Unraveling the Multivariate Determinants of Road Traffic Accident Severity in the UK: A Rigorous Statistical Analysis of Casualty Status 19 using 2019 Road Safety Data in R

MD ABU SUFIAN

2023-04-27

***** loading libraries

```
library(readr)
library(dplyr)
library(ggplot2)
```

****reading the dataset

```
library(readxl)
Accidents_2019_1 <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')
```

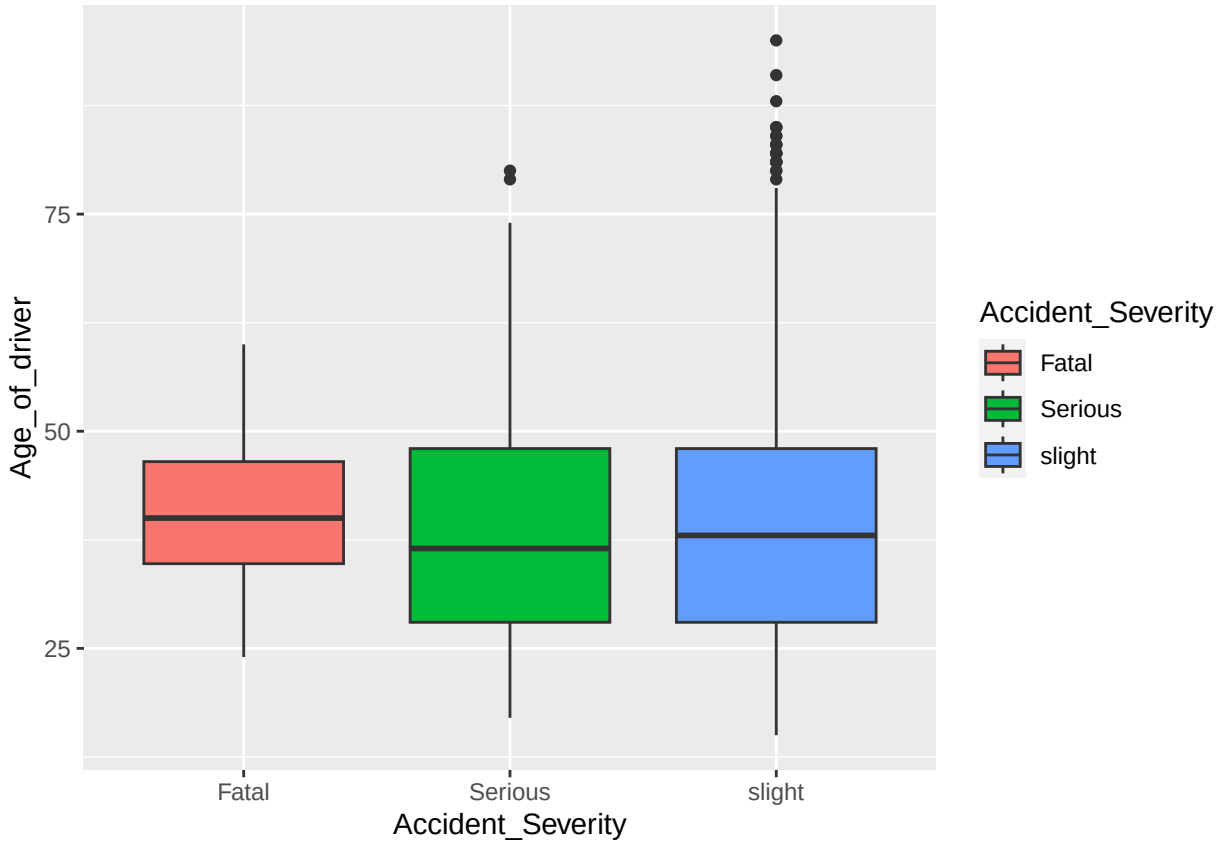
```
# Assuming Accidents_2019_1 is the name of your data frame
column_names <- colnames(Accidents_2019_1)

# Print each column name on a separate line
for (name in column_names) {
  cat(name, "\n")
}
```

```
## Accident_Index
## Location_Easting_OSGR
## Location_Northing_OSGR
## Longitude
## Latitude
## Police_Force
## Accident_Severity
## Number_of_Vehicles
## AgeBandOfDriver
## Number_of_Casualties
## Date
## Day_of_Week
## Time
## Local_Authority_(District)
## Local_Authority_(Highway)
## 1st_Road_Class
## 1st_Road_Number
## Road_Type
```

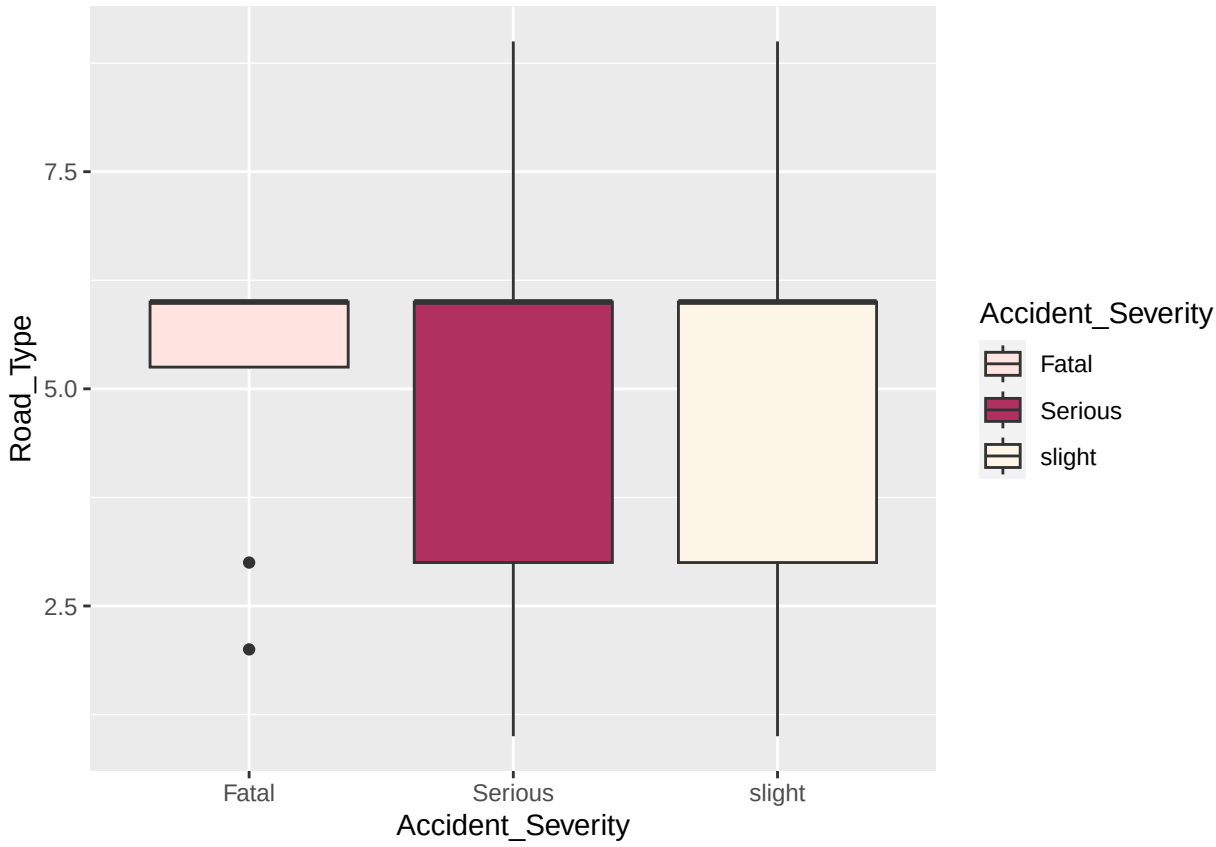
```
## Speed_limit
## Junction_Detail
## Junction_Control
## 2nd_Road_Class
## 2nd_Road_Number
## Pedestrian_Crossing-Human_Control
## Pedestrian_Crossing-Physical_Facilities
## Light_Conditions
## Weather_Conditions
## Road_Surface_Conditions
## Special_Conditions_at_Site
## Carriageway_Hazards
## Urban_or_Rural_Area
## Did_Police_Officer_Attend_Scene_of_Accident
## LSOA_of_Accident_Location
## Age_of_driver
```

