
# Calculate the summary statistics for age\_difference  
summary\_stats <- age\_gaps %>%  
 summarise(  
 min\_age\_diff = min(age\_difference),  
 max\_age\_diff = max(age\_difference),  
 median\_age\_diff = median(age\_difference),  
 mean\_age\_diff = mean(age\_difference),  
 sd\_age\_diff = sd(age\_difference)  
 )  
  
# Plot the distribution of age\_difference  
age\_diff\_plot <- ggplot(age\_gaps, aes(x = age\_difference)) +  
 geom\_histogram(binwidth = 2, fill = "skyblue", color = "white") +  
 geom\_vline(aes(xintercept = summary\_stats$mean\_age\_diff), color = "red", linetype = "dashed", size = 1) +  
 geom\_vline(aes(xintercept = summary\_stats$median\_age\_diff), color = "green", linetype = "dashed", size = 1) +  
 geom\_text(aes(x = summary\_stats$mean\_age\_diff, y = 150, label = paste("Mean Age Difference:", round(summary\_stats$mean\_age\_diff, 2))), color = "red", hjust = -0.2, vjust = -0.5) +  
 geom\_text(aes(x = summary\_stats$median\_age\_diff, y = 100, label = paste("Median Age Difference:", round(summary\_stats$median\_age\_diff, 2))), color = "green", hjust = -0.2, vjust = -0.5) +  
 scale\_x\_continuous(breaks = seq(min(age\_gaps$age\_difference), max(age\_gaps$age\_difference), by = 5)) +  
 labs(title = "Distribution of Love Interest Age Difference in Movies", x = "Age Difference", y = "Count")

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

# Print the summary statistics  
print(summary\_stats)

# A tibble: 1 × 5  
 min\_age\_diff max\_age\_diff median\_age\_diff mean\_age\_diff sd\_age\_diff  
 <dbl> <dbl> <dbl> <dbl> <dbl>  
1 0 52 8 10.4 8.51

print(age\_diff\_plot)



# How frequently does the half plus seven rule (see below) apply in this dataset?  
half\_plus\_seven\_freq <- age\_gaps %>%  
 group\_by(half\_plus\_seven) %>%  
 summarize(frequency = n()) %>%   
 mutate(percentage = frequency / nrow(age\_gaps) \* 100)  
  
print(half\_plus\_seven\_freq)

# A tibble: 2 × 3  
 half\_plus\_seven frequency percentage  
 <lgl> <int> <dbl>  
1 FALSE 360 31.2  
2 TRUE 795 68.8

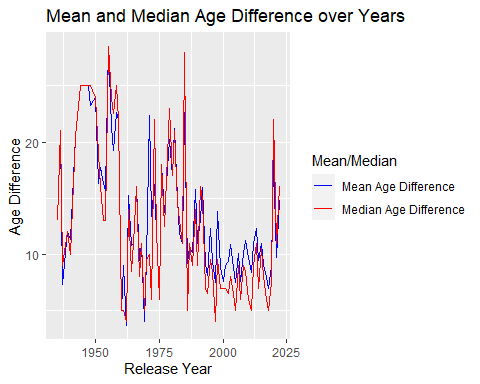
# 69% of movie romantic relationships are acceptable under the half\_plus\_seven rule  
  
# Which movie has the greatest number of love interests?  
movie\_count\_love\_interests <- age\_gaps %>%  
 group\_by(movie\_name) %>%  
 summarize(love\_interests = n\_distinct(couple\_number)) %>%  
 arrange(desc(love\_interests))  
  
print(paste(movie\_count\_love\_interests$movie\_name[1], "is the movie with the most love interests"))

[1] "Love Actually is the movie with the most love interests"

# Which actors/ actresses have the greatest number of love interests in this dataset?  
# Here I'm trying to find the actor/actress with the most unique love interestes across the dataset  
# This means, the actor/actress who has played a love interest across most other actors  
  
# Get actor pairs, disregard order actor1 and actor2  
actor\_pairs <- age\_gaps %>%  
 group\_by(group\_col1 = pmin(actor\_1\_name, actor\_2\_name), group\_col2 = pmax(actor\_1\_name, actor\_2\_name)) %>%  
 summarise(count = n()) %>%   
 arrange(desc(count))  
  
