HW4

Business Analytics and Data Mining

Zachary Palmore

4/17/2021

Assignment 4

```
# Packages
library(tidyverse)
library(kableExtra)
library(ggcorrplot)
library(reshape2)
library(bestNormalize)
library(caret)
library(MASS)
library(pROC)
library(stats)
library(ROCR)
theme_set(theme_minimal())
```

Purpose

In this homework assignment, we will explore, analyze and model a data set containing approximately 8000 records representing a customer at an auto insurance company. Each record has two response variables. The first response variable, TARGET_FLAG, is a 1 or a 0. A "1" means that the person was in a car crash. A zero means that the person was not in a car crash. The second response variable is TARGET_AMT. This value is zero if the person did not crash their car. But if they did crash their car, this number will be a value greater than zero.

Our objective is to build multiple linear regression and binary logistic regression models on the training data to predict the probability that a person will crash their car and also the amount of money it will cost if the person does crash their car. We can only use the variables given (or variables derived from the variables provided). Below is a short description of the variables of interest in the data set:

```
# short descriptions of variables as table from matrix
vardesc <- data.frame(matrix(c(
'INDEX',    'Identification variable',
'TARGET_FLAG',  'Was car in a crash? 1 = Yes, 0 = No',
'TARGET_AMT',   'Cost of car crash',
'AGE',  'Age of driver',
'BLUEBOOK', 'Value of vehicle',
'CAR_AGE',  'Vehicle age',
'CAR_TYPE', 'Type of car',
'CAR_USE', 'Main purpose the vehicle is used for',
'CLM_FREQ', 'Number of claims filed in past five years',
'EDUCATION',    'Maximum education level',
'HOMEKIDS', 'Number of children at home',
'HOME_VAL', 'Value of driver\'s home',
'INCOME',   'Annual income of the driver',
'JOB',  'Type of job by standard collar categories',
'KIDSDRIV', 'Number of children who drive',
'MSTATUS',  'Marital status',
'MVR_PTS',  'Motor vehicle inspection points',
'OLDCLAIM', 'Total claims payout in past five years',
'PARENT1',  'Single parent status',
'RED_CAR',  '1 if car is red, 0 if not',
'REVOKED',  'License revoked in past 7 years status',
'SEX',  'Driver gender',
'TIF',  'Time in force',
'TRAVETIME',    'Distance to work in minutes',
'URBANICITY',   'Category of how urban the area the driver lives is',
'YOJ', 'Number of years on the job'
),  byrow = TRUE, ncol = 2))
colnames(vardesc) <- c('Variable', 'Description')
kbl(vardesc, booktabs = T, caption = "Variable Descriptions") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"), full_width = F)
```

Table 1: Variable Descriptions

Variable	Description
INDEX	Identification variable
TARGET_FLAG	Was car in a crash? 1 = Yes, 0 = No
TARGET_AMT	Cost of car crash
AGE	Age of driver
BLUEBOOK	Value of vehicle
CAR_AGE	Vehicle age
CAR_TYPE	Type of car
CAR_USE	Main purpose the vehicle is used for
CLM_FREQ	Number of claims filed in past five years
EDUCATION	Maximum education level
HOMEKIDS	Number of children at home
HOME_VAL	Value of driver's home
INCOME	Annual income of the driver
JOB	Type of job by standard collar categories
KIDSDRIV	Number of children who drive
MSTATUS	Marital status
MVR_PTS	Motor vehicle inspection points
OLDCLAIM	Total claims payout in past five years
PARENT1	Single parent status
RED_CAR	1 if car is red, 0 if not
REVOKED	License revoked in past 7 years status
SEX	Driver gender
TIF	Time in force
TRAVETIME	Distance to work in minutes
URBANICITY	Category of how urban the area the driver lives is
YOJ	Number of years on the job

Introduction

There are [ ] observations of [ ] variables in this data set.

```
tdata <- read.csv(
  "https://raw.githubusercontent.com/palmorezm/msds/main/621/HW4/insurance_training_data.csv")
edata <- read.csv(
  "https://raw.githubusercontent.com/palmorezm/msds/main/621/HW4/insurance-evaluation-data.csv")
```

```
initialobs <- tdata[1:4,]
kbl(t(initialobs), booktabs = T, caption = "Initial Observations") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"), full_width = F) %>%
  add_header_above(c(" ", " ", "Row Number", " ", " ")) %>%
  footnote(c("Includes the first four observations of all variables in the data"))
```

Table 2: Initial Observations

	Row Number			
	1	2	3	4
INDEX	1	2	4	5
TARGET_FLAG	0	0	0	0
TARGET_AMT	0	0	0	0
KIDSDRIV	0	0	0	0
AGE	60	43	35	51
HOMEKIDS	0	0	1	0
YOJ	11	11	10	14
INCOME	$67,349	$91,449	$16,039	
PARENT1	No	No	No	No
HOME_VAL	$0	$257,252	$124,191	$306,251
MSTATUS	z_No	z_No	Yes	Yes
SEX	M	M	z_F	M
EDUCATION	PhD	z_High School	z_High School	<High School
JOB	Professional	z_Blue Collar	Clerical	z_Blue Collar
TRAVTIME	14	22	5	32
CAR_USE	Private	Commercial	Private	Private
BLUEBOOK	$14,230	$14,940	$4,010	$15,440
TIF	11	1	4	7
CAR_TYPE	Minivan	Minivan	z_SUV	Minivan
RED_CAR	yes	yes	no	yes
OLDCLAIM	$4,461	$0	$38,690	$0
CLM_FREQ	2	0	2	0
REVOKED	No	No	No	No
MVR_PTS	3	0	3	0
CAR_AGE	18	1	10	6
URBANICITY	Highly Urban/ Urban	Highly Urban/ Urban	Highly Urban/ Urban	Highly Urban/ Urban

Note:
Includes the first four observations of all variables in the data

Data Exploration

Describe the size and the variables in the insurance training data set. Consider that too much detail will cause a manager to lose interest while too little detail will make the manager consider that you aren't doing your job. Some suggestions are given below. Please do NOT treat this as a check list of things to do to complete the assignment. You should have your own thoughts on what to tell the boss. These are just ideas. a. Mean / Standard Deviation / Median b. Bar Chart or Box Plot of the data c. Is the data correlated to the target variable (or to other variables?) d. Are any of the variables missing and need to be imputed "fixed"?

Before we delve into the nitty gritty of this data set, we should consider what effect each of these variables might exert on the outcome. Since there are two targets of different types, and thus two models (one logistic classifier and one regression) there could be an influence on either or both models. As we understand it, the theoretical effects of each variable are recorded in the table below.

```
# theoretical effects
vareffects <- data.frame(matrix(c(
'INDEX',    'None',
'TARGET_FLAG',  'None',
'TARGET_AMT',   'None',
'AGE',  'Youngest and Oldest may have higher risk of accident',
'BLUEBOOK', 'Unknown on probability of collision but correlated with payout',
'CAR_AGE',  'Unknown on probability of collision but correlated with payout',
'CAR_TYPE', 'Unknown on probability of collision but correlated with payout',
'CAR_USE', 'Commerical vehicles might increase risk of accident',
'CLM_FREQ', 'Higher claim frequency increases likelihood of future claims',
'EDUCATION',    'Theoretically higher education levels lower risk',
'HOMEKIDS', 'Unknown',
'HOME_VAL', 'Theoretically home owners reduce risk due to more responsible driving',
'INCOME',   'Theoretically wealthier drivers have fewer accidents',
'JOB',  'Theoretically white collar+ jobs are safer',
'KIDSDRIV', 'Increased risk of accident from inexperienced driver',
'MSTATUS',  'Theoretically married people drive safer',
'MVR_PTS',  'Increased risk of accident',
'OLDCLAIM', 'Increased risk of higher payout with previous payout',
'PARENT1',  'Unknown',
'RED_CAR',  'Theoretically increased risk of accident based on urban legend',
'REVOKED',  'Increased risk of accident if revoked',
'SEX',  'Theoretically increased risk of accident for women based on urban legend',
'TIF',  'Decreased risk for those who have greater loyalty',
'TRAVETIME',    'Longer distances increase risk of accident',
'URBANICITY',   'The more urban the area the greater the risk of accident',
'YOJ', 'Decreased risk for those with greater longevity'
),  byrow = TRUE, ncol = 2))
colnames(vareffects) <- c('Variable', 'Effect')
kbl(vareffects, booktabs = T, caption = "Theoretical Variable Effects") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"), full_width = F)
```

Table 3: Theoretical Variable Effects

Variable	Effect
INDEX	None
TARGET_FLAG	None
TARGET_AMT	None
AGE	Youngest and Oldest may have higher risk of accident
BLUEBOOK	Unknown on probability of collision but correlated with payout
CAR_AGE	Unknown on probability of collision but correlated with payout
CAR_TYPE	Unknown on probability of collision but correlated with payout
CAR_USE	Commerical vehicles might increase risk of accident
CLM_FREQ	Higher claim frequency increases likelihood of future claims
EDUCATION	Theoretically higher education levels lower risk
HOMEKIDS	Unknown
HOME_VAL	Theoretically home owners reduce risk due to more responsible driving
INCOME	Theoretically wealthier drivers have fewer accidents
JOB	Theoretically white collar+ jobs are safer
KIDSDRIV	Increased risk of accident from inexperienced driver
MSTATUS	Theoretically married people drive safer
MVR_PTS	Increased risk of accident
OLDCLAIM	Increased risk of higher payout with previous payout
PARENT1	Unknown
RED_CAR	Theoretically increased risk of accident based on urban legend
REVOKED	Increased risk of accident if revoked
SEX	Theoretically increased risk of accident for women based on urban legend
TIF	Decreased risk for those who have greater loyalty
TRAVETIME	Longer distances increase risk of accident
URBANICITY	The more urban the area the greater the risk of accident
YOJ	Decreased risk for those with greater longevity

This table considers the effects of both models but they are only theoretical and may not necessarily reflect the true influence. We will evaluate these directly in the model selection process. For now, they will serve as general baseline expectations for exploration and preparation. We continue by exploring the data to determine where munging may be necessary.

Unfortunately, this data needs work before we are able to make visualizations and contemplate improvements to the model. We consider the amount of missing values in relative proportions to each variable, followed by their respective data types, an example observation of each type, and the quantity of unique factors to each variable. This will help narrow down what is needed to prepare the data for modeling. Results are shown in the table:

```
tdata.nas <- lapply(tdata, function(x) sum(is.na(x)))
tdata.len <- lapply(tdata, function(x) length(x))
tdata.permis <- lapply(tdata, function(x) round(sum(is.na(x))/length(x)*100, 1))
tdata.types <- lapply(tdata, function(x) class(x))
tdata.firstob <- lapply(tdata, function(x) head(x, 1))
tdata.uniques <- lapply(tdata, function(x) length(unique(factor(x))))
tdata.tbl.natypes <- cbind(tdata.nas, tdata.len, tdata.permis, tdata.types, tdata.firstob, tdata.unique
colnames(tdata.tbl.natypes) <- c("Missing", "Total", "%", "Data Type", "Example", "Factors")
kbl(tdata.tbl.natypes, booktabs = T, caption = "Data Characteristics") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"), full_width = F)
```

Table 4: Data Characteristics

	Missing	Total	%	Data Type	Example	Factors
INDEX	0	8161	0	integer	1	8161
TARGET_FLAG	0	8161	0	integer	0	2
TARGET_AMT	0	8161	0	numeric	0	1949
KIDSDRIV	0	8161	0	integer	0	5
AGE	6	8161	0.1	integer	60	61
HOMEKIDS	0	8161	0	integer	0	6
YOJ	454	8161	5.6	integer	11	22
INCOME	0	8161	0	character	$67,349	6613
PARENT1	0	8161	0	character	No	2
HOME_VAL	0	8161	0	character	$0	5107
MSTATUS	0	8161	0	character	z_No	2
SEX	0	8161	0	character	M	2
EDUCATION	0	8161	0	character	PhD	5
JOB	0	8161	0	character	Professional	9
TRAVTIME	0	8161	0	integer	14	97
CAR_USE	0	8161	0	character	Private	2
BLUEBOOK	0	8161	0	character	$14,230	2789
TIF	0	8161	0	integer	11	23
CAR_TYPE	0	8161	0	character	Minivan	6
RED_CAR	0	8161	0	character	yes	2
OLDCLAIM	0	8161	0	character	$4,461	2857
CLM_FREQ	0	8161	0	integer	2	6
REVOKED	0	8161	0	character	No	2
MVR_PTS	0	8161	0	integer	3	13
CAR_AGE	510	8161	6.2	integer	18	31
URBANICITY	0	8161	0	character	Highly Urban/ Urban	2

Three variables contain incomplete records including 'AGE', 'YOJ', and 'CAR_AGE' with 0.1%, 5.6%, and 6.2% of their data missing. Theoretically each variable would have 8161 total observations as noted in the table. The data types are either integer or numeric and the examples display what the type looks like for easy referencing. A calculation of the unique factors for each variable is included to gauge whether converting to a factor data type would be right for the variable and count the number of unique values to each. These are major concerns.

