Homework 2

Sam Kuhn

11/28/22

Table of contents

Objective

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

Your objective is to build multiple linear regression and binary logistic regression models on the training data to predict the probability that a person will crash their car and also the amount of money it will cost if the person does crash their car. You can only use the variables given to you (or variables that you derive from the variables provided).

Data Exploration

Load data

```
# Install pacman package and load libraries
# install.packages("pacman")
pacman::p_load(tidyverse, here, tidymodels, corrplot, MASS, gt, stargazer, vtable, glmnet)

# Makes sure dplyr::filter and dplyr::select will be used
conflicted::conflict_prefer("select", "dplyr")
conflicted::conflict_prefer("filter", "dplyr")

# Load training set from data folder and clean variable names
training_set <- readr::read_csv(
  here::here("data", "insurance_training_data.csv")
) |>
  janitor::clean_names()
```

First, lets rename the variables to a more readable format. Then, print a summary of the data to get a sense of what datatypes there are. From the table below, we can see we have a few variables where the datatype is wrong. In particular, `Income`, `Bluebook Value`, and `Total Claims` are all character columns rather than numeric. We will convert them.

```
Rows: 8,161
Columns: 26
$ index                 <dbl> 1, 2, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, ~
$ `Crash Dummy`         <dbl> 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, ~
$ `Crash Damage`        <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 2946.000, ~
$ `Teen Drivers`        <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, ~
$ `Age of Driver`       <dbl> 60, 43, 35, 51, 50, 34, 54, 37, 34, 50, 53, 4~
$ `Kids at Home`        <dbl> 0, 0, 1, 0, 0, 1, 0, 2, 0, 0, 0, 0, 0, 0, 0, ~
$ `Years on Job`        <dbl> 11, 11, 10, 14, NA, 12, NA, NA, 10, 7, 14, 5,~
$ Income                <chr> "$67,349", "$91,449", "$16,039", NA, "$114,98~
$ `Single Parent`       <chr> "No", "No", "No", "No", "No", "Yes", "No", "N~
$ `Home Value`          <chr> "$0", "$257,252", "$124,191", "$306,251", "$2~
$ `Marital Status`      <chr> "z_No", "z_No", "Yes", "Yes", "Yes", "z_No", ~
$ Sex                   <chr> "M", "M", "z_F", "M", "z_F", "z_F", "z_F", "M~
$ Education             <chr> "PhD", "z_High School", "z_High School", "<Hi~
$ Job                   <chr> "Professional", "z_Blue Collar", "Clerical", ~
```

```
$ `Distance to Work`       <dbl> 14, 22, 5, 32, 36, 46, 33, 44, 34, 48, 15, 36~
$ car_use                  <chr> "Private", "Commercial", "Private", "Private"~
$ `Bluebook Value`         <chr> "$14,230", "$14,940", "$4,010", "$15,440", "$~
$ `Time as Customer`       <dbl> 11, 1, 4, 7, 1, 1, 1, 1, 1, 7, 1, 7, 7, 6, 1,~
$ `Car Type`               <chr> "Minivan", "Minivan", "z_SUV", "Minivan", "z_~
$ `Is Red Car`             <chr> "yes", "yes", "no", "yes", "no", "no", "no", ~
$ `Total Claims`           <chr> "$4,461", "$0", "$38,690", "$0", "$19,217", "~
$ `Claim Frequency`        <dbl> 2, 0, 2, 0, 2, 0, 0, 1, 0, 0, 0, 0, 2, 0, 0, ~
$ `License Revoked`        <chr> "No", "No", "No", "No", "Yes", "No", "No", "Y~
$ `Licences Record Points` <dbl> 3, 0, 3, 0, 3, 0, 0, 10, 0, 1, 0, 0, 3, 3, 3,~
$ `Car Age`                <dbl> 18, 1, 10, 6, 17, 7, 1, 7, 1, 17, 11, 1, 9, 1~
$ Geography                <chr> "Highly Urban/ Urban", "Highly Urban/ Urban",~
```

```r
training_set <- training_set |>
  mutate(
    across(
      .cols = c(`Income`, `Home Value`, `Bluebook Value`, `Total Claims`),
      .fns = parse_number
    )
  )
```

Check for missing values

To check for NA values, we are going to take the sum of every value matching `NA` across the entire dataframe and print the results. Then, replace all the `NA` values with the median value of the corresponding variable. The variables with the most `NA` observations are: `team_batting_hbp`, `team_baserun_cs`, and `team_fielding_dp`. Based on the percentages from `Table 1`, it does not appear that only columns have more than 10% of their observations missing.

```
# Sum NAs across columns, divide by length of column and get percent missing NAs per column
missing_vals <- training_set |>
  summarise(across(everything(), ~ sum(is.na(.) / length(.)))) |>
  pivot_longer(cols = where(is.numeric), names_to = "variable")

missing_vals |>
  filter(value > 0) |>
  gt::gt() |>
  tab_header(
    title = "Table 1: Percent of observations with missing values"
  ) |>
  cols_label(
    variable = "Variable",
    value = "Percent"
  ) |>
  fmt_percent(
    columns = value,
    decimals = 2
  )
```

Table 1: Percent of observations with missing values

Variable	Percent
Age of Driver	0.07%
Years on Job	5.56%
Income	5.45%
Home Value	5.69%
Job	6.45%
Car Age	6.25%