#####ploting the relatinship between age and accident severity

```
Bplot <- ggplot(Accidents_2019_1, aes(x = Accident_Severity, y = Age_of_driver, fill = Accident_Severity
  geom_boxplot()
Bplot
```




```
Bplotr <- ggplot(Accidents_2019_1, aes(x = Accident_Severity, y = Road_Type, fill = Accident_Severity))
  geom_boxplot() +
  scale_fill_manual(values = c("Fatal" = "Misty rose", "Serious" = "maroon", "slight" = "Old lace"))
Bplotr
```




```
(Accidents_2019_1$Age_of_driver)
```

```
##    [1] 24 45 33 32 47 25 20 65 24 45 81 34 43 28 42 33 32 30 29 34 45 30 26 35
##   [25] 42 29 19 54 63 45 19 25 20 58 52 65 23 47 35 31 31 42 35 41 42 41 49 32
##   [49] 49 44 55 17 49 48 39 40 33 38 44 22 66 42 37 25 40 47 29 23 55 16 50 45
##   [73] 25 30 44 60 41 41 23 31 39 42 44 23 20 18 27 51 40 47 71 49 57 49 42 41
##   [97] 32 33 30 30 21 70 27 42 48 45 27 52 42 24 21 18 33 18 39 59 59 32 37 34
##  [121] 39 36 30 41 53 25 62 18 50 55 18 42 52 71 36 38 25 40 55 41 43 33 48 26
##  [145] 24 19 24 31 38 38 24 41 49 24 48 28 76 39 27 45 26 30 47 82 41 64 46 56
##  [169] 34 37 46 34 29 31 26 65 23 69 68 38 59 29 40 37 48 56 34 19 26 75 81 21
##  [193] 71 47 25 59 48 58 42 22 19 29 32 33 37 33 45 30 33 53 31 45 22 47 27 20
##  [217] 29 48 42 23 24 18 75 24 38 24 42 35 40 22 58 69 40 43 24 26 46 65 32 60
##  [241] 49 53 60 58 31 56 44 26 29 40 22 32 24 42 32 19 49 24 25 28 23 81 43 39
##  [265] 30 24 21 35 35 41 34 39 31 21 78 47 27 51 59 24 28 36 51 34 28 47 57 52
##  [289] 51 49 43 30 33 38 22 34 51 34 32 33 52 44 26 31 34 23 39 28 31 56 73 25
##  [313] 28 25 26 70 22 37 38 27 43 23 18 29 39 38 33 46 31 47 48 25 18 35 43 52
##  [337] 28 58 31 27 22 35 47 26 68 38 17 30 46 30 35 23 45 46 33 44 47 38 41 34
##  [361] 23 65 55 27 25 27 25 19 34 23 32 35 27 42 42 25 35 21 75 41 28 27 34 63
##  [385] 36 35 53 41 47 72 34 41 56 27 26 36 27 19 61 88 63 68 54 36 39 28 22 52
##  [409] 54 57 31 53 44 39 57 22 48 29 52 71 27 29 28 68 43 31 26 32 58 41 34 24
##  [433] 20 29 73 50 51 42 47 68 36 48 29 59 47 30 37 36 23 32 37 27 44 46 45 32
##  [457] 62 33 43 21 30 44 30 36 50 44 29 45 41 35 47 25 40 28 38 29 26 49 37 26
##  [481] 16 44 20 30 41 32 37 28 58 73 43 68 53 33 38 33 35 24 39 18 31 21 38 33
##  [505] 39 53 37 48 56 22 51 51 36 36 20 49 59 57 39 41 40 20 64 37 74 42 26 27
```

```
##  [529] 32 23 51 45 68 26 61 24 51 23 27 20 22 24 23 35 41 48 26 41 44 52 32 84
##  [553] 74 25 33 46 30 40 31 40 42 37 51 48 38 52 52 32 47 28 20 28 32 55 30 30
##  [577] 29 34 34 37 25 26 48 70 45 32 60 25 36 27 28 41 53 53 25 44 31 57 23 45
##  [601] 37 64 47 37 33 37 30 55 55 42 70 71 57 17 22 23 19 45 25 33 19 26 33 43
##  [625] 19 41 22 20 41 42 24 41 26 80 33 20 42 61 31 41 28 49 57 42 26 55 17 24
##  [649] 35 30 41 24 20 55 55 49 36 31 50 43 62 29 27 83 19 25 45 40 26 37 25 29
##  [673] 25 80 51 70 50 35 58 55 46 25 22 22 28 71 48 32 29 27 35 22 37 36 37 38
##  [697] 42 36 24 55 79 23 20 42 21 22 23 47 49 52 19 37 45 18 28 28 19 27 22 24
##  [721] 42 55 19 85 74 25 44 28 27 21 61 37 35 50 47 24 54 29 33 23 64 46 58 51
##  [745] 31 40 24 32 28 38 36 34 18 41 29 48 31 34 35 32 37 22 26 51 42 49 33 45
##  [769] 37 69 31 21 58 36 47 48 33 59 40 37 31 59 17 45 44 34 55 43 58 18 55 26
##  [793] 25 35 35 19 59 25 36 23 40 62 44 58 29 51 34 54 48 31 37 45 45 26 24 52
##  [817] 29 37 30 54 44 18 50 62 49 51 42 27 23 68 52 28 17 65 67 20 49 28 27 23
##  [841] 43 30 19 51 37 46 30 48 44 27 48 24 19 33 40 34 31 57 23 54 29 29 38 71
##  [865] 25 30 34 41 69 36 44 95 35 52 44 41 16 46 55 35 28 62 33 19 50 41 66 61
##  [889] 54 61 27 26 53 29 36 33 54 22 52 23 48 82 44 43 19 20 49 25 57 57 44 60
##  [913] 43 76 36 38 34 73 28 46 31 38 57 43 52 35 45 40 52 36 19 37 20 42 34 45
##  [937] 43 53 21 44 49 31 42 54 42 49 33 25 20 38 38 61 65 56 40 21 42 38 24 34
##  [961] 33 38 22 33 31 37 30 62 45 32 50 19 55 47 51 27 23 60 50 43 22 20 50 62
##  [985] 29 27 43 22 24 31 62 33 39 51 36 45 58 31 25 43 46 48 57 30 48 39 20 62
## [1009] 46 44 55 25 57 21 40 42 39 29 33 25 36 50 51 47 38 47 26 19 40 49 38 33
## [1033] 56 40 55 44 32 31 34 36 40 24 28 50 49 48 37 79 54 46 31 52 50 39 35 41
## [1057] 47 51 40 36 51 49 36 45 28 26 59 40 36 48 25 25 53 19 35 59 25 33 35 48
## [1081] 21 38 39 21 32 34 34 32 26 36 55 25 85 21 36 29 43 48 25 30 48 44 44 24
## [1105] 43 60 47 53 61 39 49 29 56 34 26 23 47 56 51 47 32 41 31 45 38 31 61 27
## [1129] 38 56 44 33 66 28 43 22 47 54 56 18 36 67 41 42 56 37 38 45 46 52 49 30
## [1153] 57 63 25 77 48 47 22 38 25 22 42 43 23 34 28 66 81 18 56 33 47 29 47 31
## [1177] 26 47 15 33 49 32 33 35 42 26 36 34 57 45 24 36 60 38 32 55 40 51 39 42
## [1201] 49 19 57 32 35 49 26 34 62 45 50 17 60 62 19 38 38 40 47 46 37 50 51 29
## [1225] 59 56 43 76 38 25 36 41 48 48 59 48 32 27 45 29 34 35 19 35 55 55 46 22
## [1249] 24 54 54 21 33 31 44 36 28 41 29 34 52 18 26 31 29 48 54 62 32 30 37 21
## [1273] 19 48 66 18 49 91 23 23 30 57 30 24 28 33 29 58 44 27 44 29 65 30 81 30
## [1297] 41 24 37 68 28 35 43 29 31 53 42 38 40 23 36 23 63 25 24 23 41 61 65 45
## [1321] 17 54 45 35 37 20 21 38 27 29 59 18 39 52 49 21 38 17 23 29 26 60 24 35
## [1345] 50 34 19 23 49 39 61 37 82 22 36 48 42 22 19 57 42 39 41 64 53 44 54 24
## [1369] 40 25 54 45 35 35 50 55 38 73 25 30 32 51 38 25 44 57 51 55 32 28 55 72
## [1393] 36 33 57 37 53 40 31 55 38 28 38 64 15 20 31 25 22 29 37 54 28 60 39 49
## [1417] 55 50 32 28 39 40 53 33 51 31 40 53 57 26 26 37 39 27 35 51 26 40 31 65
## [1441] 53 49 42 47 74 46 41 35 43 54 44 29 43 24 60 22 54 58 34 59 24 19 28 35
## [1465] 32 37 40 23 23 50 76 22 28 60 29 35 75 32 24 40 43 41 32 25 37 22 38 52
## [1489] 52 44 19 51 38 34 23 38 30 26 20 42 41 43 21 28 27 55 75 51 52 27 37 47
## [1513] 31 54 67 35 52 51 24 59 32 28 30 32 40 28 41 38 29 30 39 27 51 60 63 31
## [1537] 24 29 25 83 45 34 56 39 66 25 32 42 36 50 34 27 74 46 36 32 22 34 56 31
## [1561] 48 51 49 35 25 47 46 61 36 38 60 34 56 23 53 26 71 27 33 32 38 63 44 28
## [1585] 59 50 16 63 42 40 50 28 34 33 27 30 64 48 33 32 44 63 27 56 32 46 24 50
## [1609] 32 23 25 35 35 33 46 30 35 55 32 42 33 45 56 34 43 46 16 64 64 31 52 19
## [1633] 26 36 34 31 46 49 35 33 43 60 40 68 84 26 39 31 48 44 36 59 35 26 41 46
## [1657] 55 40 17 34 57 46 40 50 27 59 37 48 17 22 67 29 39 54 39 27 36 38 32 35
## [1681] 43 68 46 24 54 16 42 51 30 41 39 68 46 23 53 45 59 23 35 24 34 22 54 29
## [1705] 33 65 26 34 25 21 43 47 21 19 35 30 25 49 29 44 29 42 43 20 25 28 24 37
## [1729] 41 35 40 18 36 20 35 33 66 23 34 33 28 48 45 61 19 56 39 41 27 21 54 50
## [1753] 45 52 30 27 30 30 26 25 18 47 65 26 44 37 19 41 34 19 35 22 50 26 29 45
## [1777] 49 80 62 41 19 22 42 29 34 40 31 57 46 40 48 45 25 59 23 69 42 23 27 29
## [1801] 53 19 25 41 25 27 45 54 33 62 43 26 26 29 25 21 46 31 42 25 30 36 25 28
```

```
## [1825] 52 28 29 60 43 24 55 27 31 27 34 20 53 85 56 54 53 31 29 62 26 31 75 34
## [1849] 35 72 26 52 43 50 56 47 44 26 32 33 69 32 55 29 30 37 70 40 37 58 44 68
## [1873] 64 21 58 58 31 29 69 35 36 28 57 41 42 56 46 23 48 18 45 26 33 71 52 30
## [1897] 70 23 48 48 18 28 31 19 32 82 38 23 34 55 21 20 27 24 19 20 42 43 29 39
## [1921] 40 42 31 46 48 43 60 43 31 40 27 28 53 53 27 26 53 35 24 75 25 43 30 43
## [1945] 32 46 46 43 24 53 30 38 26 30 45 27 47 31 44 24 83 30 29 32 30 19 24 32
## [1969] 50 37 29 23 45 38 29 44 60 19 20 33 65 67 32 42 57 27 20 29 25 63 78 22
## [1993] 28 42 68 41 41 45 36
```

```
summary(Accidents_2019_1$Age_of_driver)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.00   28.00   37.00   39.46   48.00   95.00
```

```
range_age <- diff(range(Accidents_2019_1$Age_of_driver))
print(range_age)
```

```
## [1] 80
```

```
sd_age <- sd(Accidents_2019_1$Age_of_driver)
print(sd_age)
```

```
## [1] 14.17759
```

```
#Skewness: Measure of the asymmetry of the probability distribution.
library(e1071)
skewness_age <- skewness(Accidents_2019_1$Age_of_driver)
print(skewness_age)
```

```
## [1] 0.7016752
```

```
#Kurtosis: Measure of the "tailedness" of the probability distribution.
library(e1071)
kurtosis_age <- kurtosis(Accidents_2019_1$Age_of_driver)
print(kurtosis_age)
```

```
## [1] 0.1600903
```

```
#Interquartile range (IQR): Difference between the 1st quartile (25th percentile) and the 3rd quartile
iqr_age <- IQR(Accidents_2019_1$Age_of_driver)
print(iqr_age)
```

```
## [1] 20
```

```
#Make sure to load the e1071 package to calculate skewness and kurtosis:
library(e1071)
var_age <- var(Accidents_2019_1$Age_of_driver)
print(var_age)
```

```
## [1] 201.0041
```

```r
cat("Range:", range_age, "\n")
```

```
## Range: 80
```

```r
cat("Variance:", var_age, "\n")
```

```
## Variance: 201.0041
```

```r
cat("Standard Deviation:", sd_age, "\n")
```

```
## Standard Deviation: 14.17759
```

```r
cat("Skewness:", skewness_age, "\n")
```

```
## Skewness: 0.7016752
```

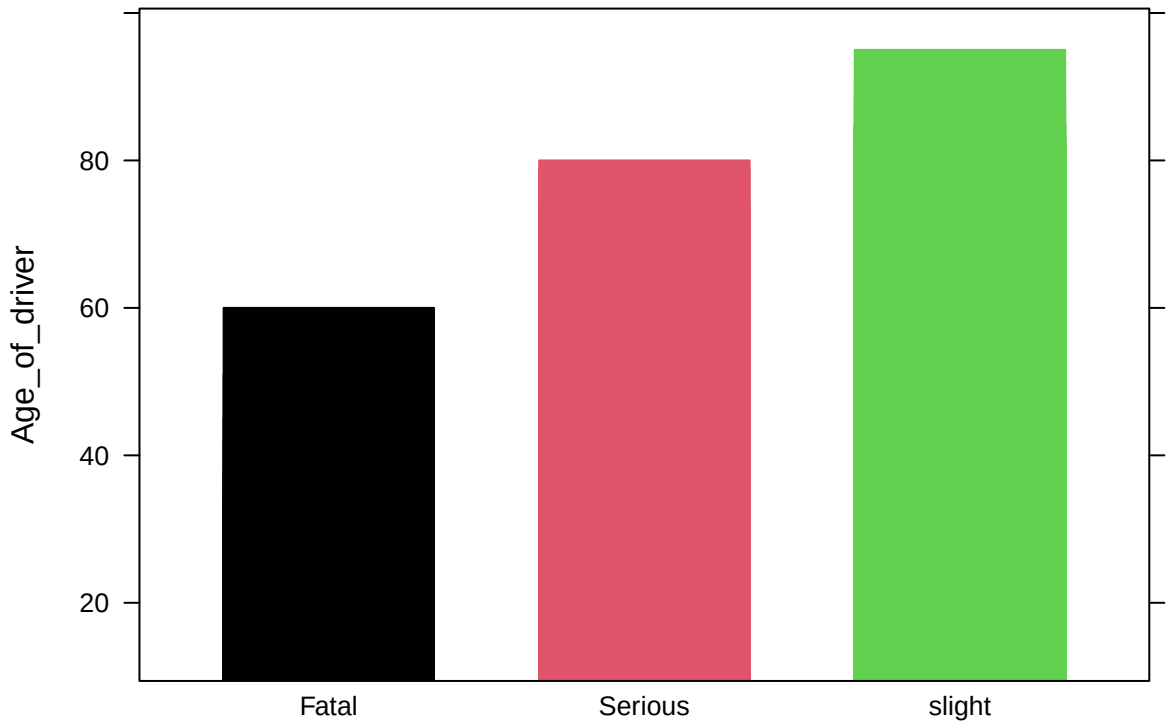
```r
cat("Kurtosis:", kurtosis_age, "\n")
```

```
## Kurtosis: 0.1600903
```

```r
cat("Interquartile Range (IQR):", iqr_age, "\n")
```

```
## Interquartile Range (IQR): 20
```

```r
library(lattice)

# Make sure the Accidents_2019_1 data frame is loaded
#data <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

# Define a custom panel function
panel.custom <- function(x, y, subscripts, col, ...){
  panel.barchart(x, y, col = col[subscripts], border = col[subscripts], ...)
}

# Define the colors for the bars
bar_colors <- with(Accidents_2019_1, factor(Accident_Severity, labels = c("red", "green", "blue")))

# Create the barchart with custom panel function and colors
barchart(Age_of_driver ~ Accident_Severity, data = Accidents_2019_1, panel = panel.custom, col = bar_col
```




```
# Uncomment this line to load the data
# df <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

# Check if the data frame is loaded correctly
print(head(df))
```

