NYC Vehicle Accident Investigation - Person

Team41 - Xufei Lang

2023-07-17

Data

Data Source

The Motor Vehicle Collisions Person table contains details for people involved in the crash. Each row represents a person (driver, occupant, pedestrian, bicyclist,..) involved in a crash. Data goes back to Jan 2013, and downloaded on 6/15/2023. https://catalog.data.gov/dataset/motor-vehicle-collisions-person. Raw data has 5056441 rows, 21 columns, 840.5 Mb.

Note: data updated regularly on DATA.GOV website, so if use the newer version of data, results might be a little different from the report.

Data Cleaning

Load relevant packages.

```r
# clean environment.
rm(list=ls())

# install packages.
if (!require("dplyr")) install.packages("dplyr")
if (!require("tidyr")) install.packages("tidyr")
if (!(require("ggplot2"))) install.packages("ggplot2")
if (!(require("stringr"))) install.packages("stringr")
if (!(require("caret"))) install.packages("caret")
if (!(require("car"))) install.packages("car")
if (!(require("ROCR"))) install.packages("ROCR")
library(dplyr)
library(tidyr)
library(ggplot2)
library(stringr)
library(caret)
library(car)
library(ROCR)
```

Warning: the following 3 code chunks should only run once for generating/exporting cleaned dataset to a csv file. Remember to set working directory, loaded data path/name, exported file path/name. This process might take a few minutes considering the data size, please be patient. Thanks!

```r
# set working directory.
setwd("/Users/mingzeli0924/Documents/STUDY/OMSA/Courses/MGT6203 - Data Analytics in Business/Group Proj

# load in person raw dataset. updated 6/15/2023.
df <- read.csv("20230615 downloaded raw data/Motor_Vehicle_Collisions_-_Person (2).csv")
head(df)
```

```
##   UNIQUE_ID COLLISION_ID CRASH_DATE CRASH_TIME
## 1  10249006      4229554 10/26/2019       9:43
## 2  10255054      4230587 10/25/2019      15:15
## 3  10253177      4230550 10/26/2019      17:55
## 4   6650180      3565527 11/21/2016      13:05
## 5  10255516      4231168 10/25/2019      11:16
## 6  10253606      4230743 10/24/2019      19:15
##                              PERSON_ID PERSON_TYPE PERSON_INJURY VEHICLE_ID
## 1 31aa2bc0-f545-444f-8cdb-f1cb5cf00b89    Occupant   Unspecified   19141108
## 2 4629e500-a73e-48dc-b8fb-53124d124b80    Occupant   Unspecified   19144075
## 3 ae48c136-1383-45db-83f4-2a5eecfb7cff    Occupant   Unspecified   19143133
## 4                              2782525    Occupant   Unspecified         NA
## 5 e038e18f-40fb-4471-99cf-345eae36e064    Occupant   Unspecified   19144329
## 6 84bcb3a7-d201-4c61-9e30-fe29268c1074    Occupant       Injured   19143343
##   PERSON_AGE    EJECTION EMOTIONAL_STATUS  BODILY_INJURY
## 1         NA
## 2         33 Not Ejected   Does Not Apply Does Not Apply
## 3         55
## 4         NA
## 5          7 Not Ejected   Does Not Apply Does Not Apply
## 6         27 Not Ejected        Conscious           Back
##                                                                     POSITION_IN_VEHICLE
## 1
## 2 Front passenger, if two or more persons, including the driver, are in the front seat
## 3
## 4
## 5                                 Right rear passenger or motorcycle sidecar passenger
## 6                                                                                Driver
##     SAFETY_EQUIPMENT PED_LOCATION PED_ACTION                   COMPLAINT
## 1
## 2 Lap Belt & Harness                                      Does Not Apply
## 3
## 4
## 5           Lap Belt                                      Does Not Apply
## 6 Lap Belt & Harness                         Complaint of Pain or Nausea
##          PED_ROLE CONTRIBUTING_FACTOR_1 CONTRIBUTING_FACTOR_2 PERSON_SEX
## 1      Registrant                                                      U
## 2       Passenger                                                      F
## 3      Registrant                                                      M
## 4 Notified Person
## 5       Passenger                                                      F
## 6          Driver                                                      M
```

```r
glimpse(df)
```

```
## Rows: 5,059,446
## Columns: 21
## $ UNIQUE_ID             <int> 10249006, 10255054, 10253177, 6650180, 10255516,~
## $ COLLISION_ID          <int> 4229554, 4230587, 4230550, 3565527, 4231168, 423~
## $ CRASH_DATE            <chr> "10/26/2019", "10/25/2019", "10/26/2019", "11/21~
## $ CRASH_TIME            <chr> "9:43", "15:15", "17:55", "13:05", "11:16", "19:~
## $ PERSON_ID             <chr> "31aa2bc0-f545-444f-8cdb-f1cb5cf00b89", "4629e50~
## $ PERSON_TYPE           <chr> "Occupant", "Occupant", "Occupant", "Occupant", ~
## $ PERSON_INJURY         <chr> "Unspecified", "Unspecified", "Unspecified", "Un~
## $ VEHICLE_ID            <int> 19141108, 19144075, 19143133, NA, 19144329, 1914~
## $ PERSON_AGE            <int> NA, 33, 55, NA, 7, 27, 41, 24, 36, NA, 30, 52, N~
## $ EJECTION              <chr> "", "Not Ejected", "", "", "Not Ejected", "Not E~
## $ EMOTIONAL_STATUS      <chr> "", "Does Not Apply", "", "", "Does Not Apply", ~
## $ BODILY_INJURY         <chr> "", "Does Not Apply", "", "", "Does Not Apply", ~
## $ POSITION_IN_VEHICLE   <chr> "", "Front passenger, if two or more persons, in~
## $ SAFETY_EQUIPMENT      <chr> "", "Lap Belt & Harness", "", "", "Lap Belt", "L~
## $ PED_LOCATION          <chr> "", "", "", "", "", "", "", "Pedestrian/Bicyclis~
## $ PED_ACTION            <chr> "", "", "", "", "", "", "", "Crossing With Signa~
## $ COMPLAINT             <chr> "", "Does Not Apply", "", "", "Does Not Apply", ~
## $ PED_ROLE              <chr> "Registrant", "Passenger", "Registrant", "Notifi~
## $ CONTRIBUTING_FACTOR_1 <chr> "", "", "", "", "", "", "", "Unspecified", "", "~
## $ CONTRIBUTING_FACTOR_2 <chr> "", "", "", "", "", "", "", "Unspecified", "", "~
## $ PERSON_SEX            <chr> "U", "F", "M", "", "F", "M", "F", "F", "M", "U",~
```

Clean dataset, remove unnecessary rows & columns.

```
# remove rows with PED_ROL in "Registrant", "Notified Person", "Witness", "Policy Holder", "Owner", tho
person <- df %>%
  filter(!(PED_ROLE %in% c("Registrant", "Notified Person", "Witness", "Policy Holder", "Owner")))

# check the percentage of missing values in each column.
person_check <- person %>%
  summarize(across(everything(), ~ sum(. == ""))) %>%
  pivot_longer(everything(), names_to = "Column", values_to = "Count") %>%
  mutate("Percentage(%)" = round(Count / nrow(person) * 100, 2)) %>%
  arrange(desc(Count))
person_check
```

```
## # A tibble: 21 x 3
##    Column                  Count ‘Percentage(%)‘
##    <chr>                   <int>           <dbl>
##  1 CONTRIBUTING_FACTOR_2 2783687           97.3
##  2 CONTRIBUTING_FACTOR_1 2783585           97.3
##  3 PED_ACTION            2782391           97.3
##  4 PED_LOCATION          2782290           97.3
##  5 EJECTION               268323            9.38
##  6 SAFETY_EQUIPMENT       268030            9.37
##  7 POSITION_IN_VEHICLE    267957            9.37
##  8 PERSON_SEX             195284            6.83
##  9 EMOTIONAL_STATUS       195214            6.83
## 10 BODILY_INJURY          195171            6.82
## # i 11 more rows
```

```
# use a threshold of 50% to remove columns.
person <- person %>%
  select(!c("CONTRIBUTING_FACTOR_2", "CONTRIBUTING_FACTOR_1", "PED_ACTION", "PED_LOCATION"))
glimpse(person)  # this is the cleaned dataset.
```

```
## Rows: 2,859,875
## Columns: 17
## $ UNIQUE_ID           <int> 10255054, 10255516, 10253606, 10248708, 10250179, ~
## $ COLLISION_ID        <int> 4230587, 4231168, 4230743, 4229547, 4229808, 42307~
## $ CRASH_DATE          <chr> "10/25/2019", "10/25/2019", "10/24/2019", "10/26/2~
## $ CRASH_TIME          <chr> "15:15", "11:16", "19:15", "1:15", "13:04", "0:41"~
## $ PERSON_ID           <chr> "4629e500-a73e-48dc-b8fb-53124d124b80", "e038e18f-~
## $ PERSON_TYPE         <chr> "Occupant", "Occupant", "Occupant", "Pedestrian", ~
## $ PERSON_INJURY       <chr> "Unspecified", "Unspecified", "Injured", "Injured"~
## $ VEHICLE_ID          <int> 19144075, 19144329, 19143343, NA, 19141630, 191433~
## $ PERSON_AGE          <int> 33, 7, 27, 24, 36, 30, 52, 42, 55, 30, 59, 37, 36,~
## $ EJECTION            <chr> "Not Ejected", "Not Ejected", "Not Ejected", "", "~
## $ EMOTIONAL_STATUS    <chr> "Does Not Apply", "Does Not Apply", "Conscious", "~
## $ BODILY_INJURY       <chr> "Does Not Apply", "Does Not Apply", "Back", "Shoul~
## $ POSITION_IN_VEHICLE <chr> "Front passenger, if two or more persons, includin~
## $ SAFETY_EQUIPMENT    <chr> "Lap Belt & Harness", "Lap Belt", "Lap Belt & Harn~
## $ COMPLAINT           <chr> "Does Not Apply", "Does Not Apply", "Complaint of ~
## $ PED_ROLE            <chr> "Passenger", "Passenger", "Driver", "Pedestrian", ~
## $ PERSON_SEX          <chr> "F", "F", "M", "F", "M", "M", "F", "M", "M", "F", ~
```

Set exported file path/name.

```
# export to a csv file.
setwd("/Users/mingzeli0924/Documents/STUDY/OMSA/Courses/MGT6203 - Data Analytics in Business/Group Proj
write.csv(person, file="Motor_Vehicle_Collisions - Person_clean.csv")
```

Warning: the above 3 code chunks should only run once for generating/exporting cleaned dataset to a csv file. Load in csv raw data several times could crash the system. For further analysis, either use the person dataframe generated from above code, or load in Person_clean.csv file using the next code chunk.

