Assignment4_Data621

Puja Roy, Mubashira Qari, Marco Castro, Zach Rose, Erick Hadi

2025-03-23

DATA EXPLORATION

We checked the overview of the dataset's structure while, summary() provided statistical summaries for each variable. The dataset contains 26 columns and 8161 rows.

```
# Check the structure of the dataset
str(training_df)
```

```
spc_tbl_ [8,161 x 26] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ INDEX      : num [1:8161] 1 2 4 5 6 7 8 11 12 13 ...
 $ TARGET_FLAG: num [1:8161] 0 0 0 0 0 1 0 1 1 0 ...
 $ TARGET_AMT : num [1:8161] 0 0 0 0 0 ...
 $ KIDSDRIV   : num [1:8161] 0 0 0 0 0 0 0 1 0 0 ...
 $ AGE        : num [1:8161] 60 43 35 51 50 34 54 37 34 50 ...
 $ HOMEKIDS   : num [1:8161] 0 0 1 0 0 1 0 2 0 0 ...
 $ YOJ        : num [1:8161] 11 11 10 14 NA 12 NA NA 10 7 ...
 $ INCOME     : chr [1:8161] "$67,349" "$91,449" "$16,039" NA ...
 $ PARENT1    : chr [1:8161] "No" "No" "No" "No" ...
 $ HOME_VAL   : chr [1:8161] "$0" "$257,252" "$124,191" "$306,251" ...
 $ MSTATUS    : chr [1:8161] "z_No" "z_No" "Yes" "Yes" ...
 $ SEX        : chr [1:8161] "M" "M" "z_F" "M" ...
 $ EDUCATION  : chr [1:8161] "PhD" "z_High School" "z_High School" "<High School" ...
 $ JOB        : chr [1:8161] "Professional" "z_Blue Collar" "Clerical" "z_Blue Collar" ...
 $ TRAVTIME   : num [1:8161] 14 22 5 32 36 46 33 44 34 48 ...
 $ CAR_USE    : chr [1:8161] "Private" "Commercial" "Private" "Private" ...
 $ BLUEBOOK   : chr [1:8161] "$14,230" "$14,940" "$4,010" "$15,440" ...
 $ TIF        : num [1:8161] 11 1 4 7 1 1 1 1 1 7 ...
 $ CAR_TYPE   : chr [1:8161] "Minivan" "Minivan" "z_SUV" "Minivan" ...
 $ RED_CAR    : chr [1:8161] "yes" "yes" "no" "yes" ...
 $ OLDCLAIM   : chr [1:8161] "$4,461" "$0" "$38,690" "$0" ...
```

```
$ CLM_FREQ   : num [1:8161] 2 0 2 0 2 0 0 1 0 0 ...
$ REVOKED    : chr [1:8161] "No" "No" "No" "No" ...
$ MVR_PTS    : num [1:8161] 3 0 3 0 3 0 0 10 0 1 ...
$ CAR_AGE    : num [1:8161] 18 1 10 6 17 7 1 7 1 17 ...
$ URBANICITY : chr [1:8161] "Highly Urban/ Urban" "Highly Urban/ Urban" "Highly Urban/ Urban
- attr(*, "spec")=
 .. cols(
 ..   INDEX = col_double(),
 ..   TARGET_FLAG = col_double(),
 ..   TARGET_AMT = col_double(),
 ..   KIDSDRIV = col_double(),
 ..   AGE = col_double(),
 ..   HOMEKIDS = col_double(),
 ..   YOJ = col_double(),
 ..   INCOME = col_character(),
 ..   PARENT1 = col_character(),
 ..   HOME_VAL = col_character(),
 ..   MSTATUS = col_character(),
 ..   SEX = col_character(),
 ..   EDUCATION = col_character(),
 ..   JOB = col_character(),
 ..   TRAVTIME = col_double(),
 ..   CAR_USE = col_character(),
 ..   BLUEBOOK = col_character(),
 ..   TIF = col_double(),
 ..   CAR_TYPE = col_character(),
 ..   RED_CAR = col_character(),
 ..   OLDCLAIM = col_character(),
 ..   CLM_FREQ = col_double(),
 ..   REVOKED = col_character(),
 ..   MVR_PTS = col_double(),
 ..   CAR_AGE = col_double(),
 ..   URBANICITY = col_character()
 .. )
- attr(*, "problems")=<externalptr>
```

Preliminary Data Cleaning

In this section, we reformat variable values into numeric and factor formats. Currency values are converted from a string format to a numeric format by removing `$` and commas.

```r
# Remove $ and , from currency columns
# Function to clean dollar values
clean_money <- function(x) {
  as.numeric(gsub("[$,]", "", x))
}

money_vars <- c("INCOME", "HOME_VAL", "BLUEBOOK", "OLDCLAIM", "TARGET_AMT")
factor_vars <- c("TARGET_FLAG", "SEX",  "PARENT1", "MSTATUS", "URBANICITY", "REVOKED",  "RED_
numeric_vars <- c("TARGET_AMT", "OLDCLAIM", "CLM_FREQ", "MVR_PTS", "CAR_AGE", "BLUEBOOK", "HO

# Apply to relevant columns
clean_df <-  function(df) {

  df <- df |>
    mutate(
      JOB = as.factor(ifelse(is.na(JOB), "Unknown", as.character(JOB))),
      RED_CAR = if_else(RED_CAR == "yes", "Yes", "No"),
      URBANICITY = if_else(URBANICITY == "Highly Urban/ Urban", "Rural", "Urban")
    )

  df[money_vars] <- lapply(df[money_vars], clean_money)
  df[factor_vars] <- lapply(df[factor_vars], function(x) { as.factor(x) })
  df[numeric_vars] <- lapply(df[numeric_vars], function(x) { as.numeric(x) })

  return (df |>
    mutate(across(
      .cols = where(is.factor),
      .fns = ~ factor(sub("^z_", "", .))
    ))
  )
}

training_df <- clean_df(training_df) |>
  subset(select=-c(INDEX))
testing_df <- clean_df(testing_df)
```

Summary Statistics

We review the mean, standard deviation, and median for all numeric variables, helping us understand the data distribution and central tendencies.

Summary Stats for Numeric Variables

```
  TARGET_AMT        OLDCLAIM        CLM_FREQ         MVR_PTS
 Min.   :     0   Min.   :    0   Min.   :0.0000   Min.   : 0.000
 1st Qu.:     0   1st Qu.:    0   1st Qu.:0.0000   1st Qu.: 0.000
 Median :     0   Median :    0   Median :0.0000   Median : 1.000
 Mean   :  1504   Mean   : 4037   Mean   :0.7986   Mean   : 1.696
 3rd Qu.:  1036   3rd Qu.: 4636   3rd Qu.:2.0000   3rd Qu.: 3.000
 Max.   :107586   Max.   :57037   Max.   :5.0000   Max.   :13.000

   CAR_AGE          BLUEBOOK        HOME_VAL          INCOME
 Min.   :-3.000   Min.   : 1500   Min.   :     0   Min.   :     0
 1st Qu.: 1.000   1st Qu.: 9280   1st Qu.:     0   1st Qu.: 28097
 Median : 8.000   Median :14440   Median :161160   Median : 54028
 Mean   : 8.328   Mean   :15710   Mean   :154867   Mean   : 61898
 3rd Qu.:12.000   3rd Qu.:20850   3rd Qu.:238724   3rd Qu.: 85986
 Max.   :28.000   Max.   :69740   Max.   :885282   Max.   :367030
 NA's   :510                      NA's   :464      NA's   :445
      YOJ           HOMEKIDS         KIDSDRIV           AGE
 Min.   : 0.0   Min.   :0.0000   Min.   :0.0000   Min.   :16.00
 1st Qu.: 9.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:39.00
 Median :11.0   Median :0.0000   Median :0.0000   Median :45.00
 Mean   :10.5   Mean   :0.7212   Mean   :0.1711   Mean   :44.79
 3rd Qu.:13.0   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:51.00
 Max.   :23.0   Max.   :5.0000   Max.   :4.0000   Max.   :81.00
 NA's   :454                                      NA's   :6
```

Standard Deviviations for Numeric Variables

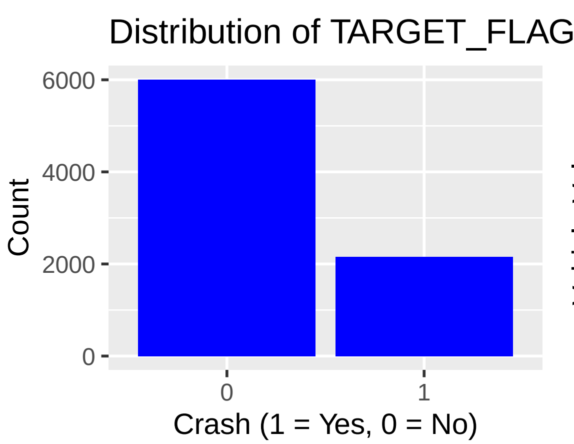
Summary Stats for Categorical Variables

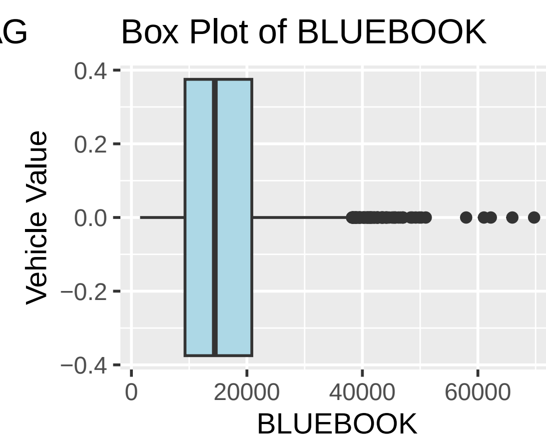
```
 TARGET_FLAG SEX      PARENT1    MSTATUS    URBANICITY   REVOKED    RED_CAR
 0:6008      F:4375   No :7084   No :3267   Rural:6492   No :7161   No :5783
 1:2153      M:3786   Yes:1077   Yes:4894   Urban:1669   Yes:1000   Yes:2378
```

```
             JOB                EDUCATION          CAR_TYPE            CAR_USE
 Blue Collar :1825   <High School:1203   Minivan     :2145   Commercial:3029
 Clerical    :1271   Bachelors   :2242   Panel Truck: 676   Private   :5132
 Professional:1117   High School :2330   Pickup     :1389
 Manager     : 988   Masters     :1658   Sports Car : 907
 Lawyer      : 835   PhD         : 728   SUV        :2294
 Student     : 712                       Van        : 750
 (Other)     :1413
```

Visualizing Our Data

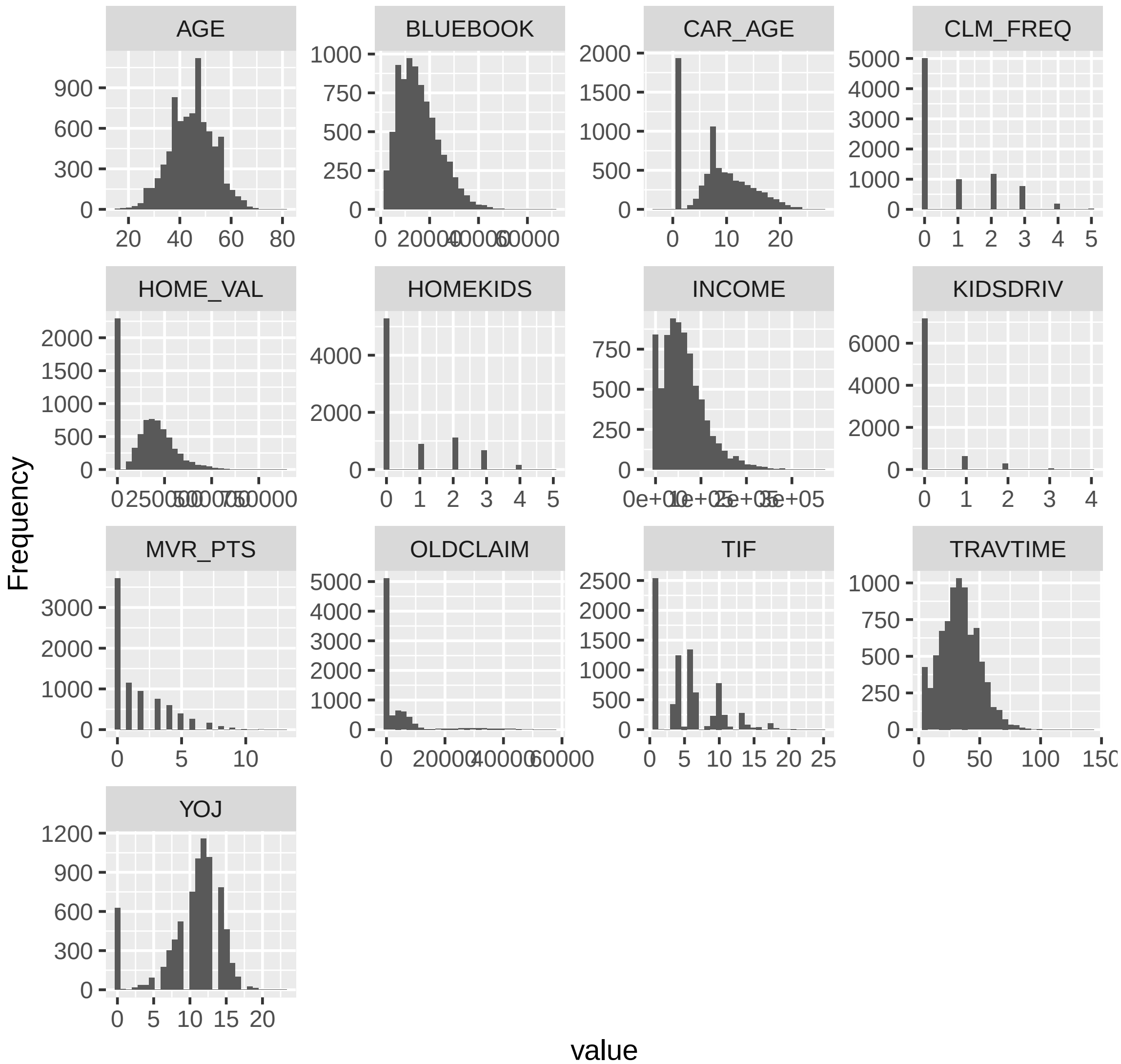
The first visualization is a bar chart displaying the distribution of TARGET_FLAG, helping to assess the proportion of crashes vs. non-crashes. The second visualization is a box plot for BLUEBOOK, showing the distribution and identifying potential outliers in vehicle values. TARGET_FLAG ratio of having more 0s than 1s indicates that most of the customers in the dataset did not have a crash. It's class imbalance, and it impacts the accuracy of classification models. The BLUEBOOK boxplot indicates that the variable has a strongly dominant majority of values or is long-tailed by outliers. There could be a chance that the data is non-representative, i.e., the majority of the cars have low values with few cars of high value on the tail end of the distribution.



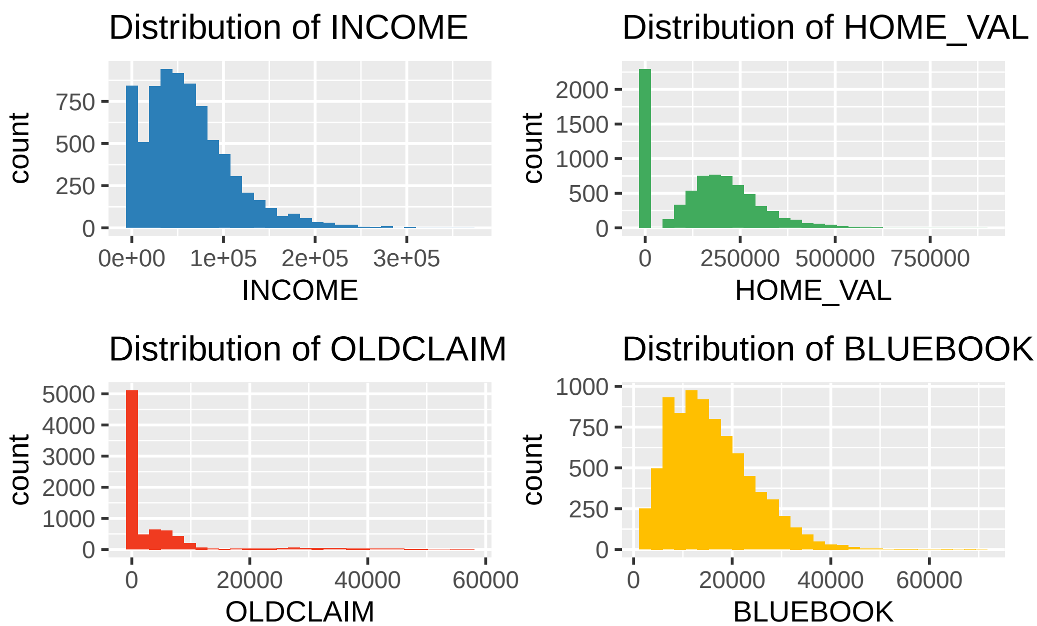


Visualize Distributions

Next we will use DataExplorer for visualize the distributions for our numberic variables. This gives mini-histograms for all numerical variables, so we can quickly spot skewness, check distribution and value ranges, and identify variables with spikes or unusual spread.



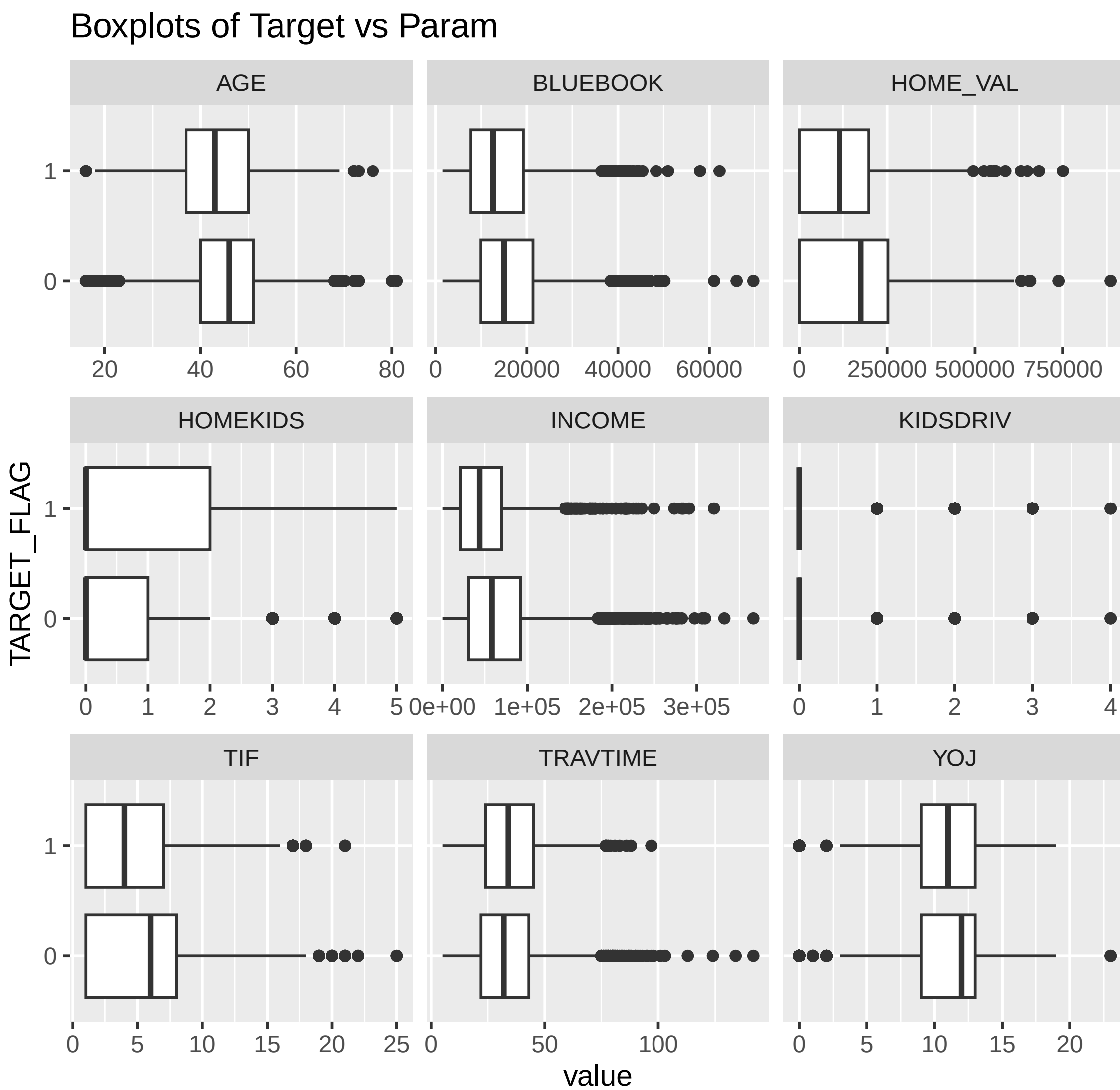
Here we focus on four variables that show skewness from their histograms that may benefit from a log transforation for when we apply a linear model. Skewness distorts relationships in regression. Log transformations may reduce skewness by compressing extreme values.