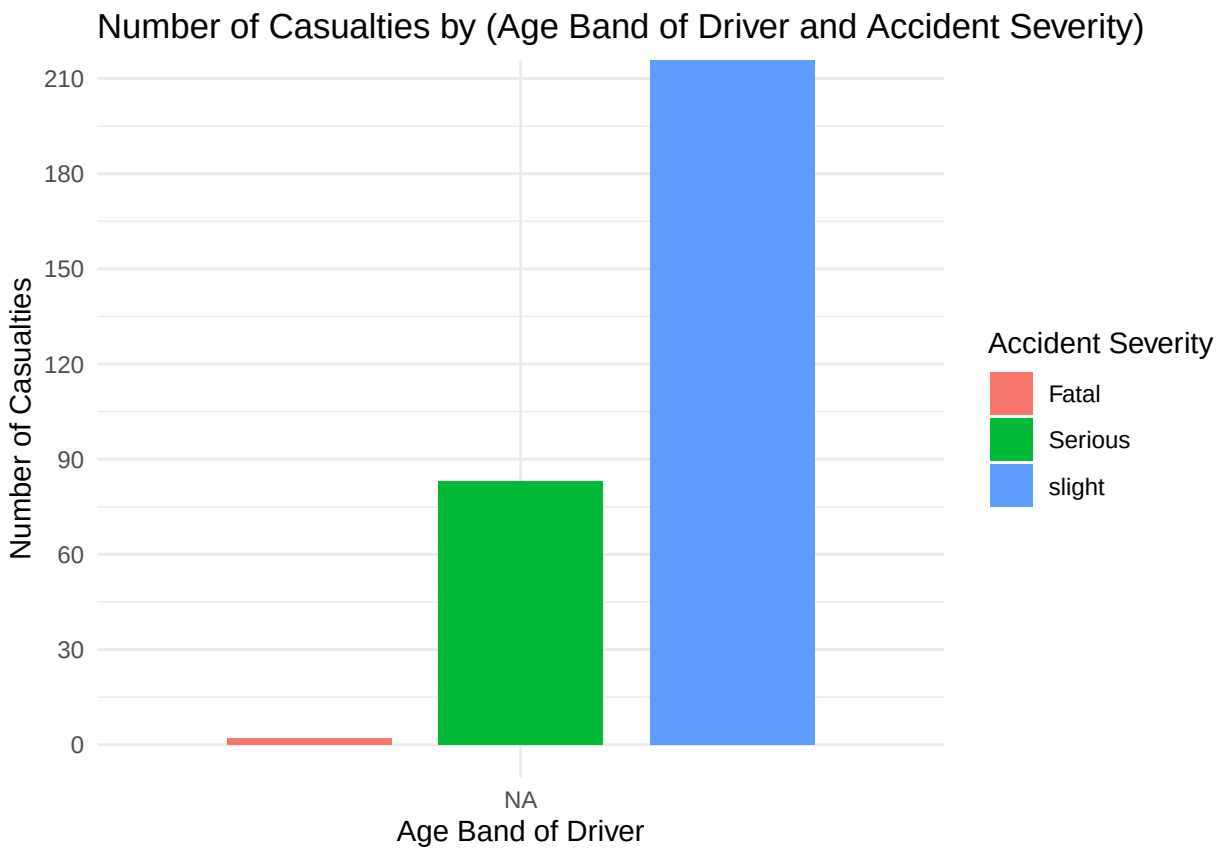
```
##
## 1 function (x, df1, df2, ncp, log = FALSE)
## 2 {
## 3     if (missing(ncp))
## 4         .Call(C_df, x, df1, df2, log)
## 5     else .Call(C_dnf, x, df1, df2, ncp, log)
## 6 }
```

```
# Check if the required columns exist in the data frame
print(colnames(df))
```

```
## NULL
```

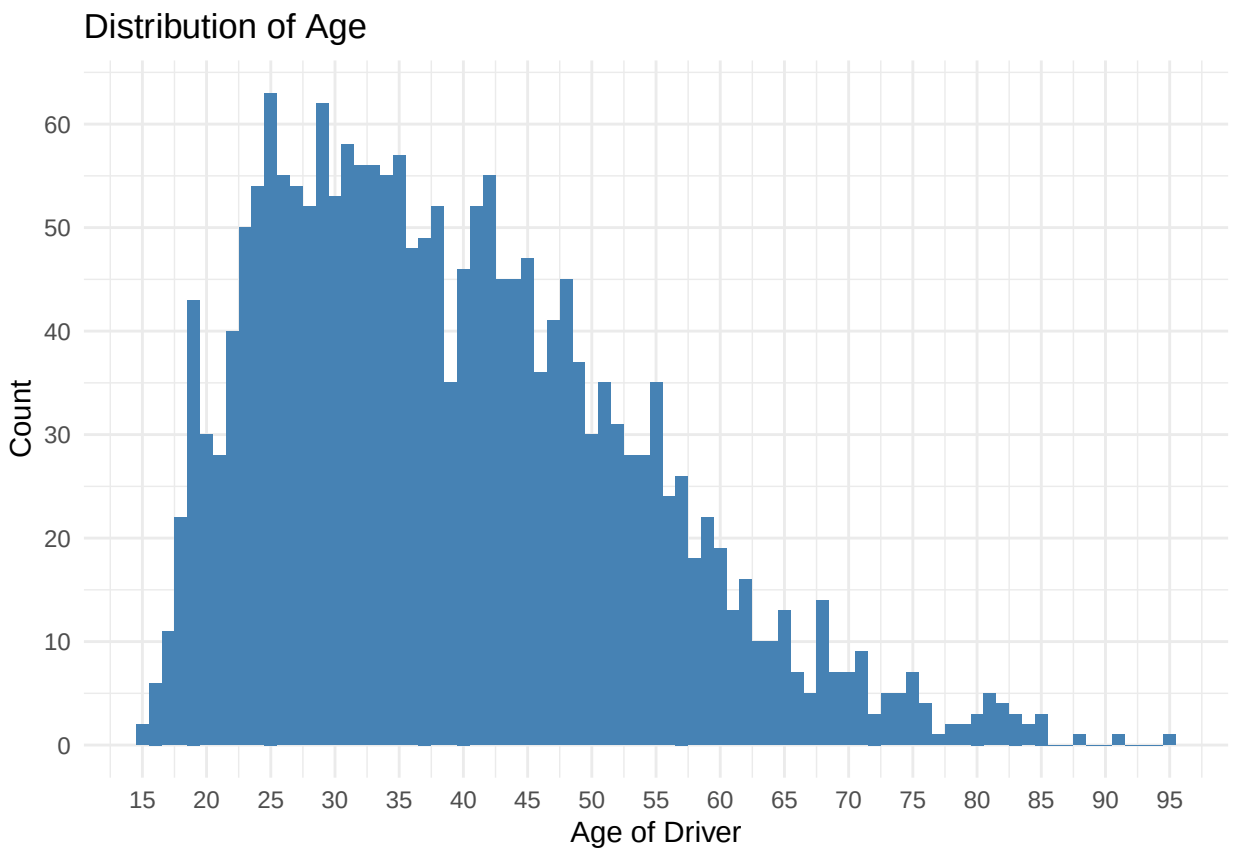
```
library(ggplot2)

# Import the data
df <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

# Create a table with the number of casualties grouped by age band and accident severity
casualty_table <- table(df$AgeBandOfDriver, df$Accident_Severity)

# Convert the table to a data frame
df_table <- as.data.frame.table(casualty_table)
```

```r
# Rename the columns
colnames(df_table) <- c("AgeBandOfDriver", "AccidentSeverity", "Number_Casualties")

# Convert AgeBandOfDriver to a factor variable with the levels in order
df_table$AgeBandOfDriver <- factor(df_table$AgeBandOfDriver, levels = c("0-5", "6-10", "11-15", "16-20"

# Create a bar graph
ggplot(df_table, aes(x = AgeBandOfDriver, y = Number_Casualties, fill = AccidentSeverity)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9), width = 0.7) +
  labs(x = "Age Band of Driver", y = "Number of Casualties", title = "Number of Casualties by (Age Band
  theme_minimal() +
  coord_cartesian(ylim = c(0, max(df_table$Number_Casualties) * 0.5)) +
  scale_y_continuous(breaks = seq(0, max(df_table$Number_Casualties), by = 30))
```

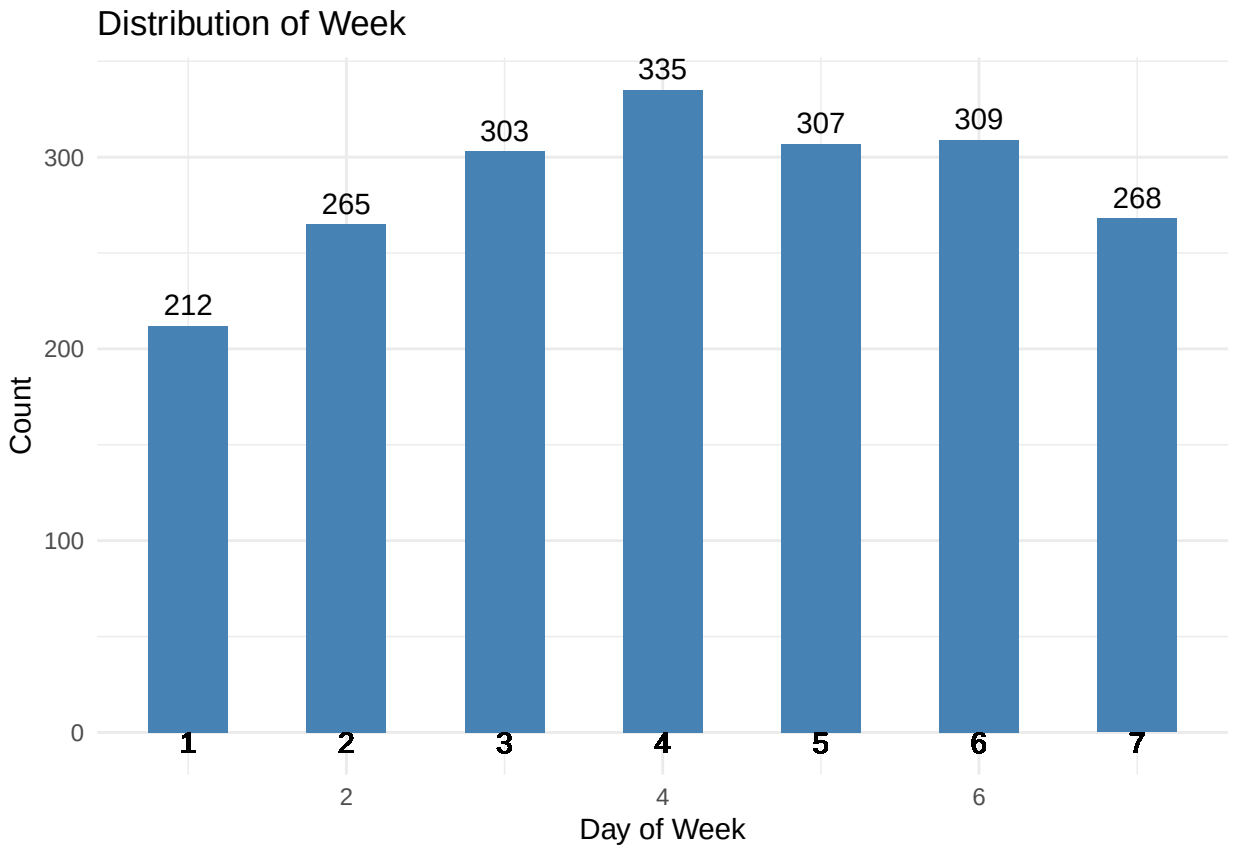



```r
library(ggplot2)

# Make sure the Accidents_2019_1 data frame is loaded
# data <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

hist_d1 <- ggplot(Accidents_2019_1, aes(x = Age_of_driver)) +
  geom_histogram(fill = "steelblue", binwidth = 1) +
  ggtitle("Distribution of Age") +
  labs(x = "Age of Driver", y = "Count") +
  scale_x_continuous(breaks = seq(min(Accidents_2019_1$Age_of_driver), max(Accidents_2019_1$Age_of_drive
  scale_y_continuous(breaks = seq(0, max(hist(Accidents_2019_1$Age_of_driver, plot = FALSE)$counts), by
```