Minima, quartiles, averages, and maximums were computed to compare the numeric integer variables. Although the order of the variables remains the same as in the previous table, we added a missing values column with the row identifier 'NA' to count the number missing for tracking purposes. We put this together in a table called Data Characteristics. Of course, several of the variables will need to be altered before we can evaluate if the data makes sense in a real-life scenario. These are shown as NA in the table.

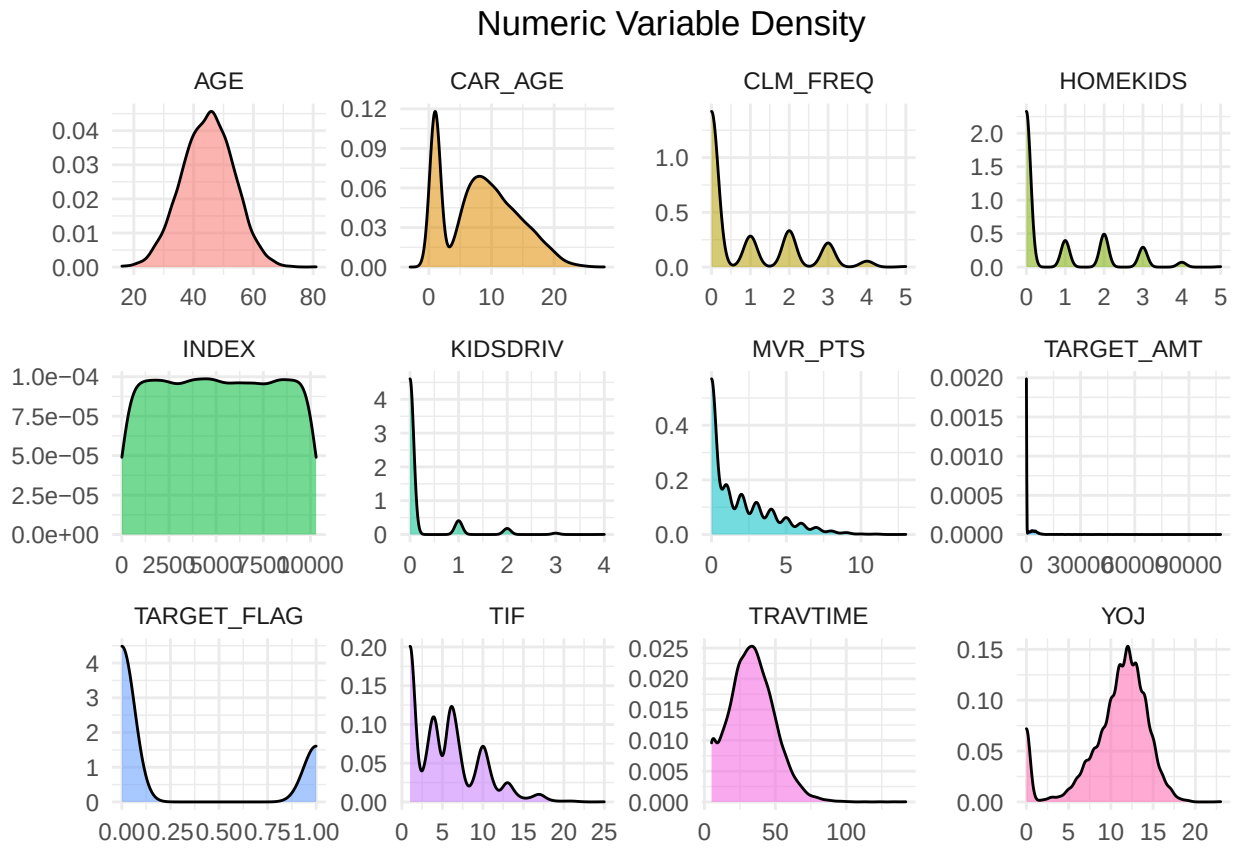
```
tdata.summary.tbl <- summary(tdata)
kbl(t(tdata.summary.tbl), booktabs = T, caption = "Data Characteristics") %>%
  kable_styling(latex_options = c("striped", "scale_down", "hold_position"), full_width = F)
```

Table 5: Data Characteristics

INDEX	Min. : 1	1st Qu.: 2559	Median : 5133	Mean : 5152	3rd Qu.: 7745	Max. :10302	NA
TARGET_FLAG	Min. :0.0000	1st Qu.:0.0000	Median :0.0000	Mean :0.2638	3rd Qu.:1.0000	Max. :1.0000	NA
TARGET_AMT	Min. : 0	1st Qu.: 0	Median : 0	Mean : 1504	3rd Qu.: 1036	Max. :107586	NA
KIDSDRIV	Min. :0.0000	1st Qu.:0.0000	Median :0.0000	Mean :0.1711	3rd Qu.:0.0000	Max. :4.0000	NA
AGE	Min. :16.00	1st Qu.:39.00	Median :45.00	Mean :44.79	3rd Qu.:51.00	Max. :81.00	NA's :6
HOMEKIDS	Min. :0.0000	1st Qu.:0.0000	Median :0.0000	Mean :0.7212	3rd Qu.:1.0000	Max. :5.0000	NA
YOJ	Min. : 0.0	1st Qu.: 9.0	Median :11.0	Mean :10.5	3rd Qu.:13.0	Max. :23.0	NA's :454
INCOME	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
PARENT1	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
HOME_VAL	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
MSTATUS	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
SEX	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
EDUCATION	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
JOB	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
TRAVTIME	Min. : 5.00	1st Qu.: 22.00	Median : 33.00	Mean : 33.49	3rd Qu.: 44.00	Max. :142.00	NA
CAR_USE	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
BLUEBOOK	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
TIF	Min. : 1.000	1st Qu.: 1.000	Median : 4.000	Mean : 5.351	3rd Qu.: 7.000	Max. :25.000	NA
CAR_TYPE	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
RED_CAR	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
OLDCLAIM	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
CLM_FREQ	Min. :0.0000	1st Qu.:0.0000	Median :0.0000	Mean :0.7986	3rd Qu.:2.0000	Max. :5.0000	NA
REVOKED	Length:8161	Class :character	Mode :character	NA	NA	NA	NA
MVR_PTS	Min. : 0.000	1st Qu.: 0.000	Median : 1.000	Mean : 1.696	3rd Qu.: 3.000	Max. :13.000	NA
CAR_AGE	Min. :-3.000	1st Qu.: 1.000	Median : 8.000	Mean : 8.328	3rd Qu.:12.000	Max. :28.000	NA's :510
URBANICITY	Length:8161	Class :character	Mode :character	NA	NA	NA	NA

```
tdata %>%
  select_if(is.numeric) %>%
  gather %>%
  ggplot() +
  facet_wrap(~ key, scales = "free") +
  geom_density(aes(value, color = value, fill = key, alpha = .5)) + theme(axis.title = element_blank(),
```

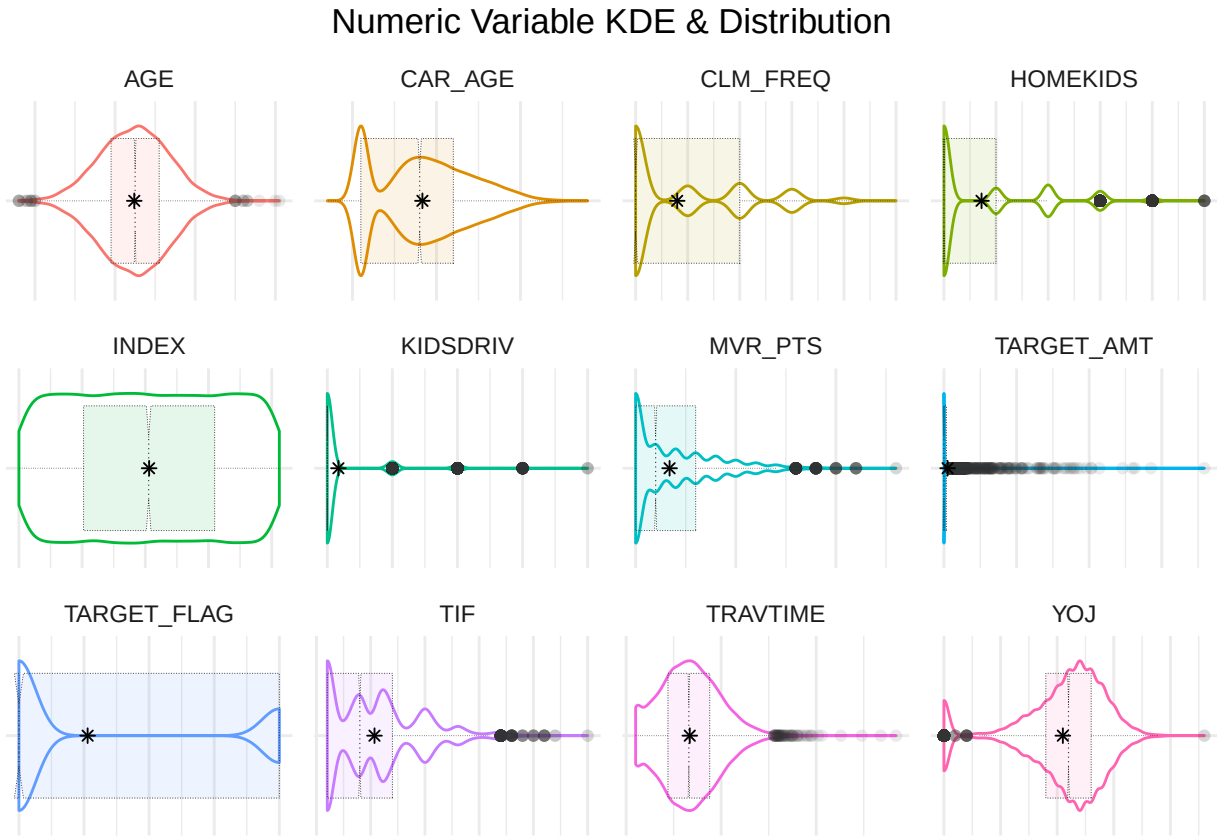
```
## Warning: Removed 970 rows containing non-finite values (stat_density).
```




```
tdata %>%
  select_if(is.numeric) %>%
  gather %>%
  ggplot(aes(value, key)) +
  facet_wrap(~ key, scales = "free") +
  geom_violin(aes(color = key, alpha = 1)) +
  geom_boxplot(aes(fill = key, alpha = .5), notch = TRUE, size = .1, lty = 3) +
  stat_summary(fun.y = mean, geom = "point",
               shape = 8, size = 1.5, color = "#000000") +
  theme(axis.text = element_blank(),
        axis.title = element_blank(),
        legend.position = "none") +
  ggtitle("Numeric Variable KDE & Distribution") +
  theme(plot.title = element_text(hjust = 0.5))
```

