CDA project

```r
# Assuming you have loaded your dataset into a variable named 'crash_data'
rm(list = ls())
library(readr)
crash_data <- read_csv("Crash_Reporting.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame
for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)

## Rows: 168515 Columns: 43
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (38): Report Number, Agency Name, ACRS Report Type, Crash Date/Time,
Rou...
## dbl  (5): Local Case Number, Speed Limit, Vehicle Year, Latitude,
Longitude
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```r
# summary of data set
summary(crash_data)
```

```
##  Report Number      Local Case Number   Agency Name        ACRS Report
Type
##  Length:168515      Min.   :1.800e+03   Length:168515      Length:168515
##  Class :character   1st Qu.:1.700e+07   Class :character   Class
:character
##  Mode  :character   Median :1.801e+08   Mode  :character   Mode
:character
##                     Mean   :1.463e+08
##                     3rd Qu.:2.100e+08
##                     Max.   :2.200e+10
##                     NA's   :17
##  Crash Date/Time     Route Type         Road Name         Cross-Street
Type
##  Length:168515      Length:168515      Length:168515      Length:168515
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

```
##
##  Cross-Street Name  Off-Road Description Municipality
##  Length:168515      Length:168515        Length:168515
##  Class :character   Class :character     Class :character
##  Mode  :character   Mode  :character     Mode  :character
##
##
##
##
##  Related Non-Motorist Collision Type       Weather          Surface
Condition
##  Length:168515        Length:168515      Length:168515      Length:168515
##  Class :character     Class :character   Class :character   Class
:character
##  Mode  :character     Mode  :character   Mode  :character   Mode
:character
##
##
##
##
##     Light           Traffic Control    Driver Substance Abuse
##  Length:168515      Length:168515      Length:168515
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Non-Motorist Substance Abuse  Person ID         Driver At Fault
##  Length:168515                 Length:168515      Length:168515
##  Class :character              Class :character   Class :character
##  Mode  :character              Mode  :character   Mode  :character
##
##
##
##
##  Injury Severity    Circumstance       Driver Distracted By
##  Length:168515      Length:168515      Length:168515
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Drivers License State  Vehicle ID        Vehicle Damage Extent
##  Length:168515          Length:168515      Length:168515
##  Class :character       Class :character   Class :character
##  Mode  :character       Mode  :character   Mode  :character
##
##
```

```
##
##
##  Vehicle First Impact Location Vehicle Second Impact Location
##  Length:168515                 Length:168515
##  Class :character              Class :character
##  Mode  :character              Mode  :character
##
##
##
##
##  Vehicle Body Type  Vehicle Movement   Vehicle Continuing Dir
##  Length:168515      Length:168515      Length:168515
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Vehicle Going Dir   Speed Limit    Driverless Vehicle Parked Vehicle
##  Length:168515      Min.   : 0.00   Length:168515      Length:168515
##  Class :character   1st Qu.:25.00   Class :character   Class :character
##  Mode  :character   Median :35.00   Mode  :character   Mode  :character
##                     Mean   :32.56
##                     3rd Qu.:40.00
##                     Max.   :75.00
##
##   Vehicle Year  Vehicle Make       Vehicle Model      Equipment Problems
##  Min.   :   0   Length:168515      Length:168515      Length:168515
##  1st Qu.:2006   Class :character   Class :character   Class :character
##  Median :2011   Mode  :character   Mode  :character   Mode  :character
##  Mean   :1966
##  3rd Qu.:2015
##  Max.   :9999
##
##     Latitude       Longitude        Location
##  Min.   :37.72   Min.   :-79.49   Length:168515
##  1st Qu.:39.02   1st Qu.:-77.19   Class :character
##  Median :39.07   Median :-77.11   Mode  :character
##  Mean   :39.08   Mean   :-77.11
##  3rd Qu.:39.14   3rd Qu.:-77.04
##  Max.   :39.99   Max.   :-75.53
##
```

```
str(crash_data)
```

```
## spc_tbl_ [168,515 × 43] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Report Number                 : chr [1:168515] "MCP3040003N"
"EJ78850038" "MCP2009002G" "MCP3201004C" ...
##  $ Local Case Number             : num [1:168515] 1.9e+08 2.3e+08 2.3e+08
2.3e+08 2.3e+08 ...
```

```
##  $ Agency Name                    : chr [1:168515] "Montgomery County
Police" "Gaithersburg Police Depar" "Montgomery County Police" "Montgomery
County Police" ...
##  $ ACRS Report Type               : chr [1:168515] "Property Damage Crash"
"Property Damage Crash" "Property Damage Crash" "Property Damage Crash" ...
##  $ Crash Date/Time                : chr [1:168515] "05/31/2019 03:00:00 PM"
"07/21/2023 05:59:00 PM" "07/20/2023 03:10:00 PM" "07/23/2023 12:10:00 PM"
...
##  $ Route Type                     : chr [1:168515] NA "Maryland (State)"
"Maryland (State)" "County" ...
##  $ Road Name                      : chr [1:168515] NA "FREDERICK RD"
"GEORGIA AVE" "CRYSTAL ROCK DR" ...
##  $ Cross-Street Type              : chr [1:168515] NA "Unknown" "Maryland
(State)" "County" ...
##  $ Cross-Street Name              : chr [1:168515] NA "WATKINS MILL RD"
"NORBECK RD" "WATERS LANDING DR" ...
##  $ Off-Road Description           : chr [1:168515] "PARKING LOT OF 3215
SPARTAN RD" NA NA NA ...
##  $ Municipality                   : chr [1:168515] NA "N/A" "N/A" "N/A" ...
##  $ Related Non-Motorist           : chr [1:168515] NA NA NA NA ...
##  $ Collision Type                 : chr [1:168515] "OTHER" "STRAIGHT
MOVEMENT ANGLE" "STRAIGHT MOVEMENT ANGLE" "STRAIGHT MOVEMENT ANGLE" ...
##  $ Weather                        : chr [1:168515] "CLEAR" "CLEAR" "CLEAR"
"CLEAR" ...
##  $ Surface Condition              : chr [1:168515] NA "DRY" "DRY" "DRY" ...
##  $ Light                          : chr [1:168515] "DAYLIGHT" "DAYLIGHT"
"DAYLIGHT" "DAYLIGHT" ...
##  $ Traffic Control                : chr [1:168515] "N/A" "TRAFFIC SIGNAL"
"TRAFFIC SIGNAL" "NO CONTROLS" ...
##  $ Driver Substance Abuse         : chr [1:168515] "UNKNOWN" "NONE
DETECTED" "NONE DETECTED" "NONE DETECTED" ...
##  $ Non-Motorist Substance Abuse   : chr [1:168515] NA NA NA NA ...
##  $ Person ID                      : chr [1:168515] "DE2A24CD-7919-4F8D-
BABF-5B75CE12D21E" "E7058A8E-4F18-4D2A-954E-04A099CFED12" "2B404D6D-8DB5-
4CB6-9E71-9F1B8D0A8925" "637D8107-0381-4B8D-848A-B4A93B4D53CE" ...
##  $ Driver At Fault                : chr [1:168515] "Yes" "No" "Yes" "Yes"
...
##  $ Injury Severity                : chr [1:168515] "NO APPARENT INJURY" "NO
APPARENT INJURY" "NO APPARENT INJURY" "NO APPARENT INJURY" ...
##  $ Circumstance                   : chr [1:168515] "N/A" "N/A" "N/A" "N/A"
...
##  $ Driver Distracted By           : chr [1:168515] "UNKNOWN" "NOT
DISTRACTED" "NOT DISTRACTED" "LOOKED BUT DID NOT SEE" ...
##  $ Drivers License State          : chr [1:168515] NA "MD" "MD" "MD" ...
##  $ Vehicle ID                     : chr [1:168515] "165AD539-A8C8-4004-
AF73-B7DCAAA8B3CC" "1C3C3E2F-9A23-4ED0-9BB3-B6C370D99C37" "0483CE47-E0FC-
4BCA-BAB0-B7541820FEE6" "4406AA84-07F8-45F4-88A2-09AD89AC9AAF" ...
##  $ Vehicle Damage Extent          : chr [1:168515] "SUPERFICIAL"
"DISABLING" "FUNCTIONAL" "FUNCTIONAL" ...
##  $ Vehicle First Impact Location : chr [1:168515] "ONE OCLOCK" "THREE
```