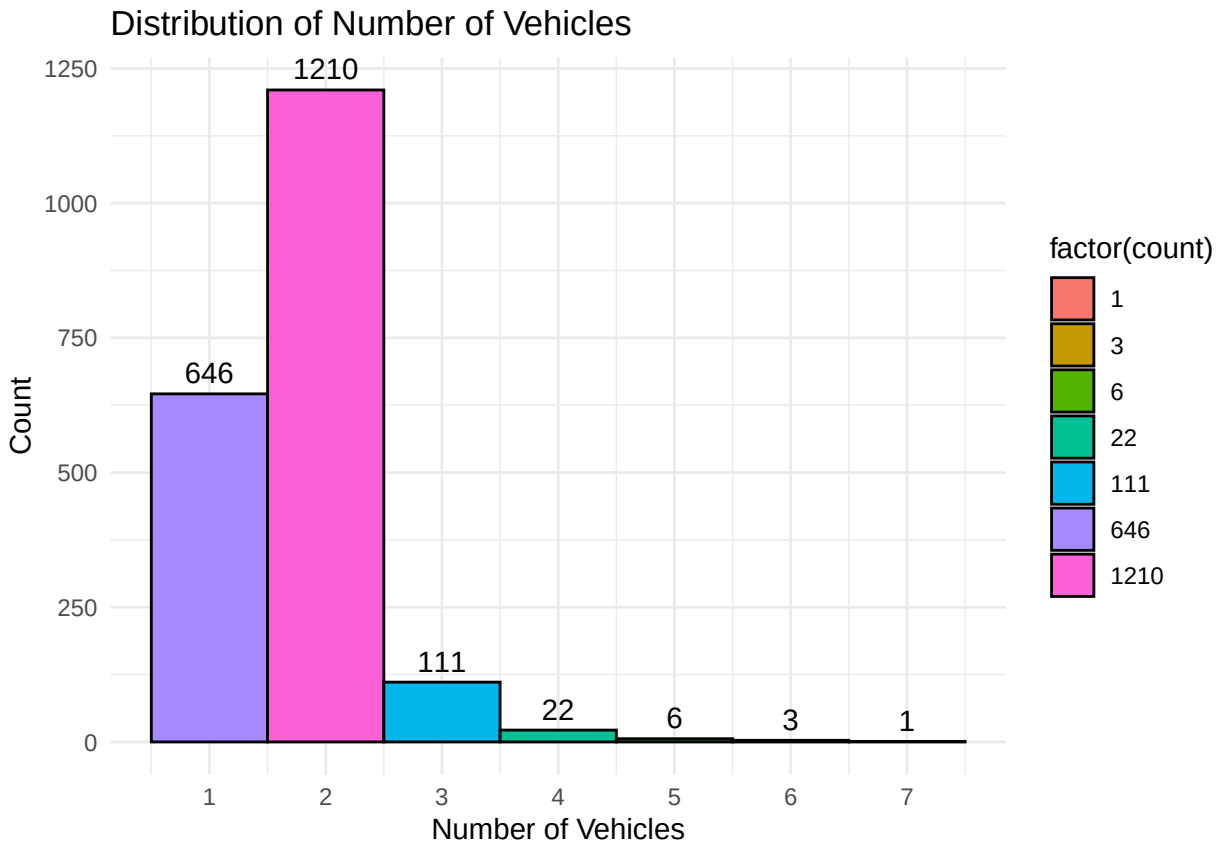
```
  theme_minimal()

hist_d1
```



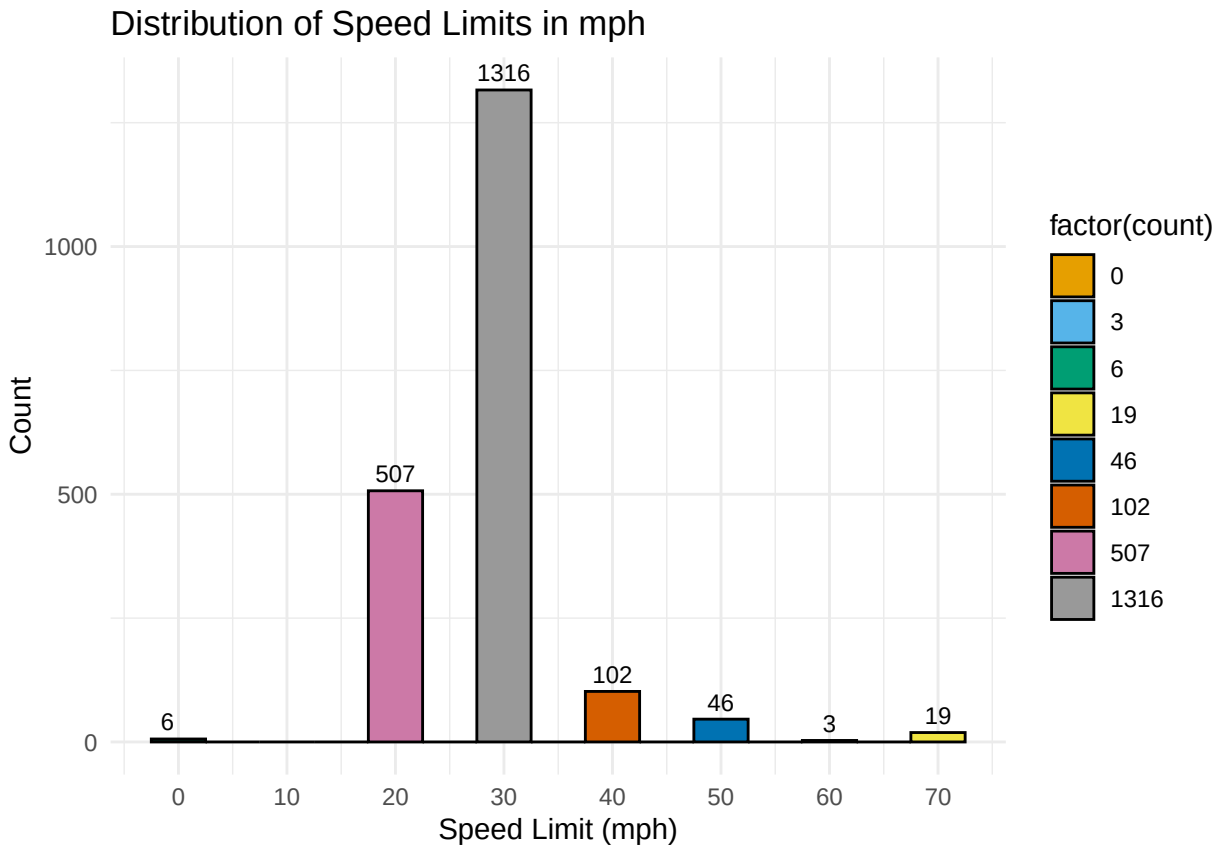

```
library(ggplot2)

# Make sure the Accidents_2019_1 data frame is loaded
# data <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

hist_d2 <- ggplot(Accidents_2019_1, aes(x = Day_of_Week)) +
  geom_bar(aes(y = ..count..), fill = "steelblue", width = 0.5) +
  geom_text(aes(x = Day_of_Week, y = ..count.., label = ..count..),
            stat = "count", vjust = -0.5) +
  geom_text(aes(x = Day_of_Week, y = -5, label = Day_of_Week), hjust = 0.5) +
  ggtitle("Distribution of Week") +
  labs(x = "Day of Week", y = "Count") +
  theme_minimal()

hist_d2
```




```
library(ggplot2)

# Make sure the Accidents_2019_1 data frame is loaded
# data <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

ggplot(Accidents_2019_1, aes(x = Number_of_Vehicles)) +
  geom_histogram(aes(y = ..count.., fill = factor(..count..)),
                 binwidth = 1, color = "black") +
  geom_text(stat = "count", aes(x = Number_of_Vehicles, y = ..count..,
                                label = ..count..), vjust = -0.5) +
  scale_x_continuous(breaks = seq(0, 10, 1)) +
  labs(x = "Number of Vehicles", y = "Count") +
  ggtitle("Distribution of Number of Vehicles") +
  theme_minimal()
```




```
library(ggplot2)

# Make sure the Accidents_2019_1 data frame is loaded
# data <- read_excel('/Users/mdabusufian/Downloads/R_project/Accidents 2019_1.xlsx')

# Define custom colors
my_colors <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7", "#999999")

ggplot(Accidents_2019_1, aes(x = Speed_limit)) +
  geom_histogram(aes(y = ..count.., fill = factor(..count..)), binwidth = 5, color = "black") +
  geom_text(stat = "count", aes(label = ..count.., x = Speed_limit, y = ..count..), vjust = -0.5, size =
  scale_x_continuous(breaks = seq(0, 80, 10)) +
  labs(x = "Speed Limit (mph)", y = "Count") +
  ggtitle("Distribution of Speed Limits in mph") +
  theme_minimal() +
  scale_fill_manual(values = my_colors)
```




```
#### descriptive statistics

summary(Accidents_2019_1)
```

```
##  Accident_Index      Location_Easting_OSGR Location_Northing_OSGR
##  Min.   :2.019e+12   Min.   :503851        Min.   :157259
##  1st Qu.:2.019e+12   1st Qu.:524828        1st Qu.:175768
##  Median :2.019e+12   Median :530877        Median :181220
##  Mean   :2.019e+12   Mean   :530675        Mean   :180631
##  3rd Qu.:2.019e+12   3rd Qu.:536780        3rd Qu.:185808
##  Max.   :2.019e+12   Max.   :558362        Max.   :200283
##
##    Longitude           Latitude      Police_Force Accident_Severity
##  Min.   :-0.50617   Min.   :51.30   Min.   :1     Length:1999
##  1st Qu.:-0.20293   1st Qu.:51.47   1st Qu.:1     Class :character
##  Median :-0.11664   Median :51.51   Median :1     Mode  :character
##  Mean   :-0.11837   Mean   :51.51   Mean   :1
##  3rd Qu.:-0.02885   3rd Qu.:51.56   3rd Qu.:1
##  Max.   : 0.28334   Max.   :51.69   Max.   :1
##
##  Number_of_Vehicles AgeBandOfDriver  Number_of_Casualties     Date
##  Min.   :1.000      Min.   : 0.000   Min.   :1.000        Length:1999
##  1st Qu.:1.000      1st Qu.: 5.000   1st Qu.:1.000        Class :character
##  Median :2.000      Median : 6.000   Median :1.000        Mode  :character
##  Mean   :1.772      Mean   : 6.145   Mean   :1.196
##  3rd Qu.:2.000      3rd Qu.: 7.500   3rd Qu.:1.000
```

```
##  Max.   :7.000      Max.   :11.000   Max.   :6.000
##
##   Day_of_Week        Time                      Local_Authority_(District)
##  Min.   :1.00   Min.   :1899-12-31 00:01:00   Min.   : 1.00
##  1st Qu.:3.00   1st Qu.:1899-12-31 09:38:00   1st Qu.: 7.00
##  Median :4.00   Median :1899-12-31 14:55:00   Median :14.00
##  Mean   :4.13   Mean   :1899-12-31 14:07:36   Mean   :15.28
##  3rd Qu.:6.00   3rd Qu.:1899-12-31 18:20:00   3rd Qu.:25.00
##  Max.   :7.00   Max.   :1899-12-31 23:59:00   Max.   :32.00
##                 NA's   :2
##  Local_Authority_(Highway) 1st_Road_Class  1st_Road_Number    Road_Type
##  Length:1999               Min.   :1.000   Min.   :   0.0   Min.   :1.000
##  Class :character          1st Qu.:3.000   1st Qu.:   0.0   1st Qu.:3.000
##  Mode  :character          Median :3.000   Median : 105.0   Median :6.000
##                            Mean   :3.851   Mean   : 471.6   Mean   :5.002
##                            3rd Qu.:5.000   3rd Qu.: 316.0   3rd Qu.:6.000
##                            Max.   :6.000   Max.   :5205.0   Max.   :9.000
##
##   Speed_limit    Junction_Detail  Junction_Control 2nd_Road_Class
##  Min.   :-1.00   Min.   :-1.000   Min.   :-1.000   Min.   :-1.00
##  1st Qu.:20.00   1st Qu.: 0.000   1st Qu.:-1.000   1st Qu.: 3.00
##  Median :30.00   Median : 3.000   Median : 2.000   Median : 5.00
##  Mean   :28.77   Mean   : 3.301   Mean   : 1.934   Mean   : 3.79
##  3rd Qu.:30.00   3rd Qu.: 6.000   3rd Qu.: 4.000   3rd Qu.: 6.00
##  Max.   :70.00   Max.   : 9.000   Max.   : 4.000   Max.   : 6.00
##
##  2nd_Road_Number  Pedestrian_Crossing-Human_Control
##  Min.   :   0.0   Min.   :-1.00000
##  1st Qu.:   0.0   1st Qu.: 0.00000
##  Median :   0.0   Median : 0.00000
##  Mean   : 192.5   Mean   :-0.05053
##  3rd Qu.:   0.0   3rd Qu.: 0.00000
##  Max.   :5203.0   Max.   : 2.00000
##
##  Pedestrian_Crossing-Physical_Facilities Light_Conditions Weather_Conditions
##  Min.   :-1.00                           Min.   :1.000    Min.   :1.000
##  1st Qu.: 0.00                           1st Qu.:1.000    1st Qu.:1.000
##  Median : 0.00                           Median :4.000    Median :1.000
##  Mean   : 1.53                           Mean   :2.747    Mean   :1.808
##  3rd Qu.: 4.00                           3rd Qu.:4.000    3rd Qu.:1.000
##  Max.   : 8.00                           Max.   :7.000    Max.   :9.000
##
##  Road_Surface_Conditions Special_Conditions_at_Site Carriageway_Hazards
##  Min.   :-1.00           Min.   :-1.00000           Min.   :-1.00000
##  1st Qu.: 1.00           1st Qu.: 0.00000           1st Qu.: 0.00000
##  Median : 1.00           Median : 0.00000           Median : 0.00000
##  Mean   : 1.28           Mean   : 0.02551           Mean   : 0.02101
##  3rd Qu.: 2.00           3rd Qu.: 0.00000           3rd Qu.: 0.00000
##  Max.   : 4.00           Max.   : 7.00000           Max.   : 7.00000
##
##  Urban_or_Rural_Area Did_Police_Officer_Attend_Scene_of_Accident
##  Min.   :1.000       Min.   :1.00
##  1st Qu.:1.000       1st Qu.:1.00
##  Median :1.000       Median :1.00
```

```
##  Mean   :1.037       Mean   :1.59
##  3rd Qu.:1.000       3rd Qu.:3.00
##  Max.   :2.000       Max.   :3.00
##
##  LSOA_of_Accident_Location Age_of_driver
##  Length:1999               Min.   :15.00
##  Class :character          1st Qu.:28.00
##  Mode  :character          Median :37.00
##                            Mean   :39.46
##                            3rd Qu.:48.00
##                            Max.   :95.00
##
```

```
#### Correlation Analysis
####The correlation coefficient is a statistical measure that describes the direction and strength of t

cor(Accidents_2019_1$Age_of_driver, Accidents_2019_1$Number_of_Casualties)
```

```
## [1] 0.002893352
```

```
### regression analysis

model_casualties<-lm(Number_of_Casualties~Age_of_driver+Speed_limit+Weather_Conditions+Light_Conditions
model_casualties
```

```
##
## Call:
## lm(formula = Number_of_Casualties ~ Age_of_driver + Speed_limit +
##     Weather_Conditions + Light_Conditions + Road_Type, data = Accidents_2019_1)
##
## Coefficients:
##        (Intercept)       Age_of_driver         Speed_limit  Weather_Conditions
##          0.8590780           0.0002889           0.0105837          -0.0083090
##   Light_Conditions           Road_Type
##          0.0086377           0.0023913
```