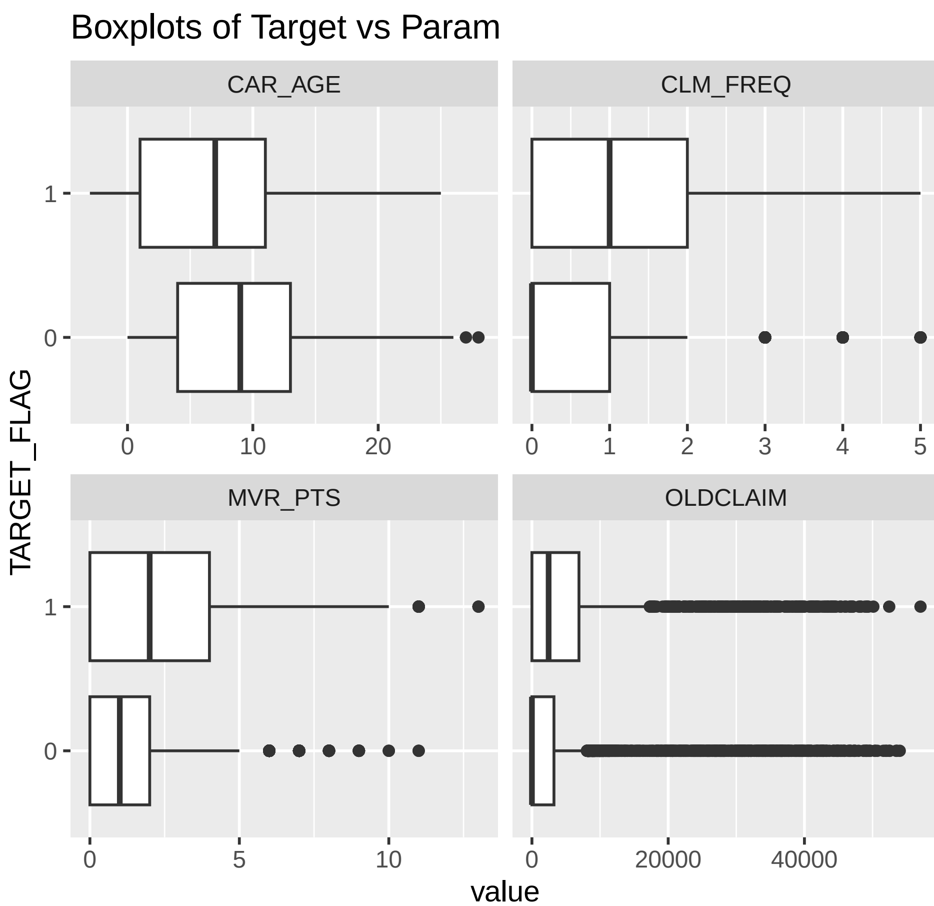

```
Skewness value for INCOME: 1.186317

Skewness value for HOME_VAL: 0.488595

Skewness value for OLDCLAIM: 3.11904

Skewness value for BLUEBOOK: 0.7942141
```

Visualizing Distributions for Numberical Variables

The boxplots on the following pages show the Interquartile Ranges for our numerical variables vs TARGET_FLAG. The IQR helps us visualize how similar/disimilar the ranges are between people who crashed (1) vs those who didn't crash (0) for each given parameter. The plots also gives us a sense of potential outliers for each parameter; we will want to look out for observations beyond the whiskers or address transform our data.

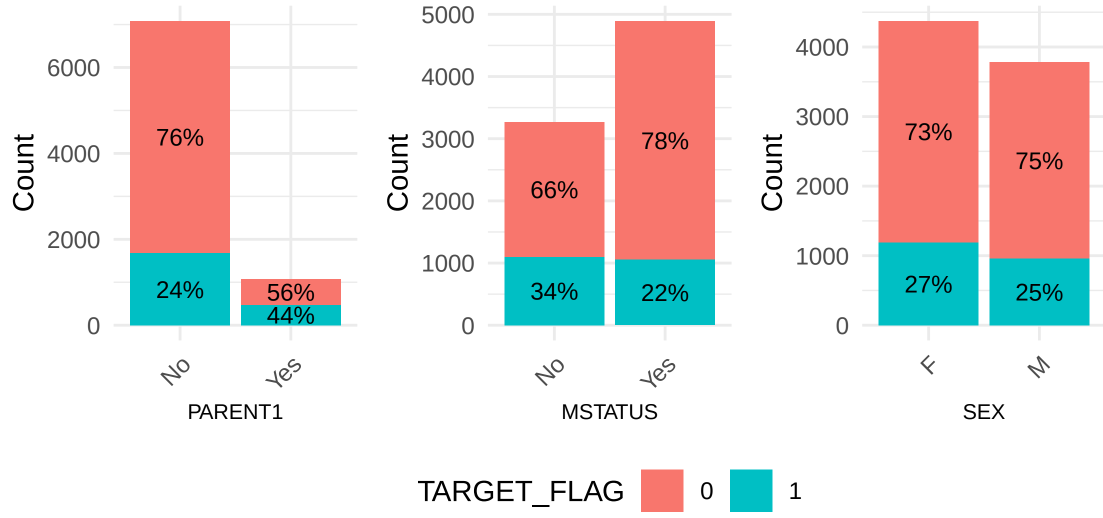


Our boxplots show that there is good difference between the medians for MVR_PTS (motor vehicle record points), HOME_VAL (home value), and TIF (time in force) for crashers and non-crashes and moderate differences for INCOME, YOJ (years on Job) and CAR_AGE. CLM_FREQ (claim frequency) how high variation but this is likely due to the disproportionate number of claims from crashers versus non-crashers who may be processing other claim types outside of automotive vehicle accidents.



Visualizing Distributions for Categorical Variables

Visualizing the distributions of out categorical variables helps ensure variables are treated as discrete categories, not continuous numbers.



Key takeaways:

PARENT1 - greater percentage of parents crashed their cars vs. non-parents (though smaler total number)

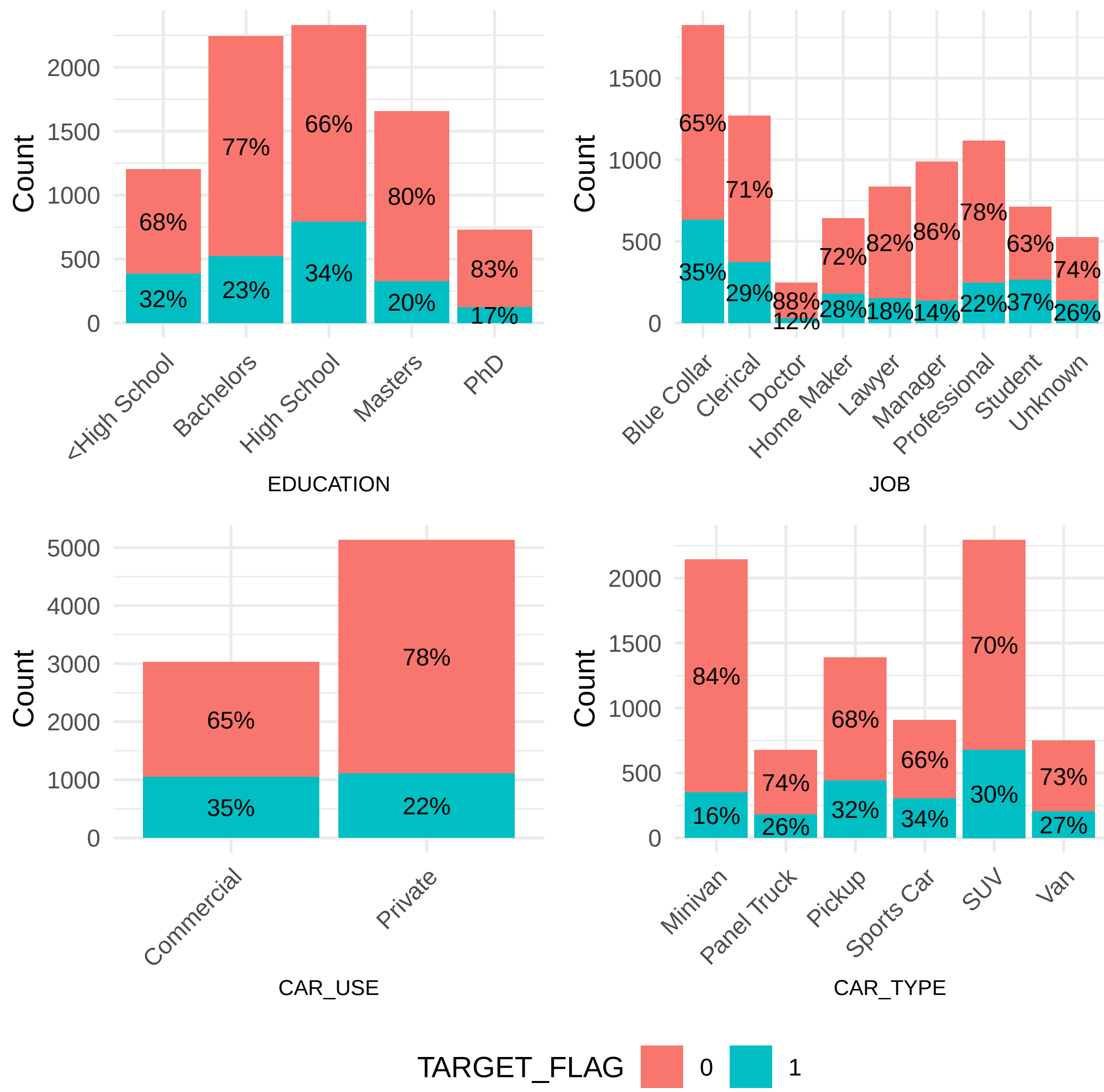
REVOKED - greater percentage of people whose license was revoked in the last 7 years crashed their cars vs. others (though smaler total number)

EDUCATION - High School graduates had the highest number of crashes among all educational groups; a higher percentage of high school graduates were also in a crash when compared to the breakdown of other groups.

JOB - Blue collar workers had the highest number of crashes among all job groups; a higher percentage of blue collar workers were also in a crash when compared to the breakdown of other groups.

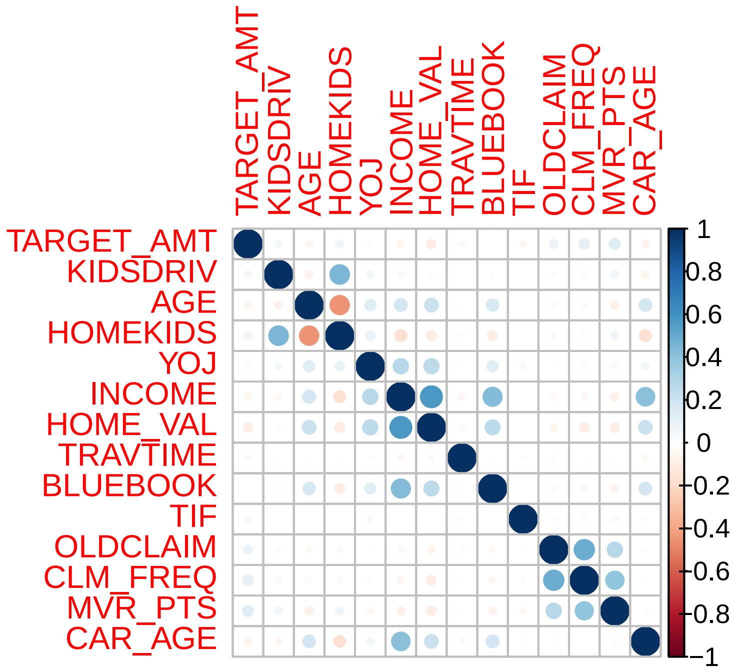
CAR_USE - A similar number of commerical and private policy holders that were involved in a crash, though private policies accounted for a smaller percentage of crashes within that group.

CARTYPE - SUV's had highest total number of crashes among all vehicle types and the second highest percentage when compared to other groups



Correlation Analysis

The correlation plot provides a graphical impression of the relationship between different numerical variables of the data set. The size and color of the circles convey the strength of the relationship. Dark blue circles indicate strong positive relationships, which predict that if one variable is increasing, so will the other. Dark red circles on the other hand produce negative high correlations, where the rise of one variable is accompanied by a fall in the other. Light-colored circles show weak or no correlation.



Plots suggest that TARGET_FLAG is not strongly correlated with most of the numeric features. Hence, no numeric feature is particularly useful to a customer to make a claim. What it says is perhaps claims will be predicted more accurately with a more advanced method, i.e., interactions, categorical features, or non-linear models.

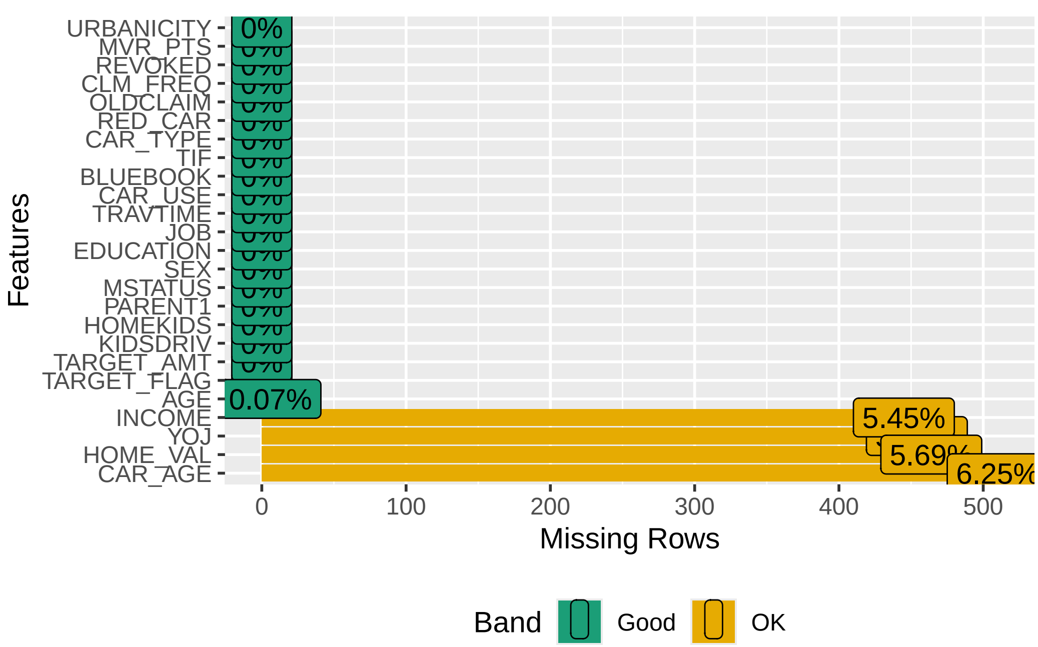
Moreover, some variables such as MVR_PTS (Motor Vehicle Record Points) and CLM_FREQ (Claim Frequency) show high positive correlation. As would be expected—more holders of violations file more claims. Also, correlations such as AGE and HOMEKIDS or KIDSDRIV reflect older people with children at home, a typical demographic pattern.

Knowledge of such relationships is helpful in model building and data preparation. Correlated variables cause multicollinearity in regression models, necessitating extra work in the form of variance inflation factor (VIF) testing to eliminate redundant information. Target_FLAG correlations also imply that other transformations, engineered features, or other sources of data might be helpful in enhancing predictiveness.

Missing Value Analysis

Missing values show that the data have missing values in relatively high counts of major variables, and most missing values exist in JOB (526 missing), CAR_AGE (510 missing), HOME_VAL (464 missing), INCOME (445 missing), YOJ (454 missing), and AGE (6 missing). Missing values may produce bias or weaken the ability of the model to predict unless appropriately handled.

```
TARGET_FLAG  TARGET_AMT    KIDSDRIV         AGE    HOMEKIDS         YOJ
          0           0           0           6           0         454
     INCOME     PARENT1    HOME_VAL     MSTATUS         SEX   EDUCATION
        445           0         464           0           0           0
        JOB    TRAVTIME     CAR_USE    BLUEBOOK         TIF    CAR_TYPE
          0           0           0           0           0           0
    RED_CAR    OLDCLAIM    CLM_FREQ     REVOKED     MVR_PTS     CAR_AGE
          0           0           0           0           0         510
 URBANICITY
          0
```



Missing values for large numbers for variables like JOB and HOME_VAL mean the work details weren't reported for certain customers or home values weren't reported. Missing values for YOJ and INCOME may be for self-employed or the ones having the wrong job history. For

CAR_AGE, missing values may be for new acquisitions or lease of automobiles where the age wasn't reported.

Depending on the count of missing values, different imputation methods will be needed. Quantitative features like AGE, INCOME, and CAR_AGE can be imputed with median to remove the impact of outliers, while features like JOB may require one extra "Unknown" value. Additionally, making missing value indicators for features like HOME_VAL will enable the model to learn missing value patterns and enhance the predictive power. Closing gaps like these in an efficient way is vital to achieve model reliability and precision.

DATA PREPARATION

Create Flags

```
# Create binary flags for missing values (1 = missing, 0 = not missing)
training_df <- training_df %>%
  mutate(
    YOJ_MISSING = ifelse(is.na(YOJ), 1, 0),
    INCOME_MISSING = ifelse(is.na(INCOME), 1, 0),
    HOME_VAL_MISSING = ifelse(is.na(HOME_VAL), 1, 0),
    CAR_AGE_MISSING = ifelse(is.na(CAR_AGE), 1, 0),
    JOB_MISSING = ifelse(JOB == "Unknown", 1, 0)
  )
```

Handling Missing Values

Missing values in key numeric columns (AGE, YOJ, INCOME, HOME_VAL, and CAR_AGE) are replaced with their respective medians. Data is preserved and no biased. We then convert YOJ, INCOME, HOME_VAL, and CAR_AGE into numeric to make it easier for managing in the list of upcoming transformations.

