Predicting Term Life Insurance Purchases

PSTAT 196

June 10th, 2018

Group Members:

Wendy Gao
Emeric Szaboky
Jacobo Pereira-Pacheco

Advisor:

Ian Duncan

Contents

1 Term Life Insurance

The purpose of this project, pursued on behalf of The Society of Actuaries, is to develop predictive models for the prediction of purchasers of term life insurance policies, and for predicting quantities of policies purchased. The data consists of information regarding survey responders for a survey regarding inquiry about term life insurance purchase. As a part of this research, exploratory analysis was performed in order to investigate those predictors that may be associated with the purchase of term insurance, and to derive other variables that may reveal key characteristics of those who purchase term insurance. The report is extensive, including a plethora of answers to this question, including multiple drawbacks that individuals may run into.

2 Data Exploration

Before building predictive models, we conducted basic exploratory analysis. The variables ETHNICITY, NETVALUE, and BORROWCVLIFEPOL were dropped from the dataset, because they were either unexplained or ambiguously defined in documentation. We began by examining the predictors given in the dataset in order to highlight any trends or areas of interests that may be associated with term insurance purchase. To do this, we first created variables EDUDIFF and AGEDIFF to see if the difference in education level and age between the term insurance purchaser and spouse could potentially reveal term insurance purchasing patterns. We then aggregated basic summary statistics (mean, median, min, max, variance) for covariates to further understand the data. This revealed whether some variables should be treated as continuous or categorical, and which variables had missing data that was not stored as NA values. The observations with missing values were altered to store NA values.

Next, we computed a correlation coefficient matrix (using the Pearson measure) to see the correlations between the variables. Due to the fact that there were 136 missing values for the variable SAGE (spouse age), the correlation coefficient for AGEDIFF and Term_Flag could not be computed until those 136 values were ignored. We proceeded to instruct R to ignore the rows with missing observations and obtained correlation values for all the variables, which reveal interesting insights. It can be seen that although the values of the correlation coefficients are small, AGEDIFF and Term_Flag have a negative correlation, suggesting that it is possible that couples with smaller age differences are more likely to purchase life insurance.

Subsequently, log-transformations were explored to examine how the scatterplot matrix would change. The logarithmic transformations merited investigation because upon observing the initial scatterplot matrix, there appeared to be some exponential trends that would benefit from a logarithmic transformation. Specifically, the variables FACE and INCOME showcased moderately strong linear trends after being transformed. Other variables (AGE and EDUCATION) were transformed to simply see what else the data may convey.

In addition to the exploration above, we plotted bar plots with AGEDIFF on the x-axis against variables INCOME, EDUCATION, and FACE. It was found that those responders with smaller age differences tended to have higher incomes, education levels, and face values of life insurance. It was also discovered that those with higher education levels tended to have higher face values of life insurance. (See figures 1-4)

Next, we looked at boxplots in an effort to investigate the general relationship between the covariates and Term_Flag. It is observed that the responders who purchased insurance tended to have higher education levels, and slightly more household members. (See figures 5-8)

Afterwards, we examined the distributions of each variable using histograms. (See figures 9-14) Most variables were fairly normally distributed, but some variables needed to be truncated. (See figures 15-21) We decided to truncate the variables INCOME and TOTINCOME both at 200,000. We decided to truncate the variables CHARITY, FACE, FACECVLIFEPOLICIES, and CASHCVLIFEPOLICIES at 50,000, 50,000, 100,000, and 6,500 respectively.

We created additional bar graphs to learn the demographics of those who purchase term insurance. We found that married couples, men, and educated people make the majority of those who purchase term insurance.

We also found that a good number of people who purchased insurance reported 2 as the number of household members. (See figures 22-25)

3 Data Preparation

In preparing the data for model building it was most important to correctly split the data into training and test sets. Because the data is imbalanced and fairly small, it was integral to ensure that purchasers and non-purchasers of term insurance were balanced. Therefore, when building the train and test sets respectively, the original data was split into purchasers and non-purchasers, randomly sampled, and later re-combined to create the training and test sets to be used for model building.

4 Data Modeling

For predicting purchasers of term life insurance, three different logistic regression models were fit as required. Moreover, probit and complementary log log (c-log-log) models were explored as alternatives to logistic regression. We found this to be beneficial as it would be interesting to see how different link functions within the binomial family would perform. Although creating three different logistic regression models is beneficial, depending on the situation, probit regression or c-log-log regression may outperform logistic regression.

For predicting FACE, we explored ridge regression, lasso regression and quantile regression. Ridge regression shrinks the coefficients that do not influence the dependent variable as much, thus highlighting which variables are influential on the face value. Similar to ridge regression, lasso regression also shrinks the coefficients that are "unimportant," except it shrinks the coefficients to zero, creating simpler and, sometimes, more interpretable models. Quantile regression is explored to investigate which predictors of face are sufficient with linear regression, and which will need more robust regression techniques.

4.1 Logistic Regression

In the process of modeling with logistic regression, it was discovered that particular predictors had significant multi-collinearity resulting in modeling issues. In order to resolve the multicollinearity among predictors, variance inflation factor (VIF function) from the "car" package was used to identify those highly correlated predictors (variables with VIF > 5 dropped). VIF recognizes variables with perfect correlation as aliased variables ("same") and is unable to produce output with the presence of these variables. Therefore, the alias function from the "car" package was also used to identify perfectly correlated variables. In combination with one another, these two functions were used to identify the variables to be removed in order to resolve multicollinearity. Due to strong multicollinearity, all spousal variables and derived variables (AGEDIFF, EDUDIFF) were removed from regression modeling.

From this point, the logistic regression full model (Model 1) was acquired, including all predictors which were verified to be independent. It possessed an area under the ROC curve (AUC) of 0.7186257. A reduced logistic regression model (Model 2) with an AUC of 0.7075949 was obtained by the removal of FACECVLIFEPOLICIES, TOTINCOME and NUMHH. Model 2 possessed a lower Akaike Information Criterion (AIC), but a smaller AUC as well. Low AIC and high AUC respectively are generally considered desirable. However, in this situation, Model 2 is able to obtain the same class prediction accuracy even with a lower AUC. This is because the prediction accuracy is computed at the threshold value of 0.5, while AUC is computed by adding all of the accuracies computed for all possible threshold values[1]. It can be concluded that class prediction accuracy is our most important measure of model significance since it assesses the model with respect to the chosen threshold. Lastly, another reduced model (Model 3) was obtained by inclusion of the same predictors from Model 2 with the addition of a log transformed AGE predictor, instead of AGE. Model 3 possessed the lowest AIC, and an AUC of 0.7106691, less than that of Model 1 but greater than that of Model 2. It was decided that the reduced Model 3 was the most appropriate model because it had the best class prediction accuracy of the 3 models, while having the 2nd-largest AUC.

4.2 Probit Regression

Although the goal was to produce a suitable logistic regression, we explored probit regression models because probit models may be better suited when non-constant variances is present in the datasets[2]. Observing the logit fitted values versus the probit fitted values, the two estimations are very similar. Inspecting the ROC curve and AUC values for the probit model, an AUC of 0.7459313 was achieved, which proved favorable than logistics AUC value of 0.7106691. However, the goal of probit was to showcase that if a different probability threshold was chosen other than 0.5, probit may classify better than logistic. Interestingly, the KS statistic for the probit model is 44% compared to 33.6% by the logistic, indicating by KS standards that the data is technically suitable for a logistic model[3].

4.3 C-Log-Log Regression

Exploring a complementary log-log (C-Log-Log) model is appropriate when the prediction of the probability of an event may be very small of very large[4]. In relation to our term life insurance data, the prediction of a term insurance may be very small or very large contingent on specific variables e.g. level of education. Thus, a c-log-log model was created and has a wider range of fitted values compared to the fitted values by the logit model. Inspecting the ROC curve, the AUC value of c-log-log was 0.7466546, showcasing that the c-log-log model is beneficial when choosing different probability thresholds as opposed to in doing so with the logistic model. Similar to the probit model, the c-log-log model has a higher KS statistic value than logistic, being 43.1%. Again, the data was not developed with the intention of a c-log-log regression model.

4.4 Random Forest

Although the central focus of this project was the prediction of life insurance purchasers using logistic regression, random forest modeling was used in addition for classification and variable selection. Random forest produced a valuable variable importance plot, which was used as a reference for variable selection in regression model building. Since random forest is a generally robust black-box machine learning method, it achieved its expected result of being the highest performer in class prediction, with only 9 observations mis-classified and an AUC of 0.9652803.

4.5 Quantile Regression

Quantile regression is a good addition to model building when data tends to be skewed in distributions as shown in the histograms (figures 9-21), and it will be better for a model to consider the change between FACE and predictors depending on the quantile. Visualizing the figures with the confidence bound included, one can observe that age and charity are the only continuous variables that would require quantile regression. This is concluded by the red bounds; if the model is within the model then linear regression is sufficient, but if the model exceeds the bounds, quantile regression would be necessary for those specific quantiles. Running quantile on age reveals that after the 0.65 quantile, the older the term purchaser is, the higher the FACE value will be. Quantile regression on charity reveals that after the 0.5 quantile, the higher the charitable contribution is, the higher the FACE value will be. Other variables included in the logistic regression model were revealed to have linear regression as a sufficient model to predict FACE value.

4.6 Ridge Regression

While the main focus of this project was to predict life insurance purchasers, we also were interested in predicting the face value of insurance purchased from those who purchased insurance. Because we had collinear variables, and a relatively large number of variables compared to the number of observations given, we decided to create use ridge regression to predict the face value of term insurance purchased. From the data of term insurance purchasers, we fitted a ridge regression model using cross-validation to find the optimal lambda that had the lowest mean squared error. The optimal lambda is 0.08595841. The test MSE is 2.568472 and the average percent error of the predicted face values is 3.15%. Included is a figure with the predicted values on the y-axis, and true test values on the x-axis, and a y = x line drawn through. The predictions are distributed around the y = x line with a couple of outliers. Additionally, the ridge regression model was able to accurately predict some face-values (the points lying on the y = x) line.

4.7 Lasso Regression

In addition to the ridge regression model, we also chose to perform a lasso regression model to see if a lasso regression model would perform better. Unlike ridge regression, lasso regression will shrink coefficients that are not associated with the dependent variable to zero, another form of variable selection. The variables that were not shrunken to zero were: GENDER1, MARSTAT1, EDUCATION, NUMHH2, NUMHH6, NUMHH9, TOTINCOME, CHARITY, FACECVLIFEPOLICIES, and CASHCVLIFEPOLICIES. The optimal lambda found through cross-validation was 0.08398222. The test MSE was 2.403266 and the average percent error was 3.11%, slightly outperforming the ridge regression model. Similar to the ridge regression model, it was also able to accurate predict some values (the values lying on the y = x line).

4.8 Conclusion

Out of the three logistic models, the logistic regression model that performed the best was the reduced model that included the log-transformed AGE variable (Model 3). This logistic regression model had the lowest AIC and highest classification accuracy.

When comparing the best logistic regression model with C-Log-Log and Probit regression models, the logistic regression still performed best in regards to classification accuracy. While C-Log-Log and Probit regression had higher AUC values, logistic regression had the lowest number of misclassifications. It was chosen as the best model due to the reasons explained in the logistic regression section above.

Quantile regression was solely used to understand which predictors for FACE needed more robust modeling techniques. Inspecting individual predictors concluded that only AGE and CHARITY benefited from quantile regression.

When predicting the face value of insurance purchased, the ridge and lasso regression performance were similar, with lasso regression barely outperforming ridge regression because of its lower test MSE and average percent errors.

5 Shortcomings

Some remarks on what made this problem difficult and provided strong stopping points in the report:

- The way in which the data was reported, not all of the variables had definitions describing what were key differences i.e. totincome and income differences.
- There is a lot of multicollinearity among the variables, thus regression techniques will continue to produce errors along with NA outpouts until the predictors involved with the multicollinearity were dropped.

- A lot of the multicollinearity was present among the spouse variables. So the spouse variables needed to be dropped in order to produce a model with complete output.
- Spousal variables had to be dropped in order to perform ridge and lasso regression due to the large number of NA values.
- Fluency in R is necessary to attempt this problem.

6 Exploratory Analysis

6.1 Summary Output

```
##       AGE          EDUCATION          SAGE          AGEDIFF
##  Min.   :20.00   Min.   : 2.00   Min.   :19.00   Min.   :-25.000
##  1st Qu.:37.00   1st Qu.:12.00   1st Qu.:36.75   1st Qu.:  0.000
##  Median :47.00   Median :14.00   Median :45.50   Median :  2.000
##  Mean   :47.16   Mean   :14.06   Mean   :45.88   Mean   :  2.379
##  3rd Qu.:58.00   3rd Qu.:16.00   3rd Qu.:55.00   3rd Qu.:  4.000
##  Max.   :85.00   Max.   :17.00   Max.   :78.00   Max.   : 24.000
##                                  NA's   :136     NA's   :136
##    SEDUCATION        NUMHH          INCOME           TOTINCOME
##  Min.   : 0.00   Min.   :1.00   Min.   :     260   Min.   :       0
##  1st Qu.:12.00   1st Qu.:2.00   1st Qu.:   28000   1st Qu.:       0
##  Median :14.00   Median :2.00   Median :   54000   Median :   42500
##  Mean   :13.76   Mean   :2.87   Mean   :  321022   Mean   :  803513
##  3rd Qu.:16.00   3rd Qu.:4.00   3rd Qu.:  106000   3rd Qu.:  121000
##  Max.   :17.00   Max.   :9.00   Max.   :75000000   Max.   :73400000
##  NA's   :136
##     CHARITY             FACE          FACECVLIFEPOLICIES
##  Min.   :      0   Min.   :       0   Min.   :       0
##  1st Qu.:      0   1st Qu.:       0   1st Qu.:       0
##  Median :    500   Median :   10000   Median :       0
##  Mean   :  34089   Mean   :  411170   Mean   :  684656
##  3rd Qu.:   3000   3rd Qu.:  200000   3rd Qu.:   40000
##  Max.   :9010000   Max.   :14000000   Max.   :77000000
##
##  CASHCVLIFEPOLICIES
##  Min.   :      0
##  1st Qu.:      0
##  Median :      0
##  Mean   :  72772
##  3rd Qu.:   1850
##  Max.   :7000000
##
```

```
## [1] "Variance of Variables"
```

```
##                AGE          EDUCATION               SAGE
##       1.917686e+02       8.655467e+00                 NA
##            AGEDIFF         SEDUCATION              NUMHH
##                 NA                 NA       2.233567e+00
##             INCOME          TOTINCOME            CHARITY
##       1.163448e+13       2.212969e+13       1.665269e+11
##               FACE FACECVLIFEPOLICIES CASHCVLIFEPOLICIES
##       1.677991e+12       2.175984e+13       3.575805e+11
```

6.2 Number of Unique Factor-Levels of Covariates

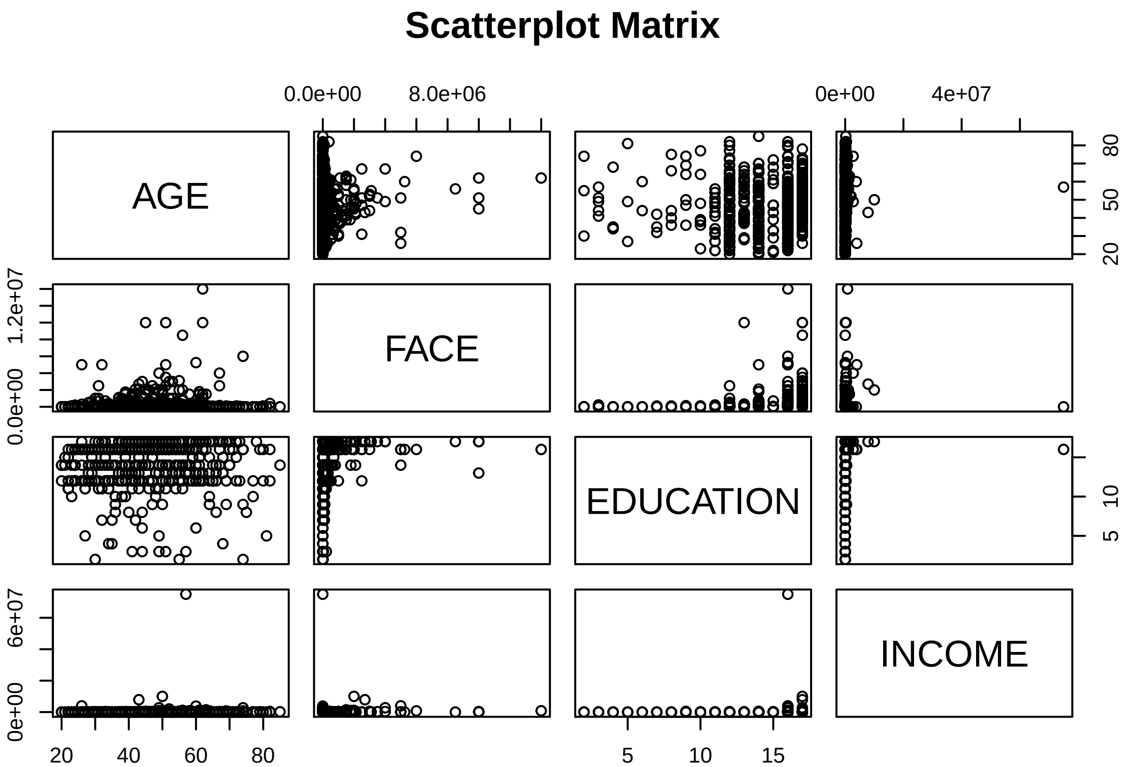
```
## 'data.frame':    500 obs. of  9 variables:
##  $ GENDER    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 1 2 2 2 ...
##  $ SGENDER   : Factor w/ 3 levels "0","1","2": 3 3 3 3 3 3 1 3 3 3 ...
##  $ MARSTAT   : Factor w/ 3 levels "0","1","2": 2 2 2 2 2 3 1 2 3 2 ...
##  $ SMARSTAT  : Factor w/ 4 levels "0","1","2","3": 3 2 3 2 3 2 1 3 2 3 ...
##  $ EDUCATION : Factor w/ 16 levels "2","3","4","5",..: 15 8 15 16 14 10 7 15 3 16 ...
##  $ SEDUCATION: Factor w/ 16 levels "0","2","3","4",..: 15 7 15 13 11 13 NA 16 8 15 ...
##  $ EDUDIFF   : Factor w/ 19 levels "-10","-9","-5",..: 8 9 8 11 11 5 NA 7 3 9 ...
##  $ NUMHH     : Factor w/ 9 levels "1","2","3","4",..: 3 3 5 4 2 4 1 3 2 2 ...
##  $ Term_Flag : Factor w/ 2 levels "0","1": 2 2 2 2 1 2 1 2 1 1 ...
```

6.3 Variable Correlation Assessments

6.3.1 Pearson Correlation Statistics

```
##                   AGE   EDUCATION         FACE     INCOME   TOTINCOME
## AGE        1.00000000  0.14712235  0.038787450 0.05112451  0.13619177
## EDUCATION  0.14712235  1.00000000  0.222841995 0.17097283  0.14464356
## FACE       0.03878745  0.22284200  1.000000000 0.20251571  0.14745440
## INCOME     0.05112451  0.17097283  0.202515707 1.00000000  0.33879883
## TOTINCOME  0.13619177  0.14464356  0.147454395 0.33879883  1.00000000
## NUMHH     -0.42899152 -0.11911464  0.093176516 0.07222389 -0.08411541
## CHARITY    0.12607846  0.06217315  0.003775168 0.24321530  0.18605206
## AGEDIFF    0.29444594  0.01892509 -0.010448524 0.04052349 -0.02425638
## Term_Flag -0.03585052  0.13999824  0.315453790 0.05782060 -0.03086045
##                 NUMHH      CHARITY     AGEDIFF   Term_Flag
## AGE       -0.42899152  0.126078460  0.29444594 -0.03585052
## EDUCATION -0.11911464  0.062173151  0.01892509  0.13999824
## FACE       0.09317652  0.003775168 -0.01044852  0.31545379
## INCOME     0.07222389  0.243215301  0.04052349  0.05782060
## TOTINCOME -0.08411541  0.186052057 -0.02425638 -0.03086045
## NUMHH      1.00000000 -0.056034166 -0.02708953  0.06565501
## CHARITY   -0.05603417  1.000000000 -0.01744347 -0.08035025
## AGEDIFF   -0.02708953 -0.017443468  1.00000000 -0.05079566
## Term_Flag  0.06565501 -0.080350246 -0.05079566  1.00000000
```

