Homerwork 2

MIKE DIZON

5/21/23

# Mass shootings in the US

## Obtain the data

Rows: 125  
Columns: 14  
$ case <chr> "Oxford High School shooting", "San Jose VTA shoo…  
$ year <dbl> 2021, 2021, 2021, 2021, 2021, 2021, 2020, 2020, 2…  
$ month <chr> "Nov", "May", "Apr", "Mar", "Mar", "Mar", "Mar", …  
$ day <dbl> 30, 26, 15, 31, 22, 16, 16, 26, 10, 6, 31, 4, 3, …  
$ location <chr> "Oxford, Michigan", "San Jose, California", "Indi…  
$ summary <chr> "Ethan Crumbley, a 15-year-old student at Oxford …  
$ fatalities <dbl> 4, 9, 8, 4, 10, 8, 4, 5, 4, 3, 7, 9, 22, 3, 12, 5…  
$ injured <dbl> 7, 0, 7, 1, 0, 1, 0, 0, 3, 8, 25, 27, 26, 12, 4, …  
$ total\_victims <dbl> 11, 9, 15, 5, 10, 9, 4, 5, 7, 11, 32, 36, 48, 15,…  
$ location\_type <chr> "School", "Workplace", "Workplace", "Workplace", …  
$ male <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, T…  
$ age\_of\_shooter <dbl> 15, 57, 19, NA, 21, 21, 31, 51, NA, NA, 36, 24, 2…  
$ race <chr> NA, NA, "White", NA, NA, "White", NA, "Black", "B…  
$ prior\_mental\_illness <chr> NA, "Yes", "Yes", NA, "Yes", NA, NA, NA, NA, NA, …

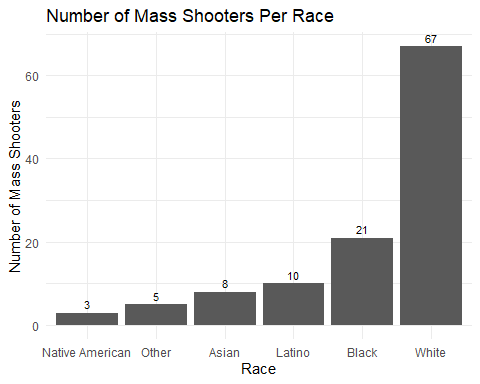
### Specific questions

* Generate a data frame that summarizes the number of mass shootings per year.

#Create a variable to store dataframe of number of mass shootings  
number\_mass\_shootings <- mass\_shootings %>%   
  
#Group by year  
 group\_by(year) %>%   
   
#Count the number of mass shootings per year  
 summarize(number\_mass\_shootings = n())

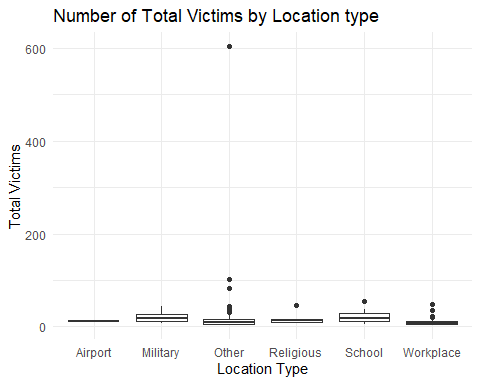
* Generate a bar chart that identifies the number of mass shooters associated with each race category. The bars should be sorted from highest to lowest and each bar should show its number.

#Create a variable to store the dataframe of number of mass shooters per race  
mass\_shooters\_per\_race <- mass\_shootings %>%  
   
#Group by race  
 group\_by(race) %>%  
   
#Count the number of mass shooters per race  
 summarize(mass\_shooters\_per\_race = n())  
  
#Drop the NA values in dataframe  
mass\_shooters\_per\_race %>%   
 drop\_na(race) %>%   
  
#Reorder the race based on the number of mass shooters per race  
 mutate(race = fct\_reorder(race,mass\_shooters\_per\_race)) %>%   
   
#Create a plot of race vs number of mass shooters  
 ggplot() +  
  
#Set the race as x-axis and number of mass shooters y-axis  
 aes(x = race, y = mass\_shooters\_per\_race) +  
   
#Create a bar graph  
 geom\_col() +  
   
#Create data labels for each bar  
 geom\_text(aes(label = mass\_shooters\_per\_race), vjust = -0.5, size = 3) +  
   
#Create labels for the graph  
 labs(title = "Number of Mass Shooters Per Race", x = "Race", y = "Number of Mass Shooters") +  
   
#Set theme  
 theme\_minimal()



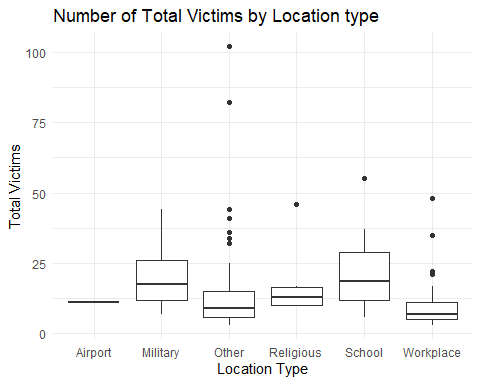
* Generate a boxplot visualizing the number of total victims, by type of location.

#Create a plot of total victims by location type  
mass\_shootings %>%   
 ggplot() +  
   
#Set location type as x-axis and total victims as y-axis  
 aes(x = location\_type, y = total\_victims) +  
   
#Generate a boxplot  
 geom\_boxplot() +  
  
#Create labels for the graph  
 labs(title = "Number of Total Victims by Location type", x = "Location Type", y = "Total Victims") +  
   
#Set theme  
 theme\_minimal()



* Redraw the same plot, but remove the Las Vegas Strip massacre from the dataset.

#Filter data to exclude the Las Vegas Strip massacre  
mass\_shootings %>%   
 filter(location != "Las Vegas, Nevada") %>%   
   
#Create the same boxplot as above  
 ggplot() +  
   
#Set location type as x-axis and total victims as y-axis  
 aes(x = location\_type, y = total\_victims) +  
   
#Generate a boxplot  
 geom\_boxplot() +  
   
#Create labels for the graph  
 labs(title = "Number of Total Victims by Location type", x = "Location Type", y = "Total Victims") +  
   
#Set theme  
 theme\_minimal()



### More open-ended questions

Address the following questions. Generate appropriate figures/tables to support your conclusions.