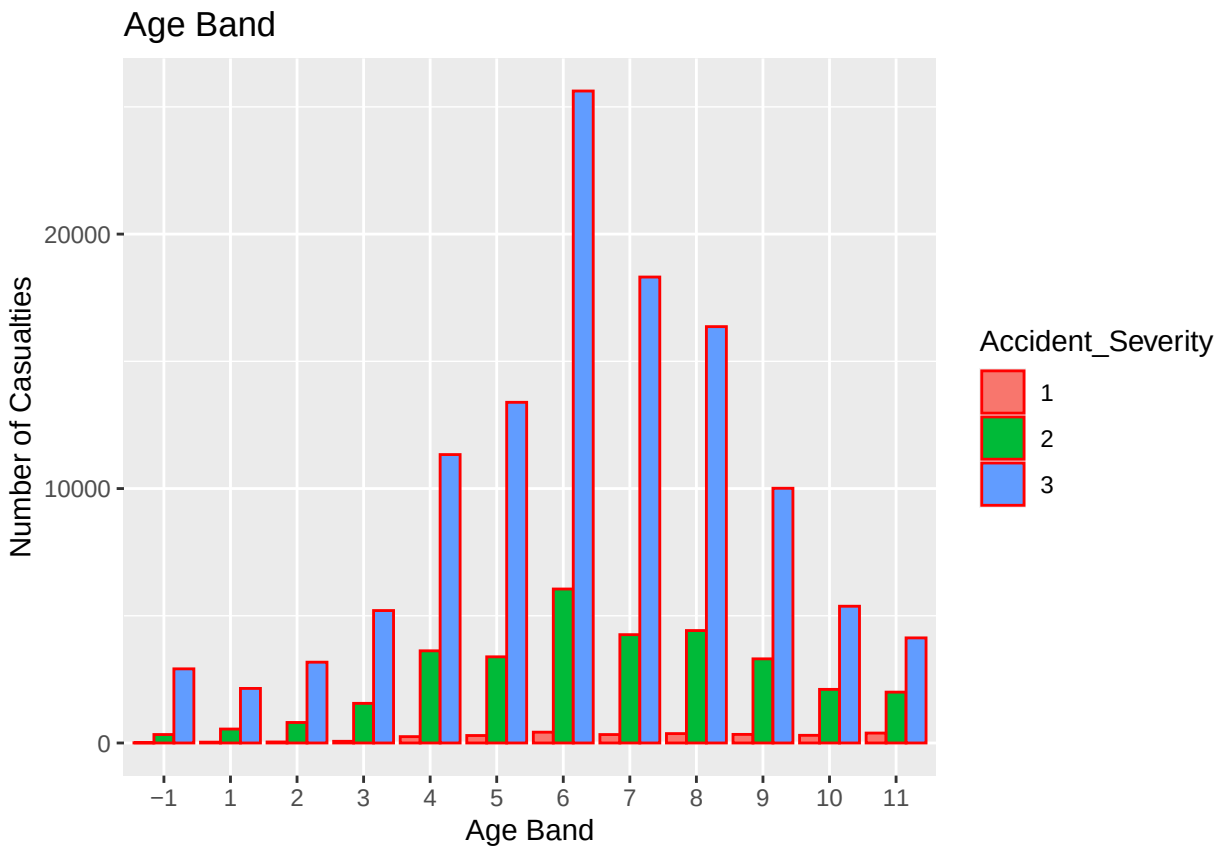
```
summary(model_casualties)
```

```
##
## Call:
## lm(formula = Number_of_Casualties ~ Age_of_driver + Speed_limit +
##     Weather_Conditions + Light_Conditions + Road_Type, data = Accidents_2019_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6769 -0.2221 -0.1970 -0.1001  4.8128
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.8590780  0.0791672  10.851  < 2e-16 ***
## Age_of_driver       0.0002889  0.0009137   0.316    0.752
## Speed_limit         0.0105837  0.0017403   6.081 1.42e-09 ***
```

```
## Weather_Conditions -0.0083090  0.0063179  -1.315    0.189
## Light_Conditions    0.0086377  0.0070818   1.220    0.223
## Road_Type           0.0023913  0.0069203   0.346    0.730
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5784 on 1993 degrees of freedom
## Multiple R-squared:  0.02032,    Adjusted R-squared:  0.01787
## F-statistic: 8.269 on 5 and 1993 DF,  p-value: 9.591e-08
```

```
Casualties <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Casualties 20
Accidents <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Accidents 2019
Vehicles <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data- Vehicles 2019.csv
Population <- read_xlsx('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Population.xlsx')

## Combine Data into 1 document using 'Accident_Index' as reference
temp <- merge(Casualties, Accidents)
final <- merge(temp, Vehicles)
data <- final %>% select(Accident_Index, Accident_Severity, Age_Band_of_Casualty, Age_of_Driver)
data$Age_Band_of_Casualty <- as.factor(data$Age_Band_of_Casualty)
```

```
data$Accident_Severity <- as.factor(data$Accident_Severity)
ggplot(data, aes(x=Age_Band_of_Casualty, fill= Accident_Severity)) + geom_bar(color="red", position="do
```



