Homework-4

Waheeb Algabri, Joe Garcia, Lwin Shwe, Mikhail Broomes

Homework 4 - Binary Logistic Regression & Multiple Linear Regression

Introduction:

This research focuses on analyzing an auto insurance company dataset comprising `8,161` records, each representing a customer. The dataset encompasses two response variables: `TARGET_FLAG` and `TARGET_AMT`. `TARGET_FLAG` indicates whether a customer was involved in a car crash `(1)` or not `(0)`, while `TARGET_AMT` represents the cost incurred if the customer was involved in a crash (zero if not). Our aim is to construct robust models using multiple linear regression and binary logistic regression techniques to predict both the likelihood of a car crash and the associated financial impact. The dataset includes the following variables:

Multiple Linear Regression and Binary Logistic Regression

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

Our objective is twofold: to develop predictive models using multiple linear regression and binary logistic regression on the training data. These models aim to forecast the probability of a customer being involved in a car crash and the monetary consequences of such incidents. We are constrained to utilize only the provided variables or those derived from them. Below is a concise description of the dataset's variables of interest

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
`INDEX`	Identification Variable	None
`TARGET_FLAG`	Was Car in a crash? 1=YES 0=NO	None
`TARGET_AMT`	If car was in a crash, what was the cost	None
`AGE`	Age of Driver	Very young and very old people tend to be risky
`BLUEBOOK`	Value of Vehicle	Unknown effect on probability of collision, but probably effect the payout if there is a crash
`CAR_AGE`	Vehicle Age	Unknown effect on probability of collision, but probably effect the payout if there is a crash
`CAR_TYPE`	Type of Car	Unknown effect on probability of collision, but probably effect the payout if there is a crash

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
`CAR_USE`	Vehicle Use	Commercial vehicles are driven more, so might increase probability of collision
`CLM_FREQ`	# Claims (Past 5 Years)	The more claims you filed in the past, the more you are likely to file in the future
`EDUCATION`	Max Education Level	Unknown but possible more educated people tend to drive safer
`HOMEKIDS`	# Children at Home	Unknown
`HOME_VAL`	Home Value	Homeowners tend to drive safer
`INCOME`	Income	Rich people tend to be in fewer crashes
`JOB`	Job Category	White collar jobs tend to be safer
`KIDSDRIV`	# Driving Children	When teenagers drive your car, you are more likely to get into crashes
`MSTATUS`	Marital Status	Married people driver safer
`MVR_PTS`	Motor Vehicle Record Points	If you get a lot of traffic tickets, you tend to get into more accidents
`OLDCLAIM`	Total Claims (Past 5 Years)	If your total payout over the past five years was high, this suggests future payouts will be high
`PARENT1`	Single Parent	Unknown
`RED_CAR`	A Red Car	Urban legend says that red cars (especially red sports cars) are more risky
`REVOKED`	License Revoked (Past 7 Years)	If your license was revoked in the past 7 years, you probably are a more risky driver
`SEX`	Gender	Urban legend says that women have less crashes then men
`TIF`	Time in Force	People who have been customers for a long time are usually more safe
`TRAVTIME`	Distance to Work	Long drives to work usually suggest greater risk
`URBANICITY`	Home/Work Area	Unknown
`YOJ`	Years on Job	People who stay at a job for a long time are usually more safe

Data Exploration:

We check the classes of our variables to determine whether any of them need to be coerced to numeric or other classes prior to exploratory data analysis.

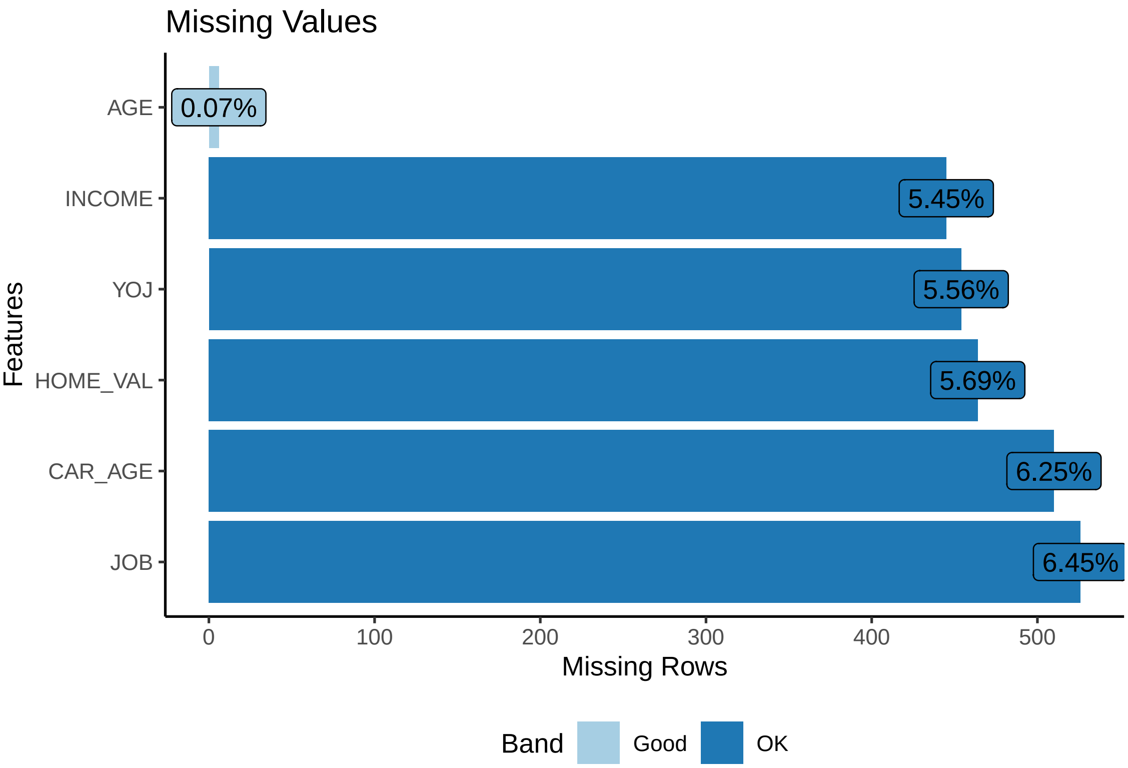
Class	Count	Variables
character	14	BLUEBOOK, CAR_TYPE, CAR_USE, EDUCATION, HOME_VAL, INCOME, JOB, MSTATUS, OLDCLAIM, PARENT1, RED_CAR, REVOKED, SEX, URBANICITY
integer	11	AGE, CAR_AGE, CLM_FREQ, HOMEKIDS, INDEX, KIDSDRIV, MVR_PTS, TARGET_FLAG, TIF, TRAVTIME, YOJ
numeric	1	TARGET_AMT

INCOME, HOME_VAL, BLUEBOOK, and OLDCLAIM are all character variables that will need to be coerced to integers after we strip the "$" from their strings. TARGET_FLAG and the remaining character variables will all need to be coerced to factors.

We remove the identification variable INDEX and take a look at a summary of the dataset's completeness.

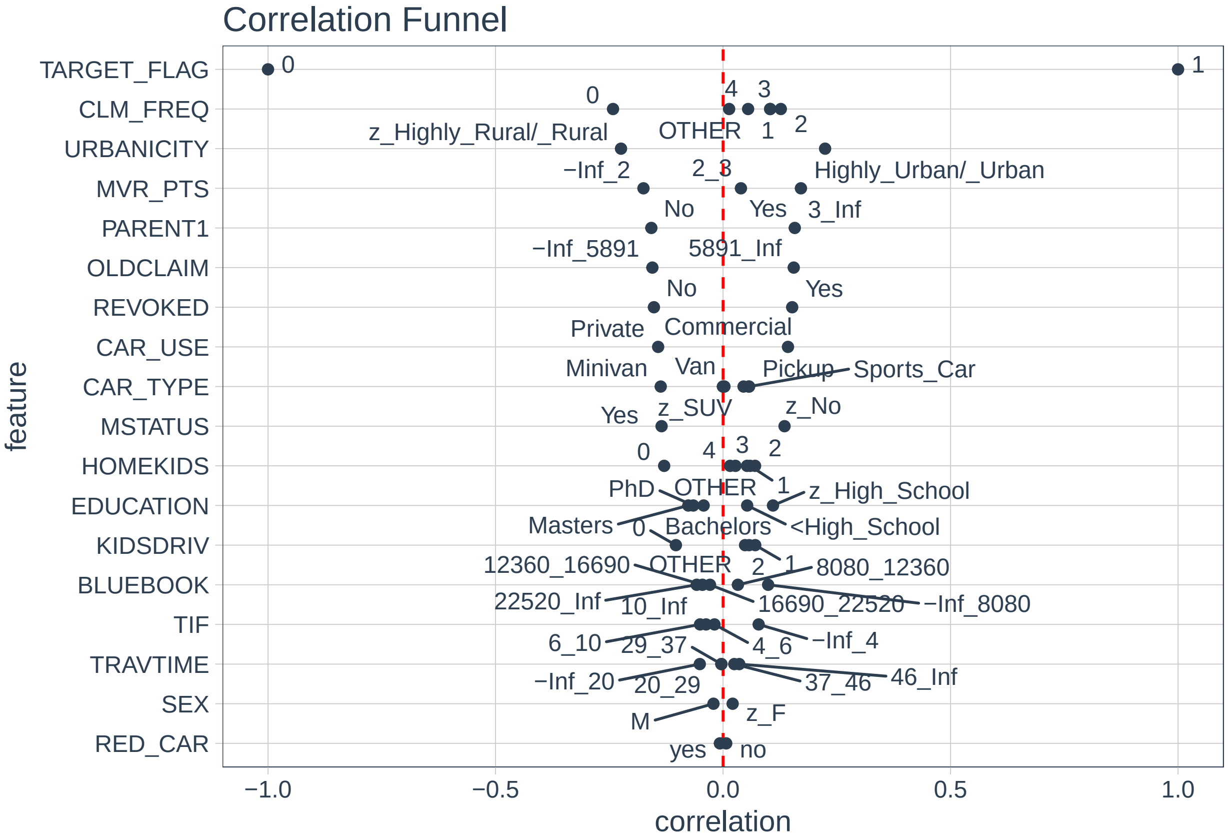
rows	8161
columns	25
all_missing_columns	0
total_missing_values	2405
complete_rows	6045

None of our columns are completely devoid of data. There are 6,045 complete rows in the dataset, which is about 74% of our observations. There are 2,405 total missing values. We take a look at which variables contain these missing values and what the spread is.



A very small percentage of observations contain missing AGE values. The INCOME, YOJ, HOME_VAL, CAR_AGE, and JOB variables are each missing around 5.5 to 6.5 percent of values. There are no variables containing such extreme proportions of missing values that removal would be warranted on that basis alone.

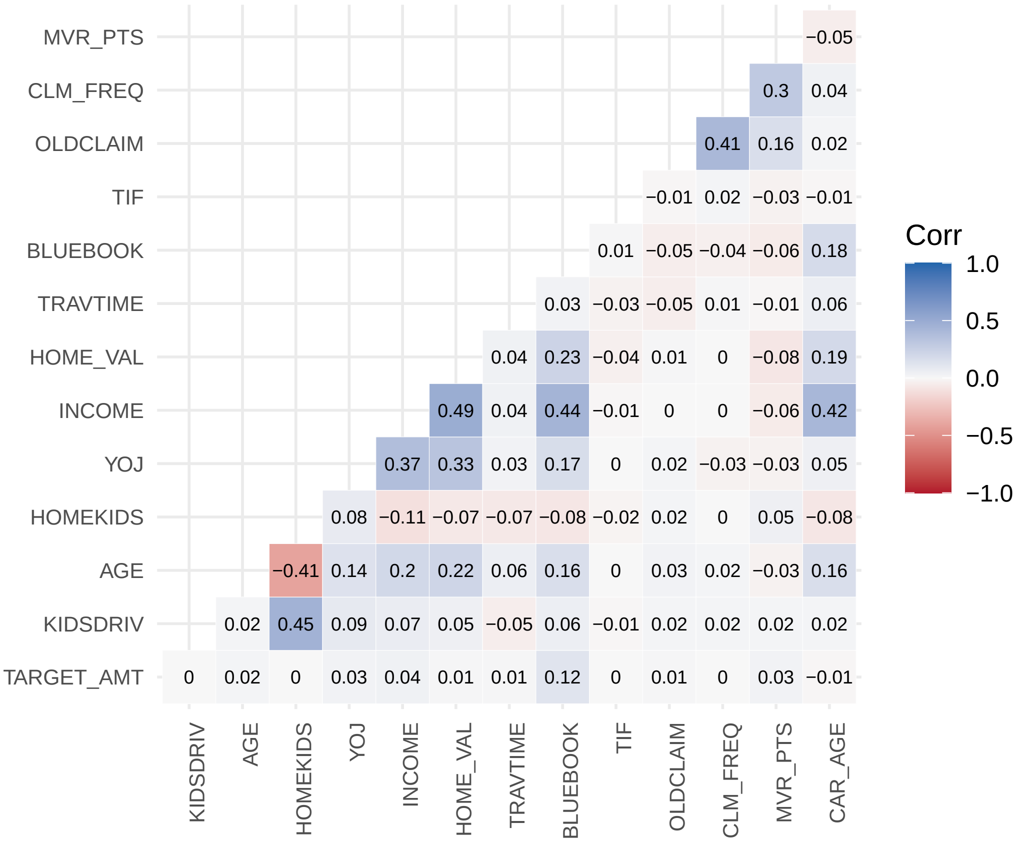
To check whether the predictor variables are correlated with the binary response variable, we produce a correlation funnel that visualizes the strength of the relationships between our predictors and TARGET_FLAG. This correlation funnel will not include variables for which there are any missing values.



The predictor variables without missing values that are most correlated with getting into a car crash are CLM_FREQ, URBANICITY, MVR_PTS, OLDCLAIM, PARENT1, REVOKED, and CAR_USE. Some of this is unsurprising. Increased claim frequency, increased numbers of traffic tickets, increased past payouts, having your license previously revoked, and using your car commercially all positively correlate with getting into a car crash, as we expected they would. We did not expect URBANICITY to be so relevant, but urban areas can often be more difficult to drive through and have more traffic, so that combination could reasonably make urban-dwellers more likely to get into car crashes, as the correlation suggests. We also did not expect PARENT1 to be so relevant, but the correlation between being a single parent and getting into a car crash is very similar to that of having your license previously revoked and getting into a car crash.

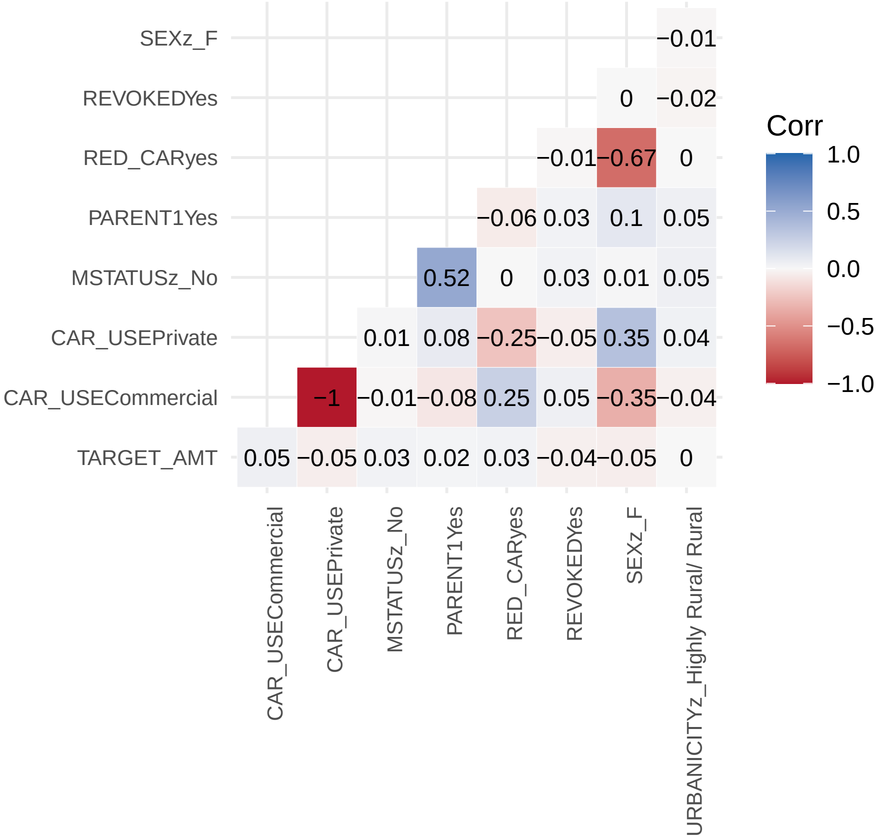
The predictor variables without missing values that are least correlated with getting into a car crash are SEX and RED_CAR. Being a woman has a very slight positive correlation with getting into a car crash, and driving a red car has a slightly negative correlation with getting into a car crash. These are contrary to urban legend, and more importantly they probably won't be useful when modeling.

To check whether the predictor variables are correlated with the numeric response variable, we produce correlation plots that visualize the strength of the relationships between our predictors and TARGET_AMT (only when observations involve a car crash, as otherwise we know TARGET_AMT $= 0$). For readability, first we look at numeric predictors only.



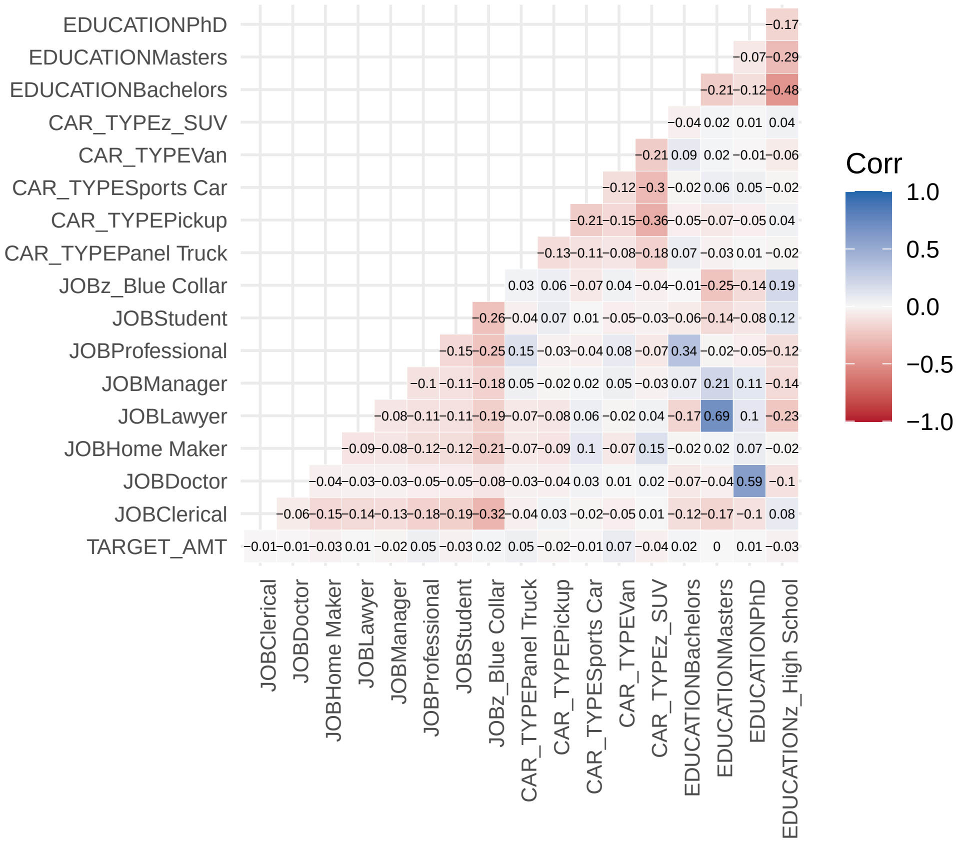
It's worth noting that `BLUEBOOK` is the single numeric variable most correlated with an increased `TARGET_AMT`, which is sensible. Cars that are currently still more valuable can be more expensive to fix. We expected `CAR_AGE` to be more negatively correlated with `TARGET_AMT`.

Next we look at two-level factors.



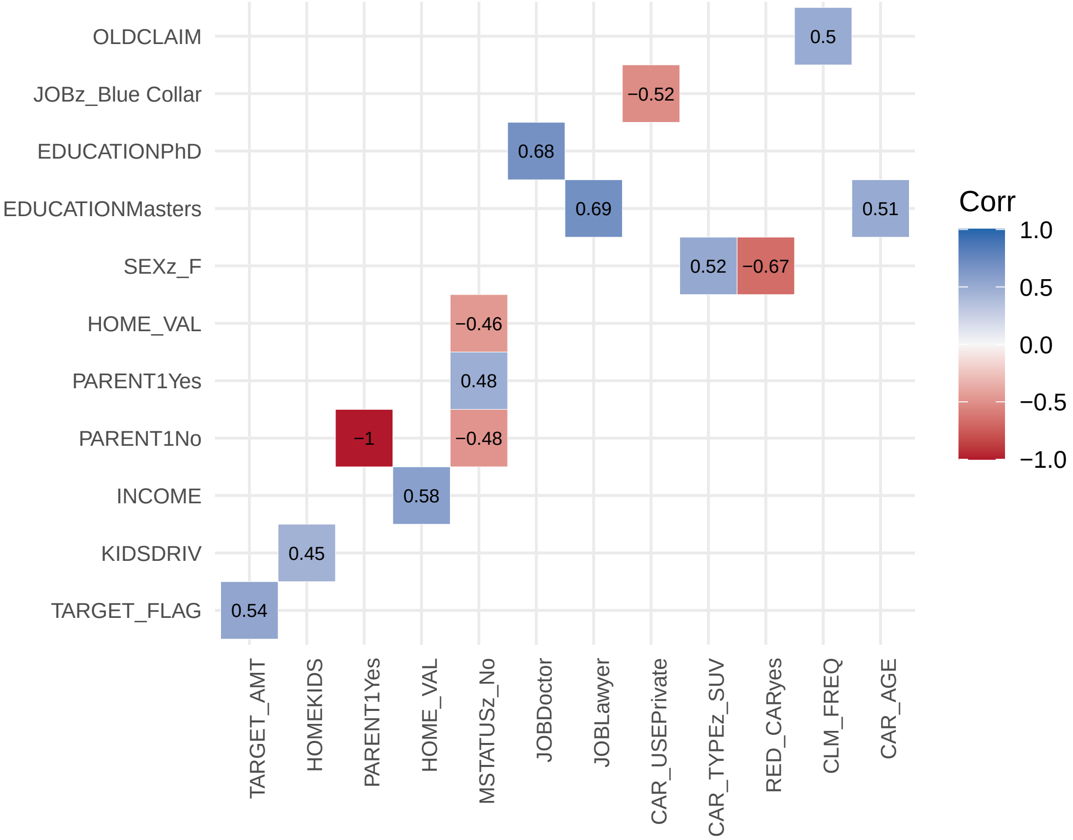
We see a small positive correlation between using your car commercially and `TARGET_AMT`. But we see an equally large negative correlation between being female and `TARGET_AMT`. The former is more logical than the latter, so neither may be a good predictor of `TARGET_AMT` ultimately.

Finally we look at factors with more than two levels.



The various car types don't have as high of a correlation (either positively or negatively) with `TARGET_AMT` as expected, but we still believe `CAR_TYPE` will be somewhat useful for modeling.