Data Exploring

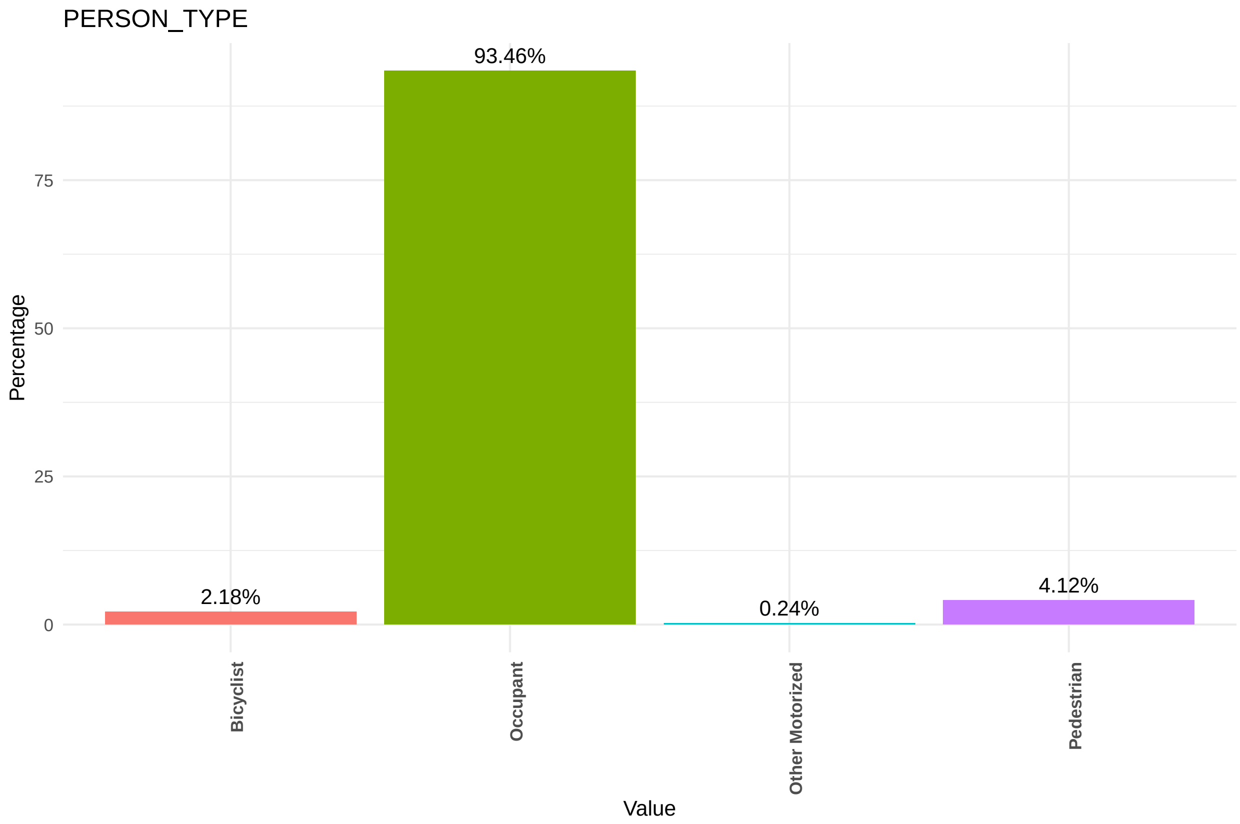
Load in Person_clean.csv file only when something goes wrong in further analysis and reload data is necessary. If so, skip above cleaning process and start from loading in cleaned dataset. Otherwise, skip this part.

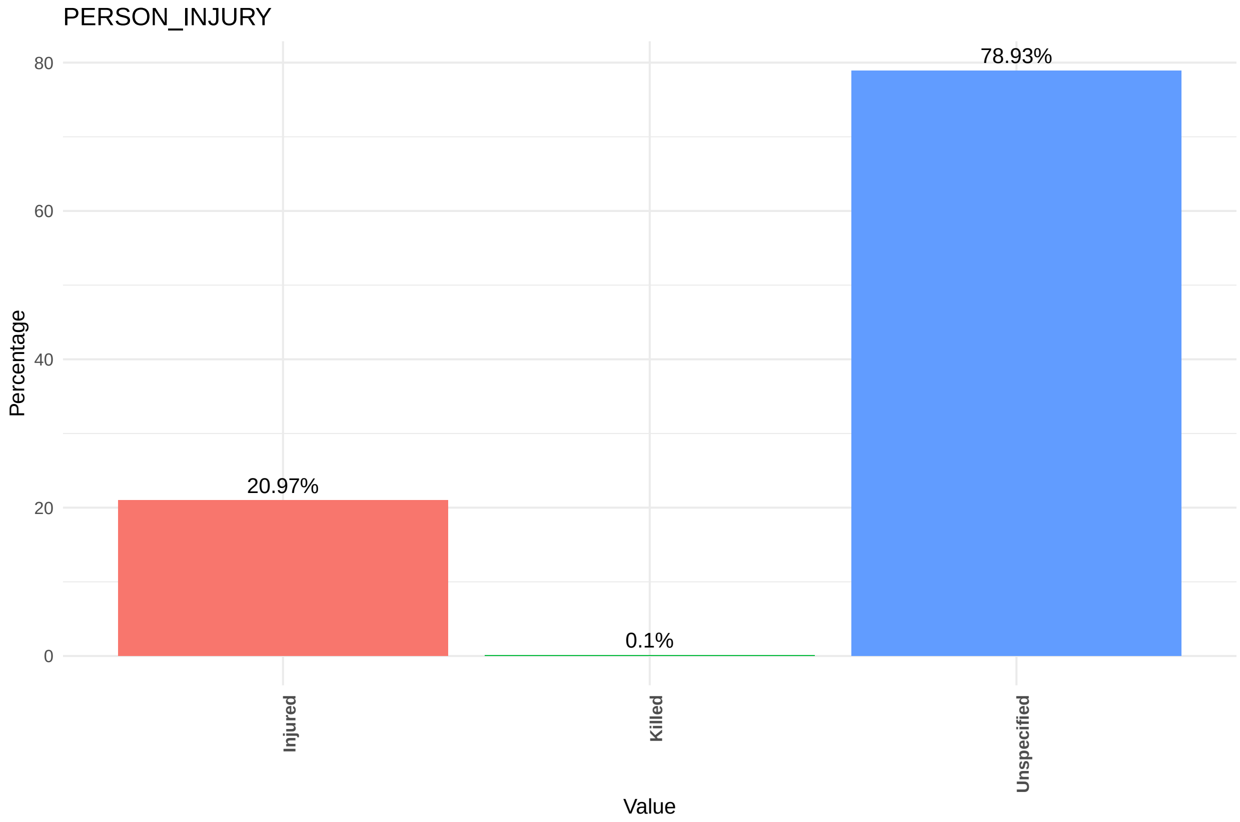
```
# load cleaned person data.
setwd("/Users/mingzeli0924/Documents/STUDY/OMSA/Courses/MGT6203 - Data Analytics in Business/Group Proj
person <- read.csv("Motor_Vehicle_Collisions - Person_clean.csv")
person <- person[, -1]
```

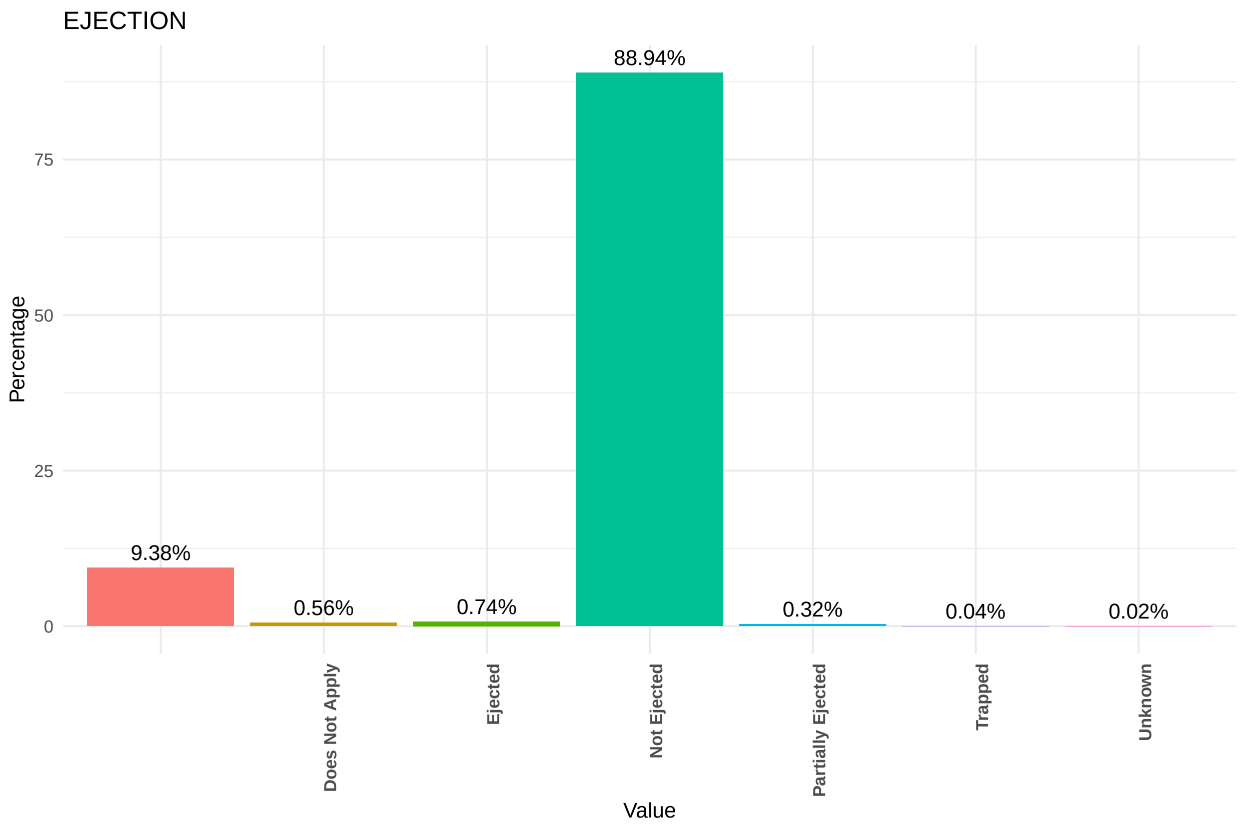
Histograms for the following categorical columns: "PERSON_TYPE" "PERSON_INJURY" "EJECTION" "EMOTIONAL_STATUS" "BODILY_INJURY" "POSITION_IN_VEHICLE" "SAFETY_EQUIPMENT" "COMPLAINT" "PED_ROLE" "PERSON_SEX"