* How many white males with prior signs of mental illness initiated a mass shooting after 2000?

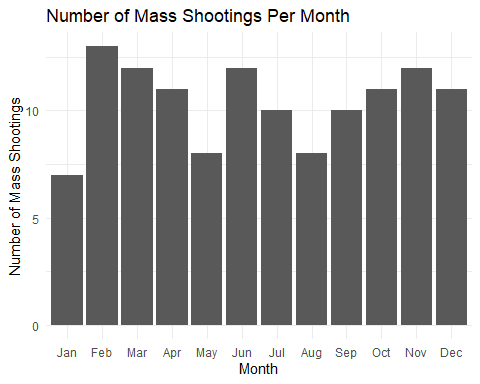
#Filter the data based on set conditions to determine the number of white males with prior signs of mental illness after 2000  
white\_males\_mental\_illness\_after\_2000 <- mass\_shootings %>%   
 filter(race == "White" & male == "TRUE" & prior\_mental\_illness == "Yes" & year>2000)  
  
#Count the number  
count(white\_males\_mental\_illness\_after\_2000)

# A tibble: 1 × 1  
 n  
 <int>  
1 22

**Answer:** There were 22 white males with prior signs of mental illness who initiated a mass shooting after 2000.

* Which month of the year has the most mass shootings? Generate a bar chart sorted in chronological (natural) order (Jan-Feb-Mar- etc) to provide evidence of your answer.

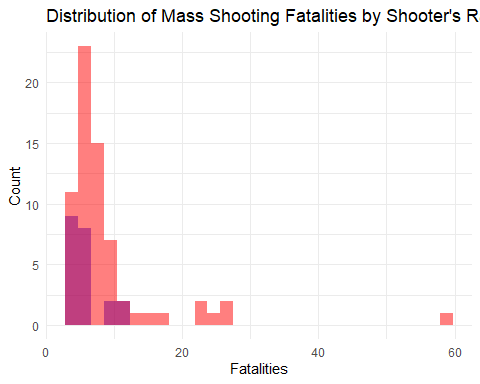
#Create a variable to store the data frame with the month of the year with the most mass shootings  
month\_most\_mass\_shootings <- mass\_shootings %>%  
   
#Count the number of mass shootings per year  
 count(month) %>%   
   
#Arrange in descending order  
 arrange(desc(n))  
  
#Create a variable that would indicate the correct order of months in a calendar year  
month\_order <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")  
  
#Create a plot of the number of mass shootings per month  
month\_most\_mass\_shootings %>%   
  
#Mutate so that the months will be in the desired order  
 mutate(month = factor(month, levels = month\_order, ordered = TRUE)) %>%   
   
#Use ggplot to create the diagram  
 ggplot() +  
   
#Set month as the x-axis and the number of mass shootings as the y-axis  
 aes(x = month, y = n) +  
   
#Create a bar graph  
 geom\_col() +  
   
#Include labels for the graph  
 labs(title = "Number of Mass Shootings Per Month", x = "Month", y = "Number of Mass Shootings") +  
  
#Set theme  
 theme\_minimal()



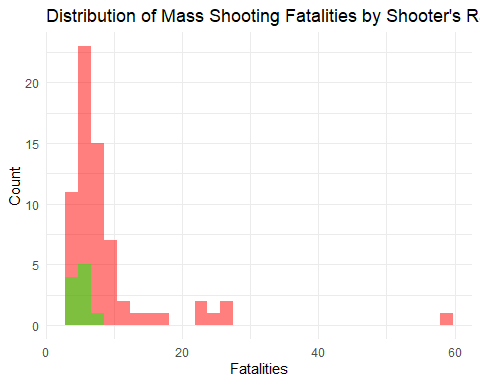
**Answer:** As observed from the bar graph, February was the month with the most mass shootings at 13.

* How does the distribution of mass shooting fatalities differ between White and Black shooters? What about White and Latino shooters?

#Filter the data for mass shooting fatalities with Black shooters  
black\_shooters <- mass\_shootings %>%   
 filter(race == "Black")  
  
#Filter the data for mass shooting fatalities with White shooters  
white\_shooters <- mass\_shootings %>%   
 filter(race == "White")  
  
#Create a histogram to check the shape of the distribution of fatalities for Black and White shooters  
ggplot() +  
   
#Plot the first histogram for the mass shooting fatalities with Black shooters  
 geom\_histogram(  
 data = black\_shooters,  
  
#Set the number of fatalities as the x-axis  
 aes(x = fatalities),   
  
#Fill the color of the histogram bars with blue  
 fill = "blue",  
  
#Set transparency of histogram bars  
 alpha = 0.5,   
  
#Set number of intervals into which data is divided  
 bins = 30) +  
   
#Plot the second histogram for the mass shooting fatalities with White shooters  
 geom\_histogram(  
 data = white\_shooters,   
   
#Set the number of fatalities as the x-axis  
 aes(x=fatalities),  
  
#Fill the color of the histogram bars with red  
 fill = "red",   
  
#Set transparency of histogram bars  
 alpha = 0.5,   
  
#Set number of intervals into which data is divided  
 bins = 30) +  
   
#Create labels for the histogram  
 labs(title = "Distribution of Mass Shooting Fatalities by Shooter's Race", x = "Fatalities", y = "Count", fill = "Shooter's Race") +  
   
#Manually specify the fill color for Black and White shooters  
 scale\_fill\_manual(values = c("blue", "red"), labels = c("Black Shooter", "White Shooter")) +  
   
#Set theme  
 theme\_minimal()



#Filter the data for mass shooting fatalities with Latino shooters  
latino\_shooters <- mass\_shootings %>%   
 filter(race == "Latino")  
  
#Create a histogram to check the shape of the distribution of fatalities for Latino and White shooters  
ggplot() +  
   
#Plot the first histogram for the mass shooting fatalities with White shooters  
 geom\_histogram(  
 data = white\_shooters,  
  