Because we have so many variables, it would be difficult to check for and visualize collinearity for our responses and predictors all at the same time without setting a threshold. So we will set a correlation threshold of 0.45 (in absolute value) and only visualize variables with any correlation values at or above that level.



We see some expected collinearity. KIDSDRIV and HOMEKIDS are moderately positively correlated because teenagers driving your car depends on you having kids at all, but the number of teens driving your car won't always exactly match the number of kids you have. HOME_VAL and INCOME are pretty positively correlated, as higher incomes lead to the ability to purchase higher valued homes. Not being married is also moderately negatively correlated with HOME_VAL, likely because married people often have two incomes instead of one and can therefore purchase higher valued homes. Having a PhD is equally correlated with being a doctor or lawyer, which makes sense because those jobs require them. Working a blue collar job is logically pretty negatively correlated with driving your car privately since driving your car commercially is itself a blue collar job. Being a woman is very negatively correlated with driving a red car. Lastly of note, claim frequency is moderately correlated with higher past payouts, which adds up.

We have 14 numeric variables and 11 categorical variables (including the dummy variable TARGET_FLAG). We list the possible ranges or values for each variable in the breakdown below:

Variable	Type	Values
AGE	Numeric	16 - 81
BLUEBOOK	Numeric	1500 - 69740
CAR_AGE	Numeric	-3 - 28
CLM_FREQ	Numeric	0 - 5
HOME_VAL	Numeric	0 - 885282
HOMEKIDS	Numeric	0 - 5
INCOME	Numeric	0 - 367030
KIDSDRIV	Numeric	0 - 4
MVR_PTS	Numeric	0 - 13
OLDCLAIM	Numeric	0 - 57037
TARGET_AMT	Numeric	0 - 107586.1
TIF	Numeric	1 - 25
TRAVTIME	Numeric	5 - 142
YOJ	Numeric	0 - 23
CAR_TYPE	Categorical	Minivan, Panel Truck, Pickup, Sports Car, Van, z_SUV
CAR_USE	Categorical	Commercial, Private
EDUCATION	Categorical	<High School, Bachelors, Masters, PhD, z_High School
JOB	Categorical	Clerical, Doctor, Home Maker, Lawyer, Manager, Professional, Student, z_Blue Collar
MSTATUS	Categorical	Yes, z_No
PARENT1	Categorical	No, Yes
RED_CAR	Categorical	no, yes
REVOKED	Categorical	No, Yes
SEX	Categorical	M, z_F
TARGET_FLAG	Categorical	0, 1
URBANICITY	Categorical	Highly Urban/ Urban, z_Highly Rural/ Rural

The ranges for `TARGET_AMT`, `HOME_VAL`, and `INCOME` all include zero, and recoding these zero values as `NA` will make analyzing summary statistics for these variables more meaningful than if we included zeroes in their calculations. (We will maintain a separate copy of the data, in which we do not introduce additional `NA` values, for later use when creating the fully imputed dataset that some of our models will rely on for completeness.)

The range for `CAR_AGE` includes -3. Since the variable can only take positive or zero values logically, and only one observation in the dataset has a negative sign, we make the assumption that the age of 3 years is correct for this observation, and the sign is simply a data entry error. We fix this observation.

Some of the factor levels are named inconsistently, so we will rename and relevel them in the next section.

Let’s take a look at the summary statistics for each variable.

```
##  TARGET_FLAG   TARGET_AMT          KIDSDRIV           AGE
##  0:6008      Min.   :    30.28   Min.   :0.0000   Min.   :16.00
##  1:2153      1st Qu.:  2609.78   1st Qu.:0.0000   1st Qu.:39.00
##              Median :  4104.00   Median :0.0000   Median :45.00
##              Mean   :  5702.18   Mean   :0.1711   Mean   :44.79
##              3rd Qu.:  5787.00   3rd Qu.:0.0000   3rd Qu.:51.00
##              Max.   :107586.14   Max.   :4.0000   Max.   :81.00
##              NA’s   :6008                         NA’s   :6
##     HOMEKIDS           YOJ           INCOME       PARENT1       HOME_VAL
##  Min.   :0.0000   Min.   : 0.0   Min.   :     5   No :7084   Min.   : 50223
##  1st Qu.:0.0000   1st Qu.: 9.0   1st Qu.: 34135   Yes:1077   1st Qu.:153074
##  Median :0.0000   Median :11.0   Median : 58438              Median :206692
##  Mean   :0.7212   Mean   :10.5   Mean   : 67259              Mean   :220621
##  3rd Qu.:1.0000   3rd Qu.:13.0   3rd Qu.: 90053              3rd Qu.:270022
##  Max.   :5.0000   Max.   :23.0   Max.   :367030              Max.   :885282
##                   NA's   :454    NA's   :1060                NA's   :2758
##  MSTATUS      SEX               EDUCATION            JOB
##  Yes :4894   M  :3786   <High School :1203   z_Blue Collar:1825
##  z_No:3267   z_F:4375   Bachelors    :2242   Clerical     :1271
##                         Masters      :1658   Professional :1117
##                         PhD          : 728   Manager      : 988
##                         z_High School:2330   Lawyer       : 835
##                                              (Other)      :1599
##                                              NA's         : 526
##     TRAVTIME           CAR_USE        BLUEBOOK         TIF
##  Min.   :  5.00   Commercial:3029   Min.   : 1500   Min.   : 1.000
##  1st Qu.: 22.00   Private   :5132   1st Qu.: 9280   1st Qu.: 1.000
##  Median : 33.00                     Median :14440   Median : 4.000
##  Mean   : 33.49                     Mean   :15710   Mean   : 5.351
##  3rd Qu.: 44.00                     3rd Qu.:20850   3rd Qu.: 7.000
##  Max.   :142.00                     Max.   :69740   Max.   :25.000
##
##         CAR_TYPE    RED_CAR       OLDCLAIM        CLM_FREQ      REVOKED
##  Minivan    :2145   no :5783   Min.   :    0   Min.   :0.0000   No :7161
##  Panel Truck: 676   yes:2378   1st Qu.:    0   1st Qu.:0.0000   Yes:1000
##  Pickup     :1389              Median :    0   Median :0.0000
##  Sports Car : 907              Mean   : 4037   Mean   :0.7986
##  Van        : 750              3rd Qu.: 4636   3rd Qu.:2.0000
##  z_SUV      :2294              Max.   :57037   Max.   :5.0000
##
##     MVR_PTS          CAR_AGE                         URBANICITY
##  Min.   : 0.000   Min.   : 0.000   Highly Urban/ Urban  :6492
##  1st Qu.: 0.000   1st Qu.: 1.000   z_Highly Rural/ Rural:1669
##  Median : 1.000   Median : 8.000
##  Mean   : 1.696   Mean   : 8.329
##  3rd Qu.: 3.000   3rd Qu.:12.000
##  Max.   :13.000   Max.   :28.000
##                   NA's   :510
```

The majority of observations live/work in a highly urban or urban area. There are more married than unmarried observations, and there are also more female than male observations. The average observation has a median age of 45 years old, has been in their job for a median of 11 years, and has a median income of roughly $58,500.00. Most cars in the dataset are driven for private use rather than commercially, and the median car age is 8 years.

6,008 observations, which is the majority of observations, do not involve car crashes, and we now correctly record 6,008 `NA` observations for `TARGET_AMT`. (Since we introduced `NA` values for `TARGET_AMT` on purpose, we will not consider imputing them.)

There are 6 `NA` values in `AGE`, 510 in `CAR_AGE`, 454 in `YOJ`, 1,060 in `INCOME`, 2,758 in `HOME_VAL`, and 526 in `JOB`. In the next section, we will impute all these missing values in an alternate version of our dataset, as we mentioned earlier, and in the main version of our dataset, we will only impute the variables if we determine their data is at least Missing at Random (MAR), and there's no other evidence we should exclude them from imputation.

We check whether there is evidence that the data are Missing Completely at Random (MCAR), a higher standard than MAR, using the `mcar_test` function from the `naniar` package. Meeting this standard is unlikely with real data, but still worth checking.

statistic	df	p.value	missing.patterns
16862.3	1116	0	51

The low p-value provides evidence that missing data on these variables are **not** MCAR.

Excluding `AGE` since the number of missing values is so small for that variable, and we plan to impute it anyway, let's check whether missingness in any of the others is associated with any of the other predictors or the response variables using the `missing_compare` function from the `finalfit` package. Due to the large number of variables, we exclude any observed variables that could not account for a variable's missingness in the output by setting a p-value threshold of 0.05.

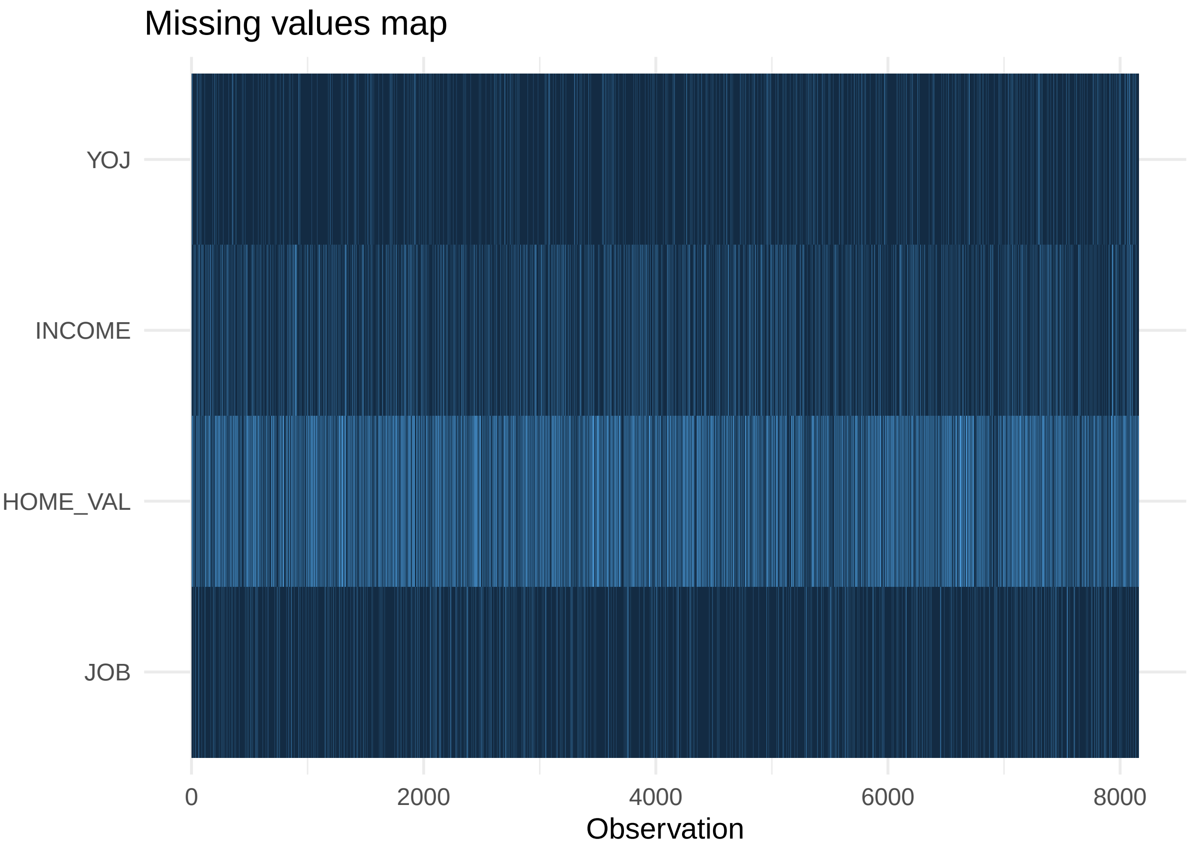
Dependant	Explanatory	Ref	Not Missing	Missing	p
INCOME	TARGET_FLAG	0	5308 (88.3)	700 (11.7)	0.001
INCOME	TARGET_FLAG	1	1793 (83.3)	360 (16.7)	NA
INCOME	AGE	Mean (SD)	44.9 (8.6)	43.8 (9.0)	0.001
INCOME	HOMEKIDS	Mean (SD)	0.7 (1.1)	0.9 (1.2)	0.001
INCOME	YOJ	Mean (SD)	11.4 (2.8)	4.3 (5.8)	0.001
INCOME	PARENT1	No	6188 (87.4)	896 (12.6)	0.022
INCOME	PARENT1	Yes	913 (84.8)	164 (15.2)	NA
INCOME	HOME_VAL	Mean (SD)	227842.0 (93771.4)	155319.7 (92741.6)	0.001
INCOME	SEX	M	3420 (90.3)	366 (9.7)	0.001
INCOME	SEX	z_F	3681 (84.1)	694 (15.9)	NA
INCOME	EDUCATION	<High School	982 (81.6)	221 (18.4)	0.001
INCOME	EDUCATION	Bachelors	1972 (88.0)	270 (12.0)	NA
INCOME	EDUCATION	Masters	1547 (93.3)	111 (6.7)	NA
INCOME	EDUCATION	PhD	652 (89.6)	76 (10.4)	NA
INCOME	EDUCATION	z_High School	1948 (83.6)	382 (16.4)	NA
INCOME	JOB	Clerical	1198 (94.3)	73 (5.7)	0.001
INCOME	JOB	Doctor	232 (94.3)	14 (5.7)	NA
INCOME	JOB	Home Maker	308 (48.0)	333 (52.0)	NA
INCOME	JOB	Lawyer	792 (94.9)	43 (5.1)	NA
INCOME	JOB	Manager	937 (94.8)	51 (5.2)	NA
INCOME	JOB	Professional	1055 (94.4)	62 (5.6)	NA
INCOME	JOB	Student	350 (49.2)	362 (50.8)	NA
INCOME	JOB	z_Blue Collar	1727 (94.6)	98 (5.4)	NA
INCOME	CAR_USE	Commercial	2675 (88.3)	354 (11.7)	0.008
INCOME	CAR_USE	Private	4426 (86.2)	706 (13.8)	NA
INCOME	BLUEBOOK	Mean (SD)	16199.2 (8430.5)	12432.0 (7574.9)	0.001
INCOME	TIF	Mean (SD)	5.4 (4.2)	5.1 (4.0)	0.045
INCOME	CAR_TYPE	Minivan	1922 (89.6)	223 (10.4)	0.001
INCOME	CAR_TYPE	Panel Truck	632 (93.5)	44 (6.5)	NA
INCOME	CAR_TYPE	Pickup	1225 (88.2)	164 (11.8)	NA
INCOME	CAR_TYPE	Sports Car	729 (80.4)	178 (19.6)	NA
INCOME	CAR_TYPE	Van	683 (91.1)	67 (8.9)	NA
INCOME	CAR_TYPE	z_SUV	1910 (83.3)	384 (16.7)	NA
INCOME	RED_CAR	no	4974 (86.0)	809 (14.0)	0.001
INCOME	RED_CAR	yes	2127 (89.4)	251 (10.6)	NA

Dependant	Explanatory	Ref	Not Missing	Missing	p
INCOME	CLM_FREQ	Mean (SD)	0.8 (1.2)	0.9 (1.2)	0.048
INCOME	CAR_AGE	Mean (SD)	8.5 (5.7)	7.2 (5.3)	0.001
INCOME	URBANICITY	Highly Urban/ Urban	5753 (88.6)	739 (11.4)	0.001
INCOME	URBANICITY	z_Highly Rural/ Rural	1348 (80.8)	321 (19.2)	NA
HOME_VAL	TARGET_FLAG	0	4217 (70.2)	1791 (29.8)	0.001
HOME_VAL	TARGET_FLAG	1	1186 (55.1)	967 (44.9)	NA
HOME_VAL	AGE	Mean (SD)	45.4 (8.5)	43.5 (8.7)	0.001
HOME_VAL	HOMEKIDS	Mean (SD)	0.7 (1.1)	0.8 (1.1)	0.009
HOME_VAL	YOJ	Mean (SD)	11.1 (3.7)	9.3 (4.6)	0.001
HOME_VAL	INCOME	Mean (SD)	68771.2 (44434.0)	63968.7 (48518.0)	0.001
HOME_VAL	PARENT1	No	5055 (71.4)	2029 (28.6)	0.001
HOME_VAL	PARENT1	Yes	348 (32.3)	729 (67.7)	NA
HOME_VAL	MSTATUS	Yes	4267 (87.2)	627 (12.8)	0.001
HOME_VAL	MSTATUS	z_No	1136 (34.8)	2131 (65.2)	NA
HOME_VAL	EDUCATION	<High School	729 (60.6)	474 (39.4)	0.001
HOME_VAL	EDUCATION	Bachelors	1545 (68.9)	697 (31.1)	NA
HOME_VAL	EDUCATION	Masters	1166 (70.3)	492 (29.7)	NA
HOME_VAL	EDUCATION	PhD	474 (65.1)	254 (34.9)	NA
HOME_VAL	EDUCATION	z_High School	1489 (63.9)	841 (36.1)	NA
HOME_VAL	JOB	Clerical	913 (71.8)	358 (28.2)	0.001
HOME_VAL	JOB	Doctor	154 (62.6)	92 (37.4)	NA
HOME_VAL	JOB	Home Maker	456 (71.1)	185 (28.9)	NA
HOME_VAL	JOB	Lawyer	596 (71.4)	239 (28.6)	NA
HOME_VAL	JOB	Manager	703 (71.2)	285 (28.8)	NA
HOME_VAL	JOB	Professional	817 (73.1)	300 (26.9)	NA
HOME_VAL	JOB	Student	100 (14.0)	612 (86.0)	NA
HOME_VAL	JOB	z_Blue Collar	1309 (71.7)	516 (28.3)	NA
HOME_VAL	CAR_USE	Commercial	1942 (64.1)	1087 (35.9)	0.002
HOME_VAL	CAR_USE	Private	3461 (67.4)	1671 (32.6)	NA
HOME_VAL	BLUEBOOK	Mean (SD)	16073.5 (8388.1)	14997.6 (8437.5)	0.001
HOME_VAL	OLDCLAIM	Mean (SD)	3726.1 (8512.2)	4646.3 (9245.6)	0.001
HOME_VAL	CLM_FREQ	Mean (SD)	0.7 (1.1)	0.9 (1.2)	0.001
HOME_VAL	REVOKED	No	4801 (67.0)	2360 (33.0)	0.001
HOME_VAL	REVOKED	Yes	602 (60.2)	398 (39.8)	NA
HOME_VAL	MVR_PTS	Mean (SD)	1.6 (2.0)	1.9 (2.3)	0.001
HOME_VAL	CAR_AGE	Mean (SD)	8.4 (5.7)	8.1 (5.7)	0.012
HOME_VAL	URBANICITY	Highly Urban/ Urban	4345 (66.9)	2147 (33.1)	0.007
HOME_VAL	URBANICITY	z_Highly Rural/ Rural	1058 (63.4)	611 (36.6)	NA
JOB	KIDSDRIV	Mean (SD)	0.2 (0.5)	0.1 (0.4)	0.005
JOB	AGE	Mean (SD)	44.7 (8.7)	46.5 (8.0)	0.001
JOB	HOMEKIDS	Mean (SD)	0.7 (1.1)	0.4 (0.9)	0.001
JOB	YOJ	Mean (SD)	10.4 (4.2)	11.3 (2.7)	0.001
JOB	INCOME	Mean (SD)	63334.1 (42157.2)	118852.9 (58861.8)	0.001
JOB	PARENT1	No	6601 (93.2)	483 (6.8)	0.001
JOB	PARENT1	Yes	1034 (96.0)	43 (4.0)	NA
JOB	HOME_VAL	Mean (SD)	213485.5 (89924.5)	322080.5 (121344.9)	0.001
JOB	SEX	M	3365 (88.9)	421 (11.1)	0.001
JOB	SEX	z_F	4270 (97.6)	105 (2.4)	NA
JOB	EDUCATION	<High School	1203 (100.0)	0 (0.0)	0.001
JOB	EDUCATION	Bachelors	2242 (100.0)	0 (0.0)	NA
JOB	EDUCATION	Masters	1330 (80.2)	328 (19.8)	NA
JOB	EDUCATION	PhD	530 (72.8)	198 (27.2)	NA

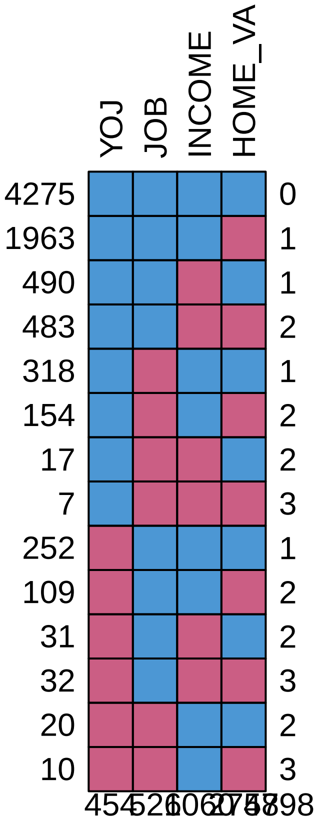
Dependant	Explanatory	Ref	Not Missing	Missing	p
JOB	EDUCATION	z_High School	2330 (100.0)	0 (0.0)	NA
JOB	CAR_USE	Commercial	2557 (84.4)	472 (15.6)	0.001
JOB	CAR_USE	Private	5078 (98.9)	54 (1.1)	NA
JOB	BLUEBOOK	Mean (SD)	15161.5 (8018.6)	23669.5 (9952.7)	0.001
JOB	CAR_TYPE	Minivan	2123 (99.0)	22 (1.0)	0.001
JOB	CAR_TYPE	Panel Truck	435 (64.3)	241 (35.7)	NA
JOB	CAR_TYPE	Pickup	1265 (91.1)	124 (8.9)	NA
JOB	CAR_TYPE	Sports Car	902 (99.4)	5 (0.6)	NA
JOB	CAR_TYPE	Van	634 (84.5)	116 (15.5)	NA
JOB	CAR_TYPE	z_SUV	2276 (99.2)	18 (0.8)	NA
JOB	RED_CAR	no	5510 (95.3)	273 (4.7)	0.001
JOB	RED_CAR	yes	2125 (89.4)	253 (10.6)	NA
JOB	OLDCLAIM	Mean (SD)	3980.4 (8722.8)	4859.5 (9501.7)	0.026
JOB	CLM_FREQ	Mean (SD)	0.8 (1.2)	1.0 (1.3)	0.001
JOB	CAR_AGE	Mean (SD)	7.9 (5.6)	14.0 (4.6)	0.001
JOB	URBANICITY	Highly Urban/ Urban	5987 (92.2)	505 (7.8)	0.001
JOB	URBANICITY	z_Highly Rural/ Rural	1648 (98.7)	21 (1.3)	NA

There is evidence that some of the missingness for `INCOME`, `HOME_VAL`, and `JOB` can be explained by other observed information, so they could be considered Missing at Random (MAR). There is no evidence missing values for `CAR_AGE` or `YOJ` can be explained by other observed information, so we will no longer consider imputing them in the main version of our dataset.

It's reasonable to assume that the missing values in `YOJ`, `HOME_VAL`, `INCOME` and `JOB` might all be related because money, employment, and assets are interconnected. Therefore the missingness of one or more of these variables might be dependent on the missingness of one or more of the others. Let's look at the overlap of observations with missing values for these variables using the `missing_plot` function from the `finalfit` package.