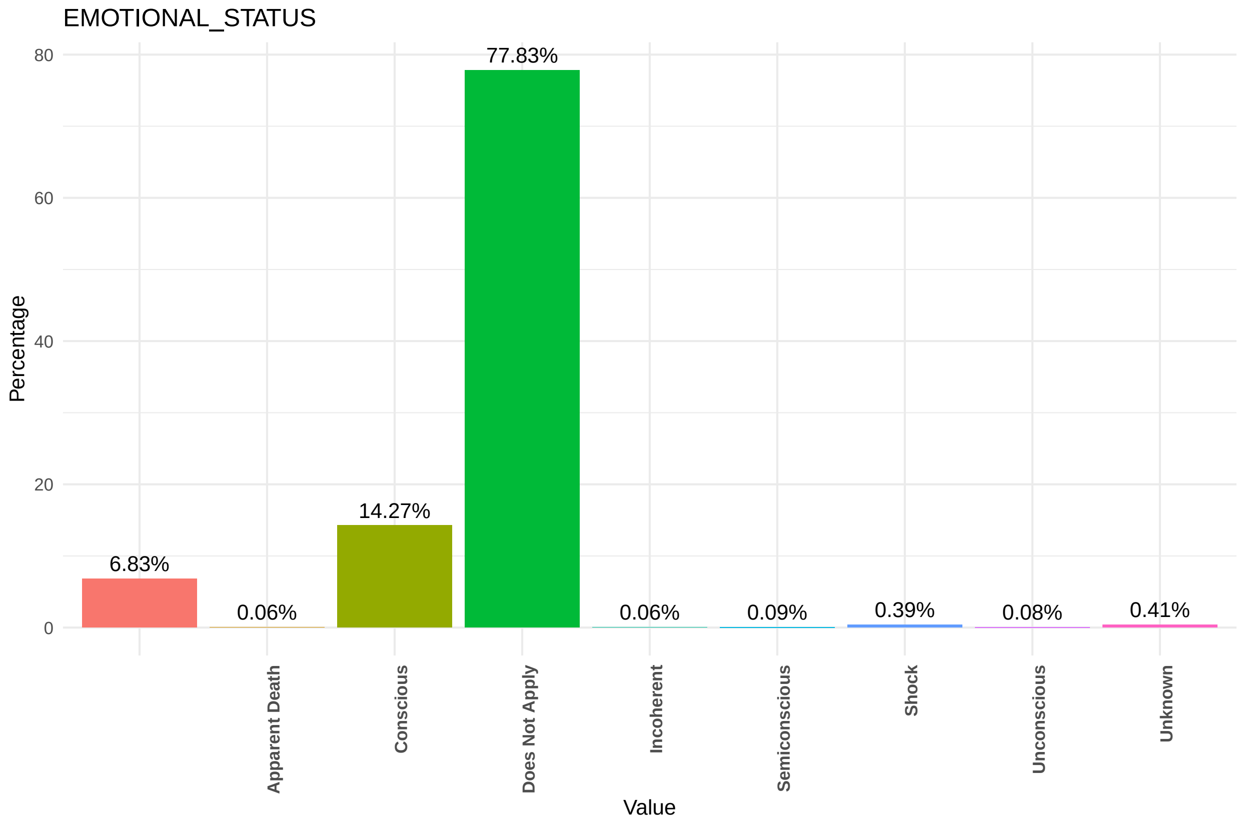
```
# create a plot function to show histogram for column.
plot_fun <- function(df,x) {
  # create dataframe for the column with unique value counts.
  title <- x
  x <- df[,x]
  x_count <- df %>%
    mutate(x = as.factor(x)) %>%
    #filter(!(x %in% c("Does Not Apply", "", "-", "Unknown", "U"))) %>%
    count(x) %>%
    mutate(Percentage = round(n/sum(n) * 100, 2))

  # bar plot.
  ggplot(x_count, aes(x = str_wrap(x, width=40), y = Percentage, fill = x)) +
    geom_bar(stat = "identity") +
    labs(x = "Value", y = "Percentage", fill = "Value") +
    scale_fill_discrete(name = "Value") +
    theme_minimal() +
    scale_x_discrete(labels = function(x) str_wrap(x, width = 40)) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, face="bold"), legend.position = "none") +
    geom_text(aes(label=paste0(Percentage, "%"), vjust = -0.5)) +
    labs(title=title)
}
```

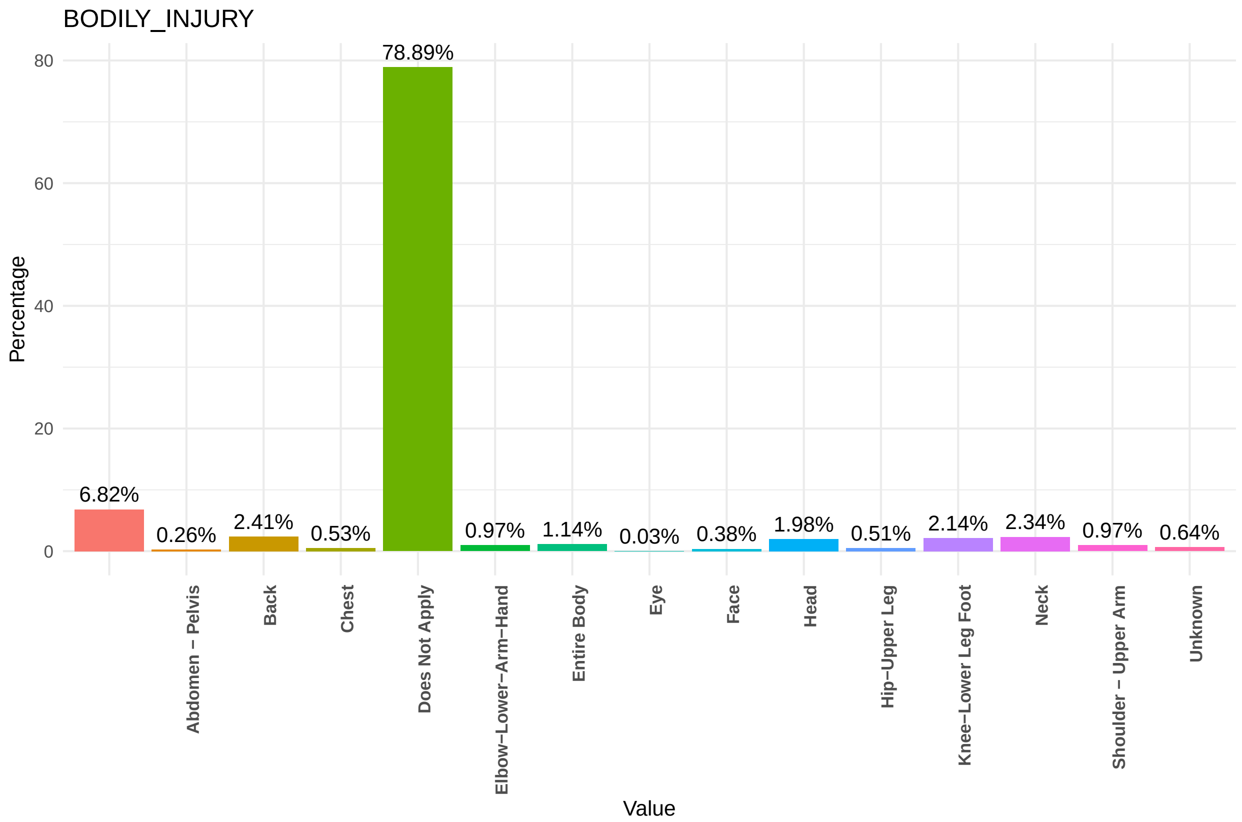
```
his_cols <- c("PERSON_TYPE","PERSON_INJURY","EJECTION","EMOTIONAL_STATUS","BODILY_INJURY","POSITION_IN_
for (x in his_cols) {
  plot(plot_fun(person, x))
}
```

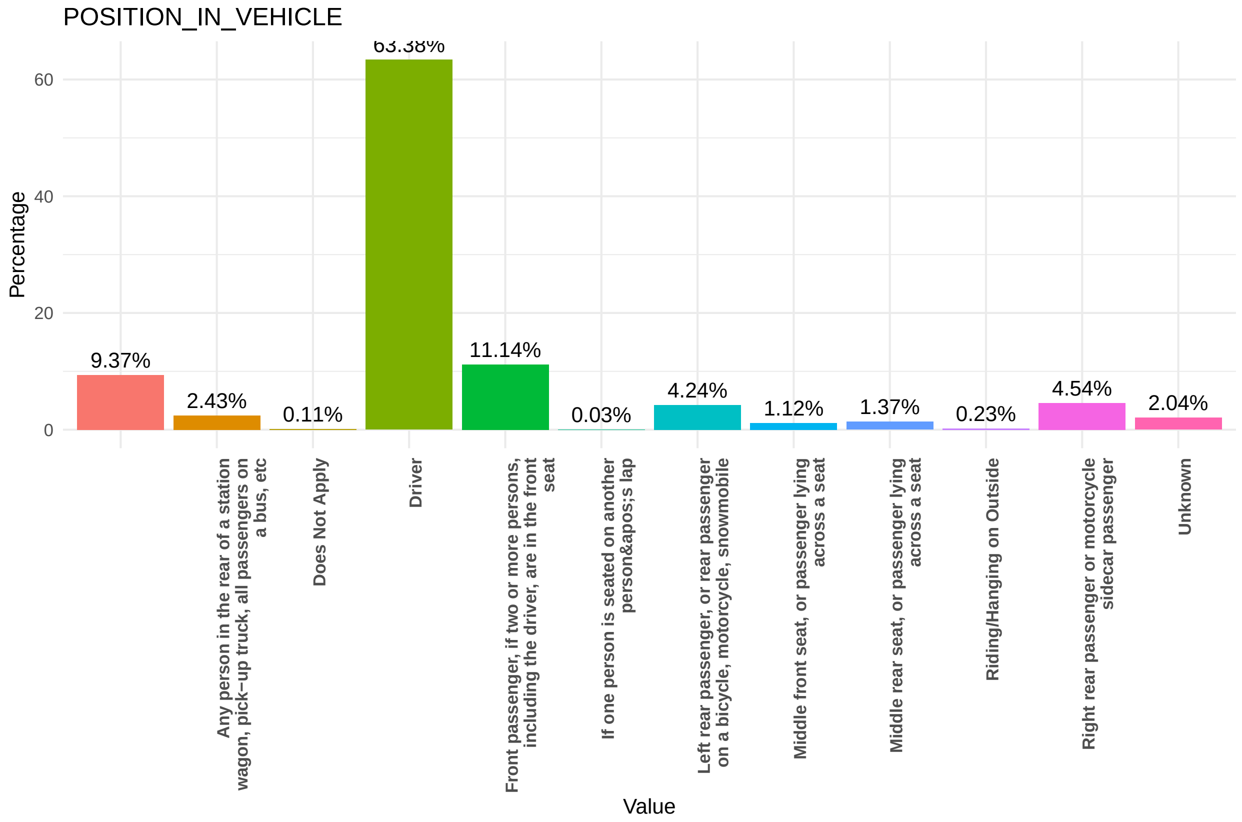

PERSON_TYPE
Percentage
Value
0
25
50
75
2.18%
93.46%
0.24%
4.12%
Bicyclist
Occupant
Other Motorized
Pedestrian

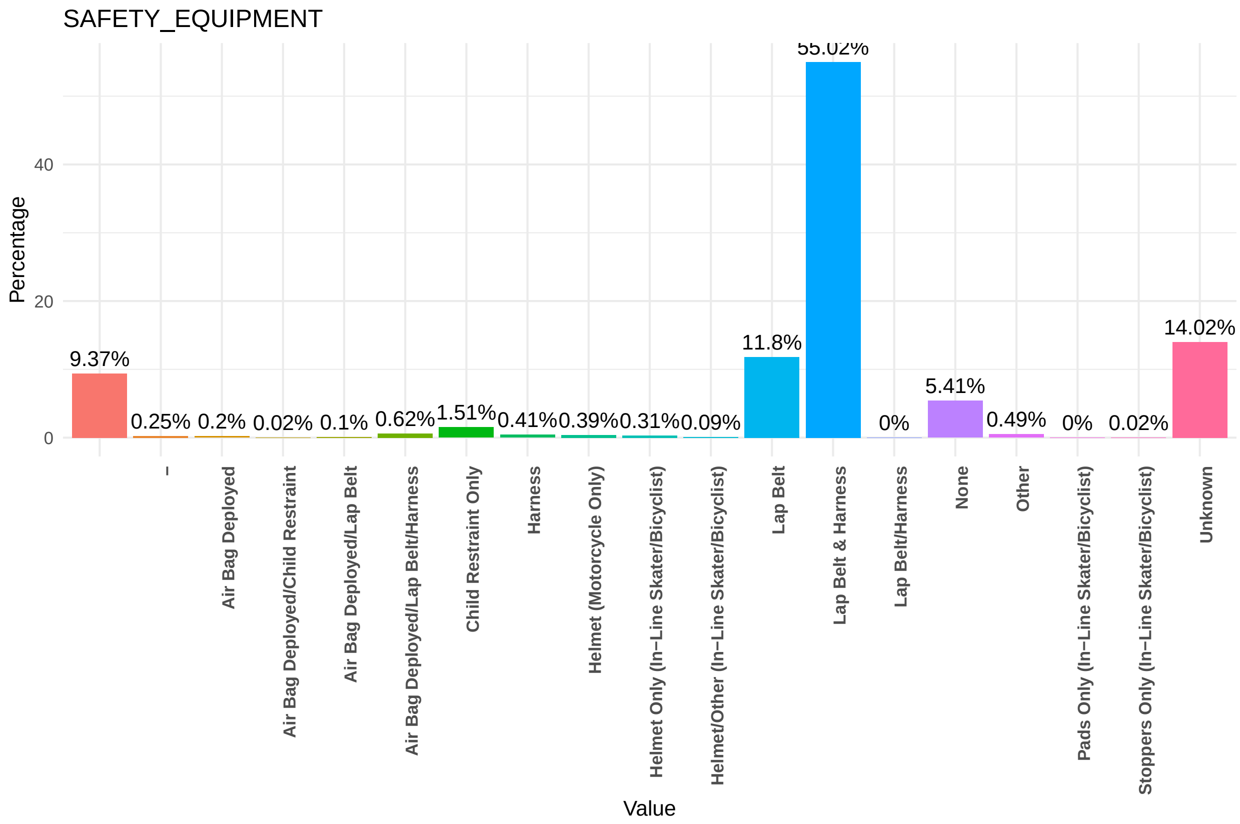

PERSON_INJURY
Percentage
Value
0
20
40
60
80
20.97%
0.1%
78.93%
Injured
Killed
Unspecified