Visualize

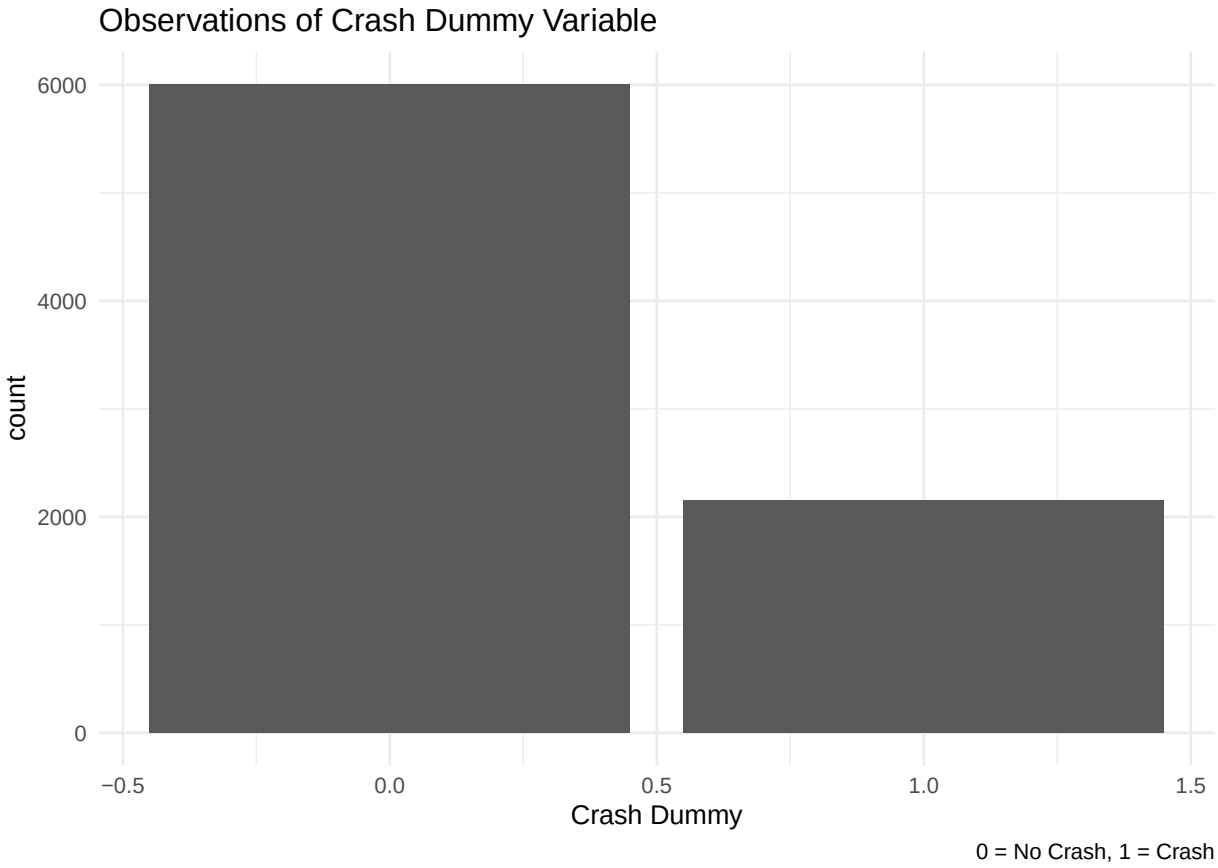
Summary Plots

Now that we have the data in the proper format, let's produce some charts to get a sense of the shape and distribution of the data itself. First, we want to investigate the distribution of the regressand, `Crash Dummy`. From the bar chart, we can so the amount of observations where there wasn't a crash was almost three times more than there being a crash.

```
vars <- training_set |>
  # select(-index) |>
  names() |>
  set_names()

plots <- map(vars, ~ ggplot(data = training_set) +
  geom_point(aes(x = .data[[.x]], y = "Crash Dummy")) +
  theme_minimal() +
  labs(y = .x))

training_set |>
  ggplot(aes(`Crash Dummy`)) +
  geom_bar() +
  labs(
    title = "Observations of Crash Dummy Variable",
    caption = "0 = No Crash, 1 = Crash"
  ) +
  theme_minimal()
```



Summary Statistics

Let's produce a summary table of the `mean`, `standard deviation`, `median` and maximum and minimum values of the dataset, then move towards transformations of the variables. From the data in `Table 2`, we can see that the for the following key varibes: mean `Crash Damage` was 1,504 dollars, mean `Age of Driver` was 44, and mean `Bluebook Value` was $15,709.