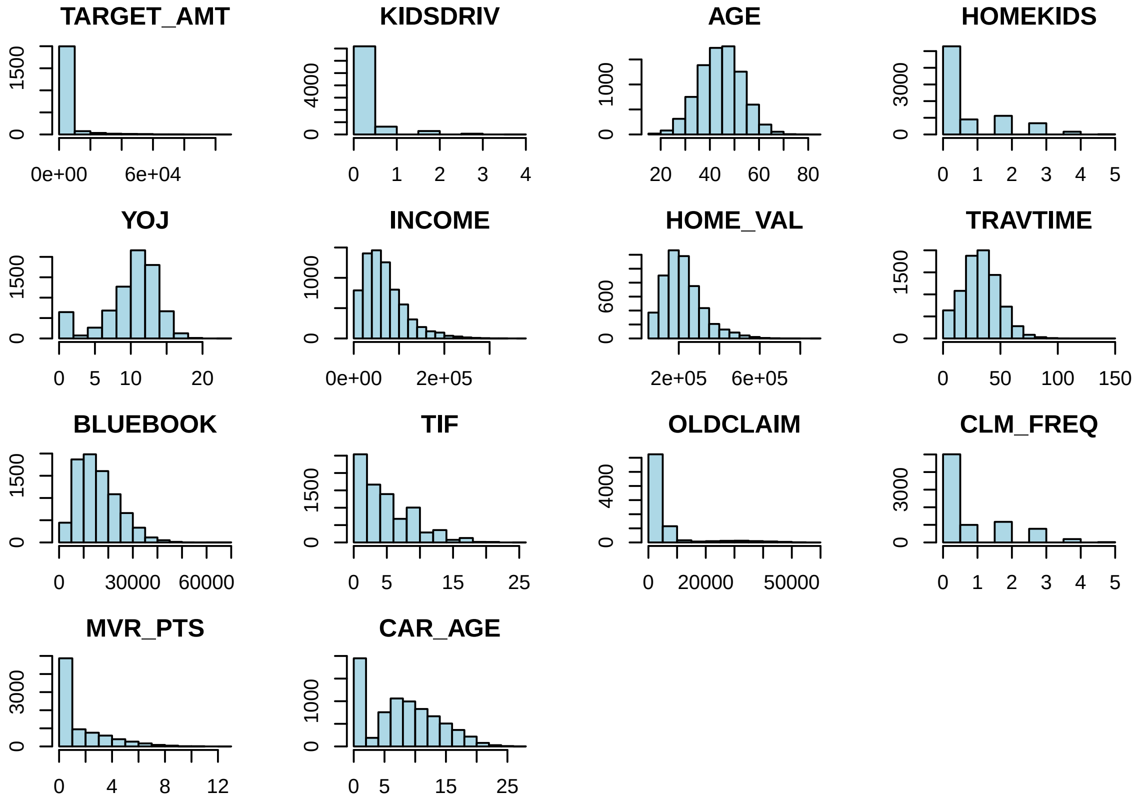


We do see some overlap in the observations that have missing values for these variables, but it's hard to detect anything more conclusive from this plot. To take a closer look at the patterns of missingness between these variables, we can use the `missing_pattern` function from the `finalfit` package. (Note that in the visualization that follows, the numbers along the bottom axis are unfortunately illegible, but they are just the column-wise counts of missing values for each variable, plus a sum of missing values for all variables, and we have already remarked on these totals.)



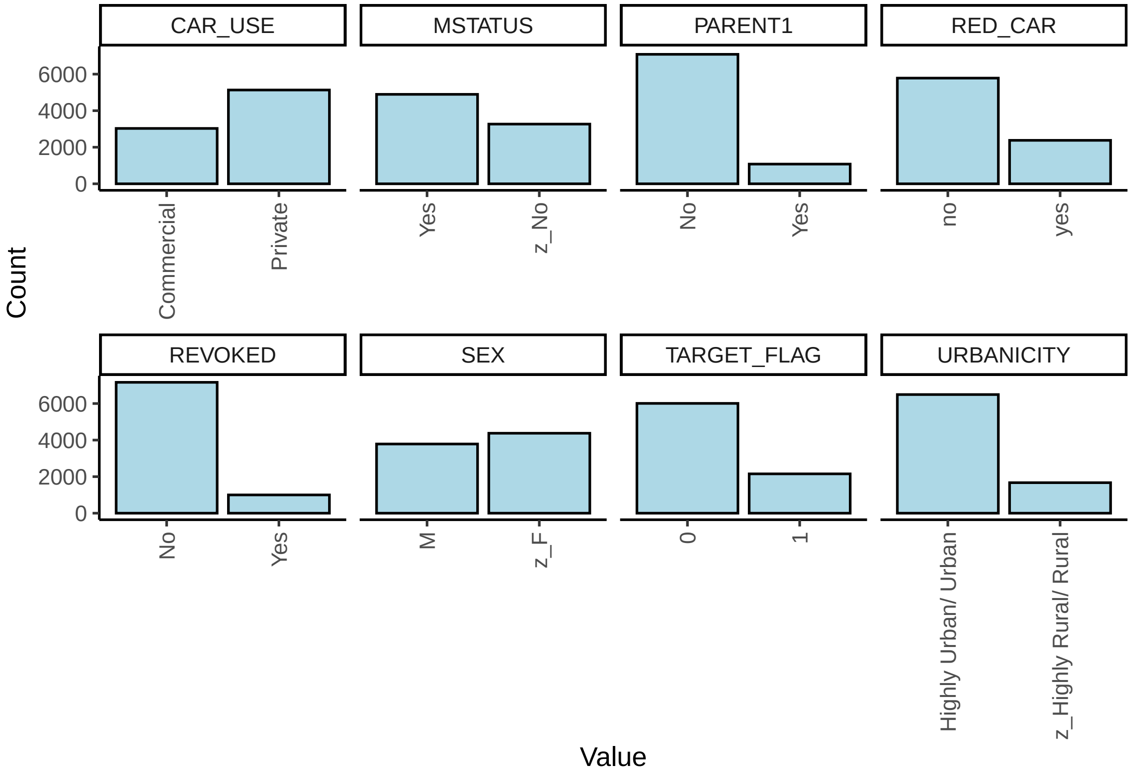
Here, we see several patterns of missingness worth noting. 814 observations are missing two out of these four variables, and 49 observations are missing three. Of the observations that are missing `HOME_VAL`, 483 are also missing `INCOME`, 154 are also missing `JOB`, and 109 are also missing `YOJ`. Due to these patterns of related missingness, we will no longer consider imputing these variables in the main version of our dataset.

Let's take a look at the distributions of the numeric variables.



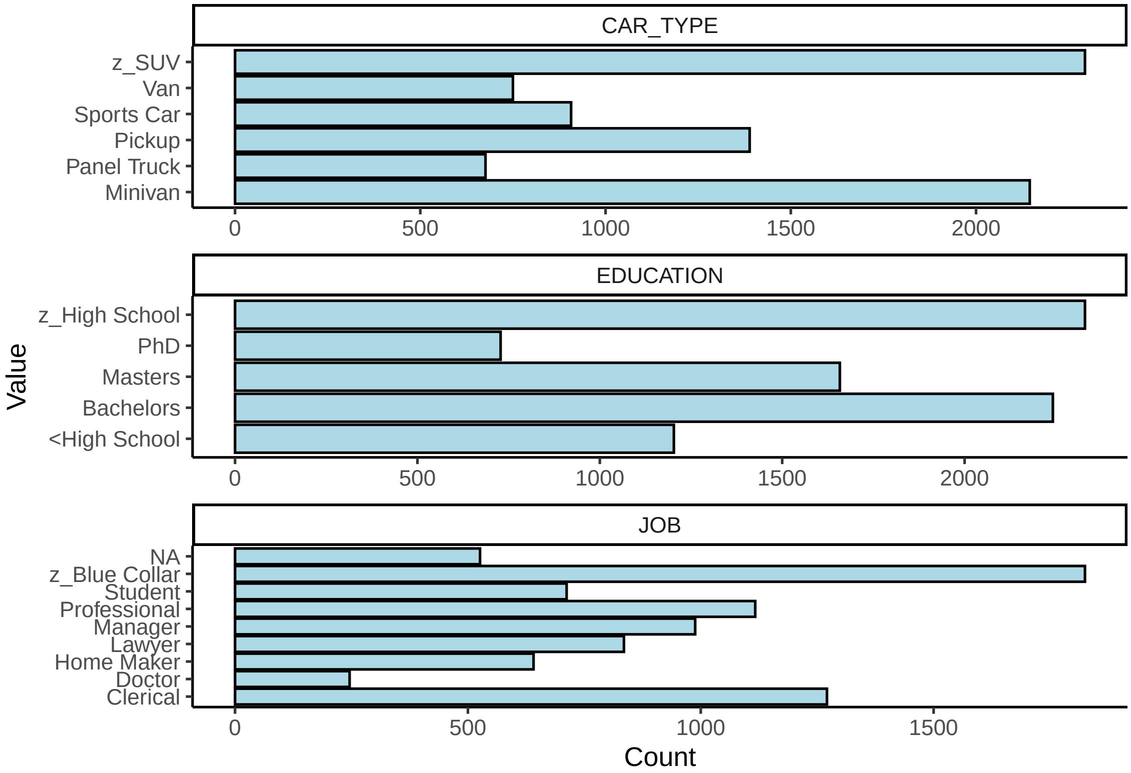
The distribution for `AGE` is approximately normal. The distribution for `YOJ` is left-skewed. The distributions for `TARGET_AMT`, `KIDSDRIV`, `HOMEKIDS`, `INCOME`, `HOME_VAL`, `TRAVTIME`, `BLUEBOOK`, `TIF`, `OLDCLAIM`, `CLM_FREQ`, `MVR_PTS`, and `CAR_AGE` are all right-skewed. 75% of observations for `TARGET_AMT` are at or below $5,787.00, but the maximum value recorded is $107,586.14.

Let's also take a look at the distributions of the categorical variables. First, we look at the distributions for categorical variables with only two levels.



Looking at `PARENT1` and `REVOKED`, we can see that single parents represent relatively few observations in the dataset, as do people whose licenses were revoked in the past seven years. `MSTATUS` and `SEX` are the most evenly split categorical variables with two levels in the dataset.

Next we look at the distributions for the categorical variables with more than two levels.



The most common profession represented in the observations is blue collar, and the most commonly represented cars are the SUV and the minivan. The number of observations with high school diplomas and bachelor's degrees are fairly similar. Having less or more education is less common.

Data Preparation

First, we rename and relevel the inconsistently named and leveled factor variables we noted earlier. A summary of only the factors we changed the levels for is below, with the first level in each list always being the reference level. For variables which have "Yes" and "No" values, we will replace these with 1/0 (1 = "Yes", 0 = "No").

Factor	New Levels
CAR_TYPE	Minivan, Panel Truck, Pickup, Sports Car, SUV, Van
EDUCATION	<High School, High School, Bachelors, Masters, PhD
JOB	Blue Collar, Clerical, Doctor, Home Maker, Lawyer, Manager, Professional, Student
PARENT1	0, 1
MSTATUS	0, 1
RED_CAR	0, 1
REVOKED	0, 1
SEX	Male, Female
URBANICITY	0, 1

We reduce the scale of the `INCOME` and `HOME_VAL` variables to thousands of dollars so the figures will be more readable when visualized. The replacement variables are `INCOME_THOU` and `HOME_VAL_THOU`.

Some observations list `Student` as their occupation as well as a value for `YOJ`. We recode these values as `NA`. The most likely interpretation is that people incorrectly listed how many years they've been in school here, which will not be useful to our analysis.

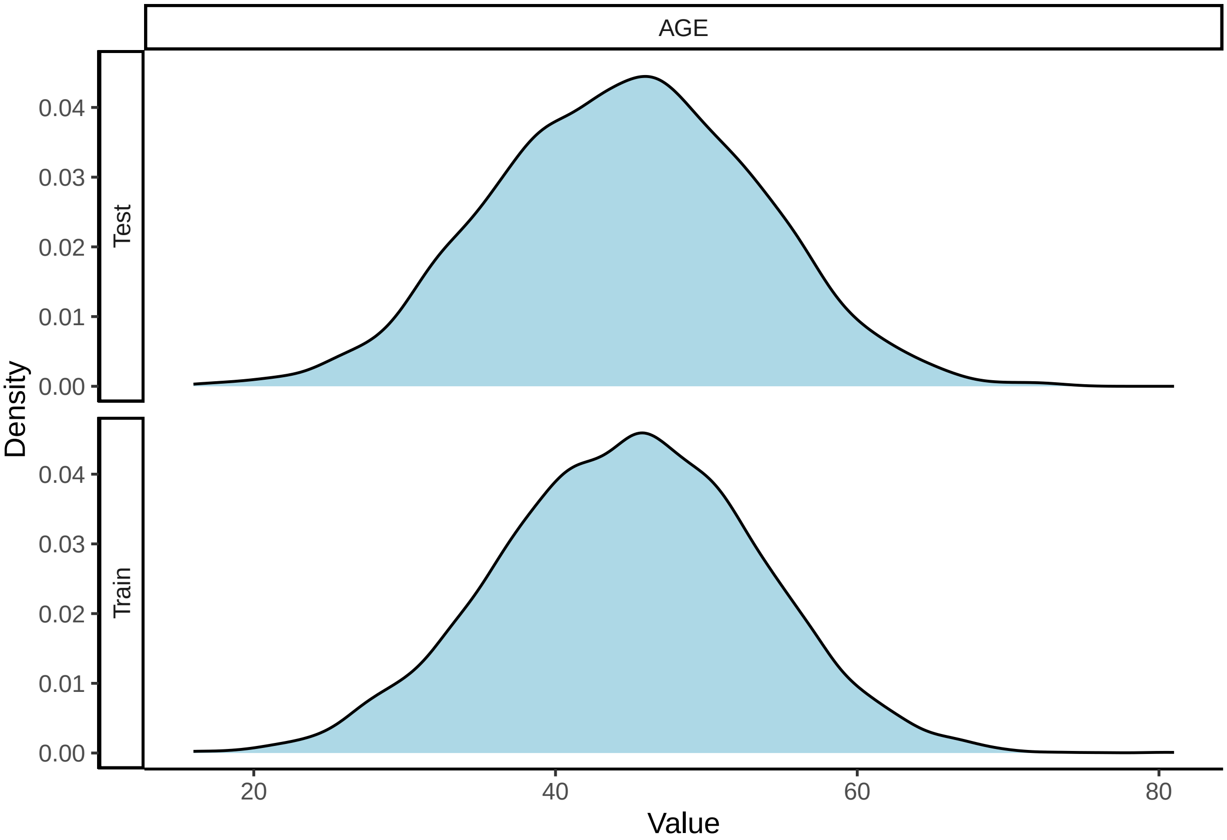
Based on the descriptions of some of the variables and their theoretical effects on the target variables, and to handle the variables that have missing data that we chose not to impute, including those for which we replaced zero or incorrect values with NA values, we create several factors that we believe will be helpful when building models:

- HOME_VAL_CAT (Levels based on HOME_VAL_THOU = "<=250K", "251-500K", "501-750K", "751K+", "")
- HOMEOWNER (1 = HOME_VAL_THOU not NA)
- INCOME_CAT (Levels based on INCOME = "<=50K", "51-100K", "101-150K", "151K+", "")
- INCOME_FLAG (1 = INCOME_THOU not NA)
- KIDSDRIV_FLAG (1 = KIDSDRIV number of children > 0)
- HOMEKIDS_FLAG (1 = HOMEKIDS number of children > 0)
- EMPLOYED (1 = JOB not NA/Student/Home Maker)
- CAR_AGE_CAT (Levels based on CAR_AGE = "<=4", "5-8", "9-12", "13+", "")
- WHITE_COLLAR (1 = JOB not NA/Student/Home Maker/Blue Collar)

We then split both the main version of our dataset and the alternate version we created earlier into train and test sets. The main version will have all the derived variables we just created, imputed values for the AGE variable, and any transformations we make. The alternate version will not include any derived variables or transformations, but it will include imputed values for all variables with missing values.

We impute missing data in the main train and test sets for one numeric variable, AGE, using the mean value since it is normally distributed.

We take a look at the distributions for our imputed variable to see if the distributions of this variable in the train and test sets differ from what we originally observed or between sets.



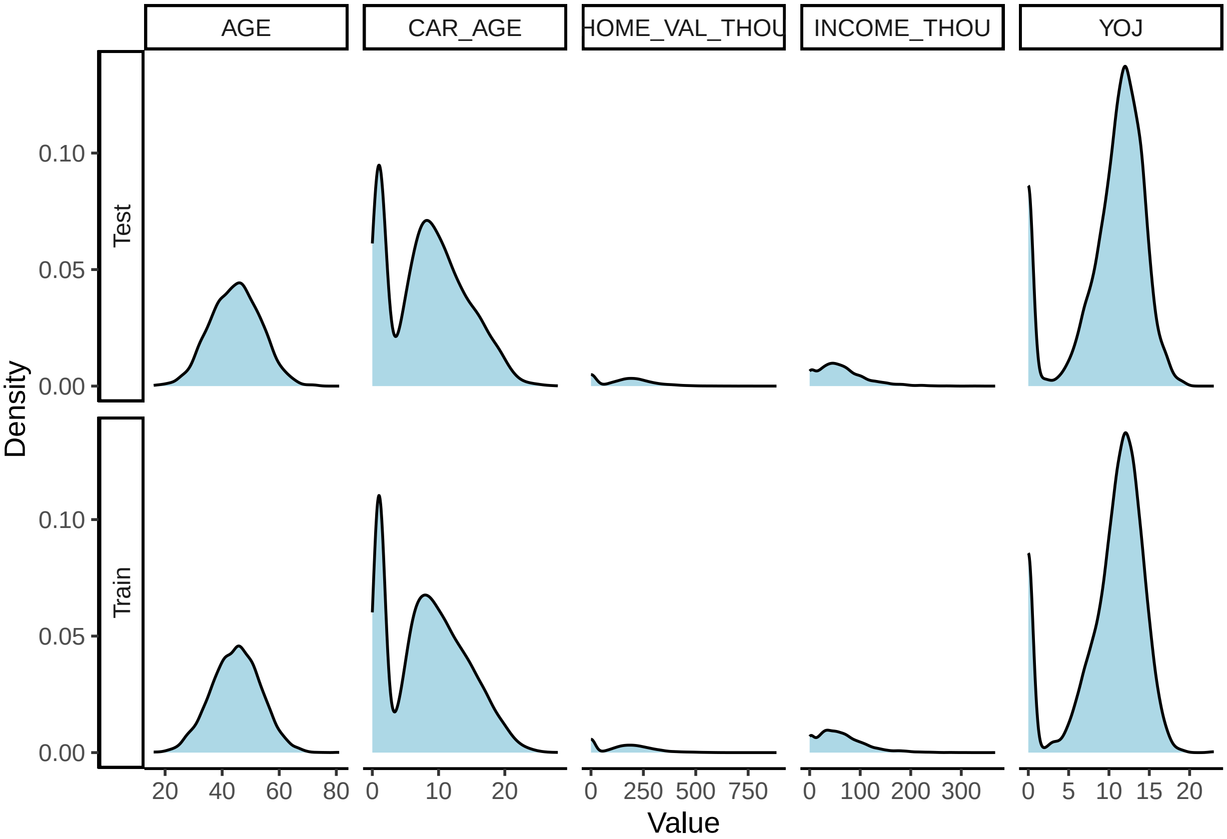
The distributions in the train and test sets for `AGE` are similar to each other and to its original distribution.

We impute missing data in the alternate train and test sets for all variables with missing values using the `mice` package.

We confirm there are no longer any missing values in the alternate train or test datasets.

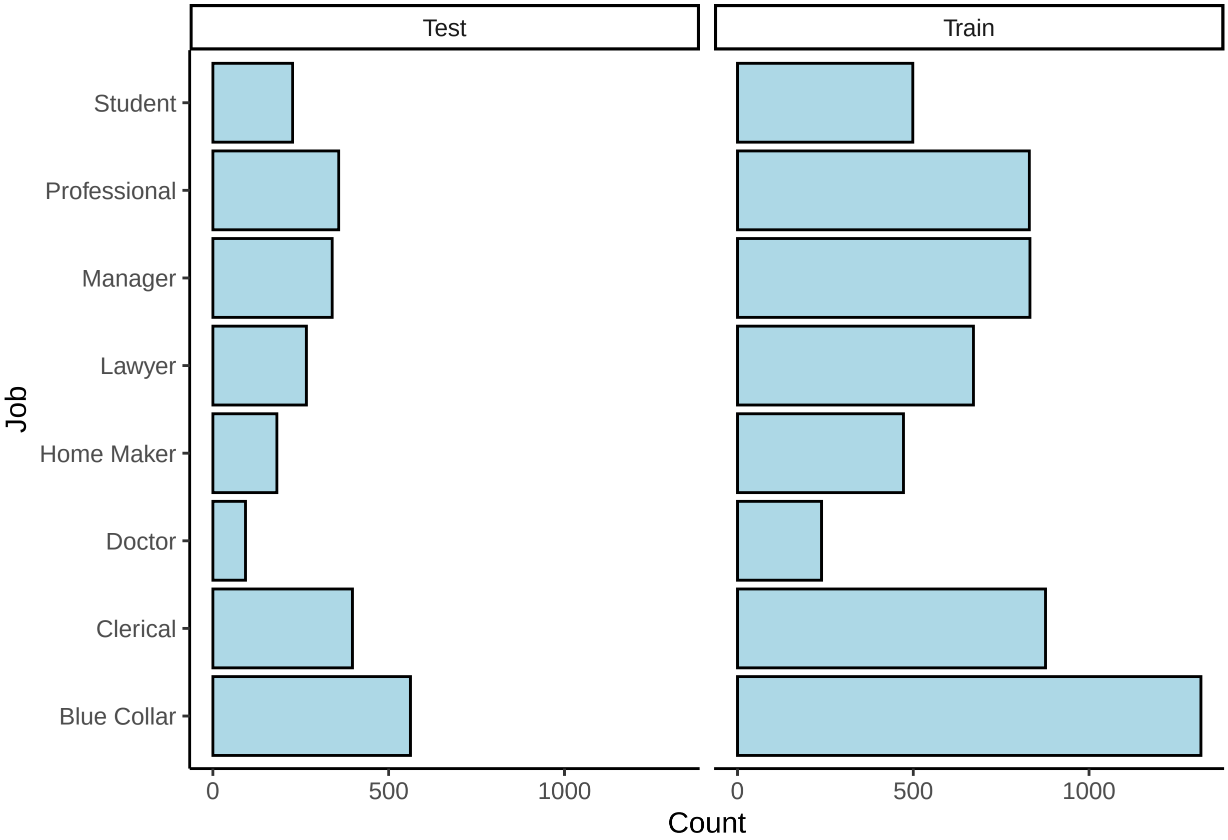
```
## [1] TRUE
```

We take a look at the distributions for the imputed numeric variables to see if their distributions in the alternate train and test sets differ from what we originally observed or between sets.



The distributions for the imputed numeric variables don't differ between the alternate train and test sets or from what we originally observed.

We also perform the same check for the single categorical variable we imputed in the alternate train and test sets: `JOB`.