###Here above graph shown that 26-35 is the age band with highest casualties and that 20-55 is age band of more casualties. Note, the only other point of attention is the higher than expected fatal severity casualties aged over 75.

```r
data %>% group_by(Age_Band_of_Casualty) %>% summarise(Accident_Index = n())
```

```
## # A tibble: 12 x 2
##    Age_Band_of_Casualty Accident_Index
##    <fct>                         <int>
##  1 -1                             3255
##  2 1                              2724
##  3 2                              4022
##  4 3                              6828
##  5 4                             15205
##  6 5                             17068
##  7 6                             32098
##  8 7                             22900
##  9 8                             21150
## 10 9                             13655
## 11 10                             7782
## 12 11                             6519
```

```r
# Load necessary packages
library(dplyr)
library(readr)
library(readxl)
library(janitor)

# Load the Accidents data frame
Accidents <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Accidents 2019

# Create the Conditions data frame
Conditions <- Accidents %>%
  select(1, 6:11, 14, 17, 25:29)

# Load the Population data frame
Population <- read_xlsx('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Population.xlsx')

# Create the AccidentConditionsV2 data frame
Code <- Conditions$`Local_Authority_(Highway)`
AccidentConditionsV2 <- mutate(Conditions, Code)

# Merge Population and AccidentConditionsV2 data frames
AccidentsList <- merge(Population, AccidentConditionsV2)

# Check the class of the 'Name' column in the AccidentsList data frame
class(AccidentsList$Name)
```

```
## [1] "NULL"
```

```r
# Obtain a summary of the 'Name' column in the AccidentsList data frame
summary(AccidentsList$Name)
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```r
# Obtain a summary of the entire AccidentsList data frame
summary(AccidentsList)
```

```
##      Code             Place Name           All ages       Accident_Index
##  Length:112867      Length:112867      Min.   :   2224   Length:112867
##  Class :character   Class :character   1st Qu.: 261317   Class :character
##  Mode  :character   Mode  :character   Median : 353134   Mode  :character
##                                        Mean   : 555541
##                                        3rd Qu.: 793139
##                                        Max.   :1581555
##   Police_Force   Accident_Severity Number_of_Vehicles Number_of_Casualties
##  Min.   : 1.00   Min.   :1.000     Min.   : 1.000     Min.   : 1.000
##  1st Qu.: 4.00   1st Qu.:3.000     1st Qu.: 1.000     1st Qu.: 1.000
##  Median :22.00   Median :3.000     Median : 2.000     Median : 1.000
##  Mean   :26.96   Mean   :2.774     Mean   : 1.841     Mean   : 1.302
##  3rd Qu.:45.00   3rd Qu.:3.000     3rd Qu.: 2.000     3rd Qu.: 1.000
##  Max.   :98.00   Max.   :3.000     Max.   :17.000     Max.   :52.000
##      Date             Day_of_Week    Local_Authority_(Highway)   Road_Type
##  Length:112867      Min.   :1.000   Length:112867              Min.   :1.000
##  Class :character   1st Qu.:2.000   Class :character           1st Qu.:6.000
##  Mode  :character   Median :4.000   Mode  :character           Median :6.000
##                     Mean   :4.112                              Mean   :5.214
##                     3rd Qu.:6.000                              3rd Qu.:6.000
##                     Max.   :7.000                              Max.   :9.000
##  Light_Conditions Weather_Conditions Road_Surface_Conditions
##  Min.   :-1.000   Min.   :1.000      Min.   :-1.00
##  1st Qu.: 1.000   1st Qu.:1.000      1st Qu.: 1.00
##  Median : 1.000   Median :1.000      Median : 1.00
##  Mean   : 2.051   Mean   :1.656      Mean   : 1.29
##  3rd Qu.: 4.000   3rd Qu.:1.000      3rd Qu.: 2.00
##  Max.   : 7.000   Max.   :9.000      Max.   : 5.00
##  Special_Conditions_at_Site Carriageway_Hazards
##  Min.   :-1.00000           Min.   :-1.00000
##  1st Qu.: 0.00000           1st Qu.: 0.00000
##  Median : 0.00000           Median : 0.00000
##  Mean   : 0.07379           Mean   : 0.04289
##  3rd Qu.: 0.00000           3rd Qu.: 0.00000
##  Max.   : 7.00000           Max.   : 7.00000
```

```r
# Create a frequency table for the 'Place Name' column in the AccidentsList data frame
tabyl(AccidentsList$`Place Name`) %>%
  arrange(desc(percent))
```

```
##    AccidentsList$‘Place Name‘    n      percent
##                         Kent 3619 3.206429e-02
##                       Surrey 2964 2.626100e-02
##                   Birmingham 2623 2.323974e-02
##                        Essex 2385 2.113107e-02
##                    Hampshire 2385 2.113107e-02
##                    Lancashire 2306 2.043113e-02
##                   West Sussex 1983 1.756935e-02
##                  Hertfordshire 1960 1.736557e-02
##                   Lincolnshire 1893 1.677195e-02
```

```
##                       Norfolk 1648 1.460126e-02
##                        Devon 1563 1.384816e-02
##                   Westminster 1521 1.347604e-02
##                Nottinghamshire 1502 1.330770e-02
##                         Leeds 1451 1.285584e-02
##                       Suffolk 1362 1.206730e-02
##                   East Sussex 1327 1.175720e-02
##                 Cambridgeshire 1258 1.114586e-02
##                      Derbyshire 1238 1.096866e-02
##                         Lambeth 1191 1.055224e-02
##                       Cornwall 1131 1.002064e-02
##                  Tower Hamlets 1131 1.002064e-02
##                Northamptonshire 1113 9.861164e-03
##                    Warwickshire 1108 9.816864e-03
##                      Southwark 1096 9.710544e-03
##                   Oxfordshire 1093 9.683964e-03
##                       Cumbria 1025 9.081485e-03
##                North Yorkshire 1024 9.072625e-03
##                       Somerset  994 8.806826e-03
##                         Ealing  983 8.709366e-03
##                Bristol, City of  967 8.567606e-03
##                       Croydon  955 8.461286e-03
##                  Staffordshire  953 8.443566e-03
##                         Barnet  948 8.399266e-03
##                        Enfield  948 8.399266e-03
##                     Wandsworth  945 8.372686e-03
##                      Wiltshire  942 8.346106e-03
##                         Camden  922 8.168907e-03
##                      Sheffield  909 8.053727e-03
##                      Liverpool  907 8.036007e-03
##                     Manchester  893 7.911967e-03
##                         Newham  879 7.787927e-03
##                  Worcestershire  879 7.787927e-03
##                         Hackney  862 7.637308e-03
##                           Brent  861 7.628448e-03
##                      Nottingham  852 7.548708e-03
##                        Haringey  849 7.522128e-03
##                Gloucestershire  845 7.486688e-03
##                        Bradford  833 7.380368e-03
##                        Lewisham  795 7.043689e-03
##   Kingston upon Hull, City of  789 6.990529e-03
##                        Hounslow  754 6.680429e-03
##                       Islington  738 6.538669e-03
##              City of Edinburgh  728 6.450070e-03
##                       Greenwich  724 6.414630e-03
##                  Leicestershire  720 6.379190e-03
##                        Bromley  709 6.281730e-03
##         Kensington and Chelsea  705 6.246290e-03
##         Hammersmith and Fulham  702 6.219710e-03
##                       Redbridge  699 6.193130e-03
##       East Riding of Yorkshire  686 6.077950e-03
##                   County Durham  684 6.060230e-03
##                      Hillingdon  683 6.051370e-03
##              Brighton and Hove  681 6.033650e-03
```

```
##               Waltham Forest  659 5.838731e-03
##                   Doncaster  646 5.723551e-03
##               Cheshire East  622 5.510911e-03
##         Barking and Dagenham  606 5.369151e-03
##                    Havering  585 5.183092e-03
##         Central Bedfordshire  583 5.165372e-03
##                     Kirklees  577 5.112212e-03
##                       Medway  577 5.112212e-03
##                    Coventry  572 5.067912e-03
##                    Sandwell  570 5.050192e-03
##    Cheshire West and Chester  563 4.988172e-03
##                    Leicester  555 4.917292e-03
##          Newcastle upon Tyne  538 4.766672e-03
##                  Southampton  506 4.483153e-03
##                Wolverhampton  489 4.332533e-03
##                    Wakefield  487 4.314813e-03
##                      Walsall  485 4.297093e-03
##                   Portsmouth  484 4.288233e-03
##                     Plymouth  483 4.279373e-03
##                       Merton  477 4.226213e-03
##                       Bexley  475 4.208493e-03
##                        Derby  467 4.137613e-03
##                   Shropshire  467 4.137613e-03
##                Milton Keynes  463 4.102173e-03
##                       Sutton  442 3.916114e-03
##                 Peterborough  438 3.880674e-03
##                      Cardiff  436 3.862954e-03
##                       Luton  436 3.862954e-03
##                    Rotherham  436 3.862954e-03
##                       Harrow  428 3.792074e-03
##      North East Lincolnshire  426 3.774354e-03
##        South Gloucestershire  425 3.765494e-03
##                   Sunderland  422 3.738914e-03
##         Richmond upon Thames  419 3.712334e-03
##                       Sefton  414 3.668034e-03
##                       Dudley  411 3.641454e-03
##           North Lincolnshire  410 3.632594e-03
##                       Wirral  410 3.632594e-03
##                      Salford  393 3.481974e-03
##                      Swindon  391 3.464254e-03
##                     Barnsley  387 3.428814e-03
##             Carmarthenshire  385 3.411094e-03
##                       Bolton  381 3.375655e-03
##                      Bedford  371 3.287055e-03
##     Herefordshire, County of  361 3.198455e-03
##               Stoke-on-Trent  359 3.180735e-03
##                   Warrington  354 3.136435e-03
##                         York  352 3.118715e-03
##                        Powys  350 3.100995e-03
##         Kingston upon Thames  337 2.985815e-03
##                     Highland  335 2.968095e-03
##            South Lanarkshire  333 2.950375e-03
##        Blackburn with Darwen  321 2.844055e-03
##                    Blackpool  321 2.844055e-03
```

```
##                         Oldham  315 2.790895e-03
##                          Wigan  305 2.702296e-03
##              Rhondda Cynon Taf  301 2.666856e-03
##                       Stockport  299 2.649136e-03
##                  North Somerset  296 2.622556e-03
##                        Swansea  288 2.551676e-03
##                  City of London  282 2.498516e-03
##                   Isle of Wight  282 2.498516e-03
##                        Rochdale  277 2.454216e-03
##                        Trafford  276 2.445356e-03
##                          Slough  275 2.436496e-03
##                        Thurrock  272 2.409916e-03
##                      Calderdale  271 2.401056e-03
##                        Solihull  266 2.356756e-03
##                  North Tyneside  264 2.339036e-03
##                         Newport  256 2.268156e-03
##                 Southend-on-Sea  256 2.268156e-03
##  Bath and North East Somerset  253 2.241576e-03
##                            Bury  250 2.214996e-03
##                   Pembrokeshire  248 2.197276e-03
##                Stockton-on-Tees  239 2.117537e-03
##                        Tameside  234 2.073237e-03
##                      St. Helens  229 2.028937e-03
##                         Reading  227 2.011217e-03
##                          Torbay  226 2.002357e-03
##         Windsor and Maidenhead  214 1.896037e-03
##                   West Lothian  212 1.878317e-03
##                  West Berkshire  210 1.860597e-03
##                  Aberdeenshire  197 1.745417e-03
##          Dumfries and Galloway  195 1.727697e-03
##                         Halton  190 1.683397e-03
##                  Middlesbrough  190 1.683397e-03
##                        Knowsley  189 1.674537e-03
##                         Gwynedd  185 1.639097e-03
##                  South Tyneside  185 1.639097e-03
##                      Caerphilly  184 1.630237e-03
##                      Flintshire  176 1.559357e-03
##                           Conwy  172 1.523918e-03
##              Telford and Wrekin  172 1.523918e-03
##                        Bridgend  170 1.506198e-03
##                       Wokingham  168 1.488478e-03
##                      Darlington  167 1.479618e-03
##                    Renfrewshire  162 1.435318e-03
##                      Ceredigion  160 1.417598e-03
##                         Wrexham  153 1.355578e-03
##              Neath Port Talbot  146 1.293558e-03
##                   Denbighshire  145 1.284698e-03
##                Scottish Borders  145 1.284698e-03
##                 Argyll and Bute  143 1.266978e-03
##              Vale of Glamorgan  133 1.178378e-03
##           Redcar and Cleveland  128 1.134078e-03
##                         Falkirk  127 1.125218e-03
##                  North Ayrshire  127 1.125218e-03
##                Bracknell Forest  126 1.116358e-03
```

```
##                   Dundee City  126 1.116358e-03
##                 Monmouthshire  125 1.107498e-03
##                South Ayrshire  122 1.080918e-03
##                      Stirling  122 1.080918e-03
##                    Midlothian  114 1.010038e-03
##                 Aberdeen City  113 1.001178e-03
##                    Hartlepool  105 9.302985e-04
##                 East Ayrshire  104 9.214385e-04
##                  East Lothian  101 8.948586e-04
##                    Inverclyde   97 8.594186e-04
##                         Angus   93 8.239787e-04
##                       Torfaen   83 7.353788e-04
##              Isle of Anglesey   82 7.265188e-04
##           West Dunbartonshire   74 6.556389e-04
##                Merthyr Tydfil   72 6.379190e-04
##                 Blaenau Gwent   67 5.936190e-04
##              East Renfrewshire   66 5.847591e-04
##                         Moray   53 4.695792e-04
##                       Rutland   41 3.632594e-04
##               Clackmannanshire   35 3.100995e-04
##            Na h-Eileanan Siar   25 2.214996e-04
##                Orkney Islands   24 2.126397e-04
##               Shetland Islands   21 1.860597e-04
##                Isles of Scilly    1 8.859986e-06
```

```r
# Print column names of the AccidentsList data frame
print(colnames(AccidentsList))
```

```
##  [1] "Code"                         "Place Name"
##  [3] "All ages"                     "Accident_Index"
##  [5] "Police_Force"                 "Accident_Severity"
##  [7] "Number_of_Vehicles"           "Number_of_Casualties"
##  [9] "Date"                         "Day_of_Week"
## [11] "Local_Authority_(Highway)"    "Road_Type"
## [13] "Light_Conditions"             "Weather_Conditions"
## [15] "Road_Surface_Conditions"      "Special_Conditions_at_Site"
## [17] "Carriageway_Hazards"
```

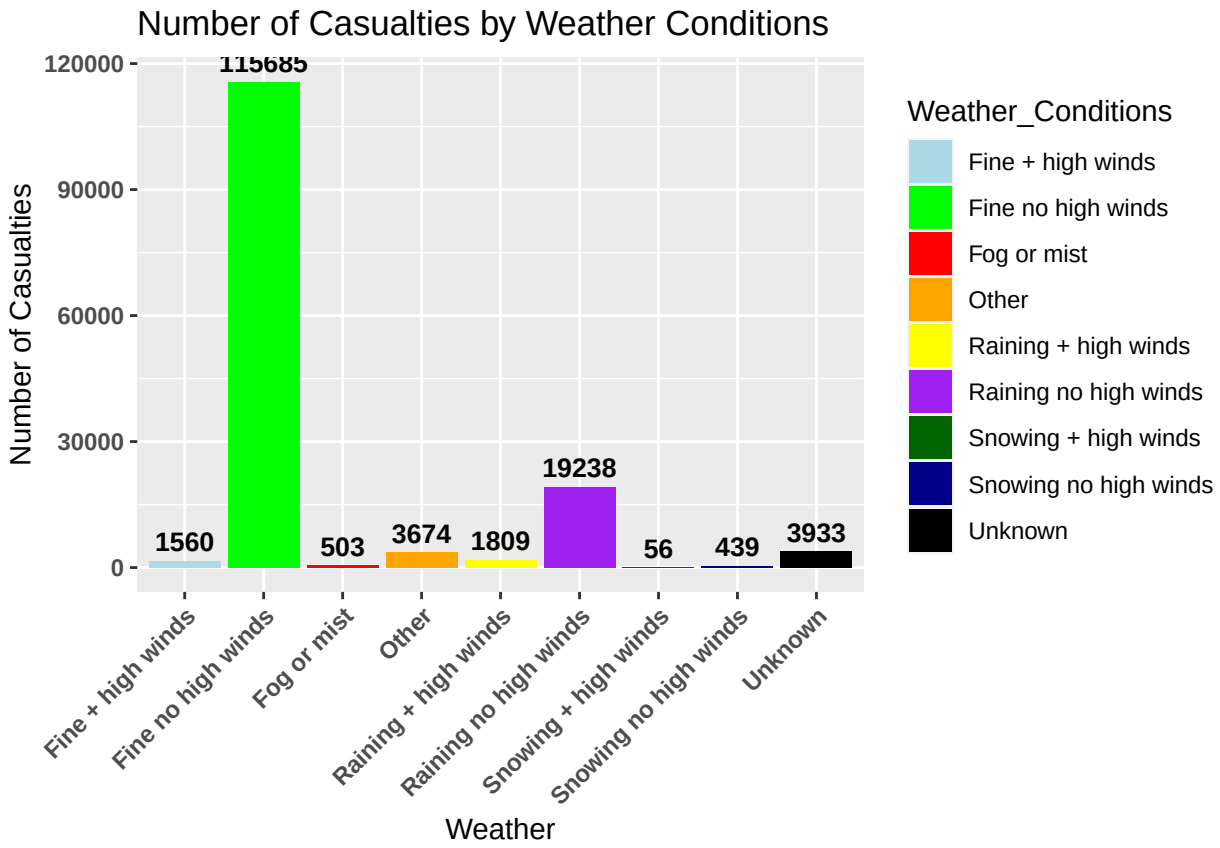
```r
AccidentsList <- na.omit(AccidentsList)
```

```r
colnames(AccidentsList)
```

```
##  [1] "Code"                         "Place Name"
##  [3] "All ages"                     "Accident_Index"
##  [5] "Police_Force"                 "Accident_Severity"
##  [7] "Number_of_Vehicles"           "Number_of_Casualties"
##  [9] "Date"                         "Day_of_Week"
## [11] "Local_Authority_(Highway)"    "Road_Type"
## [13] "Light_Conditions"             "Weather_Conditions"
## [15] "Road_Surface_Conditions"      "Special_Conditions_at_Site"
## [17] "Carriageway_Hazards"
```

```r
library(dplyr)

AccidentsList_agg <- AccidentsList %>%
  group_by(Weather_Conditions) %>%
  summarize(Number_of_Casualties = sum(Number_of_Casualties, na.rm = TRUE)) %>%
  mutate(Weather_Conditions = factor(Weather_Conditions))
```

```r
# First, create a named vector to map weather condition codes to their names
weather_names <- c(
  "1" = "Fine no high winds",
  "2" = "Raining no high winds",
  "3" = "Snowing no high winds",
  "4" = "Fine + high winds",
  "5" = "Raining + high winds",
  "6" = "Snowing + high winds",
  "7" = "Fog or mist",
  "8" = "Other",
  "9" = "Unknown"
)

# Next, map the weather condition codes to their names
AccidentsList_agg$Weather_Conditions <- weather_names[as.character(AccidentsList_agg$Weather_Conditions)

# Now, create the plot with the updated data
ggplot(AccidentsList_agg, aes(x = Weather_Conditions, y = Number_of_Casualties, fill = Weather_Condition
  geom_col() +
  scale_fill_manual(values = c("lightblue", "green", "red", "orange", "yellow", "purple", "darkgreen",
  labs(x = "Weather", y = "Number of Casualties", title = "Number of Casualties by Weather Conditions")
  theme(axis.text.x = element_text(angle = 45, hjust = 1, face = "bold"), axis.text.y = element_text(fa
  geom_text(aes(label = Number_of_Casualties), vjust = -0.5, size = 3.5, fontface = "bold")
```