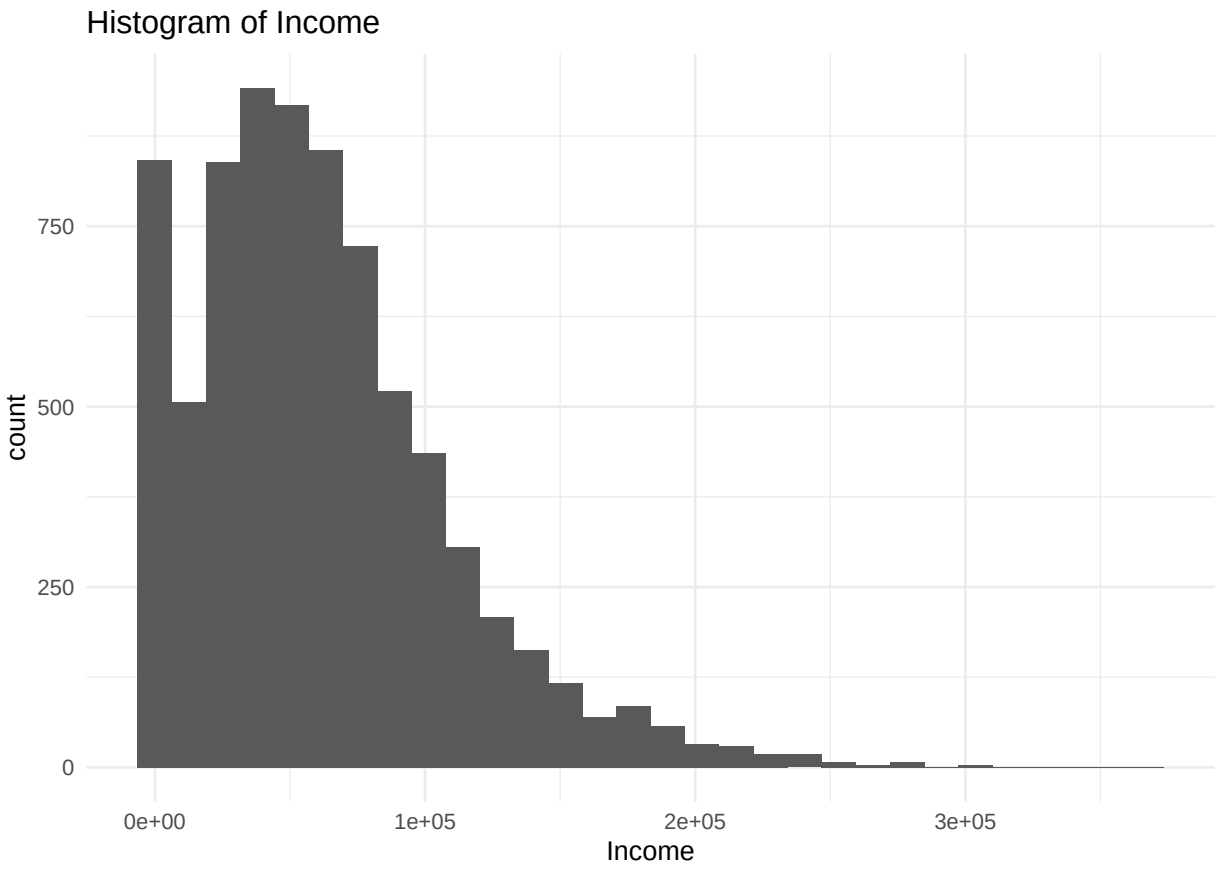
Table 2: Summary Statistics

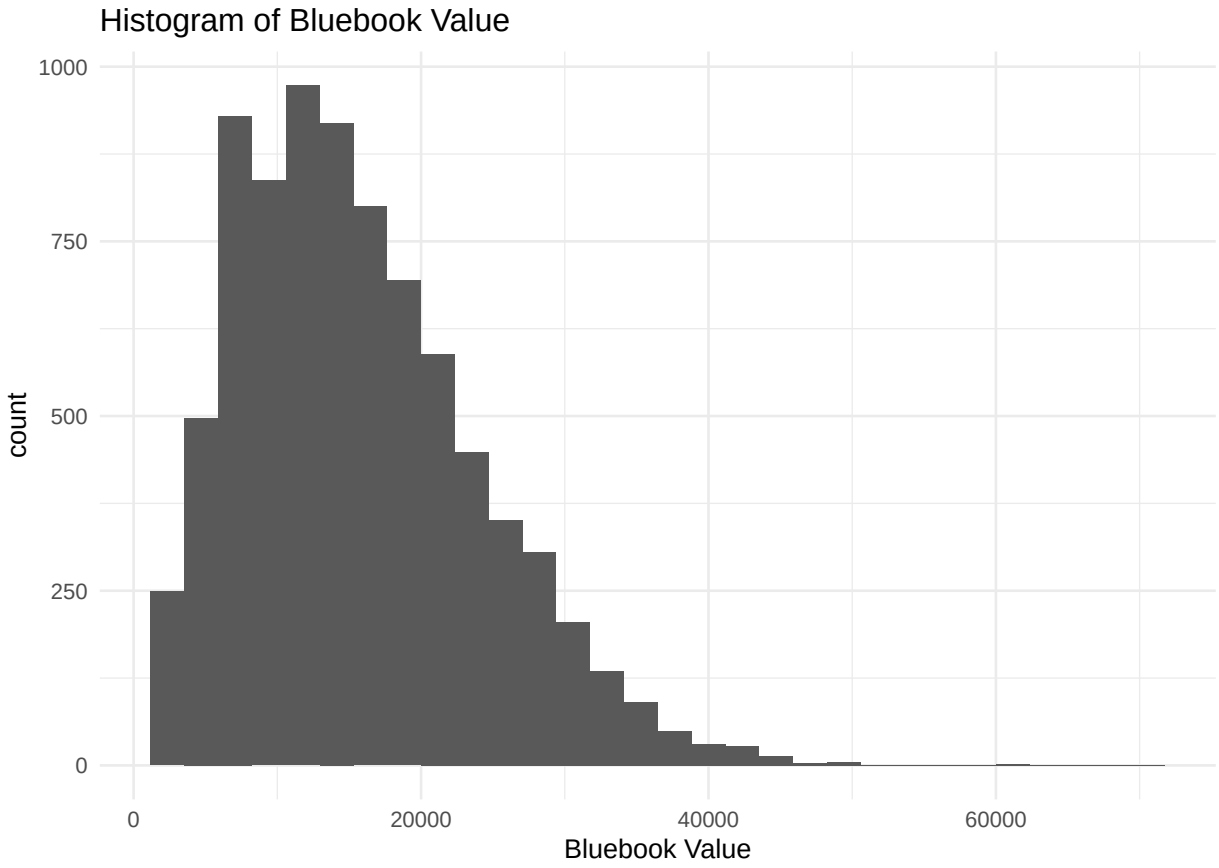
Variable	Mean	Median	Sd	Max	Min
Teen Drivers	0.2	0	0.512	4	0
Age of Driver	44.8	45	8.628	81	16
Kids at Home	0.7	0	1.116	5	0
Years on Job	10.5	11	4.092	23	0
Income	61898.1	54028	47572.683	367030	0
Home Value	154867.3	161160	129123.775	885282	0
Distance to Work	33.5	33	15.908	142	5
Bluebook Value	15709.9	14440	8419.734	69740	1500
Time as Customer	5.4	4	4.147	25	1
Total Claims	4037.1	0	8777.139	57037	0
Claim Frequency	0.8	0	1.158	5	0
Licences Record Points	1.7	1	2.147	13	0
Car Age	8.3	8	5.701	28	-3