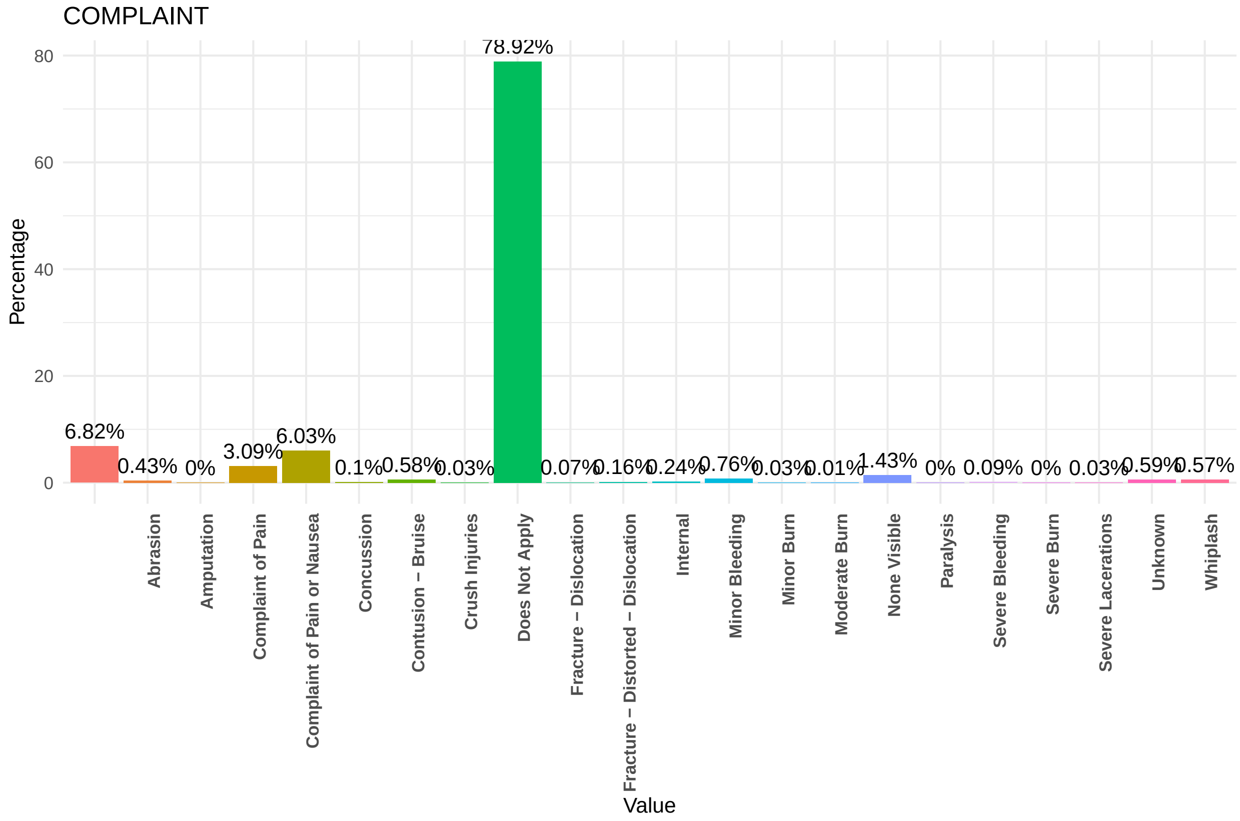

EJECTION
Percentage
0
25
50
75
9.38%
0.56%
0.74%
88.94%
0.32%
0.04%
0.02%
Does Not Apply
Ejected
Not Ejected
Partially Ejected
Trapped
Unknown
Value

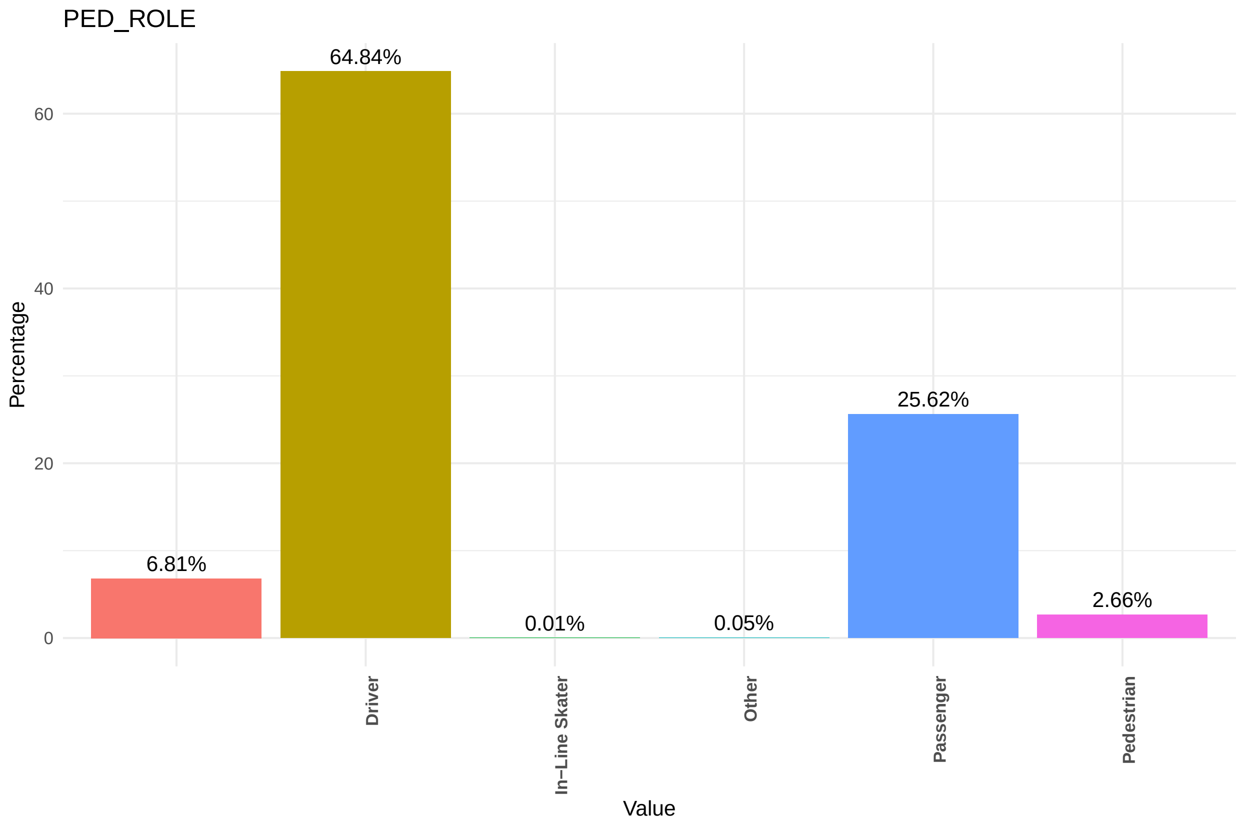

EMOTIONAL_STATUS
Percentage
0
20
40
60
80
6.83%
0.06%
14.27%
77.83%
0.06%
0.09%
0.39%
0.08%
0.41%
Apparent Death
Conscious
Does Not Apply
Incoherent
Semiconscious
Shock
Unconscious
Unknown
Value