In addition, binary flags (YOJ_MISSING, INCOME_MISSING, HOME_VAL_MISSING, CAR_AGE_MISSING, JOB_MISSING) are built to indicate missing values in variables. This allows for models to detect missingness as a predictor.

```
training_df$AGE[is.na(training_df$AGE)] <- median(training_df$AGE, na.rm = TRUE)
training_df$YOJ[is.na(training_df$YOJ)] <- median(training_df$YOJ, na.rm = TRUE)
training_df$INCOME[is.na(training_df$INCOME)] <- median(training_df$INCOME, na.rm = TRUE)
training_df$HOME_VAL[is.na(training_df$HOME_VAL)] <- median(training_df$HOME_VAL, na.rm = TRU
training_df$CAR_AGE[is.na(training_df$CAR_AGE)] <- median(training_df$CAR_AGE, na.rm = TRUE)
```

```r
colSums(is.na(training_df))
```

```
     TARGET_FLAG        TARGET_AMT          KIDSDRIV               AGE
               0                 0                 0                 0
        HOMEKIDS               YOJ            INCOME           PARENT1
               0                 0                 0                 0
        HOME_VAL           MSTATUS               SEX         EDUCATION
               0                 0                 0                 0
             JOB          TRAVTIME           CAR_USE          BLUEBOOK
               0                 0                 0                 0
             TIF          CAR_TYPE           RED_CAR          OLDCLAIM
               0                 0                 0                 0
        CLM_FREQ           REVOKED           MVR_PTS           CAR_AGE
               0                 0                 0                 0
      URBANICITY       YOJ_MISSING    INCOME_MISSING  HOME_VAL_MISSING
               0                 0                 0                 0
 CAR_AGE_MISSING       JOB_MISSING
               0                 0
```

Feature Engineering

Some of the new features aim to capture meaningful relationships in the data:

AVG_CLAIM: This is the feature that is created by dividing OLDCLAIM by CLM_FREQ so as to be able to interpret average payout per claim.

HIGH_RISK_CAR: Certain types of cars (e.g., “Sports Car”, “Luxury SUV”) are designated as high-risk by encoding them with a binary value of 1.

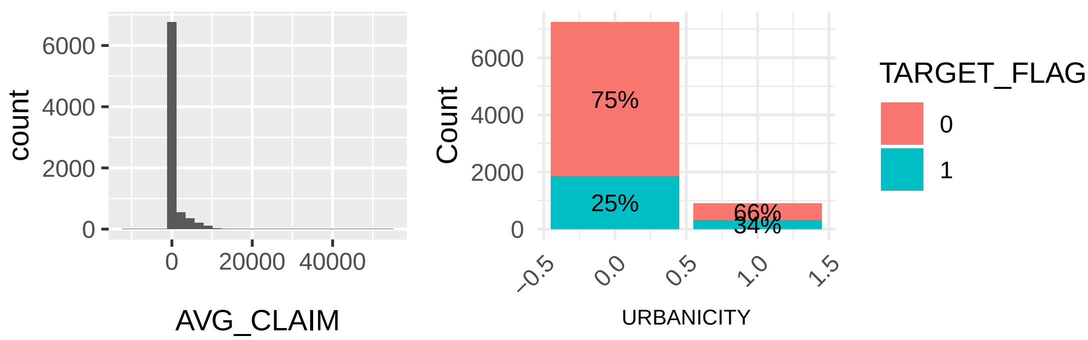
```r
# Convert CAR_AGE and OLDCLAIM to numeric to avoid errors
training_df$CLM_FREQ <- as.numeric(training_df$CLM_FREQ)
training_df$OLDCLAIM <- as.numeric(training_df$OLDCLAIM)

# Ensure no division by zero when creating CLAIMS_PER_YEAR
training_df <- training_df %>%
  mutate(
    AVG_CLAIM = ifelse(!is.na(CLM_FREQ) & CLM_FREQ > 0, OLDCLAIM / CAR_AGE, OLDCLAIM)
  )

# Create a binary flag for high-risk car types
training_df <- training_df %>%
```

```r
  mutate(
    HIGH_RISK_CAR = ifelse(CAR_TYPE %in% c("Sports Car", "Luxury SUV"), 1, 0)
  )

glimpse(training_df$AVG_CLAIM)
```

```
 num [1:8161] 248 0 3869 0 1130 ...
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
Warning: Removed 2 rows containing non-finite outside the scale range
(`stat_bin()`).
```



Transform data

Skewed numerical attributes are log-transformed to stabilize their distributions:

LOG_BLUEBOOK (log-transformed value of the vehicle),

LOG_OLDCLAIM (log-transformed old claims),

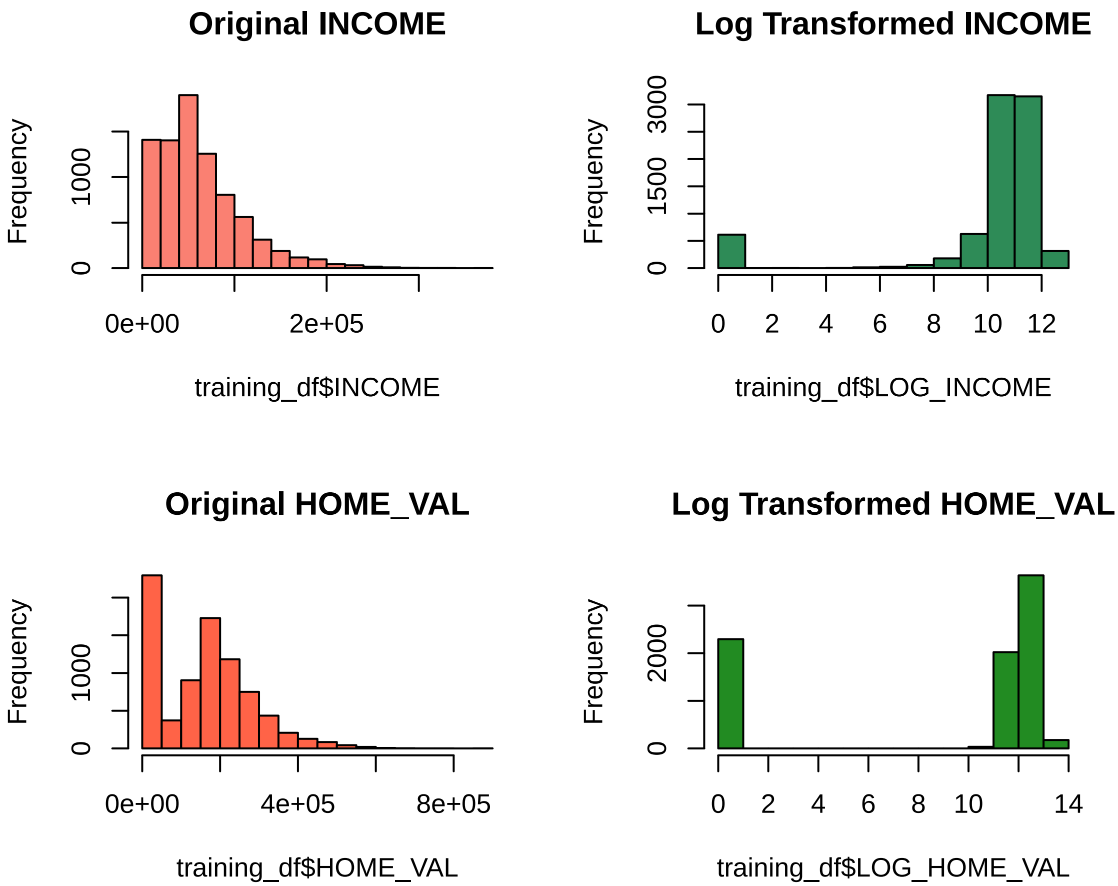
LOG_INCOME (log-transformed income). NA values are substituted with a small positive constant (0.01) before applying the log transformation to prevent computation errors.

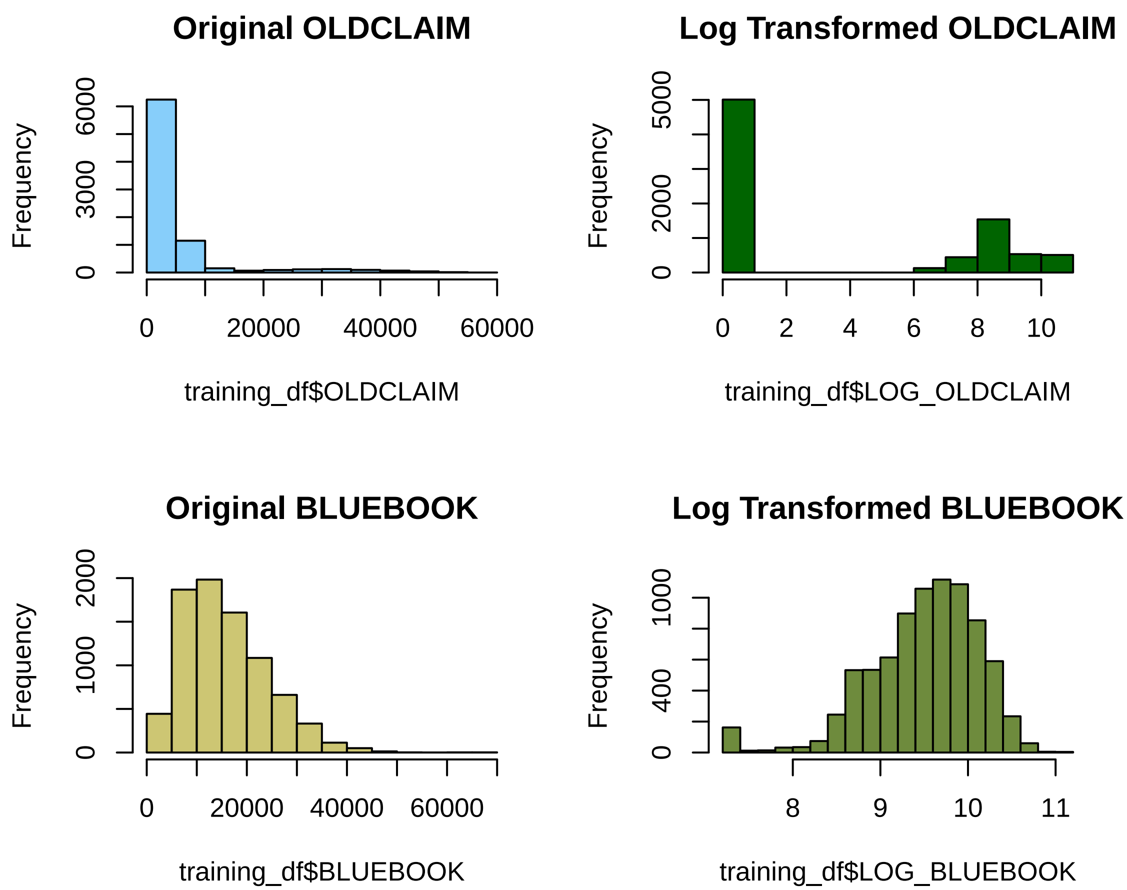
LOG_HOME_VAL (log-transformed home value)

```r
# Handle NA values before log transformation
training_df <- training_df %>%
  mutate(
    BLUEBOOK = ifelse(is.na(BLUEBOOK), 0.01, BLUEBOOK),
    OLDCLAIM = ifelse(is.na(OLDCLAIM), 0.01, OLDCLAIM),
    INCOME = ifelse(is.na(INCOME), 0.01, INCOME),
    HOME_VAL = ifelse(is.na(INCOME), 0.01, HOME_VAL),
    LOG_BLUEBOOK = log1p(BLUEBOOK),
    LOG_OLDCLAIM = log1p(OLDCLAIM),
    LOG_INCOME = log1p(INCOME),
    LOG_HOME_VAL = log1p(HOME_VAL)
  )
```

Compare Before and After Transformation

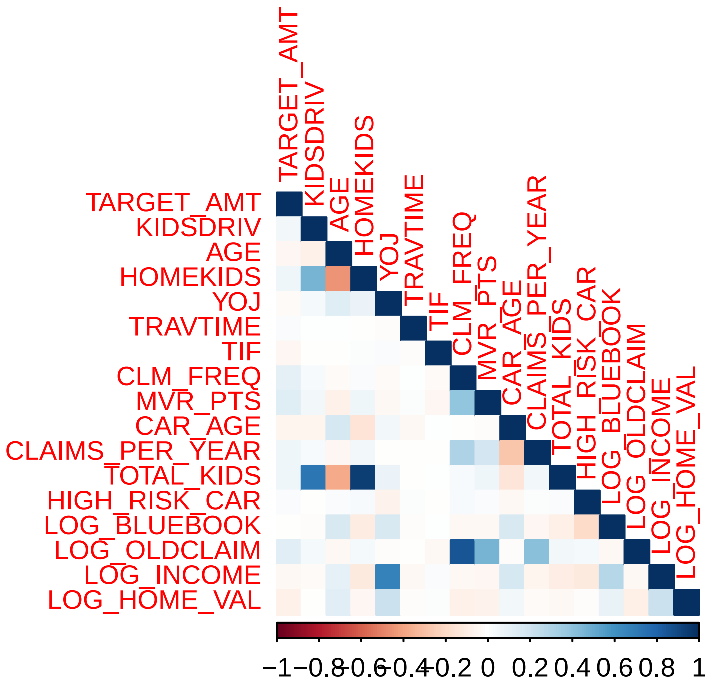
The charts below show the distributions for the values of the INCOME, HOME_VAL, OLDCLAIM, and BLUEBOOK parameters before and after applying a log transformation.


Original INCOME
Frequency
0
1000
0e+00
2e+05
training_df$INCOME
Log Transformed INCOME
Frequency
0
1500
3000
0
2
4
6
8
10
12
training_df$LOG_INCOME
Original HOME_VAL
Frequency
0
1000
0e+00
4e+05
8e+05
training_df$HOME_VAL
Log Transformed HOME_VAL
Frequency
0
2000
0
2
4
6
8
10
14
training_df$LOG_HOME_VAL



Check Variable Correlations (Multicollinearity Insight)

This updated Correlation Matrix shows correlation values for all numerical values including our log transformed values and our new variables. The matrix below shows that there is high correlation between LOG_OLDCLAIM and CLM_FREQ (Claim frequency) and LOG_INCOME and YOJ (Years on Job). There is moderate correlation between our HOMEKIDS, KIDSDRV (Kids driving) and AGE.



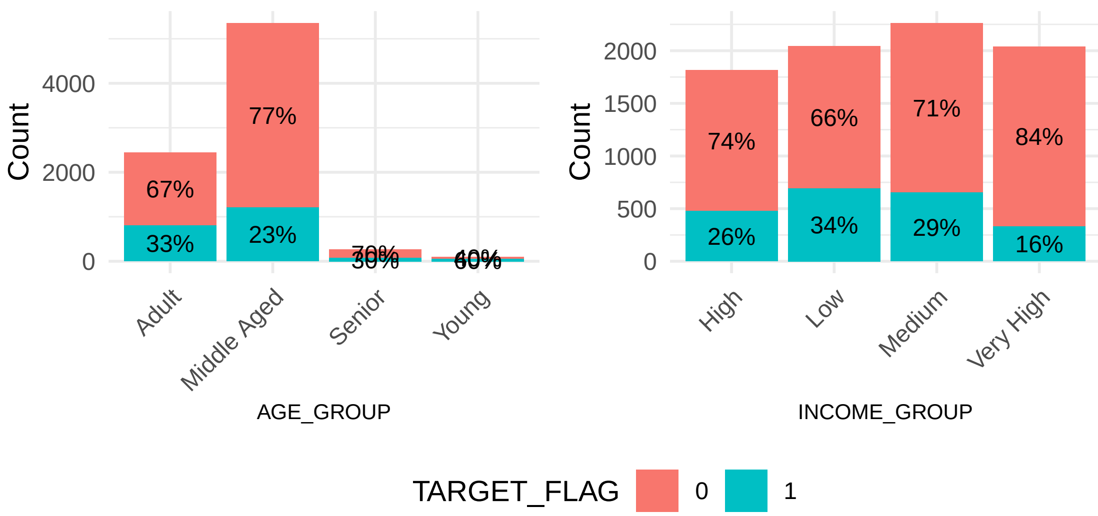
Categorical Groupings

Categorical groupings are created to reduce numerical variables: AGE_GROUP: Four age groups—Young (25), Adult (26-40), Middle-Aged (41-60), and Senior (>60)—categorize individuals.

INCOME_GROUP: Income is divided into four quantile-based groups (Low, Medium, High, Very High) such that there is an equal share of income levels.

```
# Transform data by putting it into buckets using case_when
training_df <- training_df %>%
  mutate(
    AGE_GROUP = case_when(
      AGE <= 25 ~ "Young",
      AGE > 25 & AGE <= 40 ~ "Adult",
      AGE > 40 & AGE <= 60 ~ "Middle Aged",
      AGE > 60 ~ "Senior",
      TRUE ~ NA_character_  # Handle missing values
    )
  )
```

```r
# Fix Income Grouping - Ensure proper quantile calculation
income_quantiles <- quantile(training_df$INCOME, probs = seq(0, 1, 0.25), na.rm = TRUE)

training_df <- training_df %>%
  mutate(
    INCOME_GROUP = case_when(
      INCOME <= income_quantiles[2] ~ "Low",
      INCOME > income_quantiles[2] & INCOME <= income_quantiles[3] ~ "Medium",
      INCOME > income_quantiles[3] & INCOME <= income_quantiles[4] ~ "High",
      INCOME > income_quantiles[4] ~ "Very High",
      TRUE ~ NA_character_
    )
  )
```



Creating New Features:

To capture relationships between features more effectively, new ratio-based features are formed:

CLAIMS_INCOME_RATIO: Indicates the proportion of old claims relative to income.

VEHICLE_VALUE_TO_INCOME: Indicates how cheap the vehicle is relative to income.

CLAIMS_TO_MVR_PTS: Prior claims quantity per vehicle record to indicate the severity of prior offenses.

Also an estimated risk score by averaging important risk indicators (MVR_PTS, CLM_FREQ, HIGH_RISK_CAR, and REVOKED). The risk score forms a composite measure of the subject's risk level.