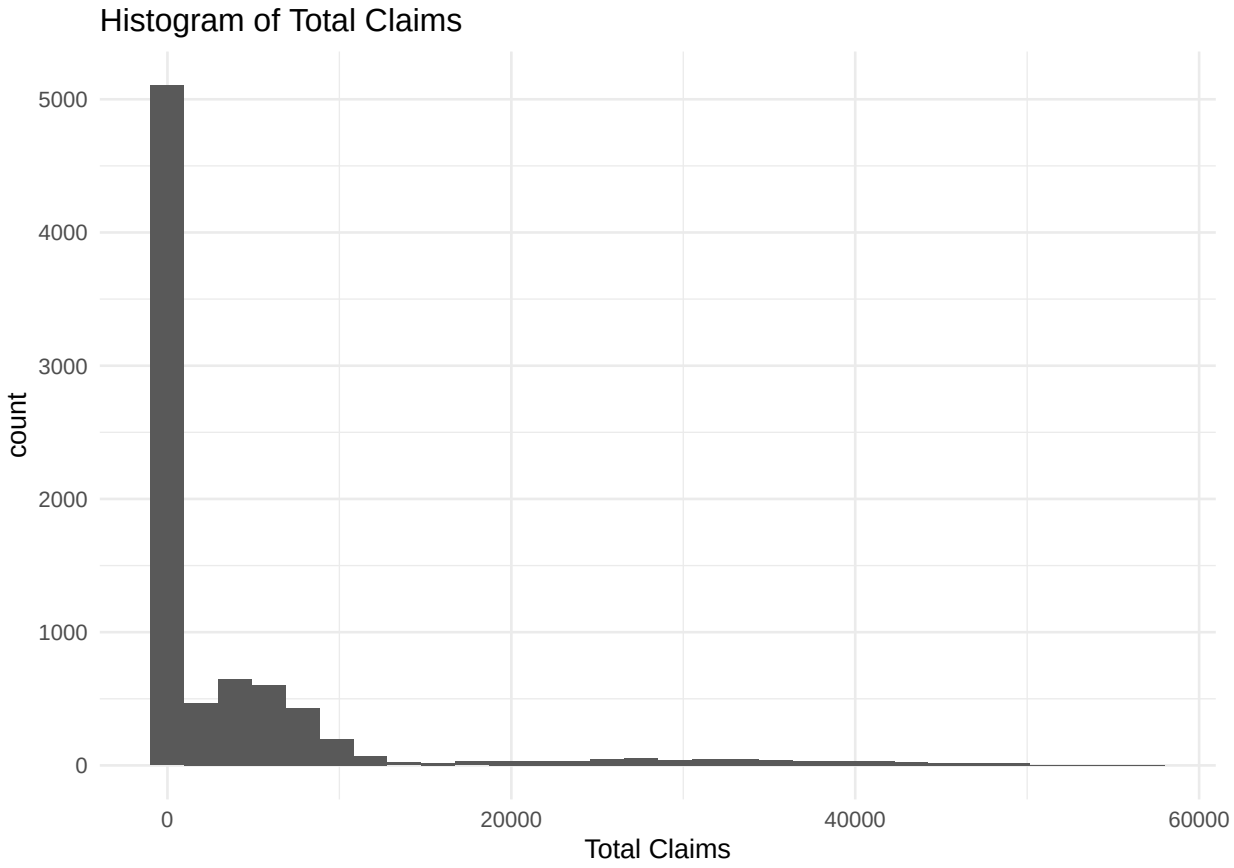
Data preparation

Log Transformation

Let's check the distribution of a few key numeric variables, then take the log transformation if necessary. From the below four plots, the most heavily right-skewed variables are `Crash Damage` and `Total Claims`. These two variables will be log-transformed, which will be incldued in the appendix.