#Set the number of fatalities as the x-axis  
 aes(x = fatalities),   
  
#Fill the color of the histogram bars with red  
 fill = "red",  
  
#Set transparency of histogram bars  
 alpha = 0.5,   
  
#Set number of intervals into which data is divided  
 bins = 30) +  
   
#Plot the second histogram for the mass shooting fatalities with Latino shooters  
 geom\_histogram(  
 data = latino\_shooters,   
   
#Set the number of fatalities as the x-axis  
 aes(x=fatalities),  
  
#Fill the color of the histogram bars with green  
 fill = "green",   
  
#Set transparency of histogram bars  
 alpha = 0.5,   
  
#Set number of intervals into which data is divided  
 bins = 30) +  
   
#Create labels for the histogram  
 labs(title = "Distribution of Mass Shooting Fatalities by Shooter's Race", x = "Fatalities", y = "Count", fill = "Shooter's Race") +  
   
#Manually specify the fill color for Black and White shooters  
 scale\_fill\_manual(values = c("red", "green"), labels = c("White Shooter", "Latino Shooter")) +  
   
#Set theme  
 theme\_minimal()



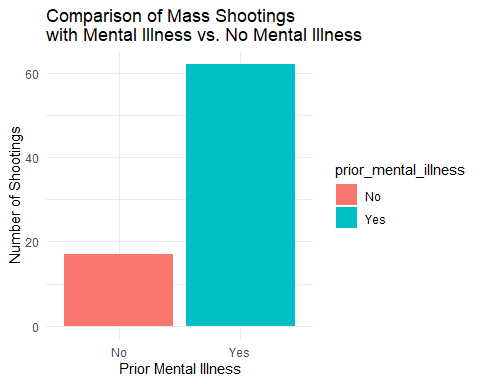
**Answer:** The shape of the distribution for the number of fatalities between Black and White shooters are similar in a sense that both are skewed to the right. However, they differ because there are more instances and extreme values for the number of fatalities with White shooters as opposed to Black shooters.

Similarly, the shape of the distribution for the number of fatalities between White and Latino shooters are almost the same with the previous comparison. That is, both are skewed to the right but there are more instances and extreme values for the number of fatalities with White shooters as opposed to Latino shooters.

### Very open-ended

* Are mass shootings with shooters suffering from mental illness different from mass shootings with no signs of mental illness in the shooter?

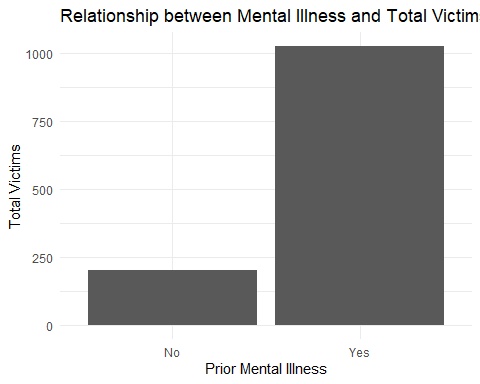
#Filter the data for only those shooters suffering from prior mental illness  
with\_mental\_illness <- mass\_shootings %>%   
 filter(prior\_mental\_illness == "Yes")  
  
#Filter the data for only those shooters not suffering from prior mental illness   
no\_mental\_illness <- mass\_shootings %>%   
 filter(prior\_mental\_illness == "No")  
  
#Combine the two datasets  
combined\_data <- rbind(with\_mental\_illness, no\_mental\_illness)  
  
#Create a grouped bar chart using ggplot  
combined\_data %>%   
ggplot() +  
  
#Set x-axis as the presence of mental illness and set the same as the fill  
 aes(x = prior\_mental\_illness, fill = prior\_mental\_illness) +  
  
#Create a bar graph  
 geom\_bar() +  
   
#Create labels for the graph  
 labs(x = "Prior Mental Illness", y = "Number of Shootings", title = "Comparison of Mass Shootings\nwith Mental Illness vs. No Mental Illness") +  
   
#Set theme  
 theme\_minimal()



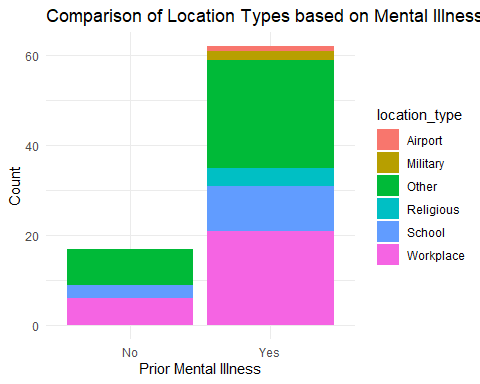
**Answer:** The graph above shows that there are more mass shooting fatalities if mass shooters had evidence of prior mental illness. We can therefore conclude that it is more likely for mass shooters in the sample to engage in mass shootings if they are mentally sick.

* Assess the relationship between mental illness and total victims, mental illness and location type, and the intersection of all three variables.

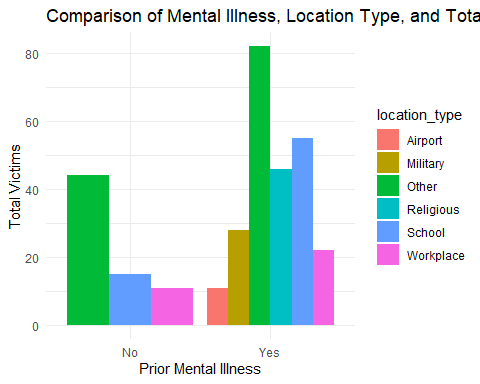
#Filter out the NA values for prior mental illness from the data frame  
mass\_shootings %>%  
 filter(!is.na(prior\_mental\_illness)) %>%  
   
#Use ggplot to assess relationship between mental illness and total victims  
ggplot() +  
   
#Set prior mental illness as x-axis and the number of total victims as y-axis  
 aes(x = prior\_mental\_illness, y = total\_victims) +  
   
#Create a scatter plot  
 geom\_col() +  
   
#Create labels for the graph  
 labs(x = "Prior Mental Illness", y = "Total Victims") +  
 ggtitle("Relationship between Mental Illness and Total Victims") +  
   
#Set theme  
 theme\_minimal()



#Filter out the NA values for prior mental illness from the data frame  
mass\_shootings %>%  
 filter(!is.na(prior\_mental\_illness)) %>%  
  
#Use ggplot to assess relationship between mental illness and location type  
ggplot() +  
   