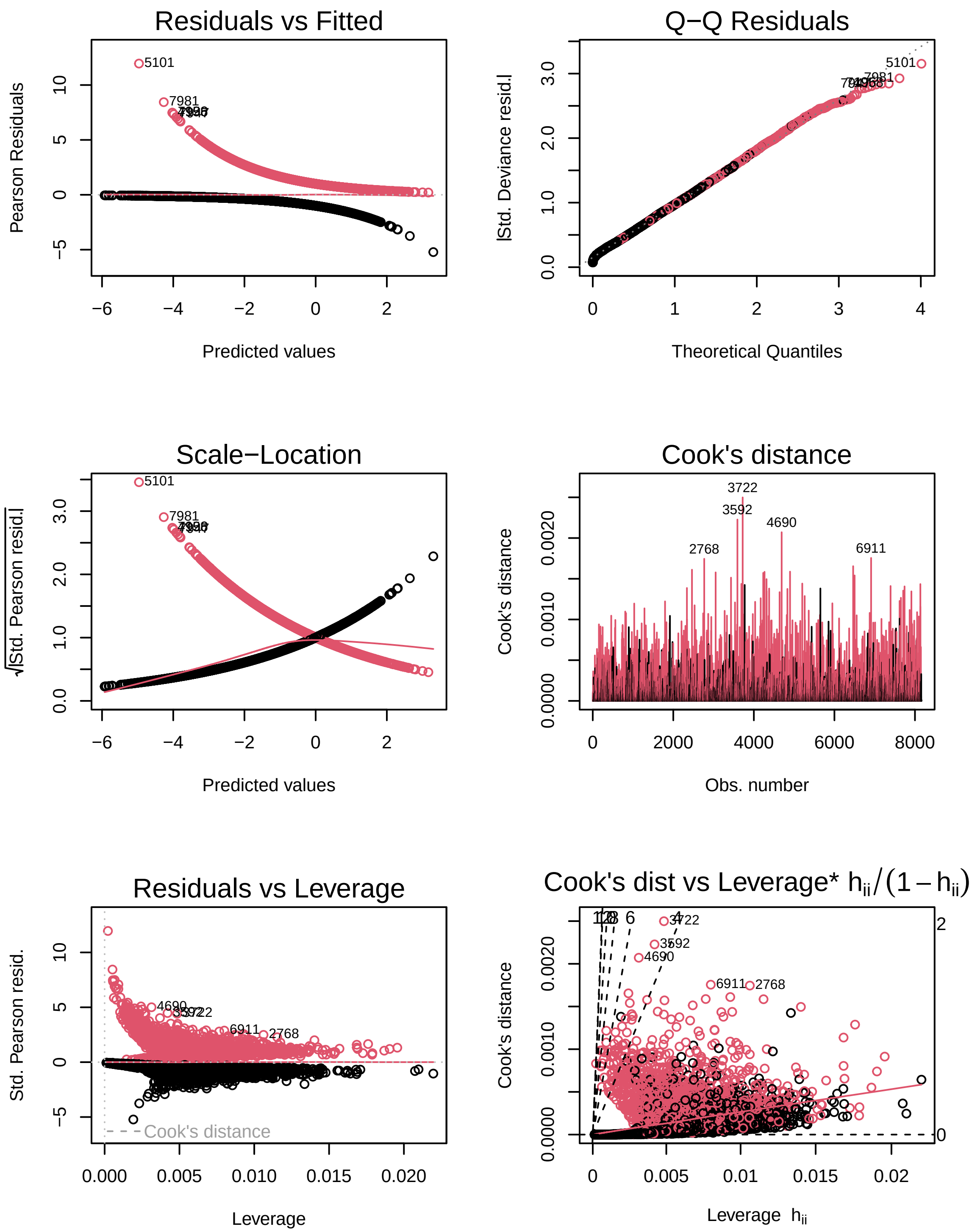
```
# Ensure no division by zero when creating ratios
training_df <- training_df %>%
  mutate(
    CLAIMS_INCOME_RATIO = ifelse(INCOME > 0, OLDCLAIM / INCOME, NA),
    VEHICLE_VALUE_TO_INCOME = ifelse(INCOME > 0, BLUEBOOK / INCOME, NA),
    CLAIMS_TO_MVR_PTS = ifelse(MVR_PTS > 0, OLDCLAIM / MVR_PTS, NA)
  )

# Create a risk score by combining various indicators
training_df <- training_df %>%
  mutate(
    RISK_SCORE = (as.numeric(MVR_PTS) * 0.4) + (as.numeric(CLM_FREQ) * 0.3) + (as.numeric(HI
  )
```

Checking for Outliers

We will fit a base model that includes all 23 original variables without any transformations. We will use this model to check for outliers using the Cook's Distance test. Below we see the output of the 10 observations with the highest Cook's Distance values.

```
Rows: 10
Columns: 7
$ index      <int> 3722, 3592, 4690, 6911, 2768, 6465, 2464, 4898, 4265, 3051
$ .fitted    <dbl> -2.966839, -2.994046, -3.217335, -2.104933, -1.810303, -3.2~
$ .resid     <dbl> 2.456430, 2.466948, 2.552103, 2.107090, 1.980821, 2.563574,~
$ .hat       <dbl> 0.004839881, 0.004203504, 0.003133085, 0.007998736, 0.01061~
$ .sigma     <dbl> 0.9474993, 0.9474961, 0.9474687, 0.9476024, 0.9476355, 0.94~
$ .cooksd    <dbl> 0.002498889, 0.002227328, 0.002071025, 0.001755391, 0.00174~
$ .std.resid <dbl> 2.462396, 2.472149, 2.556111, 2.115568, 1.991422, 2.566696,~
```

The diagnostic plots on the following page show that point 5101 stands out in our Residuals vs. Fitted and Scale Location plots. However, when take a closer look at some our other diagnostic plots and we don't see this point within the 10 highest Cook's distance values. Our Cook's distance values and respective plot shows that none of our points is an extreme outlier. Examining our Cook's distance vs Leverage plot doesn't present evidence of the presence of outliers with high leverage.


Residuals vs Fitted
Pearson Residuals
Predicted values
5101
7981
Q−Q Residuals
|Std. Deviance resid.|
Theoretical Quantiles
5101
7981
Scale−Location
√|Std. Pearson resid.|
Predicted values
5101
7981
Cook's distance
Cook's distance
Obs. number
3722
3592
4690
2768
6911
Residuals vs Leverage
Std. Pearson resid.
Leverage
Cook's distance
4690
3722
6911
2768
Cook's dist vs Leverage* $h_{ii}/(1-h_{ii})$
Cook's distance
Leverage h_{ii}
3722
3592
4690
6911
2768

MODEL BUILDING

Binary Logistic Regression Models for TARGET_FLAG

We use glm() with family = binomial for logistic regression. This model predicts the probability of a crash (1). We are predicting TARGET_FLAG (binary: 1 = made a claim, 0 = no claim).

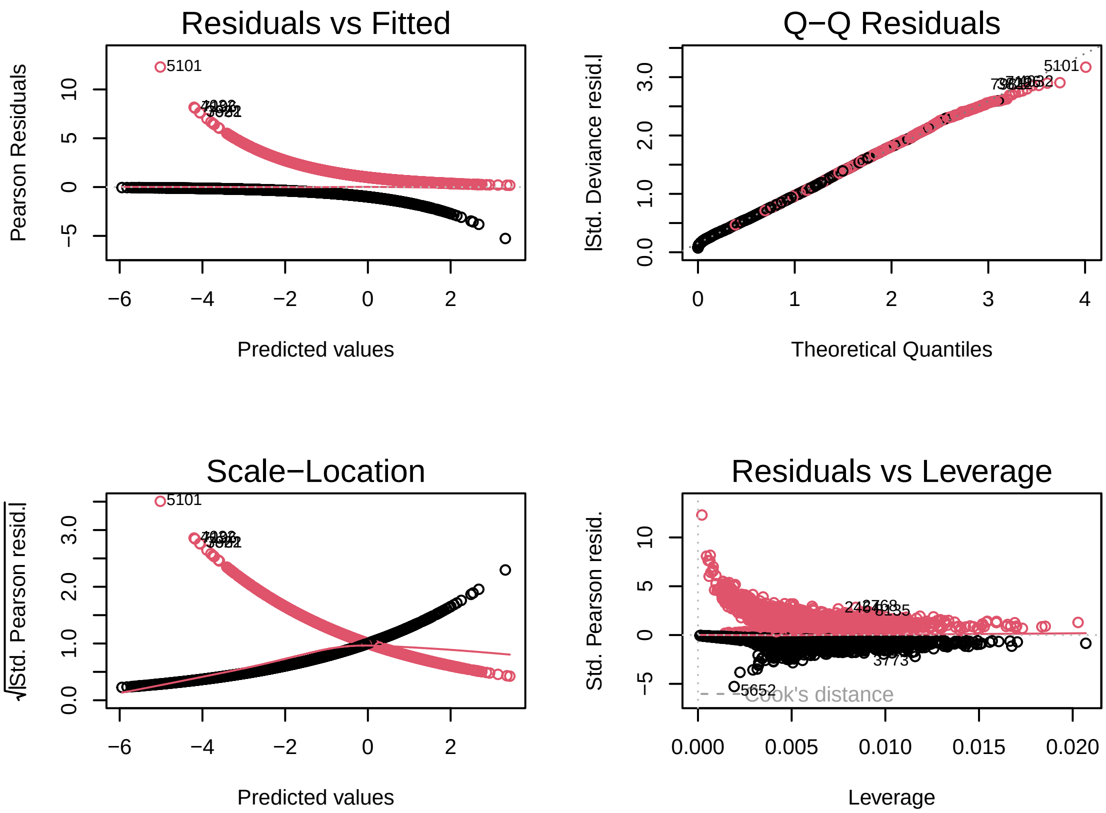
Model 1: Log Transformed Model

The log transformed model includes 19 our original variables and uses four log transformed variables: LOG_BLUEBOOK, LOG_HOME_VAL, LOG_INCOME, LOG_OLDCLAIM.

```
Call:
glm(formula = TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_AGE + CAR_TYPE +
    CAR_USE + CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL +
    LOG_INCOME + JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM +
    PARENT1 + RED_CAR + REVOKED + SEX + TIF + TRAVTIME + URBANICITY +
    YOJ, family = binomial, data = training_df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           3.0401191  0.6189290   4.912 9.02e-07 ***
AGE                  -0.0034867  0.0040448  -0.862 0.388684
LOG_BLUEBOOK         -0.3107452  0.0589589  -5.271 1.36e-07 ***
CAR_AGE              -0.0009956  0.0075215  -0.132 0.894697
CAR_TYPEPanel Truck   0.4874984  0.1501870   3.246 0.001171 **
CAR_TYPEPickup        0.5885180  0.1005952   5.850 4.91e-09 ***
CAR_TYPESports Car    1.0265903  0.1279633   8.023 1.04e-15 ***
CAR_TYPESUV           0.8080301  0.1076172   7.508 5.99e-14 ***
CAR_TYPEVan           0.6195414  0.1251736   4.949 7.44e-07 ***
CAR_USEPrivate       -0.7543996  0.0920903  -8.192 2.57e-16 ***
CLM_FREQ              0.0903450  0.0434843   2.078 0.037742 *
EDUCATIONBachelors   -0.4107709  0.1147459  -3.580 0.000344 ***
EDUCATIONHigh School  0.0265010  0.0954495   0.278 0.781285
EDUCATIONMasters     -0.3611188  0.1777855  -2.031 0.042234 *
EDUCATIONPhD         -0.3871715  0.2067055  -1.873 0.061060 .
HOMEKIDS              0.0241261  0.0374774   0.644 0.519737
LOG_HOME_VAL         -0.0291342  0.0069015  -4.221 2.43e-05 ***
LOG_INCOME           -0.0937313  0.0176949  -5.297 1.18e-07 ***
JOBClerical           0.1404673  0.1056664   1.329 0.183734
JOBDoctor            -0.7776908  0.2861833  -2.717 0.006579 **
```

```
JOBHome Maker        -0.2547855  0.1646928  -1.547 0.121855
JOBLawyer            -0.2227525  0.1877111  -1.187 0.235355
JOBManager           -0.9057062  0.1397265  -6.482 9.05e-11 ***
JOBProfessional      -0.1722614  0.1196742  -1.439 0.150031
JOBStudent           -0.3997931  0.1461271  -2.736 0.006220 **
JOBUnknown           -0.3927425  0.1845650  -2.128 0.033342 *
KIDSDRIV              0.4036578  0.0612752   6.588 4.47e-11 ***
MSTATUSYes           -0.4812207  0.0866989  -5.550 2.85e-08 ***
MVR_PTS               0.1020655  0.0140545   7.262 3.81e-13 ***
LOG_OLDCLAIM          0.0225774  0.0124662   1.811 0.070128 .
PARENT1Yes            0.3747205  0.1096208   3.418 0.000630 ***
RED_CARYes           -0.0149001  0.0863243  -0.173 0.862961
REVOKEDYes            0.7000443  0.0814072   8.599  < 2e-16 ***
SEXM                  0.1102954  0.1081743   1.020 0.307914
TIF                  -0.0545220  0.0073385  -7.430 1.09e-13 ***
TRAVTIME              0.0148258  0.0018877   7.854 4.03e-15 ***
URBANICITYUrban      -2.3978262  0.1138054 -21.070  < 2e-16 ***
YOJ                   0.0202910  0.0108561   1.869 0.061611 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9418  on 8160  degrees of freedom
Residual deviance: 7289  on 8123  degrees of freedom
AIC: 7365

Number of Fisher Scoring iterations: 5
```

Diagnostics Plots



Model 2: Stepwise Selected Log Transformed Model

We applied backward selection to the Log Transformed Model (model 1) to obtain a simplified model that removes the majority of the variables with little statistical significance. while trying to find a low AIC value. The AIC value for model 2 is 7357.7 while our original log transformed model had an AIC value of 7364.97. AIC is just one measure that we will use to evaluate and select our final model in section 4 of this work, but it is worth noting here as a preliminary point of comparison as we build our models.

```
Start:  AIC=7364.97
TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_AGE + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL + LOG_INCOME +
    JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 +
    RED_CAR + REVOKED + SEX + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- CAR_AGE       1   7289.0 7363.0
- RED_CAR       1   7289.0 7363.0
- HOMEKIDS      1   7289.4 7363.4
- AGE           1   7289.7 7363.7
- SEX           1   7290.0 7364.0
<none>              7289.0 7365.0
- LOG_OLDCLAIM  1   7292.2 7366.2
- YOJ           1   7292.5 7366.5
- CLM_FREQ      1   7293.3 7367.3
- PARENT1       1   7300.7 7374.7
- LOG_HOME_VAL  1   7306.9 7380.9
- EDUCATION     4   7314.1 7382.1
- LOG_BLUEBOOK  1   7316.5 7390.5
- LOG_INCOME    1   7317.5 7391.5
- MSTATUS       1   7319.4 7393.4
- KIDSDRIV      1   7332.4 7406.4
- MVR_PTS       1   7342.1 7416.1
- TIF           1   7346.2 7420.2
- TRAVTIME      1   7350.9 7424.9
- JOB           8   7366.2 7426.2
- CAR_USE       1   7357.1 7431.1
- REVOKED       1   7361.7 7435.7
- CAR_TYPE      5   7382.7 7448.7
- URBANICITY    1   7926.6 8000.6

Step:  AIC=7362.99
TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_TYPE + CAR_USE + CLM_FREQ +
    EDUCATION + HOMEKIDS + LOG_HOME_VAL + LOG_INCOME + JOB +
```

```
    KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + RED_CAR +
    REVOKED + SEX + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- RED_CAR       1   7289.0 7361.0
- HOMEKIDS      1   7289.4 7361.4
- AGE           1   7289.7 7361.7
- SEX           1   7290.0 7362.0
<none>              7289.0 7363.0
- LOG_OLDCLAIM  1   7292.3 7364.3
- YOJ           1   7292.5 7364.5
- CLM_FREQ      1   7293.3 7365.3
- PARENT1       1   7300.7 7372.7
- LOG_HOME_VAL  1   7306.9 7378.9
- EDUCATION     4   7319.7 7385.7
- LOG_BLUEBOOK  1   7316.5 7388.5
- LOG_INCOME    1   7317.6 7389.6
- MSTATUS       1   7319.4 7391.4
- KIDSDRIV      1   7332.4 7404.4
- MVR_PTS       1   7342.1 7414.1
- TIF           1   7346.2 7418.2
- TRAVTIME      1   7350.9 7422.9
- JOB           8   7366.2 7424.2
- CAR_USE       1   7357.2 7429.2
- REVOKED       1   7361.7 7433.7
- CAR_TYPE      5   7382.8 7446.8
- URBANICITY    1   7926.6 7998.6

Step:  AIC=7361.02
TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_TYPE + CAR_USE + CLM_FREQ +
    EDUCATION + HOMEKIDS + LOG_HOME_VAL + LOG_INCOME + JOB +
    KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED +
    SEX + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- HOMEKIDS      1   7289.4 7359.4
- AGE           1   7289.8 7359.8
- SEX           1   7290.2 7360.2
<none>              7289.0 7361.0
- LOG_OLDCLAIM  1   7292.3 7362.3
- YOJ           1   7292.5 7362.5
- CLM_FREQ      1   7293.3 7363.3
- PARENT1       1   7300.8 7370.8
```

```
- LOG_HOME_VAL  1   7306.9 7376.9
- EDUCATION     4   7319.8 7383.8
- LOG_BLUEBOOK  1   7316.5 7386.5
- LOG_INCOME    1   7317.6 7387.6
- MSTATUS       1   7319.4 7389.4
- KIDSDRIV      1   7332.5 7402.5
- MVR_PTS       1   7342.1 7412.1
- TIF           1   7346.2 7416.2
- TRAVTIME      1   7350.9 7420.9
- JOB           8   7366.4 7422.4
- CAR_USE       1   7357.2 7427.2
- REVOKED       1   7361.7 7431.7
- CAR_TYPE      5   7383.0 7445.0
- URBANICITY    1   7926.6 7996.6

Step:  AIC=7359.43
TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_TYPE + CAR_USE + CLM_FREQ +
    EDUCATION + LOG_HOME_VAL + LOG_INCOME + JOB + KIDSDRIV +
    MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED + SEX +
    TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- SEX           1   7290.5 7358.5
- AGE           1   7290.9 7358.9
<none>              7289.4 7359.4
- LOG_OLDCLAIM  1   7292.7 7360.7
- CLM_FREQ      1   7293.7 7361.7
- YOJ           1   7293.8 7361.8
- PARENT1       1   7305.9 7373.9
- LOG_HOME_VAL  1   7307.4 7375.4
- EDUCATION     4   7320.2 7382.2
- LOG_BLUEBOOK  1   7317.0 7385.0
- LOG_INCOME    1   7319.4 7387.4
- MSTATUS       1   7319.8 7387.8
- MVR_PTS       1   7342.6 7410.6
- TIF           1   7346.5 7414.5
- KIDSDRIV      1   7347.2 7415.2
- TRAVTIME      1   7351.2 7419.2
- JOB           8   7366.8 7420.8
- CAR_USE       1   7357.7 7425.7
- REVOKED       1   7362.4 7430.4
- CAR_TYPE      5   7383.6 7443.6
- URBANICITY    1   7926.8 7994.8
```

```
Step:  AIC=7358.54
TARGET_FLAG ~ AGE + LOG_BLUEBOOK + CAR_TYPE + CAR_USE + CLM_FREQ +
    EDUCATION + LOG_HOME_VAL + LOG_INCOME + JOB + KIDSDRIV +
    MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED + TIF +
    TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- AGE           1   7291.7 7357.7
<none>              7290.5 7358.5
- LOG_OLDCLAIM  1   7293.8 7359.8
- CLM_FREQ      1   7294.9 7360.9
- YOJ           1   7294.9 7360.9
- PARENT1       1   7306.9 7372.9
- LOG_HOME_VAL  1   7308.4 7374.4
- EDUCATION     4   7321.4 7381.4
- LOG_INCOME    1   7320.7 7386.7
- MSTATUS       1   7321.0 7387.0
- LOG_BLUEBOOK  1   7326.6 7392.6
- MVR_PTS       1   7343.6 7409.6
- TIF           1   7347.6 7413.6
- KIDSDRIV      1   7347.9 7413.9
- TRAVTIME      1   7352.5 7418.5
- JOB           8   7367.7 7419.7
- CAR_USE       1   7358.5 7424.5
- REVOKED       1   7363.8 7429.8
- CAR_TYPE      5   7400.2 7458.2
- URBANICITY    1   7928.3 7994.3

Step:  AIC=7357.71
TARGET_FLAG ~ LOG_BLUEBOOK + CAR_TYPE + CAR_USE + CLM_FREQ +
    EDUCATION + LOG_HOME_VAL + LOG_INCOME + JOB + KIDSDRIV +
    MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED + TIF +
    TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
<none>              7291.7 7357.7
- LOG_OLDCLAIM  1   7295.0 7359.0
- YOJ           1   7295.6 7359.6
- CLM_FREQ      1   7296.0 7360.0
- LOG_HOME_VAL  1   7309.9 7373.9
- PARENT1       1   7312.9 7376.9
- EDUCATION     4   7323.0 7381.0
```

```
- LOG_INCOME    1   7321.1 7385.1
- MSTATUS       1   7321.4 7385.4
- LOG_BLUEBOOK  1   7329.5 7393.5
- MVR_PTS       1   7345.4 7409.4
- TIF           1   7348.6 7412.6
- KIDSDRIV      1   7348.8 7412.8
- TRAVTIME      1   7353.2 7417.2
- JOB           8   7370.3 7420.3
- CAR_USE       1   7359.4 7423.4
- REVOKED       1   7365.3 7429.3
- CAR_TYPE      5   7400.6 7456.6
- URBANICITY    1   7931.1 7995.1


Call:
glm(formula = TARGET_FLAG ~ LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + LOG_HOME_VAL + LOG_INCOME + JOB +
    KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED +
    TIF + TRAVTIME + URBANICITY + YOJ, family = binomial, data = training_df)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)          3.215131   0.549675   5.849 4.94e-09 ***
LOG_BLUEBOOK        -0.338298   0.054800  -6.173 6.69e-10 ***
CAR_TYPEPanel Truck  0.533434   0.143997   3.704 0.000212 ***
CAR_TYPEPickup       0.587607   0.100499   5.847 5.01e-09 ***
CAR_TYPESports Car   0.950929   0.107973   8.807  < 2e-16 ***
CAR_TYPESUV          0.739915   0.085747   8.629  < 2e-16 ***
CAR_TYPEVan          0.651315   0.121822   5.346 8.97e-08 ***
CAR_USEPrivate      -0.751293   0.092016  -8.165 3.22e-16 ***
CLM_FREQ             0.089724   0.043461   2.064 0.038973 *
EDUCATIONBachelors  -0.419593   0.107943  -3.887 0.000101 ***
EDUCATIONHigh School 0.023430   0.095139   0.246 0.805470
EDUCATIONMasters    -0.381190   0.160061  -2.382 0.017241 *
EDUCATIONPhD        -0.419251   0.192054  -2.183 0.029037 *
LOG_HOME_VAL        -0.029360   0.006897  -4.257 2.07e-05 ***
LOG_INCOME          -0.093978   0.017465  -5.381 7.41e-08 ***
JOBClerical          0.145110   0.105466   1.376 0.168855
JOBDoctor           -0.783083   0.285877  -2.739 0.006158 **
JOBHome Maker       -0.277361   0.163372  -1.698 0.089560 .
JOBLawyer           -0.235519   0.187401  -1.257 0.208838
JOBManager          -0.916285   0.139426  -6.572 4.97e-11 ***
```

```
JOBProfessional      -0.181549   0.119448  -1.520 0.128537    
JOBStudent           -0.392047   0.145675  -2.691 0.007119 ** 
JOBUnknown           -0.394500   0.184473  -2.139 0.032474 *  
KIDSDRIV              0.418492   0.055100   7.595 3.08e-14 ***
MSTATUSYes           -0.461593   0.084059  -5.491 3.99e-08 ***
MVR_PTS               0.102545   0.014041   7.303 2.81e-13 ***
LOG_OLDCLAIM          0.022779   0.012461   1.828 0.067558 .  
PARENT1Yes            0.435845   0.094725   4.601 4.20e-06 ***
REVOKEDYes            0.704002   0.081371   8.652  < 2e-16 ***
TIF                  -0.054339   0.007335  -7.408 1.29e-13 ***
TRAVTIME              0.014768   0.001886   7.829 4.91e-15 ***
URBANICITYUrban      -2.401382   0.113860 -21.091  < 2e-16 ***
YOJ                   0.020598   0.010437   1.974 0.048429 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9418.0  on 8160  degrees of freedom
Residual deviance: 7291.7  on 8128  degrees of freedom
AIC: 7357.7

Number of Fisher Scoring iterations: 5

[1] 7357.708
```