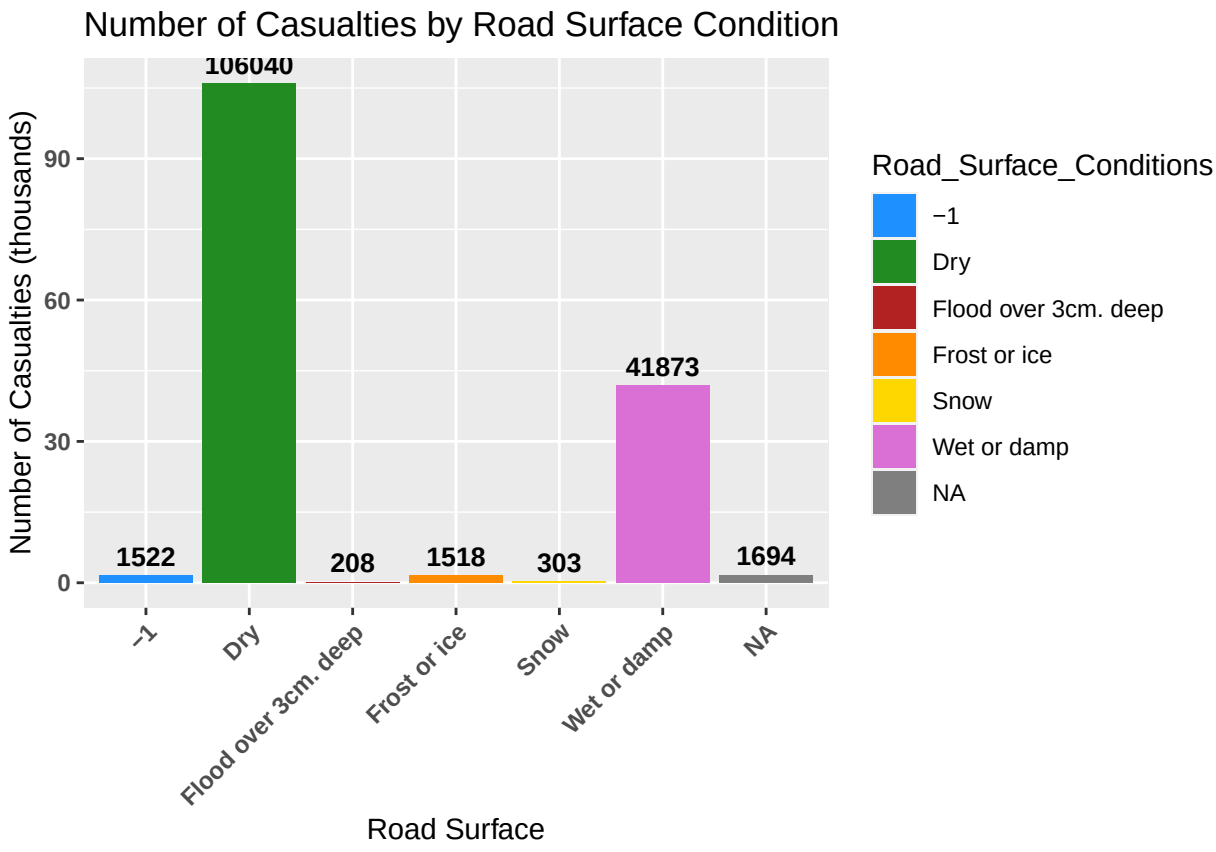



```
# Find column name containing a specific keyword
keyword <- "surface"
matching_columns <- grep(keyword, names(AccidentsList), value = TRUE, ignore.case = TRUE)
print(matching_columns)
```

```
## [1] "Road_Surface_Conditions"
```

```
# Load necessary packages
library(ggplot2)
library(scales)
library(dplyr)

# First, create a named vector to map road surface condition codes to their names
road_surface_names <- c(
  "1" = "Dry",
  "2" = "Wet or damp",
  "3" = "Snow",
  "4" = "Frost or ice",
  "5" = "Flood over 3cm. deep",
  "6" = "Oil or diesel",
  "7" = "Mud"
)

# Next, use dplyr to mutate the Road_Surface_Conditions column
Accidents_updated <- Accidents %>%
  mutate(Road_Surface_Conditions = as.character(Road_Surface_Conditions)) %>%
```

```r
  mutate(Road_Surface_Conditions = replace(Road_Surface_Conditions, Road_Surface_Conditions %in% names(

# Aggregate data by Road_Surface_Conditions
Accidents_agg <- Accidents_updated %>%
  group_by(Road_Surface_Conditions) %>%
  summarise(Number_of_Casualties = sum(Number_of_Casualties, na.rm = TRUE)) %>%
  ungroup()

# Now, create the plot with the updated data
ggplot(Accidents_agg, aes(x = Road_Surface_Conditions, y = Number_of_Casualties, fill = Road_Surface_Co
  geom_col() +
  scale_fill_manual(values = c("dodgerblue", "forestgreen", "firebrick", "darkorange", "gold", "orchid"
  labs(x = "Road Surface", y = "Number of Casualties (thousands)", title = "Number of Casualties by Roa
  theme(axis.text.x = element_text(angle = 45, hjust = 1, face = "bold"), axis.text.y = element_text(fa
  geom_text(aes(label = Number_of_Casualties), vjust = -0.5, size = 3.5, fontface = "bold") +
  scale_y_continuous(labels = scales::comma_format(scale = 1e-3))
```




```r
names(AccidentsList)
```

```
##  [1] "Code"                      "Place Name"
##  [3] "All ages"                  "Accident_Index"
##  [5] "Police_Force"              "Accident_Severity"
##  [7] "Number_of_Vehicles"        "Number_of_Casualties"
##  [9] "Date"                      "Day_of_Week"
## [11] "Local_Authority_(Highway)" "Road_Type"
## [13] "Light_Conditions"          "Weather_Conditions"
```

```
## [15] "Road_Surface_Conditions"    "Special_Conditions_at_Site"
## [17] "Carriageway_Hazards"
```

```
# Check column names
colnames(AccidentsList)
```

```
##  [1] "Code"                        "Place Name"
##  [3] "All ages"                    "Accident_Index"
##  [5] "Police_Force"                "Accident_Severity"
##  [7] "Number_of_Vehicles"          "Number_of_Casualties"
##  [9] "Date"                        "Day_of_Week"
## [11] "Local_Authority_(Highway)"   "Road_Type"
## [13] "Light_Conditions"            "Weather_Conditions"
## [15] "Road_Surface_Conditions"     "Special_Conditions_at_Site"
## [17] "Carriageway_Hazards"
```

```
# Rename the column
library(dplyr)
AccidentsList <- AccidentsList %>% rename(specialconditions = Special_Conditions_at_Site)

# Check if the column name has been updated
colnames(AccidentsList)
```

```
##  [1] "Code"                        "Place Name"
##  [3] "All ages"                    "Accident_Index"
##  [5] "Police_Force"                "Accident_Severity"
##  [7] "Number_of_Vehicles"          "Number_of_Casualties"
##  [9] "Date"                        "Day_of_Week"
## [11] "Local_Authority_(Highway)"   "Road_Type"
## [13] "Light_Conditions"            "Weather_Conditions"
## [15] "Road_Surface_Conditions"     "specialconditions"
## [17] "Carriageway_Hazards"
```

```
# Print the column names of the AccidentConditionsV2 data frame
names(AccidentConditionsV2)
```

```
##  [1] "Accident_Index"              "Police_Force"
##  [3] "Accident_Severity"           "Number_of_Vehicles"
##  [5] "Number_of_Casualties"        "Date"
##  [7] "Day_of_Week"                 "Local_Authority_(Highway)"
##  [9] "Road_Type"                   "Light_Conditions"
## [11] "Weather_Conditions"          "Road_Surface_Conditions"
## [13] "Special_Conditions_at_Site"  "Carriageway_Hazards"
## [15] "Code"
```

```
# Print the column names of the original data frames
names(Accidents)
```

```
##  [1] "Accident_Index"
##  [2] "Location_Easting_OSGR"
##  [3] "Location_Northing_OSGR"
##  [4] "Longitude"
```

```
##  [5] "Latitude"
##  [6] "Police_Force"
##  [7] "Accident_Severity"
##  [8] "Number_of_Vehicles"
##  [9] "Number_of_Casualties"
## [10] "Date"
## [11] "Day_of_Week"
## [12] "Time"
## [13] "Local_Authority_(District)"
## [14] "Local_Authority_(Highway)"
## [15] "1st_Road_Class"
## [16] "1st_Road_Number"
## [17] "Road_Type"
## [18] "Speed_limit"
## [19] "Junction_Detail"
## [20] "Junction_Control"
## [21] "2nd_Road_Class"
## [22] "2nd_Road_Number"
## [23] "Pedestrian_Crossing-Human_Control"
## [24] "Pedestrian_Crossing-Physical_Facilities"
## [25] "Light_Conditions"
## [26] "Weather_Conditions"
## [27] "Road_Surface_Conditions"
## [28] "Special_Conditions_at_Site"
## [29] "Carriageway_Hazards"
## [30] "Urban_or_Rural_Area"
## [31] "Did_Police_Officer_Attend_Scene_of_Accident"
## [32] "LSOA_of_Accident_Location"
```

```
names(Casualties)
```