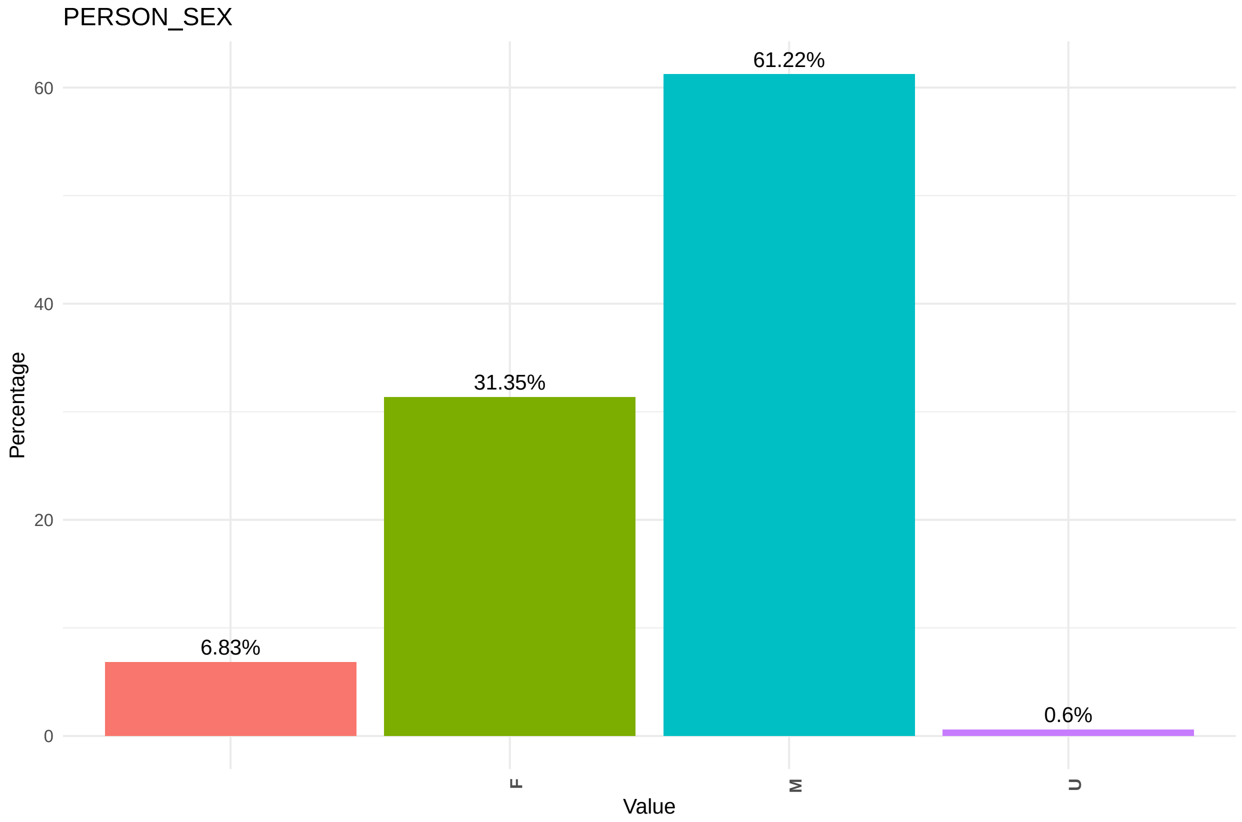

BODILY_INJURY


POSITION_IN_VEHICLE


SAFETY_EQUIPMENT


COMPLAINT





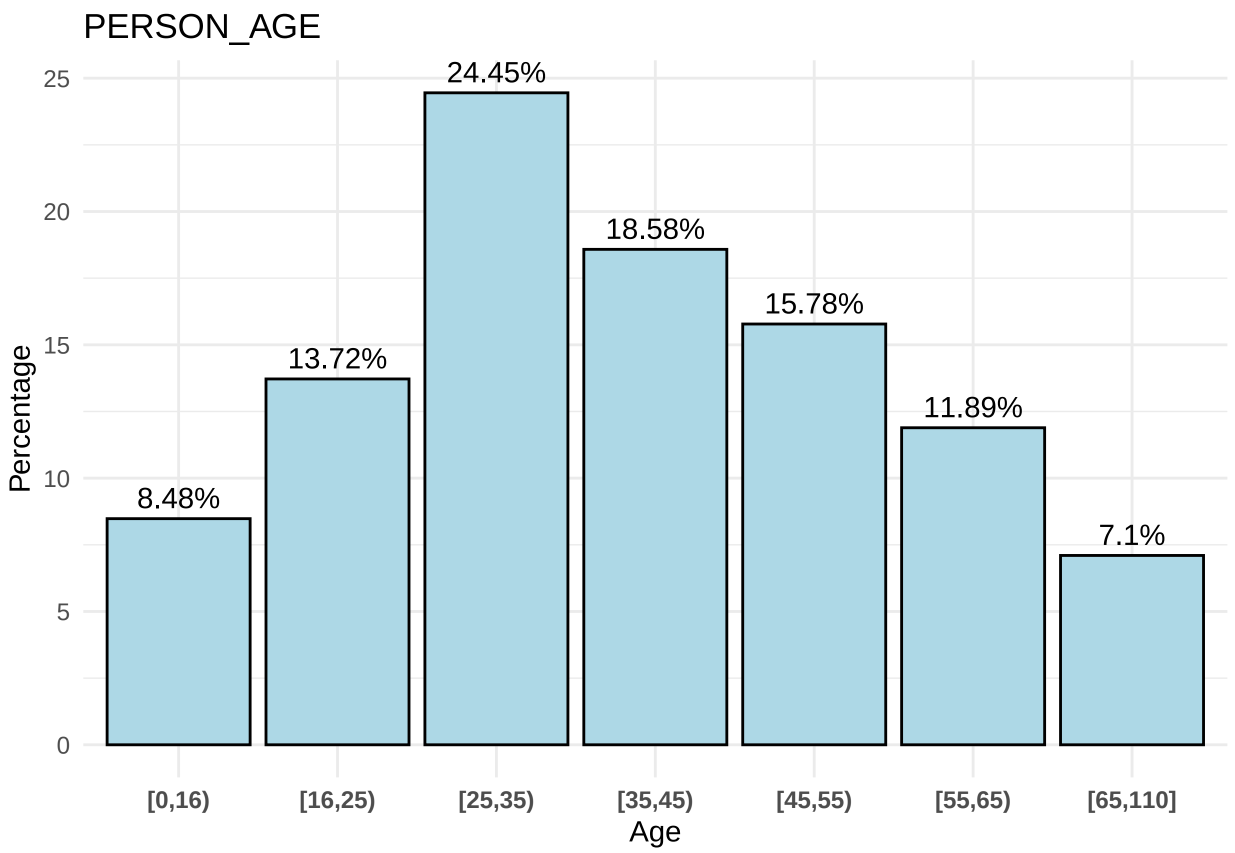
Findings regarding these histograms:

* Missing/Unknown/Unspecified/DoesNotApply values that are still existing in some columns, which could cause bias if removed without further analysis and consideration.
* About some columns:
- PERSON_TYPE: Occupant includes Driver, Passenger, Other, blank values, In-Line Skater.
- PERSON_INJURY: 79% data are Unspecified, could be uninjured person.
- POSITION_IN_VEHICLE: Driver is much more involved in accidents; front passenger is more involved than left rear or right rear passengers.
- SAFETY_EQUIPMENT: has several similar values, for example, "Lap Belt & Harness", "Lap Belt", "Harness", "Lap Belt/Harness", that could be integrated for future easier modeling.
- PED_ROLE: drivers and passengers are more involved in accidents.
- PERSON_SEX: Male has almost twice the probability of involved in accidents as female.

Discussion:
* There might also be a correlation between PED_ROLE, PERSON_TYPE and POSITION_IN_VEHICLE. Check multicollinearity if used as independent variables for future modeling.
* Since in BODILY_INJURY, EMOTIONAL_STATUS & COMPLAINT, top 2 values are "Does Not Apply" & blank; in EJECTION, top 2 values are "Not Ejected" & blank; most of other values are close to 0. Also, these are the variables that happened after the accidents, so they won't be used as independent variables in future modeling.
* Consider using SAFETY_EQUIPMENT, PERSON_SEX and one of PERSON_TYPE, POSITION_IN_VEHICLE & PED_ROLE as independent variables, PERSON_INJURY as dependent variable for logistic regression modeling.

Histogram for PERSON_AGE. Age ranges from -999 to 9999, had to filter to 0-110.
Reasons about certain bound selection:
* Individuals can apply for a learner's permit at the age of 16 in New York.
* SSA defines individuals aged 65 years and above as eligible for full retirement benefits.
* 109-year-old Layne Hall was the oldest driver in US.

```
bin_ranges <- c(0, 16, 25, 35, 45, 55, 65, 110)
age <- person %>%
  select(PERSON_AGE) %>%
  filter(PERSON_AGE >= 0 & PERSON_AGE < 110) %>%
  mutate(bin = cut(PERSON_AGE, breaks=bin_ranges, include.lowest=TRUE, right=FALSE)) %>%
  group_by(bin) %>%
  summarise(count = n()) %>%
  mutate(percentage = round(count / sum(count) * 100, 2)) %>%
  na.omit()

ggplot(age, aes(x = bin, y = percentage)) +
  geom_bar(stat = "identity", fill = "lightblue", color = "black") +
  labs(x = "Age", y = "Percentage") +
  ggtitle("PERSON_AGE") +
  theme_minimal() +
  theme(axis.text.x = element_text(face = "bold")) +
  geom_text(aes(label=paste0(percentage, "%"), vjust = -0.5))
```



Findings regarding this histogram:
* Accident Percentage vs. Age group seems like a normal distribution, with [25-35) ranks the highest percentage of age groups involved in vehicle accidents.

Discussion:
* Consider PERSON_AGE as another independent variable for logistic regression modeling.

Modeling

Data Preparing

Made some modification to values in certain columns, and filter data for modeling.

```
# create dataframe for modeling.
model_df <- person %>%
  select(UNIQUE_ID, PERSON_TYPE, PERSON_AGE, POSITION_IN_VEHICLE, SAFETY_EQUIPMENT, PED_ROLE, PERSON_SE
  filter(PERSON_AGE >= 0 & PERSON_AGE < 110) %>%
  filter(PERSON_SEX == "F" | PERSON_SEX == "M") %>%
  #mutate(PERSON_INJURY = case_when(
    #grepl("Injured|Killed", PERSON_INJURY) ~ "Injured/Killed",
    #TRUE ~ PERSON_INJURY)) %>%
  mutate(SAFETY_EQUIPMENT = case_when(
    grepl("Helmet", SAFETY_EQUIPMENT) ~ "Helmet Only",
    grepl("Air Bag Deployed", SAFETY_EQUIPMENT) ~ "Air Bag Deployed",
    grepl("Lap Belt|Harness", SAFETY_EQUIPMENT) ~ "Lap Belt/Harness",
    grepl("Stoppers Only", SAFETY_EQUIPMENT) ~ "Stoppers Only",
```

```r
    grepl("Pads Only", SAFETY_EQUIPMENT) ~ "Pads Only",
    TRUE ~ SAFETY_EQUIPMENT)) %>%
  filter(!(SAFETY_EQUIPMENT %in% c("", "Unknown", "-", "Other"))) %>%
  mutate(POSITION_IN_VEHICLE = case_when(
    grepl("Front passenger", POSITION_IN_VEHICLE) ~ "Front passenger",
    grepl("Right rear passenger", POSITION_IN_VEHICLE) ~ "RightRear/Sidecar passenger",
    grepl("Left rear passenger", POSITION_IN_VEHICLE) ~ "LeftRear/Rear passenger",
    grepl("Middle front", POSITION_IN_VEHICLE) ~ "MiddleFront/Lying passenger",
    grepl("Middle rear", POSITION_IN_VEHICLE) ~ "MiddleRear/Lying passenger",
    grepl("Any person in the rear", POSITION_IN_VEHICLE) ~ "Wagon/Truck/Bus passenger",
    grepl("seated on another", POSITION_IN_VEHICLE) ~ "Lap passenger",
    TRUE ~ POSITION_IN_VEHICLE)) %>%
  filter(!(POSITION_IN_VEHICLE %in% c("", "Unknown"))) %>%
  mutate_at(vars(-UNIQUE_ID, -PERSON_AGE), factor) #%>%
  #mutate(PERSON_INJURY = ifelse(PERSON_INJURY=="Injured/Killed",1,0))
```

Modeling

Create two datasets for modeling, model1 uses all data in model_df, model2 removes PERSON_INJURY == "Unspecified" from model_df.

Model 1: PERSON_INJURY == "Injured" | "Killed" as 1, PERSON_INJURY == "Unspecified" as 0. In this model, assume all Unspecified values are uninjured/unkilled people.

```r
# dataframe for model1.
model_df1 <- model_df %>%
  mutate(PERSON_INJURY = case_when(
    grepl("Injured|Killed", PERSON_INJURY) ~ "Injured/Killed",
    TRUE ~ PERSON_INJURY)) %>%
  mutate(INJURED_KILLED = ifelse(PERSON_INJURY=="Injured/Killed",1,0))
glimpse(model_df1)
```

```
## Rows: 2,113,343
## Columns: 9
## $ UNIQUE_ID           <int> 10255054, 10255516, 10253606, 10250179, 10253588, ~
## $ PERSON_TYPE         <fct> Occupant, Occupant, Occupant, Occupant, Occupant, ~
## $ PERSON_AGE          <int> 33, 7, 27, 36, 30, 52, 55, 30, 59, 37, 36, 67, 81,~
## $ POSITION_IN_VEHICLE <fct> Front passenger, RightRear/Sidecar passenger, Driv~
## $ SAFETY_EQUIPMENT    <fct> Lap Belt/Harness, Lap Belt/Harness, Lap Belt/Harne~
## $ PED_ROLE            <fct> Passenger, Passenger, Driver, Driver, Driver, Pass~
## $ PERSON_SEX          <fct> F, F, M, M, M, F, M, F, M, M, M, M, M, F, F, M, M,~
## $ PERSON_INJURY       <chr> "Unspecified", "Unspecified", "Injured/Killed", "U~
## $ INJURED_KILLED      <dbl> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0,~
```

Split the dataframe into 70% training set and 30% test set.

```r
set.seed(123)
train_indices <- createDataPartition(model_df1$UNIQUE_ID, p = 0.7, list = FALSE)
train1 <- model_df1[train_indices, ]
test1 <- model_df1[-train_indices, ]
```

