Homework #7

Summarization by Group

## Concepts

Carefully explain the following: 1. The two elements that define the distribution of a random variable 1. How the mean measures the location of a distribution 1. How the standard deviation measures the scale or spread of a distribution

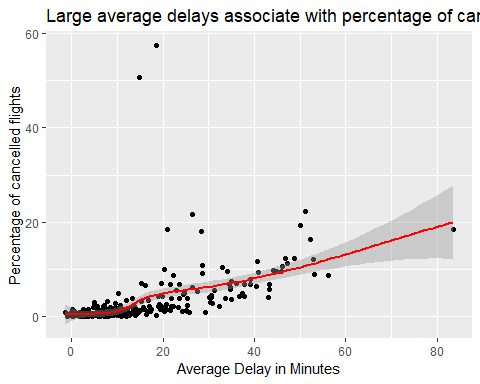
### Concept Answers

1. Possible values and probability of those values.
2. The mean is just the average of all the data and is where the distribution is centered around
3. The standard deviation value tells you how far away from the mean value that up to 68.2% of the data is for one standard deviation, and two standard deviations tells you how far away from the mean up to about 95% of the data is, etc.

## Interpretation of code

Interpret the following R code:

flights %>%  
 group\_by(day, month) %>%  
 summarize(cancelled = mean(is.na(dep\_time)) \* 100,  
 delay = mean(dep\_delay, na.rm = TRUE)) %>%  
 ggplot(mapping = aes(x = delay, y = cancelled)) +  
 geom\_point() +  
 geom\_smooth(color = "red") +  
 labs(  
 title = "Large average delays associate with percentage of cancelled flights",  
 x = "Average Delay in Minutes",  
 y = "Percentage of cancelled flights"  
 )



### Code Interpretation

1. Defines the dataset being used as “flights” then uses a pipe to tell what to do to that dataset
2. Groups the flights data by day and month, then uses a pipe to tell the code what to do with the grouped data next
3. Uses summarize() to combine all the dep\_time data into an a percentage of flights cancelled, where is.na(dep\_time) returns how many flights were never departed
4. Creates “delay” variable as the average of the departure delay times, then uses a pipe to continue on to plotting the information
5. Creates a ggplot and maps the average delay time against the average % cancelled
6. Creates a scatterplot of the data
7. Adds a smooth loess fit line colored red
8. Next three lines label the axis of the plots and gives it a title.

## Write code that accomplishes the following:

1. Create a new data frame that excludes cancelled flights
2. Rank carriers from best to worst by their average arrival delays
3. Rank destination airports from worst to best by their average arrival delays
4. Rank NYC airports from best to worst by their average departure delays
5. Rank carriers from best to worst by their on time percentage

Is it possible to use the above results to rank carriers? As a hint, think about how the following code might help you consider this question.

Of course you could rank carriers using the above results. If you look at the on-time percentages, you could use that to compute the average number of flights to specific origins from certain destinations that show up on-time by multiplying the time percentage by the count. Then you could just look at the total number of flights predicted to be on time based on that percentage and rank the carriers accordingly

#Create new data frame that excludes cancelled flights  
non\_cancelled\_flights <- filter(flights, !is.na(dep\_delay), !is.na(arr\_delay))   
  
#Rank carriers from best (smallest) to worst (highest) based on average arrival delays  
non\_cancelled\_flights %>%  
 group\_by(carrier) %>%  
 summarize(carrier\_ave\_arrival\_delay = mean(arr\_delay))%>%  
 arrange(carrier\_ave\_arrival\_delay) #Don't need to make it descending since it already arranges it from smallest to largest

## # A tibble: 16 x 2  
## carrier carrier\_ave\_arrival\_delay  
## <chr> <dbl>  
## 1 AS -9.93   
## 2 HA -6.92   
## 3 AA 0.364  
## 4 DL 1.64   
## 5 VX 1.76   
## 6 US 2.13   
## 7 UA 3.56   
## 8 9E 7.38   
## 9 B6 9.46   
## 10 WN 9.65   
## 11 MQ 10.8   
## 12 OO 11.9   
## 13 YV 15.6   
## 14 EV 15.8   
## 15 FL 20.1   
## 16 F9 21.9

#Rank destination airports from worst (highest) to best (smallest) by their average arrival delays  
non\_cancelled\_flights %>%  
 group\_by(dest) %>%  
 summarize(destination\_ave\_arrival\_delay = mean(arr\_delay)) %>%  
 arrange(desc(destination\_ave\_arrival\_delay)) #Need to make it descending so that it arranges it from largest to smallest

## # A tibble: 104 x 2  
## dest destination\_ave\_arrival\_delay  
## <chr> <dbl>  
## 1 CAE 41.8  
## 2 TUL 33.7  
## 3 OKC 30.6  
## 4 JAC 28.1  
## 5 TYS 24.1  
## 6 MSN 20.2  
## 7 RIC 20.1  
## 8 CAK 19.7  
## 9 DSM 19.0  
## 10 GRR 18.2  
## # ... with 94 more rows

#Rank NYC airports from best (smallest) to worst (highest) by their average departure delays  
non\_cancelled\_flights %>%  
 group\_by(origin) %>%  
 summarize(origin\_ave\_departure\_delay = mean(dep\_delay)) %>%  
 arrange(origin\_ave\_departure\_delay) #Don't need to make it descending since it already arranges it from smallest to largest

## # A tibble: 3 x 2  
## origin origin\_ave\_departure\_delay  
## <chr> <dbl>  
## 1 LGA 10.3  
## 2 JFK 12.0  
## 3 EWR 15.0

#Rank carriers from best (highest) to worst (smallest) by their on-time percentage  
non\_cancelled\_flights %>%  
 mutate(on\_time = arr\_delay <= 0) %>%  
 group\_by(carrier)%>%  
 summarize(percent\_ontime = mean(on\_time)\*100)%>%  
 arrange(desc(percent\_ontime)) #Need to make it descending so that it arranges it from largest to smallest

## # A tibble: 16 x 2  
## carrier percent\_ontime  
## <chr> <dbl>  
## 1 AS 73.3  
## 2 HA 71.6  
## 3 AA 66.5  
## 4 VX 65.9  
## 5 DL 65.6  
## 6 OO 65.5  
## 7 US 62.9  
## 8 9E 61.6  
## 9 UA 61.5  
## 10 B6 56.3  
## 11 WN 56.0  
## 12 MQ 53.3  
## 13 YV 52.6  
## 14 EV 52.1  
## 15 F9 42.4  
## 16 FL 40.3

flights %>%   
 group\_by(carrier, origin) %>%  
 count(dest)

## # A tibble: 439 x 4  
## # Groups: carrier, origin [35]  
## carrier origin dest n  
## <chr> <chr> <chr> <int>  
## 1 9E EWR ATL 4  
## 2 9E EWR CVG 855  
## 3 9E EWR DTW 231  
## 4 9E EWR MSP 178  
## 5 9E JFK ATL 55  
## 6 9E JFK AUS 2  
## 7 9E JFK BNA 364  
## 8 9E JFK BOS 914  
## 9 9E JFK BUF 779  
## 10 9E JFK BWI 856  
## # ... with 429 more rows