6.4 Transformation + Pairs Graph Correlation Assessment



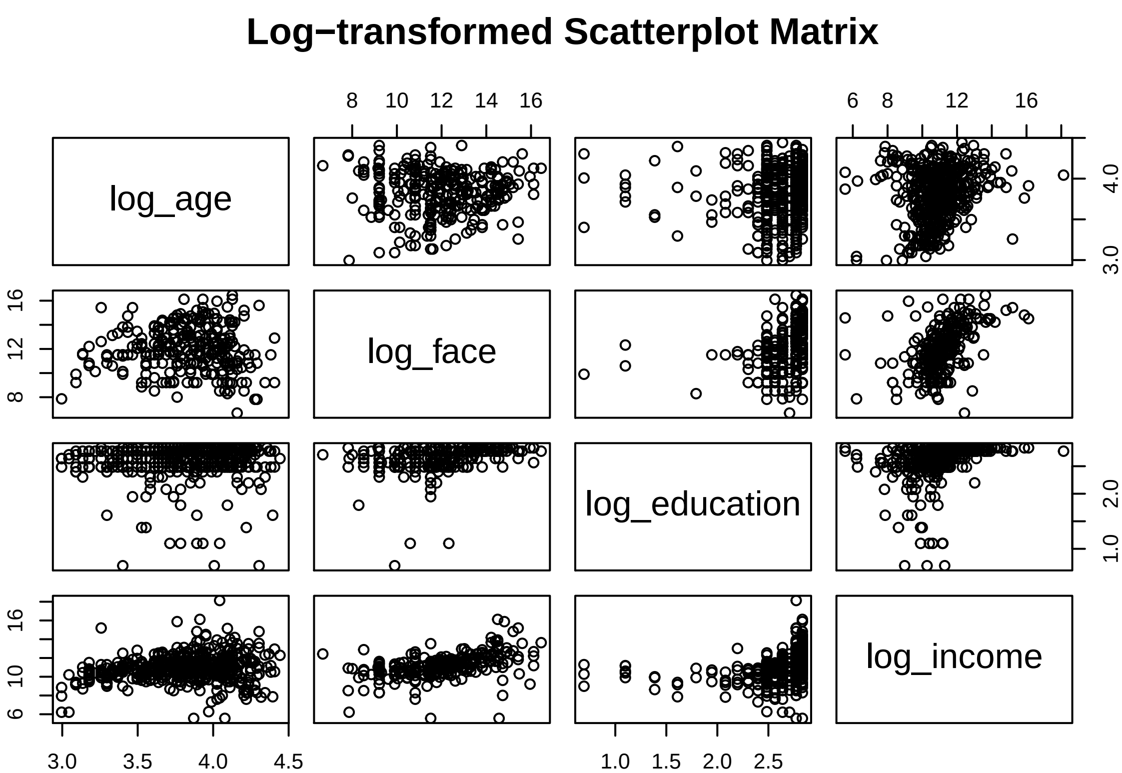


Figure 1

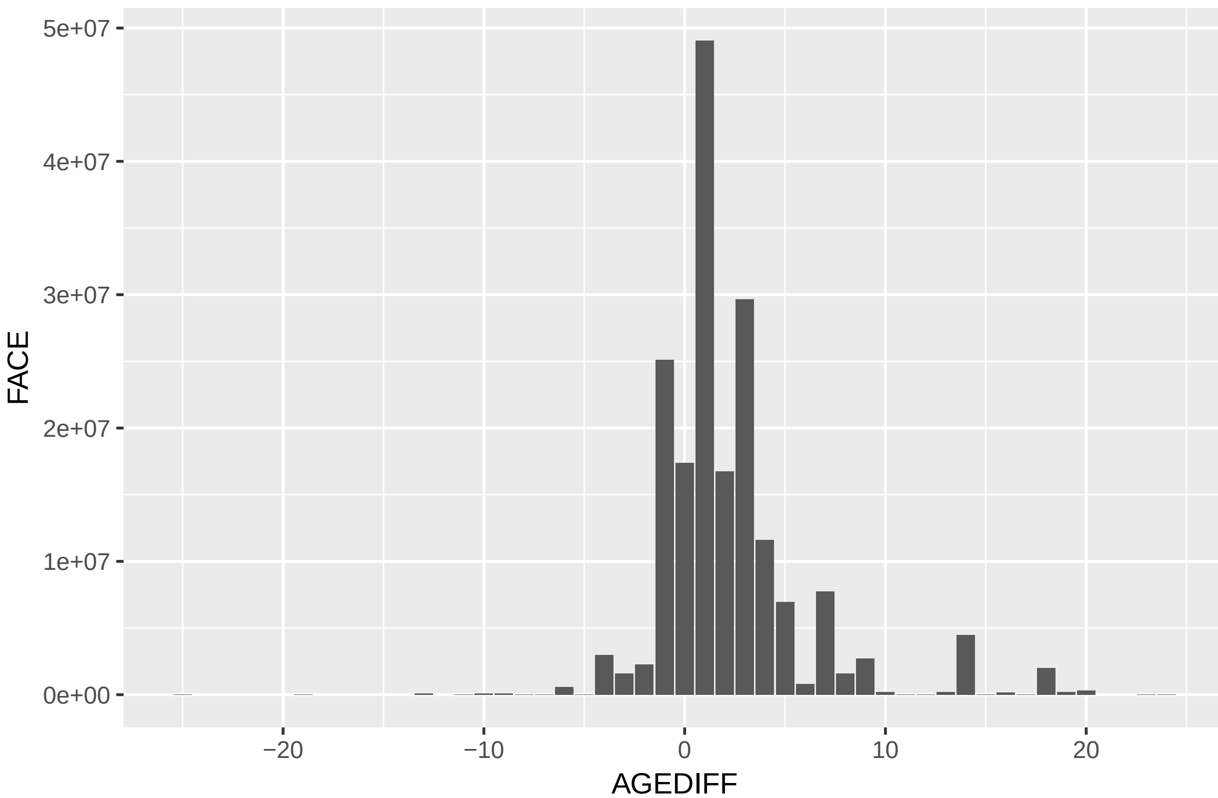


Figure 2

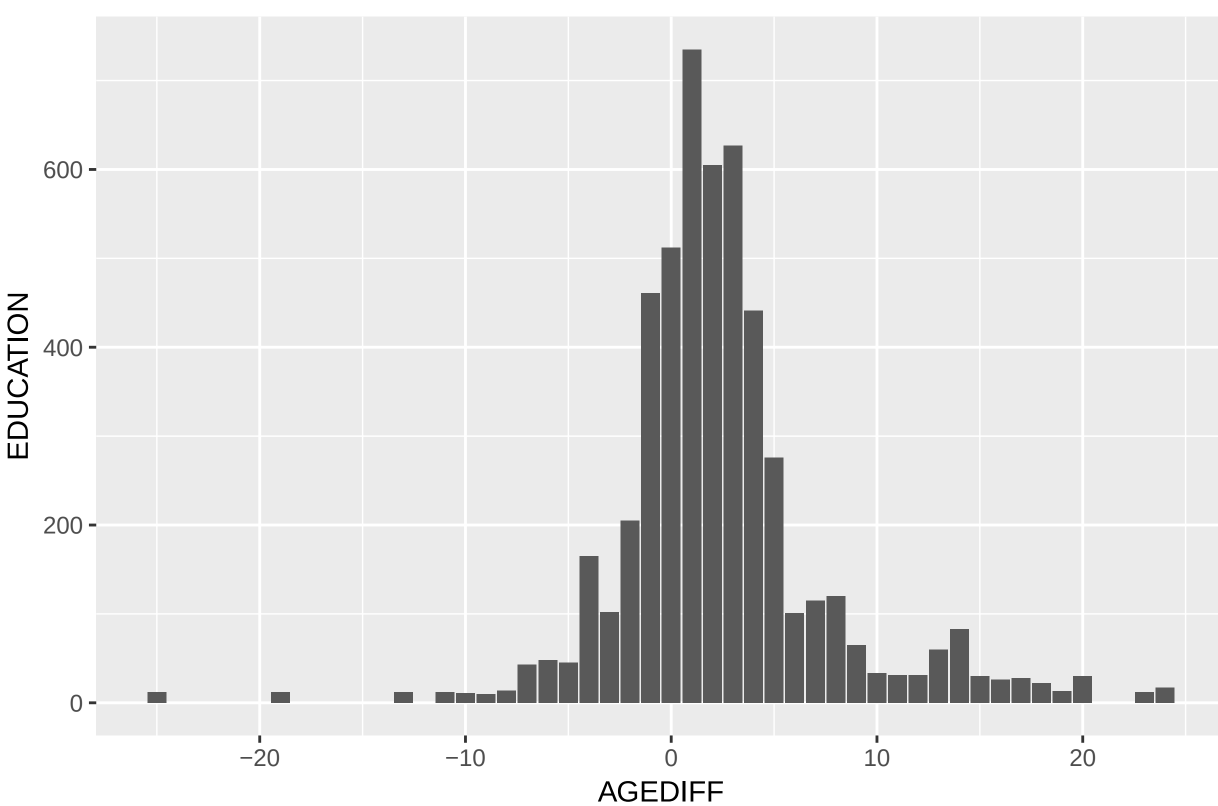


Figure 3

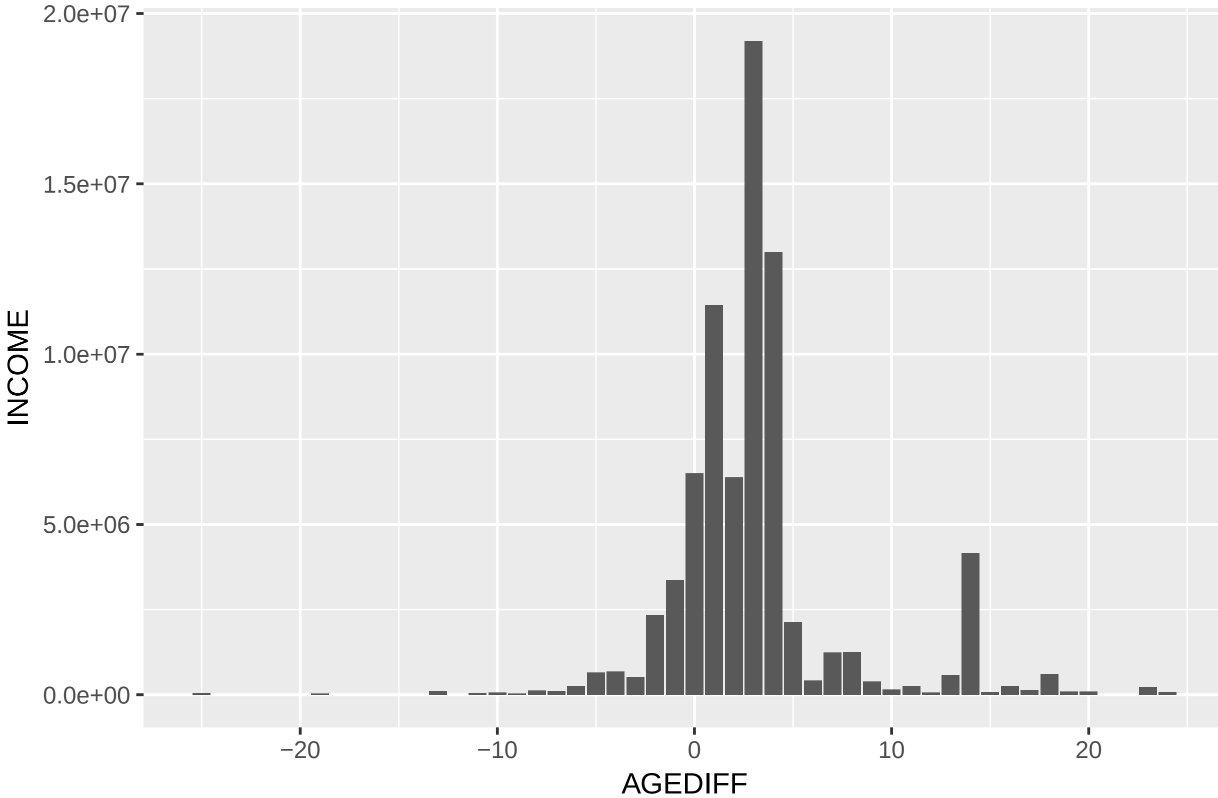


Figure 4

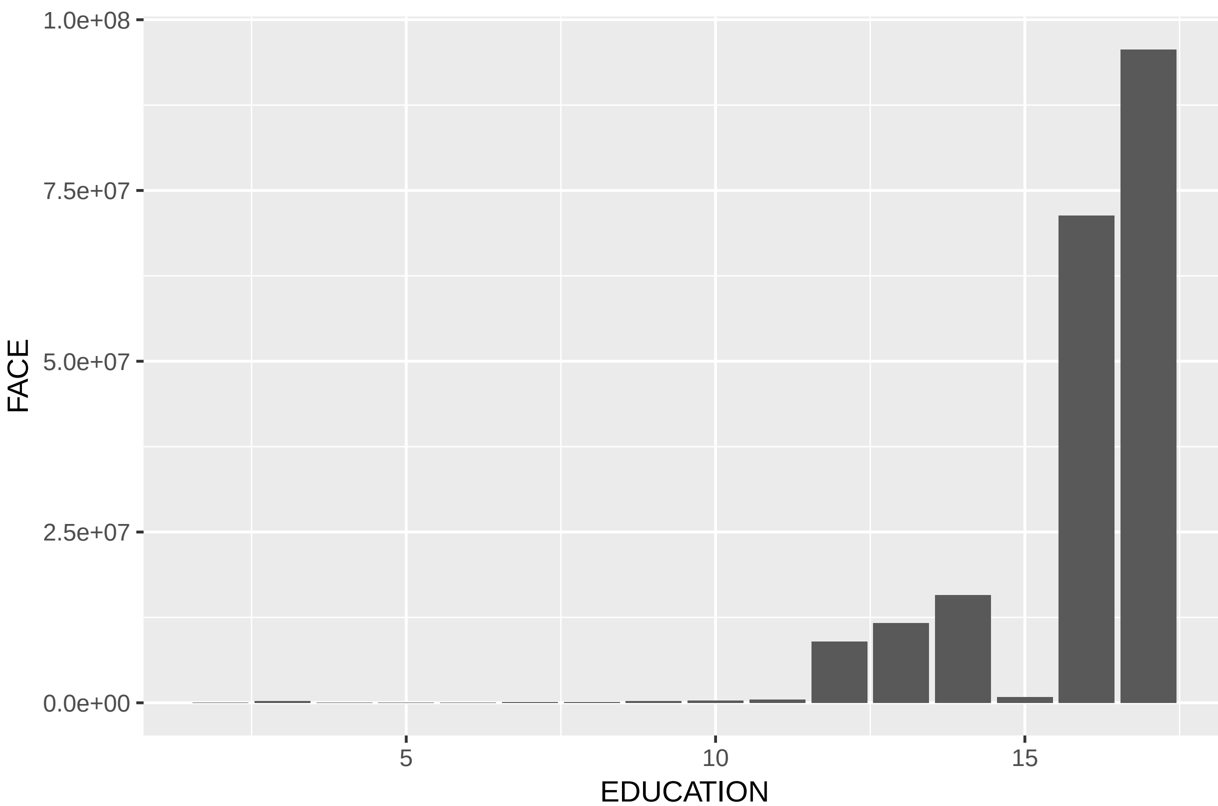


Figure 5

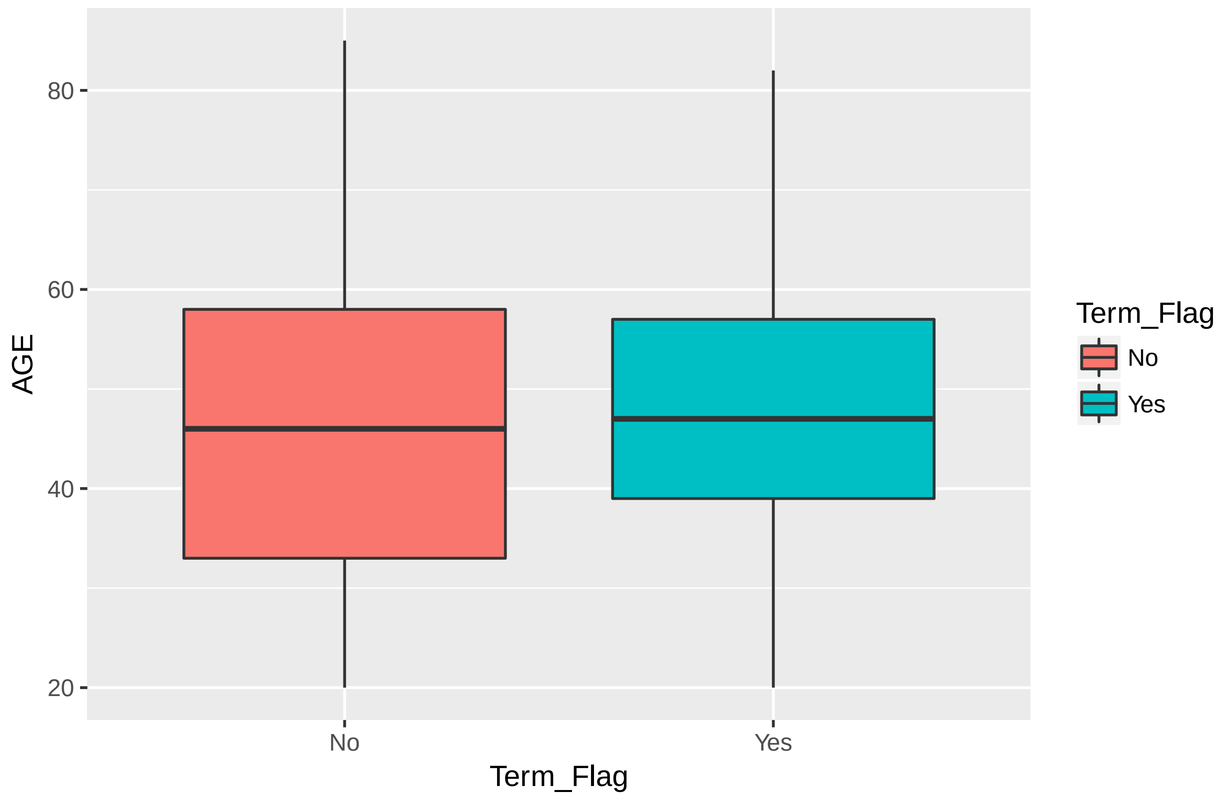


Figure 6

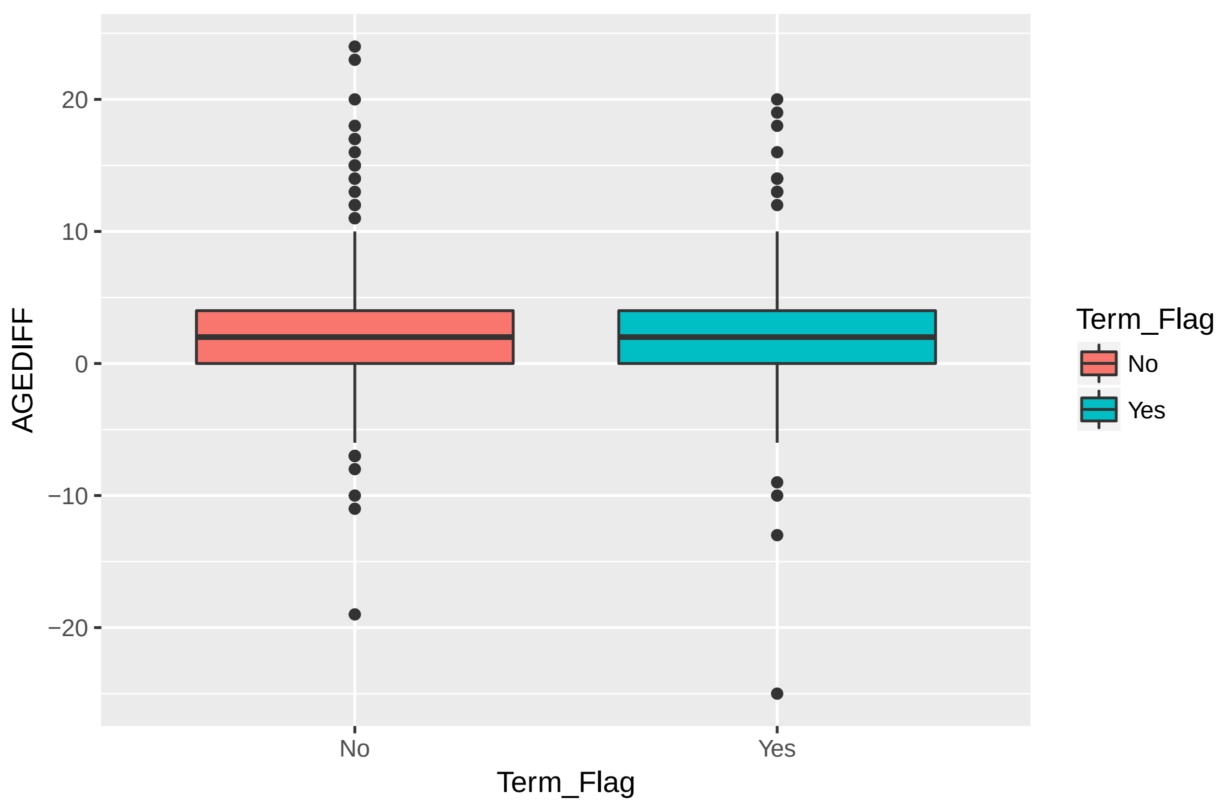


Figure 7

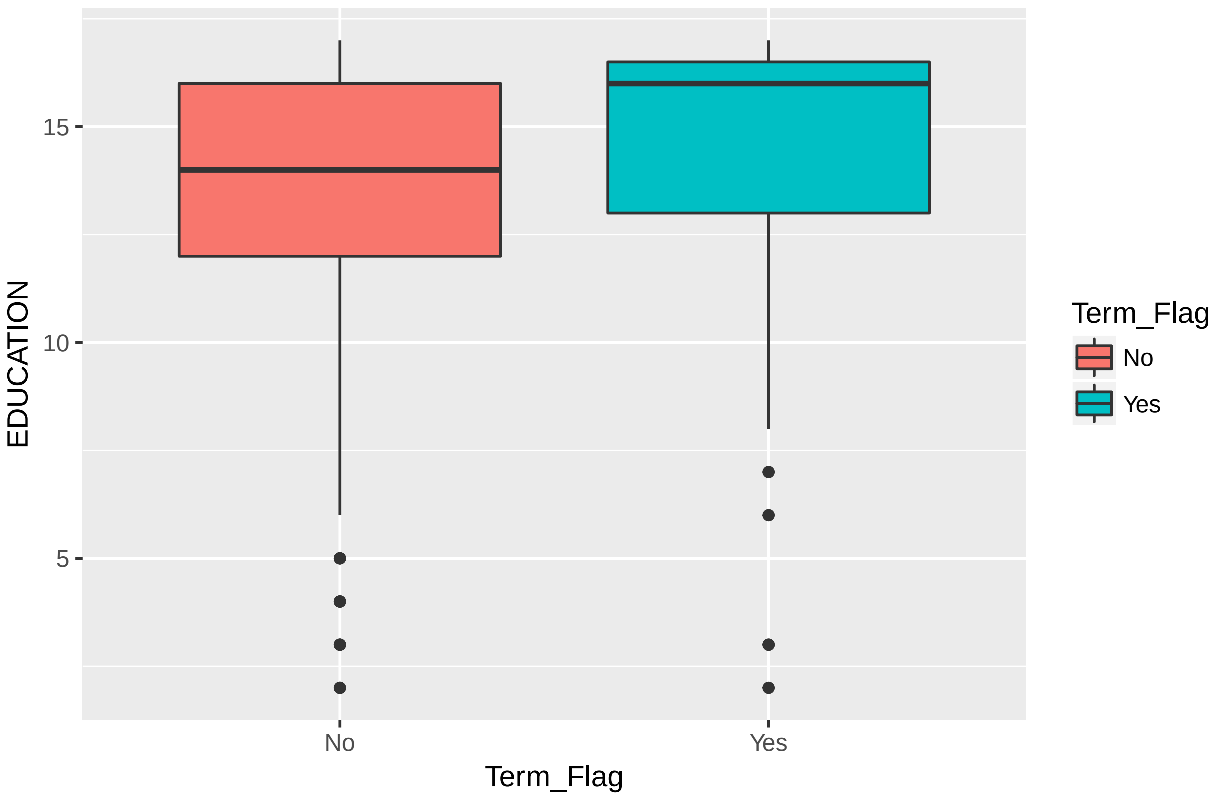


Figure 8

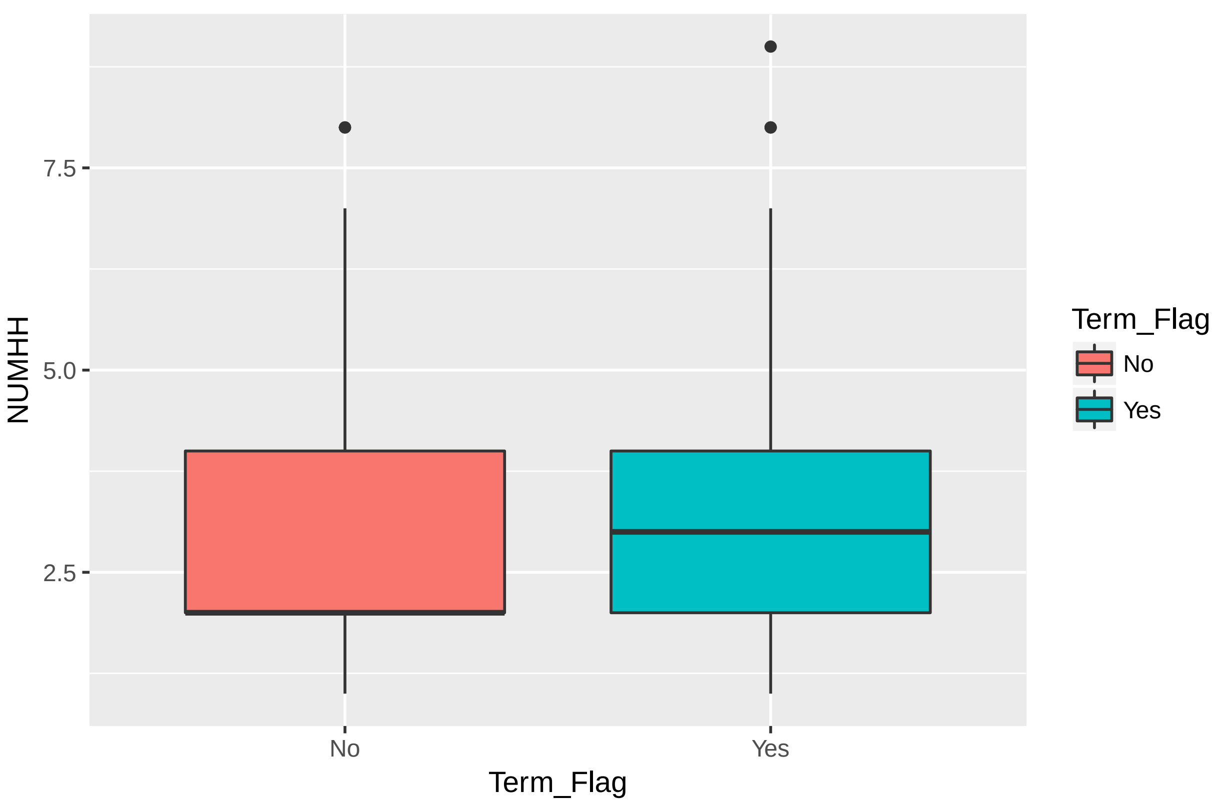


Figure 9. Histogram of AGE

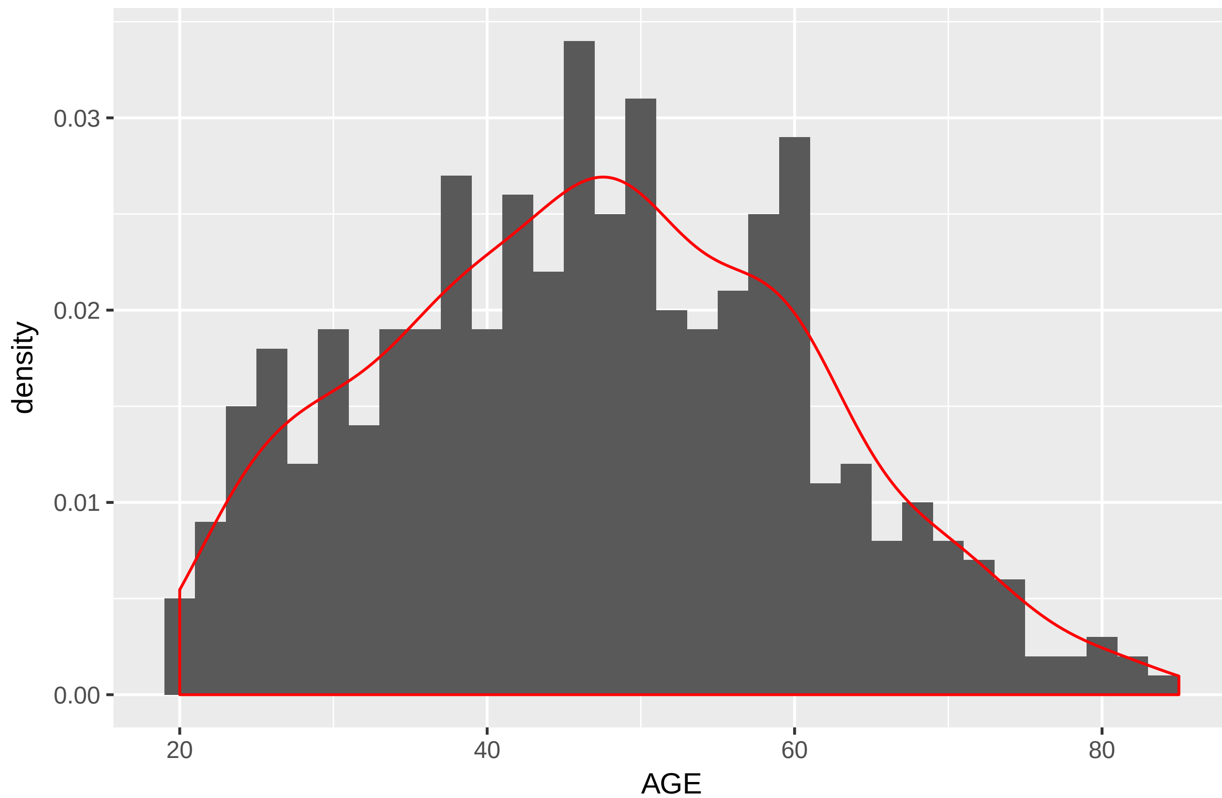


Figure 10. Histogram of SAGE

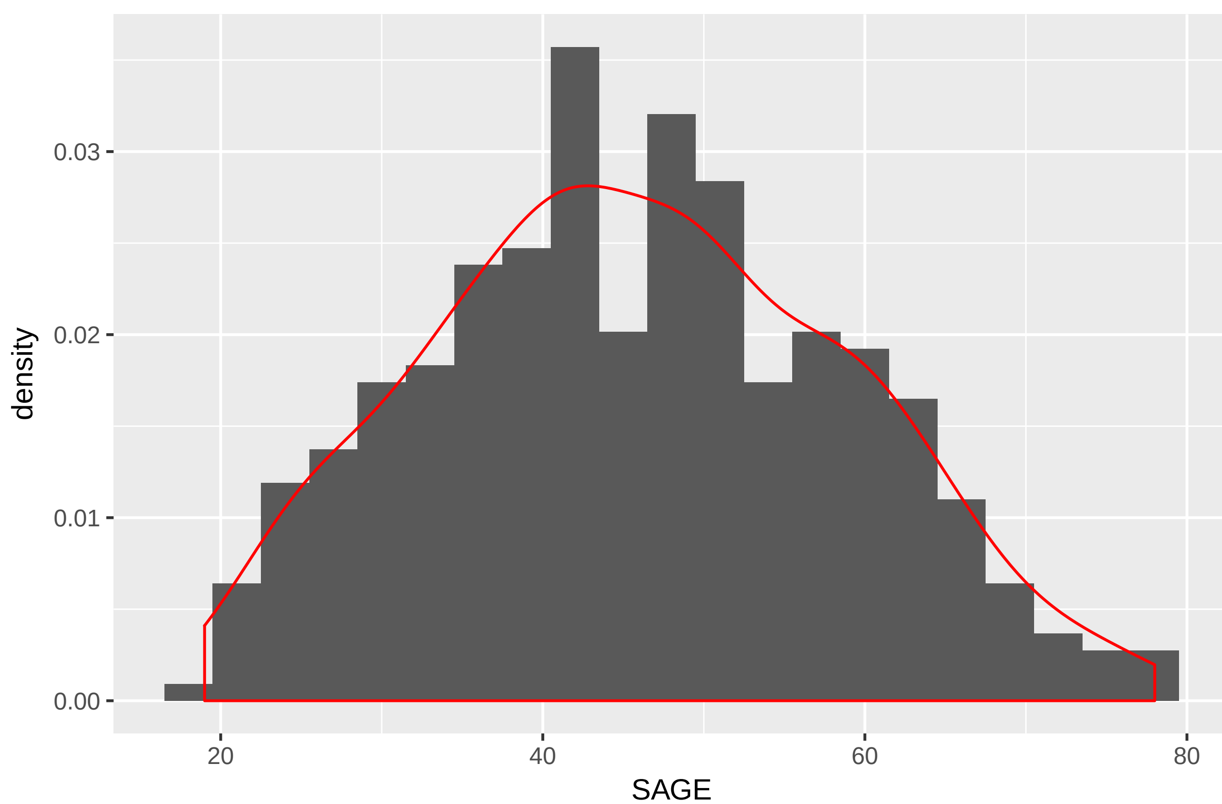


Figure 11. Histogram of EDUCATION

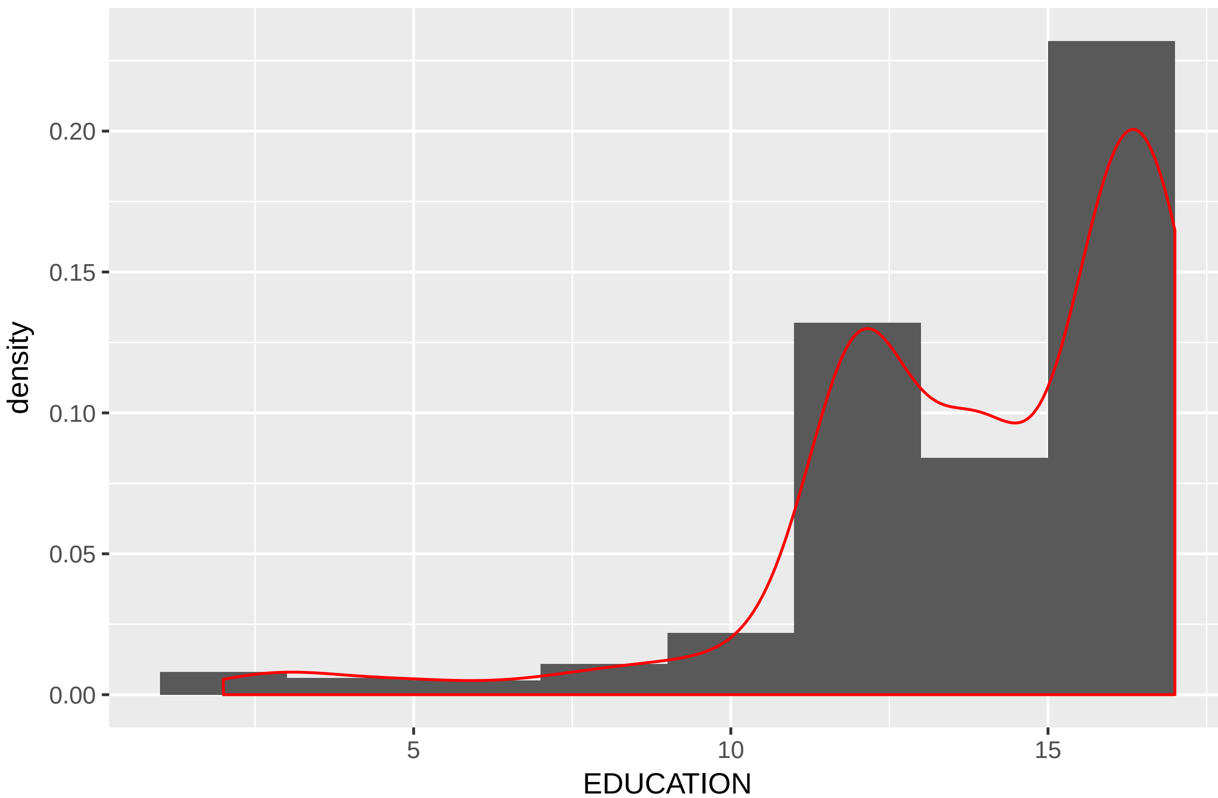


Figure 12. Histogram of SEDUCATION

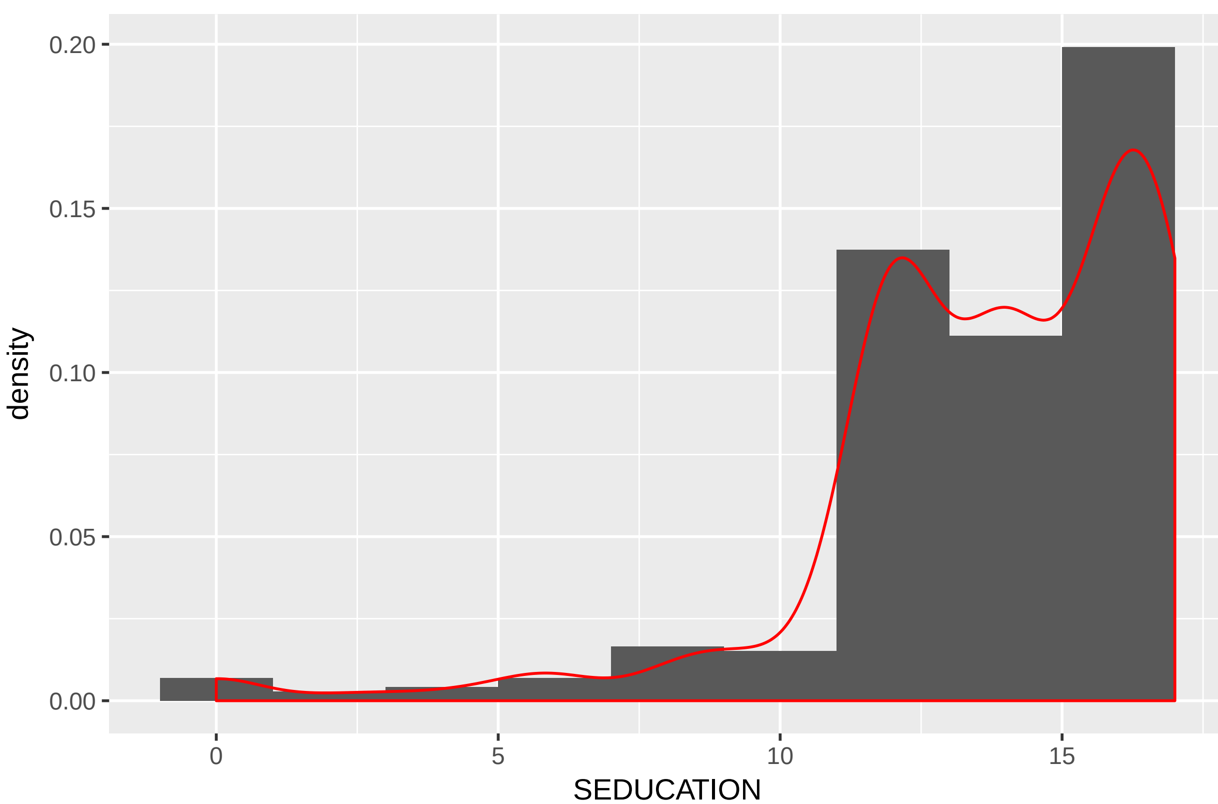


Figure 13. Histogram of AGEDIFF

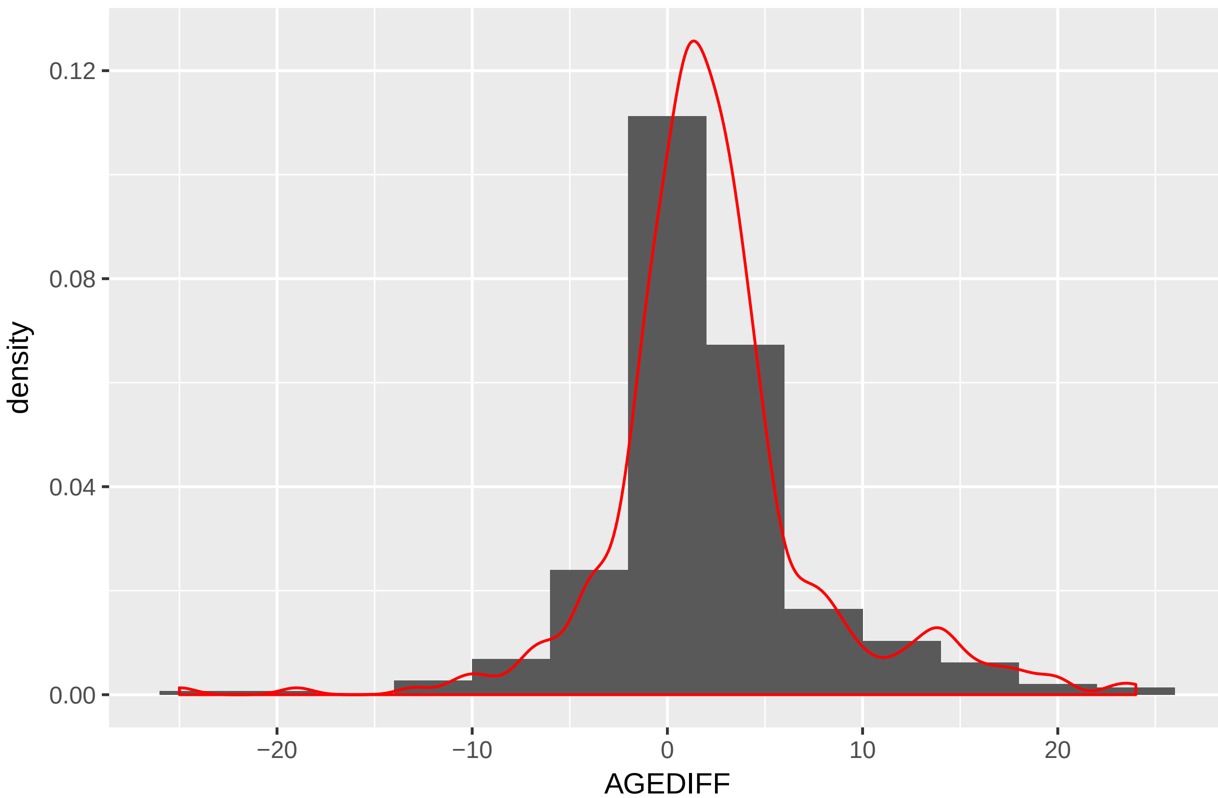


Figure 14. Histogram of EDUDIFF

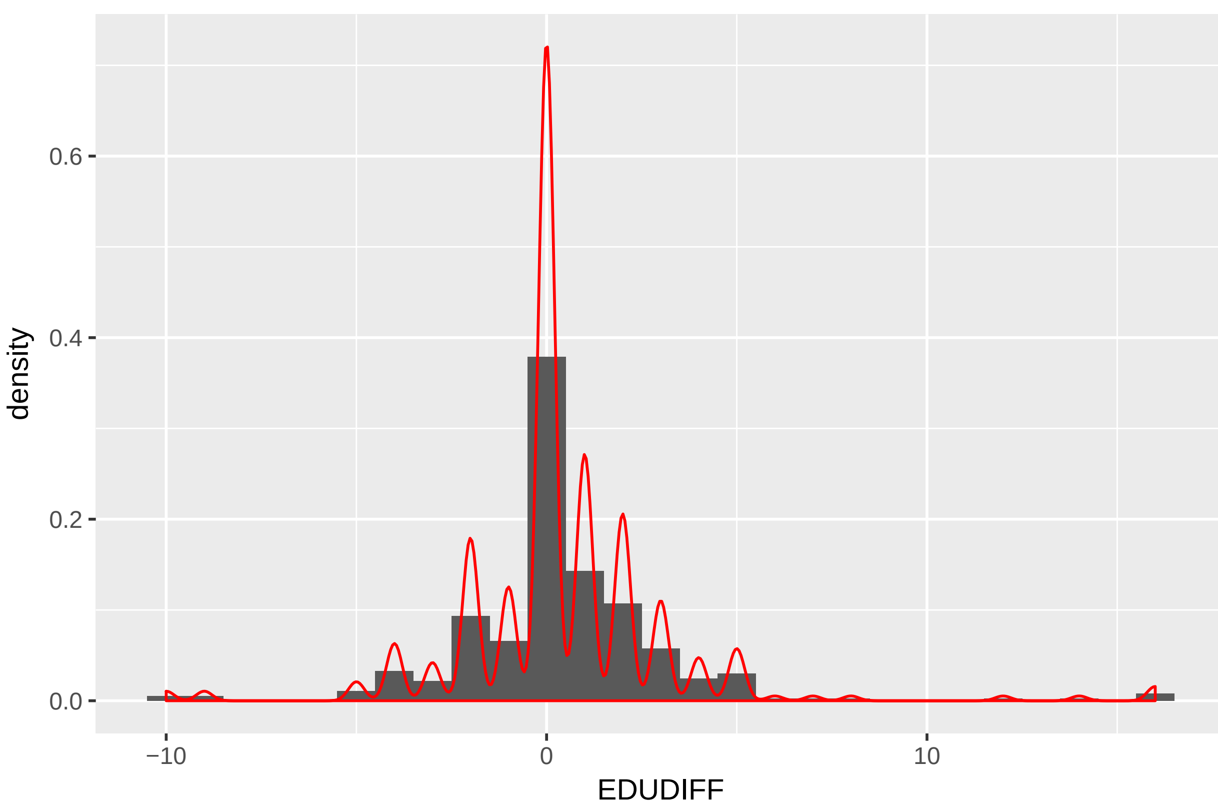


Figure 16. Histogram of INCOME

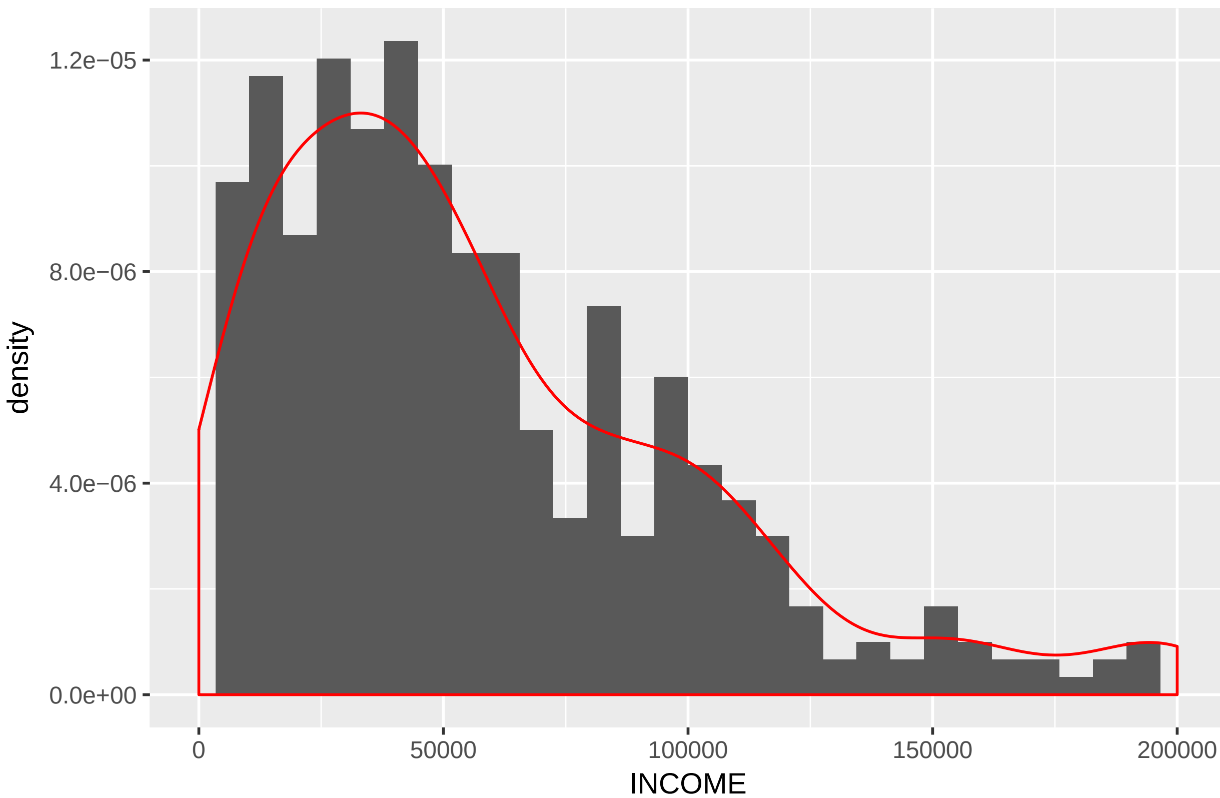


Figure 17. Histogram of TOTINCOME

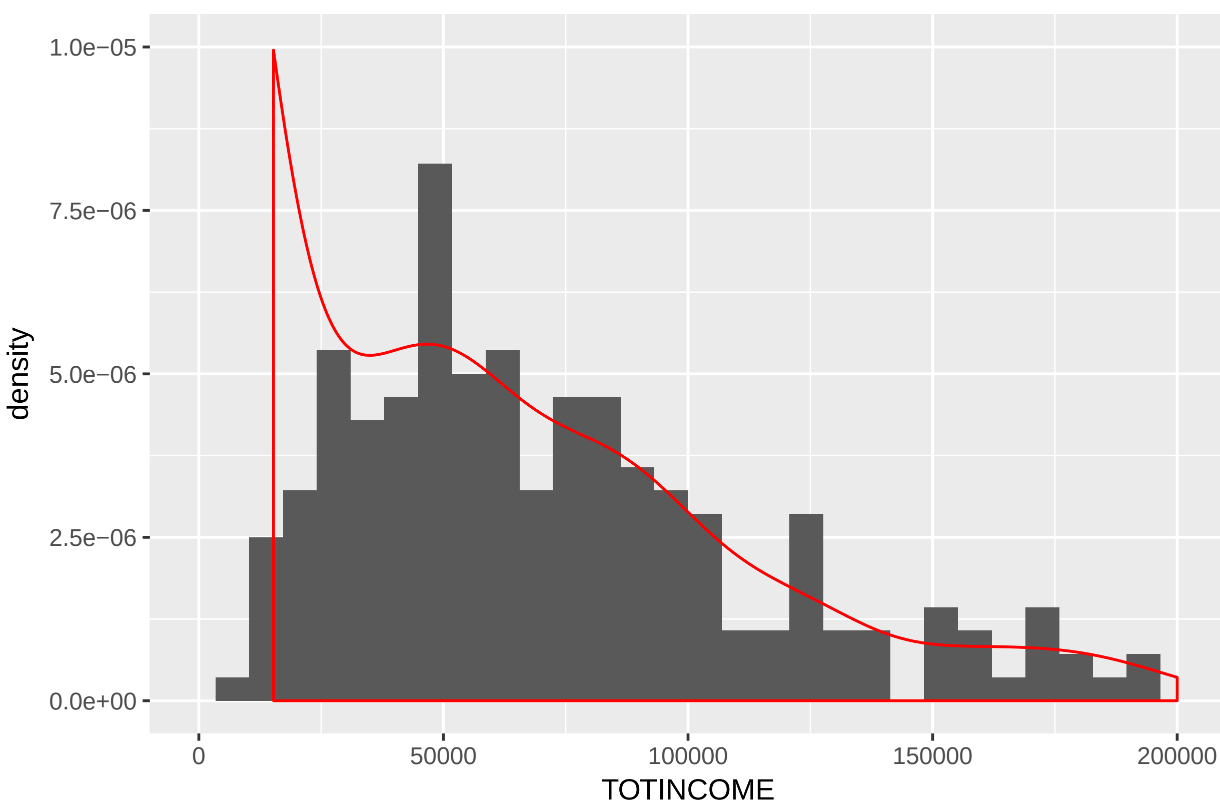


Figure 18. Histogram of CHARITY

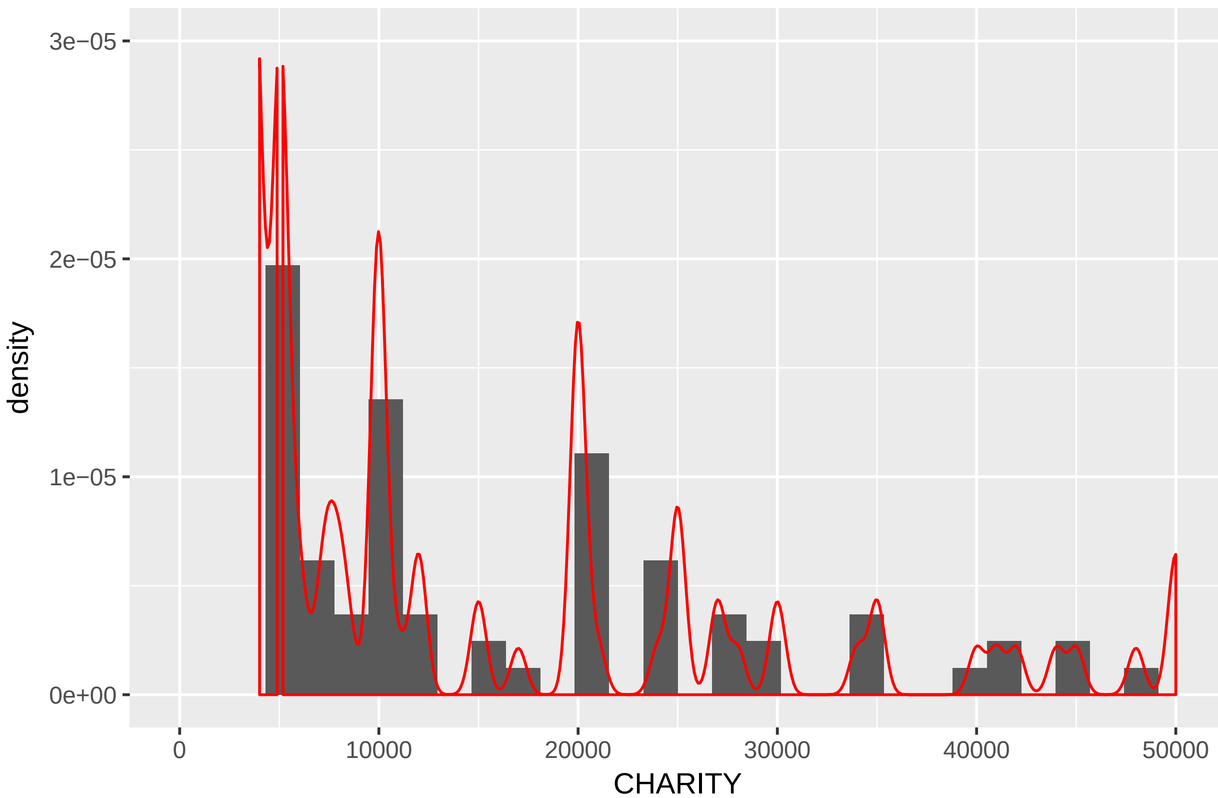


Figure 19. Histogram of FACE

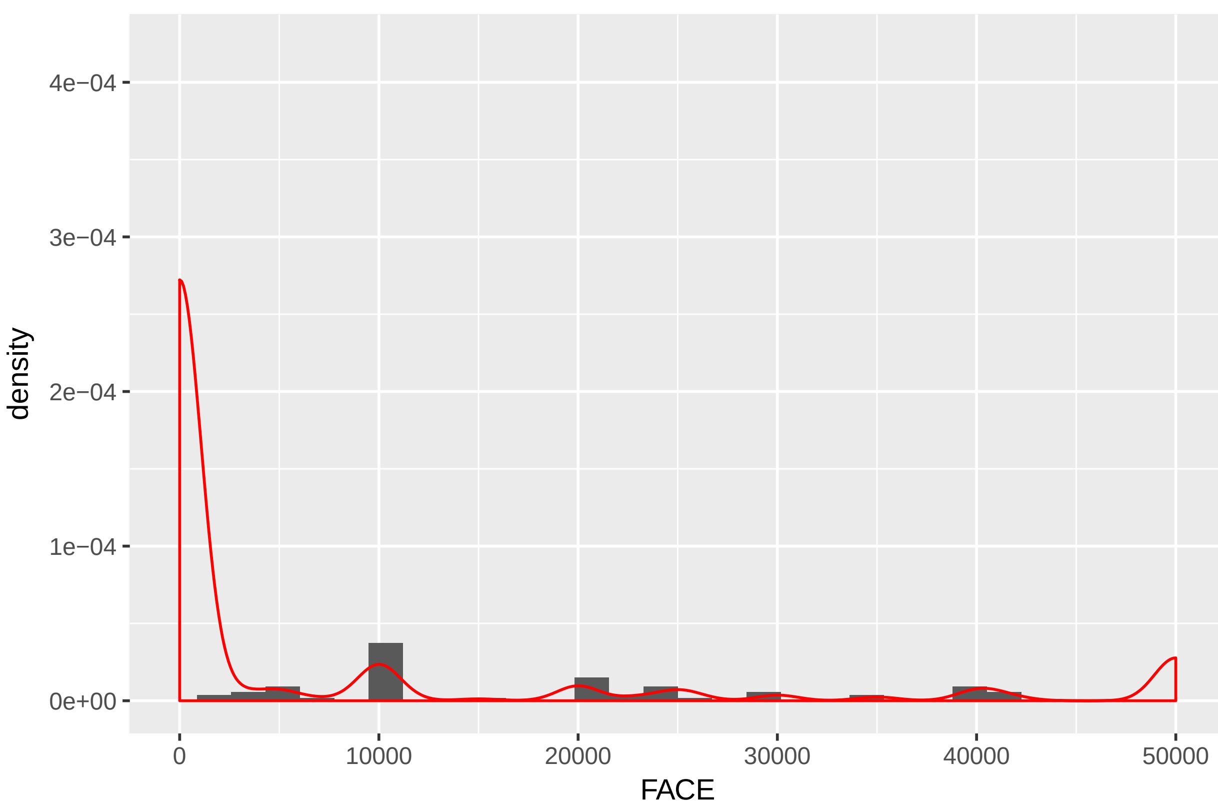


Figure 20. Histogram of FACECVLIFEPOLICIES