#Set prior mental illness as x-axis and location type as y-axis, and use location type as fill  
 aes(x = prior\_mental\_illness, fill = location\_type) +  
   
#Create bar graph  
 geom\_bar() +  
   
#Create labels for the graph  
 labs(x = "Prior Mental Illness", y = "Count") +  
 ggtitle("Comparison of Location Types based on Mental Illness") +  
   
#Set theme  
 theme\_minimal()



#Filter out the NA values for prior mental illness from the data frame  
mass\_shootings %>%   
 filter(!is.na(prior\_mental\_illness)) %>%   
   
#Use ggplot to assess relationship across the three variables  
 ggplot() +  
   
#Set prior mental illness as x-axis, total victims as y-axis, and location type as fill  
 aes(x = prior\_mental\_illness, y = total\_victims, fill = location\_type) +  
   
#Create a bar graph and set the height and positioning of the bars  
 geom\_bar(stat = "identity", position = "dodge") +  
   
#Create labels for the graph  
 labs(x = "Prior Mental Illness", y = "Total Victims") +  
 ggtitle("Comparison of Mental Illness, Location Type, and Total Victims") +  
   
#Set theme  
 theme\_minimal()



**Answer:** As seen from the first graph, the total number of victims is significantly higher for shooters with prior mental illness compared with those who do not. Further expanding this analysis through the second graph shows that mass shooters with no prior mental illness conduct most mass shootings in other location types or in the workplace, which is also similar to those with prior mental illness.The third graph shows that the number of total victims are highest in both other location types for mass shooters with no prior mental illness and for those who have.

# Exploring credit card fraud

Rows: 671,028  
Columns: 14  
$ trans\_date\_trans\_time <dttm> 2019-02-22 07:32:58, 2019-02-16 15:07:20, 2019-…  
$ trans\_year <dbl> 2019, 2019, 2019, 2019, 2019, 2019, 2019, 2020, …  
$ category <chr> "entertainment", "kids\_pets", "personal\_care", "…  
$ amt <dbl> 7.79, 3.89, 8.43, 40.00, 54.04, 95.61, 64.95, 3.…  
$ city <chr> "Veedersburg", "Holloway", "Arnold", "Apison", "…  
$ state <chr> "IN", "OH", "MO", "TN", "CO", "GA", "MN", "AL", …  
$ lat <dbl> 40.1186, 40.0113, 38.4305, 35.0149, 39.4584, 32.…  
$ long <dbl> -87.2602, -80.9701, -90.3870, -85.0164, -106.385…  
$ city\_pop <dbl> 4049, 128, 35439, 3730, 277, 1841, 136, 190178, …  
$ job <chr> "Development worker, community", "Child psychoth…  
$ dob <date> 1959-10-19, 1946-04-03, 1985-03-31, 1991-01-28,…  
$ merch\_lat <dbl> 39.41679, 39.74585, 37.73078, 34.53277, 39.95244…  
$ merch\_long <dbl> -87.52619, -81.52477, -91.36875, -84.10676, -106…  
$ is\_fraud <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …

* In this dataset, how likely are fraudulent transactions? Generate a table that summarizes the number and frequency of fraudulent transactions per year.

#Store the number and frequency of fraudulent transactions in a variable  
fraud\_summary <- card\_fraud %>%  
   
#Group by transaction year  
 group\_by(trans\_year) %>%  
   
#Compute for the sum of fraudulent transactions and get the proportion out of all transactions per year  
 summarize(num\_fraud\_transactions = sum(is\_fraud),  
 frequency = num\_fraud\_transactions / n() \* 100)

**Answer:** There were 2721 fraudulent transactions in 2019, which is 0.57% of all number of transactions for that year and there were 1215 fraudulent transactions in 2020, which is 0.63% of all number of transactions for that year.

* How much money (in US$ terms) are fraudulent transactions costing the company? Generate a table that summarizes the total amount of legitimate and fraudulent transactions per year and calculate the % of fraudulent transactions, in US$ terms.

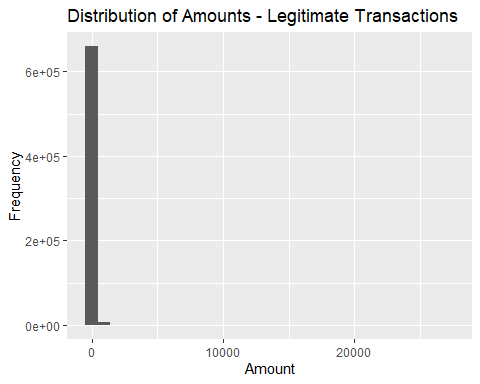
#Store the total amount of legitimate and fraudulent transactions per year into a variable  
transaction\_summary <- card\_fraud %>%  
   
#Group by transaction year  
 group\_by(trans\_year) %>%  
   
#Compute for legitimate and fraudulent amounts  
 summarize(total\_legitimate\_amount = sum(amt \* (1 - is\_fraud)),  
 total\_fraudulent\_amount = sum(amt \* is\_fraud)) %>%   
   
#Compute for the percentage of fraudulent transactions  
 mutate(fraud\_percentage = total\_fraudulent\_amount / (total\_legitimate\_amount + total\_fraudulent\_amount) \* 100)

**Answer:** Fraudulent transactions are costing the company 32182901 for 2019 and 12925914 for 2020 in US$ terms. In terms of percentage, fraudulent transactions make up 4.23% of all transactions in 2019 and 4.80% in 2020.

* Generate a histogram that shows the distribution of amounts charged to credit card, both for legitimate and fraudulent accounts. Also, for both types of transactions, calculate some quick summary statistics.