Histogram of Bluebook Value
1000
750
500
250
0
count
0
20000
40000
60000
Bluebook Value


Histogram of Total Claims
count
5000
4000
3000
2000
1000
0
0
20000
40000
60000
Total Claims

Build Models

Model 1

For the first model, we will use a full model with all predictors, then in the following models perform some analysis to either remove variables, or keep it as is. Will we return the model summary tab, then check for multicollinearity by the `vif` function. The two variables with a variance inflation factor above 10 are: `Education` and `Job`. This is not surprising, given the strong positive correlation between education and earnings.

```
Call:
lm(formula = as.numeric(`Crash Dummy`) ~ . - index - `Crash Damage`,
    data = training_set)

Residuals:
    Min      1Q  Median      3Q     Max
-0.9542 -0.2820 -0.1051  0.2871  1.2418

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   4.444e-01  4.655e-02   9.548  < 2e-16 ***
`Teen Drivers`                4.789e-02  1.106e-02   4.332 1.50e-05 ***
`Age of Driver`              -6.564e-04  6.976e-04  -0.941 0.346791
`Kids at Home`                1.465e-03  6.380e-03   0.230 0.818376
`Years on Job`               -1.724e-03  1.456e-03  -1.185 0.236179
Income                       -3.824e-07  2.014e-07  -1.899 0.057625 .
`Single Parent`Yes            8.427e-02  1.976e-02   4.264 2.04e-05 ***
`Home Value`                 -1.671e-07  6.296e-08  -2.654 0.007964 **
`Marital Status`z_No          5.972e-02  1.471e-02   4.059 4.99e-05 ***
Sexz_F                       -3.154e-02  1.796e-02  -1.756 0.079110 .
EducationBachelors           -6.023e-02  1.996e-02  -3.018 0.002559 **
EducationMasters             -5.630e-02  2.991e-02  -1.882 0.059849 .
EducationPhD                  8.443e-03  3.754e-02   0.225 0.822055
Educationz_High School        4.285e-04  1.647e-02   0.026 0.979244
JobDoctor                    -1.481e-01  4.368e-02  -3.391 0.000700 ***
JobHome Maker                -4.290e-02  2.465e-02  -1.740 0.081861 .
JobLawyer                    -5.136e-02  2.958e-02  -1.736 0.082614 .
JobManager                   -1.703e-01  2.294e-02  -7.420 1.33e-13 ***
JobProfessional              -5.125e-02  2.089e-02  -2.453 0.014188 *
JobStudent                   -4.278e-02  2.331e-02  -1.835 0.066535 .
```

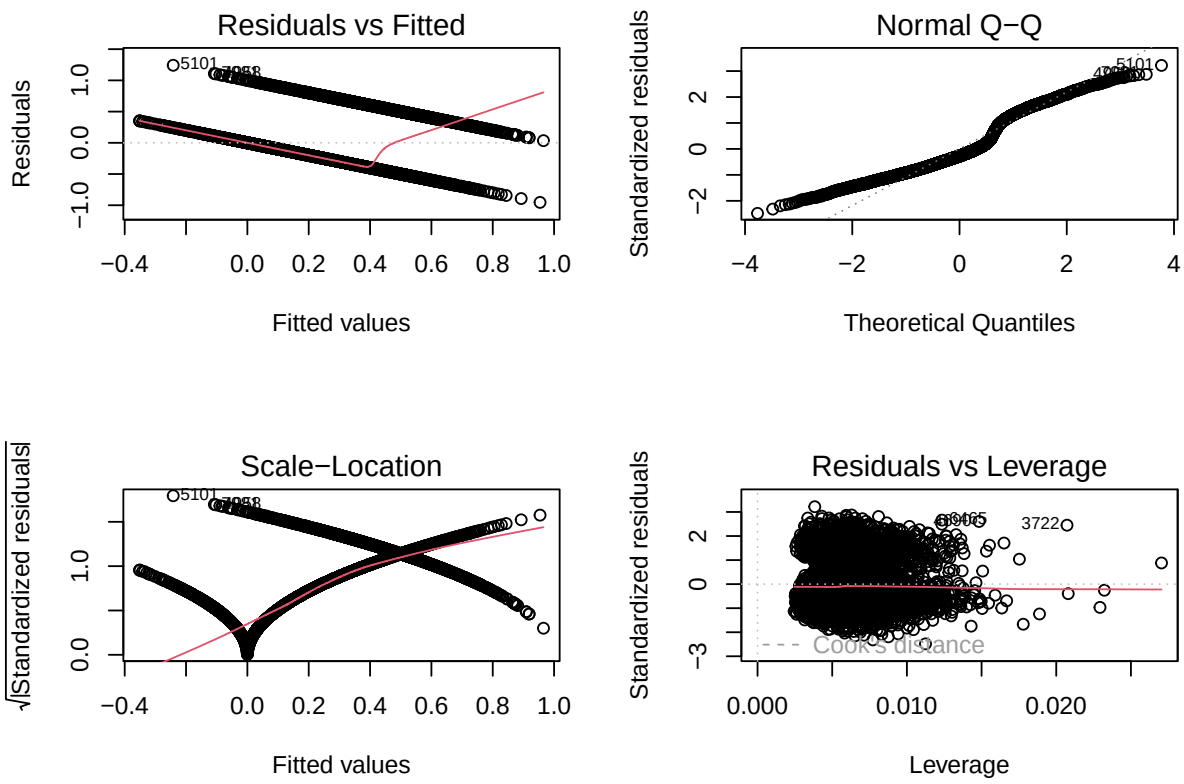
```
Jobz_Blue Collar               -2.964e-02  1.849e-02  -1.603 0.108942
`Distance to Work`              2.175e-03  3.185e-04   6.829 9.42e-12 ***
car_usePrivate                 -1.306e-01  1.613e-02  -8.093 6.97e-16 ***
`Bluebook Value`               -2.818e-06  8.463e-07  -3.329 0.000876 ***
`Time as Customer`             -7.565e-03  1.205e-03  -6.278 3.66e-10 ***
`Car Type`Panel Truck           7.310e-02  2.896e-02   2.524 0.011612 *
`Car Type`Pickup                7.018e-02  1.656e-02   4.239 2.28e-05 ***
`Car Type`Sports Car            1.542e-01  2.085e-02   7.399 1.56e-13 ***
`Car Type`Van                   6.426e-02  2.130e-02   3.017 0.002567 **
`Car Type`z_SUV                 1.109e-01  1.716e-02   6.462 1.11e-10 ***
`Is Red Car`yes                -3.488e-02  1.507e-02  -2.315 0.020661 *
`Total Claims`                 -2.337e-06  7.359e-07  -3.176 0.001500 **
`Claim Frequency`               3.363e-02  5.464e-03   6.154 8.02e-10 ***
`License Revoked`Yes            1.444e-01  1.721e-02   8.388  < 2e-16 ***
`Licences Record Points`        2.181e-02  2.562e-03   8.513  < 2e-16 ***
`Car Age`                      -6.933e-04  1.274e-03  -0.544 0.586278
Geographyz_Highly Rural/ Rural -2.945e-01  1.354e-02 -21.754  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3866 on 6008 degrees of freedom
  (2116 observations deleted due to missingness)
Multiple R-squared:  0.2375,    Adjusted R-squared:  0.2329
F-statistic: 51.97 on 36 and 6008 DF,  p-value: < 2.2e-16


[1] "The mean squared error is: 0.148529957234751"
```