```
OCLOCK" "TWELVE OCLOCK" "TWELVE OCLOCK" ...
##  $ Vehicle Second Impact Location: chr [1:168515] "ONE OCLOCK" "TWO
OCLOCK" "TWELVE OCLOCK" "TWELVE OCLOCK" ...
##  $ Vehicle Body Type             : chr [1:168515] "PASSENGER CAR"
"PASSENGER CAR" "PICKUP TRUCK" "PASSENGER CAR" ...
##  $ Vehicle Movement              : chr [1:168515] "PARKING" "MAKING LEFT
TURN" "ACCELERATING" "STARTING FROM LANE" ...
##  $ Vehicle Continuing Dir        : chr [1:168515] "North" "East" "North"
"East" ...
##  $ Vehicle Going Dir             : chr [1:168515] "North" "South" "North"
"East" ...
##  $ Speed Limit                   : num [1:168515] 15 40 35 40 35 30 25 35
35 30 ...
##  $ Driverless Vehicle            : chr [1:168515] "No" "No" "No" "No" ...
##  $ Parked Vehicle                : chr [1:168515] "No" "No" "No" "No" ...
##  $ Vehicle Year                  : num [1:168515] 2004 2011 2019 2016 2016
...
##  $ Vehicle Make                  : chr [1:168515] "HONDA" "GMC" "FORD"
"KIA" ...
##  $ Vehicle Model                 : chr [1:168515] "TK" "TK" "F150" "SW"
...
##  $ Equipment Problems            : chr [1:168515] "UNKNOWN" "NO MISUSE"
"NO MISUSE" "NO MISUSE" ...
##  $ Latitude                      : num [1:168515] 39.2 39.2 39.1 39.2 39.2
...
##  $ Longitude                     : num [1:168515] -77.1 -77.2 -77.1 -77.3
-77.2 ...
##  $ Location                      : chr [1:168515] "(39.15004368, -
77.06308884)" "(39.1592635, -77.21902483)" "(39.10953506, -77.07580619)"
"(39.19014917, -77.26676583)" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   `Report Number` = col_character(),
##   ..   `Local Case Number` = col_double(),
##   ..   `Agency Name` = col_character(),
##   ..   `ACRS Report Type` = col_character(),
##   ..   `Crash Date/Time` = col_character(),
##   ..   `Route Type` = col_character(),
##   ..   `Road Name` = col_character(),
##   ..   `Cross-Street Type` = col_character(),
##   ..   `Cross-Street Name` = col_character(),
##   ..   `Off-Road Description` = col_character(),
##   ..   Municipality = col_character(),
##   ..   `Related Non-Motorist` = col_character(),
##   ..   `Collision Type` = col_character(),
##   ..   Weather = col_character(),
##   ..   `Surface Condition` = col_character(),
##   ..   Light = col_character(),
##   ..   `Traffic Control` = col_character(),
##   ..   `Driver Substance Abuse` = col_character(),
```

```
##   ..   `Non-Motorist Substance Abuse` = col_character(),
##   ..   `Person ID` = col_character(),
##   ..   `Driver At Fault` = col_character(),
##   ..   `Injury Severity` = col_character(),
##   ..   Circumstance = col_character(),
##   ..   `Driver Distracted By` = col_character(),
##   ..   `Drivers License State` = col_character(),
##   ..   `Vehicle ID` = col_character(),
##   ..   `Vehicle Damage Extent` = col_character(),
##   ..   `Vehicle First Impact Location` = col_character(),
##   ..   `Vehicle Second Impact Location` = col_character(),
##   ..   `Vehicle Body Type` = col_character(),
##   ..   `Vehicle Movement` = col_character(),
##   ..   `Vehicle Continuing Dir` = col_character(),
##   ..   `Vehicle Going Dir` = col_character(),
##   ..   `Speed Limit` = col_double(),
##   ..   `Driverless Vehicle` = col_character(),
##   ..   `Parked Vehicle` = col_character(),
##   ..   `Vehicle Year` = col_double(),
##   ..   `Vehicle Make` = col_character(),
##   ..   `Vehicle Model` = col_character(),
##   ..   `Equipment Problems` = col_character(),
##   ..   Latitude = col_double(),
##   ..   Longitude = col_double(),
##   ..   Location = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>

# removing columns
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

crash_data <- select(crash_data,-c(`Report Number`,`Off-Road Description`,
`Municipality`,`Related Non-Motorist`,`Non-Motorist Substance Abuse`, `Person
ID`, `Circumstance`,`Vehicle ID`, `Location`))
crash_data

## # A tibble: 168,515 × 34
##    `Local Case Number` `Agency Name`        `ACRS Report Type` `Crash
Date/Time`
##                  <dbl> <chr>                <chr>              <chr>
```

```
##  1           190026050 Montgomery County P… Property Damage C… 05/31/2019
03:00…
##  2           230034791 Gaithersburg Police… Property Damage C… 07/21/2023
05:59…
##  3           230034583 Montgomery County P… Property Damage C… 07/20/2023
03:10…
##  4           230035036 Montgomery County P… Property Damage C… 07/23/2023
12:10…
##  5           230035152 Montgomery County P… Property Damage C… 07/24/2023
06:10…
##  6           230032956 Montgomery County P… Property Damage C… 07-11-2023
07:40
##  7           230033282 Montgomery County P… Property Damage C… 07-12-2023
20:28
##  8           230032124 Gaithersburg Police… Injury Crash       07-05-2023
23:25
##  9           230034697 Montgomery County P… Property Damage C… 07/21/2023
07:14…
## 10           230034445 Montgomery County P… Property Damage C… 07/19/2023
07:00…
## # ℹ168,505 more rows
## # ℹ30 more variables: `Route Type` <chr>, `Road Name` <chr>,
## #   `Cross-Street Type` <chr>, `Cross-Street Name` <chr>,
## #   `Collision Type` <chr>, Weather <chr>, `Surface Condition` <chr>,
## #   Light <chr>, `Traffic Control` <chr>, `Driver Substance Abuse` <chr>,
## #   `Driver At Fault` <chr>, `Injury Severity` <chr>,
## #   `Driver Distracted By` <chr>, `Drivers License State` <chr>, …
```

```
#removing NA and Duplicates
#is.na(crash_data)
crash_data <- na.omit(crash_data)
#crash_data
duplicates <- crash_data[duplicated(crash_data), ]
# Print the duplicate rows
print(duplicates)
```