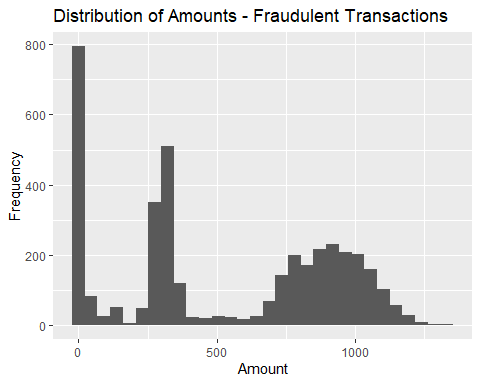
#Filter the data for legitimate transactions  
legitimate\_transactions <- card\_fraud %>%  
 filter(is\_fraud == 0)  
  
#Filter the data for fraudulent transactions  
fraudulent\_transactions <- card\_fraud %>%  
 filter(is\_fraud == 1)  
  
#Create a histogram for legitimate transactions using ggplot  
legitimate\_transactions %>%   
ggplot() +  
   
#Set the transaction amount as the x-axis  
 aes(x = amt) +  
   
#Create a histogram  
 geom\_histogram() +  
   
#Create labels for the graph  
 labs(x = "Amount", y = "Frequency", title = "Distribution of Amounts - Legitimate Transactions")

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



#Calculate summary statistics (mean, median, minimum, maximum) for legitimate transactions  
legitimate\_summary <- legitimate\_transactions %>%  
 summarize(  
 mean\_amount = mean(amt),  
 median\_amount = median(amt),  
 min\_amount = min(amt),  
 max\_amount = max(amt)  
 )  
  
#Create a histogram for fraudulent transactions using ggplot  
fraudulent\_transactions %>%   
ggplot() +  
   
#Set the transaction amount as the x-axis  
 aes(x = amt) +  
   
#Create a histogram  
 geom\_histogram() +  
   
#Create labels for the graph  
 labs(x = "Amount", y = "Frequency", title = "Distribution of Amounts - Fraudulent Transactions")

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

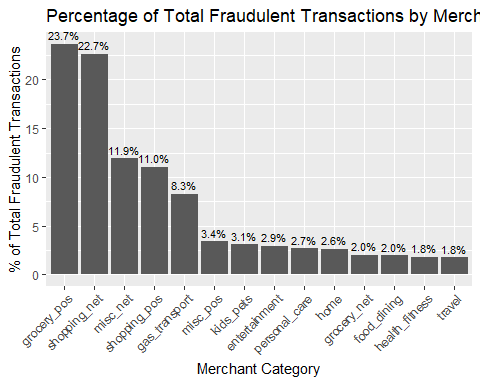


# Calculate summary statistics (mean, median, minimum, maximum) for fraudulent transactions  
fraudulent\_summary <- fraudulent\_transactions %>%  
 summarize(  
 mean\_amount = mean(amt),  
 median\_amount = median(amt),  
 min\_amount = min(amt),  
 max\_amount = max(amt)  
 )

**Answer:** The distribution of amounts charged to credit for both legitimate and fraudulent transactions are skewed to the right given that the mean for both types of transactions are greater than the median. It can be observed, however, that the distribution for the amount of legitimate transactions looks like a single stacked column given the dispersion in the amounts, which ranges from 1 to 27120. This is not the case for the distribution of the amount of fraudulent transactions as the dispersion in the amounts is lower and not too far from the mean.

* What types of purchases are most likely to be instances of fraud? Consider category of merchants and produce a bar chart that shows % of total fraudulent transactions sorted in order.

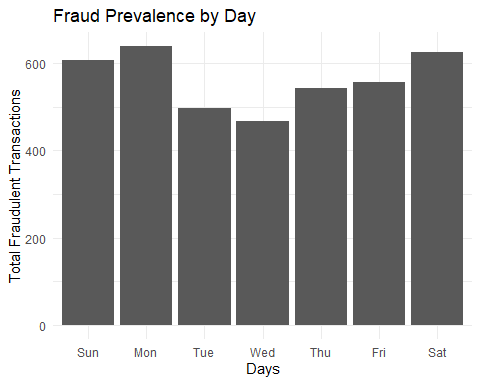
#Store into a variable the percentage of total fraudulent transactions per merchant category  
fraud\_category <- card\_fraud %>%  
   
#Group by merchant category  
 group\_by(category) %>%  
   
#Compute for the total number of fraudulent transactions  
 summarize(  
 total\_fraud\_transactions = sum(is\_fraud)) %>%  
   
#Compute for the percentage of fraudulent transactions  
 mutate(percentage = (total\_fraud\_transactions / sum(total\_fraud\_transactions)) \* 100) %>%  
   
#Arrange in descending order  
 arrange(desc(percentage))  
  
#Create bar chart using ggplot  
fraud\_category %>%   
ggplot() +  
   
#Set merchant category as the x-axis and reorder according to the largest percentage of fraudulent transactions  
 aes(  
 x = reorder(category, -percentage),   
  
#Set percentage of fraudulent transactions as the y-axis  
 y = percentage) +  
   
#Create a bar graph  
 geom\_bar(stat = "identity") +  
   
#Create labels for the graph  
 labs(x = "Merchant Category", y = "% of Total Fraudulent Transactions",   
 title = "Percentage of Total Fraudulent Transactions by Merchant Category") +  
   
#Create data labels with 1 decimal point  
 geom\_text(  
 aes(label = sprintf("%.1f%%", percentage)),   
   
#Adjust position and size of data label  
 vjust = -0.5, size = 3) +  
   
#Set theme  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))



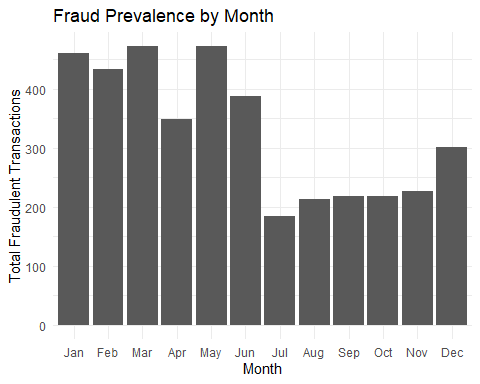
**Answer:** Fraudulent transactions are most prevalent with groceries with a percentage share of 23.7% of all categories followed by shopping with 22.7%. On the other hand, fraudulent transactions were least prevalent in health fitness and travel at 1.8% of all categories.

* When is fraud more prevalent? Which days, months, hours?