Let's check the residuals vs. fitted plot.

```r
par(mfrow=c(2, 2))
plot(lm_fit_1)
```



Coefficient Interpretation

For this first model, interpretation of the coefficients is based on the linear probability model. The formula definition is as follows: $\Delta P(y = 1|x) = \beta_j \Delta x_j$. In words, the model measures the change in the probability of success when x_j changes, holding all other factors fixed.

- `Distance to work`: Another 1 mile increase in distance to work increases the probability of a crash by .002175. This makes sense, since the longer a commute, the higher change of a crash due to driving time.
- `Age of Driver`: Another 1 year increase in age reduces the probability of a crash by .0006564. This makese sense, since older drivers have more experience.

As another holisitic check, for key variables `License Record Points`, `Distance to Work`, and `Kids at Home` all having positive coefficients makes sense.

Model 2

From the VIF results in `Model 1`, the second model will remove `Education` and `Jobs`. We will double check the VIF results in `Model 2` as well. In the appendix, it is confirmed that no variable in this model has a VIF value above 10.

```
lm_fit_2 <- lm(as.numeric(`Crash Dummy`) ~ . - Education - Job - index - `Crash Damage`, data = training
summary(lm_fit_2)
```

```
Call:
lm(formula = as.numeric(`Crash Dummy`) ~ . - Education - Job -
    index - `Crash Damage`, data = training_set)

Residuals:
    Min      1Q  Median      3Q     Max
-0.9353 -0.2830 -0.1144  0.3005  1.2159

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    4.282e-01  4.079e-02  10.499  < 2e-16 ***
`Teen Drivers`                 4.789e-02  1.112e-02   4.306 1.69e-05 ***
`Age of Driver`               -1.130e-03  6.920e-04  -1.633 0.102490
`Kids at Home`                 4.145e-03  6.357e-03   0.652 0.514356
`Years on Job`                -6.888e-04  1.324e-03  -0.520 0.602883
Income                        -8.210e-07  1.742e-07  -4.713 2.50e-06 ***
`Single Parent`Yes             8.103e-02  1.990e-02   4.071 4.74e-05 ***
`Home Value`                  -1.705e-07  6.218e-08  -2.741 0.006137 **
`Marital Status`z_No           5.916e-02  1.470e-02   4.025 5.78e-05 ***
Sexz_F                        -2.848e-02  1.793e-02  -1.588 0.112378
`Distance to Work`             2.246e-03  3.206e-04   7.005 2.73e-12 ***
car_usePrivate                -1.440e-01  1.234e-02 -11.665  < 2e-16 ***
`Bluebook Value`              -3.027e-06  8.518e-07  -3.554 0.000382 ***
`Time as Customer`            -7.383e-03  1.213e-03  -6.086 1.23e-09 ***
`Car Type`Panel Truck          5.966e-02  2.792e-02   2.137 0.032652 *
`Car Type`Pickup               6.616e-02  1.633e-02   4.051 5.16e-05 ***
`Car Type`Sports Car           1.487e-01  2.101e-02   7.077 1.64e-12 ***
`Car Type`Van                  6.541e-02  2.121e-02   3.083 0.002056 **
`Car Type`z_SUV                1.076e-01  1.728e-02   6.228 5.05e-10 ***
`Is Red Car`yes               -3.837e-02  1.518e-02  -2.527 0.011517 *
`Total Claims`                -2.345e-06  7.414e-07  -3.163 0.001568 **
`Claim Frequency`              3.451e-02  5.506e-03   6.269 3.89e-10 ***
`License Revoked`Yes           1.480e-01  1.732e-02   8.544  < 2e-16 ***
`Licences Record Points`       2.293e-02  2.575e-03   8.903  < 2e-16 ***
`Car Age`                     -4.126e-03  1.006e-03  -4.100 4.19e-05 ***
Geographyz_Highly Rural/ Rural -2.691e-01  1.331e-02 -20.209  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3898 on 6019 degrees of freedom
  (2116 observations deleted due to missingness)
```