Model 3: Using Categorical Groupings

For model 3, we substituted AGE and INCOME with our categorically grouped variables AGE_GROUP and INCOME_GROUP and used all of the other predictors from our Log Transformed model. We again used backward selection to obtaina simplified model. This model yields an AIC value of 7327.1

```
Start:  AIC=7335.06
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_AGE + CAR_TYPE + 
    CAR_USE + CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL + 
    INCOME_GROUP + JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + 
    PARENT1 + RED_CAR + REVOKED + SEX + TIF + TRAVTIME + URBANICITY + 
    YOJ

               Df Deviance    AIC
- CAR_AGE       1   7251.1 7333.1
```

```
- SEX           1   7251.2 7333.2
- HOMEKIDS      1   7251.3 7333.3
- RED_CAR       1   7251.3 7333.3
- YOJ           1   7252.8 7334.8
<none>              7251.1 7335.1
- LOG_OLDCLAIM  1   7254.3 7336.3
- CLM_FREQ      1   7255.3 7337.3
- PARENT1       1   7260.0 7342.0
- EDUCATION     4   7271.5 7347.5
- LOG_HOME_VAL  1   7267.8 7349.8
- INCOME_GROUP  3   7279.6 7357.6
- MSTATUS       1   7279.8 7361.8
- LOG_BLUEBOOK  1   7280.0 7362.0
- AGE_GROUP     3   7284.9 7362.9
- MVR_PTS       1   7298.0 7380.0
- JOB           8   7315.1 7383.1
- KIDSDRIV      1   7302.8 7384.8
- TIF           1   7311.0 7393.0
- TRAVTIME      1   7313.2 7395.2
- CAR_USE       1   7322.0 7404.0
- CAR_TYPE      5   7331.3 7405.3
- REVOKED       1   7324.7 7406.7
- URBANICITY    1   7889.9 7971.9

Step:  AIC=7333.07
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL + INCOME_GROUP +
    JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 +
    RED_CAR + REVOKED + SEX + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- SEX           1   7251.2 7331.2
- HOMEKIDS      1   7251.3 7331.3
- RED_CAR       1   7251.3 7331.3
- YOJ           1   7252.8 7332.8
<none>              7251.1 7333.1
- LOG_OLDCLAIM  1   7254.3 7334.3
- CLM_FREQ      1   7255.3 7335.3
- PARENT1       1   7260.0 7340.0
- LOG_HOME_VAL  1   7267.8 7347.8
- EDUCATION     4   7274.7 7348.7
- INCOME_GROUP  3   7279.6 7355.6
- MSTATUS       1   7279.8 7359.8
```

```
- LOG_BLUEBOOK  1   7280.0 7360.0
- AGE_GROUP     3   7284.9 7360.9
- MVR_PTS       1   7298.0 7378.0
- JOB           8   7315.1 7381.1
- KIDSDRIV      1   7302.8 7382.8
- TIF           1   7311.1 7391.1
- TRAVTIME      1   7313.2 7393.2
- CAR_USE       1   7322.0 7402.0
- CAR_TYPE      5   7331.4 7403.4
- REVOKED       1   7324.8 7404.8
- URBANICITY    1   7889.9 7969.9

Step:  AIC=7331.25
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL + INCOME_GROUP +
    JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 +
    RED_CAR + REVOKED + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- RED_CAR       1   7251.4 7329.4
- HOMEKIDS      1   7251.5 7329.5
- YOJ           1   7253.0 7331.0
<none>              7251.2 7331.2
- LOG_OLDCLAIM  1   7254.5 7332.5
- CLM_FREQ      1   7255.4 7333.4
- PARENT1       1   7260.2 7338.2
- LOG_HOME_VAL  1   7268.0 7346.0
- EDUCATION     4   7274.9 7346.9
- INCOME_GROUP  3   7279.9 7353.9
- MSTATUS       1   7280.1 7358.1
- AGE_GROUP     3   7285.5 7359.5
- LOG_BLUEBOOK  1   7284.9 7362.9
- MVR_PTS       1   7298.1 7376.1
- JOB           8   7315.2 7379.2
- KIDSDRIV      1   7303.0 7381.0
- TIF           1   7311.2 7389.2
- TRAVTIME      1   7313.5 7391.5
- CAR_USE       1   7322.1 7400.1
- REVOKED       1   7325.0 7403.0
- CAR_TYPE      5   7339.0 7409.0
- URBANICITY    1   7890.2 7968.2

Step:  AIC=7329.38
```

```
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + HOMEKIDS + LOG_HOME_VAL + INCOME_GROUP +
    JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 +
    REVOKED + TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- HOMEKIDS      1   7251.6 7327.6
- YOJ           1   7253.1 7329.1
<none>              7251.4 7329.4
- LOG_OLDCLAIM  1   7254.6 7330.6
- CLM_FREQ      1   7255.5 7331.5
- PARENT1       1   7260.4 7336.4
- LOG_HOME_VAL  1   7268.1 7344.1
- EDUCATION     4   7275.1 7345.1
- INCOME_GROUP  3   7280.0 7352.0
- MSTATUS       1   7280.1 7356.1
- AGE_GROUP     3   7285.6 7357.6
- LOG_BLUEBOOK  1   7285.6 7361.6
- MVR_PTS       1   7298.3 7374.3
- JOB           8   7315.6 7377.6
- KIDSDRIV      1   7303.2 7379.2
- TIF           1   7311.3 7387.3
- TRAVTIME      1   7313.6 7389.6
- CAR_USE       1   7322.3 7398.3
- REVOKED       1   7325.1 7401.1
- CAR_TYPE      5   7348.8 7416.8
- URBANICITY    1   7890.2 7966.2

Step:  AIC=7327.58
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + LOG_HOME_VAL + INCOME_GROUP + JOB +
    KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED +
    TIF + TRAVTIME + URBANICITY + YOJ

               Df Deviance    AIC
- YOJ           1   7253.1 7327.1
<none>              7251.6 7327.6
- LOG_OLDCLAIM  1   7254.9 7328.9
- CLM_FREQ      1   7255.7 7329.7
- PARENT1       1   7264.3 7338.3
- LOG_HOME_VAL  1   7268.3 7342.3
- EDUCATION     4   7275.5 7343.5
- INCOME_GROUP  3   7280.1 7350.1
```

```
- MSTATUS       1   7280.9 7354.9
- AGE_GROUP     3   7288.0 7358.0
- LOG_BLUEBOOK  1   7285.9 7359.9
- MVR_PTS       1   7298.6 7372.6
- JOB           8   7316.0 7376.0
- TIF           1   7311.4 7385.4
- TRAVTIME      1   7313.7 7387.7
- KIDSDRIV      1   7318.1 7392.1
- CAR_USE       1   7322.6 7396.6
- REVOKED       1   7325.6 7399.6
- CAR_TYPE      5   7349.2 7415.2
- URBANICITY    1   7890.3 7964.3

Step:  AIC=7327.13
TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE + CAR_USE +
    CLM_FREQ + EDUCATION + LOG_HOME_VAL + INCOME_GROUP + JOB +
    KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED +
    TIF + TRAVTIME + URBANICITY

               Df Deviance    AIC
<none>              7253.1 7327.1
- LOG_OLDCLAIM  1   7256.3 7328.3
- CLM_FREQ      1   7257.4 7329.4
- PARENT1       1   7265.3 7337.3
- LOG_HOME_VAL  1   7270.2 7342.2
- EDUCATION     4   7276.8 7342.8
- INCOME_GROUP  3   7282.1 7350.1
- MSTATUS       1   7284.5 7356.5
- AGE_GROUP     3   7289.7 7357.7
- LOG_BLUEBOOK  1   7288.2 7360.2
- MVR_PTS       1   7300.8 7372.8
- JOB           8   7317.4 7375.4
- TIF           1   7313.4 7385.4
- TRAVTIME      1   7315.1 7387.1
- KIDSDRIV      1   7319.0 7391.0
- CAR_USE       1   7324.8 7396.8
- REVOKED       1   7327.1 7399.1
- CAR_TYPE      5   7351.1 7415.1
- URBANICITY    1   7891.3 7963.3


Call:
```

```
glm(formula = TARGET_FLAG ~ AGE_GROUP + LOG_BLUEBOOK + CAR_TYPE +
    CAR_USE + CLM_FREQ + EDUCATION + LOG_HOME_VAL + INCOME_GROUP +
    JOB + KIDSDRIV + MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 +
    REVOKED + TIF + TRAVTIME + URBANICITY, family = binomial,
    data = training_df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)    
(Intercept)            2.572422   0.557326   4.616 3.92e-06 ***
AGE_GROUPMiddle Aged  -0.232349   0.066678  -3.485 0.000493 ***
AGE_GROUPSenior        0.300307   0.171924   1.747 0.080683 .  
AGE_GROUPYoung         0.899643   0.249159   3.611 0.000305 ***
LOG_BLUEBOOK          -0.330860   0.055608  -5.950 2.68e-09 ***
CAR_TYPEPanel Truck    0.567336   0.145279   3.905 9.42e-05 ***
CAR_TYPEPickup         0.574791   0.101034   5.689 1.28e-08 ***
CAR_TYPESports Car     0.888626   0.110084   8.072 6.90e-16 ***
CAR_TYPESUV            0.714288   0.086206   8.286  < 2e-16 ***
CAR_TYPEVan            0.677169   0.122795   5.515 3.50e-08 ***
CAR_USEPrivate        -0.773510   0.092124  -8.396  < 2e-16 ***
CLM_FREQ               0.089701   0.043394   2.067 0.038722 *  
EDUCATIONBachelors    -0.392148   0.112751  -3.478 0.000505 ***
EDUCATIONHigh School   0.011398   0.097157   0.117 0.906612    
EDUCATIONMasters      -0.283303   0.163842  -1.729 0.083787 .  
EDUCATIONPhD          -0.261605   0.196787  -1.329 0.183722    
LOG_HOME_VAL          -0.028499   0.006924  -4.116 3.85e-05 ***
INCOME_GROUPLow        0.141398   0.116152   1.217 0.223472    
INCOME_GROUPMedium    -0.006093   0.086433  -0.070 0.943804    
INCOME_GROUPVery High -0.437946   0.095255  -4.598 4.27e-06 ***
JOBClerical            0.078247   0.110581   0.708 0.479193    
JOBDoctor             -0.677301   0.286587  -2.363 0.018111 *  
JOBHome Maker          0.027213   0.155355   0.175 0.860948    
JOBLawyer             -0.194872   0.188377  -1.034 0.300912    
JOBManager            -0.878295   0.140440  -6.254 4.00e-10 ***
JOBProfessional       -0.148125   0.120502  -1.229 0.218983    
JOBStudent            -0.153824   0.141134  -1.090 0.275751    
JOBUnknown            -0.351367   0.185572  -1.893 0.058302 .  
KIDSDRIV               0.450473   0.055226   8.157 3.44e-16 ***
MSTATUSYes            -0.470878   0.083387  -5.647 1.63e-08 ***
MVR_PTS                0.097055   0.014099   6.884 5.84e-12 ***
LOG_OLDCLAIM           0.022180   0.012473   1.778 0.075372 .  
PARENT1Yes             0.344280   0.098548   3.494 0.000477 ***
REVOKEDYes             0.707333   0.081595   8.669  < 2e-16 ***
TIF                   -0.056154   0.007371  -7.618 2.58e-14 ***
```

```
TRAVTIME               0.014841   0.001890   7.854 4.03e-15 ***
URBANICITYUrban       -2.397969   0.113729 -21.085  < 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: 9418.0  on 8160  degrees of freedom
Residual deviance: 7253.1  on 8124  degrees of freedom
AIC: 7327.1

Number of Fisher Scoring iterations: 5

[1] 7327.134
```

Model 4: Using One-Hot Encoded Parameters

Building off of the predictors we obtained in model 3, uses one-hot encoded predictors for AGE_GROUP, INCOME_GROUP, EDUCATION, and JOB to remove predictors with little statistical significance from our model. Our AIC value has increased a bit to 7354.6 but our model is slightly simpler.

```
Call:
glm(formula = TARGET_FLAG ~ `AGE_GROUPMiddle Aged` + LOG_BLUEBOOK +
    CAR_AGE + CAR_TYPE + CAR_USE + CLM_FREQ + EDUCATIONBachelors +
    LOG_HOME_VAL + `INCOME_GROUPVery High` + JOBManager + KIDSDRIV +
    MSTATUS + MVR_PTS + LOG_OLDCLAIM + PARENT1 + REVOKED + TIF +
    TRAVTIME + URBANICITY + YOJ, family = binomial, data = df_training_one_hot)

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             2.967478   0.523193   5.672 1.41e-08 ***
`AGE_GROUPMiddle Aged` -0.320530   0.062697  -5.112 3.18e-07 ***
LOG_BLUEBOOK           -0.347126   0.054152  -6.410 1.45e-10 ***
CAR_AGE                -0.022936   0.005723  -4.008 6.13e-05 ***
CAR_TYPEPanel Truck     0.473037   0.135014   3.504 0.000459 ***
CAR_TYPEPickup          0.538859   0.097886   5.505 3.69e-08 ***
CAR_TYPESports Car      0.898541   0.106904   8.405  < 2e-16 ***
CAR_TYPESUV             0.711022   0.084830   8.382  < 2e-16 ***
CAR_TYPEVan             0.610915   0.119539   5.111 3.21e-07 ***
CAR_USEPrivate         -0.800728   0.070734 -11.320  < 2e-16 ***
```

```
CLM_FREQ                 0.081032   0.043068   1.881 0.059906 .
EDUCATIONBachelors      -0.252133   0.067815  -3.718 0.000201 ***
LOG_HOME_VAL            -0.026950   0.006316  -4.267 1.98e-05 ***
`INCOME_GROUPVery High` -0.630592   0.080374  -7.846 4.30e-15 ***
JOBManager              -0.777864   0.105977  -7.340 2.14e-13 ***
KIDSDRIV                 0.449855   0.054944   8.187 2.67e-16 ***
MSTATUSYes              -0.433265   0.080233  -5.400 6.66e-08 ***
MVR_PTS                  0.099115   0.013984   7.088 1.36e-12 ***
LOG_OLDCLAIM             0.025009   0.012395   2.018 0.043617 *
PARENT1Yes               0.391078   0.097158   4.025 5.69e-05 ***
REVOKEDYes               0.704209   0.081123   8.681  < 2e-16 ***
TIF                     -0.054918   0.007339  -7.483 7.28e-14 ***
TRAVTIME                 0.015043   0.001880   8.000 1.25e-15 ***
URBANICITYUrban         -2.333909   0.112827 -20.686  < 2e-16 ***
YOJ                     -0.012554   0.007470  -1.681 0.092856 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9418.0  on 8160  degrees of freedom
Residual deviance: 7304.6  on 8136  degrees of freedom
AIC: 7354.6

Number of Fisher Scoring iterations: 5

[1] 7354.647
```

Model 5: Using Selected Features

This model uses a selection of features based on our exploratory charts. In particular, categorical demographic information was selected based on the single group with the highest numerical number of crashes. Additionally, we have added LOG_HOME_VAL, TIF, URBANCITY, HIGH_RISK_CAR, REVOKED and an interaction term between CLM_FREQ and MVR_PTS. The latter four parameters are used in our weighted RISK_SCORE param, but replacing these paramters with the former results in a higher AIC value.

```
Call:
glm(formula = TARGET_FLAG ~ HIGH_RISK_CAR + CLM_FREQ * MVR_PTS +
    REVOKED + LOG_HOME_VAL + TIF + URBANICITY + `AGE_GROUPMiddle Aged` +
    INCOME_GROUPLow + `EDUCATIONHigh School` + `JOBBlue Collar` +
```

```
    PARENT1, family = binomial, data = df_training_one_hot)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)    
(Intercept)           -1.171159   0.096613 -12.122  < 2e-16 ***
HIGH_RISK_CAR          0.355659   0.085846   4.143 3.43e-05 ***
CLM_FREQ               0.285430   0.032681   8.734  < 2e-16 ***
MVR_PTS                0.195779   0.020400   9.597  < 2e-16 ***
REVOKEDYes             0.750040   0.077011   9.739  < 2e-16 ***
LOG_HOME_VAL          -0.045031   0.005247  -8.583  < 2e-16 ***
TIF                   -0.049764   0.007093  -7.016 2.28e-12 ***
URBANICITYUrban       -1.981975   0.108175 -18.322  < 2e-16 ***
`AGE_GROUPMiddle Aged` -0.285416   0.060304  -4.733 2.21e-06 ***
INCOME_GROUPLow        0.602741   0.067671   8.907  < 2e-16 ***
`EDUCATIONHigh School`  0.467429   0.062973   7.423 1.15e-13 ***
`JOBBlue Collar`       0.690839   0.068406  10.099  < 2e-16 ***
PARENT1Yes             0.646915   0.082587   7.833 4.76e-15 ***
CLM_FREQ:MVR_PTS      -0.049274   0.010849  -4.542 5.57e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9418.0  on 8160  degrees of freedom
Residual deviance: 7760.6  on 8147  degrees of freedom
AIC: 7788.6

Number of Fisher Scoring iterations: 5

[1] 7788.643
```

Model : MQ's Logit model

```
Call:
glm(formula = TARGET_FLAG ~ CAR_TYPE + CAR_USE + EDUCATION + 
    INCOME_LOG + JOB + KIDSDRIV + MSTATUS + MVR_PTS + OLDCLAIM_LOG + 
    PARENT1 + REVOKED + TIF + TRAVTIME + URBANICITY + YOJ, family = binomial, 
    data = insurance_training_clean)

Coefficients:
```

```
                     Estimate Std. Error z value Pr(>|z|)    
(Intercept)         -0.037801   0.209808  -0.180  0.85702    
CAR_TYPEPanel Truck  0.294173   0.138090   2.130  0.03315 *  
CAR_TYPEPickup       0.638988   0.099535   6.420 1.37e-10 ***
CAR_TYPESports Car   1.050250   0.106206   9.889  < 2e-16 ***
CAR_TYPESUV          0.801973   0.084784   9.459  < 2e-16 ***
CAR_TYPEVan          0.523990   0.119397   4.389 1.14e-05 ***
CAR_USEPrivate      -0.726693   0.091465  -7.945 1.94e-15 ***
EDUCATIONBachelors  -0.460832   0.107359  -4.292 1.77e-05 ***
EDUCATIONHigh School 0.007735   0.094752   0.082  0.93494    
EDUCATIONMasters    -0.418406   0.159187  -2.628  0.00858 ** 
EDUCATIONPhD        -0.476026   0.190371  -2.501  0.01240 *  
INCOME_LOG          -0.102679   0.017360  -5.915 3.33e-09 ***
JOBClerical          0.166126   0.104873   1.584  0.11318    
JOBDoctor           -0.796151   0.283724  -2.806  0.00501 ** 
JOBHome Maker       -0.279879   0.162823  -1.719  0.08563 .  
JOBLawyer           -0.273406   0.186521  -1.466  0.14270    
JOBManager          -0.929191   0.138762  -6.696 2.14e-11 ***
JOBProfessional     -0.202469   0.118749  -1.705  0.08819 .  
JOBStudent          -0.143442   0.137864  -1.040  0.29812    
JOBUnknown          -0.383798   0.183417  -2.092  0.03639 *  
KIDSDRIV             0.415107   0.054748   7.582 3.40e-14 ***
MSTATUSYes          -0.651828   0.070222  -9.282  < 2e-16 ***
MVR_PTS              0.103963   0.013958   7.448 9.44e-14 ***
OLDCLAIM_LOG         0.044326   0.007301   6.071 1.27e-09 ***
PARENT1Yes           0.435929   0.094070   4.634 3.59e-06 ***
REVOKEDYes           0.690271   0.080397   8.586  < 2e-16 ***
TIF                 -0.054047   0.007303  -7.401 1.35e-13 ***
TRAVTIME             0.014774   0.001877   7.872 3.48e-15 ***
URBANICITYUrban     -2.385583   0.113518 -21.015  < 2e-16 ***
YOJ                  0.018746   0.010393   1.804  0.07128 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9418.0  on 8160  degrees of freedom
Residual deviance: 7352.5  on 8131  degrees of freedom
AIC: 7412.5

Number of Fisher Scoring iterations: 5
```

Interpreting the Output: Significant p-values (< 0.05) indicate variables that are strong predictors of making a claim.