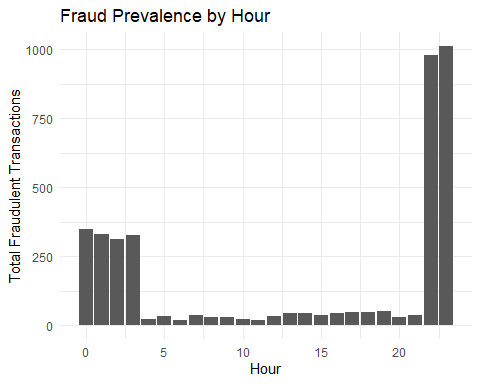
#Create new variables using the lubridate package  
card\_fraud <- card\_fraud %>%  
 mutate(  
   
#Extract the date component  
 date\_only = date(trans\_date\_trans\_time),  
   
#Extract the month component  
 month\_name = month(trans\_date\_trans\_time, label = TRUE),  
  
#Extract the hour component  
 hour = hour(trans\_date\_trans\_time),  
  
#Extract the weekday component  
 weekday = wday(trans\_date\_trans\_time, label = TRUE)  
 )  
  
#Store the analysis of fraud prevalence by day into a variable  
fraud\_by\_day <- card\_fraud %>%  
   
#Group by date  
 group\_by(weekday) %>%  
   
#Compute for the number of fraudulent transactions by date  
 summarize(total\_fraud\_transactions = sum(is\_fraud))  
  
#Store the analysis of fraud prevalence by month into a variable  
fraud\_by\_month <- card\_fraud %>%  
   
#Group by month  
 group\_by(month\_name) %>%  
   
#Compute for the number of fraudulent transactions by month  
 summarize(total\_fraud\_transactions = sum(is\_fraud))  
  
#Store the analysis of fraud prevalence by hour into a variable  
fraud\_by\_hour <- card\_fraud %>%  
   
#Group by hour  
 group\_by(hour) %>%  
   
#Compute for the number of fraudulent transactions by hour  
 summarize(total\_fraud\_transactions = sum(is\_fraud))  
  
#Create plot to visualize fraud prevalence by day  
fraud\_by\_day %>%   
ggplot() +  
   
#Set date as the x-axis and fraud total number of fraudulent transactions as the y-axis  
 aes(x = weekday, y = total\_fraud\_transactions) +  
   
#Create a bar graph  
 geom\_bar(stat = "identity") +  
   
#Create labels for the graph  
 labs(x = "Days", y = "Total Fraudulent Transactions", title = "Fraud Prevalence by Day") +  
   
#Set theme  
 theme\_minimal()



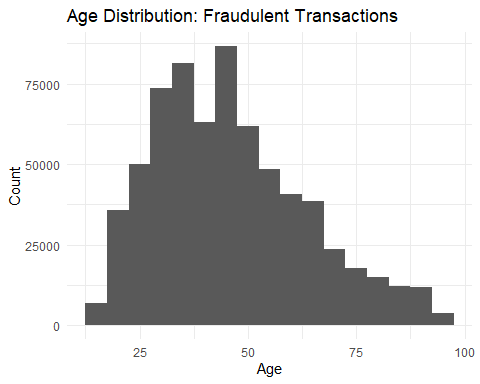
#Create plot to visualize fraud prevalence by month  
fraud\_by\_month %>%   
ggplot() +  
   
#Set month as the x-axis and fraud total number of fraudulent transactions as the y-axis  
 aes(x = month\_name, y = total\_fraud\_transactions) +  
   
#Create bar graph  
 geom\_bar(stat = "identity") +  
   
#Create labels for the graph  
 labs(x = "Month", y = "Total Fraudulent Transactions", title = "Fraud Prevalence by Month") +  
   
#Set theme  
 theme\_minimal()



#Create plot to visualize fraud prevalence by hour  
fraud\_by\_hour %>%   
ggplot() +  
   
#Set hour as the x-axis and fraud total number of fraudulent transactions as the y-axis  
 aes(x = hour, y = total\_fraud\_transactions) +  
   
#Create bar graph  
 geom\_bar(stat = "identity") +  
   
#Create labels for the graph  
 labs(x = "Hour", y = "Total Fraudulent Transactions", title = "Fraud Prevalence by Hour") +  
   
#Set theme  
 theme\_minimal()



#Calculate customer age at the time of the transaction  
card\_fraud <- card\_fraud %>%  
 mutate(  
 age = interval(dob, trans\_date\_trans\_time) / years(1)  
 )  
  
#Compare the age distribution between fraudulent and non-fraudulent transactions using ggplot  
card\_fraud %>%   
ggplot() +  
   
#Set age as the x-axis  
 aes(x = age) +  
   
#Create a histogram and set interval and position  
 geom\_histogram(binwidth = 5, position = "identity") +  
   
#Create labels for the graph  
 labs(x = "Age", y = "Count", title = "Age Distribution: Fraudulent Transactions") +  
   
#Set theme  
 theme\_minimal()



**Answer:** In terms of days, fraud is most prevalent during Monday with 639 of fraudulent transactions. In terms of month, fraud is most prevalent during March and May with 472 of fraudulent transactions. In terms of hour, fraud is most prevalent at 11 PM with 1012 of fraudulent transactions.

As seen from the shape of the distribution of ages who are exposed to fraudulent transactions, we can observe that it is skewed to the right, which implies that there is a higher frequency of younger ages in the dataset and suggests that younger people are more vicitimized by fraud especially those in the range of 30-50.

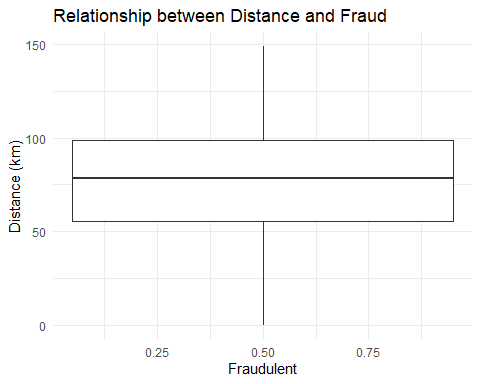
* Is fraud related to distance?