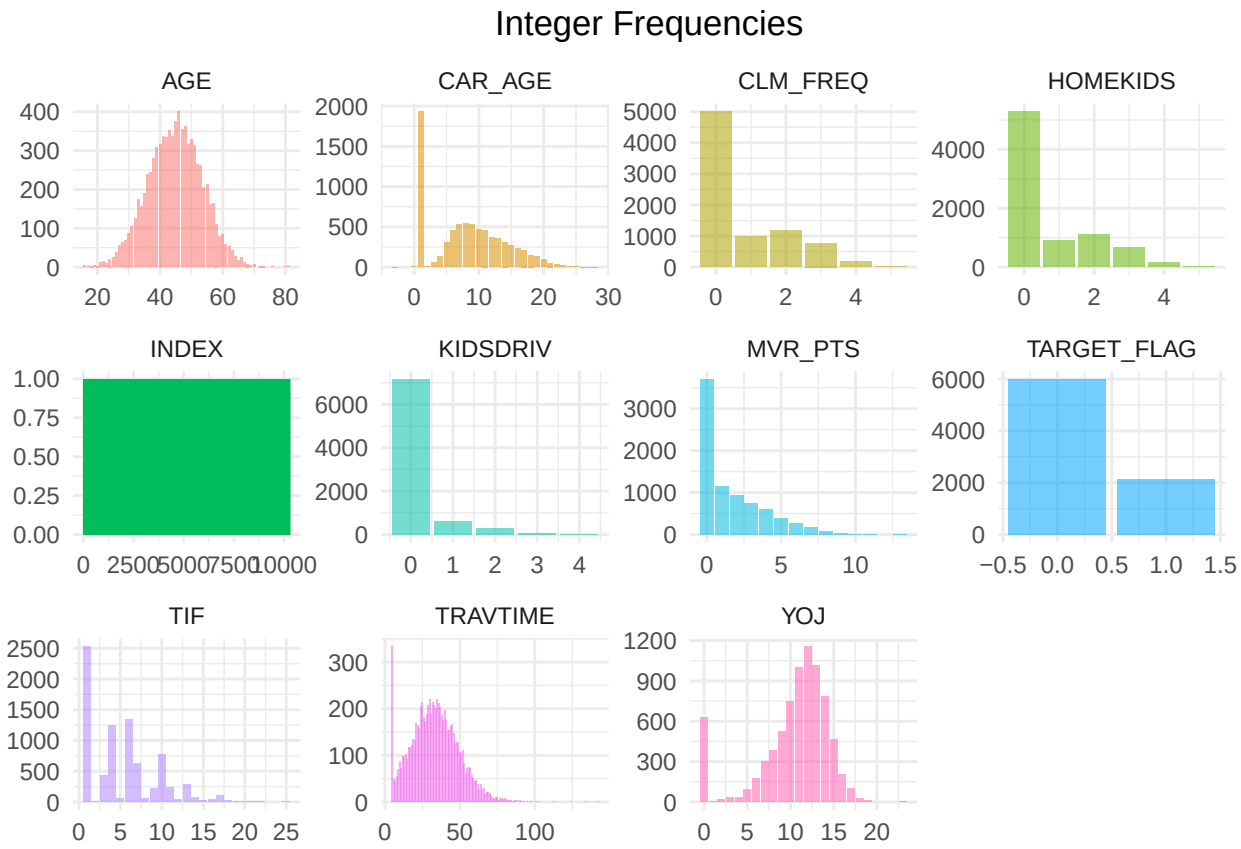
```
## Warning: `fun.y` is deprecated. Use `fun` instead.

## Warning: Removed 970 rows containing non-finite values (stat_ydensity).

## Warning: Removed 970 rows containing non-finite values (stat_boxplot).

## Warning: Removed 970 rows containing non-finite values (stat_summary).

## notch went outside hinges. Try setting notch=FALSE.
```

```
## notch went outside hinges. Try setting notch=FALSE.
## notch went outside hinges. Try setting notch=FALSE.
## notch went outside hinges. Try setting notch=FALSE.
```

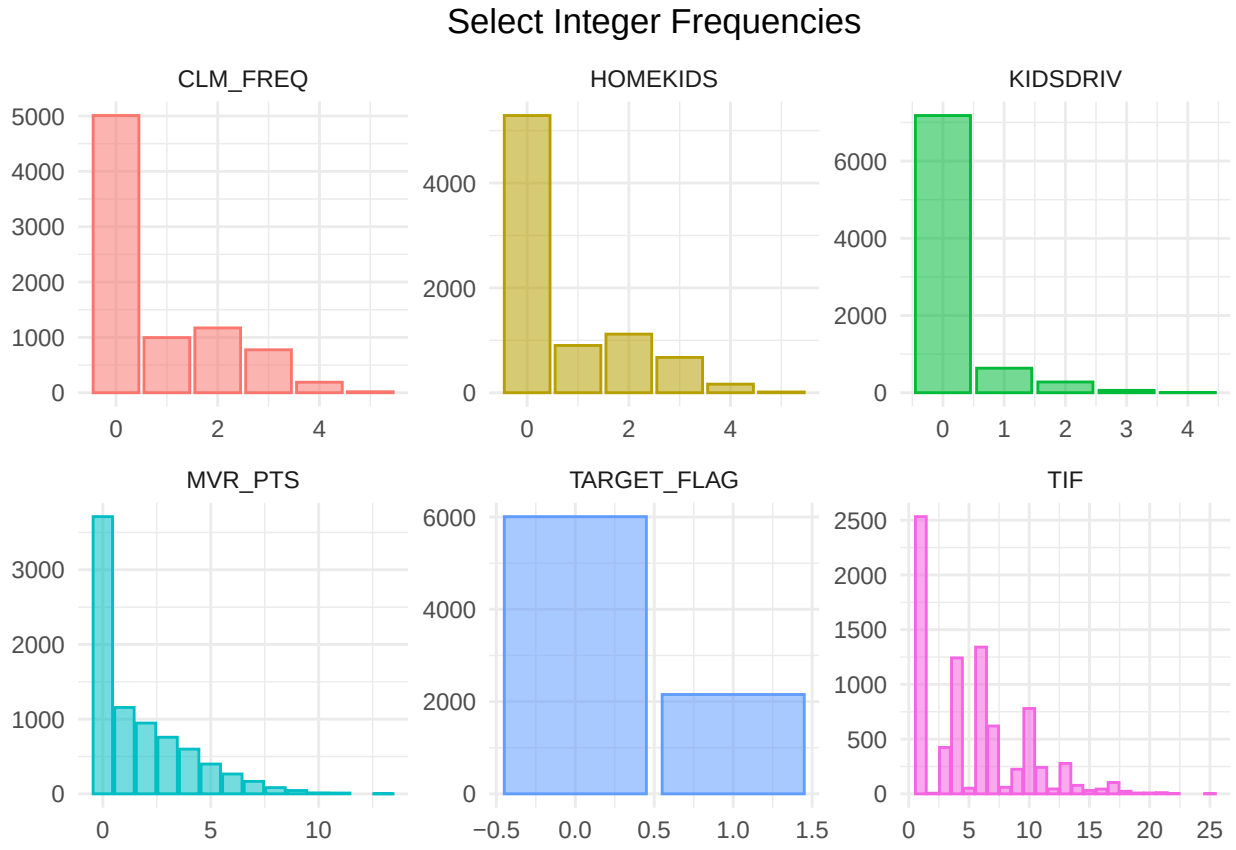



```
tdata %>%
  select_if(is.integer) %>%
  gather() %>%
  filter(value == 0 | 1) %>%
  group_by(key) %>%
  ggplot() +
  facet_wrap(~ key, scales = "free") +
  geom_bar(aes(value, color = value, fill = key, alpha = .5)) + theme(axis.title = element_blank(), leg
```

```
## Warning: Removed 970 rows containing non-finite values (stat_count).
```

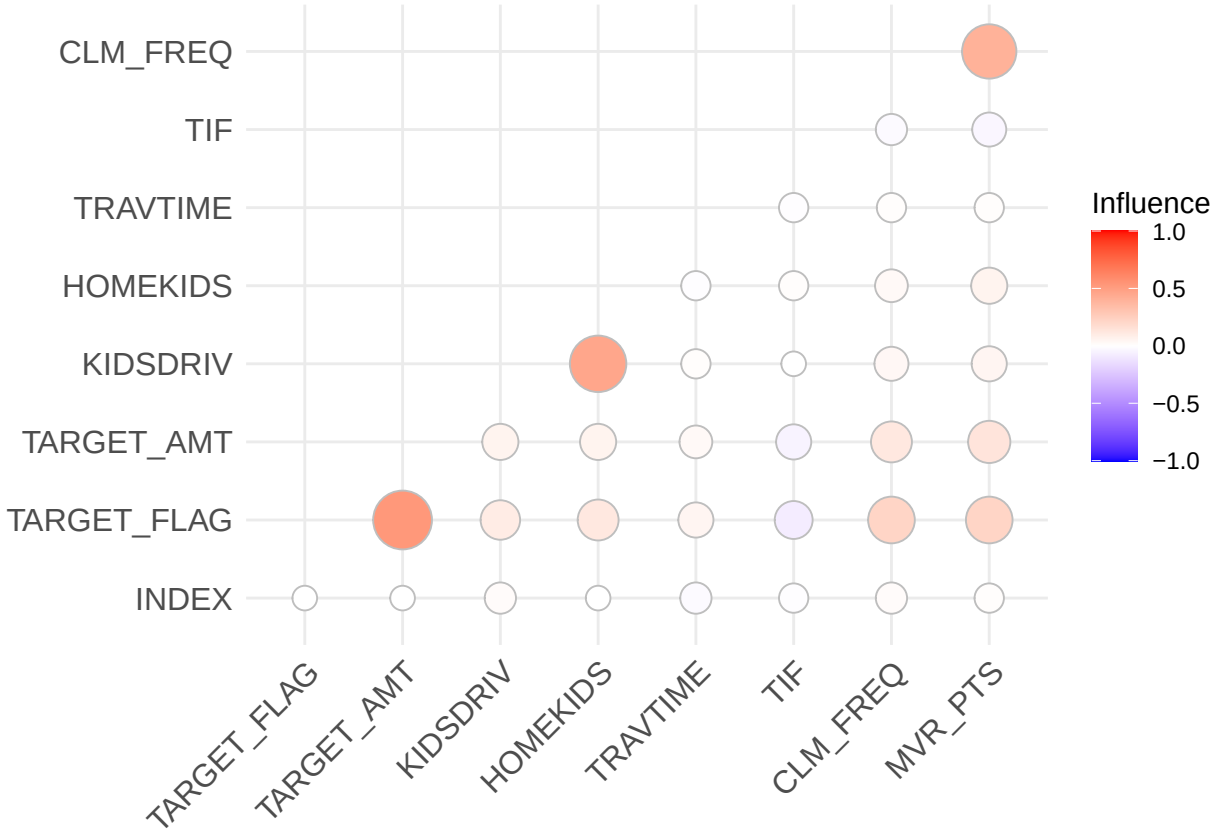



```
tdata %>%
  dplyr::select(TARGET_FLAG, MVR_PTS, CLM_FREQ, HOMEKIDS, KIDSDRIV, TIF) %>%
  gather() %>%
  ggplot(aes(value)) +
  facet_wrap(~ key, scales = "free") +
  geom_bar(aes(value, color = key, fill = key, alpha = .5)) + theme(axis.title = element_blank(), legend
```




```
tdata %>%
  select_if(is.numeric) %>%
  cor() %>%
  ggcorrplot(method = "circle", type="upper",
             ggtheme = ggplot2::theme_minimal, legend.title = "Influence") + coord_flip()
```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```


CLM_FREQ
TIF
TRAVTIME
HOMEKIDS
KIDSDRIV
TARGET_AMT
TARGET_FLAG
INDEX
TARGET_FLAG
TARGET_AMT
KIDSDRIV
HOMEKIDS
TRAVTIME
TIF
CLM_FREQ
MVR_PTS
Influence
1.0
0.5
0.0
−0.5
−1.0

Data Preparation

Describe how you have transformed the data by changing the original variables or creating new variables. If you did transform the data or create new variables, discuss why you did this. Here are some possible transformations. a. Fix missing values (maybe with a Mean or Median value) b. Create flags to suggest if a variable was missing c. Transform data by putting it into buckets d. Mathematical transforms such as log or square root (or use Box-Cox) e. Combine variables (such as ratios or adding or multiplying) to create new variables

```
# Select character variables
chars <- dplyr::select(tdata, where(is.character))
# Use function to extract dollars
to_num <- function(x){
  x <- as.character(x)
  x <- gsub(",", "", x)
  x <- gsub("\\$", "", x)
  as.numeric(x)
}
# Specify those dollar variables
income.values <- to_num(chars$INCOME)
home.values <- to_num(chars$HOME_VAL)
bluebook.values <- to_num(chars$BLUEBOOK)
oldclaim.values <- to_num(chars$OLDCLAIM)
concept_df <- as.data.frame(cbind(income.values,
                    home.values,
                    bluebook.values,
                    oldclaim.values))
income.values.stat <- to_num(chars$INCOME)
home.values.stat <- to_num(chars$HOME_VAL)
bluebook.values.stat <- to_num(chars$BLUEBOOK)
oldclaim.values.stat <- to_num(chars$OLDCLAIM)
# impute median values for missing variables
income.values[is.na(income.values)] <-
  median(income.values, na.rm = TRUE)
home.values[is.na(home.values)] <-
  median(home.values, na.rm = TRUE)
bluebook.values[is.na(bluebook.values)] <-
  median(bluebook.values, na.rm = TRUE)
oldclaim.values[is.na(oldclaim.values)] <-
  median(oldclaim.values, na.rm = TRUE)
# Recombine into data frame
dollar.values <-
  data.frame(cbind(income.values,
                   home.values,
                   bluebook.values,
                   oldclaim.values))
dollar.values.stats <-
  data.frame(cbind(income.values.stat,
                   home.values.stat,
                   bluebook.values.stat,
                   oldclaim.values.stat))
# Join with training data
tdata <- data.frame(cbind(tdata, dollar.values))
# Check the difference
```

```
dollar.values.tbl <- summary(dollar.values)
dollar.values.stats.tbl <- summary(dollar.values.stats)
kbl(dollar.values.tbl, booktabs = T, caption = "Imputed Summary Statistics") %>%
kable_styling(latex_options = c("striped", "hold_position"), full_width = F)
```