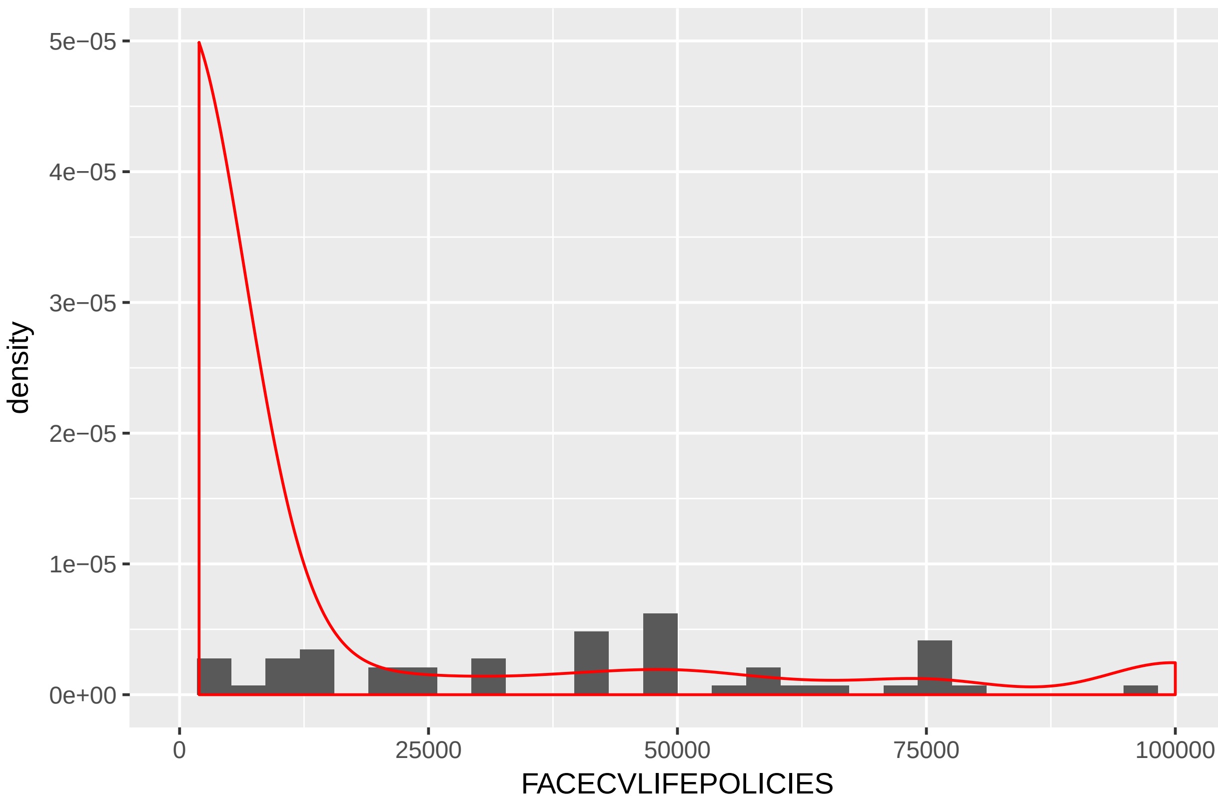


Figure 21. Histogram of CASHCVLIFEPOLICIES

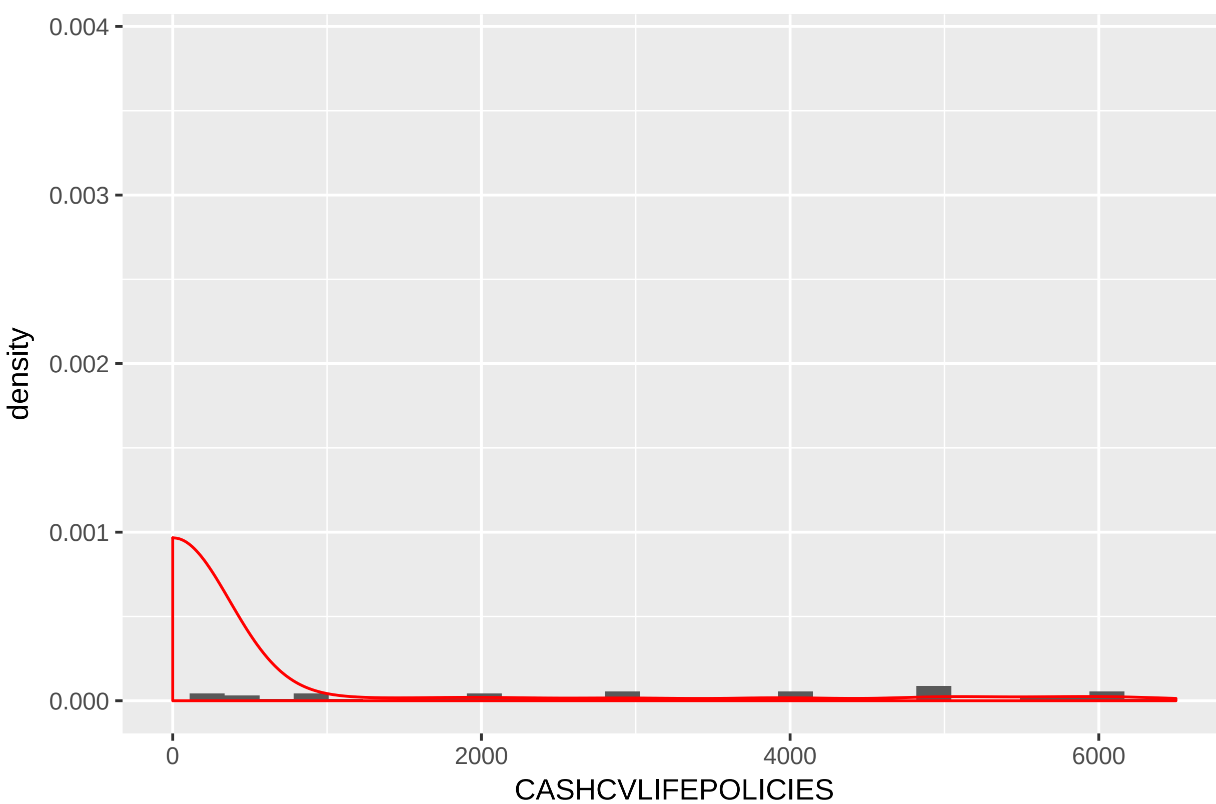


Figure 22

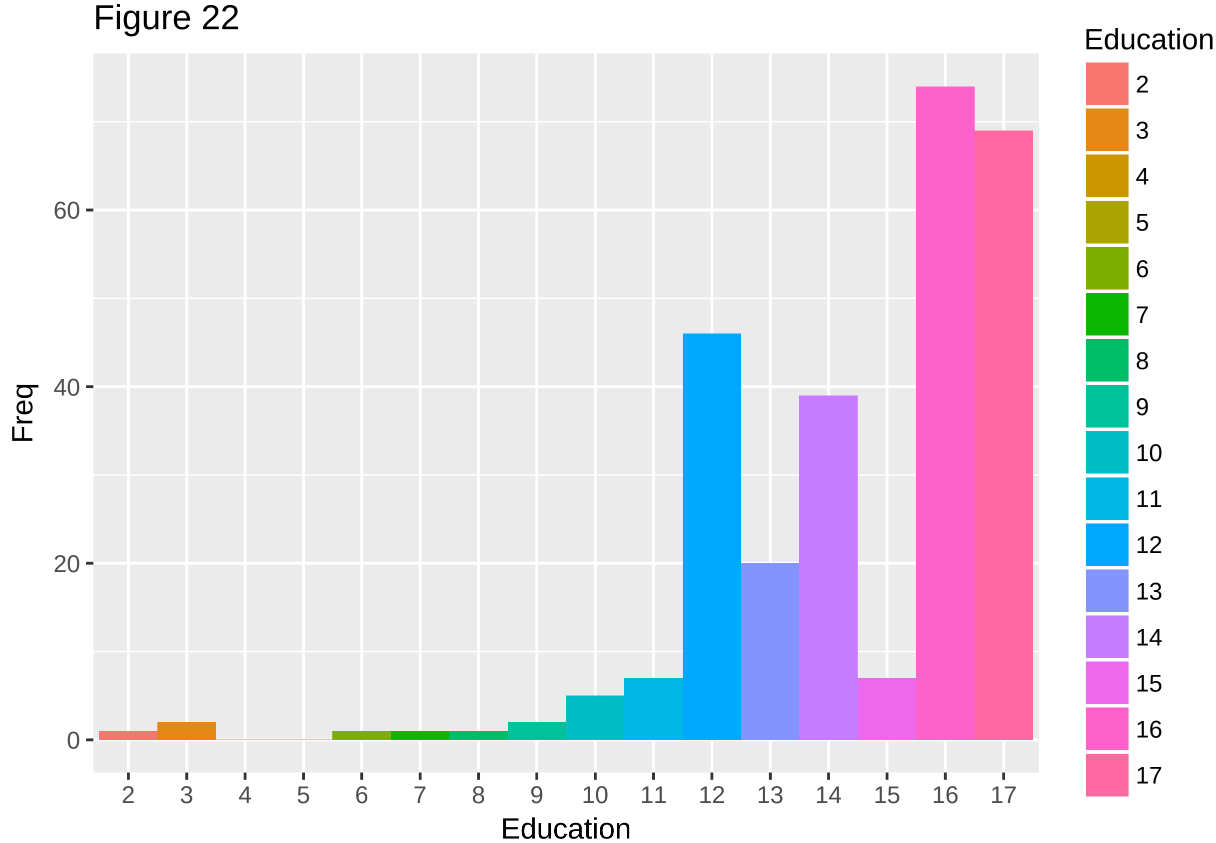


Figure 23

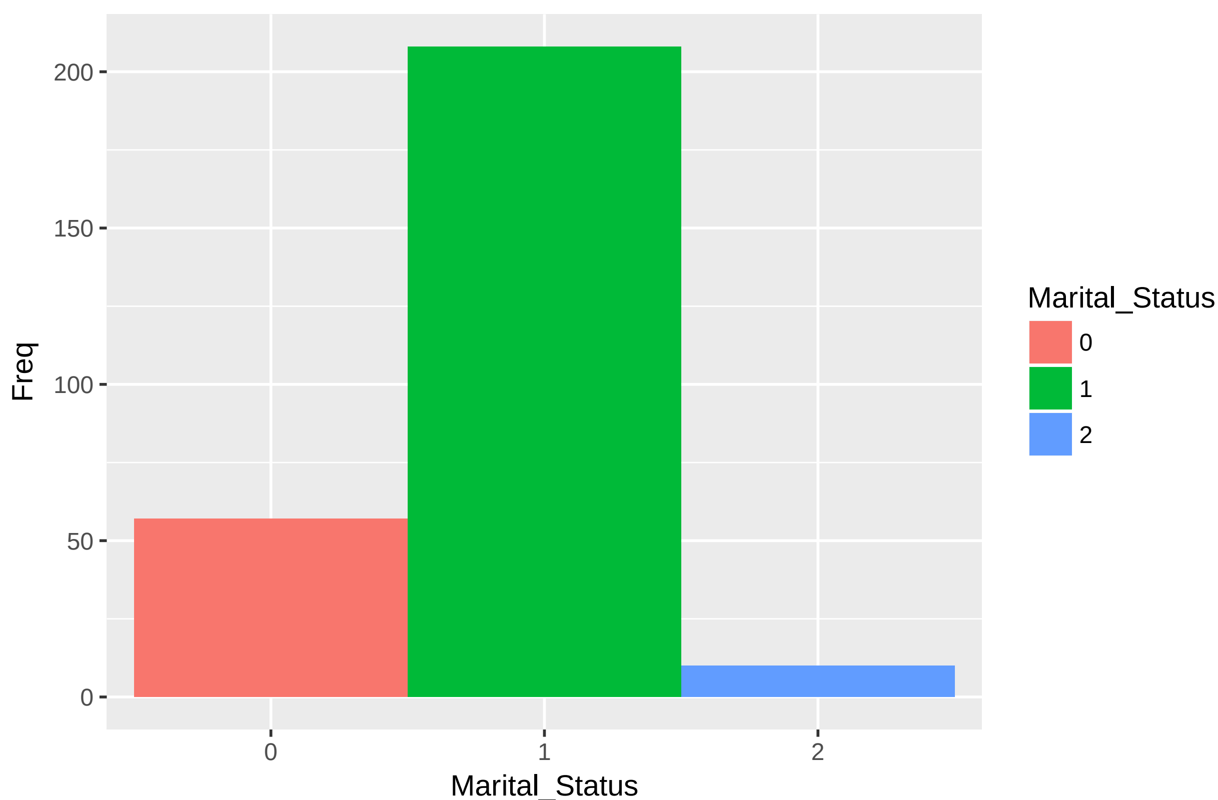


Figure 24

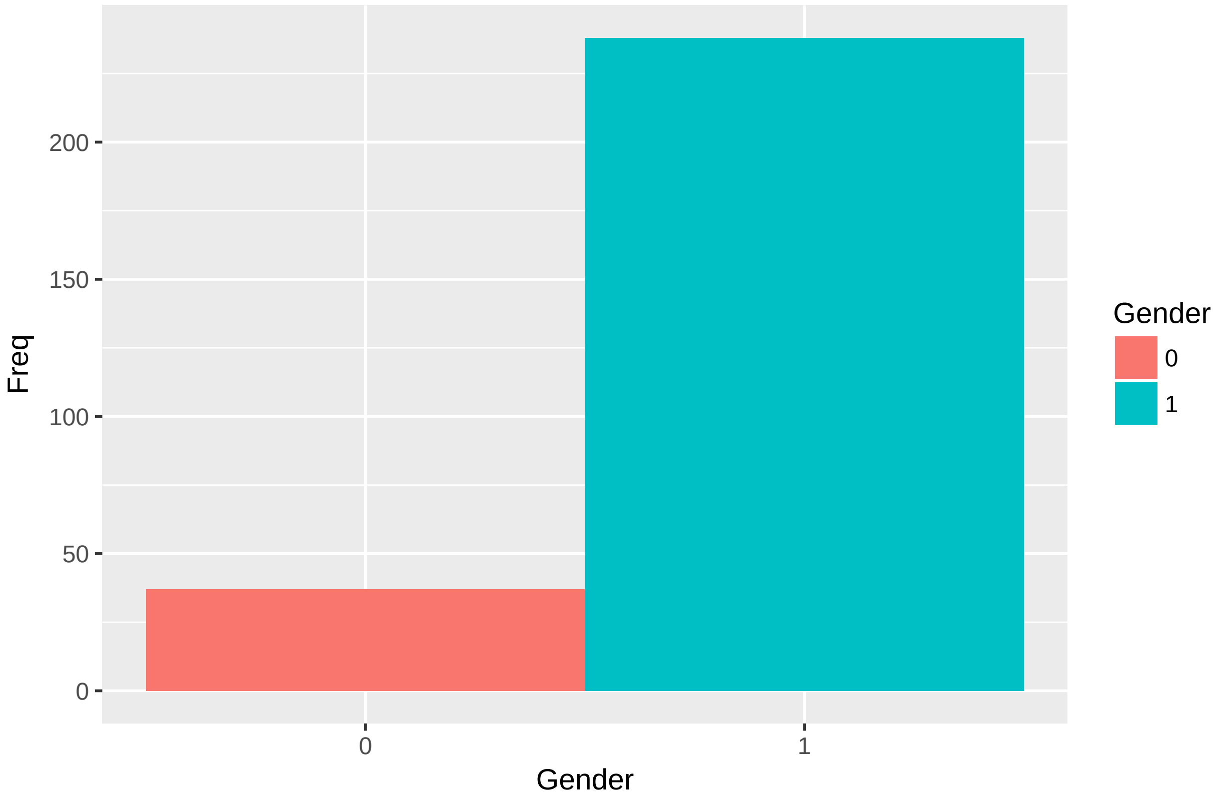
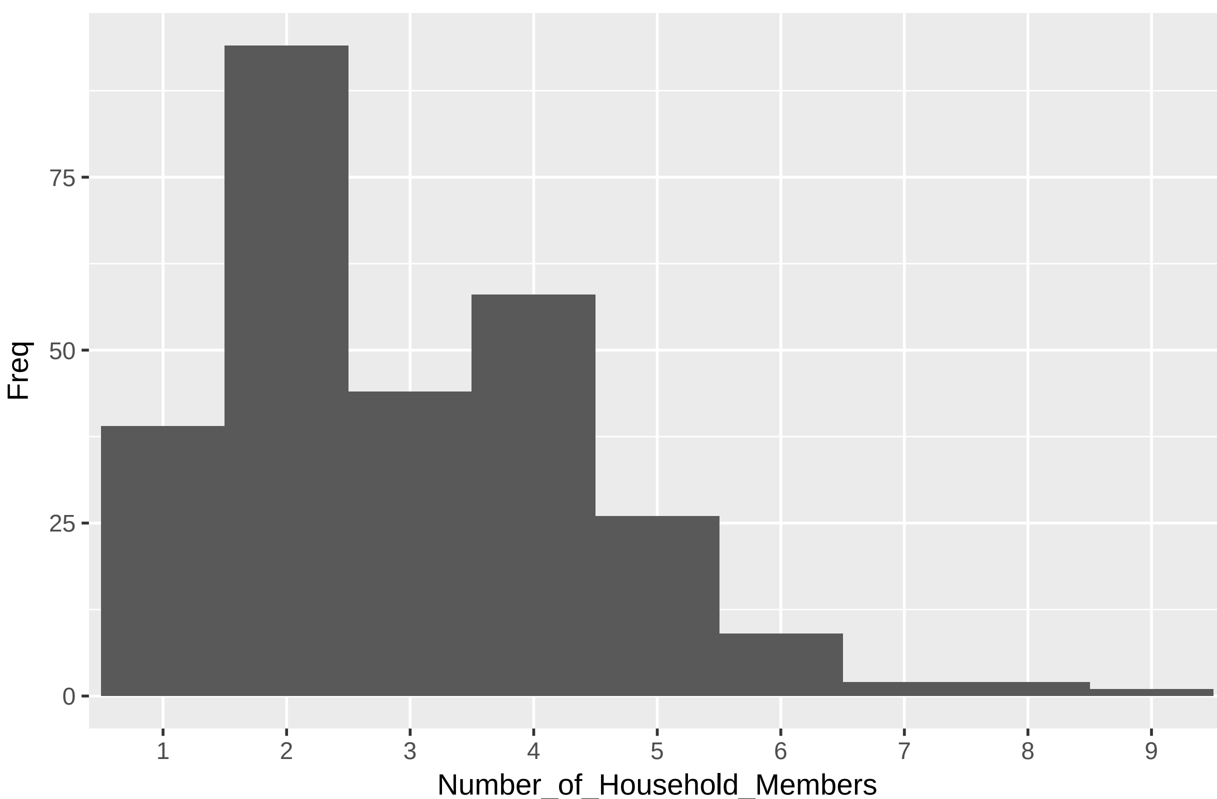
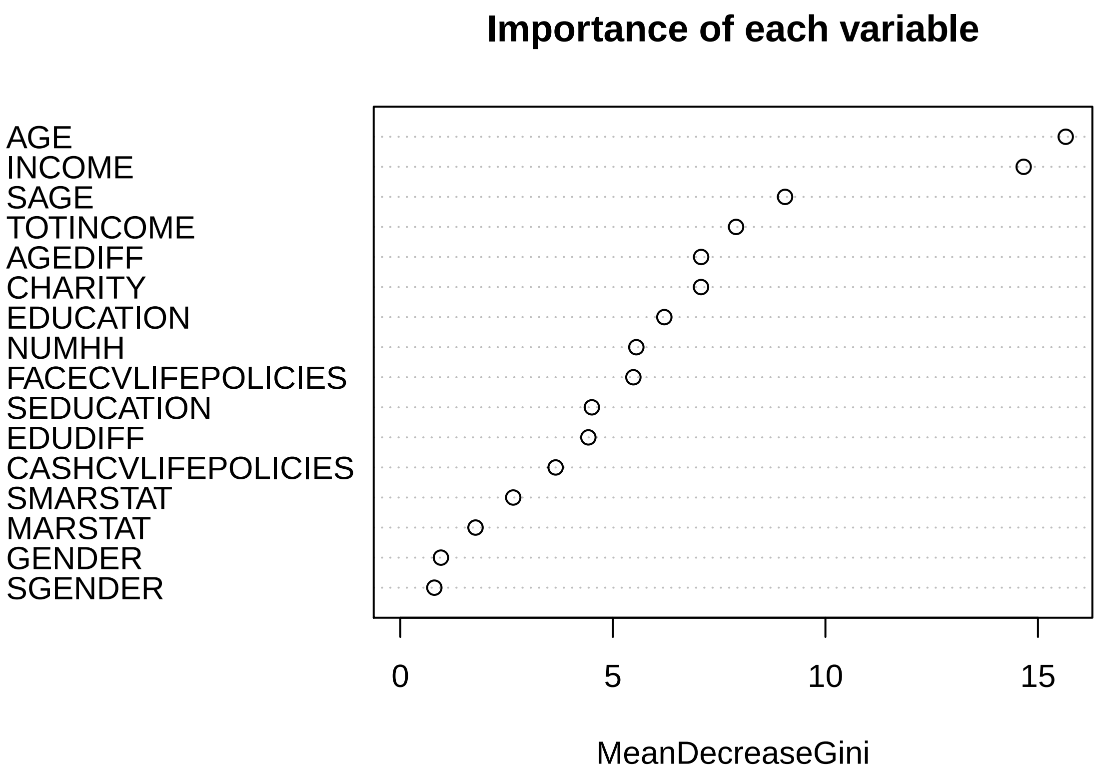


Figure 25

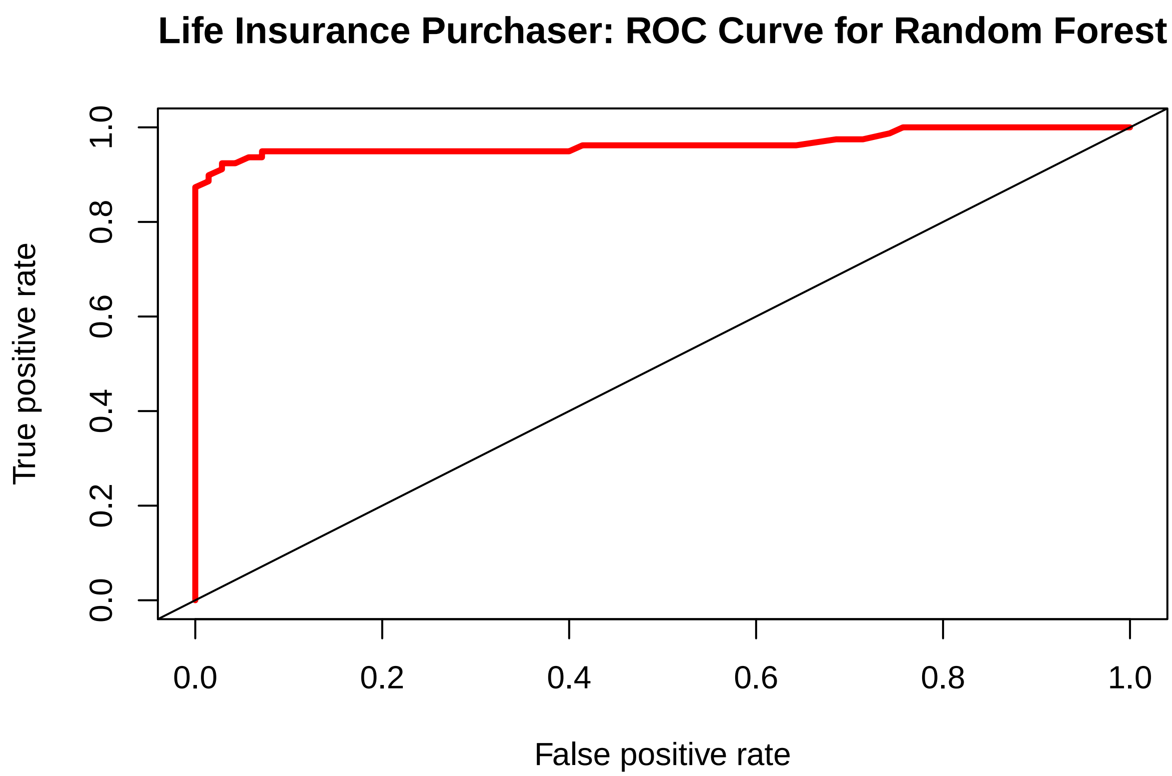


7 Data Modeling

7.1 Random Forest




```
##                    MeanDecreaseGini
## GENDER                    0.9548276
## AGE                      15.6534946
## MARSTAT                   1.7691544
## EDUCATION                 6.2102872
## SMARSTAT                  2.6557916
## SGENDER                   0.8001135
## SAGE                      9.0507765
## SEDUCATION                4.5053123
## NUMHH                     5.5516115
## INCOME                   14.6635927
## TOTINCOME                 7.8979586
## CHARITY                   7.0734983
## FACECVLIFEPOLICIES        5.4821660
## CASHCVLIFEPOLICIES        3.6531384
## AGEDIFF                   7.0760714
## EDUDIFF                   4.4233250
```




```
## [[1]]
## [1] 0.9652803

## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 67  6
##          1  3 73
##
##                Accuracy : 0.9396
##                  95% CI : (0.8884, 0.972)
##     No Information Rate : 0.5302
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.879