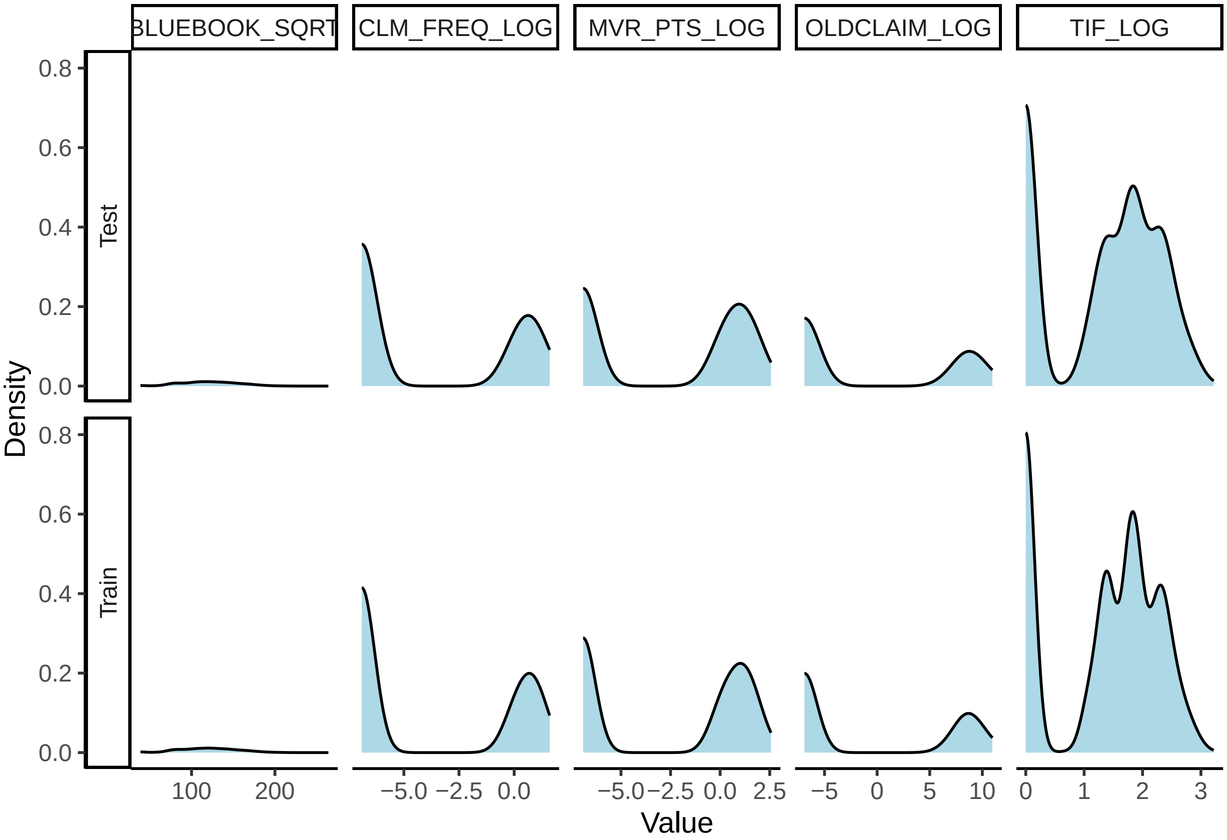
The distributions in the alternate train and test sets for the single imputed categorical variable, `JOB`, are similar to each other, and the rankings of most frequent to least frequent occupation here are similar to the rankings of the original distribution. We note that the "Professional" and "Manager" occupations are more tied in the rankings here than they were in the original distribution, however.

Since the distributions of some of our numeric variables are skewed, we transform the data for some of them. In the main dataset, we exclude any numeric variables with missing values that we decided not to impute and for which we have already created factors, as well as the response variable `TARGET_AMT`. We also use the alternate dataset, which as a reminder has no missing values, as the basis for a third version of the data, in which every skewed numeric predictor and the response variable `TARGET_AMT` have all been transformed.

Below is a breakdown of the variables, the ideal labmdas proposed by Box-Cox, and the reasonable alternative transformations we have chosen to make in the main dataset:

Skewed Variable	Ideal Lambda Proposed by Box-Cox	Reasonable Alternative Transformation
TRAVTIME	0.7	no transformation
BLUEBOOK	0.45	square root
TIF	0.25	log
OLDCLAIM	-0.0999999999999999	log
CLM_FREQ	-0.2	log
MVR_PTS	0.0500000000000003	log

We check whether the distributions of the transformed variables now differ between the train and test sets.



They do not.

Below is a breakdown of the variables, the ideal labmdas proposed by Box-Cox, and the reasonable alternative transformations we have chosen to make in the third version of the data.

Skewed Variable	Ideal Lambda Proposed by Box-Cox	Reasonable Alternative Transformation
TARGET_AMT	-0.2	log
YOJ	0.65	no transformation
TRAVTIME	0.7	no transformation
KIDSDRIV	-1.15	inverse
HOMEKIDS	-0.25	log
BLUEBOOK	0.45	square root
TIF	0.25	log
OLDCLAIM	-0.0999999999999999	log
CLM_FREQ	-0.2	log
MVR_PTS	0.0500000000000003	log
INCOME_THOU	0.45	square root
HOME_VAL_THOU	0.2	log
CAR_AGE	0.55	square root

Build Models

Binary Logistic Regression Models

Model BLR:1 - Full Model Using Original, Untransformed Variables, with All Missing Values Imputed - Reduced via Stepwise AIC Model Selection We create Model BLR:1, our baseline binary logistic regression model based on all the original, untransformed variables, with all missing values imputed so that no observations or predictors have to be excluded from the model. Then we perform stepwise model selection to select the model with the smallest AIC value using the `stepAIC()` function from the `MASS` package.

A summary of Model BLR:1 is below:

```
##
## Call:
## glm(formula = TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + MSTATUS +
##     EDUCATION + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE +
##     OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY +
##     INCOME_THOU + HOME_VAL_THOU, family = "binomial", data = alt_train_df_imputed)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2115  -0.7103  -0.3946   0.6089   3.1675
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -2.650e+00  2.392e-01 -11.081  < 2e-16 ***
## KIDSDRIV               4.158e-01  7.151e-02   5.814 6.09e-09 ***
## HOMEKIDS               5.995e-02  4.059e-02   1.477 0.139653
## PARENT11               3.424e-01  1.302e-01   2.630 0.008531 **
## MSTATUS1              -5.248e-01  1.019e-01  -5.149 2.62e-07 ***
## EDUCATIONHigh School -2.886e-01  1.383e-01  -2.086 0.036951 *
## EDUCATIONBachelors    -3.325e-02  1.132e-01  -0.294 0.768865
## EDUCATIONMasters      -2.060e-01  2.056e-01  -1.002 0.316383
## EDUCATIONPhD           9.573e-02  2.454e-01   0.390 0.696413
## JOBClerical            1.265e-01  1.265e-01   1.000 0.317240
## JOBDoctor             -7.704e-01  2.858e-01  -2.696 0.007022 **
## JOBHome Maker         -8.142e-03  1.669e-01  -0.049 0.961080
## JOBLawyer             -7.256e-02  1.984e-01  -0.366 0.714507
## JOBManager            -8.184e-01  1.538e-01  -5.320 1.04e-07 ***
## JOBProfessional       -2.059e-01  1.383e-01  -1.489 0.136444
## JOBStudent            -2.063e-01  1.474e-01  -1.400 0.161584
## TRAVTIME               1.636e-02  2.248e-03   7.277 3.42e-13 ***
## CAR_USEPrivate        -7.800e-01  1.049e-01  -7.435 1.05e-13 ***
## BLUEBOOK              -1.894e-05  5.645e-06  -3.356 0.000791 ***
## TIF                   -5.528e-02  8.941e-03  -6.182 6.31e-10 ***
## CAR_TYPEPanel Truck    5.569e-01  1.803e-01   3.089 0.002009 **
## CAR_TYPEPickup         5.279e-01  1.201e-01   4.397 1.10e-05 ***
## CAR_TYPESports Car     9.627e-01  1.278e-01   7.535 4.87e-14 ***
## CAR_TYPESUV            6.849e-01  1.033e-01   6.628 3.41e-11 ***
## CAR_TYPEVan            6.620e-01  1.449e-01   4.568 4.93e-06 ***
## OLDCLAIM              -1.496e-05  4.689e-06  -3.190 0.001425 **
## CLM_FREQ               1.833e-01  3.384e-02   5.417 6.06e-08 ***
## REVOKED1               9.225e-01  1.105e-01   8.347  < 2e-16 ***
## MVR_PTS                1.312e-01  1.627e-02   8.066 7.24e-16 ***
## CAR_AGE               -1.420e-02  9.019e-03  -1.574 0.115487
## URBANICITY1            2.385e+00  1.336e-01  17.849  < 2e-16 ***
## INCOME_THOU           -4.250e-03  1.335e-03  -3.184 0.001452 **
## HOME_VAL_THOU         -1.392e-03  4.149e-04  -3.354 0.000797 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6640.0  on 5736  degrees of freedom
## Residual deviance: 5107.1  on 5704  degrees of freedom
## AIC: 5173.1
##
## Number of Fisher Scoring iterations: 5
```

The AIC of Model BLR:1 is 5173.1.

Feature	Coefficient	Percentage Change in Odds of Car Crash
URBANICITY1	10.8643928	986.4
CAR_TYPESports Car	2.6188460	161.9
REVOKED1	2.5155223	151.6
CAR_TYPESUV	1.9835867	98.4
CAR_TYPEVan	1.9386614	93.9
CAR_TYPEPanel Truck	1.7452765	74.5
CAR_TYPEPickup	1.6953038	69.5
KIDSDRIV	1.5155229	51.6
PARENT11	1.4083716	40.8
CLM_FREQ	1.2011971	20.1
MVR_PTS	1.1402160	14.0
JOBClerical	1.1348452	13.5
EDUCATIONPhD	1.1004642	10.0
HOMEKIDS	1.0617818	6.2
TRAVTIME	1.0164960	1.6
BLUEBOOK	0.9999811	0.0
OLDCLAIM	0.9999850	0.0
HOME_VAL_THOU	0.9986093	-0.1
INCOME_THOU	0.9957593	-0.4
JOBHome Maker	0.9918908	-0.8
CAR_AGE	0.9859046	-1.4
EDUCATIONBachelors	0.9672964	-3.3
TIF	0.9462246	-5.4
JOBLawyer	0.9300077	-7.0
EDUCATIONMasters	0.8137984	-18.6
JOBProfessional	0.8139173	-18.6
JOBStudent	0.8135512	-18.6
EDUCATIONHigh School	0.7493022	-25.1
MSTATUS1	0.5916818	-40.8
JOBDoctor	0.4628116	-53.7
CAR_USEPrivate	0.4583885	-54.2
JOBManager	0.4411460	-55.9

The coefficients for Model BLR:1 mostly match expectations. Using your car privately is one of the biggest reducers of the odds of a car crash. While we expected more educated people to drive more safely, having a high school education is the level that reduces the odds of a car crash the most. All non-blue collar jobs reduce the odds of a car crash, with doctor and manager seeing the largest reductions. The biggest increaser of the odds of a car crash is living/working in an urban area. Some other notable increasers are driving anything other than a minivan, especially a sports car; having had your license revoked; and having teenagers driving your car.

We check for possible multicollinearity within this model.

```
##                   GVIF Df GVIF^(1/(2*Df))
## KIDSDRIV       1.306531  1        1.143036
## HOMEKIDS       1.830130  1        1.352823
## PARENT1        1.899743  1        1.378312
## MSTATUS        2.139221  1        1.462608
## EDUCATION      9.505786  4        1.325100
## JOB           12.372591  7        1.196831
## TRAVTIME       1.041548  1        1.020562
## CAR_USE        2.229513  1        1.493155
## BLUEBOOK       1.756755  1        1.325426
## TIF            1.010771  1        1.005371
## CAR_TYPE       2.573303  5        1.099130
## OLDCLAIM       1.665731  1        1.290632
## CLM_FREQ       1.459245  1        1.207992
## REVOKED        1.339087  1        1.157189
## MVR_PTS        1.150930  1        1.072814
## CAR_AGE        2.140672  1        1.463104
## URBANICITY     1.142613  1        1.068931
## INCOME_THOU    2.747964  1        1.657698
## HOME_VAL_THOU  2.009943  1        1.417725
```

EDUCATION and JOB only appear to have high variance inflation factors artificially, as these variables have higher degrees of freedom. A different metric is calculated for variables like this (GVIFˆ(1/(2*Df)), and that metric squared is typically considered acceptable if it is less than five, the usual VIF threshold. So we don't need to remove either EDUCATION or JOB.

Model BLR:2 - Select Model Using Original & Derived, but Untransformed Variables, with Only AGE Values Imputed - Reduced via Stepwise AIC Model Selection We create Model BLR:2, a second binary logistic regression model based on one combination of variables we believe could be the best predictors of TARGET_FLAG, including some original variables and some variables we derived from other variables, but no transformed variables. The only value we've imputed for this model is AGE.

A summary of Model BLR:2 is below:

```
##
## Call:
## glm(formula = TARGET_FLAG ~ AGE + CLM_FREQ + HOMEOWNER + INCOME_FLAG +
##     EMPLOYED + WHITE_COLLAR + MSTATUS + PARENT1 + REVOKED + SEX +
##     TRAVTIME, family = "binomial", data = train_df_imputed)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1225  -0.7667  -0.5689   0.9085   2.2217
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.593521   0.224284  -2.646 0.008138 **
## AGE           -0.014439   0.003959  -3.647 0.000266 ***
## CLM_FREQ       0.374499   0.025589  14.635  < 2e-16 ***
## HOMEOWNER1    -0.262771   0.081438  -3.227 0.001253 **
## INCOME_FLAG1  -0.478040   0.092563  -5.164 2.41e-07 ***
## EMPLOYED1      0.473987   0.107889   4.393 1.12e-05 ***
## WHITE_COLLAR1 -0.644178   0.075688  -8.511  < 2e-16 ***
## MSTATUS1      -0.241994   0.084976  -2.848 0.004402 **
## PARENT11       0.487306   0.103513   4.708 2.51e-06 ***
## REVOKED1       0.904666   0.087874  10.295  < 2e-16 ***
## SEXFemale      0.110879   0.065567   1.691 0.090822 .
## TRAVTIME       0.008946   0.001991   4.494 6.99e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6640  on 5736  degrees of freedom
## Residual deviance: 5993  on 5725  degrees of freedom
## AIC: 6017
##
## Number of Fisher Scoring iterations: 4
```

The AIC of Model BLR:2 is 6017.

Feature	Coefficient	Percentage Change in Odds of Car Crash
REVOKED1	2.4711065	147.1
PARENT11	1.6279239	62.8
EMPLOYED1	1.6063864	60.6
CLM_FREQ	1.4542633	45.4
SEXFemale	1.1172602	11.7
TRAVTIME	1.0089865	0.9
AGE	0.9856651	-1.4
MSTATUS1	0.7850611	-21.5
HOMEOWNER1	0.7689178	-23.1
INCOME_FLAG1	0.6199974	-38.0
WHITE_COLLAR1	0.5250941	-47.5

In Model BLR:2, the largest reducer of the odds of being in a car crash is working a white collar job, and the largest odds increaser is having your license revoked. Being employed at all, i.e. having any job other than student or homemaker, strangely increases the odds. Since we understand the effects of the `WHITE_COLLAR` factor better than we understand the effects of the `EMPLOYED` factor, and they both describe the same information, we favor the `WHITE_COLLAR` factor here and remove the `EMPLOYED` factor. We don't reprint a summary, but the new AIC is 6034.6. We've mentioned before that we don't understand being a single parent's correlation with increased car crash odds, but it is worth noting it's the second largest increaser of odds in this subset of predictors. Lastly, being a woman also slightly increases the odds of a car crash despite our prior expectations.

We check for possible multicollinearity within this model.

```
##          AGE     CLM_FREQ    HOMEOWNER  INCOME_FLAG WHITE_COLLAR      MSTATUS
##     1.152491     1.003834     1.456116     1.036460     1.059972     1.732481
##      PARENT1      REVOKED          SEX     TRAVTIME
##     1.485383     1.002159     1.043544     1.005053
```

All of the variance inflation factors are less than five, so there are no issues of multicollinearity within this model.

Model BLR:3 - Select Model Using Original, Derived, & Transformed Variables, with Only AGE Values Imputed - Reduced via Stepwise AIC Model Selection We create Model BLR:3, a third binary logistic regression model based on another combination of variables we believe could be the best predictors of TARGET_FLAG, including some original variables, some variables we derived from other variables, and some variables we transformed. The only value we've imputed for this model is AGE.

```
##  [1] "AGE"           "CLM_FREQ_LOG"  "URBANICITY"    "MVR_PTS_LOG"
##  [5] "OLDCLAIM_LOG"  "PARENT1"       "REVOKED"       "CAR_USE"
##  [9] "CAR_TYPE"      "MSTATUS"       "EDUCATION"     "KIDSDRIV_FLAG"
## [13] "INCOME_CAT"    "EMPLOYED"      "HOMEOWNER"     "WHITE_COLLAR"
```

In choosing some of these variables, we excluded others for which collinearity might be an issue. That is, our factor describing income was chosen over the home value factor, the kids driving factor was chosen over the kids at home factor, and the education factor was chosen over the job factor.

A summary of Model BLR:3 is below:

```
##
## Call:
## glm(formula = TARGET_FLAG ~ AGE + CLM_FREQ_LOG + URBANICITY +
##     MVR_PTS_LOG + OLDCLAIM_LOG + PARENT1 + REVOKED + CAR_USE +
##     CAR_TYPE + MSTATUS + EDUCATION + KIDSDRIV_FLAG + INCOME_CAT +
##     EMPLOYED + HOMEOWNER + WHITE_COLLAR, family = "binomial",
##     data = train_df_trans)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1421  -0.7339  -0.4448   0.6908   3.2199
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -0.923096   0.381740  -2.418 0.015601 *
## AGE                 -0.008696   0.004280  -2.032 0.042169 *
## CLM_FREQ_LOG         0.248283   0.074101   3.351 0.000806 ***
## URBANICITY1          2.089626   0.130303  16.037  < 2e-16 ***
## MVR_PTS_LOG          0.061683   0.009240   6.676 2.46e-11 ***
## OLDCLAIM_LOG        -0.088658   0.035745  -2.480 0.013129 *
## PARENT11             0.315903   0.116011   2.723 0.006468 **
## REVOKED1             0.857847   0.099265   8.642  < 2e-16 ***
## CAR_USEPrivate      -0.736742   0.104881  -7.025 2.15e-12 ***
## CAR_TYPEPanel Truck  0.151060   0.159413   0.948 0.343335
## CAR_TYPEPickup       0.525694   0.116545   4.511 6.46e-06 ***
## CAR_TYPESports Car   1.015507   0.123022   8.255  < 2e-16 ***
## CAR_TYPESUV          0.756766   0.099151   7.632 2.30e-14 ***
## CAR_TYPEVan          0.496401   0.139351   3.562 0.000368 ***
## MSTATUS1            -0.497166   0.092775  -5.359 8.38e-08 ***
## EDUCATIONHigh School -0.191783   0.109515  -1.751 0.079912 .
## EDUCATIONBachelors  -0.718931   0.122381  -5.875 4.24e-09 ***
## EDUCATIONMasters    -0.786428   0.137006  -5.740 9.46e-09 ***
## EDUCATIONPhD        -1.095949   0.167519  -6.542 6.06e-11 ***
## KIDSDRIV_FLAG1       0.720830   0.101306   7.115 1.12e-12 ***
## INCOME_CAT.L         0.092508   0.076456   1.210 0.226299
## INCOME_CAT.Q         0.368222   0.088759   4.149 3.35e-05 ***
## INCOME_CAT.C         0.233418   0.086987   2.683 0.007289 **
## EMPLOYED1             0.169179   0.126425   1.338 0.180839
## HOMEOWNER1           -0.252892   0.086945  -2.909 0.003630 **
## WHITE_COLLAR1        -0.119849   0.105581  -1.135 0.256317
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6640.0  on 5736  degrees of freedom
## Residual deviance: 5330.8  on 5711  degrees of freedom
## AIC: 5382.8
##
## Number of Fisher Scoring iterations: 5
```

We remove the least statistically significant variable, `WHITE_COLLAR`, check the new summary, remove the only remaining statistically insignificant variable, `EMPLOYED`, and reprint only the final summary. We’re slightly surprised these variables were significant to the previous model, but not this one. However, that could be because the `INCOME_CAT` factor supersedes both in this model.

```
##
## Call:
## glm(formula = TARGET_FLAG ~ AGE + CLM_FREQ_LOG + URBANICITY +
##     MVR_PTS_LOG + OLDCLAIM_LOG + PARENT1 + REVOKED + CAR_USE +
##     CAR_TYPE + MSTATUS + EDUCATION + KIDSDRIV_FLAG + INCOME_CAT +
##     HOMEOWNER, family = "binomial", data = train_df_trans)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1846  -0.7360  -0.4436   0.6989   3.2096
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -0.808918   0.372886  -2.169 0.030057 *
## AGE                  -0.008346   0.004261  -1.959 0.050154 .
## CLM_FREQ_LOG          0.248601   0.074047   3.357 0.000787 ***
## URBANICITY1           2.094943   0.130333  16.074  < 2e-16 ***
## MVR_PTS_LOG           0.061745   0.009238   6.684 2.33e-11 ***
## OLDCLAIM_LOG         -0.088956   0.035719  -2.490 0.012758 *
## PARENT11              0.310786   0.115917   2.681 0.007338 **
## REVOKED1              0.856583   0.099228   8.632  < 2e-16 ***
## CAR_USEPrivate       -0.787060   0.085768  -9.177  < 2e-16 ***
## CAR_TYPEPanel Truck   0.097836   0.154648   0.633 0.526972
## CAR_TYPEPickup        0.498138   0.114647   4.345 1.39e-05 ***
## CAR_TYPESports Car    1.012436   0.122911   8.237  < 2e-16 ***
## CAR_TYPESUV           0.755746   0.099107   7.626 2.43e-14 ***
## CAR_TYPEVan           0.473068   0.138070   3.426 0.000612 ***
## MSTATUS1             -0.511067   0.091593  -5.580 2.41e-08 ***
## EDUCATIONHigh School -0.207637   0.107779  -1.927 0.054040 .
## EDUCATIONBachelors   -0.746540   0.117738  -6.341 2.29e-10 ***
## EDUCATIONMasters     -0.846365   0.129320  -6.545 5.96e-11 ***
## EDUCATIONPhD         -1.149027   0.162915  -7.053 1.75e-12 ***
## KIDSDRIV_FLAG1        0.726144   0.101196   7.176 7.20e-13 ***
## INCOME_CAT.L          0.075360   0.075063   1.004 0.315402
## INCOME_CAT.Q          0.343078   0.086893   3.948 7.87e-05 ***
## INCOME_CAT.C          0.234976   0.086869   2.705 0.006831 **
## HOMEOWNER1           -0.230535   0.083544  -2.759 0.005790 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6640.0  on 5736  degrees of freedom
## Residual deviance: 5332.9  on 5713  degrees of freedom
## AIC: 5380.9
##
## Number of Fisher Scoring iterations: 5
```

The AIC of Model BLR:3 is 5380.9.