Table 6: Imputed Summary Statistics

income.values	home.values	bluebook.values	oldclaim.values
Min. : 0	Min. : 0	Min. : 1500	Min. : 0
1st Qu.: 29707	1st Qu.: 0	1st Qu.: 9280	1st Qu.: 0
Median : 54028	Median :161160	Median :14440	Median : 0
Mean : 61469	Mean :155225	Mean :15710	Mean : 4037
3rd Qu.: 83304	3rd Qu.:233352	3rd Qu.:20850	3rd Qu.: 4636
Max. :367030	Max. :885282	Max. :69740	Max. :57037

```
kbl(dollar.values.stats.tbl, booktabs = T, caption = "Original Summary Statistics") %>%
kable_styling(latex_options = c("striped", "hold_position"), full_width = F)
```

Table 7: Original Summary Statistics

income.values.stat	home.values.stat	bluebook.values.stat	oldclaim.values.stat
Min. : 0	Min. : 0	Min. : 1500	Min. : 0
1st Qu.: 28097	1st Qu.: 0	1st Qu.: 9280	1st Qu.: 0
Median : 54028	Median :161160	Median :14440	Median : 0
Mean : 61898	Mean :154867	Mean :15710	Mean : 4037
3rd Qu.: 85986	3rd Qu.:238724	3rd Qu.:20850	3rd Qu.: 4636
Max. :367030	Max. :885282	Max. :69740	Max. :57037
NA's :445	NA's :464	NA	NA

```
# Covert categorical variables to factors
factors <- tdata %>%
  dplyr::select("PARENT1",
          "MSTATUS",
          "SEX",
          "EDUCATION",
          "JOB",
          "CAR_USE",
          "CAR_TYPE",
          "RED_CAR",
          "REVOKED",
          "URBANICITY")
factors <- data.frame(lapply(factors, function(x) as.factor(x)))
factors <- factors %>%
  rename("parent1" = "PARENT1",
          "mstatus" = "MSTATUS",
          "sex" = "SEX",
          "education" = "EDUCATION",
```

```
         "job" = "JOB",
         "car_use" = "CAR_USE",
         "car_type" = "CAR_TYPE",
         "red_car" = "RED_CAR",
         "revoked" = "REVOKED",
         "urbanicity" = "URBANICITY")
tdata <- cbind(tdata, factors)
```

```
# Exclude unrealistic values
tdata <- tdata %>%
  mutate(car_age = ifelse(CAR_AGE<0, NA, CAR_AGE))
summary(tdata$car_age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    1.00    8.00    8.33   12.00   28.00     511
```

```
summary(tdata$CAR_AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -3.000   1.000   8.000   8.328  12.000  28.000     510
```

```
# Drop INDEX and other unnecessary columns
tdata <- tdata %>%
  dplyr::select("TARGET_FLAG",
         "TARGET_AMT",
         "KIDSDRIV",
         "AGE",
         "HOMEKIDS",
         "YOJ",
         "TRAVTIME",
         "TIF",
         "CLM_FREQ",
         "MVR_PTS",
         "income.values",
         "home.values",
         "bluebook.values",
         "oldclaim.values",
         "parent1",
         "mstatus",
         "sex",
         "education",
         "job",
         "car_use",
         "car_age",
         "car_type",
         "red_car",
         "revoked",
         "urbanicity")
# Check total variables present
length(colnames(tdata))
```

```
## [1] 25
```

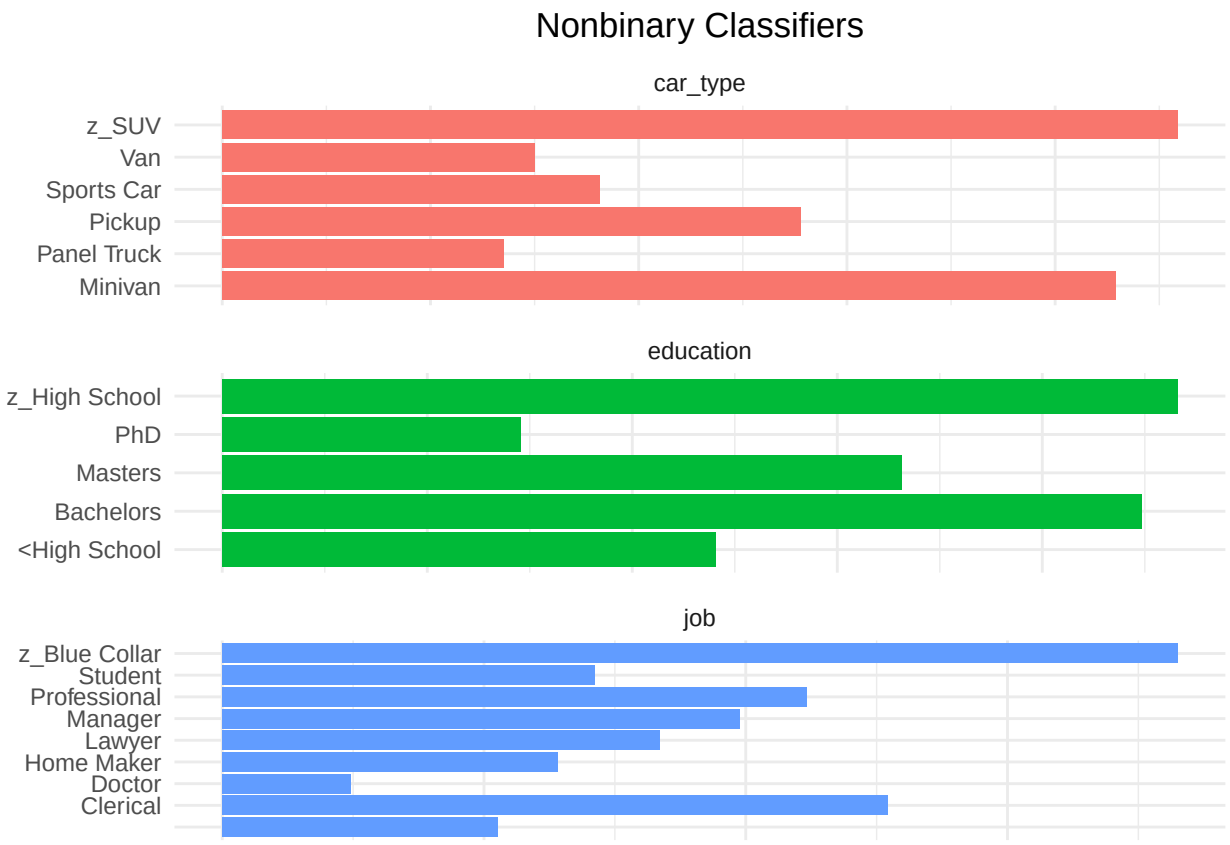
```r
# More imputation
tdata$AGE[is.na(tdata$AGE)] <-
  median(tdata$AGE, na.rm = T)
tdata$YOJ[is.na(tdata$YOJ)] <-
  median(tdata$YOJ, na.rm = T)
tdata$car_age[is.na(tdata$car_age)] <-
  median(tdata$car_age, na.rm = T)
sum(is.na(tdata))
```

```
## [1] 0
```

```r
tdata %>%
  dplyr::select(is.factor) %>%
  dplyr::select("car_type", "education", "job") %>%
  gather() %>%
  ggplot(aes(value)) +
  facet_wrap(~ key, nrow = 3, scales = "free") +
  geom_bar(aes(, fill = key )) + theme(axis.title = element_blank(), axis.text.x = element_blank(), leg
```

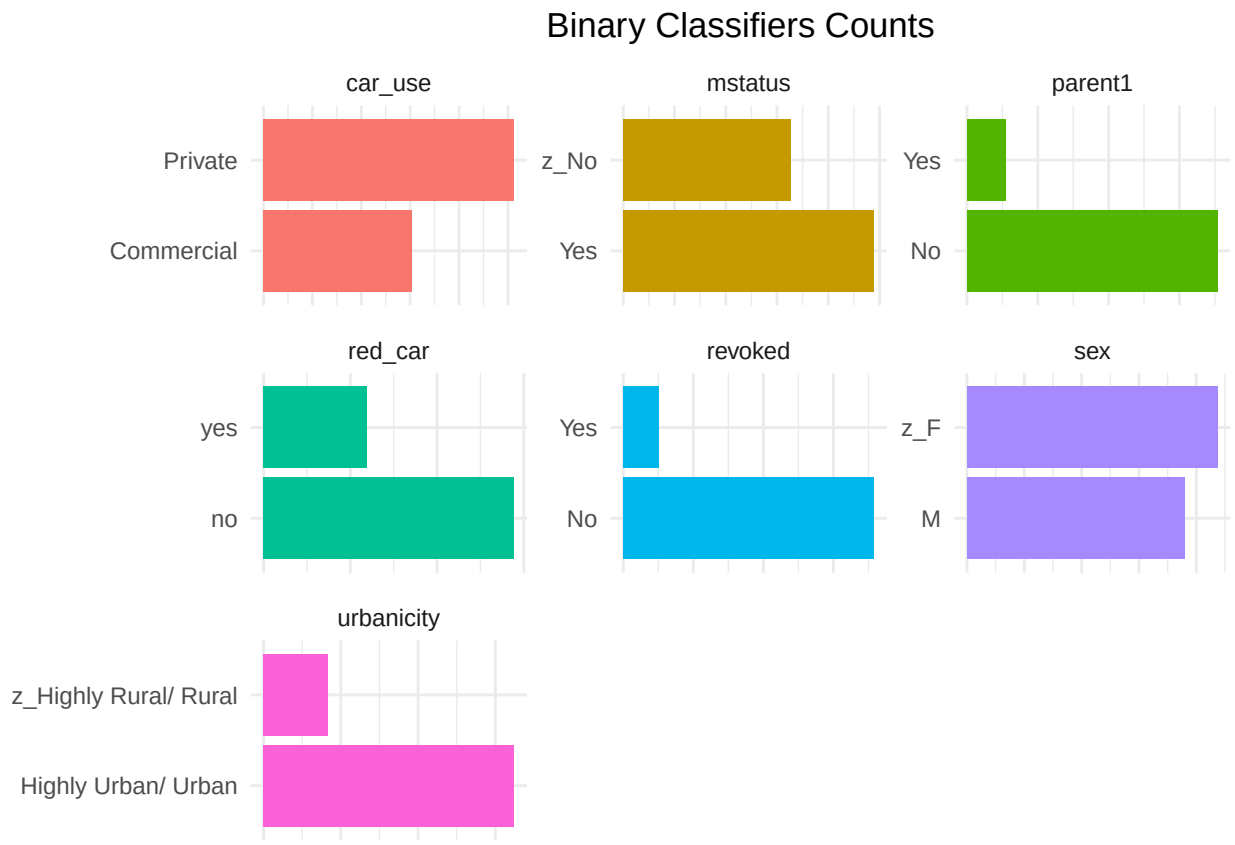
```
## Warning: Predicate functions must be wrapped in `where()`.
##
##   # Bad
##   data %>% select(is.factor)
##
##   # Good
##   data %>% select(where(is.factor))
##
## i Please update your code.
## This message is displayed once per session.
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

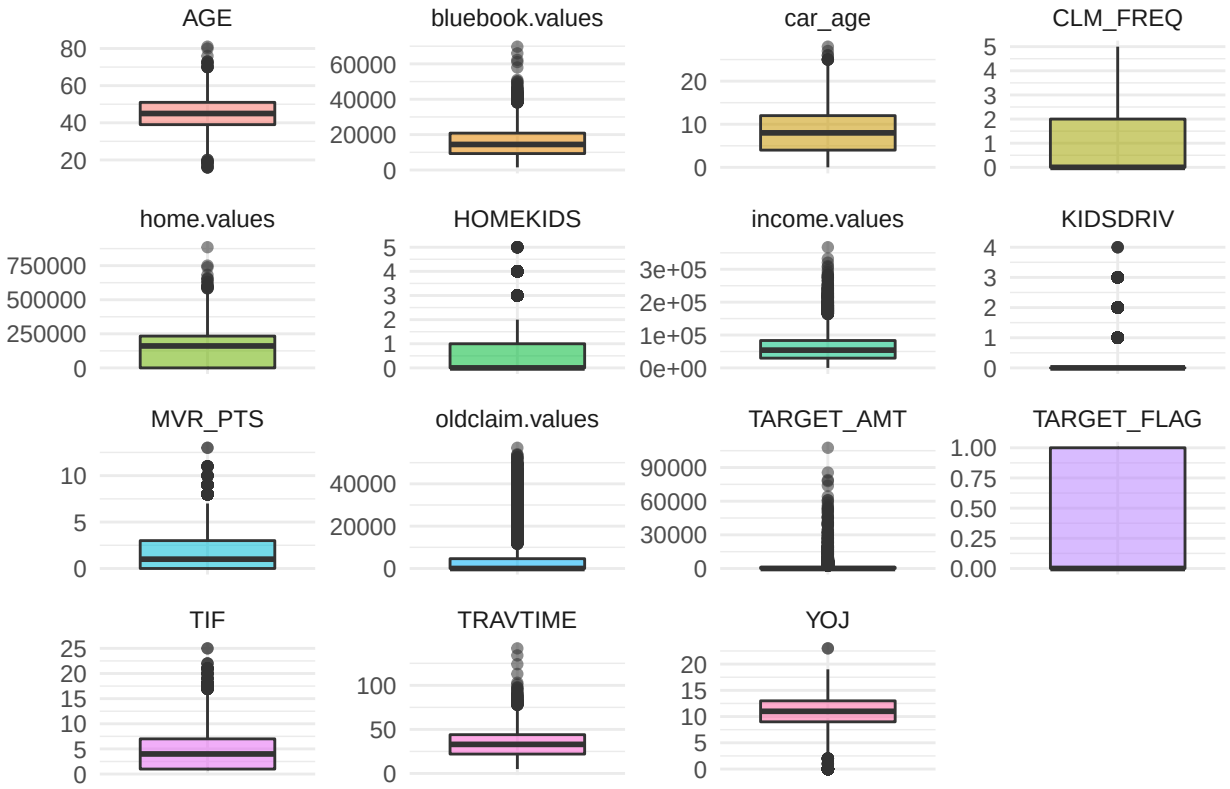