##  Mcnemar's Test P-Value : 0.505
##
##             Sensitivity : 0.9571
##             Specificity : 0.9241
##          Pos Pred Value : 0.9178
##          Neg Pred Value : 0.9605
##              Prevalence : 0.4698
##          Detection Rate : 0.4497
##    Detection Prevalence : 0.4899
##       Balanced Accuracy : 0.9406
##
##        'Positive' Class : 0
##
```

7.2 Modeling via Regression

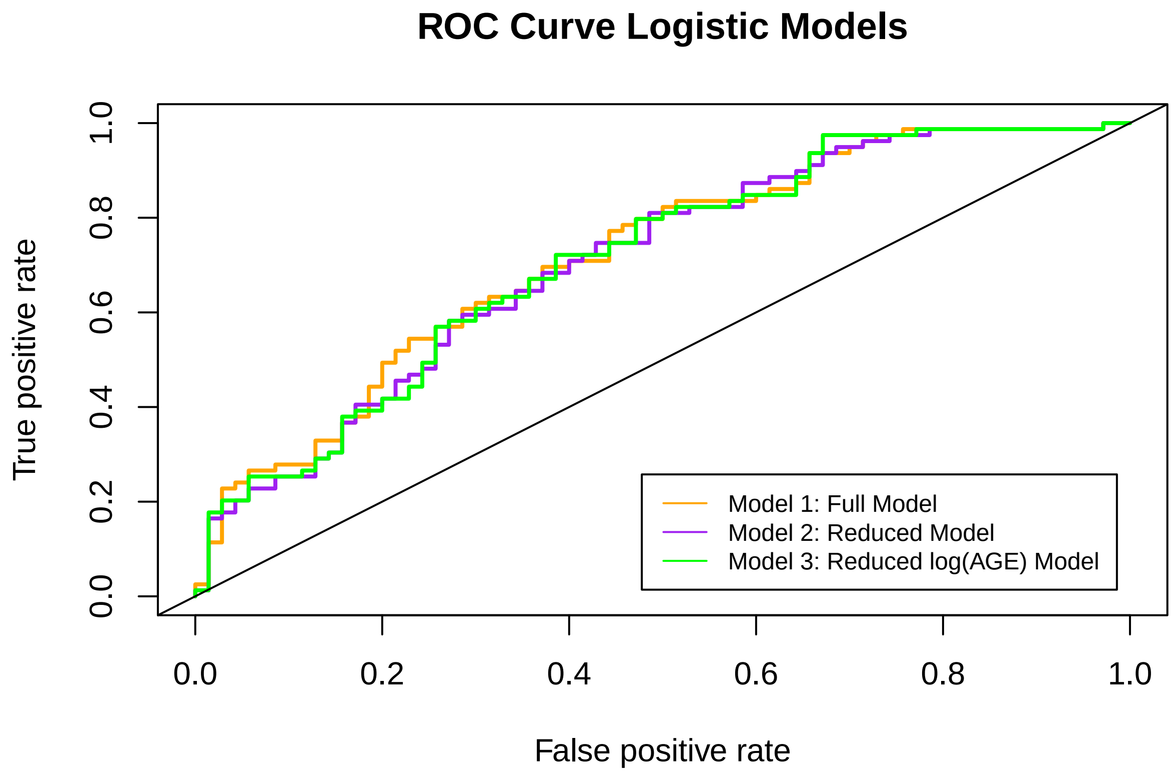
7.2.1 Logistic Regression Model and Variable Selection

```
## Model :
## Term_Flag ~ (GENDER + AGE + MARSTAT + EDUCATION + SMARSTAT +
##     SGENDER + SAGE + SEDUCATION + NUMHH + INCOME + TOTINCOME +
##     CHARITY + FACE + FACECVLIFEPOLICIES + CASHCVLIFEPOLICIES +
##     AGEDIFF + EDUDIFF) - FACE - AGEDIFF - EDUDIFF
##
## Complete :
##           (Intercept) GENDER1 AGE MARSTAT1 MARSTAT2 EDUCATION SMARSTAT1
## SMARSTAT3  0           0       0   1        1        0         -1
## SGENDER2   0           0       0   1        1        0          0
##           SMARSTAT2 SGENDER1 SAGE SEDUCATION NUMHH INCOME TOTINCOME
## SMARSTAT3 -1         0        0    0          0     0      0
## SGENDER2   0        -1        0    0          0     0      0
##           CHARITY FACECVLIFEPOLICIES