Feature	Coefficient	Percentage Change in Odds of Car Crash
URBANICITY1	8.1249789	712.5
CAR_TYPESports Car	2.7522987	175.2
REVOKED1	2.3551001	135.5
CAR_TYPESUV	2.1292001	112.9
KIDSDRIV_FLAG1	2.0670935	106.7
CAR_TYPEPickup	1.6456545	64.6
CAR_TYPEVan	1.6049102	60.5
INCOME_CAT.Q	1.4092787	40.9
PARENT11	1.3644972	36.4
CLM_FREQ_LOG	1.2822305	28.2
INCOME_CAT.C	1.2648788	26.5
CAR_TYPEPanel Truck	1.1027816	10.3
INCOME_CAT.L	1.0782723	7.8
MVR_PTS_LOG	1.0636907	6.4
AGE	0.9916890	-0.8
OLDCLAIM_LOG	0.9148860	-8.5
EDUCATIONHigh School	0.8125019	-18.7
HOMEOWNER1	0.7941084	-20.6
MSTATUS1	0.5998551	-40.0
EDUCATIONBachelors	0.4740035	-52.6
CAR_USEPrivate	0.4551809	-54.5
EDUCATIONMasters	0.4289714	-57.1
EDUCATIONPhD	0.3169449	-68.3

Interestingly, in Model BLR:3, education levels do reduce the odds of a car crash in the order expected. That is, having a PhD decreases the odds more than a Master's, having a Master's decreases the odds more than a Bachelor's, and having a Bachelor's decreases the odds more than having a High School Diploma. Otherwise, coefficients follow similar patterns to what we discussed with the first model. Private car use is one of the biggest car crash odds reducers; the biggest increaser of the odds of a car crash is living/working in an urban area; and driving anything other than a minivan, having had your license revoked, and having teenagers driving your car all big odds increasers as well. The `INCOME_CAT` factor has the opposite effect we were expecting. Perhaps the reason higher income categories are associated with higher car crash odds is incomes are usually higher in urban areas, and urban areas are very associated with higher car crash odds.

We check for possible multicollinearity within this model.

```
##                     GVIF Df GVIF^(1/(2*Df))
## AGE             1.192534  1        1.092032
## CLM_FREQ_LOG  68.540964  1        8.278947
## URBANICITY     1.102783  1        1.050135
## MVR_PTS_LOG    1.104418  1        1.050913
## OLDCLAIM_LOG  68.607176  1        8.282945
## PARENT1        1.583420  1        1.258340
## REVOKED        1.129558  1        1.062807
## CAR_USE        1.567173  1        1.251868
## CAR_TYPE       1.599229  5        1.048072
## MSTATUS        1.815128  1        1.347267
## EDUCATION      1.782365  4        1.074916
## KIDSDRIV_FLAG  1.106473  1        1.051890
## INCOME_CAT     1.533759  3        1.073889
## HOMEOWNER      1.437353  1        1.198897
```

`OLDCLAIM_LOG` and `CLM_FREQ_LOG` have variance inflation factors greater than five. Since we believe claim frequency has more to do with `TARGET_FLAG`, and past claim amounts have more to do with `TARGET_AMT`, we choose to remove `OLDCLAIM_LOG` from this model. We don't reprint a summary, but the new AIC is 5385.1, and none of the variables have variance inflation factors greater than five any longer.

Multiple Linear Regression Models

Model MLR:1 - Full Model Using Original, Untransformed Variables, with All Missing Values Imputed - Reduced via Stepwise Backward Model Selection We create Model MLR:1, our baseline multiple linear regression model based on all the original, untransformed variables, with all missing values imputed so that no observations or predictors have to be excluded from the model. Then we perform stepwise backward model selection.

A summary of Model MLR:1 is below:

```
##
## Call:
## lm(formula = TARGET_AMT ~ MSTATUS + SEX + BLUEBOOK + RED_CAR +
##     REVOKED + MVR_PTS, data = select(filter(alt_train_df_imputed,
##     TARGET_FLAG == 1), -TARGET_FLAG))
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8227  -3166  -1587    411 100814
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3861.32677  554.66624   6.962  5.0e-12 ***
## MSTATUS1    -598.72182  414.43071  -1.445   0.1488
## SEXFemale   1437.88719  570.49291   2.520   0.0118 *
## BLUEBOOK       0.11586    0.02558   4.529  6.4e-06 ***
## RED_CAR1    -919.61610  626.13920  -1.469   0.1421
## REVOKED1    -971.02786  511.74890  -1.897   0.0580 .
## MVR_PTS      127.62768   80.12720   1.593   0.1114
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8073 on 1516 degrees of freedom
## Multiple R-squared:  0.02348,    Adjusted R-squared:  0.01962
## F-statistic: 6.076 on 6 and 1516 DF,  p-value: 2.67e-06
```

Only a small number of variables remain, and they explain very little variance in our data. Some of them are not statistically significant by normal standards. We will leave them in and see whether they provide predictive power.

We check for multicollinearity within this model.

```
##  MSTATUS      SEX BLUEBOOK  RED_CAR  REVOKED  MVR_PTS
## 1.002852 1.873415 1.008312 1.872006 1.006370 1.004409
```

None of the variance inflation factors are greater than five, so there are no multicollinearity issues to address for this model.

Model MLR:2 - Full Model Using Original and Transformed Variables, with All Missing Values Imputed - Reduced via Stepwise Backward Model Selection We create Model MLR:2, a second multiple linear regression model using original and transformed variables (including the response variable), with all missing values imputed so that no observations or predictors have to be excluded from the model. Then we perform stepwise backward model selection.

A summary of Model MLR:2 is below:

```
##
## Call:
## lm(formula = TARGET_AMT_LOG ~ MSTATUS + SEX + BLUEBOOK_SQRT,
##     data = select(filter(alt_train_df_trans, TARGET_FLAG == 1),
##         -TARGET_FLAG))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0399 -0.4154  0.0412  0.4118  3.2477
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.9364878  0.0761413 104.234  < 2e-16 ***
## MSTATUS1      -0.0779400  0.0412953  -1.887   0.0593 .
## SEXFemale      0.0967186  0.0416973   2.320   0.0205 *
## BLUEBOOK_SQRT  0.0028464  0.0006045   4.709 2.72e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8054 on 1519 degrees of freedom
## Multiple R-squared:  0.02127,    Adjusted R-squared:  0.01934
## F-statistic: 11.01 on 3 and 1519 DF,  p-value: 3.777e-07
```

Again, only a small number of predictors remain, and they explain very little variance in our data.

We check for multicollinearity within this model.

```
##       MSTATUS           SEX BLUEBOOK_SQRT
##      1.000487      1.005604      1.005612
```

There are no variance inflation factors greater than five, so there are no issues of multicollinearity to address.

Model MLR:3 - Robust Model Using Select Original and Transformed Variables We create Model MLR:3, a robust model designed to deal with outliers using select original and transformed variables (including the response variable). The predictors were chosen from among those retained in the previous two models, as stepwise backward selection is not possible with a robust model, and the full robust model's residual standard error was higher than that of this reduced model.

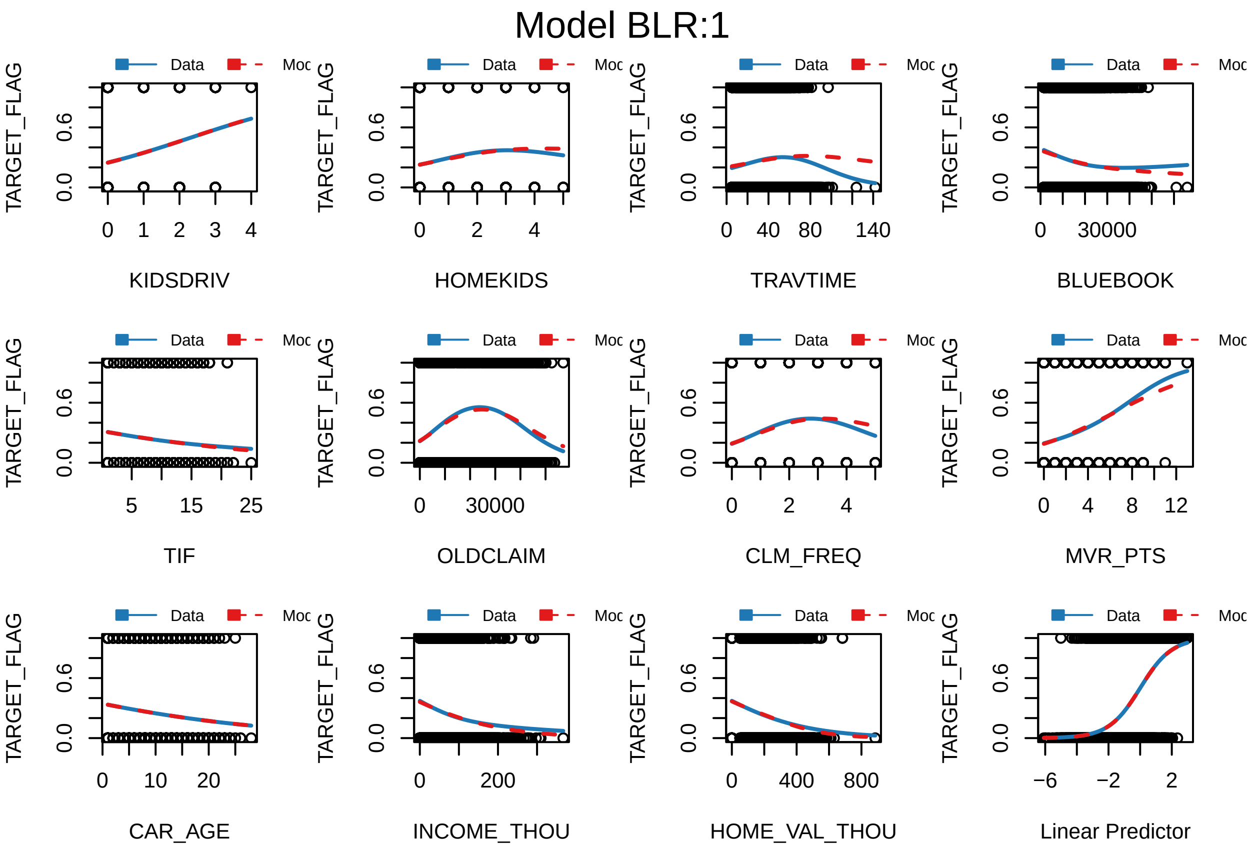
A summary of Model MLR:3 is below:

```
##
## Call: rlm(formula = TARGET_AMT_LOG ~ MSTATUS + SEX + BLUEBOOK_SQRT +
##     RED_CAR + REVOKED + MVR_PTS_LOG, data = select(filter(alt_train_df_trans,
##     TARGET_FLAG == 1), -TARGET_FLAG))
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.10955 -0.40365  0.03963  0.40579  3.26592
##
## Coefficients:
##               Value    Std. Error t value
## (Intercept)     8.0765   0.0643   125.6189
## MSTATUS1       -0.0546   0.0343    -1.5935
## SEXFemale       0.0939   0.0472     1.9900
## BLUEBOOK_SQRT   0.0017   0.0005     3.3032
## RED_CAR1       -0.0269   0.0517    -0.5196
## REVOKED1        0.0043   0.0423     0.1020
## MVR_PTS_LOG     0.0042   0.0045     0.9331
##
## Residual standard error: 0.6011 on 1516 degrees of freedom
```

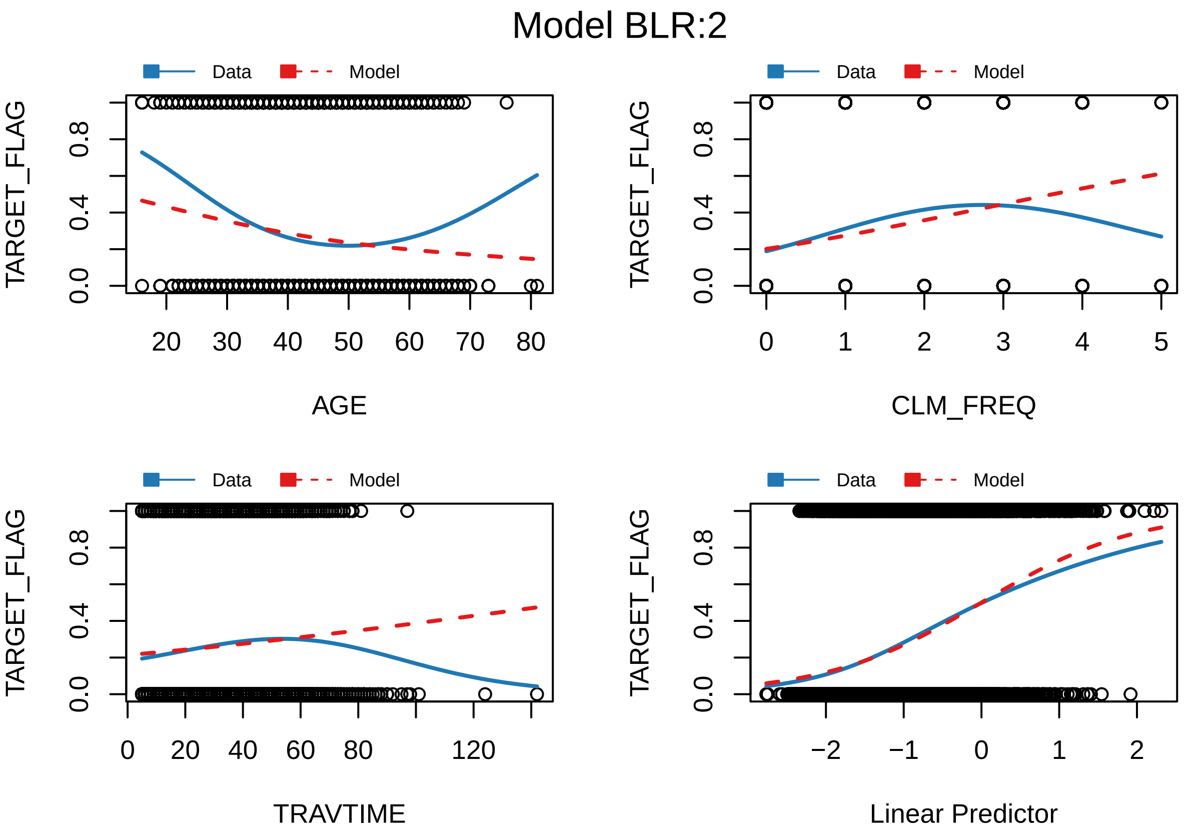
Select Models

Binary Logistic Regression Models To choose our binary logistic regression model, we consider that false positives would likely result in the company charging too high a premium for those customers, and false negatives would likely result in the company charging too low a premium for those customers. Therefore, the effects of those inaccurate predictions could be equally costly. False positive customers might jump to competitors offering them lower rates (perhaps because those competitors more accurately identified them as lower risk), and false negative customers might cost the company more in unanticipated claim costs. So we will rely primarily on the F1 Score, which incorporates both precision and recall to accurately classify positives while minimizing false positives and false negatives, to select the best model. However, we will look at other metrics and goodness of fit checks as well.

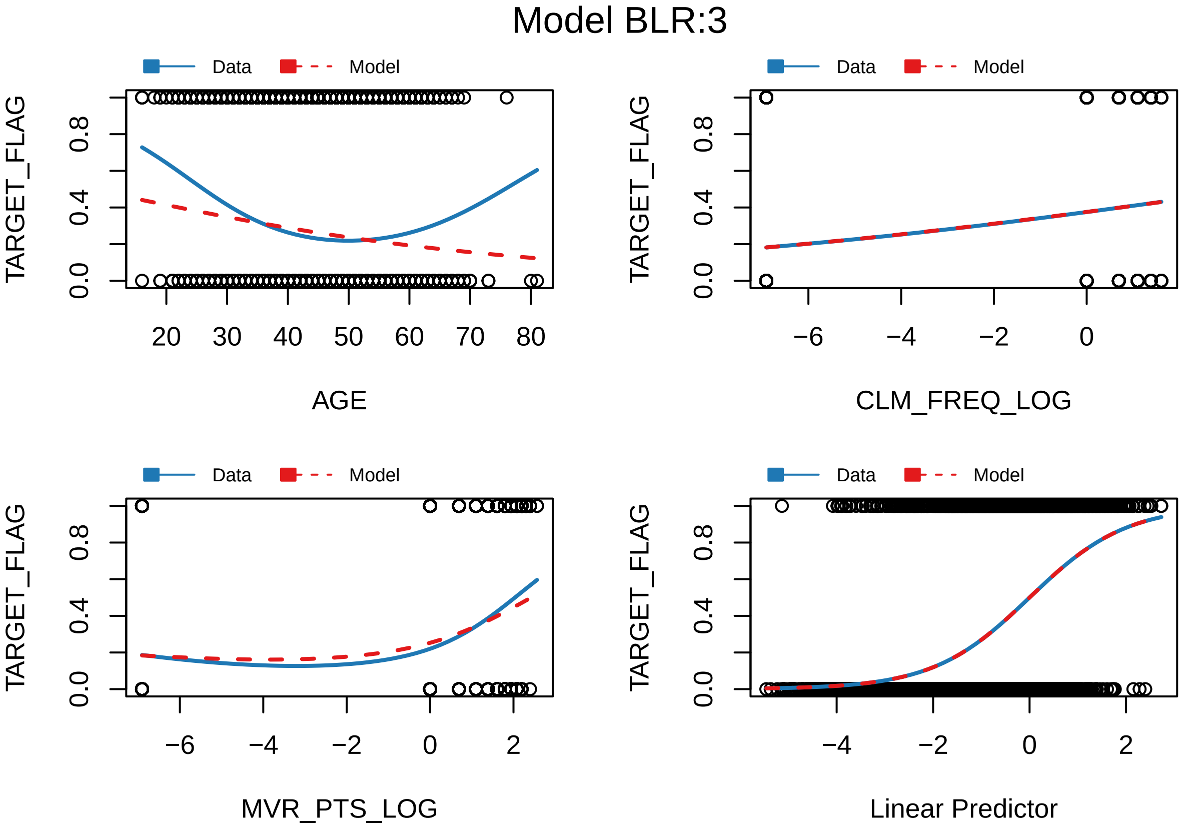
To first check for goodness of fit, we create marginal model plots for the response and each predictor in each binary logistic regression model. (Note that the `mmps` function from the `car` package used to generate these plots skips any factors and interaction terms within the models intentionally.)



The marginal models plots for Model BLR:1 reveal some small fit issues, mostly with `TRAVTIME`.



The marginal models plots for Model BLR:2 reveal more fit issues than Model BLR:1 had, but Model BLR:2 relies on fewer numeric variables than Model BLR:1, and remember these plots can't visualize factors.



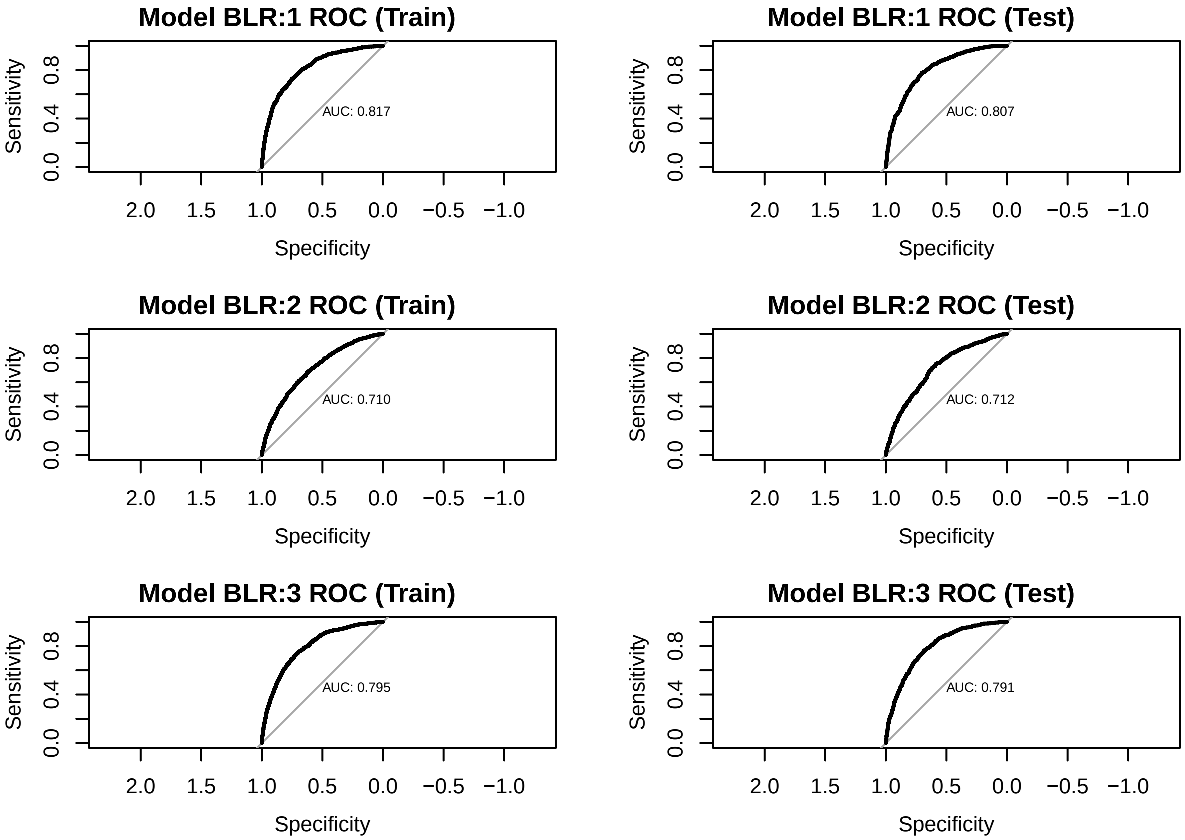
The marginal models plots for Model BLR:3 reveals one fit issue for `AGE`. This model also relies on a lot of factors, which the marginal models plots can't visualize.

We calculate the Hosmer-Lemeshow statistic for each model to further check for lack of fit.

Model	HL Statistic	DoF	P Value
Model BLR:1	17.512	8	0.0251979
Model BLR:2	11.1989	8	0.1906817
Model BLR:3	22.34053	8	0.004322592

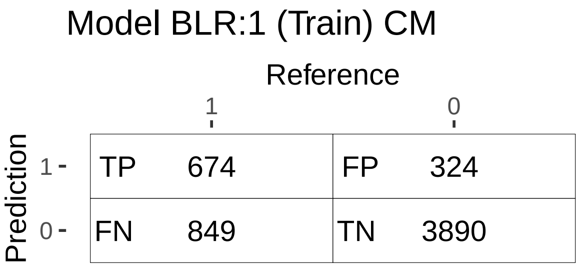
The low p-values for Models BLR:1 and BLR:3 suggest some lack of fit there. The moderate p-value for Model BLR:2 suggests no lack of fit there. This is not what we expected based on the incomplete pictures provided by looking at just the marginal models plots.

We produce ROC curves to visualize how each model performs on the training and test data.



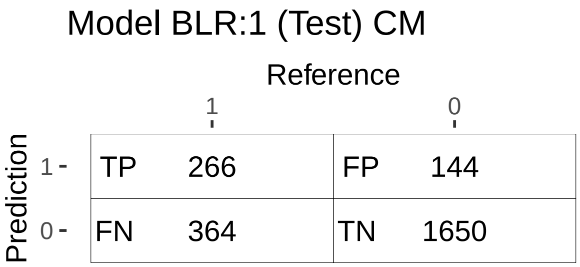
Model BLR:1 has the highest AUC on both the training and the test data, although Model BLR:3 is not far behind.

We produce confusion matrices for all three models based on the training and test data.