```
tdata %>%
  dplyr::select(is.factor) %>%
  dplyr::select("car_use",
         "mstatus",
         "parent1",
         "red_car",
         "revoked",
         "sex",
         "urbanicity") %>%
  gather() %>%
  ggplot(aes(value)) +
  facet_wrap(~ key, scales = "free") +
  geom_bar(aes(, fill = key )) + theme(axis.title = element_blank(), axis.text.x = element_blank(), leg
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```



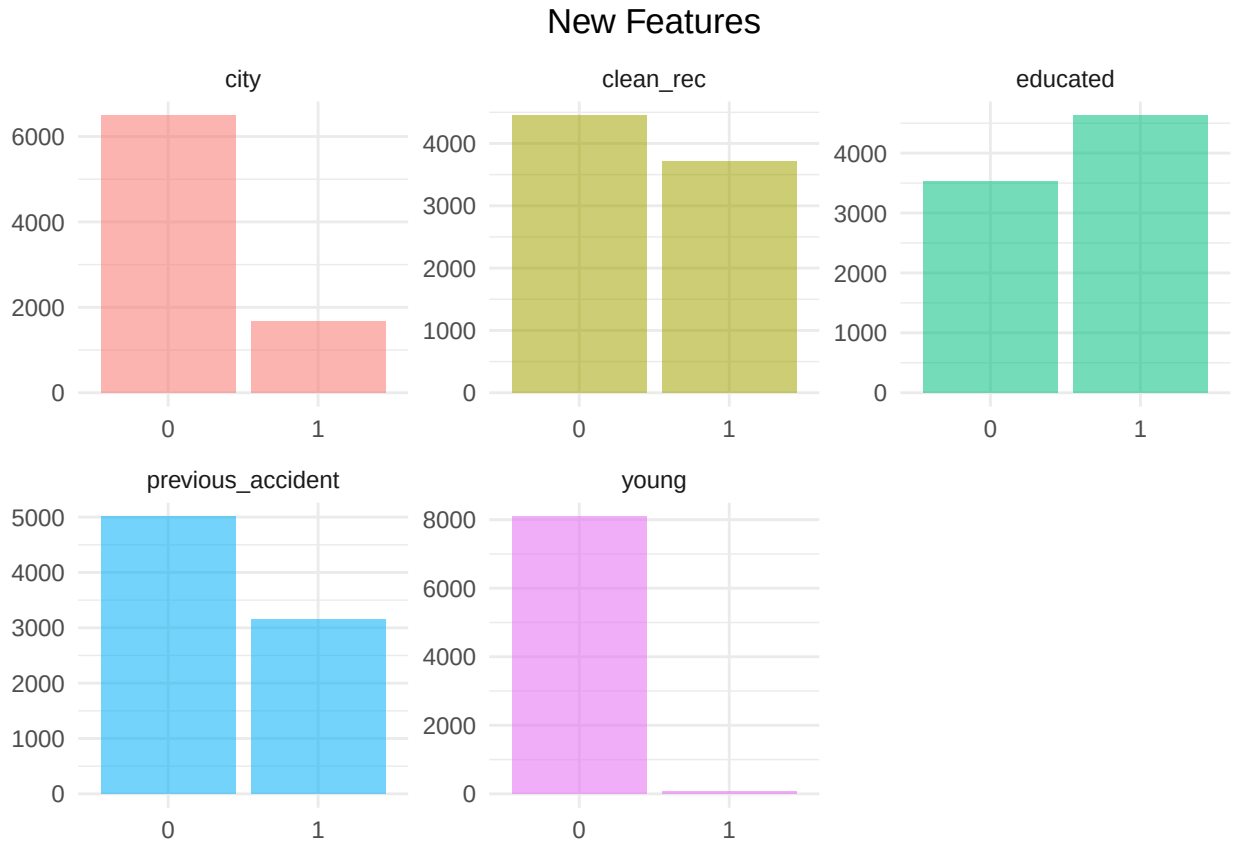

```
tdata %>%
  select_if(is.numeric) %>%
  gather() %>%
  ggplot(aes(key)) +
  facet_wrap(~ key, scales = "free") +
  geom_boxplot(aes(key, value, fill = key, alpha = .5)) + theme(axis.title = element_blank(), axis.text
```




```
# New features
tdata <- tdata %>%
  mutate(city = ifelse(urbanicity == "Highly Urban/ Urban", 0, 1)) %>%
  mutate(young = ifelse(AGE < 25, 1, 0)) %>%
  mutate(clean_rec = ifelse(MVR_PTS == 0, 1, 0)) %>%
  mutate(previous_accident = ifelse(CLM_FREQ == 0 & oldclaim.values == 0, 0, 1)) %>%
  mutate(educated = ifelse(education %in% c("Bachelors", "Masters", "PhD"), 1, 0)) %>%
  mutate(avg_claim = ifelse(CLM_FREQ > 0, oldclaim.values/CLM_FREQ, 0))
```

```
# Convert to factors
tdata$city <- as.factor(tdata$city)
tdata$young <- as.factor(tdata$young)
tdata$clean_rec <- as.factor(tdata$clean_rec)
tdata$previous_accident <- as.factor(tdata$previous_accident)
tdata$educated <- as.factor(tdata$educated)
```

```
tdata[26:31] %>%
  select_if(is.factor) %>%
  gather() %>%
  ggplot(aes(value)) +
  facet_wrap(~key, scales = "free") +
  geom_bar(aes(fill = key, alpha = .5)) + theme(legend.position = "none", axis.title = element_blank())
```




```
# Produce recommended transformations
bestNorms <- tdata[1:11,1:16]
df <- tdata %>%
  select_if(is.numeric)
for (i in colnames(df)) {
  bestNorms[[i]] <- bestNormalize(df[[i]],
                                  allow_orderNorm = FALSE,
                                  out_of_sample =FALSE)
}
```

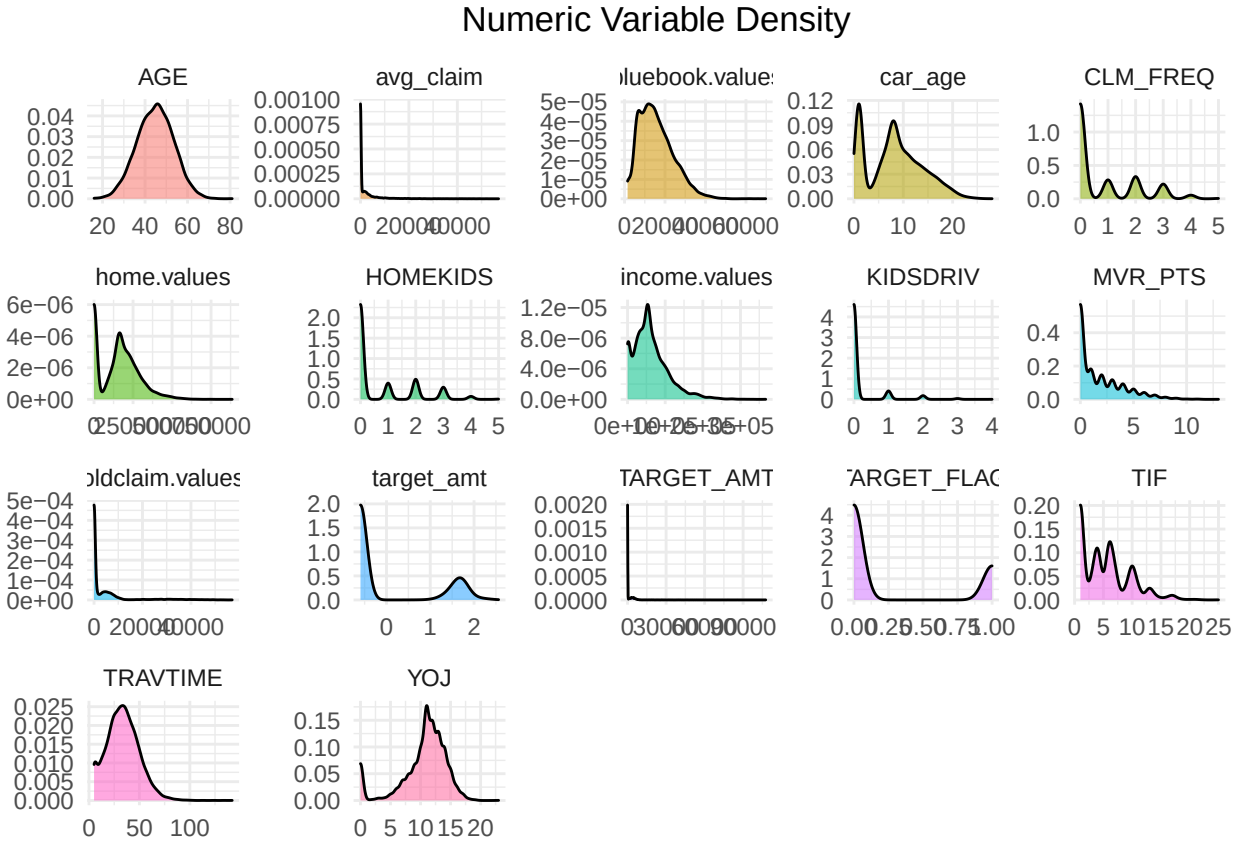
```
# Continue focusing on realistic values
accident_costs <- tdata$TARGET_AMT[tdata$TARGET_AMT>.0]
```

```
# Focus on selected variables
bestNorms$target_amt$chosen_transform
```

```
## NULL
```

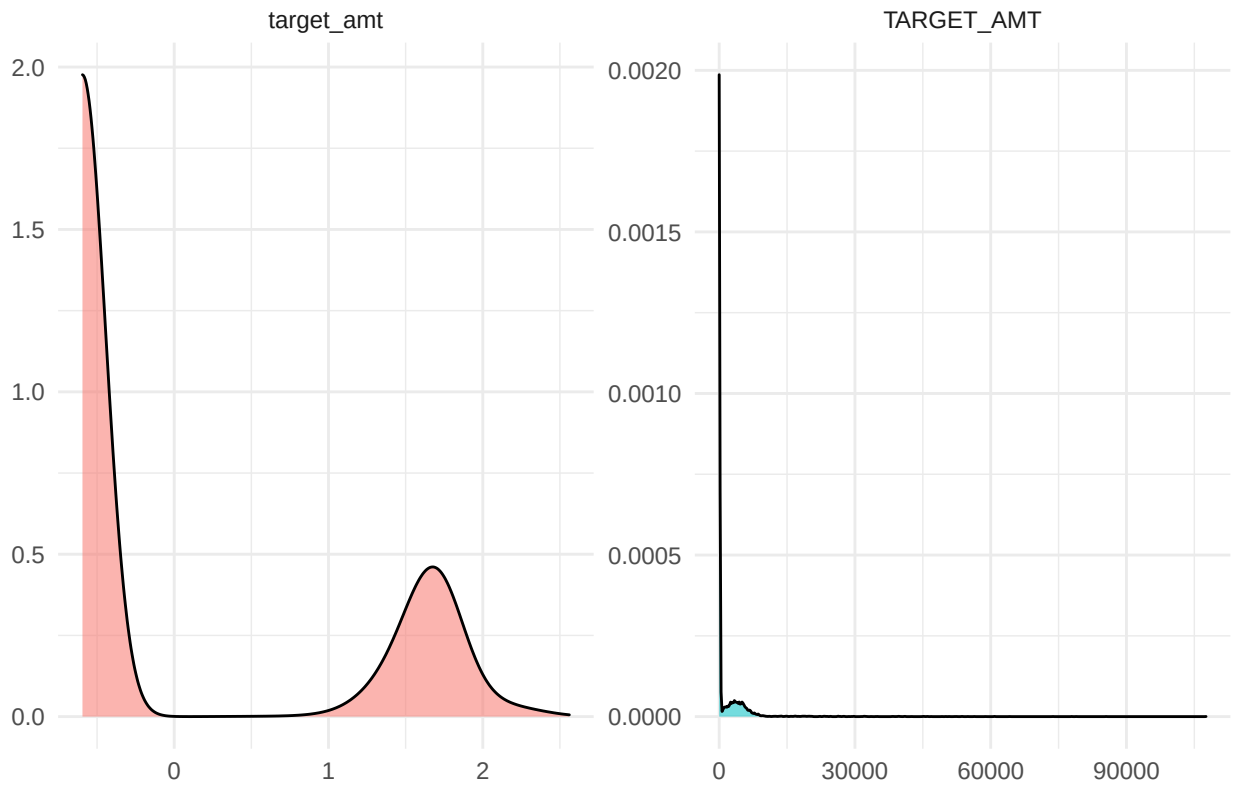
```
tdata$target_amt <- scale(log(tdata$TARGET_AMT + 1))
tdata %>%
  dplyr::select(where(is.numeric)) %>%
  gather %>%
  ggplot() +
  facet_wrap(~ key, scales = "free") +
  geom_density(aes(value, color = value, fill = key, alpha = .5)) + theme(axis.title = element_blank(),
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```