```
## # A tibble: 2 × 34
##   `Local Case Number` `Agency Name`         `ACRS Report Type` `Crash
Date/Time`
##                 <dbl> <chr>                 <chr>              <chr>
## 1           180059574 Montgomery County Po… Property Damage C… 11/28/2018
08:15…
## 2           220049292 Montgomery County Po… Property Damage C… 11-08-2022
13:27
## # ℹ30 more variables: `Route Type` <chr>, `Road Name` <chr>,
## #   `Cross-Street Type` <chr>, `Cross-Street Name` <chr>,
## #   `Collision Type` <chr>, Weather <chr>, `Surface Condition` <chr>,
## #   Light <chr>, `Traffic Control` <chr>, `Driver Substance Abuse` <chr>,
## #   `Driver At Fault` <chr>, `Injury Severity` <chr>,
## #   `Driver Distracted By` <chr>, `Drivers License State` <chr>,
```

```
## #   `Vehicle Damage Extent` <chr>, `Vehicle First Impact Location` <chr>,
…

#Remove duplicate
crash_data <- distinct(crash_data)
crash_data

## # A tibble: 143,439 × 34
##    `Local Case Number` `Agency Name`        `ACRS Report Type` `Crash
Date/Time`
##                  <dbl> <chr>                <chr>              <chr>
##  1           230034791 Gaithersburg Police… Property Damage C… 07/21/2023
05:59…
##  2           230034583 Montgomery County P… Property Damage C… 07/20/2023
03:10…
##  3           230035036 Montgomery County P… Property Damage C… 07/23/2023
12:10…
##  4           230035152 Montgomery County P… Property Damage C… 07/24/2023
06:10…
##  5           230032956 Montgomery County P… Property Damage C… 07-11-2023
07:40
##  6           230033282 Montgomery County P… Property Damage C… 07-12-2023
20:28
##  7           230032124 Gaithersburg Police… Injury Crash       07-05-2023
23:25
##  8           230034445 Montgomery County P… Property Damage C… 07/19/2023
07:00…
##  9           230034690 Montgomery County P… Property Damage C… 07/20/2023
05:00…
## 10           230034583 Montgomery County P… Property Damage C… 07/20/2023
03:10…
## # ℹ143,429 more rows
## # ℹ30 more variables: `Route Type` <chr>, `Road Name` <chr>,
## #   `Cross-Street Type` <chr>, `Cross-Street Name` <chr>,
## #   `Collision Type` <chr>, Weather <chr>, `Surface Condition` <chr>,
## #   Light <chr>, `Traffic Control` <chr>, `Driver Substance Abuse` <chr>,
## #   `Driver At Fault` <chr>, `Injury Severity` <chr>,
## #   `Driver Distracted By` <chr>, `Drivers License State` <chr>, …

library(dplyr)
# Assuming 'drivers_data' is your dataset
crash_data <-crash_data %>%
  rename(Local_Case_Number = "Local Case Number",
         ACRS_Report_Type = "ACRS Report Type",
         Collision_type ="Collision Type",
         Weather_rep="Weather",
         Traffic_Control="Traffic Control")

library(dplyr)
# Assuming 'drivers_data' is your dataset
crash_data <-crash_data %>%
```

```
  rename(Light_rep="Light",
         Injury_Severity="Injury Severity",
         Driver_distracted_by="Driver Distracted By",
         Vehicle_damage_extent="Vehicle Damage Extent",
         Vehicle_First_Impact="Vehicle First Impact Location",
         Vehicle_Second_Impact ="Vehicle Second Impact Location",
         Vehicle_movement="Vehicle Movement",
         Speed_limit="Speed Limit",
         Equipment_problems="Equipment Problems")
colnames(crash_data)
```

```
##  [1] "Local_Case_Number"      "Agency Name"            "ACRS_Report_Type"
##  [4] "Crash Date/Time"        "Route Type"             "Road Name"
##  [7] "Cross-Street Type"      "Cross-Street Name"      "Collision_type"
## [10] "Weather_rep"            "Surface Condition"      "Light_rep"
## [13] "Traffic_Control"        "Driver Substance Abuse" "Driver At Fault"
## [16] "Injury_Severity"        "Driver_distracted_by"   "Drivers License
State"
## [19] "Vehicle_damage_extent"  "Vehicle_First_Impact"
"Vehicle_Second_Impact"
## [22] "Vehicle Body Type"      "Vehicle_movement"       "Vehicle Continuing
Dir"
## [25] "Vehicle Going Dir"      "Speed_limit"            "Driverless
Vehicle"
## [28] "Parked Vehicle"         "Vehicle Year"           "Vehicle Make"
## [31] "Vehicle Model"          "Equipment_problems"     "Latitude"
## [34] "Longitude"
```

```
# Load necessary libraries
library(dplyr)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.2
```

```
## corrplot 0.92 loaded
```

```
library(modelr)  # Load 'modelr' for 'encode_numeric()'
```

```
## Warning: package 'modelr' was built under R version 4.3.2
```

```
# Assuming 'crash_data' is your dataset

# Convert categorical variables to factors (if they are not already)
categorical_cols <- sapply(crash_data, is.character)
crash_data[categorical_cols] <- lapply(crash_data[categorical_cols],
as.factor)

# Encode factors as numeric (one-hot encoding)
encoded_data <- crash_data %>%
  mutate_all(funs(as.numeric(as.factor(.))))  # Convert factors to numeric
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## ℹ Please use a list of either functions or lambdas:
##
## # Simple named list: list(mean = mean, median = median)
##
## # Auto named with `tibble::lst()`: tibble::lst(mean, median)
##
## # Using lambdas list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
# Compute correlation matrix
#correlation_matrix <- cor(data_encoded)

correlation_matrix <- cor(encoded_data, use = "pairwise.complete.obs")
```

```
## Warning in cor(encoded_data, use = "pairwise.complete.obs"): the standard
## deviation is zero
```

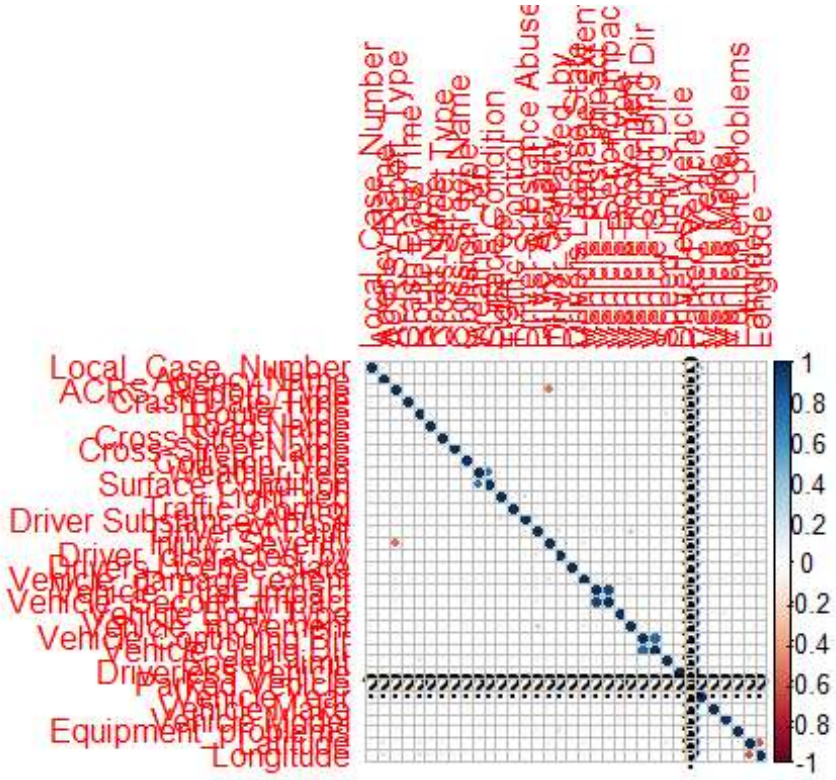
```
correlation_with_target <- cor(encoded_data)[,"ACRS_Report_Type"]
```