CASHCVLIFEPOLICIES
## SMARSTAT3  0       0                  0
## SGENDER2   0       0                  0
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## GENDER              1.571636  1        1.253649
## AGE                 2.125245  1        1.457822
## MARSTAT            17.831144  2        2.054919
## EDUCATION           1.331632  1        1.153964
## SAGE               12.324722  1        3.510658
## SEDUCATION          7.082210  1        2.661242
## NUMHH               1.793681  1        1.339284
## INCOME              1.247007  1        1.116694
## TOTINCOME           1.312815  1        1.145781
## CHARITY             1.551207  1        1.245475
## FACECVLIFEPOLICIES  1.283566  1        1.132946
## CASHCVLIFEPOLICIES  1.177593  1        1.085169
```

```
##                        GVIF Df GVIF^(1/(2*Df))
## GENDER             1.516149  1        1.231320
## AGE                1.204209  1        1.097365
## MARSTAT            2.249333  2        1.224654
## EDUCATION          1.135499  1        1.065598
## NUMHH              1.623746  1        1.274263
## INCOME             1.251545  1        1.118725
## TOTINCOME          1.354567  1        1.163859
## CHARITY            1.475439  1        1.214677
## FACECVLIFEPOLICIES 1.200218  1        1.095545
## CASHCVLIFEPOLICIES 1.140220  1        1.067811
```