```
tdata %>%
  dplyr::select(where(is.numeric)) %>%
  dplyr::select("TARGET_AMT","target_amt") %>%
  gather %>%
  ggplot() +
  facet_wrap(~ key, scales = "free") +
  geom_density(aes(value, color = value, fill = key, alpha = .5)) + theme(axis.title = element_blank(),
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```




```
# Split 70-30 training test
set.seed(1102)
tindex <- createDataPartition(tdata$TARGET_FLAG, p = .7, list = FALSE, times = 1)
train <- tdata[tindex,]
test <- tdata[-tindex,]
rindex <- tdata %>%
  filter(TARGET_FLAG == 1)
reg.tindex <- createDataPartition(rindex$TARGET_AMT, p = .7, list = FALSE, times = 1)
reg.train <- rindex[reg.tindex,]
reg.test <- rindex[-reg.tindex,]
```

Model Building

Using the training data set, build at least two different multiple linear regression models and three different binary logistic regression models, using different variables (or the same variables with different transformations). You may select the variables manually, use an approach such as Forward or Stepwise, use a different approach such as trees, or use a combination of techniques. Describe the techniques you used. If you manually selected a variable for inclusion into the model or exclusion into the model, indicate why this was done.

Discuss the coefficients in the models, do they make sense? For example, if a person has a lot of traffic tickets, you would reasonably expect that person to have more car crashes. If the coefficient is negative (suggesting that the person is a safer driver), then that needs to be discussed. Are you keeping the model even though it is counter intuitive? Why? The boss needs to know.

```
model1 <- glm(TARGET_FLAG ~ previous_accident,
              family = binomial(link = "logit"), train)
summary(model1)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ previous_accident, family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0069  -0.6351  -0.6351   1.3581   1.8441
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -1.49869    0.04369  -34.30   <2e-16 ***
## previous_accident1  1.08338    0.06167   17.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6613.0  on 5712  degrees of freedom
## Residual deviance: 6297.9  on 5711  degrees of freedom
## AIC: 6301.9
##
## Number of Fisher Scoring iterations: 4
```

```
model2 <- glm(TARGET_FLAG ~ previous_accident +
                city + young + clean_rec +
                educated, family = binomial(link = "logit"), train)
summary(model2)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ previous_accident + city + young +
##     clean_rec + educated, family = binomial(link = "logit"),
##     data = train)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8218  -0.8464  -0.5816   1.1001   2.6595
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -0.52339    0.06962  -7.518 5.57e-14 ***
## previous_accident1  0.70805    0.06737  10.510  < 2e-16 ***
## city1              -1.81562    0.12542 -14.477  < 2e-16 ***
## young1              1.26389    0.29142   4.337 1.44e-05 ***
## clean_rec1         -0.31886    0.06865  -4.645 3.41e-06 ***
## educated1          -0.84903    0.06517 -13.027  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6613  on 5712  degrees of freedom
## Residual deviance: 5859  on 5707  degrees of freedom
## AIC: 5871
##
## Number of Fisher Scoring iterations: 5
```

```
model3 <- glm(TARGET_FLAG ~ previous_accident +
                city + mstatus + income.values +
                sex + car_use + educated + KIDSDRIV +
                revoked, family = binomial(link = "logit"),
              train)
summary(model3)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ previous_accident + city + mstatus +
##     income.values + sex + car_use + educated + KIDSDRIV + revoked,
##     family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0469  -0.7604  -0.4518   0.7784   2.8387
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -5.071e-01  9.509e-02  -5.333 9.66e-08 ***
## previous_accident1  6.289e-01  6.806e-02   9.240  < 2e-16 ***
## city1              -2.106e+00  1.306e-01 -16.128  < 2e-16 ***
## mstatusz_No         8.075e-01  6.805e-02  11.866  < 2e-16 ***
## income.values      -8.460e-06  9.462e-07  -8.941  < 2e-16 ***
## sexz_F              3.020e-01  7.077e-02   4.268 1.98e-05 ***
## car_usePrivate     -7.490e-01  7.226e-02 -10.366  < 2e-16 ***
## educated1          -5.469e-01  7.762e-02  -7.046 1.84e-12 ***
## KIDSDRIV            4.418e-01  6.038e-02   7.318 2.52e-13 ***
## revokedYes          7.863e-01  9.258e-02   8.494  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6613.0  on 5712  degrees of freedom
## Residual deviance: 5447.3  on 5703  degrees of freedom
## AIC: 5467.3
##
## Number of Fisher Scoring iterations: 5
```

```
model4 <- lm(target_amt ~ ., train)
summary(model4)
```

```
##
## Call:
## lm(formula = target_amt ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.16361 -0.00472  0.00052  0.00713  0.15233
##
## Coefficients: (2 not defined because of singularities)
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -5.890e-01  1.112e-02 -52.981   <2e-16 ***
## TARGET_FLAG                    2.110e+00  2.952e-03 714.825   <2e-16 ***
## TARGET_AMT                     2.538e-05  2.834e-07  89.561   <2e-16 ***
## KIDSDRIV                      -1.115e-03  2.188e-03  -0.510   0.6102
## AGE                            3.253e-05  1.400e-04   0.232   0.8163
## HOMEKIDS                       2.005e-04  1.276e-03   0.157   0.8752
## YOJ                           -4.639e-04  2.924e-04  -1.586   0.1127
## TRAVTIME                       2.985e-05  6.247e-05   0.478   0.6327
## TIF                           -7.299e-05  2.354e-04  -0.310   0.7565
## CLM_FREQ                      -5.426e-03  2.161e-03  -2.511   0.0121 *
## MVR_PTS                        4.976e-04  7.063e-04   0.704   0.4812
## income.values                 -2.435e-08  3.468e-08  -0.702   0.4826
## home.values                   -9.550e-09  1.135e-08  -0.841   0.4001
## bluebook.values                1.247e-07  1.661e-07   0.750   0.4530
## oldclaim.values                5.064e-07  3.252e-07   1.557   0.1195
## parent1Yes                    -4.423e-05  3.880e-03  -0.011   0.9909
## mstatusz_No                   -4.518e-05  2.811e-03  -0.016   0.9872
## sexz_F                         2.346e-04  3.573e-03   0.066   0.9477
## educationBachelors            -3.203e-03  3.962e-03  -0.808   0.4188
## educationMasters              -3.571e-04  5.732e-03  -0.062   0.9503
## educationPhD                  -5.652e-04  6.885e-03  -0.082   0.9346
## educationz_High School         2.997e-03  3.306e-03   0.906   0.3647
## jobClerical                   -7.368e-03  6.615e-03  -1.114   0.2654
## jobDoctor                      2.577e-03  7.818e-03   0.330   0.7417
## jobHome Maker                 -1.657e-02  7.070e-03  -2.344   0.0191 *
## jobLawyer                     -6.127e-03  5.702e-03  -1.074   0.2827
## jobManager                    -1.398e-03  5.581e-03  -0.251   0.8022
## jobProfessional               -2.482e-03  5.923e-03  -0.419   0.6752
## jobStudent                    -7.713e-03  7.278e-03  -1.060   0.2893
## jobz_Blue Collar              -9.370e-03  6.238e-03  -1.502   0.1332
## car_usePrivate                 1.575e-03  3.182e-03   0.495   0.6206
## car_age                        1.995e-04  2.468e-04   0.808   0.4189
```

```
## car_typePanel Truck              1.576e-03  5.386e-03   0.293   0.7698
## car_typePickup                   1.829e-03  3.311e-03   0.552   0.5807
## car_typeSports Car               2.875e-04  4.172e-03   0.069   0.9451
## car_typeVan                      4.292e-04  4.067e-03   0.106   0.9159
## car_typez_SUV                    2.666e-03  3.480e-03   0.766   0.4437
## red_caryes                       3.818e-03  2.895e-03   1.319   0.1873
## revokedYes                      -1.898e-05  3.417e-03  -0.006   0.9956
## urbanicityz_Highly Rural/ Rural  5.741e-04  2.821e-03   0.203   0.8388
## city1                                   NA         NA      NA       NA
## young1                           5.431e-03  1.033e-02   0.526   0.5989
## clean_rec1                      -1.693e-03  2.809e-03  -0.603   0.5467
## previous_accident1               7.992e-03  5.323e-03   1.501   0.1333
## educated1                               NA         NA      NA       NA
## avg_claim                       -6.315e-07  4.650e-07  -1.358   0.1745
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0733 on 5669 degrees of freedom
## Multiple R-squared:  0.9947, Adjusted R-squared:  0.9946
## F-statistic: 2.461e+04 on 43 and 5669 DF,  p-value: < 2.2e-16
```

```
model5 <- lm(target_amt ~ income.values +
               home.values + bluebook.values +
               oldclaim.values + avg_claim,
             train)
summary(model5)
```

```
##
## Call:
## lm(formula = target_amt ~ income.values + home.values + bluebook.values +
##     oldclaim.values + avg_claim, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.617 -0.655 -0.472  1.050  2.803
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.676e-01  3.014e-02   8.877  < 2e-16 ***
## income.values   -7.328e-07  3.502e-07  -2.092 0.036450 *
## home.values     -1.224e-06  1.218e-07 -10.055  < 2e-16 ***
## bluebook.values -5.320e-06  1.678e-06  -3.170 0.001532 **
## oldclaim.values  1.224e-05  3.347e-06   3.656 0.000258 ***
## avg_claim        2.875e-06  4.850e-06   0.593 0.553302
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9728 on 5707 degrees of freedom
## Multiple R-squared:  0.05508,    Adjusted R-squared:  0.05425
## F-statistic: 66.53 on 5 and 5707 DF,  p-value: < 2.2e-16
```

```
model6 <- lm(target_amt ~ . -TARGET_AMT -TARGET_FLAG, train)
pm <- stepAIC(model6, trace = F, direction = "both")
summary(pm)
```

```
##
## Call:
## lm(formula = target_amt ~ KIDSDRIV + HOMEKIDS + TRAVTIME + TIF +
##     CLM_FREQ + MVR_PTS + income.values + home.values + bluebook.values +
##     oldclaim.values + parent1 + mstatus + education + job + car_use +
##     car_type + revoked + urbanicity + young + avg_claim, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9709 -0.6370 -0.2437  0.6152  2.8975
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -5.839e-02  1.024e-01  -0.570 0.568598
## KIDSDRIV                         1.212e-01  2.589e-02   4.680 2.94e-06 ***
## HOMEKIDS                         2.363e-02  1.400e-02   1.687 0.091643 .
## TRAVTIME                         4.161e-03  7.497e-04   5.550 2.99e-08 ***
## TIF                             -1.612e-02  2.822e-03  -5.711 1.18e-08 ***
## CLM_FREQ                         9.387e-02  1.388e-02   6.764 1.48e-11 ***
## MVR_PTS                          4.286e-02  6.062e-03   7.070 1.74e-12 ***
## income.values                   -7.632e-07  4.154e-07  -1.837 0.066250 .
## home.values                     -4.547e-07  1.360e-07  -3.343 0.000835 ***
## bluebook.values                 -5.873e-06  1.803e-06  -3.257 0.001133 **
## oldclaim.values                 -1.847e-05  3.727e-06  -4.957 7.39e-07 ***
## parent1Yes                       1.443e-01  4.650e-02   3.104 0.001920 **
## mstatusz_No                      1.861e-01  3.358e-02   5.540 3.15e-08 ***
## educationBachelors              -1.306e-01  4.519e-02  -2.890 0.003863 **
## educationMasters                -9.238e-02  6.260e-02  -1.476 0.140030
## educationPhD                    -1.123e-01  7.774e-02  -1.445 0.148627
## educationz_High School           5.226e-03  3.969e-02   0.132 0.895246
## jobClerical                      1.916e-01  7.951e-02   2.410 0.015982 *
## jobDoctor                       -5.598e-02  9.397e-02  -0.596 0.551349
## jobHome Maker                    1.420e-01  8.276e-02   1.716 0.086256 .
## jobLawyer                        5.229e-02  6.852e-02   0.763 0.445413
## jobManager                      -1.522e-01  6.708e-02  -2.268 0.023346 *
## jobProfessional                  8.451e-02  7.125e-02   1.186 0.235625
## jobStudent                       1.792e-01  8.604e-02   2.083 0.037315 *
## jobz_Blue Collar                 1.803e-01  7.503e-02   2.403 0.016312 *
## car_usePrivate                  -2.224e-01  3.818e-02  -5.825 6.03e-09 ***
## car_typePanel Truck              2.003e-01  6.092e-02   3.288 0.001015 **
## car_typePickup                   1.682e-01  3.978e-02   4.228 2.40e-05 ***
## car_typeSports Car               3.131e-01  4.246e-02   7.374 1.90e-13 ***
## car_typeVan                      1.931e-01  4.732e-02   4.082 4.53e-05 ***
## car_typez_SUV                    2.248e-01  3.274e-02   6.867 7.24e-12 ***
## revokedYes                       3.823e-01  4.039e-02   9.465  < 2e-16 ***
## urbanicityz_Highly Rural/ Rural -6.588e-01  3.253e-02 -20.254  < 2e-16 ***
## young1                           3.380e-01  1.197e-01   2.824 0.004759 **
## avg_claim                        1.749e-05  4.780e-06   3.659 0.000255 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8824 on 5678 degrees of freedom
```

```
## Multiple R-squared:  0.2264, Adjusted R-squared:  0.2218
## F-statistic: 48.89 on 34 and 5678 DF,  p-value: < 2.2e-16
```