```
## Warning in cor(encoded_data): the standard deviation is zero
```

```
# Sort and identify features highly correlated with ACRS_Report_Type
best_features <- names(sort(abs(correlation_with_target), decreasing =
TRUE))[1:5]
best_features
```

```
## [1] "ACRS_Report_Type"      "Injury_Severity"
"Vehicle_damage_extent"
## [4] "Vehicle_Second_Impact" "Vehicle_First_Impact"
```

```
# Plot the correlation matrix
corrplot(correlation_matrix, method = "circle")
```




```
unique(encoded_data$ACRS_Report_Type)
```

```
## [1] 3 2 1
```

```
unique(encoded_data$Vehicle_damage_extent)
```

```
## [1] 2 3 1 7 5 8 4 6
```

```
unique(encoded_data$Injury_Severity)
```

```
## [1] 2 4 5 3 1
```

```
unique(encoded_data$Vehicle_First_Impact)
```

```
##  [1] 12 13 10  4  5  7 11  9  2  1 14  3 16  8  6 15
```

```
unique(encoded_data$Vehicle_Second_Impact)
```

```
##  [1] 14 13 10  4  3  5  7 11  9  2  1 12 16 15  8  6
```

```
encoded_data$ACRS_Report_Type1 <-encoded_data$ACRS_Report_Type == "1"
```

```
encoded_data$ACRS_Report_Type2 <-encoded_data$ACRS_Report_Type == "2"
```

```
encoded_data$ACRS_Report_Type3 <-encoded_data$ACRS_Report_Type == "3"
```

```
dummy_var5 <- model.matrix(~ 0 + encoded_data$ACRS_Report_Type)
```

```
# Creating dummy variables
dummy_vars1 <- model.matrix(~ Vehicle_damage_extent - 1, data = encoded_data)
```

```r
# Creating dummy variables
dummy_vars2 <- model.matrix(~ Injury_Severity - 1, data = encoded_data)

# Creating dummy variables
dummy_vars3 <- model.matrix(~ Vehicle_First_Impact - 1, data = encoded_data)

# Creating dummy variables
dummy_vars4 <- model.matrix(~ Vehicle_Second_Impact - 1, data = encoded_data)

encoded_data <-
cbind(encoded_data,dummy_vars1,dummy_vars2,dummy_vars3,dummy_vars4,dummy_var5
)

str(encoded_data)
```

```
## 'data.frame':    143439 obs. of  42 variables:
##  $ Local_Case_Number            : num  77494 77459 77527 77544 77257 ...
##  $ Agency Name                  : num  2 6 6 6 6 6 2 6 6 6 ...
##  $ ACRS_Report_Type             : num  3 3 3 3 3 3 2 3 3 3 ...
##  $ Crash Date/Time              : num  42038 41803 42474 42689 39818 ...
##  $ Route Type                   : num  4 4 1 1 1 4 4 4 1 4 ...
##  $ Road Name                    : num  1075 1120 697 1910 3082 ...
##  $ Cross-Street Type            : num  9 4 1 1 1 10 9 1 1 4 ...
##  $ Cross-Street Name            : num  5792 3671 5783 857 70 ...
##  $ Collision_type               : num  18 18 18 5 11 11 17 11 11 18 ...
##  $ Weather_rep                  : num  3 3 3 4 3 3 3 3 3 3 ...
##  $ Surface Condition            : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Light_rep                    : num  5 5 5 5 5 6 1 5 5 5 ...
##  $ Traffic_Control              : num  9 9 3 9 3 3 3 4 9 9 ...
##  $ Driver Substance Abuse       : num  10 10 10 10 10 10 2 10 10 10 ...
##  $ Driver At Fault              : num  1 3 3 3 1 3 3 3 1 1 ...
##  $ Injury_Severity              : num  2 2 2 2 2 2 4 2 2 2 ...
##  $ Driver_distracted_by         : num  10 10 8 10 10 10 17 7 10 10 ...
##  $ Drivers License State        : num  28 28 28 8 28 28 28 28 28 28 ...
##  $ Vehicle_damage_extent        : num  2 3 3 3 3 2 1 7 3 2 ...
##  $ Vehicle_First_Impact         : num  12 13 13 13 10 13 13 13 10 4 ...
##  $ Vehicle_Second_Impact        : num  14 13 13 13 10 13 4 13 10 3 ...
##  $ Vehicle Body Type            : num  20 21 20 1 28 1 20 1 31 20 ...
##  $ Vehicle_movement             : num  7 1 19 7 21 18 10 18 21 1 ...
##  $ Vehicle Continuing Dir       : num  1 2 1 5 1 3 2 2 5 1 ...
##  $ Vehicle Going Dir            : num  3 2 1 2 1 3 2 2 2 1 ...
##  $ Speed_limit                  : num  9 8 9 8 7 6 8 7 8 8 ...
##  $ Driverless Vehicle           : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Parked Vehicle               : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Vehicle Year                 : num  80 88 85 85 85 83 82 91 82 86 ...
##  $ Vehicle Make                 : num  522 371 801 1519 481 ...
##  $ Vehicle Model                : num  5223 2312 5043 5223 1094 ...
##  $ Equipment_problems           : num  6 6 6 6 6 6 11 6 6 5 ...
##  $ Latitude                     : num  58901 45357 65078 61518 6915 ...
##  $ Longitude                    : num  10998 44582 3923 15187 56706 ...
##  $ ACRS_Report_Type1            : logi  FALSE FALSE FALSE FALSE FALSE
```

```
FALSE ...
##  $ ACRS_Report_Type2            : logi  FALSE FALSE FALSE FALSE FALSE
FALSE ...
##  $ ACRS_Report_Type3            : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ Vehicle_damage_extent        : num  2 3 3 3 3 2 1 7 3 2 ...
##  $ Injury_Severity              : num  2 2 2 2 2 2 4 2 2 2 ...
##  $ Vehicle_First_Impact         : num  12 13 13 13 10 13 13 13 10 4 ...
##  $ Vehicle_Second_Impact        : num  14 13 13 13 10 13 4 13 10 3 ...
##  $ encoded_data$ACRS_Report_Type: num  3 3 3 3 3 3 2 3 3 3 ...
```

```
library(dplyr)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
# Rename duplicated columns
names(encoded_data) <- make.unique(names(encoded_data))