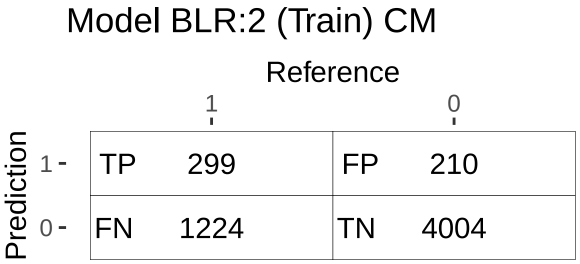
Model BLR:1 (Train) CM

	Reference	
Prediction	1	0
1	TP 674	FP 324
0	FN 849	TN 3890



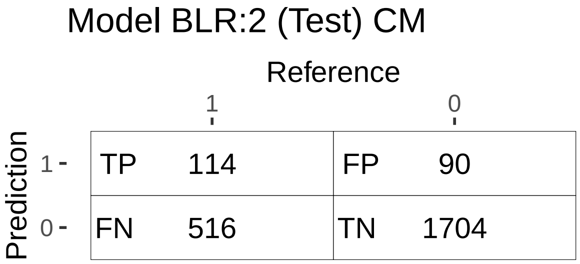
Model BLR:1 (Test) CM

	Reference	
Prediction	1	0
1	TP 266	FP 144
0	FN 364	TN 1650



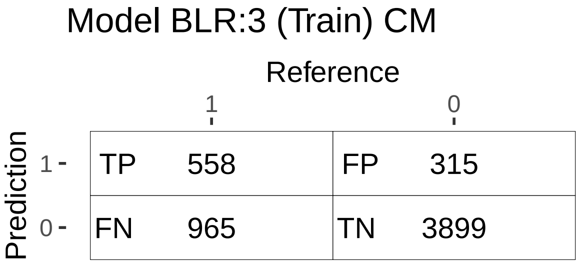
Model BLR:2 (Train) CM

	Reference	
Prediction	1	0
1	TP 299	FP 210
0	FN 1224	TN 4004



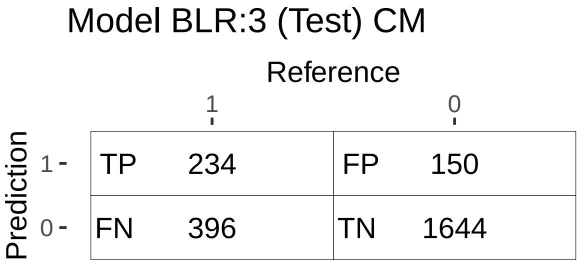
Model BLR:2 (Test) CM

	Reference	
Prediction	1	0
1	TP 114	FP 90
0	FN 516	TN 1704



Model BLR:3 (Train) CM

	Reference	
Prediction	1	0
1	TP 558	FP 315
0	FN 965	TN 3899



Model BLR:3 (Test) CM

	Reference	
Prediction	1	0
1	TP 234	FP 150
0	FN 396	TN 1644

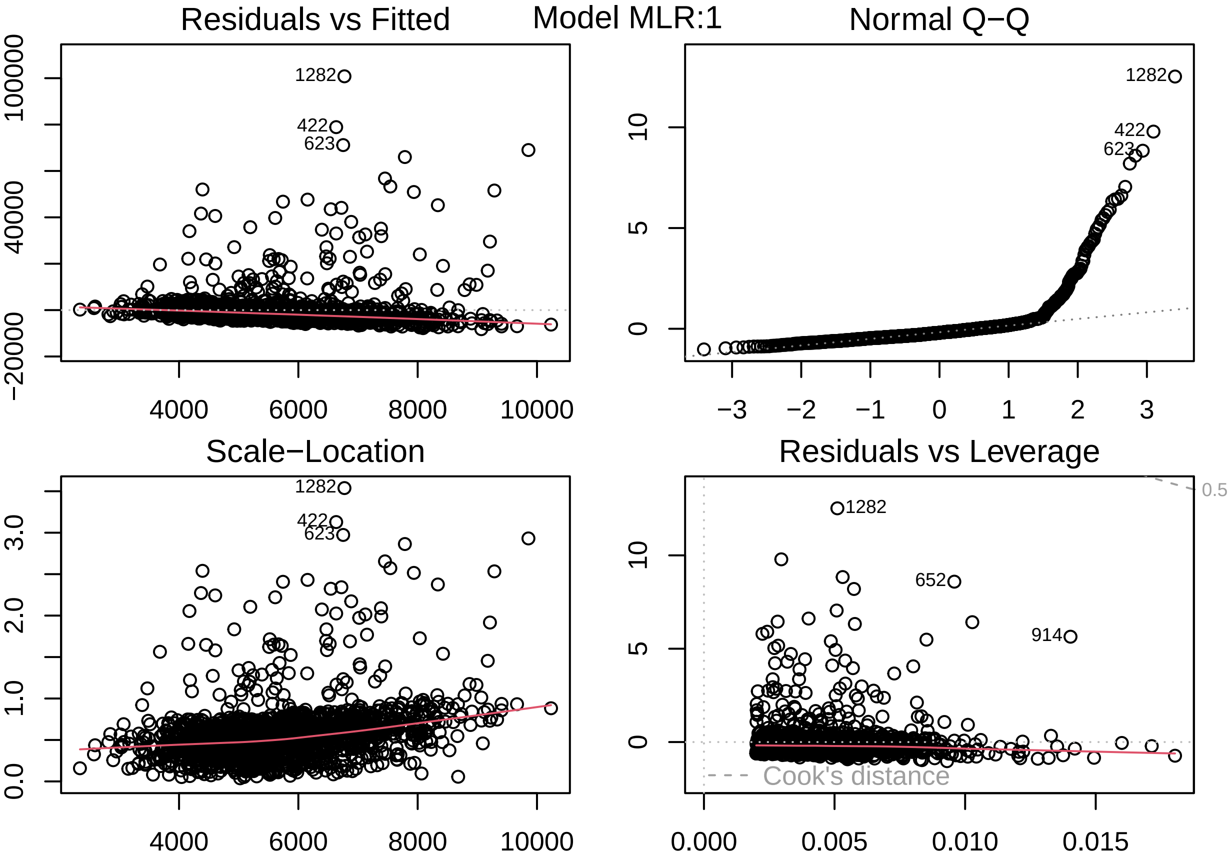
We calculate performance metrics for all models on the training and test data.

	1 (Train)	1 (Test)	2 (Train)	2 (Test)	3 (Train)	3 (Test)
Sensitivity	0.443	0.422	0.196	0.181	0.366	0.371
Specificity	0.923	0.920	0.950	0.950	0.925	0.916
Pos Pred Value	0.675	0.649	0.587	0.559	0.639	0.609
Neg Pred Value	0.821	0.819	0.766	0.768	0.802	0.806
Precision	0.675	0.649	0.587	0.559	0.639	0.609
Recall	0.443	0.422	0.196	0.181	0.366	0.371
F1	0.535	0.512	0.294	0.273	0.466	0.462
Prevalence	0.265	0.260	0.265	0.260	0.265	0.260
Detection Rate	0.117	0.110	0.052	0.047	0.097	0.097
Detection Prevalence	0.174	0.169	0.089	0.084	0.152	0.158
Balanced Accuracy	0.683	0.671	0.573	0.565	0.646	0.644
Accuracy	0.796	0.790	0.750	0.750	0.777	0.775
Classification Error Rate	0.204	0.210	0.250	0.250	0.223	0.225
AUC	0.817	0.807	0.710	0.712	0.795	0.791

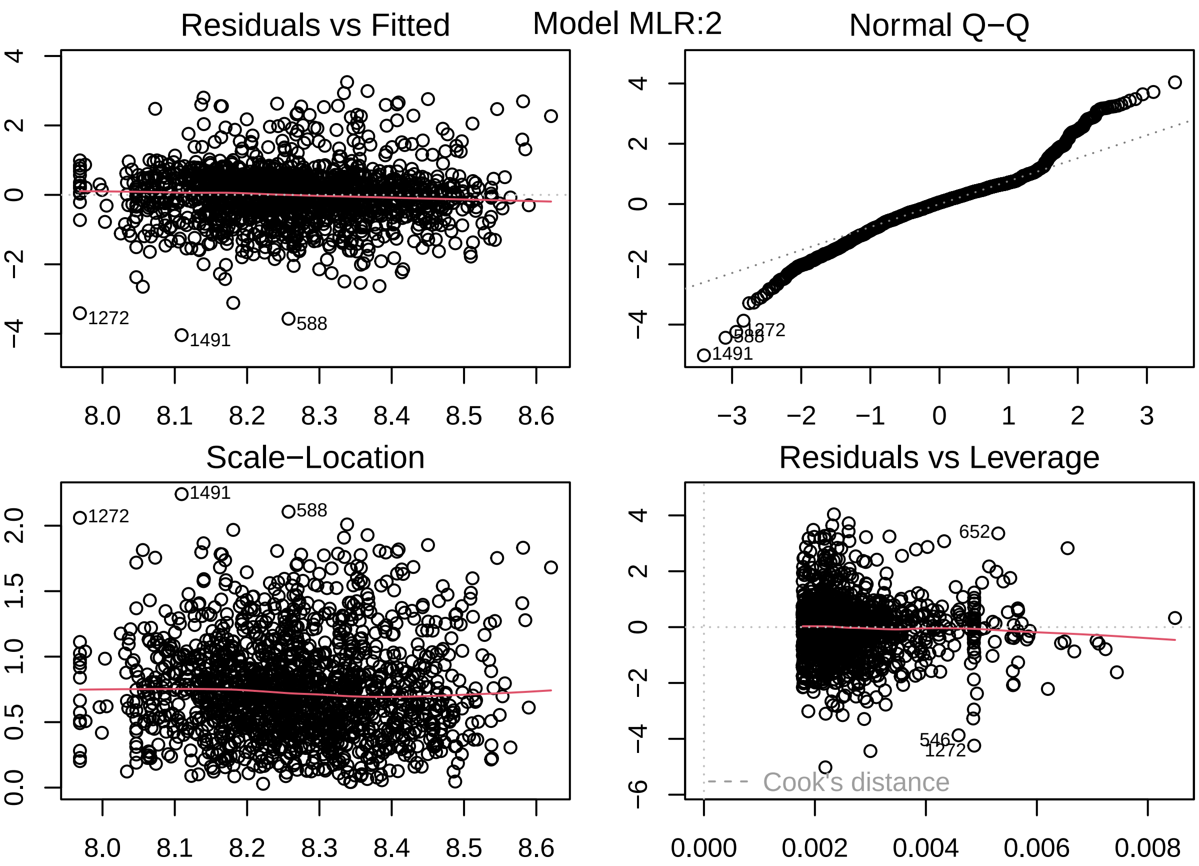
Model BLR:1 performs best based on most metrics, whether on the training data or on the test data. It had the lowest AIC, and its AUC was slightly higher than Model BLR:3, but most importantly, it balances Precision and Recall the best, and it therefore has the highest F1 Score. (Note that Model BLR:2's Recall is particularly low compared to the other models, while the Precision among them varies less.) Since the F1 Score is our primary metric, we select Model BLR:1 as the final binary logistic regression model we will use to make predictions on the evaluation data.

Multiple Linear Regression Models To select our multiple linear regression model, we will primarily rely on predictive R^2 and RMSE based on the test data, but we will also check for goodness of fit.

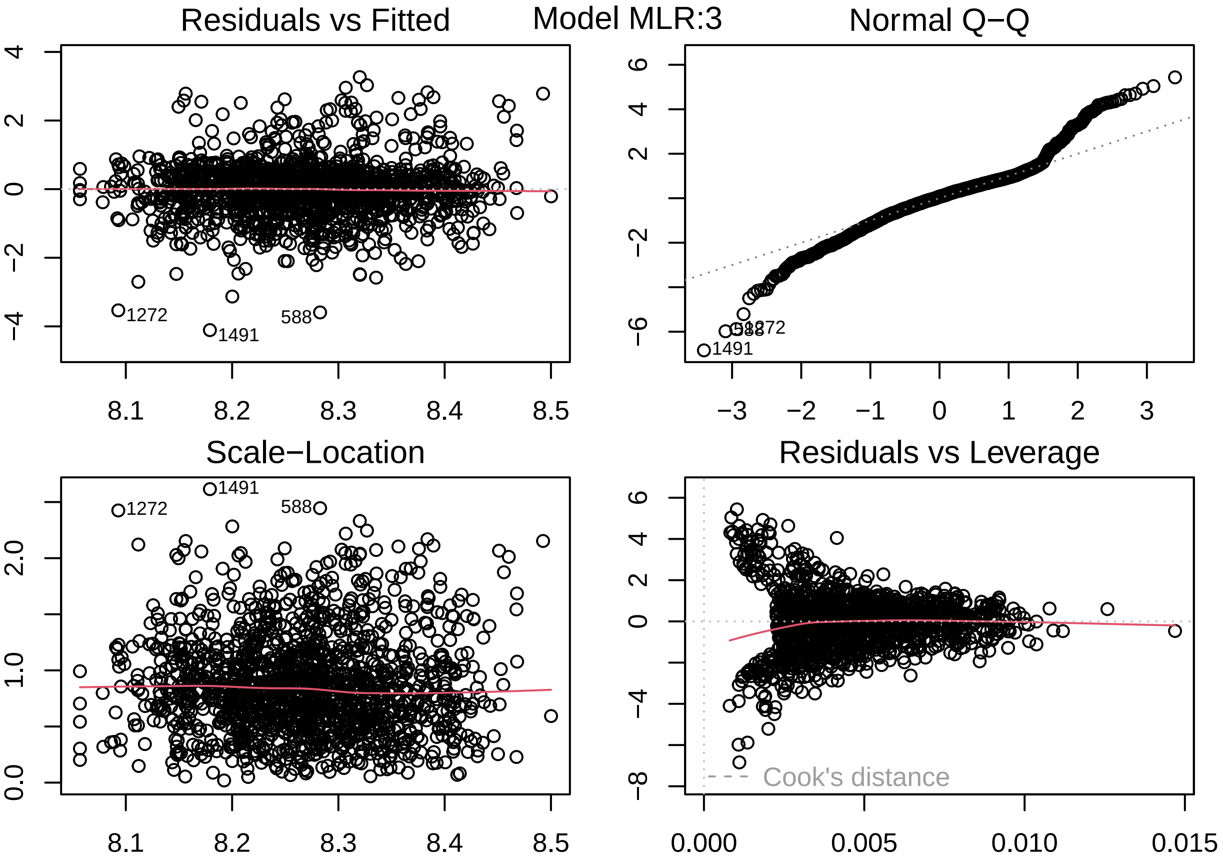
To check for goodness of fit, we primarily examine Residuals vs. Fitted Values and Q-Q Residuals plots for all three models.



Model MLR:1 does not fit the data very well. The residuals vs. fitted values are not randomly/evenly spaced above and below 0, and there is deviation from the normal line in the Q-Q plot on the right end. There are noted outliers.



Model MLR:2 fits the data better, if not well. The residuals vs. fitted values are more randomly/evenly spaced above and below 0. There is deviation from the normal line in the Q-Q plot on both ends though.



Model MLR:3 can't be said to fit the data better or worse than Model MLR:2.

We calculate predictive R^2 and RMSE using the test set for all three models. (In cases where the response variable was transformed, predictions are back-transformed.)

models	pred_rsq	rmse
Model MLR:1	0.00164553582810001	6643.62198126247
Model MLR:2	-0.0572331141972695	6836.72183171634
Model MLR:3	-0.0616884808018465	6851.11226184339

The predictive power of all these models is pretty low, as expected. Model MLR:1 is the only model that beats predicting using the mean, and it doesn't beat it by much. We know the assumptions of OLS are violated in Model MLR:1, even though the other models have flaws as well. So we do select it as the final multiple linear regression model we will use to make predictions on the evaluation data, but we approach these predictions with caution. Models that are more superior to the naive method of predicting using the mean, and that do not have statistical flaws, should be further investigated. No alternate models we have attempted have fared better as of yet though.

Predictions on Evaluation Data We make predictions on the evaluation dataset, and we save the file with the predicted probabilities, classifications, and costs as "HW4_Eval_PredProbs_Flags_Amounts.csv."

While we can't know whether the `TARGET_FLAG` classifications in particular are accurate, we summarize them below so we can compare the percentage of observations related to car crashes in the original data to the percentage in the evaluation data.

```
## # A tibble: 2 x 2
##   TARGET_FLAG   cnt
##   <fct>       <int>
## 1 0            1774
## 2 1             367
```

In the original data, about 26.4% of observations were related to car crashes, and in the evaluation data, we've identified only 17.1% of observations as being related to car crashes. Again, this does not speak to accuracy.