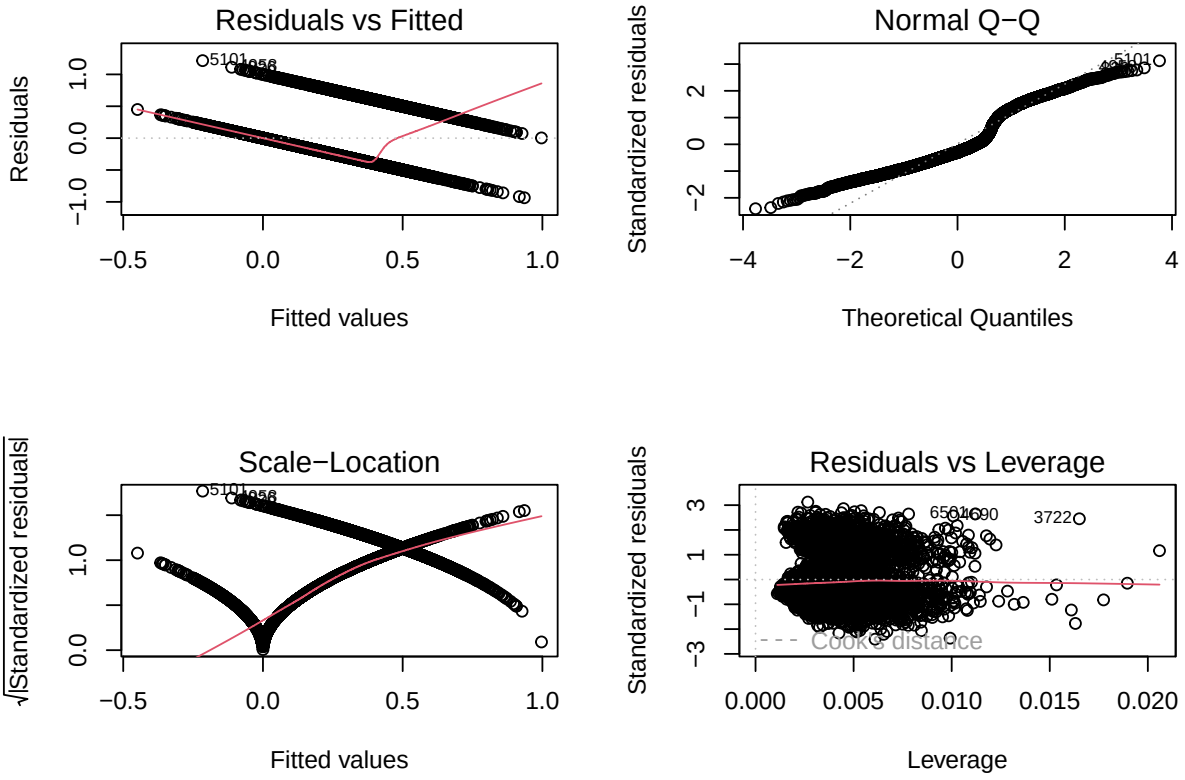
```
Multiple R-squared:  0.2235,    Adjusted R-squared:  0.2202
F-statistic: 69.28 on 25 and 6019 DF,  p-value: < 2.2e-16
```

```r
print(paste0("The mean squared error is: ", mean(lm_fit_2$residuals^2)))
```

```
[1] "The mean squared error is: 0.15125674117899"
```

Let's check the residuals vs. fitted plot.

```r
par(mfrow=c(2, 2))
plot(lm_fit_2)
```



Coefficient Interpretation

For the second model, interpretation of the coefficients is based on the linear probability model. The formula definition is as follows: $\Delta P(y = 1|x) = \beta_j \Delta x_j$. In words, the model measures the change in the probability of success when x_j changes, holding all other factors fixed.

- `Teenage Drivers`: Another 1 additional teenage driver increases the probability of a crash by .004789. This would seem intuitive, since teen drivers are younger and have less experience operating a vehicle.
- `Age of Driver`: Another 1 year increase in age reduces the probability of a crash by .0006564. This also seems intuitive, since having more experience driving would reduce mistakes due to knowledge of the road.