Model Selection

Decide on the criteria for selecting the best multiple linear regression model and the best binary logistic regression model. Will you select models with slightly worse performance if it makes more sense or is more parsimonious? Discuss why you selected your models. For the multiple linear regression model, will you use a metric such as Adjusted R2 , RMSE, etc.? Be sure to explain how you can make inferences from the model, discuss multi-collinearity issues (if any), and discuss other relevant model output. Using the training data set, evaluate the multiple linear regression model based on (a) mean squared error, (b) R2 , (c) F-statistic, and (d) residual plots. For the binary logistic regression model, will you use a metric such as log likelihood, AIC, ROC curve, etc.? Using the training data set, evaluate the binary logistic regression model based on (a) accuracy, (b) classification error rate, (c) precision, (d) sensitivity, (e) specificity, (f) F1 score, (g) AUC, and (h) confusion matrix. Make predictions using the evaluation data set.

```
# Calculate predicted values
# Classifier Model
mod1.pred <- predict.glm(model1, test)
mod2.pred <- predict.glm(model2, test)
mod3.pred <- predict.glm(model3, test)
# Regression Model
mod4.pred <- predict(model4, test, interval = "prediction")
```

```
## Warning in predict.lm(model4, test, interval = "prediction"): prediction from a
## rank-deficient fit may be misleading
```

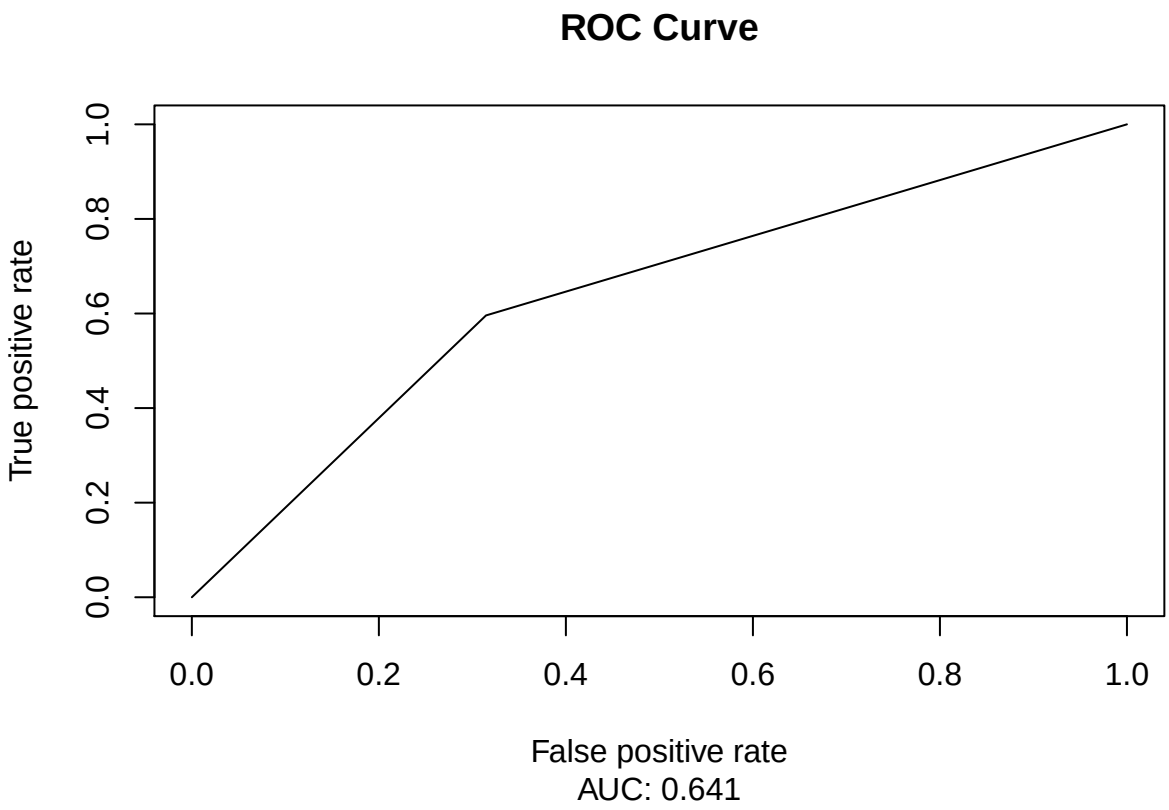
```
mod5.pred <- predict(model5, test, interval = "prediction")
mod6.pred <- predict(model6, test, interval = "prediction")
```

```
## Warning in predict.lm(model6, test, interval = "prediction"): prediction from a
## rank-deficient fit may be misleading
```

```
modstat <- function(model, test, target = "TARGET_FLAG", threshold = 0.5){
  test$new <- ifelse(predict.glm(model, test, "response") >= threshold, 1, 0)
  cm <- confusionMatrix(factor(test$new), factor(test[[target]]), "1")
  df <- data.frame(obs = test$TARGET_FLAG, predicted = test$new, probs = predict(model, test))
  Pscores <- prediction(df$probs, df$obs)
  AUC <- performance(Pscores, measure = "auc")@y.values[[1]]
  pscores <- performance(Pscores, "tpr", "fpr")
  plot(pscores,main="ROC Curve", sub = paste0("AUC: ", round(AUC, 3)))
  results <- paste(cat("F1 = ", cm$byClass[7], " "), cm)
  return(results)
}
```

```
modstat(model1, test)
```

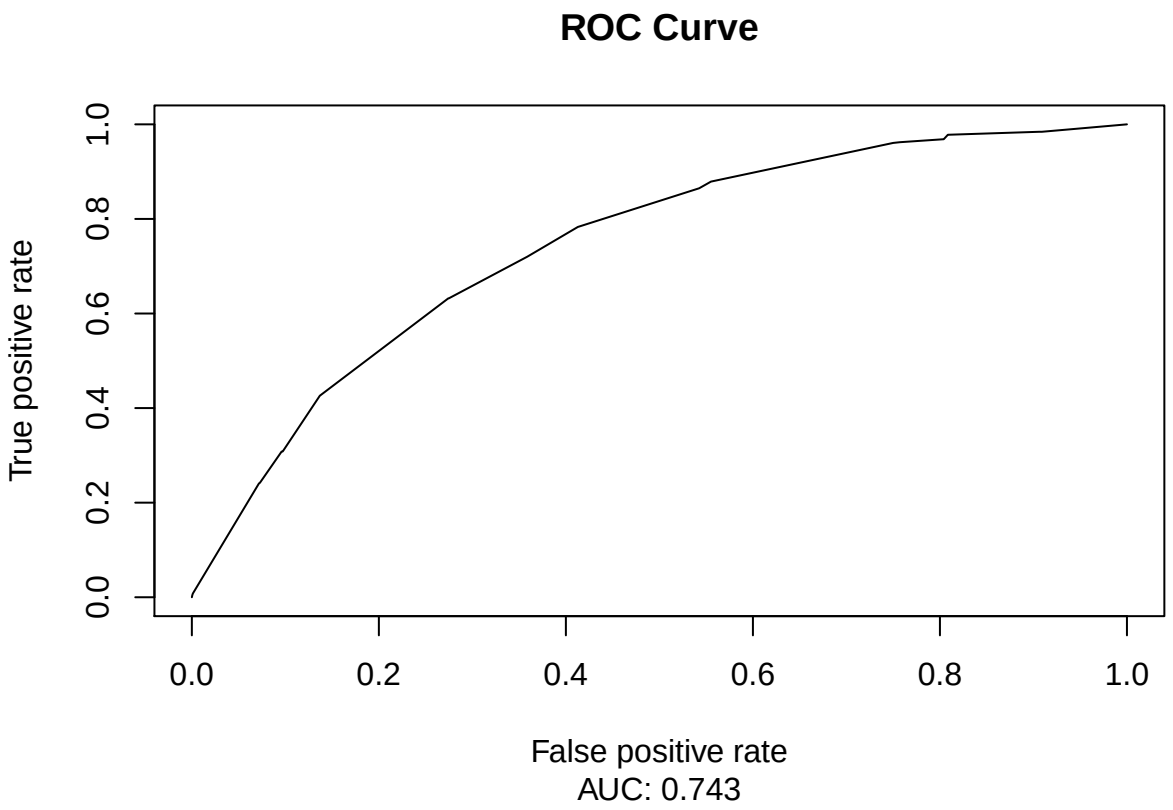
```
## Warning in confusionMatrix.default(factor(test$new), factor(test[[target]]), :
## Levels are not in the same order for reference and data. Refactoring data to
## match.
```




```
## F1 =  NA
```

```
## [1] " 1"
## [2] " c(1812, 0, 636, 0)"
## [3] " c(Accuracy = 0.740196078431373, Kappa = 0, AccuracyLower = 0.722339505019352, AccuracyUpper = 
## [4] " c(Sensitivity = 0, Specificity = 1, ‘Pos Pred Value‘ = NaN, ‘Neg Pred Value‘ = 0.7401960784313
## [5] " sens_spec"
## [6] " list()"
```

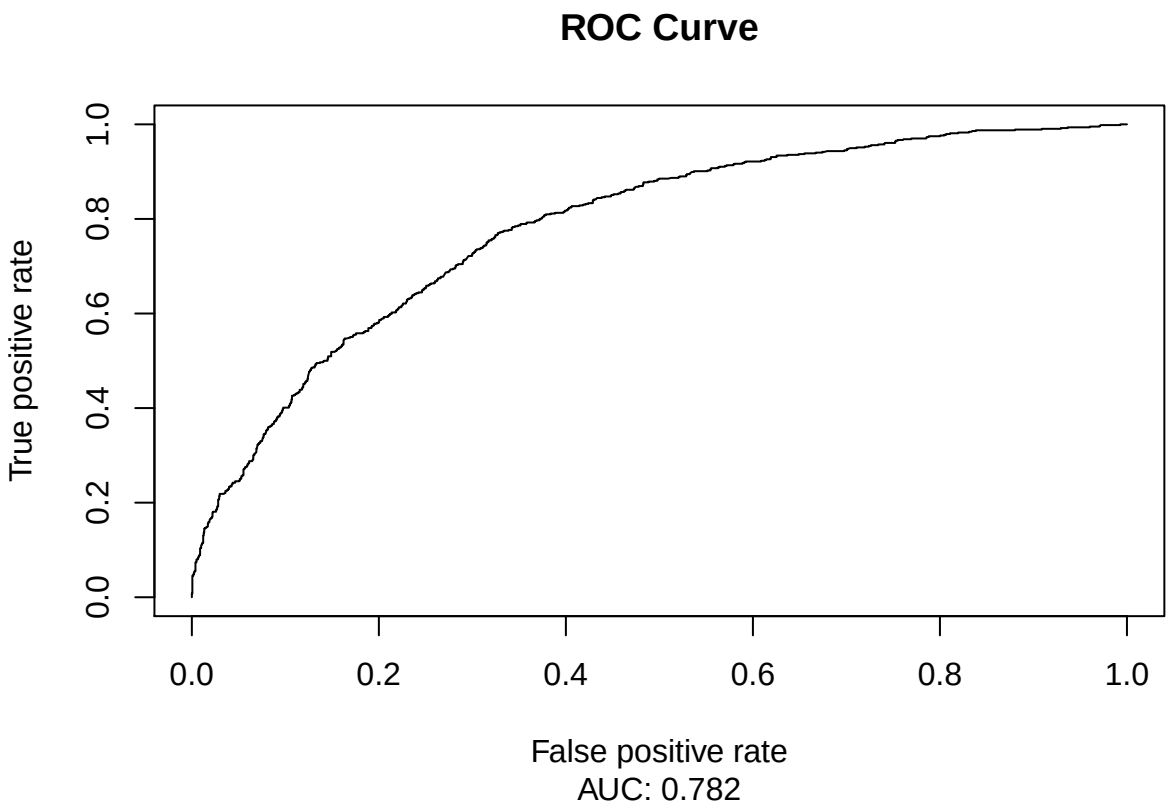
```
modstat(model2, test)
```




```
## F1 =  0.3329706
```

```
## [1] " 1"
## [2] " c(1682, 130, 483, 153)"
## [3] " c(Accuracy = 0.749591503267974, Kappa = 0.205908126849465, AccuracyLower = 0.731932472926179, A
## [4] " c(Sensitivity = 0.240566037735849, Specificity = 0.928256070640177, ‘Pos Pred Value‘ = 0.54063
## [5] " sens_spec"
## [6] " list()"
```

```
modstat(model3, test)
```