Create **logistic model** using training set.

```
model1 <- glm(INJURED_KILLED ~ PERSON_AGE+SAFETY_EQUIPMENT+PED_ROLE+PERSON_SEX+PERSON_TYPE+ POSITION_IN_
# vif(model1)
# Error in vif.default(model1) :
#  there are aliased coefficients in the model
```

As expected, there's high multicollinearity in the model, probably due to the existing of all three columns PED_ROLE, PERSON_TYPE & POSITION_IN_VEHICLE. Use only PED_ROLE to model again.

```
model1 <- glm(INJURED_KILLED ~ PERSON_AGE+SAFETY_EQUIPMENT+PED_ROLE+PERSON_SEX, data=train1, family="bi
vif(model1)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## PERSON_AGE       1.162373  1        1.078134
## SAFETY_EQUIPMENT 1.126466  6        1.009973
## PED_ROLE         1.210674  4        1.024185
## PERSON_SEX       1.091441  1        1.044721
```

```
summary(model1)
```

```
##
## Call:
## glm(formula = INJURED_KILLED ~ PERSON_AGE + SAFETY_EQUIPMENT +
##     PED_ROLE + PERSON_SEX, family = "binomial", data = train1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6008  -0.5203  -0.4315  -0.4156   2.3136
##
## Coefficients:
##                                      Estimate Std. Error  z value Pr(>|z|)
## (Intercept)                         0.1779251  0.0166624   10.678  < 2e-16
## PERSON_AGE                         -0.0030206  0.0001577  -19.159  < 2e-16
## SAFETY_EQUIPMENTChild Restraint Only -2.1368117  0.0240225  -88.950  < 2e-16
## SAFETY_EQUIPMENTHelmet Only         1.4404021  0.0240625   59.861  < 2e-16
## SAFETY_EQUIPMENTLap Belt/Harness   -1.9825457  0.0152592 -129.924  < 2e-16
## SAFETY_EQUIPMENTNone               -0.5859834  0.0165960  -35.309  < 2e-16
## SAFETY_EQUIPMENTPads Only           1.1394363  0.3443126    3.309 0.000935
## SAFETY_EQUIPMENTStoppers Only      -0.9161871  0.1270280   -7.212 5.49e-13
## PED_ROLEIn-Line Skater              1.8138099  0.2467411    7.351 1.97e-13
## PED_ROLEOther                       1.5683979  0.1411270   11.113  < 2e-16
## PED_ROLEPassenger                   0.2685770  0.0058722   45.737  < 2e-16
## PED_ROLEPedestrian                  3.0124011  0.1936458   15.556  < 2e-16
## PERSON_SEXM                        -0.4408620  0.0053652  -82.170  < 2e-16
##
## (Intercept)                        ***
## PERSON_AGE                         ***
## SAFETY_EQUIPMENTChild Restraint Only ***
## SAFETY_EQUIPMENTHelmet Only        ***
## SAFETY_EQUIPMENTLap Belt/Harness   ***
## SAFETY_EQUIPMENTNone               ***
## SAFETY_EQUIPMENTPads Only            ***
## SAFETY_EQUIPMENTStoppers Only        ***
## PED_ROLEIn-Line Skater               ***
## PED_ROLEOther                        ***
## PED_ROLEPassenger                    ***
## PED_ROLEPedestrian                   ***
## PERSON_SEXM                          ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1160944  on 1479342  degrees of freedom
## Residual deviance: 1073824  on 1479330  degrees of freedom
## AIC: 1073850
##
## Number of Fisher Scoring iterations: 5
```

All independent variables are statistically significant.

Make **prediction** using test set.

```
test1$prob <- predict(model1, test1, type="response")
summary(test1$prob)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.06596 0.08578 0.10525 0.13308 0.13203 0.99020
```

```
test1$pred <- ifelse(test1$prob >= 0.13, 1, 0)  # average pred value is 0.13308.
head(test1)
```

```
##    UNIQUE_ID PERSON_TYPE PERSON_AGE     POSITION_IN_VEHICLE SAFETY_EQUIPMENT
## 14  10252474    Occupant         50         Front passenger Lap Belt/Harness
## 15  10253763    Occupant         62         Front passenger Lap Belt/Harness
## 17  10253130    Occupant         42                  Driver Lap Belt/Harness
## 18  10248665    Occupant         78 LeftRear/Rear passenger Lap Belt/Harness
## 20  10251087    Occupant         36         Front passenger Lap Belt/Harness
## 21  10250609    Occupant         71                  Driver Lap Belt/Harness
##     PED_ROLE PERSON_SEX  PERSON_INJURY INJURED_KILLED       prob pred
## 14 Passenger          F Injured/Killed              1 0.15616139    1
## 15 Passenger          F    Unspecified              0 0.15144429    1
## 17    Driver          M    Unspecified              0 0.08530592    0
## 18 Passenger          M    Unspecified              0 0.09863281    0
## 20 Passenger          M Injured/Killed              1 0.11049985    0
## 21    Driver          M    Unspecified              0 0.07871455    0
```

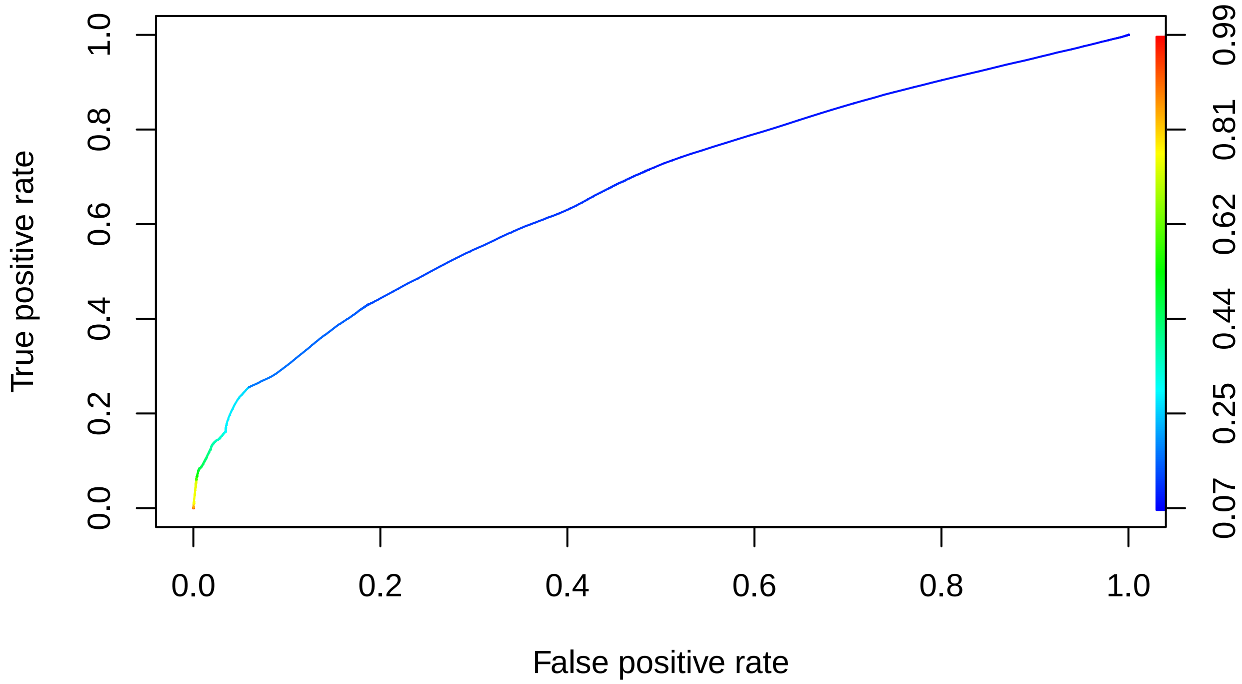
Evaluate model1 using confusion matrix and ROC Curve.

```
# confusion matrix.
confusionMatrix(as.factor(test1$pred), as.factor(test1$INJURED_KILLED), positive="1")
```

```
## Confusion Matrix and Statistics
```

```
##
##           Reference
## Prediction      0      1
##          0 415066  43018
##          1 134534  41382
##
##                Accuracy : 0.7199
##                  95% CI : (0.7188, 0.7211)
##     No Information Rate : 0.8669
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1683
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.49031
##             Specificity : 0.75521
##          Pos Pred Value : 0.23524
##          Neg Pred Value : 0.90609
##              Prevalence : 0.13312
##          Detection Rate : 0.06527
##    Detection Prevalence : 0.27747
##       Balanced Accuracy : 0.62276
##
##        'Positive' Class : 1
##
```

```
# ROC Curve
pred <- prediction(test1$prob, test1$INJURED_KILLED) # create a prediction object in R
perf <- performance(pred, "tpr", "fpr") # tpr and fpr are true and false positive rates
plot(perf, colorize=T)
```



```
# calculate Area Under the Curve for this Logit Model
auc.perf <-  performance(pred, measure = "auc")
auc.perf <- auc.perf@y.values[[1]]
print(paste("AUC value for logistic regression: ", round(auc.perf, 6)))
```

```
## [1] "AUC value for logistic regression:  0.671097"
```