7.2.2 ROC Curves for 3 Logistic Models



7.2.3 Comparing Prediction Accuracy for Logistic Models: Confusion Matrix

```
##      true
## pred  No Yes
##   No  18   2
##   Yes 52  77
```

```
##      true
## pred  No Yes
##   No  18   2
##   Yes 52  77
```

```
##      true
## pred  No Yes
##   No  19   2
##   Yes 51  77
```

7.2.4 AUC for Logistic Models

```
## AUC for logistic regression model 1: 0.7186257
```

```
## AUC for logistic regression model 2: 0.7075949
```

```
## AUC for logistic regression model 3: 0.7106691
```

7.3 Logistic Regression

```
##
## Call:
## glm(formula = Term_Flag ~ . - FACE - AGEDIFF - EDUDIFF - SMARSTAT -
##     SGENDER - SAGE - SEDUCATION - FACECVLIFEPOLICIES - TOTINCOME -
##     NUMHH - AGE + log(AGE), family = binomial(link = "logit"),
##     data = train_full)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4358  -0.8793   0.5866   0.7631   1.4413
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -4.989e+00  2.117e+00  -2.357  0.01845 *
## GENDER1            -5.728e-01  4.746e-01  -1.207  0.22747
## MARSTAT1            1.269e+00  3.933e-01   3.226  0.00126 **
## MARSTAT2            3.144e-01  6.632e-01   0.474  0.63549
## EDUCATION           1.043e-01  5.836e-02   1.787  0.07402 .
## INCOME              1.357e-06  1.073e-06   1.265  0.20600
## CHARITY            -8.990e-06  5.653e-06  -1.590  0.11177
## CASHCVLIFEPOLICIES -6.624e-07  8.298e-07  -0.798  0.42469
## log(AGE)            1.105e+00  5.065e-01   2.182  0.02909 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 306.61  on 265  degrees of freedom
## Residual deviance: 271.79  on 257  degrees of freedom
## AIC: 289.79
##
## Number of Fisher Scoring iterations: 6
```

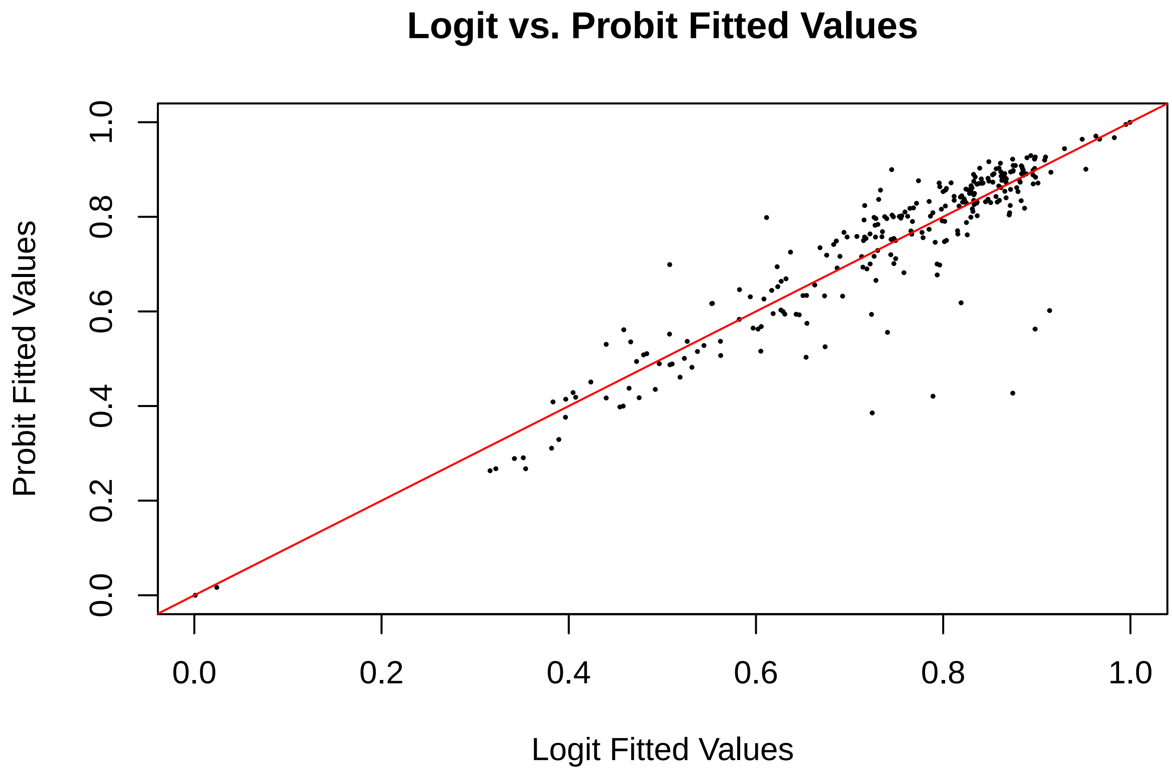
7.4 Probit Regression

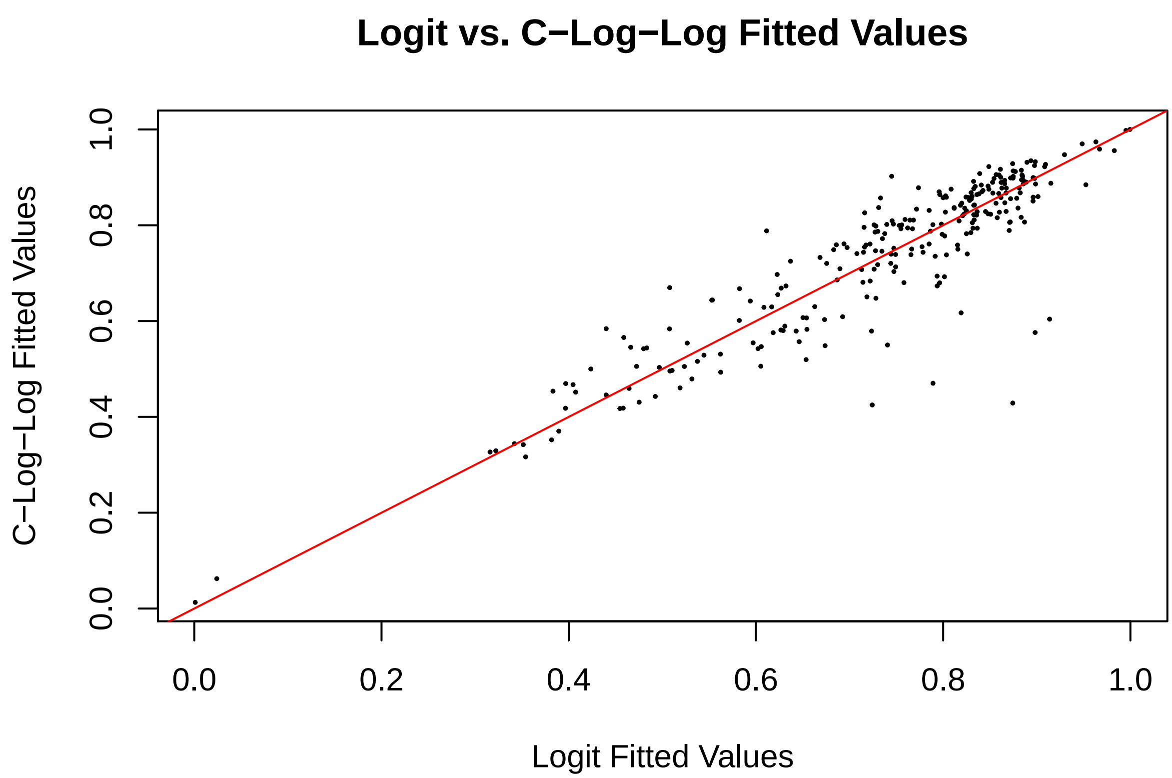
```
##
## Call:
## glm(formula = Term_Flag ~ . - FACE - SMARSTAT - SGENDER - SAGE -
##     SEDUCATION - FACECVLIFEPOLICIES - TOTINCOME - NUMHH - AGE +
##     log(AGE), family = binomial(link = "probit"), data = train_full)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5784  -0.7872   0.5463   0.7343   1.6241
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -3.365e+00  1.267e+00  -2.655 0.007941 **
## GENDER1            -5.764e-01  2.965e-01  -1.944 0.051870 .
## MARSTAT1            9.470e-01  2.482e-01   3.816 0.000136 ***
## MARSTAT2            3.752e-01  4.143e-01   0.906 0.365073
```

```
## EDUCATION           8.897e-02  3.633e-02   2.449 0.014329 *
## INCOME              6.210e-07  4.726e-07   1.314 0.188831
## CHARITY            -5.048e-06  3.427e-06  -1.473 0.140753
## CASHCVLIFEPOLICIES -4.345e-07  4.113e-07  -1.056 0.290784
## AGEDIFF            -1.134e-02  2.102e-02  -0.540 0.589458
## EDUDIFF            -9.518e-02  3.451e-02  -2.758 0.005819 **
## log(AGE)            7.011e-01  3.038e-01   2.307 0.021027 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 306.61  on 265  degrees of freedom
## Residual deviance: 264.77  on 255  degrees of freedom
## AIC: 286.77
##
## Number of Fisher Scoring iterations: 7
```

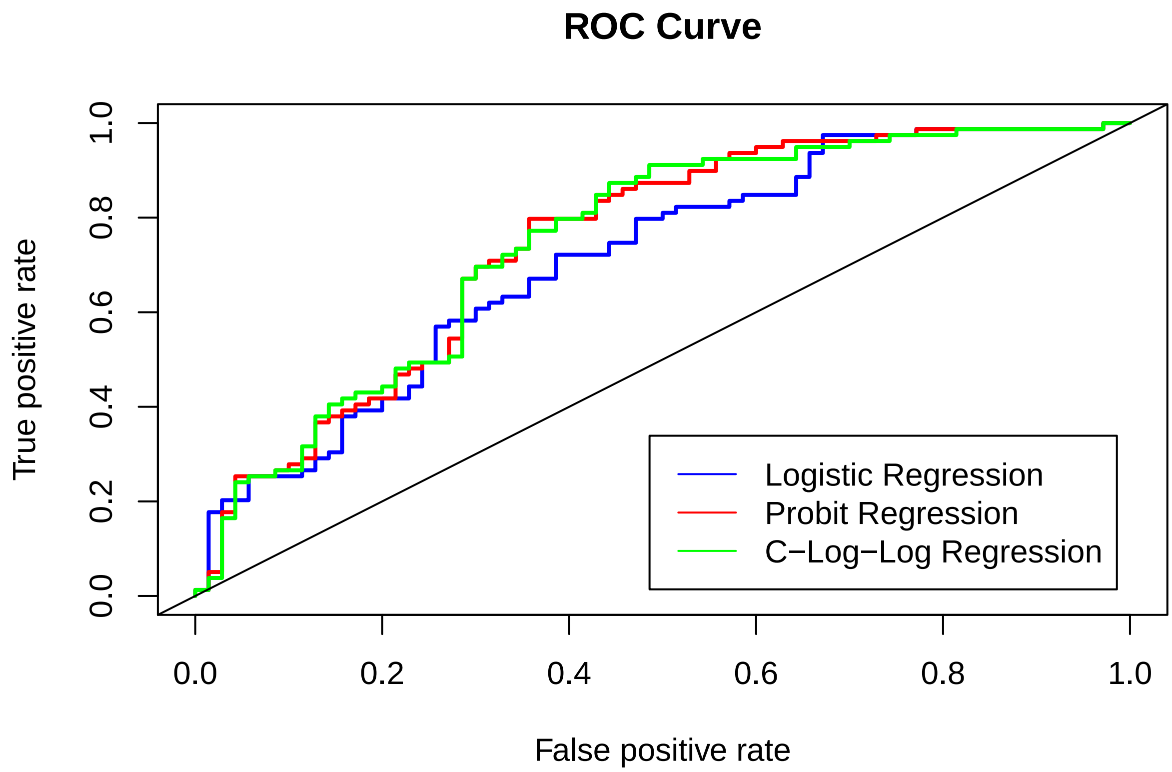
7.5 Complementary Log Log Regression

```
##
## Call:
## glm(formula = Term_Flag ~ . - FACE - SMARSTAT - SGENDER - SAGE -
##     SEDUCATION - FACECVLIFEPOLICIES - TOTINCOME - NUMHH - AGE +
##     log(AGE), family = binomial(link = "cloglog"), data = train_full)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6473  -0.8927   0.5529   0.7432   1.5169
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -3.209e+00  1.223e+00  -2.623 0.008704 **
## GENDER1            -6.159e-01  3.112e-01  -1.979 0.047785 *
## MARSTAT1            9.210e-01  2.639e-01   3.490 0.000483 ***
## MARSTAT2            3.757e-01  4.586e-01   0.819 0.412632
## EDUCATION           8.616e-02  3.496e-02   2.464 0.013727 *
## INCOME              4.638e-07  3.522e-07   1.317 0.187861
## CHARITY            -4.088e-06  3.358e-06  -1.218 0.223377
## CASHCVLIFEPOLICIES -4.587e-07  4.472e-07  -1.026 0.304982
## AGEDIFF            -8.701e-03  1.842e-02  -0.472 0.636629
## EDUDIFF            -9.176e-02  3.514e-02  -2.611 0.009022 **
## log(AGE)            5.836e-01  2.913e-01   2.003 0.045135 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 306.61  on 265  degrees of freedom
## Residual deviance: 266.33  on 255  degrees of freedom
## AIC: 288.33
##
## Number of Fisher Scoring iterations: 7
```


Logit vs. Probit Fitted Values
Probit Fitted Values
0.0
0.2
0.4
0.6
0.8
1.0
Logit Fitted Values


Logit vs. C−Log−Log Fitted Values
C−Log−Log Fitted Values
0.0
0.2
0.4
0.6
0.8
1.0
Logit Fitted Values

7.6 ROC Curve



7.7 Confusion Matrix

```
##       true
## pred  No Yes
##   No  19   2
##   Yes 51  77
```

```
##       true
## pred  No Yes
##   No  20   3
##   Yes 50  76
```

```
##       true
## pred  No Yes
##   No  19   3
##   Yes 51  76
```

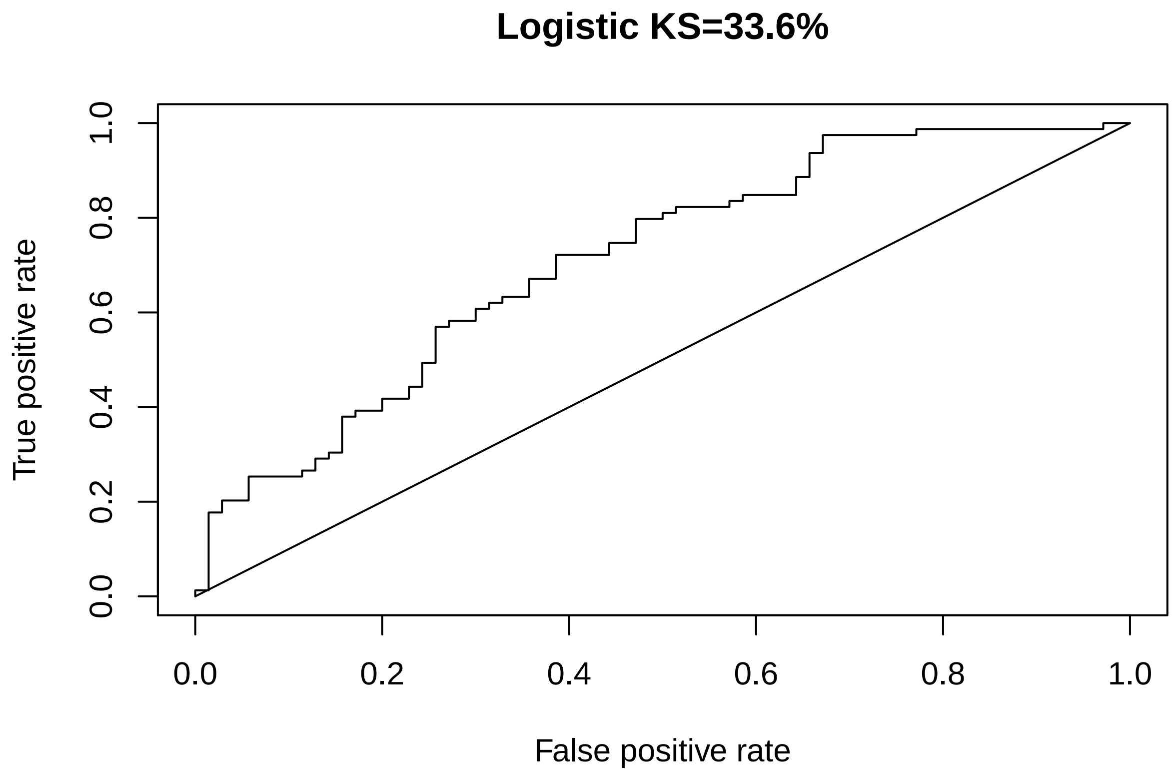
7.8 AUC

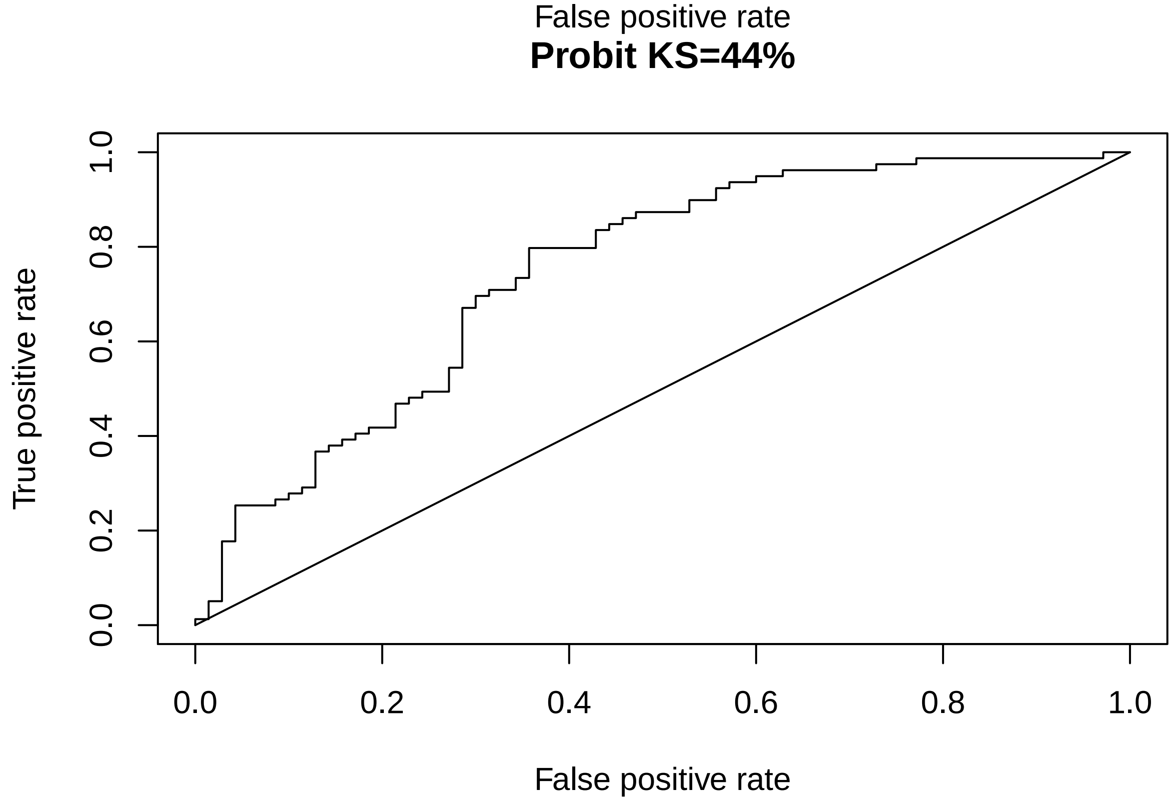
```
## AUC for logistic regression: 0.7106691
```