# Sample 10% of the original data
sampled_data <- encoded_data %>% sample_frac(0.1, replace = FALSE)

# Split the sampled data into 70% for training and 30% for testing
set.seed(123) # For reproducibility
train_indices <- sample(nrow(sampled_data), 0.7 * nrow(sampled_data))
train_data <- sampled_data[train_indices, ]
test_data <- sampled_data[-train_indices, ]

str(sampled_data)
```

```
## 'data.frame':    14344 obs. of  42 variables:
##  $ Local_Case_Number            : num  77532 11947 73658 31418 27076 ...
##  $ Agency Name                  : num  6 5 6 6 6 6 6 6 6 6 ...
##  $ ACRS_Report_Type             : num  3 3 2 3 3 3 3 2 3 3 ...
##  $ Crash Date/Time              : num  42462 17365 8254 3718 49157 ...
##  $ Route Type                   : num  4 1 4 4 4 10 4 4 4 4 ...
##  $ Road Name                    : num  2535 2023 2093 2512 592 ...
##  $ Cross-Street Type            : num  1 1 1 7 4 2 1 1 1 5 ...
##  $ Cross-Street Name            : num  2386 3703 3776 4210 2205 ...
##  $ Collision_type               : num  11 11 5 11 14 16 17 5 16 18 ...
##  $ Weather_rep                  : num  3 3 3 3 3 3 3 3 4 3 ...
##  $ Surface Condition            : num  1 1 1 10 1 1 1 1 12 1 ...
##  $ Light_rep                    : num  5 5 2 5 5 5 4 5 5 5 ...
##  $ Traffic_Control              : num  9 3 9 9 3 4 3 3 9 9 ...
##  $ Driver Substance Abuse       : num  9 10 10 9 10 10 10 10 10 10 ...
##  $ Driver At Fault              : num  1 3 3 1 1 1 3 1 3 1 ...
##  $ Injury_Severity              : num  2 2 3 2 2 2 2 2 2 2 ...
##  $ Driver_distracted_by         : num  10 10 10 10 10 10 17 10 15 10 ...
##  $ Drivers License State        : num  28 28 28 28 28 28 28 28 28 64 ...
##  $ Vehicle_damage_extent        : num  3 3 2 7 7 7 2 2 7 2 ...
```

```
##  $ Vehicle_First_Impact          : num  10 13 7 10 13 9 13 13 2 2 ...
##  $ Vehicle_Second_Impact         : num  10 13 7 10 13 9 13 13 2 2 ...
##  $ Vehicle Body Type             : num  21 20 20 1 19 20 20 20 20 1 ...
##  $ Vehicle_movement              : num  21 10 19 21 10 10 10 10 3 10 ...
##  $ Vehicle Continuing Dir        : num  3 5 2 5 1 3 1 2 2 2 ...
##  $ Vehicle Going Dir             : num  3 5 1 5 1 3 1 2 2 2 ...
##  $ Speed_limit                   : num  8 10 9 8 9 8 9 11 8 8 ...
##  $ Driverless Vehicle            : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Parked Vehicle                : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Vehicle Year                  : num  69 73 76 85 72 74 77 82 76 84 ...
##  $ Vehicle Make                  : num  1519 201 667 758 744 ...
##  $ Vehicle Model                 : num  5102 462 4822 1391 1061 ...
##  $ Equipment_problems            : num  5 6 6 6 6 6 6 6 6 6 ...
##  $ Latitude                      : num  17128 48329 54211 5470 59969 ...
##  $ Longitude                     : num  37438 58921 51931 25131 1926 ...
##  $ ACRS_Report_Type1             : logi  FALSE FALSE FALSE FALSE FALSE
FALSE ...
##  $ ACRS_Report_Type2             : logi  FALSE FALSE TRUE FALSE FALSE FALSE
...
##  $ ACRS_Report_Type3             : logi  TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ Vehicle_damage_extent.1       : num  3 3 2 7 7 7 2 2 7 2 ...
##  $ Injury_Severity.1             : num  2 2 3 2 2 2 2 2 2 2 ...
##  $ Vehicle_First_Impact.1        : num  10 13 7 10 13 9 13 13 2 2 ...
##  $ Vehicle_Second_Impact.1       : num  10 13 7 10 13 9 13 13 2 2 ...
##  $ encoded_data$ACRS_Report_Type: num  3 3 2 3 3 3 3 2 3 3 ...
```

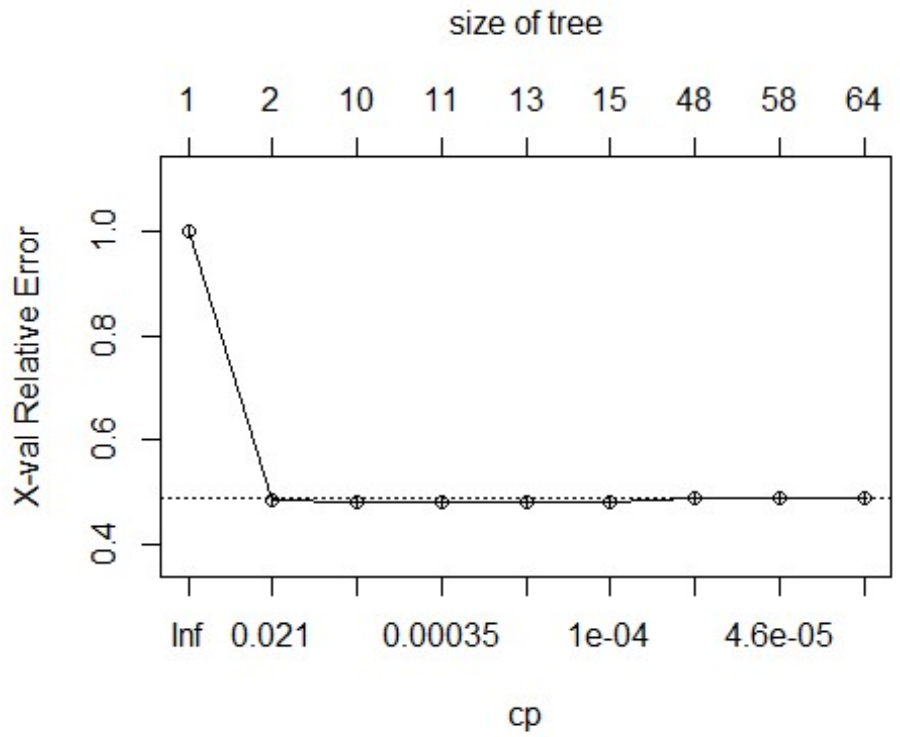
```
# Model-1
# DECISION TREE
library(rpart)
library(rattle)
```

```
## Loading required package: tibble
```

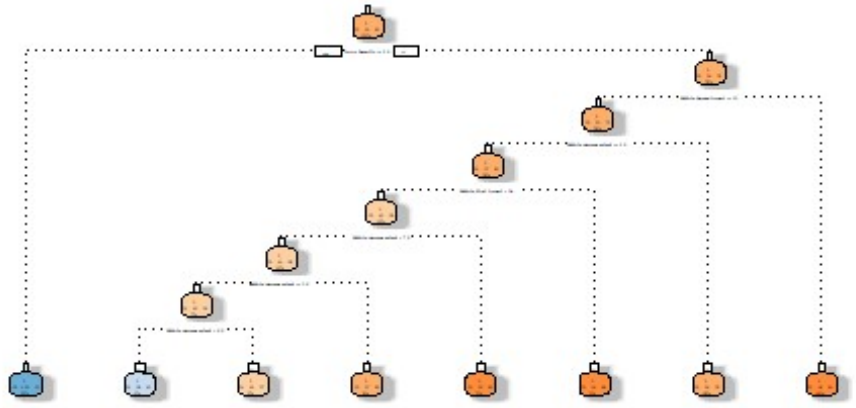
```
## Loading required package: bitops
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
library(rpart.plot)
tree4 = rpart(ACRS_Report_Type~Vehicle_damage_extent + Injury_Severity +
Vehicle_First_Impact +
Vehicle_Second_Impact,data=train_data,method="class",control =
rpart.control(cp=.00001,minsplit=1))
plotcp(tree4)
```