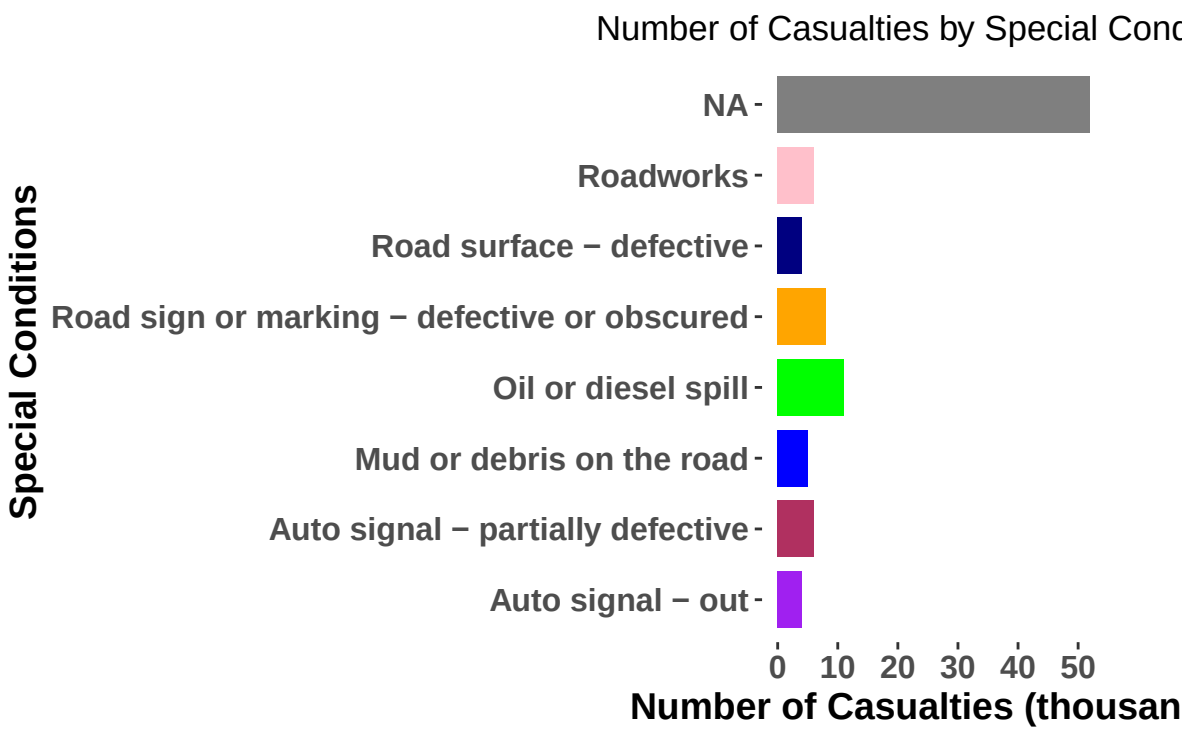
```
##  [1] "Accident_Index"                    "Vehicle_Reference"
##  [3] "Casualty_Reference"                "Casualty_Class"
##  [5] "Sex_of_Casualty"                   "Age_of_Casualty"
##  [7] "Age_Band_of_Casualty"              "Casualty_Severity"
##  [9] "Pedestrian_Location"               "Pedestrian_Movement"
## [11] "Car_Passenger"                     "Bus_or_Coach_Passenger"
## [13] "Pedestrian_Road_Maintenance_Worker" "Casualty_Type"
## [15] "Casualty_Home_Area_Type"           "Casualty_IMD_Decile"
```

```
names(Vehicles)
```

```
##  [1] "Accident_Index"                 "Vehicle_Reference"
##  [3] "Vehicle_Type"                   "Towing_and_Articulation"
##  [5] "Vehicle_Manoeuvre"              "Vehicle_Location-Restricted_Lane"
##  [7] "Junction_Location"              "Skidding_and_Overturning"
##  [9] "Hit_Object_in_Carriageway"      "Vehicle_Leaving_Carriageway"
## [11] "Hit_Object_off_Carriageway"     "1st_Point_of_Impact"
## [13] "Was_Vehicle_Left_Hand_Drive?"   "Journey_Purpose_of_Driver"
## [15] "Sex_of_Driver"                  "Age_of_Driver"
## [17] "Age_Band_of_Driver"             "Engine_Capacity_(CC)"
## [19] "Propulsion_Code"                "Age_of_Vehicle"
## [21] "Driver_IMD_Decile"              "Driver_Home_Area_Type"
## [23] "Vehicle_IMD_Decile"
```

```r
#Load required libraries
library(readr)
library(readxl)
library(dplyr)
library(ggplot2)
library(scales)

#Load the dataset from CSV and Excel files
Casualties <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Casualties 201
Accidents <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Accidents 2019
Vehicles <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data- Vehicles 2019.csv
Population <- read_xlsx('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Population.xlsx')

#Combine the data frames
AccidentConditions <- inner_join(Accidents, Casualties, by = "Accident_Index")
AccidentConditionsV2 <- inner_join(AccidentConditions, Vehicles, by = "Accident_Index")

#Define a named vector to map special condition codes to their names
special_condition_names <- c(
"1" = "Roadworks",
"2" = "Auto signal - out",
"3" = "Auto signal - partially defective",
"4" = "Road sign or marking - defective or obscured",
"5" = "Road surface - defective",
"6" = "Oil or diesel spill",
"7" = "Mud or debris on the road",
"8" = "Temporary traffic lights",
"9" = "Pedestrian crossing - defective",
"10" = "Accident involving animals on the road"
)

#Update the 'Special_Conditions_at_Site' column with the mapped names
AccidentConditionsV2$Special_Conditions_at_Site <- special_condition_names[as.character(AccidentConditi

#Create a vector of unique colors
special_condition_colors <- c("purple", "maroon", "blue", "green", "orange", "navy blue", "pink", "brown

#Create the plot with the updated data and display values on bars
ggplot(AccidentConditionsV2, aes(y = Special_Conditions_at_Site, x = Number_of_Casualties, fill = Specia
geom_col(width = 0.8, color = "black", position = "dodge", linetype = "blank") +
scale_fill_manual(values = special_condition_colors) +
labs(x = "Number of Casualties (thousands)", y = "Special Conditions",
title = "Number of Casualties by Special Conditions") +
theme(axis.text.x = element_text(face = "bold", size = 12),
axis.text.y = element_text(face = "bold", size = 12),
axis.title.y = element_text(face = "bold", size = 14),
axis.title.x = element_text(face = "bold", size = 14),
plot.margin = unit(c(1, 1, 1, 1), "cm"),
panel.grid.major = element_blank(),
panel.grid.minor = element_blank(),
panel.border = element_blank(),
panel.background = element_blank(),
legend.position = "none",
```

```
plot.title = element_text(hjust = 0.5))
```




```
unique(AccidentsList$Road_Type)
```

```
## [1] 6 3 1 7 9 2
```

```
library(janitor)
class(AccidentConditions$Road_Type)
```

```
## [1] "numeric"
```

```
summary(AccidentConditions$Road_Type)
```

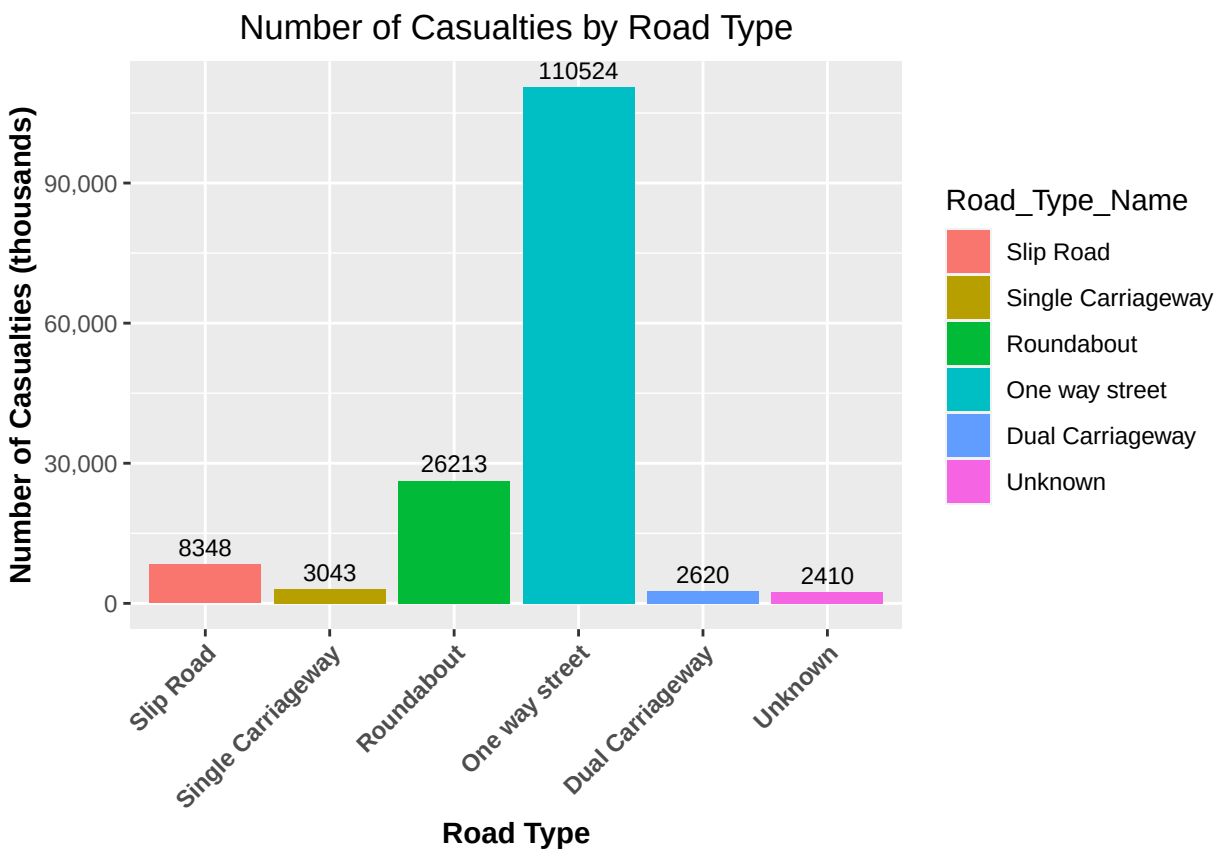
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   6.000   6.000   5.199   6.000   9.000
```

```
tabyl(AccidentConditions$Road_Type) %>%
  arrange(desc(percent))
```

```
##  AccidentConditions$Road_Type      n    percent
##                             6 110533 0.72163609
##                             3  26216 0.17115623
##                             1   8348 0.05450153
##                             2   3043 0.01986681
##                             7   2620 0.01710518
##                             9   2410 0.01573415
```

```r
library(dplyr)
```

```r
library(ggplot2)
library(readr)
library(dplyr)

# Load the data
Accidents <- read_csv('/Users/mdabusufian/Downloads/FDS_CLASS_PROJECT/Road Safety Data - Accidents 2019

# Aggregate the data
AccidentsList <- Accidents %>%
  group_by(Road_Type) %>%
  summarise(Number_of_Casualties = sum(Number_of_Casualties, na.rm = TRUE))

# Add road type names
AccidentsList$Road_Type_Name <- factor(AccidentsList$Road_Type,
                                       levels = unique(AccidentsList$Road_Type))

# Update levels of Road_Type_Name with desired road type names
levels(AccidentsList$Road_Type_Name) <- c("Slip Road", "Single Carriageway", "Roundabout", "One way str

# Plot the data
ggplot(AccidentsList, aes(x = Road_Type_Name, y = Number_of_Casualties, fill = Road_Type_Name)) +
  geom_col() +
  geom_text(aes(label = Number_of_Casualties), vjust = -0.5, size = 3) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, face = "bold"),
        axis.title.x = element_text(face = "bold"),
        axis.title.y = element_text(face = "bold"),
        plot.title = element_text(hjust = 0.5)) +
  labs(x = "Road Type", y = "Number of Casualties (thousands)", title = "Number of Casualties by Road Ty
  scale_y_continuous(labels = scales::comma)
```




```r
# Load necessary packages
library(ggplot2)
library(scales)
library(dplyr)

# Filter out unexpected values in Light_Conditions
Accidents_filtered <- Accidents %>%
  filter(Light_Conditions %in% 1:5)

# Aggregate data by Light_Conditions
Accidents_agg <- Accidents_filtered %>%
  group_by(Light_Conditions) %>%
  summarise(Total_Casualties = sum(Number_of_Casualties, na.rm = TRUE)) %>%
  ungroup()

# Create a named vector for Light Conditions categories
light_conditions_labels <- c("Daylight", "Darkness - lights lit", "Darkness - lights unlit",
                             "Darkness - no lighting", "Darkness - lighting unknown")

names(light_conditions_labels) <- 1:5

# Convert Light_Conditions to factor
Accidents_agg$Light_Conditions <- factor(Accidents_agg$Light_Conditions, levels = 1:5, labels = light_c

# Create the ggplot2 bar chart
ggplot(Accidents_agg, aes(x = Light_Conditions, y = Total_Casualties)) +
  geom_col(fill = "steelblue", width = 0.7) +
```

```
  geom_text(aes(label = format(Total_Casualties, big.mark = ",")), vjust = -0.5, size = 3, fontface = "
  labs(x = "Light Conditions", y = "Number of Casualties (thousands)", title = "Number of Casualties by
  theme(axis.text.x = element_text(angle = 45, hjust = 1, face = "bold"),
        axis.text.y = element_text(face = "bold"),
        axis.title.y = element_text(face = "bold"),
        axis.title.x = element_text(face = "bold"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"),
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(labels = scales::comma_format(scale = 1e-3), limits = c(0, max(Accidents_agg$Total_
```