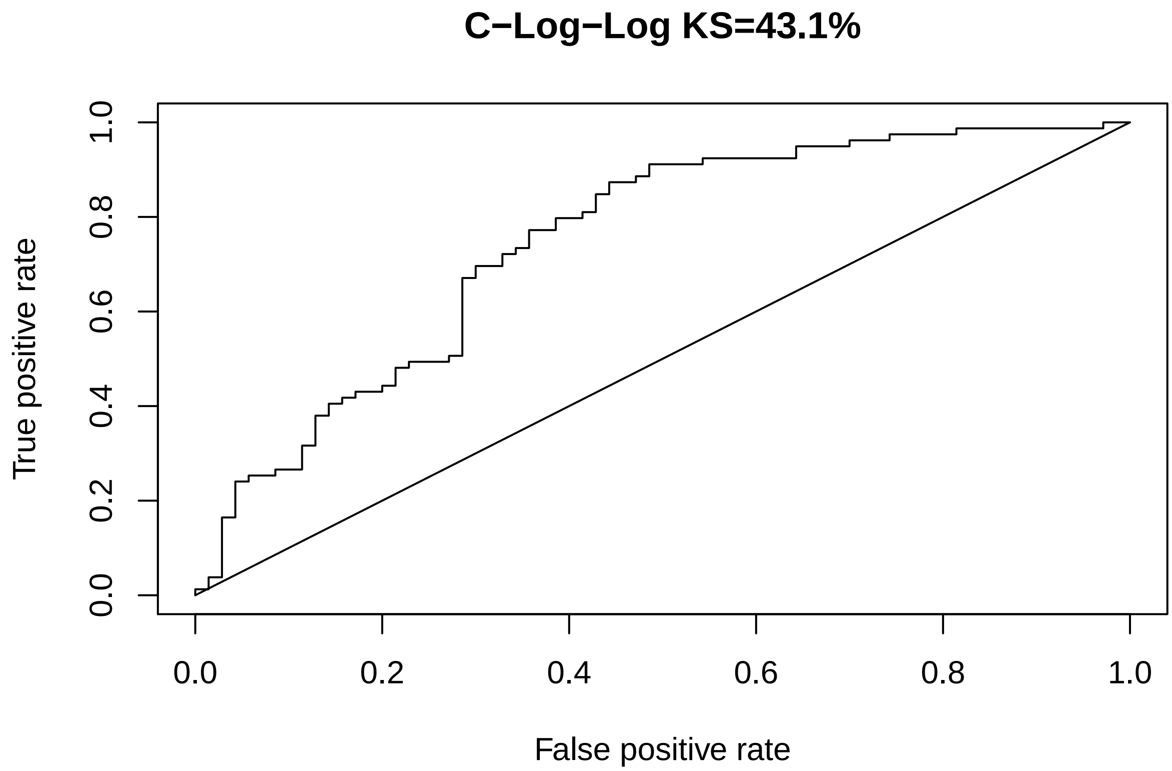
```
## AUC for probit regression: 0.7459313
```

```
## AUC for c-log-log regression: 0.7466546
```

7.9 KS Statistics





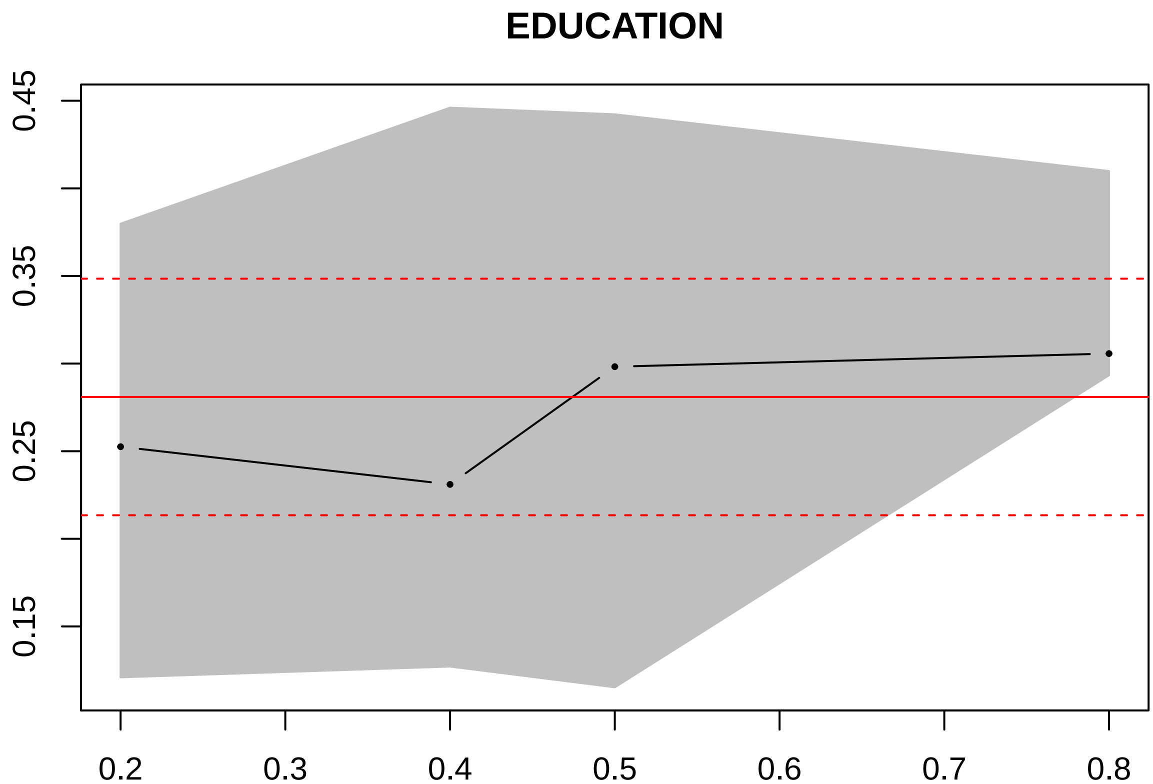


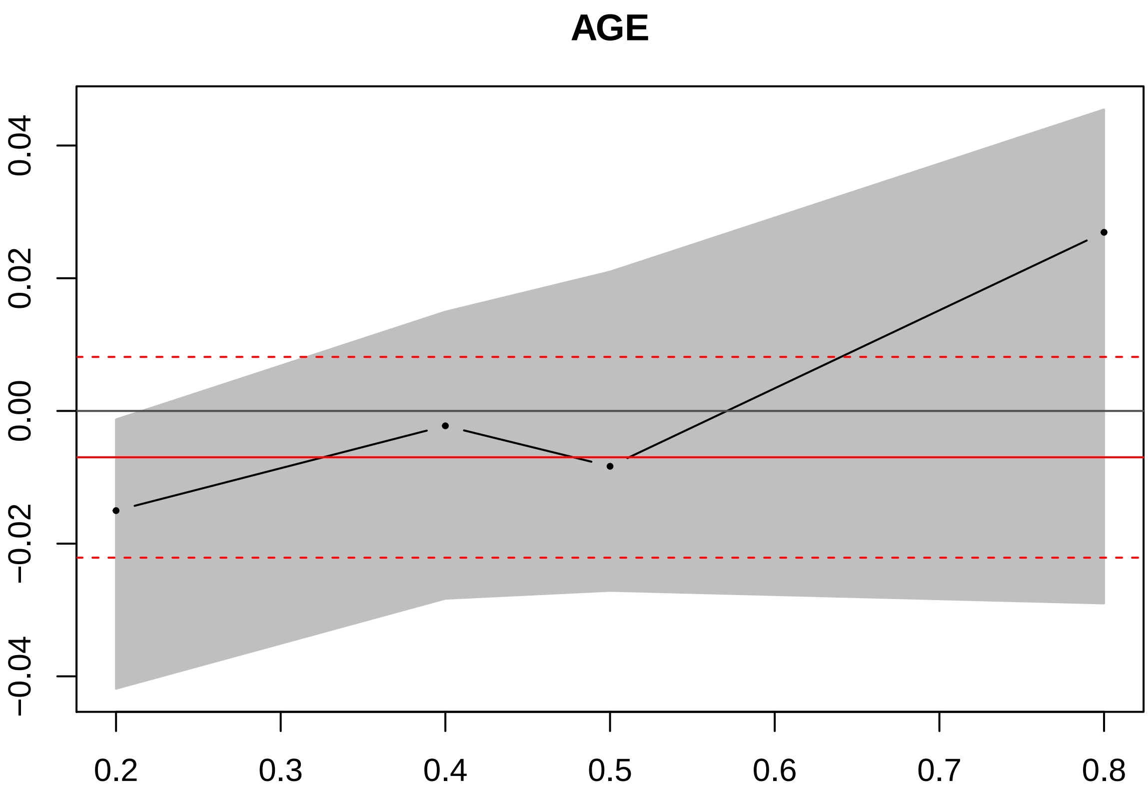
7.10 Quantile Regression

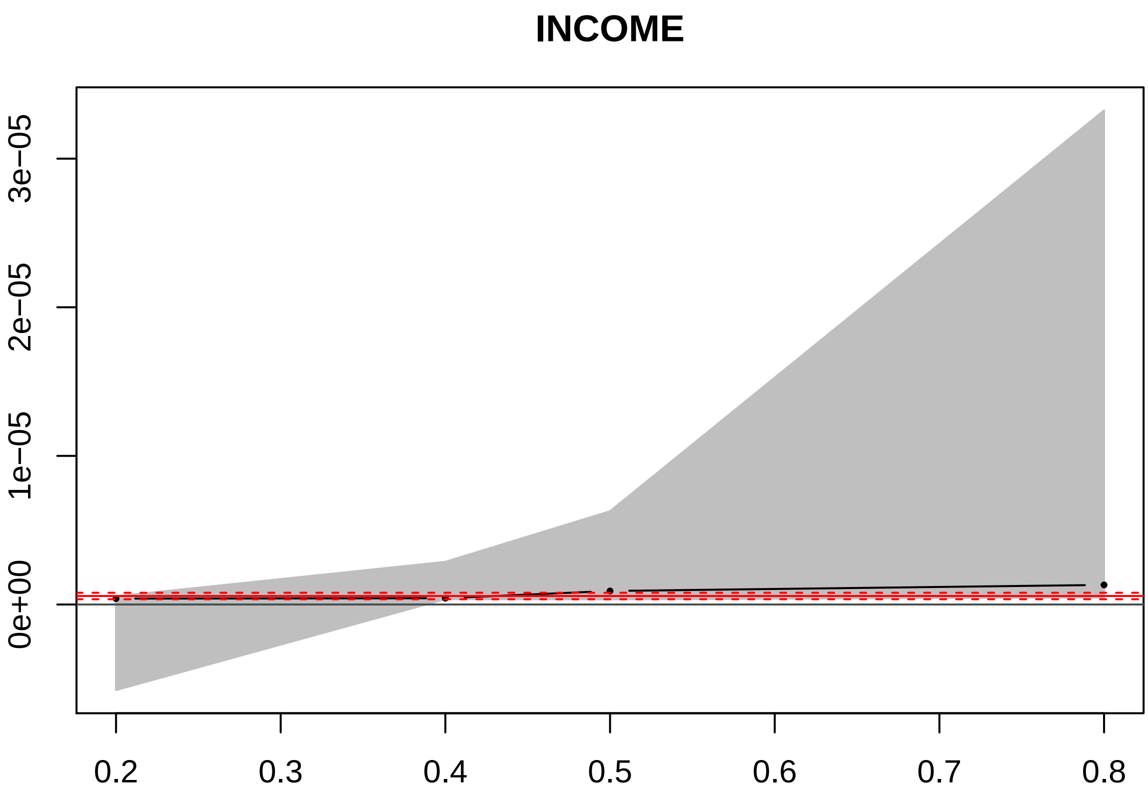
```
##
## Call: rq(formula = log_face ~ EDUCATION + AGE + INCOME + TOTINCOME +
##     NUMHH + MARSTAT + SMARSTAT + CHARITY + FACECVLIFEPOLICIES +
##     CASHCVLIFEPOLICIES, tau = taus, data = termlf)
##
## tau: [1] 0.2
##
## Coefficients:
##                      coefficients   lower bd       upper bd
## (Intercept)            6.347980e+00   4.722220e+00   8.373810e+00
## EDUCATION              2.537300e-01   7.762000e-02   3.300700e-01
## AGE                   -1.057000e-02  -3.358000e-02   2.920000e-03
## INCOME                 0.000000e+00   0.000000e+00   0.000000e+00
## TOTINCOME              0.000000e+00  -1.600000e-04   0.000000e+00
## NUMHH                  3.105100e-01   8.840000e-02   4.103900e-01
## MARSTAT                2.013000e-01  -1.038880e+00   9.989800e-01
## SMARSTAT               5.230000e-02  -2.249100e-01   8.373400e-01
## CHARITY                1.000000e-05   0.000000e+00   2.000000e-05
## FACECVLIFEPOLICIES     0.000000e+00 -1.797693e+308   0.000000e+00
## CASHCVLIFEPOLICIES     0.000000e+00 -1.797693e+308   0.000000e+00
##
## Call: rq(formula = log_face ~ EDUCATION + AGE + INCOME + TOTINCOME +
##     NUMHH + MARSTAT + SMARSTAT + CHARITY + FACECVLIFEPOLICIES +
##     CASHCVLIFEPOLICIES, tau = taus, data = termlf)
##
## tau: [1] 0.4
```

```
##
## Coefficients:
##                     coefficients lower bd upper bd
## (Intercept)          6.31977      5.04366  8.96439
## EDUCATION            0.29810      0.14934  0.39868
## AGE                 -0.01200     -0.02722  0.00443
## INCOME               0.00000      0.00000  0.00000
## TOTINCOME            0.00000      0.00000  0.00000
## NUMHH                0.29364      0.19994  0.51475
## MARSTAT              0.33303     -0.40848  0.82321
## SMARSTAT             0.26902     -0.06730  0.69100
## CHARITY              0.00001      0.00000  0.00003
## FACECVLIFEPOLICIES  0.00000      0.00000  0.00000
## CASHCVLIFEPOLICIES  0.00000     -0.00001  0.00000
##
## Call: rq(formula = log_face ~ EDUCATION + AGE + INCOME + TOTINCOME +
##     NUMHH + MARSTAT + SMARSTAT + CHARITY + FACECVLIFEPOLICIES +
##     CASHCVLIFEPOLICIES, tau = taus, data = termlf)
##
## tau: [1] 0.5
##
## Coefficients:
##                     coefficients lower bd upper bd
## (Intercept)          6.61877      4.45774  8.74480
## EDUCATION            0.29764      0.18018  0.44788
## AGE                 -0.00945     -0.01997  0.00589
## INCOME               0.00000      0.00000  0.00000
## TOTINCOME            0.00000      0.00000  0.00000
## NUMHH                0.29452      0.19703  0.46524
## MARSTAT              0.23774     -0.26143  0.88800
## SMARSTAT             0.27090     -0.11543  0.59187
## CHARITY              0.00001      0.00001  0.00003
## FACECVLIFEPOLICIES  0.00000      0.00000  0.00000
## CASHCVLIFEPOLICIES  0.00000      0.00000  0.00000
##
## Call: rq(formula = log_face ~ EDUCATION + AGE + INCOME + TOTINCOME +
##     NUMHH + MARSTAT + SMARSTAT + CHARITY + FACECVLIFEPOLICIES +
##     CASHCVLIFEPOLICIES, tau = taus, data = termlf)
##
## tau: [1] 0.8
##
## Coefficients:
##                     coefficients   lower bd       upper bd
## (Intercept)           8.506810e+00   6.451620e+00   9.624660e+00
## EDUCATION             2.038500e-01   1.659200e-01   2.750500e-01
## AGE                   5.000000e-03  -3.662000e-02   3.034000e-02
## INCOME                0.000000e+00   0.000000e+00   0.000000e+00
## TOTINCOME             0.000000e+00   0.000000e+00   1.500000e-04
## NUMHH                 2.738200e-01   1.288800e-01   5.872700e-01
## MARSTAT              -2.007400e-01  -7.480400e-01   6.977500e-01
## SMARSTAT              5.199600e-01   6.042000e-02   8.634400e-01
## CHARITY               2.000000e-05   0.000000e+00   4.000000e-05
## FACECVLIFEPOLICIES   0.000000e+00   0.000000e+00  1.797693e+308
## CASHCVLIFEPOLICIES   0.000000e+00   0.000000e+00  1.797693e+308
```

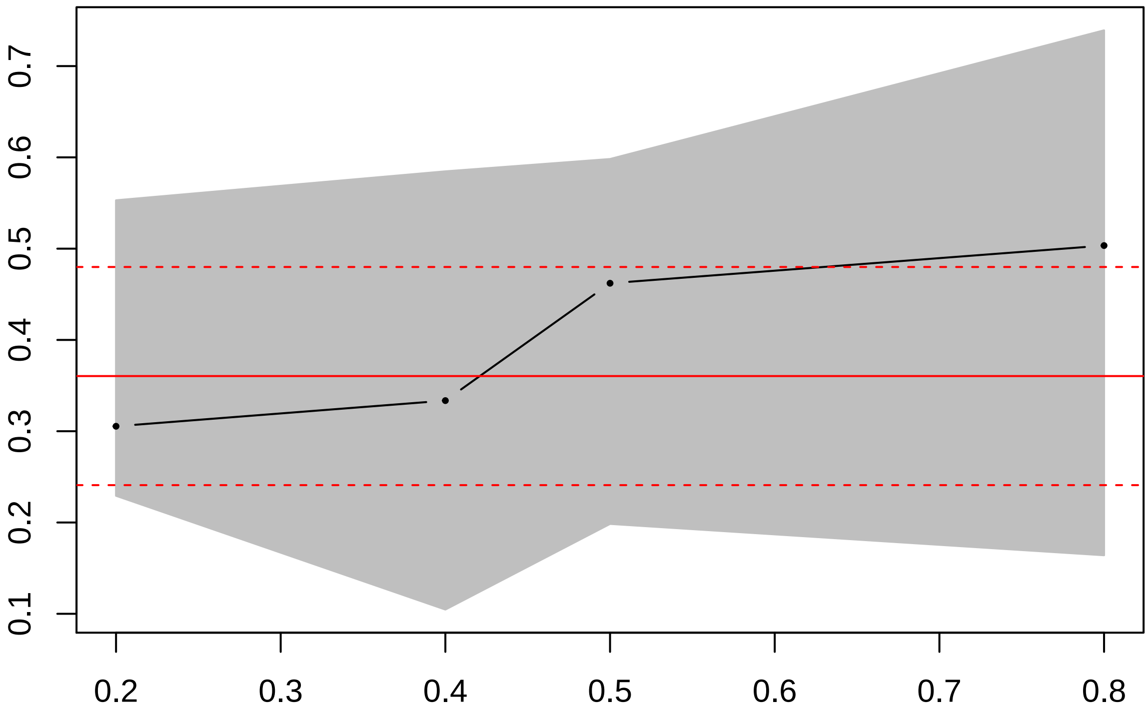
```
##                         tau= 0.2      tau= 0.4      tau= 0.5      tau= 0.8
## (Intercept)          6.347978e+00  6.319770e+00  6.618770e+00  8.506810e+00
## EDUCATION            2.537325e-01  2.981028e-01  2.976444e-01  2.038532e-01
## AGE                 -1.057002e-02 -1.200179e-02 -9.451786e-03  4.995986e-03
## INCOME              -6.660493e-07 -2.800499e-07 -3.194331e-07  8.081486e-08
## TOTINCOME            4.955405e-08  3.041856e-08  2.825978e-08  2.521381e-08
## NUMHH                3.105056e-01  2.936401e-01  2.945220e-01  2.738205e-01
## MARSTAT              2.012987e-01  3.330250e-01  2.377353e-01 -2.007374e-01
## SMARSTAT             5.229798e-02  2.690180e-01  2.708972e-01  5.199562e-01
## CHARITY              1.402465e-05  1.448509e-05  1.492271e-05  2.076850e-05
## FACECVLIFEPOLICIES   6.960046e-08  4.874074e-08  4.139950e-08  9.484065e-09
## CASHCVLIFEPOLICIES   2.783608e-06  1.544451e-06  1.321255e-06  3.425062e-07
```