```
cp_val2 = tree4$cptable[which.min(tree4$cptable[,"xerror"]),"CP"]
cp_val2

## [1] 0.0002492522

tree5 = rpart(ACRS_Report_Type~Vehicle_damage_extent + Injury_Severity +
Vehicle_First_Impact + Vehicle_Second_Impact
,data=train_data,method="class",control = rpart.control(cp = cp_val2))

fancyRpartPlot(tree5)
```



Rattle 2023-Dec-01 21:27:07 Administrator

```
summary(tree5)

## Call:
## rpart(formula = ACRS_Report_Type ~ Vehicle_damage_extent + Injury_Severity
+
##     Vehicle_First_Impact + Vehicle_Second_Impact, data = train_data,
##     method = "class", control = rpart.control(cp = cp_val2))
##   n= 10040
##
##             CP nsplit rel error    xerror        xstd
## 1 0.5134596211      0 1.0000000 1.0000000 0.012233178
## 2 0.0008723829      1 0.4865404 0.4865404 0.009883999
## 3 0.0002492522      7 0.4813061 0.4815553 0.009845384
##
## Variable importance
##       Injury_Severity Vehicle_Second_Impact  Vehicle_First_Impact
##                    89                     4                     4
## Vehicle_damage_extent
##                     3
##
## Node number 1: 10040 observations,    complexity param=0.5134596
##   predicted class=3  expected loss=0.3996016  P(node) =1
##     class counts:    27  3985  6028
##    probabilities: 0.003 0.397 0.600
##   left son=2 (2067 obs) right son=3 (7973 obs)
##   Primary splits:
##       Injury_Severity       < 2.5  to the right, improve=1874.37900, (0
```

```
missing)
##       Vehicle_damage_extent < 2.5  to the left,  improve=  75.55826, (0
missing)
##       Vehicle_Second_Impact < 9.5  to the right, improve=  44.51983, (0
missing)
##       Vehicle_First_Impact  < 9.5  to the right, improve=  39.61243, (0
missing)
##   Surrogate splits:
##       Vehicle_damage_extent < 1.5  to the left,  agree=0.796, adj=0.01, (0
split)
##
## Node number 2: 2067 observations
##   predicted class=2  expected loss=0.003386551  P(node) =0.2058765
##     class counts:     7  2060     0
##    probabilities: 0.003 0.997 0.000
##
## Node number 3: 7973 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.2439483  P(node) =0.7941235
##     class counts:    20  1925  6028
##    probabilities: 0.003 0.241 0.756
##   left son=6 (3585 obs) right son=7 (4388 obs)
##   Primary splits:
##       Vehicle_Second_Impact < 12.5 to the right, improve=46.850470, (0
missing)
##       Vehicle_First_Impact  < 12.5 to the right, improve=45.008310, (0
missing)
##       Injury_Severity       < 1.5  to the left,  improve=14.640600, (0
missing)
##       Vehicle_damage_extent < 2.5  to the right, improve= 3.354039, (0
missing)
##   Surrogate splits:
##       Vehicle_First_Impact  < 12.5 to the right, agree=0.974, adj=0.941,
(0 split)
##       Vehicle_damage_extent < 2.5  to the left,  agree=0.615, adj=0.145,
(0 split)
##       Injury_Severity       < 1.5  to the left,  agree=0.550, adj=0.000,
(0 split)
##
## Node number 6: 3585 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.3046025  P(node) =0.3570717
##     class counts:    14  1078  2493
##    probabilities: 0.004 0.301 0.695
##   left son=12 (1763 obs) right son=13 (1822 obs)
##   Primary splits:
##       Vehicle_damage_extent < 2.5  to the right, improve=29.22909, (0
missing)
##       Vehicle_Second_Impact < 13.5 to the left,  improve=18.20912, (0
missing)
##       Vehicle_First_Impact  < 13.5 to the left,  improve=15.77242, (0
missing)
```

```
##   Surrogate splits:
##       Vehicle_First_Impact  < 13.5 to the right, agree=0.550, adj=0.085,
(0 split)
##       Vehicle_Second_Impact < 13.5 to the right, agree=0.547, adj=0.079,
(0 split)
##
## Node number 7: 4388 observations
##   predicted class=3  expected loss=0.1943938  P(node) =0.4370518
##     class counts:     6   847  3535
##    probabilities: 0.001 0.193 0.806
##
## Node number 12: 1763 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.3698242  P(node) =0.1755976
##     class counts:     8   644  1111
##    probabilities: 0.005 0.365 0.630
##   left son=24 (1512 obs) right son=25 (251 obs)
##   Primary splits:
##       Vehicle_First_Impact  < 13.5 to the left,  improve=30.992640, (0
missing)
##       Vehicle_Second_Impact < 13.5 to the left,  improve=30.065520, (0
missing)
##       Vehicle_damage_extent < 7.5  to the left,  improve= 3.923532, (0
missing)
##   Surrogate splits:
##       Vehicle_Second_Impact < 13.5 to the left,  agree=0.989, adj=0.924,
(0 split)
##       Vehicle_damage_extent < 7.5  to the left,  agree=0.860, adj=0.016,
(0 split)
##
## Node number 13: 1822 observations
##   predicted class=3  expected loss=0.2414929  P(node) =0.1814741
##     class counts:     6   434  1382
##    probabilities: 0.003 0.238 0.759
##
## Node number 24: 1512 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.4080688  P(node) =0.1505976
##     class counts:     7   610   895
##    probabilities: 0.005 0.403 0.592
##   left son=48 (1492 obs) right son=49 (20 obs)
##   Primary splits:
##       Vehicle_damage_extent < 7.5  to the left,  improve=5.1309480, (0
missing)
##       Vehicle_First_Impact  < 12.5 to the right, improve=0.6322254, (0
missing)
##       Vehicle_Second_Impact < 13.5 to the left,  improve=0.4291018, (0
missing)
##
## Node number 25: 251 observations
##   predicted class=3  expected loss=0.1394422  P(node) =0.025
##     class counts:     1    34   216
```