#Combine the actor1 and actor2 colmns into 1  
combined\_actor\_col <- c(actor\_pairs$group\_col1, actor\_pairs$group\_col2)  
combined\_actor\_occurances <- sort(table(combined\_actor\_col),decreasing = TRUE)  
print(combined\_actor\_occurances[1])

Keanu Reeves   
 25

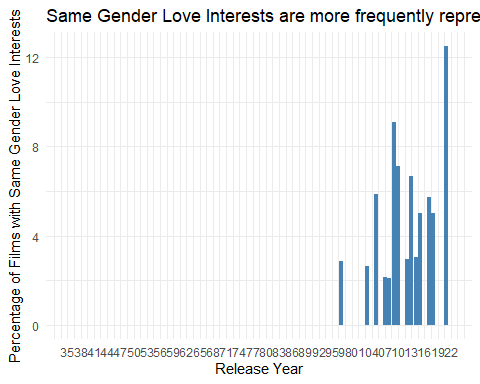
# Keanu Reeves has acted as love interest to 25 other actors/actresses  
# This is the highest across the dataset  
  
# Is the mean/median age difference staying constant over the years (1935 - 2022)?  
age\_gaps %>%  
 group\_by(release\_year) %>%  
 summarise(mean\_age\_diff = mean(age\_difference),  
 median\_age\_diff = median(age\_difference)) %>%  
 filter(release\_year >= 1935 & release\_year <= 2022) %>%  
 ggplot(aes(x = release\_year)) +  
 geom\_line(aes(y = mean\_age\_diff, color = "Mean Age Difference")) +  
 geom\_line(aes(y = median\_age\_diff, color = "Median Age Difference")) +  
 labs(title = "Mean and Median Age Difference over Years",  
 x = "Release Year",  
 y = "Age Difference",  
 color = "Mean/Median") +  
 scale\_color\_manual(values = c("Mean Age Difference" = "blue", "Median Age Difference" = "red"))



# Seems like age difference has been quite volatile over time  
# The late 50s and 60s saw age differences decline before slowly rising again throught 1990  
# Age differences declines again in the 2000s  
  
# How frequently does Hollywood depict same-gender love interests?  
age\_gaps <- age\_gaps %>%  
 # Add boolean column if same gender  
 mutate(same\_gender = character\_1\_gender == character\_2\_gender)  
   
same\_gender\_love\_interests <- age\_gaps %>%   
 group\_by(same\_gender) %>%  
 summarise(count = n(),  
 percentage = n() / nrow(age\_gaps) \* 100)  
print(same\_gender\_love\_interests)

# A tibble: 2 × 3  
 same\_gender count percentage  
 <lgl> <int> <dbl>  
1 FALSE 1132 98.0   
2 TRUE 23 1.99

# Same gender love interests occur 2% of the time  
  
percentage\_table <- age\_gaps %>%  
 group\_by(release\_year) %>%  
 summarize(percentage = mean(same\_gender == "TRUE") \* 100)  
  
scale\_x\_yr <- seq(min(percentage\_table$release\_year), max(percentage\_table$release\_year), by = 3)  
  
# Bar plot of percentage of films with same gender love interests  
same\_gender\_love\_interests\_plot <- ggplot(percentage\_table, aes(x = release\_year, y = percentage)) +  
 geom\_bar(stat = "identity", fill = "steelblue") +  
 labs(x = "Release Year", y = "Percentage of Films with Same Gender Love Interests") +  
 ggtitle("Same Gender Love Interests are more frequently represented in movies") +  
 scale\_x\_continuous(  
 breaks = scale\_x\_yr,  
 labels = substring(scale\_x\_yr, nchar(scale\_x\_yr) - 1)  
 ) +  
 theme\_minimal()  
  
plot(same\_gender\_love\_interests\_plot)



* 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:

# Details

* Who did you collaborate with: Brent Lewis
* Approximately how much time did you spend on this problem set: 5-7 hours. It took a long time.
* What, if anything, gave you the most trouble: All the data manipulation to get the right table to chart