# distance between card holder's home and transaction  
# code adapted from https://www.geeksforgeeks.org/program-distance-two-points-earth/amp/  
  
  
card\_fraud <- card\_fraud %>%  
 mutate(  
   
 # convert latitude/longitude to radians  
 lat1\_radians = lat / 57.29577951,  
 lat2\_radians = merch\_lat / 57.29577951,  
 long1\_radians = long / 57.29577951,  
 long2\_radians = merch\_long / 57.29577951,  
   
 # calculate distance in miles  
 distance\_miles = 3963.0 \* acos((sin(lat1\_radians) \* sin(lat2\_radians)) + cos(lat1\_radians) \* cos(lat2\_radians) \* cos(long2\_radians - long1\_radians)),  
  
 # calculate distance in km  
 distance\_km = 6377.830272 \* acos((sin(lat1\_radians) \* sin(lat2\_radians)) + cos(lat1\_radians) \* cos(lat2\_radians) \* cos(long2\_radians - long1\_radians))  
  
 )

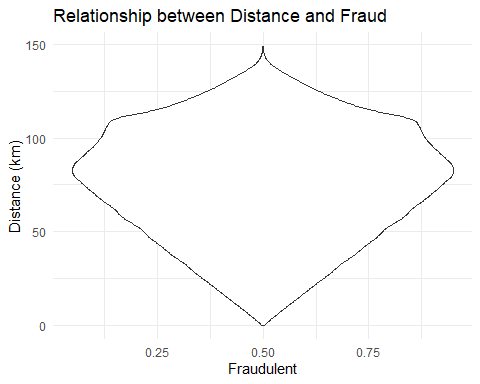
Plot a boxplot or a violin plot that looks at the relationship of distance and is\_fraud. Does distance seem to be a useful feature in explaining fraud?

#Create a boxplot for fraudulent transactions vs distance using ggplot  
card\_fraud %>%   
ggplot() +  
   
#Set fraud indicator as the x-axis and distance as the y-axis  
 aes(x = is\_fraud, y = distance\_km)+  
   
#Create a boxplot  
 geom\_boxplot() +  
   
#Create labels for the graph  
 labs(x = "Fraudulent", y = "Distance (km)") +  
 ggtitle("Relationship between Distance and Fraud") +  
   
#Set theme  
 theme\_minimal()

Warning: Continuous x aesthetic  
ℹ did you forget `aes(group = ...)`?



#Create a violin plot for fraudulent transactions vs distance using ggplot  
card\_fraud %>%   
ggplot() +  
   
#Set fraud indicator as the x-axis and distance as the y-axis  
 aes(x = is\_fraud, y = distance\_km) +  
   
#Create violin plot  
 geom\_violin() +  
   
#Create labels for the graph  
 labs(x = "Fraudulent", y = "Distance (km)") +  
 ggtitle("Relationship between Distance and Fraud") +  
   
#Set theme  
 theme\_minimal()



**Answer:** From the boxplot, we can observe that fraudulent transactions tend to happen over a wide range of distance although there is a concentration of data points within a relatively narrow distance range. This is also a similar observation with the violin plot given the width of the plot, which suggests a higher concentration of data points in narrow distances.

# Exploring sources of electricity production, CO2 emissions, and GDP per capita.

There are many sources of data on how countries generate their electricity and their CO2 emissions. I would like you to create three graphs:

## 1. A stacked area chart that shows how your own country generated its electricity since 2000.

You will use

geom\_area(colour="grey90", alpha = 0.5, position = "fill")

## 2. A scatter plot that looks at how CO2 per capita and GDP per capita are related

## 3. A scatter plot that looks at how electricity usage (kWh) per capita/day GDP per capita are related

We will get energy data from the Our World in Data website, and CO2 and GDP per capita emissions from the World Bank, using the wbstatspackage.

# Download electricity data  
url <- "https://nyc3.digitaloceanspaces.com/owid-public/data/energy/owid-energy-data.csv"  
  
energy <- read\_csv(url) %>%   
 filter(year >= 1990) %>%   
 drop\_na(iso\_code) %>%   
 select(1:3,  
 biofuel = biofuel\_electricity,  
 coal = coal\_electricity,  
 gas = gas\_electricity,  
 hydro = hydro\_electricity,  
 nuclear = nuclear\_electricity,  
 oil = oil\_electricity,  
 other\_renewable = other\_renewable\_exc\_biofuel\_electricity,  
 solar = solar\_electricity,  
 wind = wind\_electricity,   
 electricity\_demand,  
 electricity\_generation,  
 net\_elec\_imports, # Net electricity imports, measured in terawatt-hours  
 energy\_per\_capita, # Primary energy consumption per capita, measured in kilowatt-hours Calculated by Our World in Data based on BP Statistical Review of World Energy and EIA International Energy Data  
 energy\_per\_gdp, # Energy consumption per unit of GDP. This is measured in kilowatt-hours per 2011 international-$.  
 per\_capita\_electricity, # Electricity generation per capita, measured in kilowatt-hours  
 )   
  
# Download data for C02 emissions per capita https://data.worldbank.org/indicator/EN.ATM.CO2E.PC  
co2\_percap <- wb\_data(country = "countries\_only",   
 indicator = "EN.ATM.CO2E.PC",   
 start\_date = 1990,   
 end\_date = 2022,  
 return\_wide=FALSE) %>%   
 filter(!is.na(value)) %>%   
 #drop unwanted variables  
 select(-c(unit, obs\_status, footnote, last\_updated)) %>%   
 rename(year = date,  
 co2percap = value)  
  
  
# Download data for GDP per capita https://data.worldbank.org/indicator/NY.GDP.PCAP.PP.KD  
gdp\_percap <- wb\_data(country = "countries\_only",   
 indicator = "NY.GDP.PCAP.PP.KD",   
 start\_date = 1990,   
 end\_date = 2022,  
 return\_wide=FALSE) %>%   
 filter(!is.na(value)) %>%   
 #drop unwanted variables  
 select(-c(unit, obs\_status, footnote, last\_updated)) %>%   
 rename(year = date,  
 GDPpercap = value)

Specific questions:

1. How would you turn energy to long, tidy format?

#Use pivot\_longer to to turn energy into a long, tidy format  
energy\_tidy <- energy %>%  
 pivot\_longer(  
 cols = starts\_with(c("biofuel", "coal", "gas", "hydro", "nuclear", "oil", "other\_renewable", "solar", "wind")),  
 names\_to = "energy\_source",  
 values\_to = "electricity\_generated")

1. You may need to join these data frames

#Join the data frames using 3-digit ISO code as the key  
co2\_gdp <- left\_join(co2\_percap, gdp\_percap, by = c("iso3c", "year"))