Model Significance with Deviance Test

```r
null_dev <- logit_model$null.deviance
resid_dev <- logit_model$deviance
df_diff <- logit_model$df.null - logit_model$df.residual
p_val <- 1 - pchisq(null_dev - resid_dev, df_diff)

cat("Model significance p-value:", p_val)
```

```
Model significance p-value: 0
```

```r
# Check each variable
anova(logit_model, test = "Chisq")
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: TARGET_FLAG

Terms added sequentially (first to last)

             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                          8160     9418.0
CAR_TYPE      5   180.07      8155     9237.9 < 2.2e-16 ***
CAR_USE       1   179.75      8154     9058.1 < 2.2e-16 ***
EDUCATION     4   113.92      8150     8944.2 < 2.2e-16 ***
INCOME_LOG    1    43.86      8149     8900.4 3.530e-11 ***
JOB           8    48.18      8141     8852.2 9.108e-08 ***
KIDSDRIV      1    66.54      8140     8785.6 3.435e-16 ***
MSTATUS       1   175.98      8139     8609.7 < 2.2e-16 ***
MVR_PTS       1   270.65      8138     8339.0 < 2.2e-16 ***
OLDCLAIM_LOG  1   167.72      8137     8171.3 < 2.2e-16 ***
PARENT1       1    19.27      8136     8152.0 1.135e-05 ***
REVOKED       1    94.14      8135     8057.9 < 2.2e-16 ***
TIF           1    50.13      8134     8007.8 1.439e-12 ***
TRAVTIME      1    16.43      8133     7991.3 5.041e-05 ***
URBANICITY    1   635.59      8132     7355.7 < 2.2e-16 ***
YOJ           1     3.27      8131     7352.5   0.07064 .
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Run Assumption Tests Using Residuals

Assumption Testing – Logistic

Multicollinearity: Variance Inflation Factor

```r
vif(logit_model)
```

```
                  GVIF Df GVIF^(1/(2*Df))
CAR_TYPE      1.904825  5        1.066561
CAR_USE       2.458052  1        1.567818
EDUCATION     7.321883  4        1.282560
INCOME_LOG    3.584614  1        1.893308
JOB          27.351174  8        1.229733
KIDSDRIV      1.091199  1        1.044605
MSTATUS       1.462453  1        1.209319
MVR_PTS       1.230555  1        1.109304
OLDCLAIM_LOG  1.255965  1        1.120698
PARENT1       1.434180  1        1.197573
REVOKED       1.012436  1        1.006199
TIF           1.008860  1        1.004420
TRAVTIME      1.037331  1        1.018494
URBANICITY    1.151000  1        1.072847
YOJ           2.153507  1        1.467483
```

Explanation: VIF > 5 or 10 indicates multicollinearity. Remove or combine correlated variables.

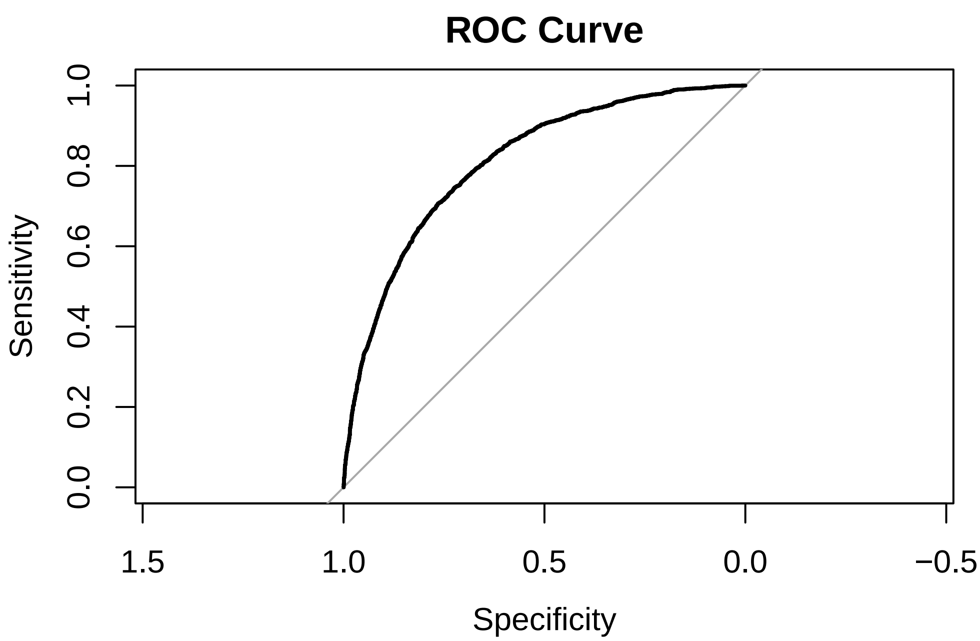
ROC Curve + AUC

```r
# Step 1: Load the library
library(pROC)

# Step 2: Generate predicted probabilities from your logistic model
pred_probs <- predict(logit_model, newdata = insurance_training_clean, type = "response")
```

```r
# Step 3: Create the ROC object
roc_obj <- roc(response = insurance_training_clean$TARGET_FLAG,
               predictor = pred_probs)
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases
```

```r
# Step 4: Plot the ROC Curve
plot(roc_obj, main = "ROC Curve")
```




```r
# Step 5: Get the AUC
pROC::auc(roc_obj)
```

```
Area under the curve: 0.8104
```

```r
identical(length(pred_probs), length(insurance_training_clean$TARGET_FLAG))  # Should return
```

```
[1] TRUE
```

Explanation: Measures model’s ability to distinguish crashers vs. non-crashers. AUC closer to 1 = better.

Confusion Matrix

```
pred_class <- ifelse(pred_probs > 0.5, 1, 0)
confusionMatrix(as.factor(pred_class), as.factor(insurance_training_clean$TARGET_FLAG))
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 5547 1288
         1  461  865

               Accuracy : 0.7857
                 95% CI : (0.7766, 0.7945)
    No Information Rate : 0.7362
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3707

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9233
            Specificity : 0.4018
         Pos Pred Value : 0.8116
         Neg Pred Value : 0.6523
             Prevalence : 0.7362
         Detection Rate : 0.6797
   Detection Prevalence : 0.8375
      Balanced Accuracy : 0.6625

       'Positive' Class : 0
```

Explanation: Helps evaluate accuracy, sensitivity, specificity using 0.5 as threshold.

Crash Model comparison

```
library(vcdExtra)
```

```
Loading required package: vcd
```

```
Loading required package: grid

Loading required package: gnm


Attaching package: 'gnm'

The following object is masked from 'package:Metrics':

    se

The following object is masked from 'package:lattice':

    barley


Attaching package: 'vcdExtra'

The following object is masked from 'package:carData':

    Burt

The following object is masked from 'package:dplyr':

    summarise
```

```r
library(pscl)
```

```
Classes and Methods for R originally developed in the
Political Science Computational Laboratory
Department of Political Science
Stanford University (2002-2015),
by and under the direction of Simon Jackman.
hurdle and zeroinfl functions by Achim Zeileis.
```

```r
stats <- LRstats(model1, model2, model3, model4, model5, logit_model)

stats$McFaddenR2 <- NA
stats$Accuracy <- NA
stats$Precision <- NA
#stats$Recall <- NA
stats$Sensitivity <- NA
stats$Specificity <- NA
stats$F1_score <- NA
stats$AUC <- NA
stats$CV_est_predict_err <- NA
stats$CV_adj_est <- NA

enhanceEvaluationMetrics <- function(df, model_name) {
  model <- get(model_name)

  if (model_name == 'logit_model') {
    model_data <- insurance_training_clean
  } else if (model_name == 'model4') {
    model_data <- df_training_one_hot
  } else  {
    model_data <- training_df
  }

  df[model_name, "McFaddenR2"] <- pR2(model)["McFadden"]
  #pred_probs <- predict(model, type = "response")
  #pred_probs_factor <- as.factor(ifelse(pred_probs > 0.5, 1, 0))
  #conf_matrix <- confusionMatrix(pred_probs_factor, as.factor(model_data$TARGET_FLAG))
  #df[model_name, "Accuracy"] <- conf_matrix$overall['Accuracy']
  #df[model_name, "Precision"] <- conf_matrix$byClass['Precision']
  #df[model_name, "Recall"] <- conf_matrix$byClass['Recall']
  #df[model_name, "F1_score"] <- conf_matrix$byClass['F1']
  #df[model_name, "Sensitivity"] <- conf_matrix$byClass["Sensitivity"]
  #df[model_name, "Specificity"] <- conf_matrix$byClass["Specificity"]

  #df[model_name, "AUC"] <- MLmetrics::AUC(y_true = model_data$target, y_pred = pred_probs)

  # Cross-Validation using 10 folsds
  cv_result <- boot::cv.glm(model_data, model, K= 10)
  df[model_name, "CV_est_predict_err"] <- cv_result$delta[1]
  df[model_name, "CV_adj_est"] <- cv_result$delta[2]
```

```r
  return(df)
}


# Loop through the list of models and update the dataframe for each
for (model_name in rownames(stats)) {

 #stats <- enhanceEvaluationMetrics(stats, model_name)
}

stats
```

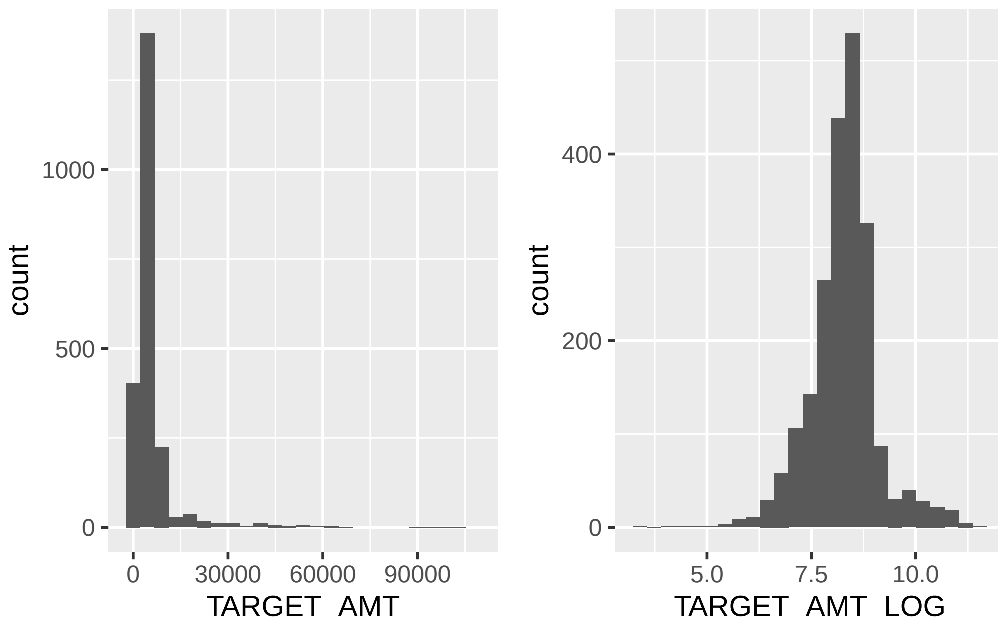
```
Likelihood summary table:
               AIC    BIC LR Chisq   Df Pr(>Chisq) McFaddenR2 Accuracy
model1      7365.0 7631.2   7289.0 8123    1.00000
model2      7357.7 7588.9   7291.7 8128    1.00000
model3      7327.1 7586.4   7253.1 8124    1.00000
model4      7354.6 7529.8   7304.6 8136    1.00000
model5      7788.6 7886.7   7760.6 8147    0.99893
logit_model 7412.5 7622.7   7352.5 8131    1.00000
            Precision Sensitivity Specificity F1_score AUC CV_est_predict_err
model1
model2
model3
model4
model5
logit_model
            CV_adj_est
model1
model2
model3
model4
model5
logit_model
```

Models for TARGET_AMT

Subset of Crashers

Since payouts can't happen without a crash event, we will create a subset where TARGET_FLAG = 1 to avoid biasing our payout predictions towards zero. We will also look at the distribution and review our IQR values. We see that the payouts are highly skewed to the right with a long tail to the right. The median payout is $4,104 and the maximum payout is $107,586. Applying a log transformation to TARGET_AMT gives a normal distribution.

```
   Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
  30.28   2609.78   4104.00   5702.18   5787.00 107586.14 
```



Payout Model 1

Since Blue Book values are used to determine the market rate value of a used car, we can assume that it will affect payout amount. Blue Book values are derived from various attributes, including car age, type, mileage and condition. We will use the logged transformed variable BLUEBOOK_LOG since we had identified that the distribution for original BLUEBOOK was right-skewed.

```
Call:
lm(formula = TARGET_AMT ~ BLUEBOOK_LOG, data = claims_only)

Residuals:
   Min     1Q Median     3Q    Max 
 -6907  -3126  -1586    282 101268 
```

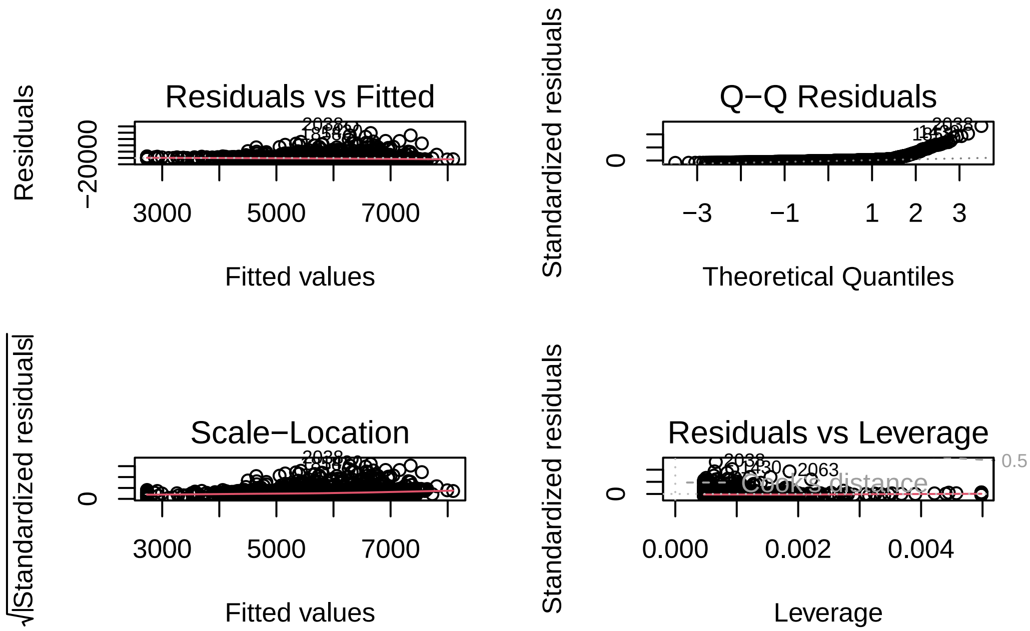
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   -7792.0     2354.3  -3.310 0.000949 ***
BLUEBOOK_LOG   1439.2      250.5   5.746 1.04e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7686 on 2151 degrees of freedom
Multiple R-squared:  0.01512,   Adjusted R-squared:  0.01466
F-statistic: 33.02 on 1 and 2151 DF,  p-value: 1.044e-08
```

```
[1] 44640.52
```

Our model summary indicates that BLUEBOOK_LOG is statistically significant.

Diagnostic Plots

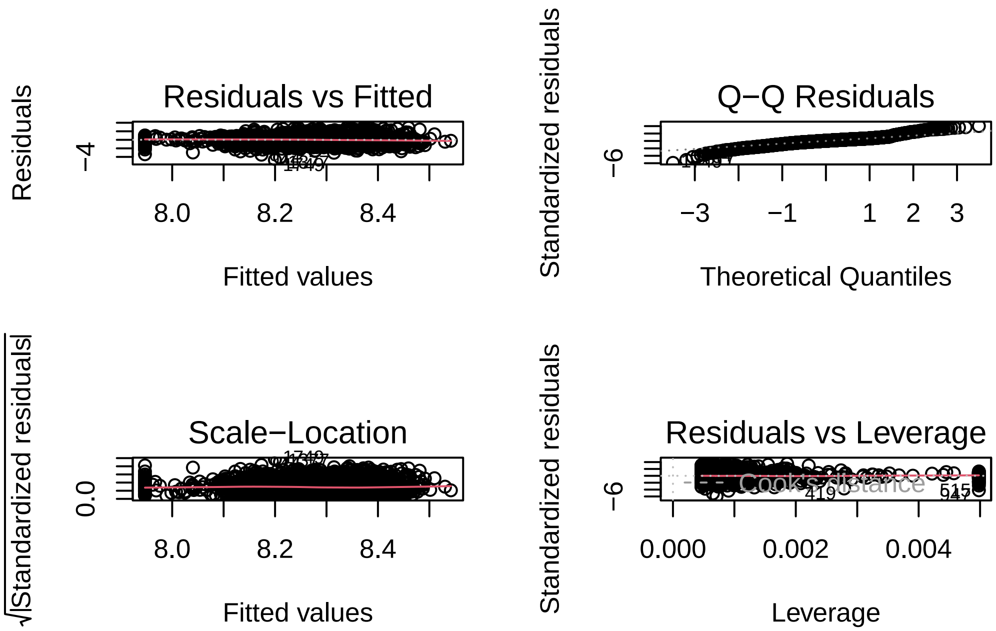
Out diagnostic plots show that our model fails our assumptions for linear regression. In particular, our Residuals vs. Fitted plot shows possible funneling in the positive values, suggesting that our model fails the linearity assumption. Our Q-Q plot shows clear high skewing on the right tail, suggesting that our model fails the normality assumption. The Scale-Location plot does not show evenly scattered points, suggesting we fail our assumption of Homoscedasticity. Our Residual vs. Leverage plot shows some extreme values but they don't appear to have high leverage. A Durbin-Watson value of 1.9844 suggests that our model meets our independence assumption.