Model 2: PERSON_INJURY == "Killed" as 1, PERSON_INJURY == "Injured" as 0, drop "Unspecified".

```
model_df2 <- model_df %>%
  filter(PERSON_INJURY == "Injured" | PERSON_INJURY == "Killed") %>%
  mutate(KILLED = ifelse(PERSON_INJURY=="Killed",1,0))

model_df2_injured <- model_df2 %>% filter(PERSON_INJURY == "Injured")
model_df2_killed <- model_df2 %>% filter(PERSON_INJURY == "Killed")
glimpse(model_df2_injured)
```

```
## Rows: 280,620
## Columns: 9
## $ UNIQUE_ID           <int> 10253606, 10250834, 10252474, 10251087, 11318006, ~
## $ PERSON_TYPE         <fct> Occupant, Bicyclist, Occupant, Occupant, Occupant,~
## $ PERSON_AGE          <int> 27, 36, 50, 36, 3, 35, 32, 51, 24, 70, 2, 47, 47, ~
## $ POSITION_IN_VEHICLE <fct> Driver, Driver, Front passenger, Front passenger, ~
## $ SAFETY_EQUIPMENT    <fct> Lap Belt/Harness, None, Lap Belt/Harness, Lap Belt~
## $ PED_ROLE            <fct> Driver, Driver, Passenger, Passenger, Passenger, D~
## $ PERSON_SEX          <fct> M, M, F, M, M, M, M, M, F, F, M, M, M, M, F, M, F,~
## $ PERSON_INJURY       <fct> Injured, Injured, Injured, Injured, Injured, Injur~
## $ KILLED              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

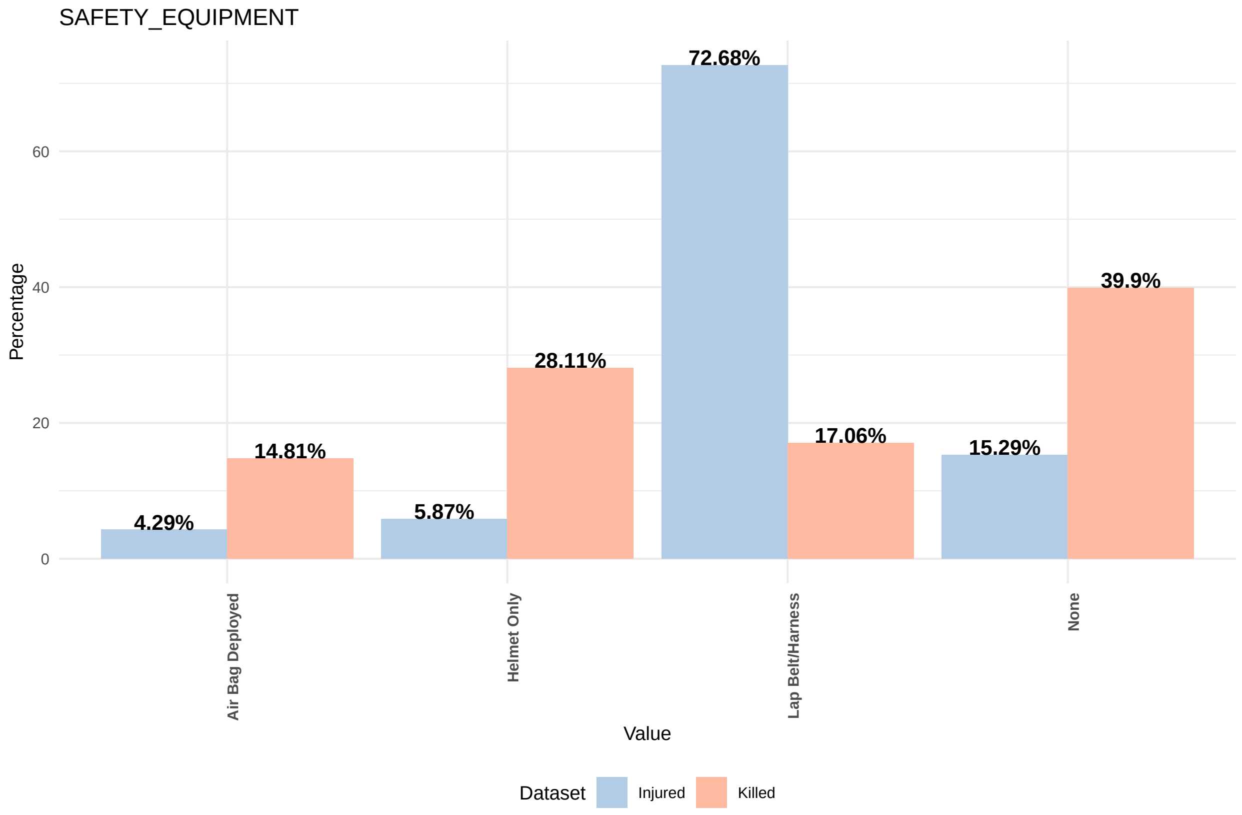
```
glimpse(model_df2_killed)
```

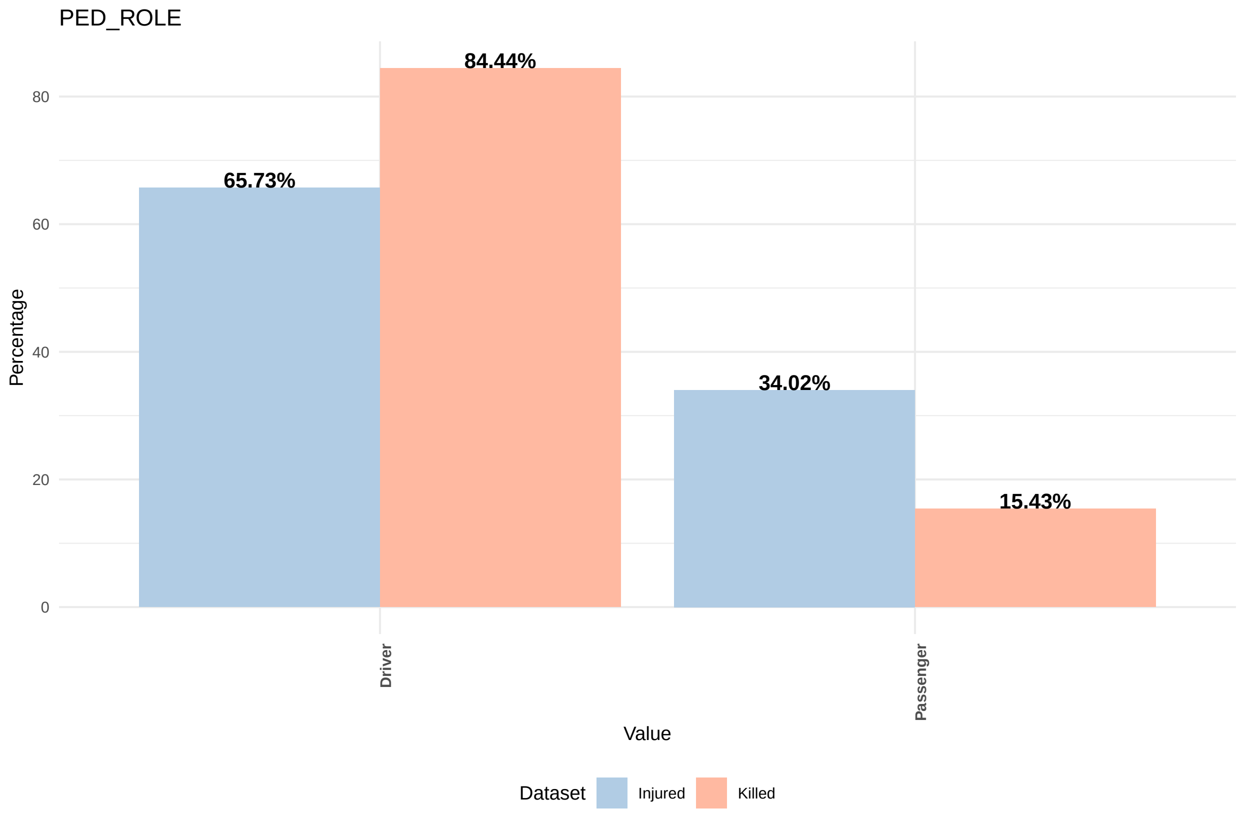
```
## Rows: 797
## Columns: 9
## $ UNIQUE_ID           <int> 10262443, 10269679, 5816773, 11318400, 9643124, 63~
## $ PERSON_TYPE         <fct> Occupant, Occupant, Occupant, Occupant, Occupant, ~
## $ PERSON_AGE          <int> 36, 27, 18, 52, 32, 28, 30, 30, 27, 42, 22, 3, 78,~
## $ POSITION_IN_VEHICLE <fct> Driver, Driver, Does Not Apply, Driver, Driver, Dr~
## $ SAFETY_EQUIPMENT    <fct> Air Bag Deployed, Helmet Only, None, Lap Belt/Harn~
## $ PED_ROLE            <fct> Driver, Driver, Other, Driver, Driver, Driver, Dri~
## $ PERSON_SEX          <fct> M, M, M, M, M, M, M, M, M, M, M, F, M, M, M, M, M,~
## $ PERSON_INJURY       <fct> Killed, Killed, Killed, Killed, Killed, Killed, Ki~
## $ KILLED              <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
```

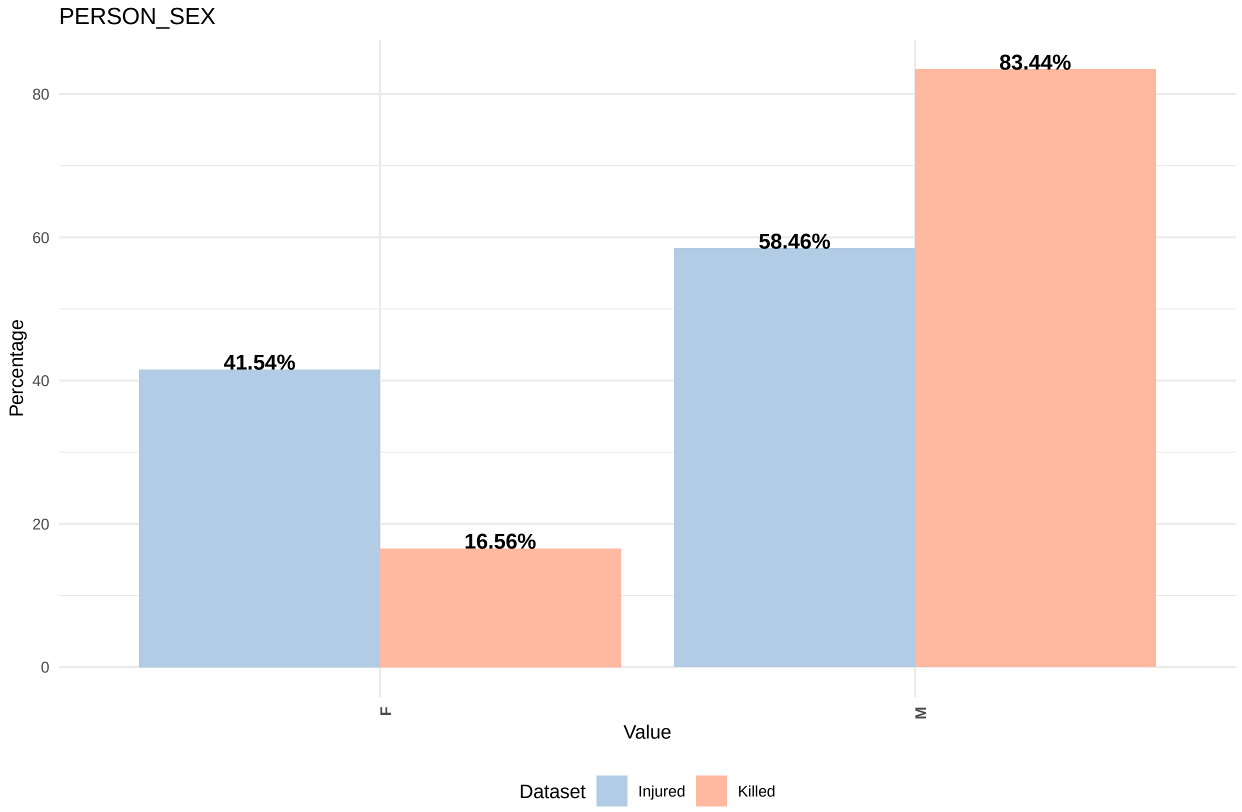
Comparison between certain columns.

```
comp_plot_fun <- function(df1, df2, x, legend_labels) {
  # Combine the dataframes
  combined_df <- rbind(transform(df1, dataset = "df1"), transform(df2, dataset = "df2"))

  # Create a dataframe for the column with unique value counts
  x_count <- combined_df %>%
    group_by(dataset) %>%
    count(dataset, !!sym(x)) %>%
    mutate(Percentage = round(n / sum(n) * 100, 2))

  # Filter out bars with percentage less than 1%
  x_count_filtered <- x_count %>%
    filter(Percentage >= 2)

  # Bar plot
  ggplot(x_count_filtered, aes(x = str_wrap(!!sym(x), width = 40), y = Percentage, fill = dataset)) +
    geom_bar(stat = "identity", position = "dodge") +
    labs(x = "Value", y = "Percentage", fill = "Dataset") +
    scale_fill_manual(values = c("df1" = "#B1CCE4", "df2" = "#FFB9A1"), name = "Dataset",
                      labels = legend_labels) +
    theme_minimal() +
    scale_x_discrete(labels = function(x) str_wrap(x, width = 40)) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, face = "bold"),
          legend.position = "bottom",
          text = element_text(size = 10, color = "black")) +
    geom_text(aes(label = paste0(Percentage, "%"), y = Percentage, vjust = 0),
              position = position_dodge(width = 0.9), fontface = "bold", hjust = 0.5) +
    ggtitle(x)
}
```

```r
comp_cols <- c("SAFETY_EQUIPMENT","PED_ROLE","PERSON_SEX")
for (x in comp_cols) {
  plot(comp_plot_fun(model_df2_injured, model_df2_killed, x, c("Injured", "Killed")))
}
```







Comparing between Injured and Killed, there are several differences between them:

* SAFETY_EQUIPMENT: numbers of Lap Belt/Harness dropped from 73% to 17%; people not wearing any safety equipment jumped from 15% to 40%; Air Bag Deployed indicates the accident is probably severe, thus contributes to mortality; Helmet Only means there's In-Line Skater/Bicyclist or Motorcycle involved, who's in vulnerable position in vehicle accidents. Overall, SAFETY_EQUIPMENT plays a big role between live and death.
* PED_ROLE: driver's percentage is even higher in Killed than Injured.
* PERSON_SEX: male is at a even higher percentage in Killed than Injured

Split the dataframe into 70% training set and 30% test set.
Since model_df2_killed data volume is really small compared to model_df2_injured, split them separately then merge together respectively to make sure they are distributed evenly into training and test set.