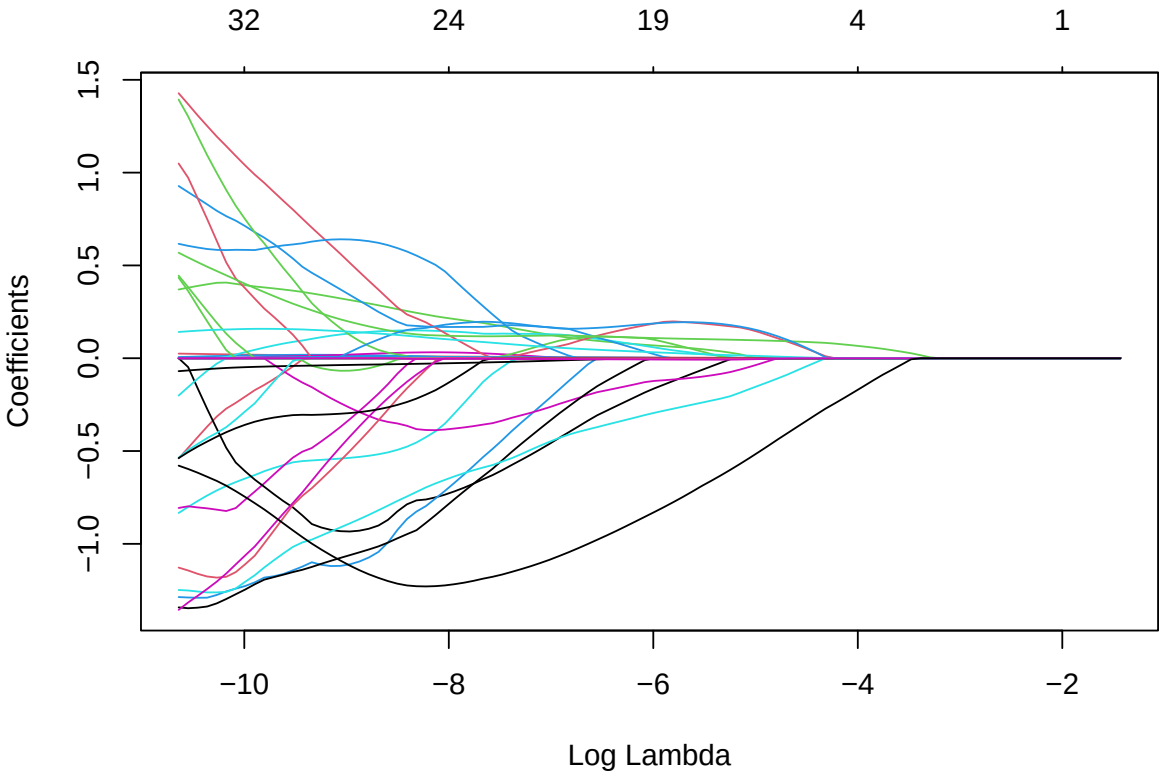
Model 3

For the last model, we will relax an assumption about the relationship between the regressors and regressand. The last model will be a probit model with the same predictors in `Model 2`, which has two advantages: The conditional probability function does not have to be linear, and it has the benefits of lower multicollinearity due to removing `Education` and `Jobs`. To accomplish this, we will use lasso regression from the `glmnet` package.

```
glm_df <- training_set |>
  mutate(
    across(
      .cols = c(`Is Red Car`, `Crash Dummy`, `Marital Status`, "Sex", `License Revoked`,
                `Single Parent`, `Car Type`, "Geography"),
      .fns = as.factor
    )
  ) |>
  # select(-index, -"Education", -"Job") |>
  na.omit()

x <- model.matrix(`Crash Dummy` ~., glm_df)[, -1]
y <- glm_df$`Crash Dummy`

glm_fit_3 <- glmnet(x, as.factor(y), family = "binomial")
par(mfrow=c(1, 1))
plot(glm_fit_3, xvar = "lambda")
```



Model Presentation

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Sun, Dec 04, 2022 - 18:25:09

Table 3: Model 1 and 2 Regression Output

	Dependent variable:	
	as.numeric(‘Crash Dummy‘)	
	Probability of Crash	
	(1)	(2)
Teen Drivers	0.048^{***}	0.048^{***}
	(0.011)	(0.011)
Age of Driver	-0.001	-0.001
	(0.001)	(0.001)
Kids at Home	0.001	0.004
	(0.006)	(0.006)
Years on Job	-0.002	-0.001
	(0.001)	(0.001)
Income	-0.00000^{*}	-0.00000^{***}
	(0.00000)	(0.00000)
Is Single Parent	0.084^{***}	0.081^{***}
	(0.020)	(0.020)
Home Value	-0.00000^{***}	-0.00000^{***}
	(0.00000)	(0.00000)
Not Married	0.060^{***}	0.059^{***}
	(0.015)	(0.015)
Female	-0.032^{*}	-0.028
	(0.018)	(0.018)
Bachelors	-0.060^{***}	
	(0.020)	
Masters	-0.056^{*}	
	(0.030)	
PhD	0.008	
	(0.038)	
High School	0.0004	
	(0.016)	
Doctor	-0.148^{***}	
	(0.044)	
Home Maker	-0.043^{*}	
	(0.025)	
Lawyer	-0.051^{*}	
	(0.030)	
Manager	-0.170^{***}	
	(0.023)	

Select Models

Appendix