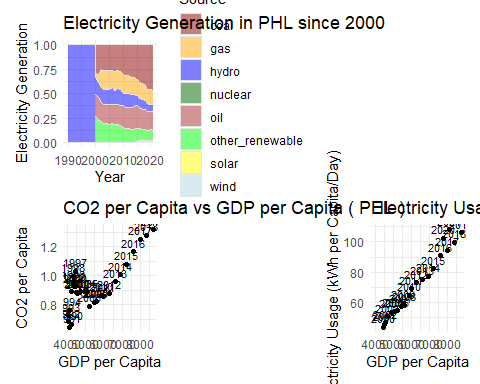
1. Write a function that takes as input any country’s name and returns all three graphs. You can use the patchwork package to arrange the three graphs as shown below

#Write a function that takes into country name as the input  
generate\_country\_graphs <- function(country\_name) {  
  
#GRAPH 1: STACKED AREA CHART  
   
# Filter the energy dataset for the specified country and store it into a variable  
 country\_energy <- energy %>%  
 filter(iso\_code == country\_name) %>%  
   
#Select desired columns  
 select(year, coal, gas, hydro, nuclear, oil, other\_renewable, solar, wind)  
  
#Reshape the data from wide to long format  
 country\_energy\_long <- country\_energy %>%  
 pivot\_longer(  
 cols = -year,   
 names\_to = "source",   
 values\_to = "electricity\_generation")  
  
#Create the stacked area chart for electricity generation  
 electricity\_generation\_plot <- country\_energy\_long %>%  
 ggplot() +  
   
#Set year as the x-axis, electricity generation as the y-axis, and energy source as the color fill  
 aes(x = year, y = electricity\_generation, fill = source) +  
   
#Create stacked area chart and specify color, transaprency, and positioning  
 geom\_area(  
 colour = "grey90",   
 alpha = 0.5,   
 position = "fill") +  
   
#Create labels for the graph  
 labs(x = "Year", y = "Electricity Generation", fill = "Source") +  
 ggtitle(paste("Electricity Generation in", country\_name, "since 2000")) +  
   
#Manually specify color for labels  
 scale\_fill\_manual(  
 values = c("coal" = "darkred",   
 "gas" = "orange",   
 "hydro" = "blue",  
 "nuclear" = "darkgreen",   
 "oil" = "brown",   
 "other\_renewable" = "green",  
 "solar" = "yellow", "wind" = "lightblue")) +  
   
#Set theme  
 theme\_minimal()  
  
#GRAPH 2: SCATTER PLOT (CO2 PER CAPITA VS GDP PER CAPITA)  
   
#Filter data for the specified country  
 country\_co2\_gdp <- co2\_gdp %>%  
 filter(iso3c == country\_name)  
  
#Create scatter plot for CO2 per capita and GDP per capita  
 co2\_gdp\_plot <- country\_co2\_gdp %>%  
 ggplot() +  
  
#Set GDP per capita as the x-axis and co2 per capita as the y-axis  
 aes(x = GDPpercap, y = co2percap) +  
   
#Create scatter plot  
 geom\_point() +  
   
#Create data labels with specified adjustments and size  
 geom\_text(aes(  
 label = year),   
 vjust = -0.5,   
 hjust = 0.5,   
 size = 3) +  
   
#Create labels for the graph  
 labs(x = "GDP per Capita", y = "CO2 per Capita") +  
 ggtitle(paste("CO2 per Capita vs GDP per Capita (", country\_name, ")")) +  
   
#Set theme  
 theme\_minimal()  
   
#GRAPH 3: SCATTER PLOT (ELECTRICITY USAGE VS GDP PER CAPITA)  
  
#Merge energy and GDP per capita data into a single data frame using left join and rename iso3c in GDP per capita data frame to match that of the energy data frame  
 energy\_gdp <- left\_join(energy, gdp\_percap %>% rename(iso\_code = iso3c), by = c("iso\_code", "year"))  
  
#Filter data for the specified country  
 country\_energy\_gdp <- energy\_gdp %>%  
 filter(iso\_code == country\_name)  
  
#Create scatter plot for electricity usage per capita and GDP per capita using gg plot  
 energy\_gdp\_plot <- country\_energy\_gdp %>%  
 ggplot() +  
   
#Set GDP per capita as the x-axis and electricity demand as the y-axis  
 aes(x = GDPpercap, y = electricity\_demand) +  
   
#Create scatter plot  
 geom\_point() +  
   
#Create data labels with specified adjustment on position and size  
 geom\_text(aes(  
 label = year),   
 vjust = -0.5,   
 hjust = 0.5,   
 size = 3) +  
   
#Create labels for the graph  
 labs(x = "GDP per Capita", y = "Electricity Usage (kWh per Capita/Day)") +  
 ggtitle(paste("Electricity Usage vs GDP per Capita (", country\_name, ")")) +  
   
#Set theme  
 theme\_minimal()  
  
#Arrange the plots in the desired layout using the patchwork package  
all\_plots <- electricity\_generation\_plot +  
 plot\_spacer() +  
 co2\_gdp\_plot +  
 energy\_gdp\_plot +  
 plot\_layout(  
 ncol = 2,  
 widths = c(1, 1),  
 heights = c(1, 1),  
 )  
  
# Display the arranged plots  
print(all\_plots)  
  
}  
  
# Generate graphs for the Philippines  
generate\_country\_graphs("PHL")

Warning: Removed 40 rows containing non-finite values (`stat\_align()`).

Warning: Removed 10 rows containing missing values (`geom\_point()`).

Warning: Removed 10 rows containing missing values (`geom\_text()`).



**Answer:** In the case of the Philippines, we can observe from the stacked area chart that the major energy source from 1990s to 2000 is hydro, which should not be the case. The main reason for the difference is that there is no electricity generation data available for coal during that period from the World Bank statistics. However, the graph is correct in portraying the trend in energy source as traditional fossil fuel sources continue to dominate while renewables are gradually increasing. On the other hand, the scatter plots show that as the GDP per capita in the Philippines increased, the CO2 per capita and electricity usage increased as well, as seen in the positive trends from the early 2000s up until the latest date.

# Details

* Who did you collaborate with: N/A
* Approximately how much time did you spend on this problem set: 2 days
* What, if anything, gave you the most trouble: writing the function for the three graphs