```
set.seed(123)

injured_indices <- createDataPartition(model_df2_injured$UNIQUE_ID, p = 0.7, list = FALSE)
train_injured <- model_df2_injured[injured_indices, ]
test_injured <- model_df2_injured[-injured_indices, ]

killed_indices <- createDataPartition(model_df2_killed$UNIQUE_ID, p = 0.7, list = FALSE)
train_killed <- model_df2_killed[killed_indices, ]
test_killed <- model_df2_killed[-killed_indices, ]

train2 <- train_injured %>%
  bind_rows(train_killed) %>%
  mutate(KILLED = ifelse(PERSON_INJURY=="Killed",1,0))
test2 <- test_injured %>%
  bind_rows(test_killed) %>%
  mutate(KILLED = ifelse(PERSON_INJURY=="Killed",1,0))
train2 %>% group_by(KILLED) %>% summarise(n=n()) %>% mutate(percentage=n/sum(n))
```

```
## # A tibble: 2 x 3
##   KILLED      n percentage
##    <dbl>  <int>      <dbl>
## 1      0 196436    0.997
## 2      1    560    0.00284
```

```
test2 %>% group_by(KILLED) %>% summarise(n=n()) %>% mutate(percentage=n/sum(n))
```

```
## # A tibble: 2 x 3
##   KILLED     n percentage
##    <dbl> <int>      <dbl>
## 1      0 84184    0.997
## 2      1   237    0.00281
```

KILLED==1 is evenly distributed into training and test set, but the percentage is really low, only about 0.3%, which will cause "fitted probabilities numerically 0 or 1 occurred" as shown below.

Create **logistic model** using training set.

```
model2 <- glm(KILLED ~ PERSON_AGE+SAFETY_EQUIPMENT+PED_ROLE+PERSON_SEX, data=train2, family="binomial")
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(model2)
```

```
##
## Call:
## glm(formula = KILLED ~ PERSON_AGE + SAFETY_EQUIPMENT + PED_ROLE +
##     PERSON_SEX, family = "binomial", data = train2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.3332  -0.0703  -0.0386  -0.0293   4.1632
##
## Coefficients:
##                                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)                           -6.050704   0.195876 -30.890  < 2e-16 ***
## PERSON_AGE                             0.021721   0.002626   8.271  < 2e-16 ***
## SAFETY_EQUIPMENTChild Restraint Only  -2.501351   1.015611  -2.463   0.0138 *
## SAFETY_EQUIPMENTHelmet Only            0.293147   0.144571   2.028   0.0426 *
## SAFETY_EQUIPMENTLap Belt/Harness      -2.543989   0.155165 -16.395  < 2e-16 ***
## SAFETY_EQUIPMENTNone                  -0.162610   0.135807  -1.197   0.2312
## SAFETY_EQUIPMENTPads Only            -12.928073 782.556163  -0.017   0.9868
## SAFETY_EQUIPMENTStoppers Only        -12.779934 424.043993  -0.030   0.9760
## PED_ROLEIn-Line Skater               -12.675124 518.662269  -0.024   0.9805
## PED_ROLEOther                         -0.328856   1.005260  -0.327   0.7436
## PED_ROLEPassenger                     -0.178857   0.127035  -1.408   0.1592
## PED_ROLEPedestrian                   -12.494717 257.897200  -0.048   0.9614
## PERSON_SEXM                            0.743959   0.124658   5.968  2.4e-09 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7685.0  on 196995  degrees of freedom
## Residual deviance: 6731.2  on 196983  degrees of freedom
## AIC: 6757.2
##
## Number of Fisher Scoring iterations: 16
```

Now PED_ROLE isn’t statistically significant anymore. Remove it and create model again.

```
model2 <- glm(KILLED ~ PERSON_AGE + SAFETY_EQUIPMENT + PERSON_SEX, data=train2, family="binomial")
summary(model2)
```

```
##
## Call:
## glm(formula = KILLED ~ PERSON_AGE + SAFETY_EQUIPMENT + PERSON_SEX,
##     family = "binomial", data = train2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.3352  -0.0731  -0.0384  -0.0293   4.1706
##
## Coefficients:
```

```
##                                          Estimate Std. Error z value Pr(>|z|)    
## (Intercept)                           -6.139360   0.186879 -32.852  < 2e-16 ***
## PERSON_AGE                             0.021937   0.002606   8.417  < 2e-16 ***
## SAFETY_EQUIPMENTChild Restraint Only  -2.623196   1.011699  -2.593  0.00952 ** 
## SAFETY_EQUIPMENTHelmet Only            0.315633   0.143588   2.198  0.02794 *  
## SAFETY_EQUIPMENTLap Belt/Harness      -2.555742   0.154961 -16.493  < 2e-16 ***
## SAFETY_EQUIPMENTNone                  -0.177863   0.135524  -1.312  0.18938    
## SAFETY_EQUIPMENTPads Only            -10.913990 287.643809  -0.038  0.96973    
## SAFETY_EQUIPMENTStoppers Only        -10.789569 156.610537  -0.069  0.94507    
## PERSON_SEXM                            0.801248   0.118773   6.746 1.52e-11 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 7685  on 196995  degrees of freedom
## Residual deviance: 6737  on 196987  degrees of freedom
## AIC: 6755
## 
## Number of Fisher Scoring iterations: 14
```

Note: Stoppers Only & Pads Only are for only In-Line Skater/Bicyclist, which only account for less than 0.1% of data.

Make **prediction** using test set.

```
test2$prob <- predict(model2, test2, type="response")
summary(test2$prob)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
## 5.000e-08 4.348e-04 7.435e-04 2.826e-03 2.731e-03 4.165e-02
```

```
test2$pred <- ifelse(test2$prob >= 0.0028, 1, 0)  # average pred value is 0.002826.
head(test2)
```

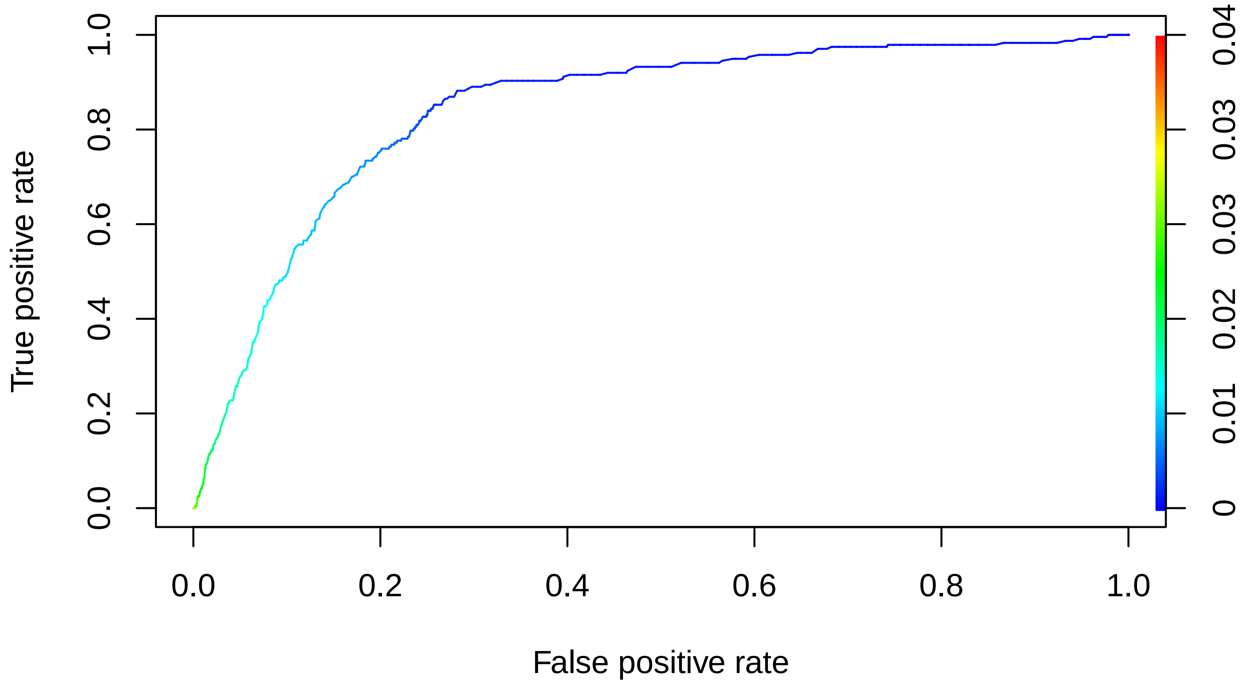
```
##   UNIQUE_ID PERSON_TYPE PERSON_AGE         POSITION_IN_VEHICLE SAFETY_EQUIPMENT
## 1  10251087    Occupant         36             Front passenger Lap Belt/Harness
## 2  10251056   Bicyclist         35                      Driver             None
## 3  10250390    Occupant         51                      Driver Lap Belt/Harness
## 4  10251721    Occupant         38 RightRear/Sidecar passenger Lap Belt/Harness
## 5  10255076    Occupant         15 MiddleFront/Lying passenger Lap Belt/Harness
## 6  10247087    Occupant         31                      Driver Lap Belt/Harness
##    PED_ROLE PERSON_SEX PERSON_INJURY KILLED         prob pred
## 1 Passenger          M       Injured      0 0.0008210288    0
## 2    Driver          M       Injured      0 0.0085929017    1
## 3    Driver          M       Injured      0 0.0011405771    0
## 4 Passenger          F       Injured      0 0.0003851449    0
## 5 Passenger          F       Injured      0 0.0002325783    0
## 6    Driver          F       Injured      0 0.0003303383    0
```

Evaluate using confusion matrix and ROC Curve.

```r
# confusion matrix.
confusionMatrix(as.factor(test2$pred), as.factor(test2$KILLED), positive="1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 63426    41
##          1 20758   196
##
##                Accuracy : 0.7536
##                  95% CI : (0.7507, 0.7565)
##     No Information Rate : 0.9972
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.013
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.827004
##             Specificity : 0.753421
##          Pos Pred Value : 0.009354
##          Neg Pred Value : 0.999354
##              Prevalence : 0.002807
##          Detection Rate : 0.002322
##    Detection Prevalence : 0.248208
##       Balanced Accuracy : 0.790213
##
##        'Positive' Class : 1
##
```

```r
# ROC Curve
pred <- prediction(test2$prob, test2$KILLED) # create a prediction object in R
perf <- performance(pred, "tpr", "fpr") # tpr and fpr are true and false positive rates
plot(perf, colorize=T)
```



```r
# calculate Area Under the Curve for this Logit Model
auc.perf <-  performance(pred, measure = "auc")
auc.perf <- auc.perf@y.values[[1]]
print(paste("AUC value for logistic regression: ", round(auc.perf, 6)))
```

```
## [1] "AUC value for logistic regression:  0.84247"
```

Model2 shows the evidence that SAFETY_EQUIPMENT did play a big role between live and death in vehicle accidents; Age and Sex also have some influence on the results.