Payout Model 2

As our Target Amount value is highly skewed to the right, we will update our model with a log transformed dependent variable (TARGET_AMT_LOG).

```
Call:
lm(formula = TARGET_AMT_LOG ~ BLUEBOOK_LOG, data = claims_only)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.7568 -0.3908  0.0424  0.3912  3.2413 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)   6.77923    0.24669  27.481  < 2e-16 ***
BLUEBOOK_LOG  0.15967    0.02624   6.084 1.38e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8054 on 2151 degrees of freedom
Multiple R-squared:  0.01692,	Adjusted R-squared:  0.01646 
F-statistic: 37.01 on 1 and 2151 DF,  p-value: 1.384e-09
```




```
    Durbin-Watson test

data:  amt_model2
DW = 2.0272, p-value = 0.7356
alternative hypothesis: true autocorrelation is greater than 0


    studentized Breusch-Pagan test

data:  amt_model2
BP = 1.2159, df = 1, p-value = 0.2702


    Shapiro-Wilk normality test

data:  residuals(amt_model2)
W = 0.95891, p-value < 2.2e-16


           df       AIC
amt_model2  3  5181.999
amt_model1  3 44640.523
```

Using the log transformed dependent variable improves our diagnostic plots. Our Residuals vs. Fitted plot shows points fairly evenly scattered around the y-axis, suggesting that our model meets the linearity assumption. The Scale-Location plot shows a fairly even cloud of points around the horizontal line; while the points above the line have a greater spread than those below, the plot suggests that our model is mostly homoscedastic. A Breusch-Pagan test shows that there is evidence of constant variance, supporting that we meet homoscedasticity assumption. A Durbin-Watson value of 1.9844 suggests that our model meets our independence assumption.

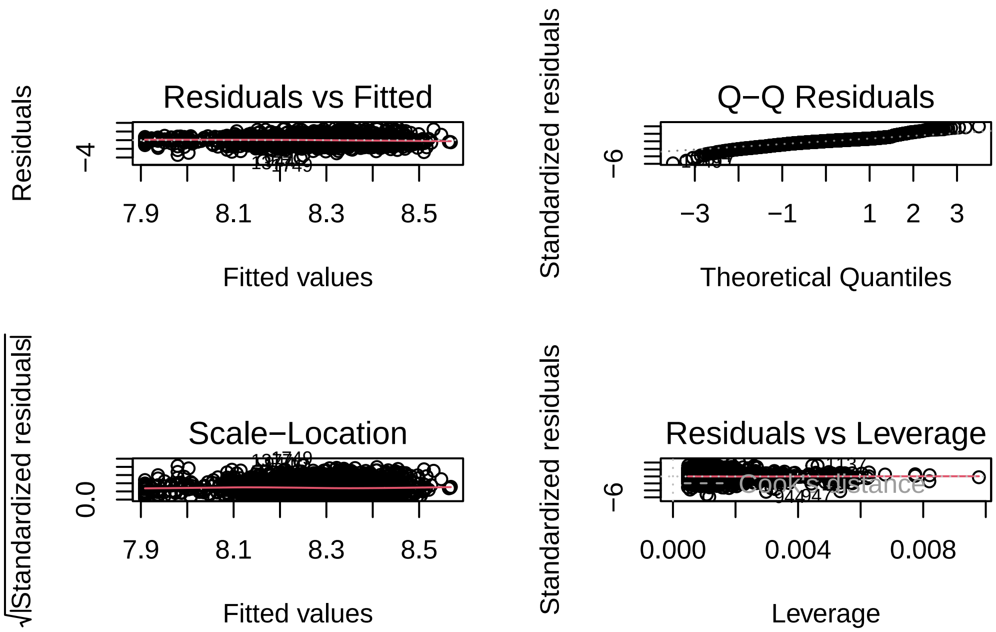
Our Q-Q plot shows skewing at the tails suggesting that our model may not meet our normality assumption. A Shapiro-Wilk normality test with a very low p-value supports this assessment. However, this violation may be acceptable, as we know that our payout values will be right skewed due to lower payouts for lesser accidents such as minor "fender-benders;" there will be some edge cases with higher payouts for accidents with greater damage, such as those involving multiple-vehicles or personal injuries. This is evident in the Residual vs. Leverage plot which shows the presence of extreme values although they don't appear to have high leverage.

Payout Model 3

This model introduces MVR_PTS (Motor Vehicle Record Points) as a predictor under the assumption that drivers with MVR_PTS may be involved in accidents resulting in greater damage or may face a greater share of the financial liability if insurance companies have policies that restrict payout amounts for drivers with higher MVR_PTS. Our corresponding diagnostic plots show similar results as amt_model2.

```
Call:
lm(formula = TARGET_AMT_LOG ~ BLUEBOOK_LOG + MVR_PTS, data = claims_only)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.7210 -0.3962  0.0424  0.3975  3.1770 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.729306   0.247644  27.173  < 2e-16 ***
BLUEBOOK_LOG 0.161265   0.026235   6.147 9.39e-10 ***
MVR_PTS      0.014101   0.006729   2.096   0.0362 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8048 on 2150 degrees of freedom
Multiple R-squared:  0.01892,   Adjusted R-squared:  0.01801
```

```
F-statistic: 20.73 on 2 and 2150 DF,  p-value: 1.207e-09
```



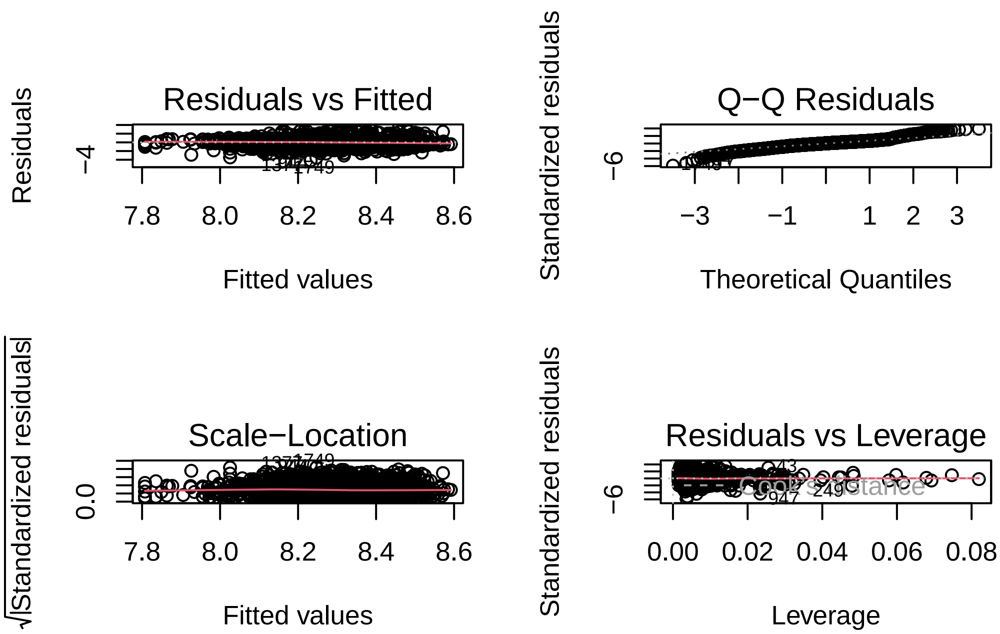

```
           df      AIC
amt_model3  4 5179.606
amt_model2  3 5181.999
```

Payout Model 4

This Model introduces CAR_TYPE as an interaction predictor for BLUEBOOK_LOG to identify if type of car impacts payout amount. Our corresponding diagnostic plots show similar results as amt_model2.

```
Call:
lm(formula = TARGET_AMT_LOG ~ BLUEBOOK_LOG * CAR_TYPE + MVR_PTS,
    data = claims_only)

Residuals:
    Min      1Q  Median      3Q     Max
-4.7245 -0.4073  0.0389  0.3954  3.1284

Coefficients:
```

```
                                     Estimate Std. Error t value Pr(>|t|)
(Intercept)                          6.319299   0.594755  10.625  < 2e-16 ***
BLUEBOOK_LOG                         0.203460   0.063217   3.218  0.00131 **
CAR_TYPEPanel Truck                  1.561888   3.011251   0.519  0.60403
CAR_TYPEPickup                       0.084950   0.905087   0.094  0.92523
CAR_TYPESports Car                   0.716695   0.826611   0.867  0.38602
CAR_TYPESUV                          0.739925   0.768180   0.963  0.33555
CAR_TYPEVan                          6.343200   3.161910   2.006  0.04497 *
MVR_PTS                              0.014017   0.006753   2.076  0.03806 *
BLUEBOOK_LOG:CAR_TYPEPanel Truck    -0.149147   0.294372  -0.507  0.61245
BLUEBOOK_LOG:CAR_TYPEPickup         -0.005257   0.097274  -0.054  0.95690
BLUEBOOK_LOG:CAR_TYPESports Car     -0.077733   0.089114  -0.872  0.38315
BLUEBOOK_LOG:CAR_TYPESUV            -0.078511   0.082375  -0.953  0.34065
BLUEBOOK_LOG:CAR_TYPEVan            -0.642343   0.319654  -2.009  0.04461 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8054 on 2140 degrees of freedom
Multiple R-squared:  0.02194,   Adjusted R-squared:  0.01645
F-statistic:     4 on 12 and 2140 DF,  p-value: 3.704e-06
```




```
          df       AIC
```

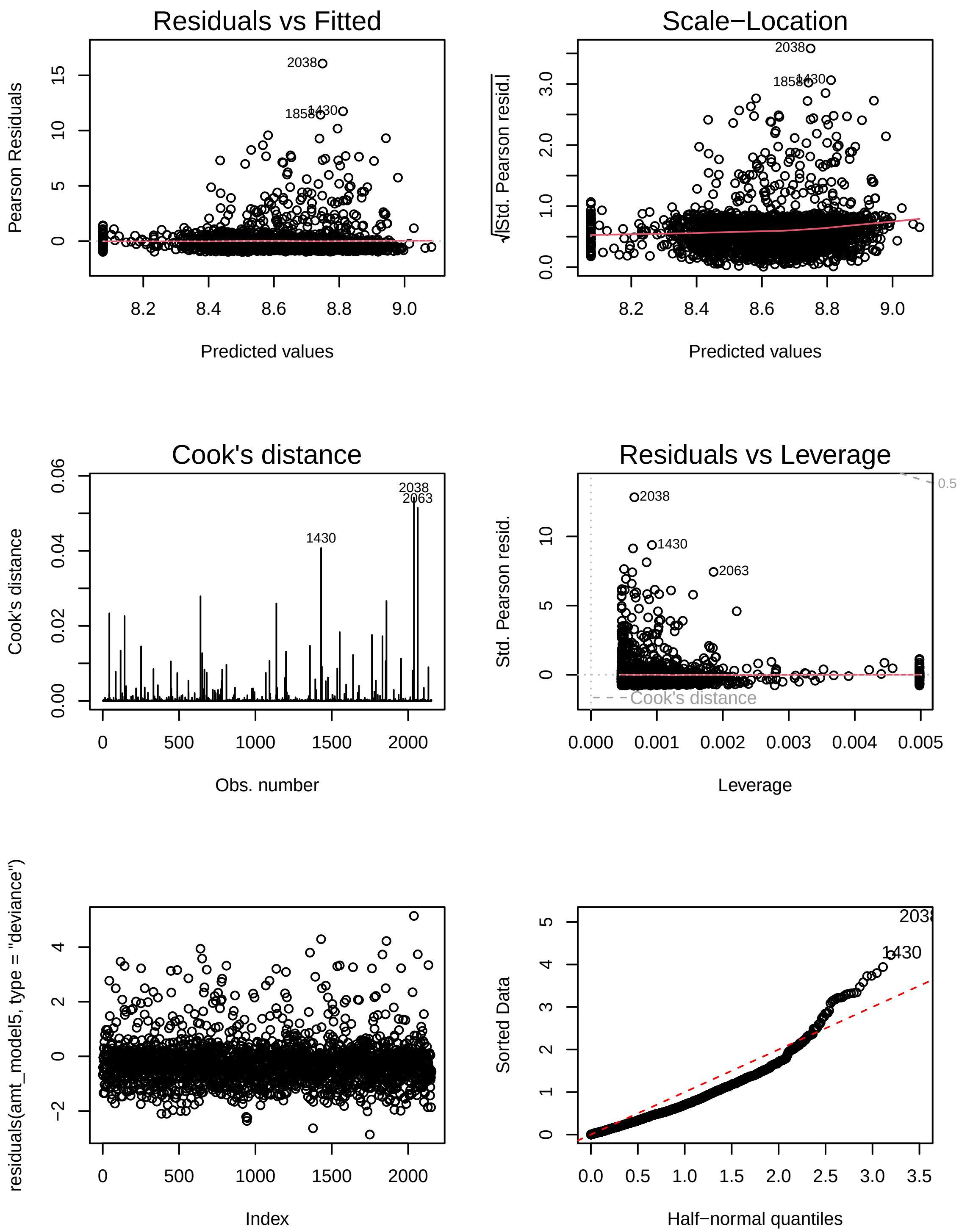
```
amt_model4 14 5192.980
amt_model3  4 5179.606
```

Payout Model 5: Gamma GLM

Given that our dependent payout variable TARGET_AMT is highly right skewed and that out OLS models fail the linearity assumption, we can attempt a Gamma GLM with a Log link. This will also mean that we won't need to retransform the predicted payout amounts if we use a Gamma GLM model as our selected model.

```
Attaching package: 'boot'


The following object is masked from 'package:car':

    logit


The following object is masked from 'package:lattice':

    melanoma



Attaching package: 'faraway'


The following objects are masked from 'package:boot':

    logit, melanoma


The following object is masked from 'package:gnm':

    wheat


The following objects are masked from 'package:car':

    logit, vif


The following object is masked from 'package:mice':

    mammalsleep
```

```
The following object is masked from 'package:lattice':

    melanoma



Call:
glm(formula = TARGET_AMT ~ BLUEBOOK_LOG, family = Gamma(link = "log"),
    data = claims_only)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.10091    0.38378  15.897  < 2e-16 ***
BLUEBOOK_LOG  0.27009    0.04083   6.615 4.67e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Gamma family taken to be 1.569876)

    Null deviance: 1604.5  on 2152  degrees of freedom
Residual deviance: 1538.9  on 2151  degrees of freedom
AIC: 41282

Number of Fisher Scoring iterations: 5
```


Residuals vs Fitted
Pearson Residuals
Predicted values
2038
1858
1430
Scale−Location
√|Std. Pearson resid.|
Predicted values
Cook's distance
Cook's distance
Obs. number
1430
2038
2063
Residuals vs Leverage
Std. Pearson resid.
Leverage
Cook's distance
0.5
2038
1430
2063
residuals(amt_model5, type = "deviance")
Index
Sorted Data
Half−normal quantiles
1430

```
           df       AIC
amt_model5  3 41282.035
amt_model2  3  5181.999
```

Our Residual vs fitted plot shows some funneling suggesting that our data is heteroschedastic; however, this may be acceptable for our payout predictions when using GLM's. Our Scale-Location plot shows a roughly horizontal plot. Additionally, three observations are moderately influential (2038, 2063, and 1430). Our Residuals vs Leverage highlights the same points but they don't appear to exert much leverage on our model. Our deviance residuals show a random scatter around the 0, which suggests the gamma model is a good fit for our data. Our half-normal plot of the absolute deviance residuals shows points generally following the slope line, though we do see flaring on the right tail and note points 2038 and 1430 deviating from the line.

```
# A tibble: 1 x 36
  TARGET_AMT KIDSDRIV   AGE HOMEKIDS   YOJ INCOME PARENT1 HOME_VAL MSTATUS SEX
       <dbl>    <dbl> <dbl>    <dbl> <dbl>  <dbl> <fct>      <dbl> <fct>   <fct>
1    107586.        0    51        0    11  72354 No        213346 Yes     M
# i 26 more variables: EDUCATION <fct>, JOB <fct>, TRAVTIME <dbl>,
#   CAR_USE <fct>, BLUEBOOK <dbl>, TIF <dbl>, CAR_TYPE <fct>, RED_CAR <fct>,
#   OLDCLAIM <dbl>, CLM_FREQ <dbl>, REVOKED <fct>, MVR_PTS <dbl>,
#   CAR_AGE <dbl>, URBANICITY <fct>, CLAIMS_PER_YEAR <dbl>, AVG_CLAIM <dbl>,
#   BLUEBOOK_LOG <dbl>, OLDCLAIM_LOG <dbl>, INCOME_LOG <dbl>,
#   HOME_VAL_LOG <dbl>, AGE_GROUP <chr>, INCOME_GROUP <chr>,
#   CLAIMS_INCOME_RATIO <dbl>, VEHICLE_VALUE_TO_INCOME <dbl>, ...
```

Looking closer at point 2038 shows that the policy holder had unusually high payout amount of 107586. However, it is not possible to tell if this is due to a data entry error or is representative of a true payout amount for an accident with greater damage.