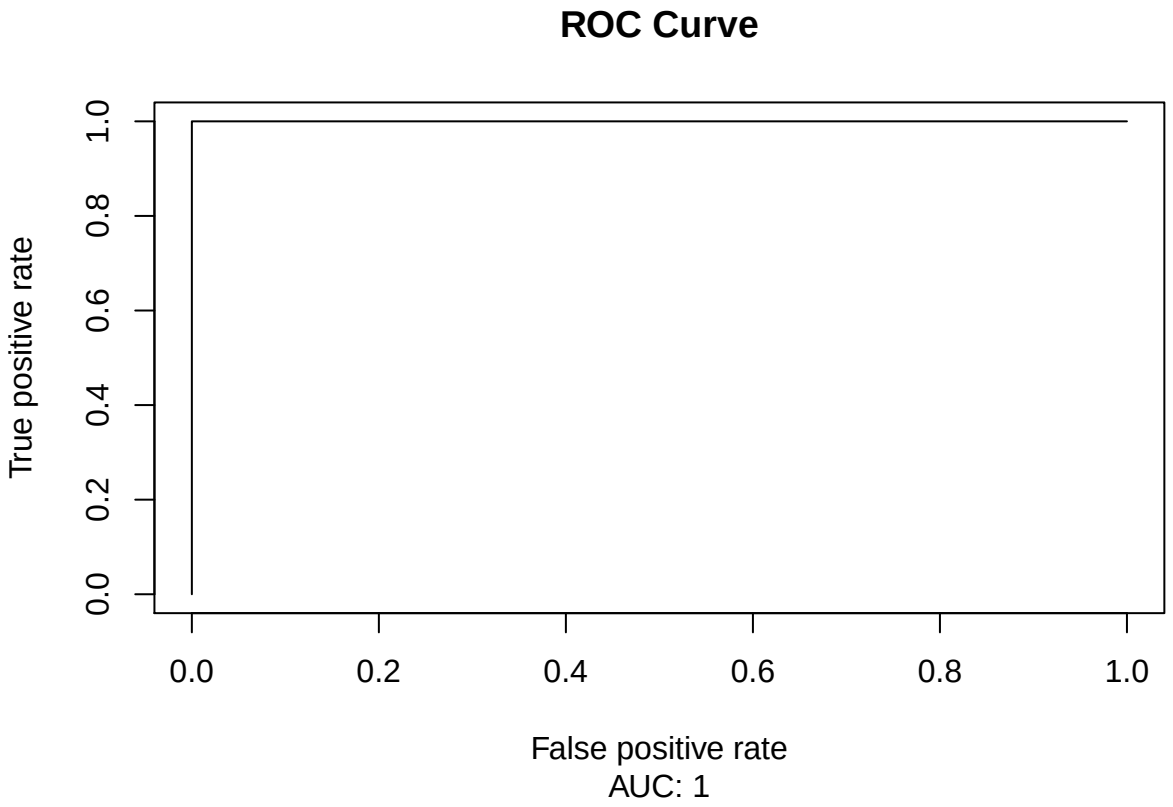

```
## F1 =  0.4219235
```

```
## [1] " 1"
## [2] " c(1685, 127, 432, 204)"
## [3] " c(Accuracy = 0.771650326797386, Kappa = 0.296864016968591, AccuracyLower = 0.754498018991632, A
## [4] " c(Sensitivity = 0.320754716981132, Specificity = 0.929911699779249, ‘Pos Pred Value‘ = 0.61631
## [5] " sens_spec"
## [6] " list()"
```

```
modstat(model4, test)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

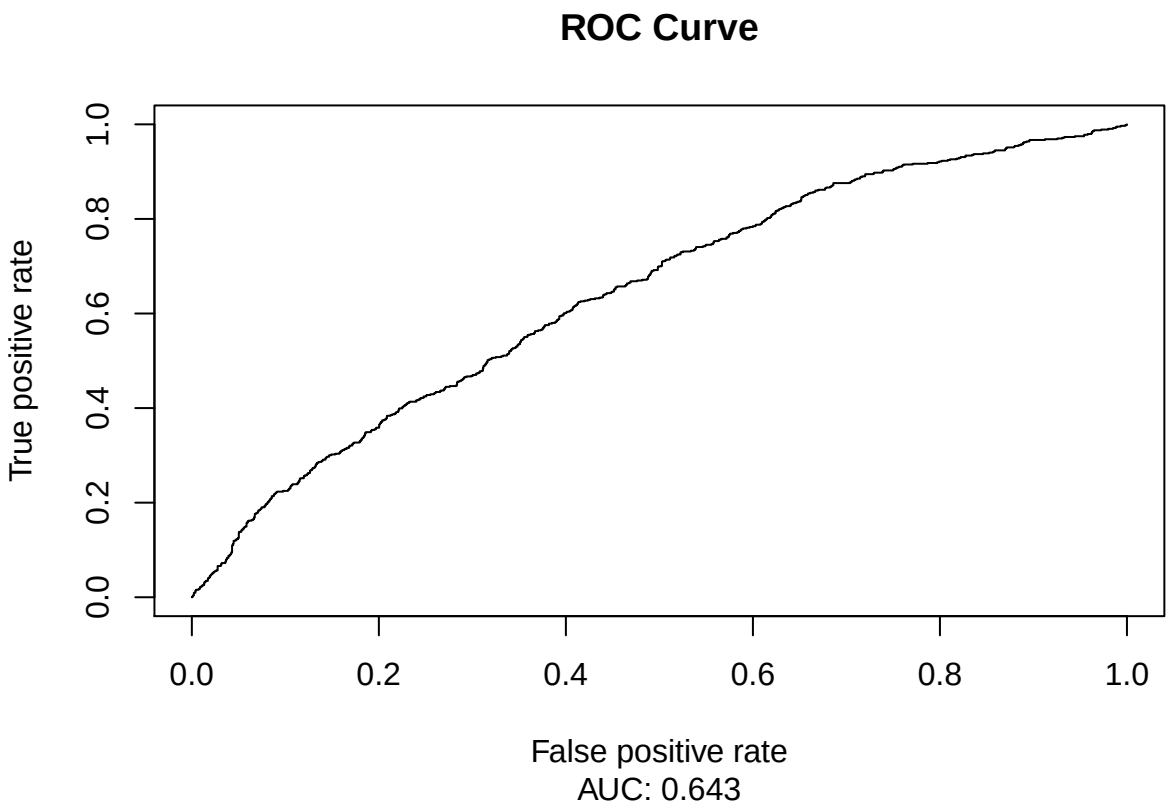
```
## Warning in predict.lm(model, test): prediction from a rank-deficient fit may be
## misleading
```




```
## F1 =  1
```

```
## [1] " 1"
## [2] " c(1812, 0, 0, 636)"
## [3] " c(Accuracy = 1, Kappa = 1, AccuracyLower = 0.998494239594653, AccuracyUpper = 1, AccuracyNull =
## [4] " c(Sensitivity = 1, Specificity = 1, ‘Pos Pred Value‘ = 1, ‘Neg Pred Value‘ = 1, Precision = 1,
## [5] " sens_spec"
## [6] " list()"
```

```
modstat(model5, test)
```



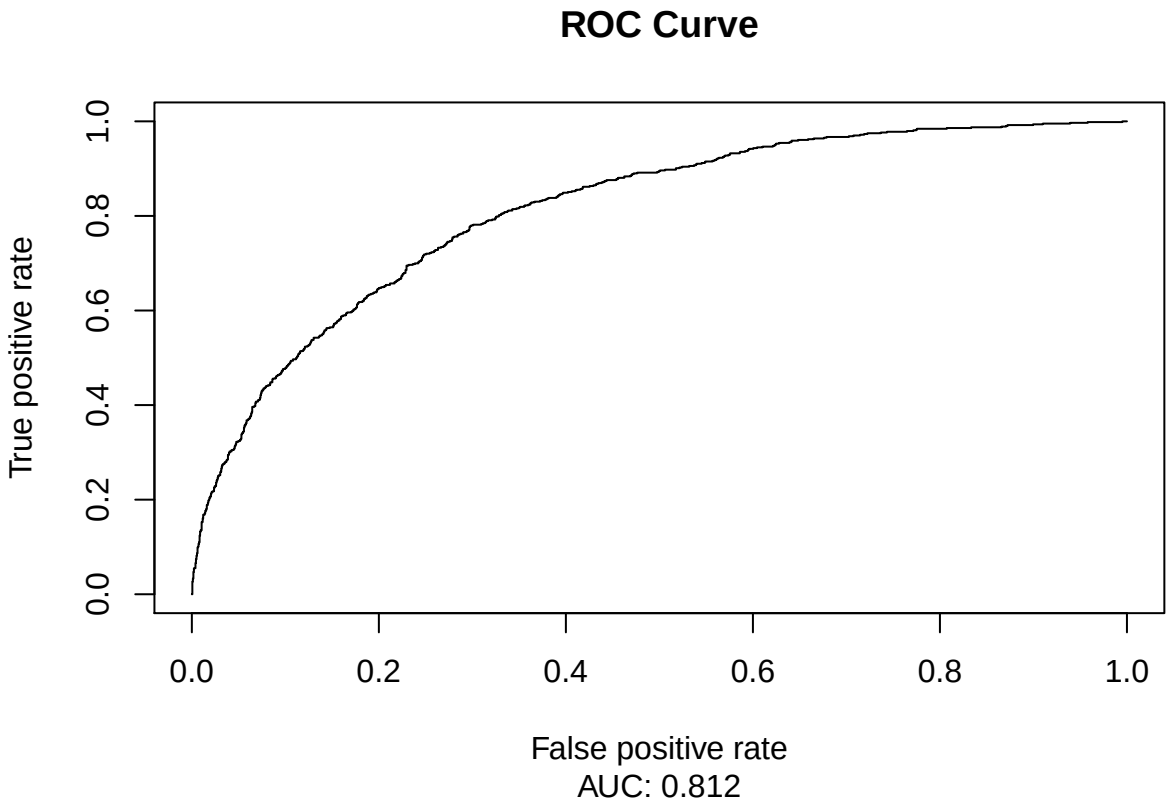

```
## F1 =  0.09014085
```

```
## [1] " 1"
## [2] " c(1770, 42, 604, 32)"
## [3] " c(Accuracy = 0.736111111111111, Kappa = 0.0380449064206253, AccuracyLower = 0.718171898328195,
## [4] " c(Sensitivity = 0.050314465408805, Specificity = 0.97682119205298, ‘Pos Pred Value‘ = 0.432432
## [5] " sens_spec"
## [6] " list()"
```

```
modstat(model6, test)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
## Warning in predict.lm(model, test): prediction from a rank-deficient fit may be
## misleading
```




```
## F1 =  0.4838057
```

```
## [1] " 1"
## [2] " c(1699, 113, 397, 239)"
## [3] " c(Accuracy = 0.791666666666667, Kappa = 0.36653677545056, AccuracyLower = 0.775030346732191, A
## [4] " c(Sensitivity = 0.375786163522013, Specificity = 0.937637969094923, ‘Pos Pred Value‘ = 0.67897
## [5] " sens_spec"
## [6] " list()"
```

Conclusion