```
##    probabilities: 0.004 0.135 0.861
##
## Node number 48: 1492 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.4128686  P(node) =0.1486056
##     class counts:     7   609   876
##    probabilities: 0.005 0.408 0.587
##   left son=96 (656 obs) right son=97 (836 obs)
##   Primary splits:
##       Vehicle_damage_extent < 3.5  to the right, improve=4.8020800, (0
missing)
##       Vehicle_First_Impact  < 12.5 to the right, improve=0.5650677, (0
missing)
##       Vehicle_Second_Impact < 13.5 to the left,  improve=0.3113380, (0
missing)
##   Surrogate splits:
##       Vehicle_Second_Impact < 13.5 to the right, agree=0.564, adj=0.009,
(0 split)
##
## Node number 49: 20 observations
##   predicted class=3  expected loss=0.05  P(node) =0.001992032
##     class counts:     0     1    19
##    probabilities: 0.000 0.050 0.950
##
## Node number 96: 656 observations,    complexity param=0.0008723829
##   predicted class=3  expected loss=0.4588415  P(node) =0.06533865
##     class counts:     4   297   355
##    probabilities: 0.006 0.453 0.541
##   left son=192 (110 obs) right son=193 (546 obs)
##   Primary splits:
##       Vehicle_damage_extent < 5.5  to the left,  improve=5.1574990, (0
missing)
##       Vehicle_First_Impact  < 12.5 to the right, improve=0.5561301, (0
missing)
##       Vehicle_Second_Impact < 13.5 to the left,  improve=0.1906517, (0
missing)
##   Surrogate splits:
##       Vehicle_Second_Impact < 15   to the right, agree=0.835, adj=0.018,
(0 split)
##
## Node number 97: 836 observations
##   predicted class=3  expected loss=0.3767943  P(node) =0.08326693
##     class counts:     3   312   521
##    probabilities: 0.004 0.373 0.623
##
## Node number 192: 110 observations
##   predicted class=2  expected loss=0.4090909  P(node) =0.01095618
##     class counts:     1    65    44
##    probabilities: 0.009 0.591 0.400
##
## Node number 193: 546 observations
```

```
##   predicted class=3  expected loss=0.4304029  P(node) =0.05438247
##     class counts:     3   232   311
##    probabilities: 0.005 0.425 0.570
```

```
pred_tree2 =  predict(tree5, newdata=test_data, type="class")
# Confusion Matrix
cof_matrix1 <- table(predictions = pred_tree2, actual =
test_data$ACRS_Report_Type)
cof_matrix1
```

```
##            actual
## predictions    1    2    3
##           1    0    0    0
##           2    5  899   20
##           3   11  762 2607
```

```
# Calculate Accuracy
accuracy1 <- sum(diag(cof_matrix1)) / sum(cof_matrix1)
accuracy1
```

```
## [1] 0.8145911
```

```
precision1 <- diag(cof_matrix1) / rowSums(cof_matrix1)
precision1
```

```
##         1         2         3
##       NaN 0.9729437 0.7713018
```

```
#Model-2
#LOGISTIC REGRESSION
fit_ols1 = glm(ACRS_Report_Type2~ Vehicle_damage_extent +
Injury_Severity+Vehicle_First_Impact+Vehicle_Second_Impact
,data=train_data,family="binomial",control=glm.control(trace=TRUE))
```

```
## Deviance = 9616.425 Iterations - 1
## Deviance = 9102.288 Iterations - 2
## Deviance = 8923.84 Iterations - 3
## Deviance = 8872.16 Iterations - 4
## Deviance = 8861.196 Iterations - 5
## Deviance = 8860.129 Iterations - 6
## Deviance = 8860.111 Iterations - 7
## Deviance = 8860.111 Iterations - 8
```

```
summary(fit_ols1)
```

```
##
## Call:
## glm(formula = ACRS_Report_Type2 ~ Vehicle_damage_extent + Injury_Severity
+
##     Vehicle_First_Impact + Vehicle_Second_Impact, family = "binomial",
##     data = train_data, control = glm.control(trace = TRUE))
##
```

```
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -13.432904   0.621956 -21.598   <2e-16 ***
## Vehicle_damage_extent   0.002056   0.012360   0.166   0.8679
## Injury_Severity         5.810729   0.303809  19.126   <2e-16 ***
## Vehicle_First_Impact    0.019506   0.018359   1.062   0.2880
## Vehicle_Second_Impact   0.045792   0.018370   2.493   0.0127 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 13488.5  on 10039  degrees of freedom
## Residual deviance:  8860.1  on 10035  degrees of freedom
## AIC: 8870.1
##
## Number of Fisher Scoring iterations: 8
```

```
predictions_ols1 = predict(fit_ols1,newdata=test_data)>.5
c_mat <-
table(predictions=predictions_ols1,actual=test_data$ACRS_Report_Type)
c_mat
```

```
##            actual
## predictions    1    2    3
##       FALSE   11  800 2627
##       TRUE     5  861    0
```

```
accuracy_ols <- sum(diag(c_mat)) / sum(c_mat)
accuracy_ols
```

```
## [1] 0.2026022
```

```
precision_ols <- diag(c_mat) / rowSums(c_mat)
precision_ols
```

```
##       FALSE        TRUE
## 0.003199535 0.994226328
```

```
#Model-3
#Naive Bayes classifier
set.seed(123)
library(e1071)
n_bayes_fit1 = naiveBayes(ACRS_Report_Type~Vehicle_damage_extent +
Injury_Severity+Vehicle_First_Impact+Vehicle_Second_Impact,data=train_data)

predictions_nbayes1 = predict(n_bayes_fit1,newdata=test_data)

conf_matrix <-
table(predictions=predictions_nbayes1,actual=test_data$ACRS_Report_Type)
conf_matrix
```

```
##            actual
## predictions    1    2    3
##           1    0    0    0
##           2   11  861    0
##           3    5  800 2627
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
conf_matrix
```

```
##            actual
## predictions    1    2    3
##           1    0    0    0
##           2   11  861    0
##           3    5  800 2627
```

```
accuracy_nbayes1 <- sum(diag(conf_matrix)) / sum(conf_matrix)
accuracy_nbayes1
```

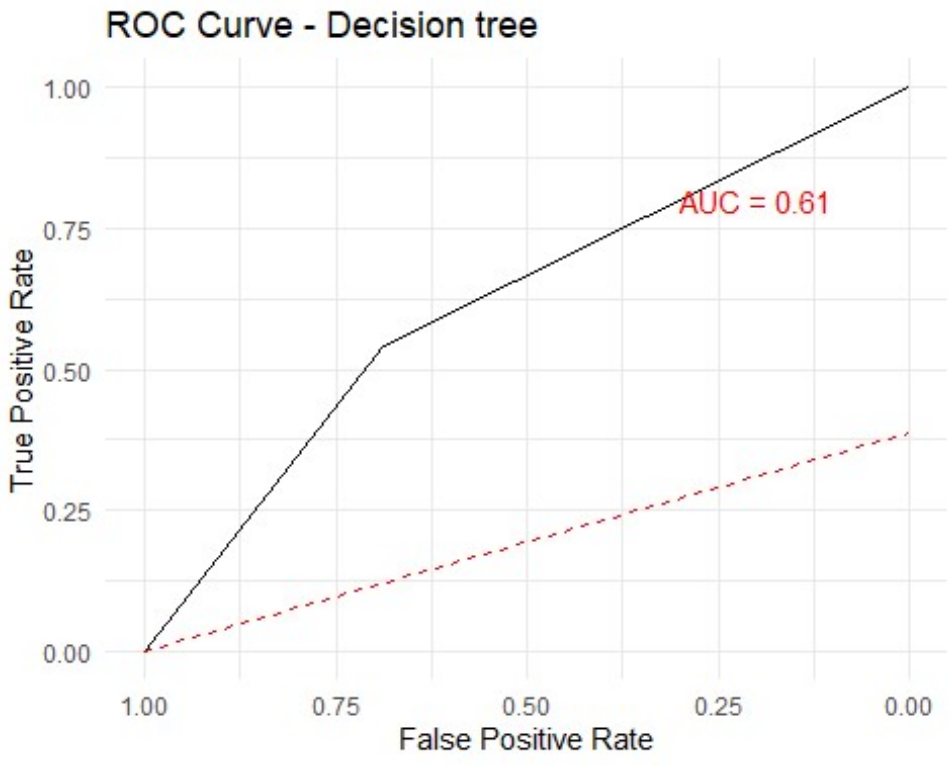
```
## [1] 0.8104089
```