MQ: Fit Linear Model

```r
lm_model <- lm(
  TARGET_AMT ~ KIDSDRIV + YOJ + INCOME_LOG + PARENT1 +
    MSTATUS + EDUCATION + JOB + TRAVTIME + CAR_USE + TIF +
    CAR_TYPE + OLDCLAIM_LOG + REVOKED + MVR_PTS + URBANICITY + BLUEBOOK_LOG,
  data = claims_only
)
summary(lm_model)
```

```
Call:
lm(formula = TARGET_AMT ~ KIDSDRIV + YOJ + INCOME_LOG + PARENT1 +
    MSTATUS + EDUCATION + JOB + TRAVTIME + CAR_USE + TIF + CAR_TYPE +
    OLDCLAIM_LOG + REVOKED + MVR_PTS + URBANICITY + BLUEBOOK_LOG,
    data = claims_only)

Residuals:
   Min     1Q Median     3Q    Max
 -8047  -3180  -1534    418  99702

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          -5227.025   3062.709  -1.707 0.088030 .
KIDSDRIV               -41.311    281.816  -0.147 0.883470
YOJ                     27.358     59.794   0.458 0.647334
INCOME_LOG               9.931     95.538   0.104 0.917222
PARENT1Yes             392.020    499.892   0.784 0.433004
MSTATUSYes            -369.428    415.710  -0.889 0.374282
EDUCATIONBachelors    -353.569    594.089  -0.595 0.551810
EDUCATIONHigh School  -555.973    514.147  -1.081 0.279664
EDUCATIONMasters         6.239    980.633   0.006 0.994924
EDUCATIONPhD          1053.966   1162.803   0.906 0.364826
JOBClerical           -102.613    578.612  -0.177 0.859256
JOBDoctor            -2591.075   1866.089  -1.389 0.165129
JOBHome Maker         -242.251    933.278  -0.260 0.795220
JOBLawyer             -228.408   1168.487  -0.195 0.845041
JOBManager           -1335.949    902.696  -1.480 0.139034
JOBProfessional        500.802    681.528   0.735 0.462529
JOBStudent             -57.273    764.696  -0.075 0.940304
JOBUnknown            -521.538   1138.629  -0.458 0.646970
TRAVTIME                -0.189     11.076  -0.017 0.986384
CAR_USEPrivate        -370.288    522.525  -0.709 0.478618
TIF                    -15.178     42.523  -0.357 0.721174
CAR_TYPEPanel Truck    450.332    849.229   0.530 0.595971
CAR_TYPEPickup        -107.191    596.116  -0.180 0.857315
CAR_TYPESports Car     273.645    619.431   0.442 0.658702
CAR_TYPESUV            -32.331    518.232  -0.062 0.950260
CAR_TYPEVan            525.350    741.739   0.708 0.478857
OLDCLAIM_LOG           -11.264     41.065  -0.274 0.783881
REVOKEDYes            -766.419    418.191  -1.833 0.066988 .
MVR_PTS                123.866     69.900   1.772 0.076530 .
URBANICITYUrban         72.277    755.168   0.096 0.923760
```

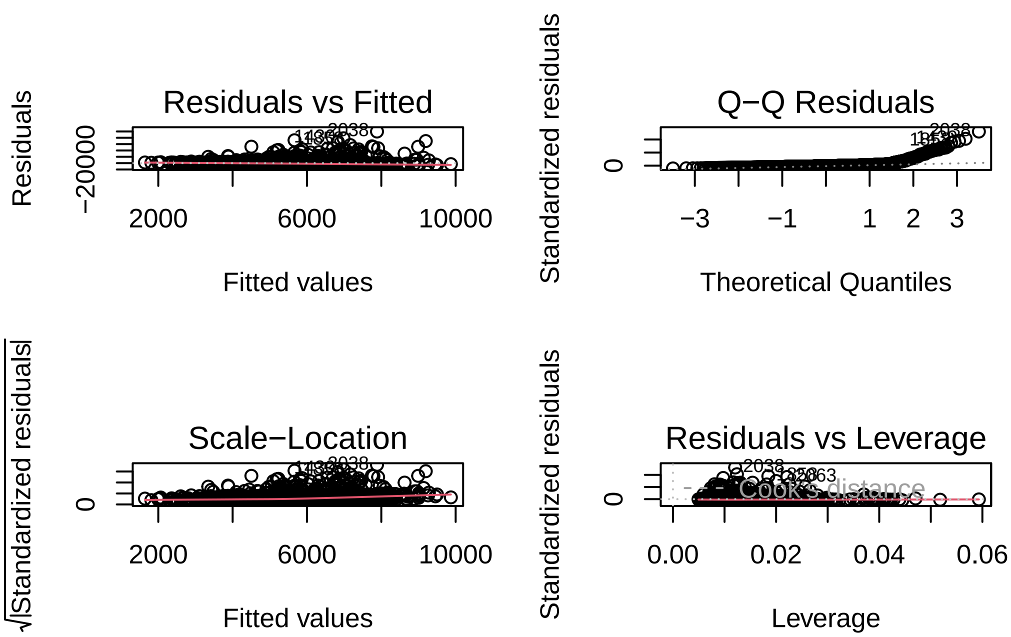
```
BLUEBOOK_LOG          1187.094    307.555   3.860 0.000117 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7700 on 2122 degrees of freedom
Multiple R-squared:  0.0249,    Adjusted R-squared:  0.01111
F-statistic: 1.806 on 30 and 2122 DF,  p-value: 0.00477
```

Assumption Testing – Linear

Linearity & Residual Plots

Explanation: p > 0.05 = residuals are normally distributed (good!)

```r
par(mfrow = c(2,2))
plot(lm_model)
```



Explanation:

Residuals vs. Fitted: should show no pattern (linearity)

Q-Q Plot: points should fall along diagonal (normality)

Normality of Residuals

```r
shapiro.test(residuals(lm_model))
```

```
	Shapiro-Wilk normality test

data:  residuals(lm_model)
W = 0.49786, p-value < 2.2e-16
```

Multicollinearity

```r
vif(lm_model)
```

```
           KIDSDRIV                 YOJ          INCOME_LOG 
           1.133044            2.497935            4.119877 
         PARENT1Yes          MSTATUSYes  EDUCATIONBachelors 
           1.562654            1.568095            2.357021 
EDUCATIONHigh School    EDUCATIONMasters        EDUCATIONPhD 
           2.233352            4.498128            2.685101 
         JOBClerical           JOBDoctor       JOBHome Maker 
           1.733914            1.680303            2.423218 
           JOBLawyer          JOBManager     JOBProfessional 
           3.272954            1.763049            1.713003 
          JOBStudent          JOBUnknown            TRAVTIME 
           2.299334            2.785990            1.027964 
      CAR_USEPrivate                 TIF CAR_TYPEPanel Truck 
           2.476775            1.015163            1.986130 
      CAR_TYPEPickup   CAR_TYPESports Car         CAR_TYPESUV 
           2.108785            1.689533            2.103972 
         CAR_TYPEVan         OLDCLAIM_LOG          REVOKEDYes 
           1.691018            1.179082            1.037816 
             MVR_PTS      URBANICITYUrban        BLUEBOOK_LOG 
           1.179704            1.047032            1.502388 
```

Payout Model comparison

```r
library(vcdExtra)
library(pscl)

stats <- LRstats(amt_model1, amt_model2, amt_model3, amt_model4, amt_model5, lm_model)

stats$McFaddenR2 <- NA
stats$CV_est_predict_err <- NA
stats$CV_adj_est <- NA

enhanceEvaluationMetrics <- function(df, model_name) {
  model <- get(model_name)
  model_data <- claims_only
  print(model_name)
  head(model)

  df[model_name, "McFaddenR2"] <- pR2(model)["McFadden"]
  # Cross-Validation using 10 folsds
  cv_result <- boot::cv.glm(model_data, model, K= 10)
  df[model_name, "CV_est_predict_err"] <- cv_result$delta[1]
  df[model_name, "CV_adj_est"] <- cv_result$delta[2]

  return(df)
}


# Loop through the list of models and update the dataframe for each
for (model_name in rownames(stats)) {

  #stats <- enhanceEvaluationMetrics(stats, model_name)
}

stats
```

```
Likelihood summary table:
             AIC   BIC   LR Chisq   Df Pr(>Chisq) McFaddenR2 CV_est_predict_err
amt_model1 44641 44658 1.2708e+11 2151          0
amt_model2  5182  5199 1.3950e+03 2151          1
amt_model3  5180  5202 1.3920e+03 2150          1
amt_model4  5193  5272 1.3880e+03 2140          1
amt_model5 41282 41299 1.5390e+03 2151          1
lm_model   44677 44859 1.2581e+11 2122          0
           CV_adj_est
```

```
amt_model1
amt_model2
amt_model3
amt_model4
amt_model5
lm_model
```

MODEL SELECTION

Make Predictions on Evaluation Data

```r
# Apply same transformations to test dataset
testing_df <- testing_df %>%
  mutate(
    BLUEBOOK = ifelse(is.na(BLUEBOOK), 0.01, BLUEBOOK),
    OLDCLAIM = ifelse(is.na(OLDCLAIM), 0.01, OLDCLAIM),
    INCOME = ifelse(is.na(INCOME), 0.01, INCOME),
    HOME_VAL = ifelse(is.na(INCOME), 0.01, HOME_VAL),
    LOG_BLUEBOOK = log1p(BLUEBOOK),
    LOG_OLDCLAIM = log1p(OLDCLAIM),
    LOG_INCOME = log1p(INCOME),
    LOG_HOME_VAL = log1p(HOME_VAL)
  )


insurance_evaluation_df <- testing_df  %>%
  rename(
    INCOME_LOG = LOG_INCOME,
    HOME_VAL_LOG = LOG_HOME_VAL,
    OLDCLAIM_LOG = LOG_OLDCLAIM,
    BLUEBOOK_LOG = LOG_BLUEBOOK
  )
```

Predict Crash Probability & Classification

```r
eval_probs <- predict(logit_model, newdata = insurance_evaluation_df, type = "response")
eval_flag_pred <- ifelse(eval_probs > 0.5, 1, 0)
```

Logistic Regression Model Comparison

```
model_list <- list(
  model1 = model1,
  model2 = model2,
  model3 = model3,
  model4 = model4,
  model5 = model5,
  logit_model = logit_model
)
```

```
logit_eval_df <- data.frame(
  Model = character(),
  AIC = numeric(),
  McFaddenR2 = numeric(),
  Accuracy = numeric(),
  Precision = numeric(),
  Recall = numeric(),
  Specificity = numeric(),
  F1 = numeric(),
  AUC = numeric(),
  stringsAsFactors = FALSE
)

for (model_name in names(model_list)) {
  model <- model_list[[model_name]]

if (model_name == "logit_model") {
  df <- insurance_training_clean
} else if (model_name %in% c("model4", "model5")) {
  df <- df_training_one_hot
} else {
  df <- training_df
}

  probs <- predict(model, newdata = df, type = "response")
  preds <- ifelse(probs > 0.5, 1, 0)

  cm <- confusionMatrix(as.factor(preds), as.factor(df$TARGET_FLAG))

  roc_obj <- roc(df$TARGET_FLAG, probs)
  auc_val <- as.numeric(roc_obj$auc)
```

```r
  logit_eval_df <- rbind(logit_eval_df, data.frame(
    Model = model_name,
    AIC = AIC(model),
    McFaddenR2 = pR2(model)["McFadden"],
    Accuracy = cm$overall["Accuracy"],
    Precision = cm$byClass["Precision"],
    Recall = cm$byClass["Sensitivity"],
    Specificity = cm$byClass["Specificity"],
    F1 = cm$byClass["F1"],
    AUC = auc_val
  ))
}
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases

fitting null model for pseudo-r2

Setting levels: control = 0, case = 1
Setting direction: controls < cases

fitting null model for pseudo-r2

Setting levels: control = 0, case = 1
Setting direction: controls < cases

fitting null model for pseudo-r2

Setting levels: control = 0, case = 1
Setting direction: controls < cases

fitting null model for pseudo-r2

Setting levels: control = 0, case = 1
Setting direction: controls < cases

fitting null model for pseudo-r2
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases

fitting null model for pseudo-r2
```

```r
knitr::kable(logit_eval_df, caption = "Logistic Regression Model Comparison")
```

Table 1: Logistic Regression Model Comparison

	Model	AIC	McFaddenR2	Accuracy	Precision	Recall	Specificity	F1	AUC
McFadden	model1	7364.974	0.2260561	0.7923049	0.8175526	0.9241012	0.4245239	0.8675678	0.8143162
McFadden1	model2	7357.708	0.2257658	0.7927950	0.8170069	0.9259321	0.4212726	0.8680659	0.8141510
McFadden2	model3	7327.134	0.2298616	0.7927950	0.8176600	0.9247670	0.4245239	0.8679216	0.8164172
McFadden3	model4	7354.647	0.2243920	0.7924274	0.8164613	0.9262650	0.4189503	0.8679039	0.8130542
McFadden4	model5	7788.643	0.1759743	0.7738022	0.7958487	0.9317577	0.3330237	0.8584573	0.7784001
McFadden5	logit_model	7412.473	0.2193138	0.7856880	0.8115582	0.9232690	0.4017650	0.8638169	0.8103768

After evaluating six logistic regression models, we selected model3 as the best binary classification model for predicting TARGET_FLAG. This model achieved the highest AUC (0.816) and highest accuracy (79.3%), while also maintaining a strong balance between precision (0.818) and recall (0.925). It demonstrated the highest McFadden R^2 (0.2299) and the lowest AIC (7327.13) among all models evaluated. Although model4 had a slightly lower AIC, its predictive metrics were marginally weaker, so model3 was prioritized for its superior performance and interpretability. For predicting TARGET_AMT, we compared six regression models. These included both standard linear models and a gamma GLM. We found that amt_model3, which includes BLUEBOOK_LOG and MVR_PTS as predictors for a log-transformed payout amount (TARGET_AMT_LOG), was the strongest overall. It achieved an AIC of 5179.6, the lowest among all linear models, and showed improved explanatory power over simpler versions. Both predictors were statistically significant, and diagnostic plots indicated acceptable performance across key assumptions. While the gamma GLM (amt_model5) handled skewness well, it had a much higher AIC (41282), and thus was not chosen.

Overall, model3 for classification and amt_model3 for payout prediction provide the most reliable, interpretable, and statistically sound results. These models are recommended for deployment, with model3 used to first classify crashers, and amt_model3 used to estimate payout amounts for those predicted to crash.

ROC Curve Comparison

```r
# Compare ROC curves visually
plot(roc(training_df$TARGET_FLAG, predict(model1, type="response")), col = "blue", main = "R
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases
```

```r
lines(roc(training_df$TARGET_FLAG, predict(model2, type="response")), col = "green")
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

```r
lines(roc(training_df$TARGET_FLAG, predict(model3, type="response")), col = "orange")
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

```r
lines(roc(df_training_one_hot$TARGET_FLAG, predict(model4, type="response")), col = "purple")
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

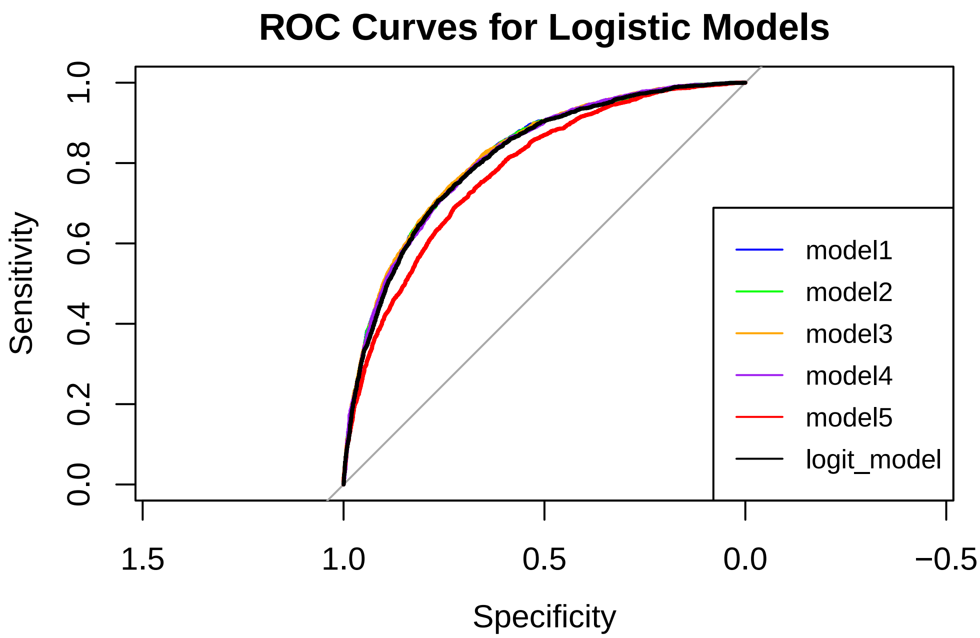
```r
lines(roc(training_df$TARGET_FLAG, predict(model5, type="response")), col = "red")
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

```r
lines(roc(insurance_training_clean$TARGET_FLAG, predict(logit_model, type="response")), col =
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

```r
legend("bottomright", legend = c("model1", "model2", "model3", "model4", "model5", "logit_mo
       col = c("blue", "green", "orange", "purple", "red", "black"), lty = 1, cex = 0.8)
```



Predict Cost for Crashers Only

"'{ #eval_amt_pred <- rep(0, nrow(eval)) #eval_crash_idx <- which(eval_flag_pred == 1) #eval_amt_pred[eval_crash_idx] <- predict(lm_model, newdata = #insurance_evaluation_df[eval_crash_idx,])

```
### Final Combined Output



::: {.cell}

```{.r .cell-code}
# Transform data by putting it into buckets using case_when
testing_df_clean <- testing_df %>%
  mutate(
    AGE_GROUP = case_when(
      AGE <= 25 ~ "Young",
      AGE > 25 & AGE <= 40 ~ "Adult",
      AGE > 40 & AGE <= 60 ~ "Middle Aged",
      AGE > 60 ~ "Senior",
```
```

```
      TRUE ~ NA_character_  # Handle missing values
    )
  )
```

:::

```
# Fix Income Grouping - Ensure proper quantile calculation
income_quantiles <- quantile(testing_df$INCOME, probs = seq(0, 1, 0.25), na.rm = TRUE)

testing_df_clean <- testing_df_clean %>%
  mutate(
    INCOME_GROUP = case_when(
      INCOME <= income_quantiles[2] ~ "Low",
      INCOME > income_quantiles[2] & INCOME <= income_quantiles[3] ~ "Medium",
      INCOME > income_quantiles[3] & INCOME <= income_quantiles[4] ~ "High",
      INCOME > income_quantiles[4] ~ "Very High",
      TRUE ~ NA_character_
    )
  )
```

```
testing_df_clean <- testing_df_clean %>%
  mutate(
    BLUEBOOK = ifelse(is.na(BLUEBOOK), 0.01, BLUEBOOK),
    OLDCLAIM = ifelse(is.na(OLDCLAIM), 0.01, OLDCLAIM),
    INCOME = ifelse(is.na(INCOME), 0.01, INCOME),
    HOME_VAL = ifelse(is.na(INCOME), 0.01, HOME_VAL),
    LOG_BLUEBOOK = log1p(BLUEBOOK),
    LOG_OLDCLAIM = log1p(OLDCLAIM),
    LOG_INCOME = log1p(INCOME),
    LOG_HOME_VAL = log1p(HOME_VAL)
  )
```

```
#final_predictions <- eval %>%
#   select(INDEX) %>%
#   mutate(
#     PRED_TARGET_FLAG = eval_flag_pred,
#     PRED_TARGET_AMT = eval_amt_pred
#   )

#head(final_predictions)

prediction <- predict(model3, testing_df_clean)
```

```r
testing_df_clean$TARGET_FLAG <- prediction

testing_df_clean <- testing_df_clean  %>%
 rename(
    INCOME_LOG = LOG_INCOME,
    HOME_VAL_LOG = LOG_HOME_VAL,
    OLDCLAIM_LOG = LOG_OLDCLAIM,
    BLUEBOOK_LOG = LOG_BLUEBOOK
  )

predictions_amt <- predict(amt_model3, testing_df_clean)
testing_df_clean$TARGET_AMT <- predictions_amt


head(testing_df_clean)
```

```
# A tibble: 6 x 32
  INDEX TARGET_FLAG TARGET_AMT KIDSDRIV   AGE HOMEKIDS   YOJ   INCOME PARENT1
  <dbl>       <dbl>      <dbl>    <dbl> <dbl>    <dbl> <dbl>    <dbl> <fct>
1     3       -1.81       8.37        0    48        0    11 52881    No
2     9       -1.09       8.35        1    40        1    11 50815    Yes
3    10       -1.98       8.13        0    44        2    12 43486    Yes
4    18       -1.15       8.20        0    35        2    NA 21204    Yes
5    21       -1.51       8.34        0    59        0    12 87460    No
6    30       -1.17       8.39        0    46        0    14     0.01 No
# i 23 more variables: HOME_VAL <dbl>, MSTATUS <fct>, SEX <fct>,
#   EDUCATION <fct>, JOB <fct>, TRAVTIME <dbl>, CAR_USE <fct>, BLUEBOOK <dbl>,
#   TIF <dbl>, CAR_TYPE <fct>, RED_CAR <fct>, OLDCLAIM <dbl>, CLM_FREQ <dbl>,
#   REVOKED <fct>, MVR_PTS <dbl>, CAR_AGE <dbl>, URBANICITY <fct>,
#   BLUEBOOK_LOG <dbl>, OLDCLAIM_LOG <dbl>, INCOME_LOG <dbl>,
#   HOME_VAL_LOG <dbl>, AGE_GROUP <chr>, INCOME_GROUP <chr>
```

Explanation:

Everyone gets a binary prediction (crash or not).

Only predicted crashers get a dollar amount predicted.

Others remain at zero.