Appendix: Report Code

Below is the code for this report to generate the models and charts above.

```
knitr::opts_chunk$set(echo = FALSE)
library(tidyverse)
library(DataExplorer)
library(knitr)
library(cowplot)
library(finalfit)
library(correlationfunnel)
library(ggcorrplot)
library(RColorBrewer)
library(naniar)
library(mice)
library(MASS)
select <- dplyr::select
library(kableExtra)
library(car)
library(glmtoolbox)
library(pROC)
library(caret)
library(robustbase)

cur_theme <- theme_set(theme_classic())

my_url <- "https://raw.githubusercontent.com/waheeb123/Data-621/main/Homeworks/Homework%204/insurance_t
main_df <- read.csv(my_url, na.strings = "")

classes <- as.data.frame(unlist(lapply(main_df, class))) |>
    rownames_to_column()
cols <- c("Variable", "Class")
colnames(classes) <- cols
classes_summary <- classes |>
    group_by(Class) |>
    summarize(Count = n(),
              Variables = paste(sort(unique(Variable)),collapse=", "))
kable(classes_summary, "latex", booktabs = T) |>
  kableExtra::column_spec(2:3, width = "7cm")

vars <- c("INCOME", "HOME_VAL", "BLUEBOOK", "OLDCLAIM")
main_df <- main_df |>
    mutate(across(all_of(vars), ~gsub("\\$|,", "", .) |> as.integer()))
```

```r
main_df <- main_df |>
    select(-INDEX)
remove <- c("discrete_columns", "continuous_columns",
            "total_observations", "memory_usage")
completeness <- introduce(main_df) |>
    select(-all_of(remove))
knitr::kable(t(completeness), format = "simple")

p1 <- plot_missing(main_df, missing_only = TRUE,
                   ggtheme = theme_classic(), title = "Missing Values")

p1 <- p1 +
    scale_fill_brewer(palette = "Paired")
p1

exclude <- c("TARGET_AMT", "AGE", "INCOME", "YOJ", "HOME_VAL", "CAR_AGE", "JOB")
main_df_binarized <- main_df |>
    select(-all_of(exclude)) |>
    binarize(n_bins = 5, thresh_infreq = 0.01, name_infreq = "OTHER",
           one_hot = TRUE)
main_df_corr <- main_df_binarized |>
    correlate(TARGET_FLAG__1)
main_df_corr |>
    plot_correlation_funnel()

palette <- brewer.pal(n = 7, name = "RdBu")[c(1, 4, 7)]
excl <- c("TARGET_FLAG", "JOB", "CAR_TYPE", "CAR_USE", "EDUCATION",
             "MSTATUS", "PARENT1", "RED_CAR", "REVOKED", "SEX", "URBANICITY")
model.matrix(~0+., data = main_df |> filter(TARGET_FLAG == 1) |>select(-all_of(excl))) |>
    cor(use = "pairwise.complete.obs") |>
    ggcorrplot(show.diag = FALSE, type = "lower", lab = TRUE, lab_size = 2.5,
               tl.cex = 8, tl.srt = 90,
               colors = palette, outline.color = "white")

incl <- c("TARGET_AMT", "CAR_USE", "MSTATUS", "PARENT1", "RED_CAR",
             "REVOKED", "SEX", "URBANICITY")
model.matrix(~0+., data = main_df |> filter(TARGET_FLAG == 1) |> select(all_of(incl))) |>
    cor(use = "pairwise.complete.obs") |>
    ggcorrplot(show.diag = FALSE, type = "lower", lab = TRUE, lab_size = 3,
               tl.cex = 8, tl.srt = 90,
               colors = palette, outline.color = "white")

incl <- c("TARGET_AMT", "JOB", "CAR_TYPE", "EDUCATION")
model.matrix(~0+., data = main_df |> filter(TARGET_FLAG == 1) |> select(all_of(incl))) |>
    cor(use = "pairwise.complete.obs") |>
    ggcorrplot(show.diag = FALSE, type = "lower", lab = TRUE, lab_size = 1.75,
               tl.cex = 8, tl.srt = 90,
               colors = palette, outline.color = "white")

r <- model.matrix(~0+., data = main_df) |>
    cor(use = "pairwise.complete.obs")
is.na(r) <- abs(r) < 0.45
r |>
```

```
    ggcorrplot(show.diag = FALSE, type = "lower", lab = TRUE, lab_size = 2.5,
               tl.cex = 8, tl.srt = 90,
               colors = palette, outline.color = "white")

output <- split_columns(main_df, binary_as_factor = TRUE)
num <- data.frame(Variable = names(output$continuous),
                   Type = rep("Numeric", ncol(output$continuous)))
cat <- data.frame(Variable = names(output$discrete),
                   Type = rep("Categorical", ncol(output$discrete)))
ranges <- as.data.frame(t(sapply(main_df |> select(-names(output$discrete)),
                                 range, na.rm = TRUE)))
factors <- names(output$discrete)
main_df <- main_df |>
    mutate(across(all_of(factors), ~as.factor(.)))
values <- as.data.frame(t(sapply(main_df |> select(all_of(factors)),
                                 levels)))
values <- values |>
    mutate(across(all_of(factors), ~toString(unlist(.))))
values <- as.data.frame(t(values)) |>
    rownames_to_column()
cols <- c("Variable", "Values")
colnames(values) <- cols
remove <- c("V1", "V2")
ranges <- ranges |>
    rownames_to_column() |>
    group_by(rowname) |>
    mutate(Values = toString(c(V1, " - ", round(V2, 1))),
           Values = str_replace_all(Values, ",", "")) |>
    select(-all_of(remove))
colnames(ranges) <- cols
num <- num |>
    merge(ranges)
cat <- cat |>
    merge(values)
num_vs_cat <- num |>
    bind_rows(cat)
knitr::kable(num_vs_cat, "latex", booktabs = T)|>
  kableExtra::column_spec(2:3, width = "6cm")

alt_df <- main_df
main_df <- main_df |>
    mutate(TARGET_AMT = case_when(as.numeric(as.character(TARGET_FLAG)) < 1 ~ NA,
                               TRUE ~ TARGET_AMT),
           HOME_VAL = case_when(HOME_VAL < 1 ~ NA,
                               TRUE ~ HOME_VAL),
           INCOME = case_when(INCOME < 1 ~ NA,
                              TRUE ~ INCOME))

main_df <- main_df |>
    mutate(CAR_AGE = case_when(CAR_AGE < 0 ~ CAR_AGE * -1,
                               TRUE ~ CAR_AGE))
alt_df <- alt_df |>
    mutate(CAR_AGE = case_when(CAR_AGE < 0 ~ CAR_AGE * -1,
                               TRUE ~ CAR_AGE))

summary(main_df)

littles_test <- main_df |>
    mcar_test()
knitr::kable(littles_test, format = "simple")

x <- colnames(main_df)
dep = c("CAR_AGE")
exp = x[!x %in% dep]
missing_comp1 <- main_df |>
    missing_compare(explanatory = exp, dependent = dep) |>
    mutate(p = as.numeric(case_when(p == "<0.001" ~ "0.001",
                                    TRUE ~ p))) |>
    mutate(Dependant = dep)
colnames(missing_comp1) <- c("Explanatory", "Ref", "Not Missing", "Missing", "p",
                             "Dependant")
dep = c("YOJ")
exp = x[!x %in% dep]
missing_comp2 <- main_df |>
    missing_compare(explanatory = exp, dependent = dep) |>
    mutate(p = as.numeric(case_when(p == "<0.001" ~ "0.001",
                                    TRUE ~ p))) |>
    mutate(Dependant = dep)
colnames(missing_comp2) <- c("Explanatory", "Ref", "Not Missing", "Missing", "p",
                             "Dependant")
dep = c("INCOME")
exp = x[!x %in% dep]
missing_comp3 <- main_df |>
    missing_compare(explanatory = exp, dependent = dep) |>
    mutate(p = as.numeric(case_when(p == "<0.001" ~ "0.001",
                                    TRUE ~ p))) |>
    mutate(Dependant = dep)
colnames(missing_comp3) <- c("Explanatory", "Ref", "Not Missing", "Missing", "p",
                             "Dependant")
dep = c("HOME_VAL")
exp = x[!x %in% dep]
missing_comp4 <- main_df |>
    missing_compare(explanatory = exp, dependent = dep) |>
    mutate(p = as.numeric(case_when(p == "<0.001" ~ "0.001",
                                    TRUE ~ p))) |>
    mutate(Dependant = dep)
colnames(missing_comp4) <- c("Explanatory", "Ref", "Not Missing", "Missing", "p",
                             "Dependant")
dep = c("JOB")
exp = x[!x %in% dep]
missing_comp5 <- main_df |>
    missing_compare(explanatory = exp, dependent = dep) |>
    mutate(p = as.numeric(case_when(p == "<0.001" ~ "0.001",
                                    TRUE ~ p))) |>
    mutate(Dependant = dep)
colnames(missing_comp5) <- c("Explanatory", "Ref", "Not Missing", "Missing", "p",
```

```r
                             "Dependant")
missing_comp <- missing_comp1 |>
    bind_rows(missing_comp2, missing_comp3, missing_comp4, missing_comp5) |>
    mutate(Explanatory = case_when(is.na(p) ~ NA,
                                   TRUE ~ Explanatory)) |>
    fill(Explanatory, .direction = "down") |>
    group_by(Dependant, Explanatory) |>
    filter(any(p < 0.05)) |>
    select(Dependant, everything())
knitr::kable(missing_comp, format = "simple")

show <- c("YOJ", "INCOME", "HOME_VAL", "JOB")
p2 <- main_df |>
    select(all_of(show)) |>
    missing_plot()
p2

explanatory = c("JOB", "INCOME", "YOJ")
dependent = "HOME_VAL"
p3 <- main_df |>
    select(all_of(show)) |>
    missing_pattern(dependent, explanatory)

# just numeric variables
numeric_train <- main_df[,sapply(main_df, is.numeric)]
par(mfrow=c(4,4))
par(mai=c(.3,.3,.3,.3))
variables <- names(numeric_train)
for (i in 1:(length(variables))) {
  hist(numeric_train[[variables[i]]], main = variables[i], col = "lightblue")
}

cat_pivot <- main_df |>
    select(all_of(factors)) |>
    pivot_longer(cols = all_of(factors),
                 names_to = "Variable",
                 values_to = "Value") |>
    group_by(Variable, Value) |>
    summarize(Count = n()) |>
    group_by(Variable) |>
    mutate(Levels = n()) |>
    ungroup()
p4 <- cat_pivot |>
    filter(Levels == 2) |>
    ggplot(aes(x = Value, y = Count)) +
    geom_col(fill = "lightblue", color = "black") +
    facet_wrap(vars(Variable), ncol = 4, scales = "free_x") +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
p4

p5 <- cat_pivot |>
    filter(Levels > 2) |>
    ggplot(aes(x = Value, y = Count)) +
    geom_col(fill = "lightblue", color = "black") +
    coord_flip() +
    facet_wrap(vars(Variable), ncol = 1, scales = "free")
p5

# car type
x <- main_df$CAR_TYPE
main_df$CAR_TYPE <- case_match(x, "z_SUV" ~ "SUV", .default = x)
main_df$CAR_TYPE <- factor(main_df$CAR_TYPE,
                           levels = c("Minivan", "Panel Truck",
                                      "Pickup", "Sports Car", "SUV", "Van"))
x <- alt_df$CAR_TYPE
alt_df$CAR_TYPE <- case_match(x, "z_SUV" ~ "SUV", .default = x)
alt_df$CAR_TYPE <- factor(alt_df$CAR_TYPE,
                           levels = c("Minivan", "Panel Truck",
                                      "Pickup", "Sports Car", "SUV", "Van"))

# education
x <- main_df$EDUCATION
main_df$EDUCATION <- case_match(x, "z_High School" ~ "High School", .default = x)
main_df$EDUCATION <- factor(main_df$EDUCATION,
                             levels = c("<High School", "High School",
                                        "Bachelors", "Masters", "PhD"))
x <- alt_df$EDUCATION
alt_df$EDUCATION <- case_match(x, "z_High School" ~ "High School", .default = x)
alt_df$EDUCATION <- factor(alt_df$EDUCATION,
                             levels = c("<High School", "High School",
                                        "Bachelors", "Masters", "PhD"))

# job
x <- main_df$JOB
main_df$JOB <- case_match(x, "z_Blue Collar" ~ "Blue Collar", .default = x)
main_df$JOB <- factor(main_df$JOB, levels = c("Blue Collar", "Clerical",
                                              "Doctor", "Home Maker","Lawyer",
                                              "Manager", "Professional", "Student"))
x <- alt_df$JOB
alt_df$JOB <- case_match(x, "z_Blue Collar" ~ "Blue Collar", .default = x)
alt_df$JOB <- factor(alt_df$JOB, levels = c("Blue Collar", "Clerical",
                                              "Doctor", "Home Maker","Lawyer",
                                              "Manager", "Professional", "Student"))

# single parent
main_df <- main_df |>
  mutate(PARENT1 = as.factor(ifelse(PARENT1 == "Yes", 1, 0)))
alt_df <- alt_df |>
  mutate(PARENT1 = as.factor(ifelse(PARENT1 == "Yes", 1, 0)))

# marital status
x <- main_df$MSTATUS
main_df$MSTATUS <- case_match(x, "z_No" ~ "No", .default = x)
main_df <- main_df |>
  mutate(MSTATUS = as.factor(ifelse(MSTATUS == "Yes", 1, 0)))
x <- alt_df$MSTATUS
```

```r
alt_df$MSTATUS <- case_match(x, "z_No" ~ "No", .default = x)
alt_df <- alt_df |>
  mutate(MSTATUS = as.factor(ifelse(MSTATUS == "Yes", 1, 0)))

# red car
x <- main_df$RED_CAR
main_df$RED_CAR <- case_match(x, "no" ~ "No", "yes" ~ "Yes", .default = x)
main_df <- main_df |>
  mutate(RED_CAR = as.factor(ifelse(RED_CAR == "Yes", 1, 0)))
x <- alt_df$RED_CAR
alt_df$RED_CAR <- case_match(x, "no" ~ "No", "yes" ~ "Yes", .default = x)
alt_df <- alt_df |>
  mutate(RED_CAR = as.factor(ifelse(RED_CAR == "Yes", 1, 0)))

# revoked
main_df <- main_df |>
  mutate(REVOKED = as.factor(ifelse(REVOKED == "Yes", 1, 0)))
alt_df <- alt_df |>
  mutate(REVOKED = as.factor(ifelse(REVOKED == "Yes", 1, 0)))

# sex
x <- main_df$SEX
main_df$SEX <- case_match(x, "M" ~ "Male", "z_F" ~ "Female", .default = x)
main_df$SEX <- factor(main_df$SEX, levels = c("Male", "Female"))
x <- alt_df$SEX
alt_df$SEX <- case_match(x, "M" ~ "Male", "z_F" ~ "Female", .default = x)
alt_df$SEX <- factor(alt_df$SEX, levels = c("Male", "Female"))

# urban city - 1 if urban, 0 if rural
x <- main_df$URBANICITY
main_df$URBANICITY <- case_match(x, "Highly Urban/ Urban" ~ "Urban",
                                 "z_Highly Rural/ Rural" ~ "Rural", .default = x)
main_df <- main_df |>
  mutate(URBANICITY = as.factor(ifelse(URBANICITY == "Urban", 1, 0)))
x <- alt_df$URBANICITY
alt_df$URBANICITY <- case_match(x, "Highly Urban/ Urban" ~ "Urban",
                                 "z_Highly Rural/ Rural" ~ "Rural", .default = x)
alt_df <- alt_df |>
  mutate(URBANICITY = as.factor(ifelse(URBANICITY == "Urban", 1, 0)))

vars <- c("CAR_TYPE", "EDUCATION", "JOB", "PARENT1", "MSTATUS", "RED_CAR",
          "REVOKED", "SEX", "URBANICITY")

levs <- c("Minivan, Panel Truck, Pickup, Sports Car, SUV, Van",
          "<High School, High School, Bachelors, Masters, PhD",
          "Blue Collar, Clerical, Doctor, Home Maker, Lawyer, Manager, Professional, Student",
          "0, 1",
          "0, 1",
          "0, 1",
          "0, 1",
          "Male, Female",
          "0, 1")
```

```r
vars_levs <- as.data.frame(cbind(vars, levs))
colnames(vars_levs) <- c("Factor", "New Levels")
knitr::kable(vars_levs, format = "simple")

drop <- c("INCOME", "HOME_VAL")
main_df <- main_df |>
    mutate(INCOME_THOU = INCOME / 1000,
           HOME_VAL_THOU = HOME_VAL / 1000) |>
    select(-all_of(drop))
alt_df <- alt_df |>
    mutate(INCOME_THOU = INCOME / 1000,
           HOME_VAL_THOU = HOME_VAL / 1000) |>
    select(-all_of(drop))

main_df <- main_df |>
    mutate(YOJ = case_when(JOB == "Student" ~ NA,
                           TRUE ~ YOJ))
alt_df <- alt_df |>
    mutate(YOJ = case_when(JOB == "Student" ~ 0,
                           TRUE ~ YOJ))

exclude1 <- c("Student", "Homemaker")
exclude2 <- c(exclude1, "Blue Collar")
main_df <- main_df |>
    mutate(HOME_VAL_CAT = factor(case_when(HOME_VAL_THOU < 251 ~ "<=250K",
                                        HOME_VAL_THOU < 501 ~ "251-500K",
                                        HOME_VAL_THOU < 751 ~ "501-750K",
                                        TRUE ~ "751K+"),
                              ordered = TRUE,
                              levels = c("<=250K", "251-500K", "501-750K", "751K+"),
                              exclude = NULL),
           HOMEOWNER = as.factor(ifelse(is.na(HOME_VAL_THOU), 0, 1)),
           INCOME_CAT = factor(case_when(INCOME_THOU < 51 ~ "<=50K",
                                          INCOME_THOU < 101 ~ "51-100K",
                                          INCOME_THOU < 151 ~ "101-150K",
                                          TRUE ~ "151K+"),
                                ordered = TRUE,
                                levels = c("<=50K", "51-100K", "101-150K", "151K+"),
                                exclude = NULL),
           INCOME_FLAG = as.factor(ifelse(is.na(INCOME_THOU), 0, 1)),
           KIDSDRIV_FLAG = as.factor(case_when(KIDSDRIV > 0 ~ 1,
                                               TRUE ~ 0)),
           HOMEKIDS_FLAG = as.factor(case_when(HOMEKIDS > 0 ~ 1,
                                               TRUE ~ 0)),
           EMPLOYED = as.factor(ifelse(JOB %in% exclude1 | is.na(JOB),
                                       0, 1)),
           CAR_AGE_CAT = factor(case_when(CAR_AGE < 5 ~ "<=4",
                                           CAR_AGE < 9 ~ "5-8",
                                           CAR_AGE < 13 ~ "9-12",
                                           TRUE ~ "13+"),
                                 ordered = TRUE,
                                 levels = c("<=4", "5-8", "9-12", "13+"),
                                 exclude = NULL),
           WHITE_COLLAR = as.factor(ifelse(JOB %in% exclude2 | is.na(JOB),
                                           0, 1)))
main_df$JOB <- factor(main_df$JOB, exclude = NULL)

set.seed(202)
rows <- sample(nrow(main_df))
main_df <- main_df[rows, ]
alt_df <- alt_df[rows, ]
sample <- sample(c(TRUE, FALSE), nrow(main_df), replace=TRUE,
                 prob=c(0.7,0.3))
train_df <- main_df[sample, ]
test_df <- main_df[!sample, ]
alt_train_df <- alt_df[sample, ]
alt_test_df <- alt_df[!sample, ]

train_df_imputed <- train_df |>
    mutate(AGE = case_when(is.na(AGE) ~ mean(AGE, na.rm = TRUE),
                           TRUE ~ AGE))
test_df_imputed <- test_df |>
    mutate(AGE = case_when(is.na(AGE) ~ mean(AGE, na.rm = TRUE),
                           TRUE ~ AGE))

missing <- c("AGE")
imp_train_num <- train_df_imputed |>
    select(all_of(missing)) |>
    mutate(Set = "Train")
imp_test_num <- test_df_imputed |>
    select(all_of(missing)) |>
    mutate(Set = "Test")
imp_num <- imp_train_num |>
    bind_rows(imp_test_num)
imp_num_pivot <- imp_num |>
    pivot_longer(!Set, names_to = "Variable", values_to = "Value")
p6 <- imp_num_pivot |>
    ggplot(aes(x = Value)) +
    geom_density(fill = "lightblue", color = "black") +
    labs(y = "Density") +
    facet_grid(rows = vars(Set), cols = vars(Variable),
               switch = "y", scales = "free_x")
p6

col_classes <- unlist(lapply(alt_train_df, class))
missing <- c("AGE", "INCOME_THOU", "YOJ", "HOME_VAL_THOU", "CAR_AGE", "JOB")
x <- names(col_classes)
not_missing <- x[!x %in% missing]
#Since the imputation process is a little slow, we only do the imputations once, save the results as .c
if (file.exists("alt_train_df_imputed.csv") & file.exists("alt_test_df_imputed.csv")){
    alt_train_df_imputed <- read.csv("alt_train_df_imputed.csv", na.strings = "",
                                     colClasses = col_classes)
    alt_test_df_imputed <- read.csv("alt_test_df_imputed.csv", na.strings = "",
                                    colClasses = col_classes)
}else{
    #Start with alt_train_df
```

```r
    init = mice(alt_train_df, maxit=0)
    meth = init$method
    predM = init$predictorMatrix

    #Skip variables without missing data
    meth[not_missing] = ""

    #Set different imputation methods for each of the variables with missing data
    meth[c("AGE")] = "pmm" #Predictive mean matching
    meth[c("INCOME_THOU")] = "pmm"
    meth[c("YOJ")] = "pmm"
    meth[c("HOME_VAL_THOU")] = "pmm"
    meth[c("CAR_AGE")] = "pmm"
    meth[c("JOB")] = "polyreg" #Polytomous (multinomial) logistic regression

    #Impute
    imputed = mice(alt_train_df, method=meth, predictorMatrix=predM, m=5,
                   printFlag = FALSE)
    alt_train_df_imputed <- complete(imputed)
    write.csv(alt_train_df_imputed, "alt_train_df_imputed.csv", row.names = FALSE,
              fileEncoding = "UTF-8")

    #Repeat for alt_test_df
    init = mice(alt_test_df, maxit=0)
    meth = init$method
    predM = init$predictorMatrix
    meth[not_missing] = ""
    meth[c("AGE")] = "pmm"
    meth[c("INCOME_THOU")] = "pmm"
    meth[c("YOJ")] = "pmm"
    meth[c("HOME_VAL_THOU")] = "pmm"
    meth[c("CAR_AGE")] = "pmm"
    meth[c("JOB")] = "polyreg"
    imputed = mice(alt_test_df, method=meth, predictorMatrix=predM, m=5,
                   printFlag = FALSE)
    alt_test_df_imputed <- complete(imputed)
    write.csv(alt_test_df_imputed, "alt_test_df_imputed.csv", row.names = FALSE,
              fileEncoding = "UTF-8")
}

#Make sure the levels stay the same
levels(alt_train_df_imputed$CAR_TYPE) <- levels(main_df$CAR_TYPE)
levels(alt_train_df_imputed$EDUCATION) <- levels(main_df$EDUCATION)
levels(alt_train_df_imputed$JOB) <- levels(main_df$JOB)
levels(alt_train_df_imputed$SEX) <- levels(main_df$SEX)
levels(alt_test_df_imputed$CAR_TYPE) <- levels(main_df$CAR_TYPE)
levels(alt_test_df_imputed$EDUCATION) <- levels(main_df$EDUCATION)
levels(alt_test_df_imputed$JOB) <- levels(main_df$JOB)
levels(alt_test_df_imputed$SEX) <- levels(main_df$SEX)

x <- sapply(alt_train_df_imputed, function(x) sum(is.na(x)))
y <- sapply(alt_test_df_imputed, function(x) sum(is.na(x)))
sum(x, y) == 0
```

```
missing_num <- c("AGE", "INCOME_THOU", "YOJ", "HOME_VAL_THOU", "CAR_AGE")
imp_alt_train_num <- alt_train_df_imputed |>
    select(all_of(missing_num)) |>
    mutate(Set = "Train")
imp_alt_test_num <- alt_test_df_imputed |>
    select(all_of(missing_num)) |>
    mutate(Set = "Test")
imp_alt_num <- imp_alt_train_num |>
    bind_rows(imp_alt_test_num)
imp_alt_num_pivot <- imp_alt_num |>
    pivot_longer(!Set, names_to = "Variable", values_to = "Value")
p7 <- imp_alt_num_pivot |>
    ggplot(aes(x = Value)) +
    geom_density(fill = "lightblue", color = "black") +
    labs(y = "Density") +
    facet_grid(rows = vars(Set), cols = vars(Variable),
               switch = "y", scales = "free_x")
p7

missing_cat <- c("JOB")
imp_alt_train_cat <- alt_train_df_imputed |>
    select(all_of(missing_cat)) |>
    pivot_longer(cols = all_of(missing_cat),
                 names_to = "Variable",
                 values_to = "Value") |>
    group_by(Variable, Value) |>
    summarize(Count = n()) |>
    mutate(Set = "Train")
imp_alt_test_cat <- alt_test_df_imputed |>
    select(all_of(missing_cat)) |>
    pivot_longer(cols = all_of(missing_cat),
                 names_to = "Variable",
                 values_to = "Value") |>
    group_by(Variable, Value) |>
    summarize(Count = n()) |>
    mutate(Set = "Test")
imp_alt_pivot_cat <- imp_alt_train_cat |>
    bind_rows(imp_alt_test_cat)
p8 <- imp_alt_pivot_cat |>
    ggplot(aes(x = Value, y = Count)) +
    geom_col(fill = "lightblue", color = "black") +
    labs(x = "Job") +
    coord_flip() +
    facet_wrap(vars(Set), ncol = 2)
p8

skewed <- c("TRAVTIME", "BLUEBOOK", "TIF", "OLDCLAIM", "CLM_FREQ", "MVR_PTS")
train_df_trans <- train_df_imputed
for (i in 1:(length(skewed))){
    #Add a small constant to columns with any 0 values
    if (sum(train_df_trans[[skewed[i]]] == 0) > 0){
        train_df_trans[[skewed[i]]] <-
            train_df_trans[[skewed[i]]] + 0.001
    }
}
for (i in 1:(length(skewed))){
    if (i == 1){
        lambdas <- c()
    }
    bc <- boxcox(lm(train_df_trans[[skewed[i]]] ~ 1),
                 lambda = seq(-2, 2, length.out = 81),
                 plotit = FALSE)
    lambda <- bc$x[which.max(bc$y)]
    lambdas <- append(lambdas, lambda)
}
lambdas <- as.data.frame(cbind(skewed, lambdas))
adj <- c("no transformation", "square root", "log", "log", "log", "log")