```
precision <- diag(conf_matrix) / rowSums(conf_matrix)
precision
```

```
##         1         2         3
##       NaN 0.9873853 0.7654429
```

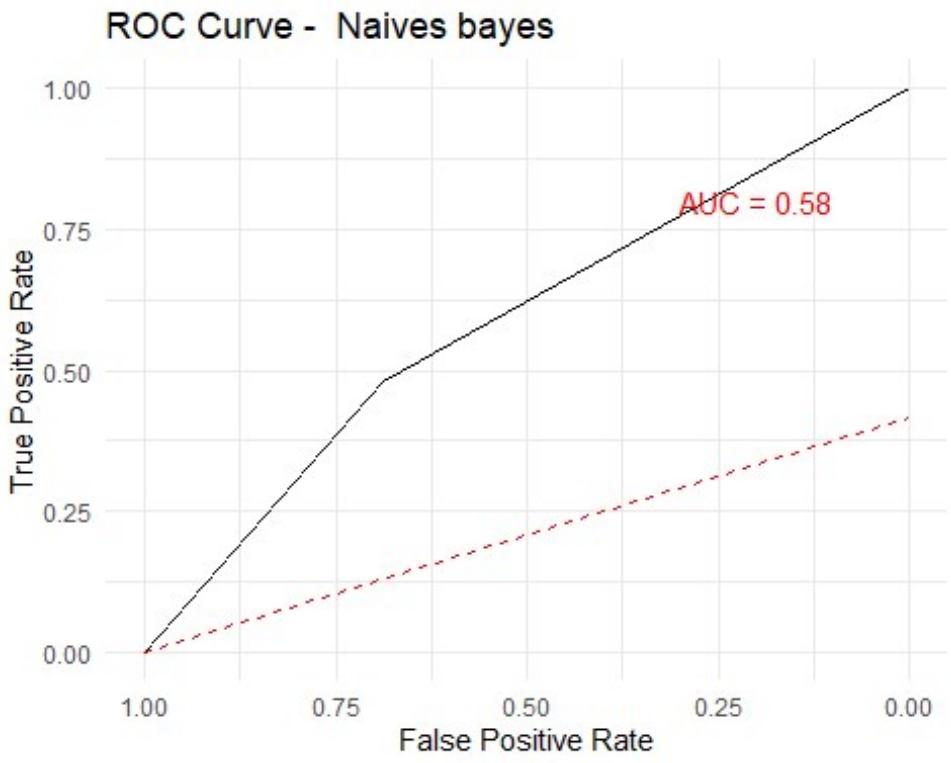
```
#ROC Curves
plot_roc_curve <- function(true_labels, predicted_probs, model_name,
plot_legend = TRUE) {
roc_data <- roc(true_labels, predicted_probs)
auc_value <- auc(roc_data)
ggroc(roc_data) +
geom_segment(aes(x = 1, y = 0, xend = 0, yend = 1 - auc_value),
linetype = "dashed", color = "red") +
annotate("text", x = 0.2, y = 0.8, label = paste("AUC =", round(auc_value,
2)),
color = "red", size = 4) +
labs(title = paste("ROC Curve -", model_name),
x = "False Positive Rate",
y = "True Positive Rate") +
theme_minimal() +
if (plot_legend) {
```

```
theme(legend.position="bottom")
} else {
theme(legend.position="none")
}
}

# Plot ROC curve for Logistic Regression Model 1
plot_roc_curve(test_data$ACRS_Report_Type,as.numeric(pred_tree2), "Decision
tree")

## Warning in roc.default(true_labels, predicted_probs): 'response' has more
than
## two levels. Consider setting 'levels' explicitly or using 'multiclass.roc'
## instead

## Setting levels: control = 1, case = 2

## Setting direction: controls > cases
```

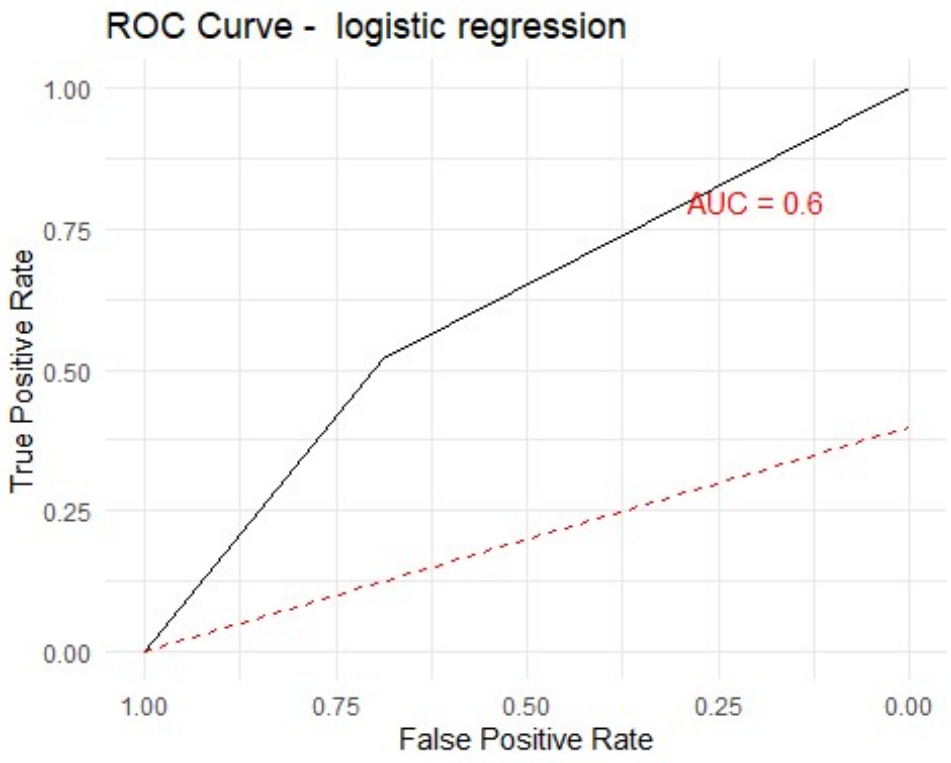



```
plot_roc_curve(test_data$ACRS_Report_Type,as.numeric(predictions_nbayes1), "
Naives bayes")

## Warning in roc.default(true_labels, predicted_probs): 'response' has more
than
## two levels. Consider setting 'levels' explicitly or using 'multiclass.roc'
## instead

## Setting levels: control = 1, case = 2
```

```
## Setting direction: controls < cases
```




```
str(predictions_ols1)

##  Named logi [1:4304] TRUE FALSE FALSE FALSE FALSE FALSE ...
##  - attr(*, "names")= chr [1:4304] "62595" "6332" "65741" "57493" ...

plot_roc_curve(test_data$ACRS_Report_Type,as.numeric(predictions_ols1), "
logistic regression")

## Warning in roc.default(true_labels, predicted_probs): 'response' has more
than
## two levels. Consider setting 'levels' explicitly or using 'multiclass.roc'
## instead

## Setting levels: control = 1, case = 2

## Setting direction: controls < cases
```


ROC Curve - logistic regression
True Positive Rate
1.00
0.75
0.50
0.25
0.00
AUC = 0.6
1.00
0.75
0.50
0.25
0.00
False Positive Rate