lambdas <- cbind(lambdas, adj)
cols <- c("Skewed Variable", "Ideal Lambda Proposed by Box-Cox", "Reasonable Alternative Transformation
colnames(lambdas) <- cols
knitr::kable(lambdas, format = "simple")

remove <- c("BLUEBOOK", "TIF", "OLDCLAIM", "CLM_FREQ", "MVR_PTS")
train_df_trans <- train_df_trans |>
    mutate(BLUEBOOK_SQRT = BLUEBOOK^0.5,
           TIF_LOG = log(TIF),
           OLDCLAIM_LOG = log(OLDCLAIM),
           CLM_FREQ_LOG = log(CLM_FREQ),
           MVR_PTS_LOG = log(MVR_PTS)) |>
    select(-all_of(remove))
test_df_trans <- test_df_imputed
for (i in 1:(length(skewed))){
    #Add a small constant to columns with any 0 values
    if (sum(test_df_trans[[skewed[i]]] == 0) > 0){
        test_df_trans[[skewed[i]]] <-
            test_df_trans[[skewed[i]]] + 0.001
    }
}
test_df_trans <- test_df_trans |>
    mutate(BLUEBOOK_SQRT = BLUEBOOK^0.5,
           TIF_LOG = log(TIF),
           OLDCLAIM_LOG = log(OLDCLAIM),
           CLM_FREQ_LOG = log(CLM_FREQ),
           MVR_PTS_LOG = log(MVR_PTS)) |>
    select(-all_of(remove))

transformed <- c("BLUEBOOK_SQRT", "TIF_LOG", "OLDCLAIM_LOG", "CLM_FREQ_LOG",
                 "MVR_PTS_LOG")
train_df_trans_set <- train_df_trans |>
    select(all_of(transformed)) |>
    mutate(Set = "Train")
test_df_trans_set <- test_df_trans |>
    select(all_of(transformed)) |>
    mutate(Set = "Test")
trans_sets <- train_df_trans_set |>
    bind_rows(test_df_trans_set)
```

```r
trans_sets_pivot <- trans_sets |>
    pivot_longer(!Set, names_to = "Variable", values_to = "Value")
p9 <- trans_sets_pivot |>
    ggplot(aes(x = Value)) +
    geom_density(fill = "lightblue", color = "black") +
    labs(y = "Density") +
    facet_grid(rows = vars(Set), cols = vars(Variable),
               switch = "y", scales = "free_x")
p9

skewed <- c("TARGET_AMT", "YOJ", "TRAVTIME", "KIDSDRIV", "HOMEKIDS", "BLUEBOOK",
            "TIF", "OLDCLAIM", "CLM_FREQ", "MVR_PTS", "INCOME_THOU",
            "HOME_VAL_THOU", "CAR_AGE")
alt_train_df_trans <- alt_train_df_imputed
for (i in 1:(length(skewed))){
    #Add a small constant to columns with any 0 values
    if (sum(alt_train_df_trans[[skewed[i]]] == 0) > 0){
        alt_train_df_trans[[skewed[i]]] <-
            alt_train_df_trans[[skewed[i]]] + 0.001
    }
}
for (i in 1:(length(skewed))){
    if (i == 1){
        lambdas <- c()
    }
    bc <- boxcox(lm(alt_train_df_trans[[skewed[i]]] ~ 1),
                 lambda = seq(-2, 2, length.out = 81),
                 plotit = FALSE)
    lambda <- bc$x[which.max(bc$y)]
    lambdas <- append(lambdas, lambda)
}
lambdas <- as.data.frame(cbind(skewed, lambdas))
adj <- c("log", "no transformation", "no transformation", "inverse", "log",
         "square root", "log", "log", "log", "log", "square root", "log",
         "square root")
lambdas <- cbind(lambdas, adj)
cols <- c("Skewed Variable", "Ideal Lambda Proposed by Box-Cox", "Reasonable Alternative Transformation
colnames(lambdas) <- cols
knitr::kable(lambdas, format = "simple")

remove <- c("TARGET_AMT", "KIDSDRIV", "HOMEKIDS", "BLUEBOOK",
            "TIF", "OLDCLAIM", "CLM_FREQ", "MVR_PTS", "INCOME_THOU",
            "HOME_VAL_THOU", "CAR_AGE")
alt_train_df_trans <- alt_train_df_trans |>
    mutate(TARGET_AMT_LOG = log(TARGET_AMT),
           KIDSDRIV_INV = KIDSDRIV^-1,
           HOMEKIDS_LOG = log(HOMEKIDS),
           BLUEBOOK_SQRT = BLUEBOOK^0.5,
           TIF_LOG = log(TIF),
           OLDCLAIM_LOG = log(OLDCLAIM),
           CLM_FREQ_LOG = log(CLM_FREQ),
           MVR_PTS_LOG = log(MVR_PTS),
           INCOME_THOU_SQRT = INCOME_THOU^0.5,
           HOME_VAL_THOU_LOG = log(HOME_VAL_THOU),
           CAR_AGE_SQRT = CAR_AGE^0.5) |>
    select(-all_of(remove))
alt_test_df_trans <- alt_test_df_imputed
for (i in 1:(length(skewed))){
    #Add a small constant to columns with any 0 values
    if (sum(alt_test_df_trans[[skewed[i]]] == 0) > 0){
        alt_test_df_trans[[skewed[i]]] <-
            alt_test_df_trans[[skewed[i]]] + 0.001
    }
}
alt_test_df_trans <- alt_test_df_trans |>
    mutate(TARGET_AMT_LOG = log(TARGET_AMT),
           KIDSDRIV_INV = KIDSDRIV^-1,
           HOMEKIDS_LOG = log(HOMEKIDS),
           BLUEBOOK_SQRT = BLUEBOOK^0.5,
           TIF_LOG = log(TIF),
           OLDCLAIM_LOG = log(OLDCLAIM),
           CLM_FREQ_LOG = log(CLM_FREQ),
           MVR_PTS_LOG = log(MVR_PTS),
           INCOME_THOU_SQRT = INCOME_THOU^0.5,
           HOME_VAL_THOU_LOG = log(HOME_VAL_THOU),
           CAR_AGE_SQRT = CAR_AGE^0.5) |>
    select(-all_of(remove))

model_blr_1 <- glm(TARGET_FLAG ~ . - TARGET_AMT, family = 'binomial',
                   data = alt_train_df_imputed)
model_blr_1 <- stepAIC(model_blr_1, trace = 0)
summary(model_blr_1)

beta <- coef(model_blr_1)
beta_exp <- as.data.frame(exp(beta)) |>
    rownames_to_column()
cols <- c("Feature", "Coefficient")
colnames(beta_exp) <- cols
beta_exp <- beta_exp |>
    filter(Feature != "(Intercept)")
beta_exp <- beta_exp |>
    mutate(diff = round(Coefficient - 1, 3) * 100) |>
    arrange(desc(diff))
cols <- c("Feature", "Coefficient", "Percentage Change in Odds of Car Crash")
colnames(beta_exp) <- cols
knitr::kable(beta_exp, format = "simple")

vif(model_blr_1)

model_blr_2 <- glm(TARGET_FLAG ~ AGE + CLM_FREQ + HOMEOWNER + INCOME_FLAG + EMPLOYED + WHITE_COLLAR + MS
                   data=train_df_imputed, family='binomial')
model_blr_2 <- stepAIC(model_blr_2, trace=0)
summary(model_blr_2)

beta <- coef(model_blr_2)
beta_exp <- as.data.frame(exp(beta)) |>
```

```
    rownames_to_column()
cols <- c("Feature", "Coefficient")
colnames(beta_exp) <- cols
beta_exp <- beta_exp |>
    filter(Feature != "(Intercept)")
beta_exp <- beta_exp |>
    mutate(diff = round(Coefficient - 1, 3) * 100) |>
    arrange(desc(diff))
cols <- c("Feature", "Coefficient", "Percentage Change in Odds of Car Crash")
colnames(beta_exp) <- cols
knitr::kable(beta_exp, format = "simple")

model_blr_2 <- update(model_blr_2, ~ . - EMPLOYED)

vif(model_blr_2)

choices <- c("AGE", "CLM_FREQ_LOG", "URBANICITY", "MVR_PTS_LOG", "OLDCLAIM_LOG", "PARENT1", "REVOKED",
print(choices)

model_blr_3 <- glm(TARGET_FLAG ~ AGE + CLM_FREQ_LOG + URBANICITY + MVR_PTS_LOG +  OLDCLAIM_LOG + PARENT
                   family = 'binomial', data = train_df_trans)
summary(model_blr_3)

model_blr_3 <- update(model_blr_3, ~ . - WHITE_COLLAR)
model_blr_3 <- update(model_blr_3, ~ . - EMPLOYED)
summary(model_blr_3)

beta <- coef(model_blr_3)
beta_exp <- as.data.frame(exp(beta)) |>
    rownames_to_column()
cols <- c("Feature", "Coefficient")
colnames(beta_exp) <- cols
beta_exp <- beta_exp |>
    filter(Feature != "(Intercept)")
beta_exp <- beta_exp |>
    mutate(diff = round(Coefficient - 1, 3) * 100) |>
    arrange(desc(diff))
cols <- c("Feature", "Coefficient", "Percentage Change in Odds of Car Crash")
colnames(beta_exp) <- cols
knitr::kable(beta_exp, format = "simple")

vif(model_blr_3)

model_blr_3 <- update(model_blr_3, ~ . - OLDCLAIM_LOG)

model_mlr_1 <- lm(TARGET_AMT ~ ., data = alt_train_df_imputed |>
                      filter(TARGET_FLAG == 1) |> select(-TARGET_FLAG))
model_mlr_1 <- step(model_mlr_1, trace=0)
summary(model_mlr_1)

vif(model_mlr_1)

model_mlr_2 <- lm(TARGET_AMT_LOG ~ ., data = alt_train_df_trans |>
                      filter(TARGET_FLAG == 1) |> select(-TARGET_FLAG))
model_mlr_2 <- step(model_mlr_2, trace=0)
summary(model_mlr_2)

vif(model_mlr_2)

model_mlr_3 <- rlm(TARGET_AMT_LOG ~ MSTATUS + SEX + BLUEBOOK_SQRT + RED_CAR + REVOKED + MVR_PTS_LOG,
                   data = alt_train_df_trans |>
                      filter(TARGET_FLAG == 1) |> select(-TARGET_FLAG))
summary(model_mlr_3)

palette <- brewer.pal(n = 12, name = "Paired")
mmps(model_blr_1, layout = c(3, 4), grid = FALSE, col.line = palette[c(2,6)],
     main = "Model BLR:1")

mmps(model_blr_2, layout = c(2, 2), grid = FALSE, col.line = palette[c(2,6)],
     main = "Model BLR:2")

mmps(model_blr_3, layout = c(2, 2), grid = FALSE, col.line = palette[c(2,6)],
     main = "Model BLR:3")

hlstat1 <- hltest(model_blr_1, verbose = FALSE)
hlstat2 <- hltest(model_blr_2, verbose = FALSE)
hlstat3 <- hltest(model_blr_3, verbose = FALSE)
models <- c("Model BLR:1",
            "Model BLR:2",
            "Model BLR:3")
hl_tbl <- as.data.frame(cbind(models, rbind(hlstat1[2:4], hlstat2[2:4],
                                            hlstat3[2:4])))
cols <- c("Model", "HL Statistic", "DoF", "P Value")
colnames(hl_tbl) <- cols
knitr::kable(hl_tbl, format = "simple")

model_blr_1_train_preds_df <- alt_train_df_imputed |>
    mutate(linpred = predict(model_blr_1),
           predprob = predict(model_blr_1, type = "response"))
model_blr_1_test_preds_df <- alt_test_df_imputed |>
    mutate(linpred = predict(model_blr_1, alt_test_df_imputed),
           predprob = predict(model_blr_1, alt_test_df_imputed, type = "response"))
model_blr_2_train_preds_df <- train_df_imputed |>
    mutate(linpred = predict(model_blr_2),
           predprob = predict(model_blr_2, type = "response"))
model_blr_2_test_preds_df <- test_df_imputed |>
    mutate(linpred = predict(model_blr_2, test_df_imputed),
           predprob = predict(model_blr_2, test_df_imputed, type = "response"))
model_blr_3_train_preds_df <- train_df_trans |>
    mutate(linpred = predict(model_blr_3),
           predprob = predict(model_blr_3, type = "response"))
model_blr_3_test_preds_df <- test_df_trans |>
    mutate(linpred = predict(model_blr_3, test_df_trans),
           predprob = predict(model_blr_3, test_df_trans, type = "response"))
par(mfrow=c(3,2))
par(mai=c(.3,.3,.3,.3))
```

```r
roc1 <- roc(model_blr_1_train_preds_df$TARGET_FLAG,
            model_blr_1_train_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:1 ROC (Train)")
roc2 <- roc(model_blr_1_test_preds_df$TARGET_FLAG,
            model_blr_1_test_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:1 ROC (Test)")
roc3 <- roc(model_blr_2_train_preds_df$TARGET_FLAG,
            model_blr_2_train_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:2 ROC (Train)")
roc4 <- roc(model_blr_2_test_preds_df$TARGET_FLAG,
            model_blr_2_test_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:2 ROC (Test)")
roc5 <- roc(model_blr_3_train_preds_df$TARGET_FLAG,
            model_blr_3_train_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:3 ROC (Train)")
roc6 <- roc(model_blr_3_test_preds_df$TARGET_FLAG,
            model_blr_3_test_preds_df$predprob,
            plot = TRUE, print.auc = TRUE, show.thres = TRUE)
title(main = "Model BLR:3 ROC (Test)")

model_blr_1_train_preds_df <- model_blr_1_train_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_1_test_preds_df <- model_blr_1_test_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_2_train_preds_df <- model_blr_2_train_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_2_test_preds_df <- model_blr_2_test_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_3_train_preds_df <- model_blr_3_train_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_3_test_preds_df <- model_blr_3_test_preds_df |>
    mutate(predicted = as.factor(ifelse(predprob>0.5,1,0)))
model_blr_1_train_cm <- confusionMatrix(model_blr_1_train_preds_df$predicted,
                                        model_blr_1_train_preds_df$TARGET_FLAG,
                                        positive = "1")
model_blr_1_test_cm <- confusionMatrix(model_blr_1_test_preds_df$predicted,
                                        model_blr_1_test_preds_df$TARGET_FLAG,
                                        positive = "1")
model_blr_2_train_cm <- confusionMatrix(model_blr_2_train_preds_df$predicted,
                                        model_blr_2_train_preds_df$TARGET_FLAG,
                                        positive = "1")
model_blr_2_test_cm <- confusionMatrix(model_blr_2_test_preds_df$predicted,
                                        model_blr_2_test_preds_df$TARGET_FLAG,
                                        positive = "1")
model_blr_3_train_cm <- confusionMatrix(model_blr_3_train_preds_df$predicted,
                                        model_blr_3_train_preds_df$TARGET_FLAG,
                                        positive = "1")
model_blr_3_test_cm <- confusionMatrix(model_blr_3_test_preds_df$predicted,
                                         model_blr_3_test_preds_df$TARGET_FLAG,
                                         positive = "1")
plt1a <- as.data.frame(model_blr_1_train_cm$table)
plt1a$Reference <- factor(plt1a$Reference, levels=rev(levels(plt1a$Reference)))
plt1a <- plt1a |>
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm1a <- plt1a |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:1 (Train) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
plt1b <- as.data.frame(model_blr_1_test_cm$table)
plt1b$Reference <- factor(plt1b$Reference, levels=rev(levels(plt1b$Reference)))
plt1b <- plt1b |>
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm1b <- plt1b |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:1 (Test) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
plt2a <- as.data.frame(model_blr_2_train_cm$table)
plt2a$Reference <- factor(plt2a$Reference, levels=rev(levels(plt2a$Reference)))
plt2a <- plt2a |>
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm2a <- plt2a |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:2 (Train) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
plt2b <- as.data.frame(model_blr_2_test_cm$table)
plt2b$Reference <- factor(plt2b$Reference, levels=rev(levels(plt2b$Reference)))
plt2b <- plt2b |>
```

```
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm2b <- plt2b |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:2 (Test) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
plt3a <- as.data.frame(model_blr_3_train_cm$table)
plt3a$Reference <- factor(plt3a$Reference, levels=rev(levels(plt3a$Reference)))
plt3a <- plt3a |>
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm3a <- plt3a |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:3 (Train) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
plt3b <- as.data.frame(model_blr_3_test_cm$table)
plt3b$Reference <- factor(plt3b$Reference, levels=rev(levels(plt3b$Reference)))
plt3b <- plt3b |>
    mutate(Label = case_when(Prediction == 0 & Reference == 0 ~ "TN",
                             Prediction == 1 & Reference == 1 ~ "TP",
                             Prediction == 0 & Reference == 1 ~ "FN",
                             Prediction == 1 & Reference == 0 ~ "FP"))
pcm3b <- plt3b |>
    ggplot(aes(x = Reference, y = Prediction)) +
    geom_tile(fill = "white", col = "black") +
    geom_text(aes(label = Freq)) +
    geom_text(aes(label = Label, hjust = 3)) +
    scale_x_discrete(position = "top") +
    labs(x = "Reference", y = "Prediction", title = "Model BLR:3 (Test) CM") +
    theme(axis.line.x = element_blank(),
          axis.line.y = element_blank())
pcm_all <- plot_grid(pcm1a, pcm1b, pcm2a, pcm2b, pcm3a, pcm3b,
                     ncol = 2)
pcm_all

metrics_a <- as.data.frame(cbind(model_blr_1_train_cm$byClass,
                                 model_blr_1_test_cm$byClass,
                                 model_blr_2_train_cm$byClass,
                                 model_blr_2_test_cm$byClass,
                                 model_blr_3_train_cm$byClass,
                                 model_blr_3_test_cm$byClass))
colnames(metrics_a) <-c('1 (Train)',
                        '1 (Test)',
                        '2 (Train)',
                        '2 (Test)',
                        '3 (Train)',
                        '3 (Test)')
metrics <- rbind(metrics_a,
                 c(model_blr_1_train_cm$overall[1],
                   model_blr_1_test_cm$overall[1],
                   model_blr_2_train_cm$overall[1],
                   model_blr_2_test_cm$overall[1],
                   model_blr_3_train_cm$overall[1],
                   model_blr_3_test_cm$overall[1]),
                 c(1-model_blr_1_train_cm$overall[1],
                   1-model_blr_1_test_cm$overall[1],
                   1-model_blr_2_train_cm$overall[1],
                   1-model_blr_2_test_cm$overall[1],
                   1-model_blr_3_train_cm$overall[1],
                   1-model_blr_3_test_cm$overall[1]),
                 c(roc1$auc,
                   roc2$auc,
                   roc3$auc,
                   roc4$auc,
                   roc5$auc,
                   roc6$auc))
metrics <- round(metrics, 3)
rownames(metrics)[12:14] <- c('Accuracy','Classification Error Rate','AUC')
knitr::kable(metrics, format = "simple")

par(mfrow=c(2,2))
par(mai=c(.3,.3,.3,.3))
plot(model_mlr_1)
mtext("Model MLR:1", side = 3, line = -1.5, outer = TRUE)

par(mfrow=c(2,2))
par(mai=c(.3,.3,.3,.3))
plot(model_mlr_2)
mtext("Model MLR:2", side = 3, line = -1.5, outer = TRUE)

par(mfrow=c(2,2))
par(mai=c(.3,.3,.3,.3))
plot(model_mlr_3)
mtext("Model MLR:3", side = 3, line = -1.5, outer = TRUE)

excl <- c("TARGET_AMT", "TARGET_FLAG")
test_x <- alt_test_df_imputed |>
    filter(TARGET_FLAG == 1) |>
    select(-all_of(excl))
test_y <- alt_test_df_imputed |>
    filter(TARGET_FLAG == 1) |>
    select(TARGET_AMT)
```

```r
test_y <- as.numeric(test_y$TARGET_AMT)
test_pred <- predict(model_mlr_1, test_x)
model_mlr_1_test_rsq <- as.numeric(R2(test_pred, test_y, form = "traditional"))
model_mlr_1_test_rmse <- as.numeric(RMSE(test_pred, test_y))
excl <- c("TARGET_AMT_LOG", "TARGET_FLAG")
test_x <- alt_test_df_trans |>
    filter(TARGET_FLAG == 1) |>
    select(-all_of(excl))
test_y <- alt_test_df_trans |>
    filter(TARGET_FLAG == 1) |>
    select(TARGET_AMT_LOG)
test_y <- exp(as.numeric(test_y$TARGET_AMT_LOG))
test_pred <- exp(predict(model_mlr_2, test_x))
model_mlr_2_test_rsq <- as.numeric(R2(test_pred, test_y, form = "traditional"))
model_mlr_2_test_rmse <- as.numeric(RMSE(test_pred, test_y))
test_x <- alt_test_df_trans |>
    filter(TARGET_FLAG == 1) |>
    select(-all_of(excl))
test_y <- alt_test_df_trans |>
    filter(TARGET_FLAG == 1) |>
    select(TARGET_AMT_LOG)
test_y <- exp(as.numeric(test_y$TARGET_AMT_LOG))
test_pred <- exp(predict(model_mlr_3, test_x))
model_mlr_3_test_rsq <- as.numeric(R2(test_pred, test_y, form = "traditional"))
model_mlr_3_test_rmse <- as.numeric(RMSE(test_pred, test_y))
models <- c("Model MLR:1", "Model MLR:2", "Model MLR:3")
mlr_summary <- as.data.frame(cbind(models,
                                   pred_rsq = c(model_mlr_1_test_rsq,
                                                model_mlr_2_test_rsq,
                                                model_mlr_3_test_rsq),
                                   rmse = c(model_mlr_1_test_rmse,
                                            model_mlr_2_test_rmse,
                                            model_mlr_3_test_rmse)))
knitr::kable(mlr_summary, format = "simple")

my_url <- "https://raw.githubusercontent.com/waheeb123/Data-621/main/Homeworks/Homework%204/insurance-ev
eval_df <- read.csv(my_url, na.strings = "")
eval_df_w_preds <- eval_df
# car type
x <- eval_df_w_preds$CAR_TYPE
eval_df_w_preds$CAR_TYPE <- case_match(x, "z_SUV" ~ "SUV", .default = x)
eval_df_w_preds$CAR_TYPE <- factor(eval_df_w_preds$CAR_TYPE,
                           levels = c("Minivan", "Panel Truck",
                                      "Pickup", "Sports Car", "SUV", "Van"))
# education
x <- eval_df_w_preds$EDUCATION
eval_df_w_preds$EDUCATION <- case_match(x, "z_High School" ~ "High School", .default = x)
eval_df_w_preds$EDUCATION <- factor(eval_df_w_preds$EDUCATION,
                             levels = c("<High School", "High School",
                                        "Bachelors", "Masters", "PhD"))
# job
x <- eval_df_w_preds$JOB
eval_df_w_preds$JOB <- case_match(x, "z_Blue Collar" ~ "Blue Collar", .default = x)
```

```
eval_df_w_preds$JOB <- factor(eval_df_w_preds$JOB, levels = c("Blue Collar", "Clerical",
                                              "Doctor", "Home Maker","Lawyer",
                                              "Manager", "Professional", "Student"))
# single parent
eval_df_w_preds <- eval_df_w_preds |>
  mutate(PARENT1 = as.factor(ifelse(PARENT1 == "Yes", 1, 0)))
# marital status
x <- eval_df_w_preds$MSTATUS
eval_df_w_preds$MSTATUS <- case_match(x, "z_No" ~ "No", .default = x)
eval_df_w_preds <- eval_df_w_preds |>
  mutate(MSTATUS = as.factor(ifelse(MSTATUS == "Yes", 1, 0)))
# red car
x <- eval_df_w_preds$RED_CAR
eval_df_w_preds$RED_CAR <- case_match(x, "no" ~ "No", "yes" ~ "Yes", .default = x)
eval_df_w_preds <- eval_df_w_preds |>
  mutate(RED_CAR = as.factor(ifelse(RED_CAR == "Yes", 1, 0)))
# revoked
eval_df_w_preds <- eval_df_w_preds |>
  mutate(REVOKED = as.factor(ifelse(REVOKED == "Yes", 1, 0)))
# sex
x <- eval_df_w_preds$SEX
eval_df_w_preds$SEX <- case_match(x, "M" ~ "Male", "z_F" ~ "Female", .default = x)
eval_df_w_preds$SEX <- factor(eval_df_w_preds$SEX, levels = c("Male", "Female"))

# urban city - 1 if urban, 0 if rural
x <- eval_df_w_preds$URBANICITY
eval_df_w_preds$URBANICITY <- case_match(x, "Highly Urban/ Urban" ~ "Urban",
                                 "z_Highly Rural/ Rural" ~ "Rural", .default = x)
eval_df_w_preds <- eval_df_w_preds |>
  mutate(URBANICITY = as.factor(ifelse(URBANICITY == "Urban", 1, 0)))
vars <- c("INCOME", "HOME_VAL", "BLUEBOOK", "OLDCLAIM")
eval_df_w_preds <- eval_df_w_preds |>
    mutate(across(all_of(vars), ~gsub("\\$|,", "", .) |> as.integer()))
drop <- c("INCOME", "HOME_VAL")
eval_df_w_preds <- eval_df_w_preds |>
    mutate(INCOME_THOU = INCOME / 1000,
           HOME_VAL_THOU = HOME_VAL / 1000) |>
    select(-all_of(drop))
missing <- c("AGE", "INCOME_THOU", "YOJ", "HOME_VAL_THOU", "CAR_AGE", "JOB")
x <- names(eval_df_w_preds)
not_missing <- x[!x %in% missing]
init = mice(eval_df_w_preds, maxit=0)
meth = init$method
predM = init$predictorMatrix
meth[not_missing] = ""
meth[c("AGE")] = "pmm" #Predictive mean matching
meth[c("INCOME_THOU")] = "pmm"
meth[c("YOJ")] = "pmm"
meth[c("HOME_VAL_THOU")] = "pmm"
meth[c("CAR_AGE")] = "pmm"
meth[c("JOB")] = "polyreg" #Polytomous (multinomial) logistic regression
imputed = mice(eval_df_w_preds, method=meth, predictorMatrix=predM, m=5,
                   printFlag = FALSE)
```

```
eval_df_w_preds <- complete(imputed)
eval_df_w_preds <- eval_df_w_preds |>
    mutate(PREDPROB = predict(model_blr_1, eval_df_w_preds,
                              type = "response"),
           TARGET_FLAG = as.factor(ifelse(PREDPROB > 0.5, 1, 0)),
           TARGET_AMT = case_when(TARGET_FLAG == 1 ~ predict(model_mlr_1,
                                                             eval_df_w_preds),
                                  TRUE ~ 0))
write.csv(eval_df_w_preds,file='HW4_Eval_PredProbs_Flags_Amounts.csv')

eval_df_w_preds |> group_by(TARGET_FLAG) |> summarise(cnt=n())
```