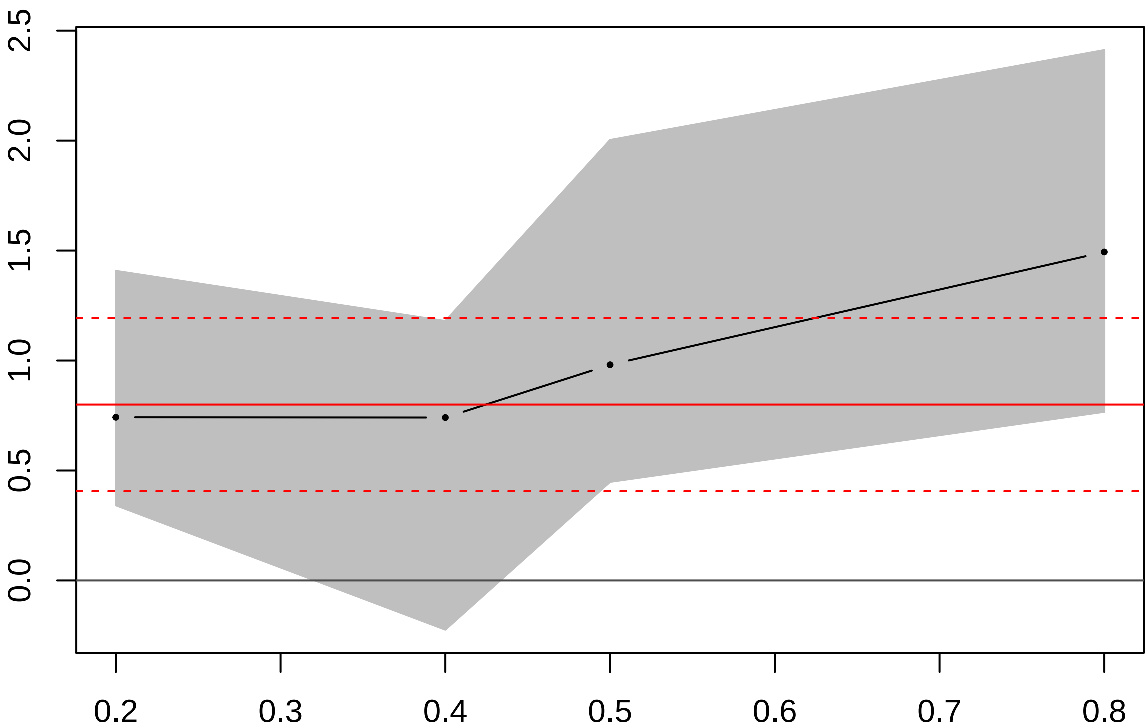

AGE
0.04
0.02
0.00
−0.02
−0.04
0.2
0.3
0.4
0.5
0.6
0.7
0.8

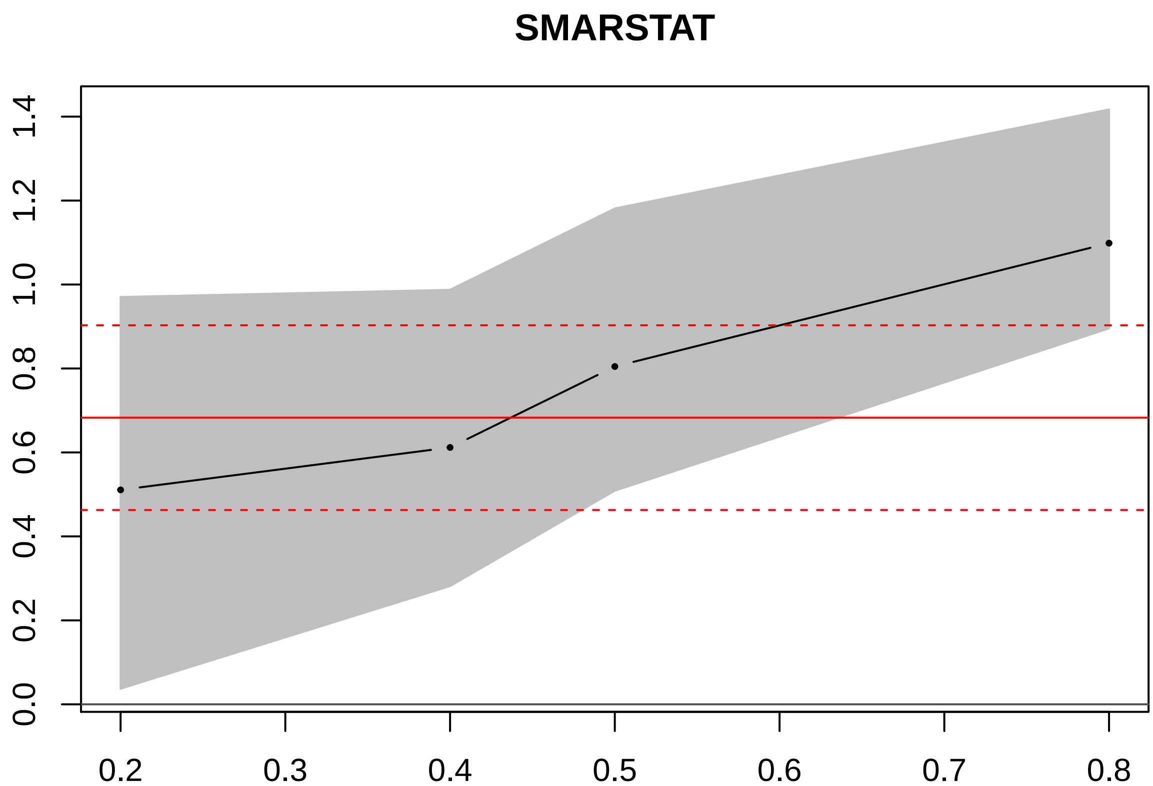

INCOME
3e−05
2e−05
1e−05
0e+00
0.2
0.3
0.4
0.5
0.6
0.7
0.8

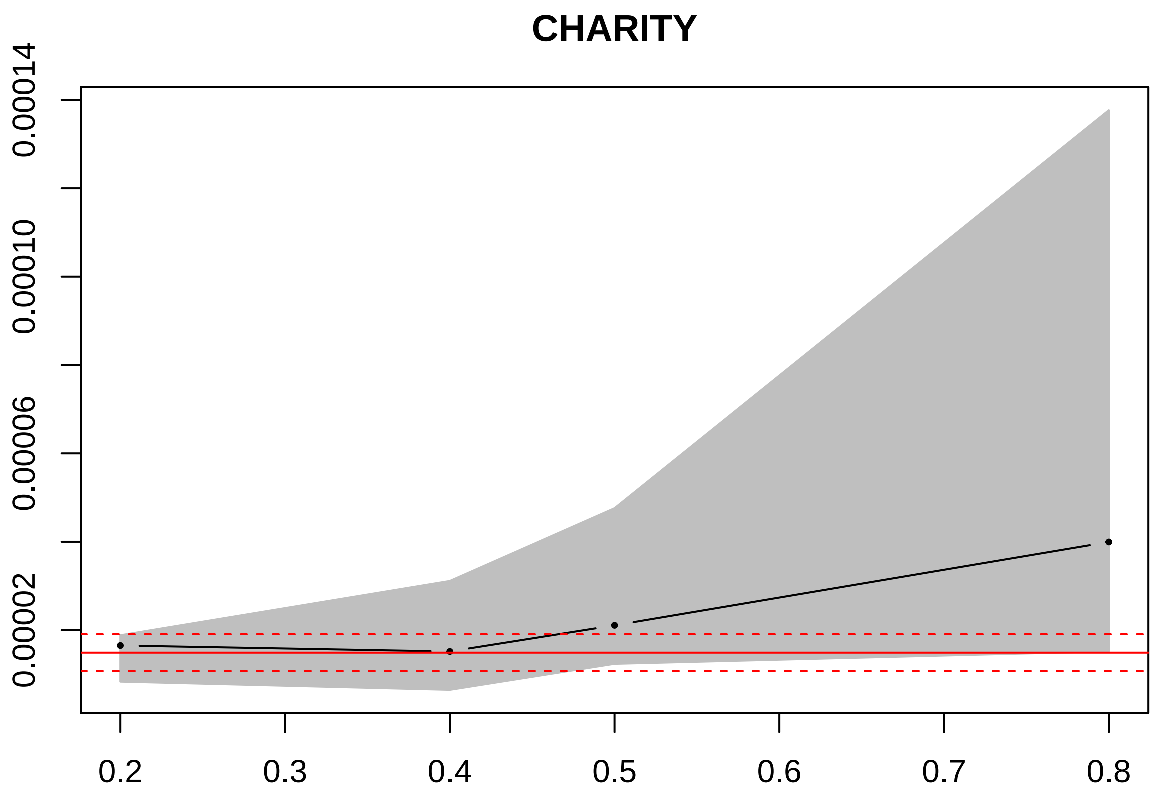
NUMHH



MARSTAT





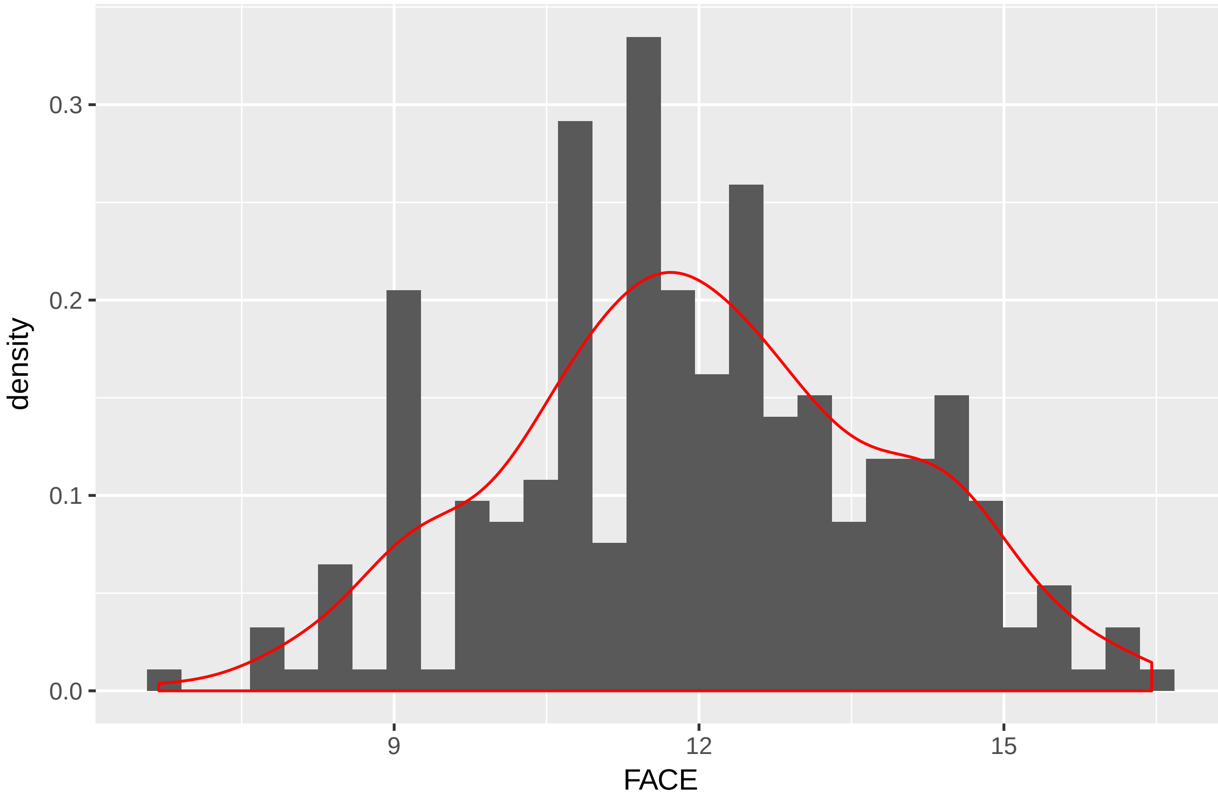


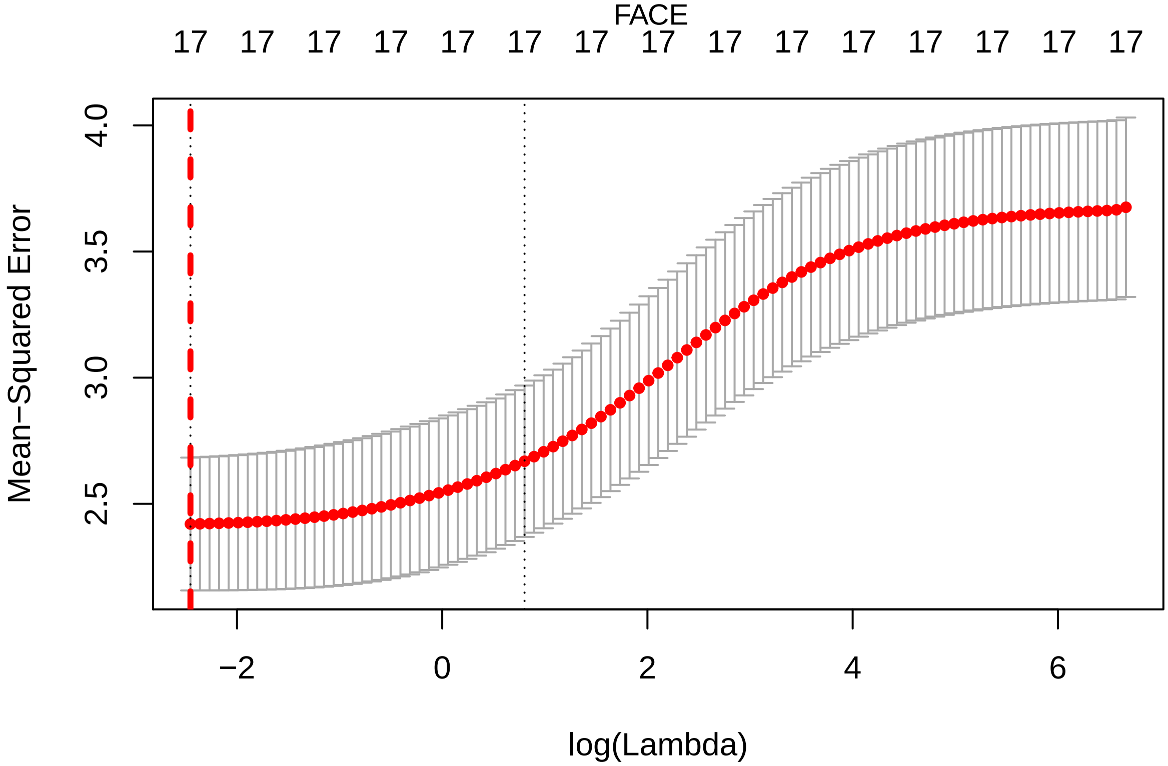
7.11 Ridge Regression

```
## [1] "GENDER"              "AGE"                   "MARSTAT"
```

```
##  [4] "EDUCATION"           "NUMHH"               "INCOME"
##  [7] "TOTINCOME"           "CHARITY"             "FACE"
## [10] "FACECVLIFEPOLICIES" "CASHCVLIFEPOLICIES"
```

Figure 19. Histogram of FACE





```
##         (Intercept)         (Intercept)             GENDER1
##        7.715131e+00        0.000000e+00        6.724101e-01
##                 AGE            MARSTAT1            MARSTAT2
##       -4.340113e-03        9.217190e-01        2.192050e-01
##           EDUCATION              NUMHH2              NUMHH3
##        2.320771e-01       -1.136074e+00        6.412676e-02
##              NUMHH5              NUMHH6              NUMHH7
##        1.645685e-01        9.339345e-01        9.584441e-01
##              NUMHH8              NUMHH9              INCOME
##       -1.959097e-01       -1.438891e+00       -2.993052e-07
##           TOTINCOME             CHARITY FACECVLIFEPOLICIES
##        3.376142e-08        1.992939e-05        3.883365e-08
## CASHCVLIFEPOLICIES
##        1.710886e-06
```

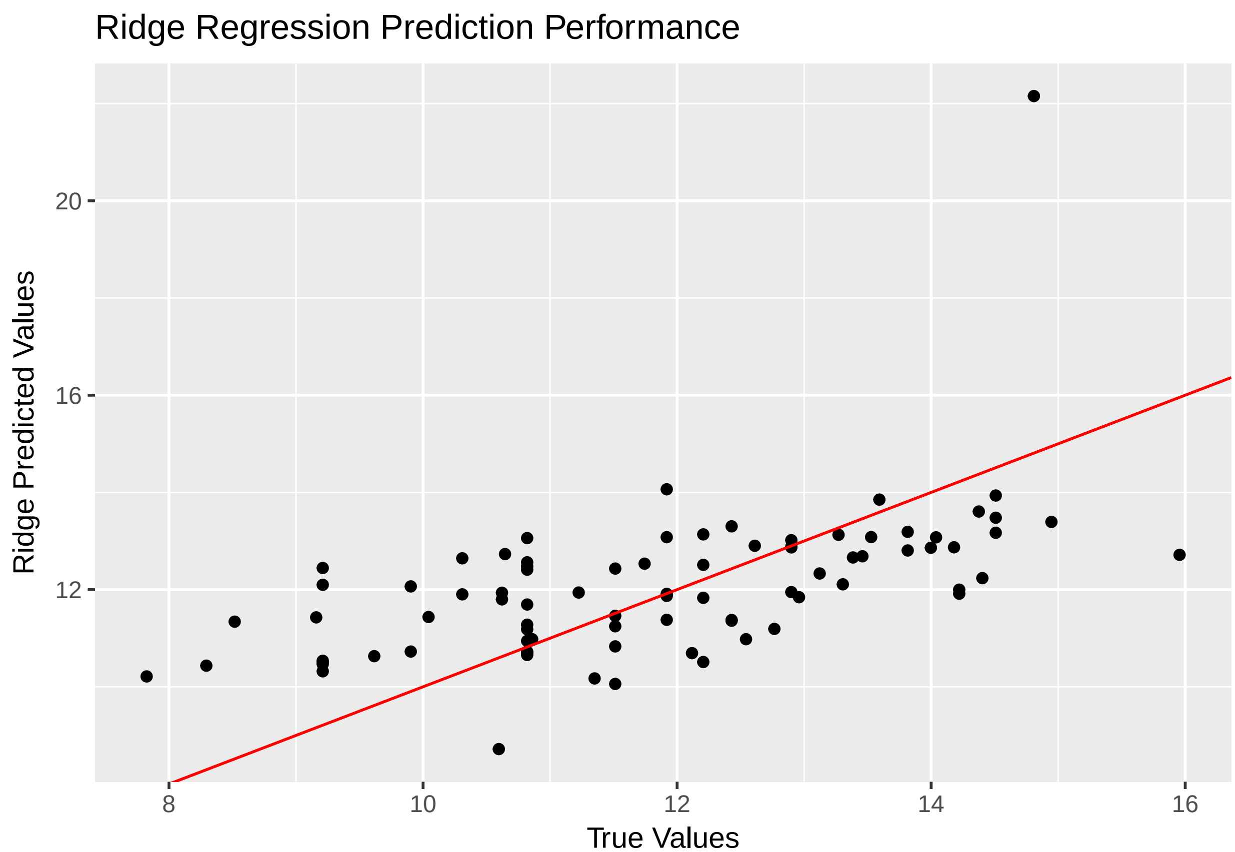
```
## [1] "Ridge Regression Test MSE"
```

```
## [1] 2.568472
```

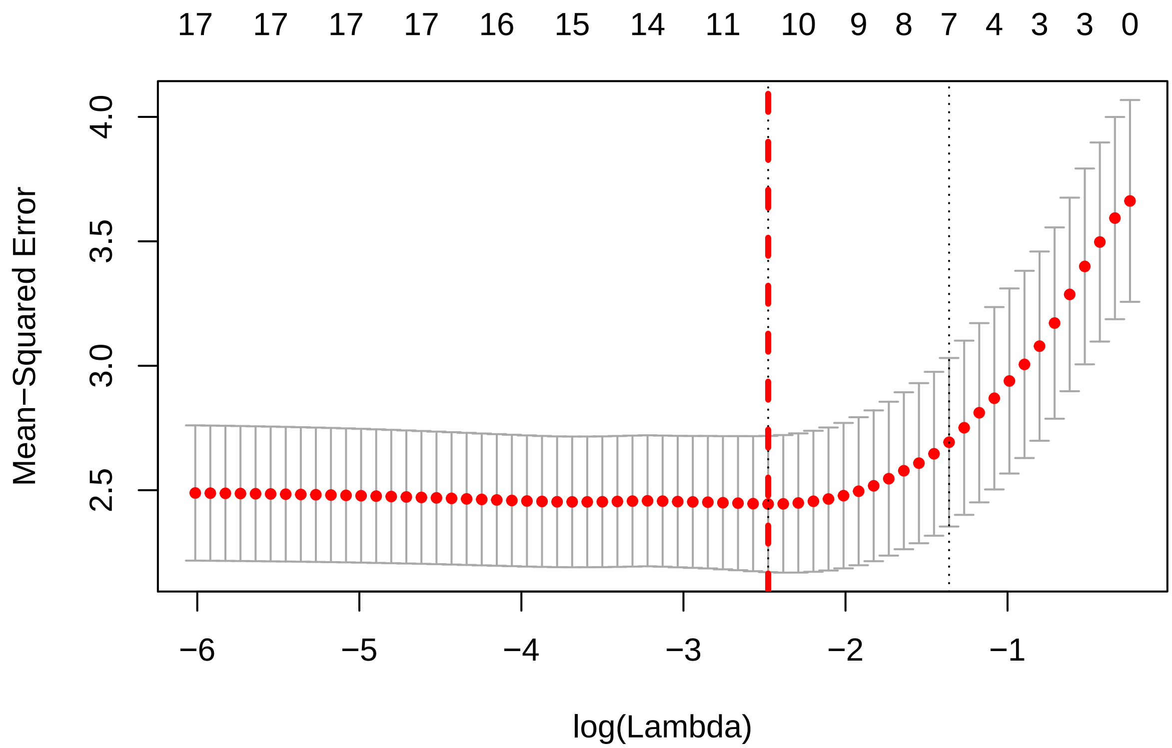
```
## [1] "Ridge Regression Predictions Average Percent Error"
```

```
## [1] 0.03152782
```

```
## 'data.frame':    83 obs. of  2 variables:
##  $ y_test: num  10.8 12.4 10.8 14.2 11.9 ...
##  $ X1    : num  13.1 11.4 10.7 11.9 11.4 ...
```



7.12 Lasso Regression



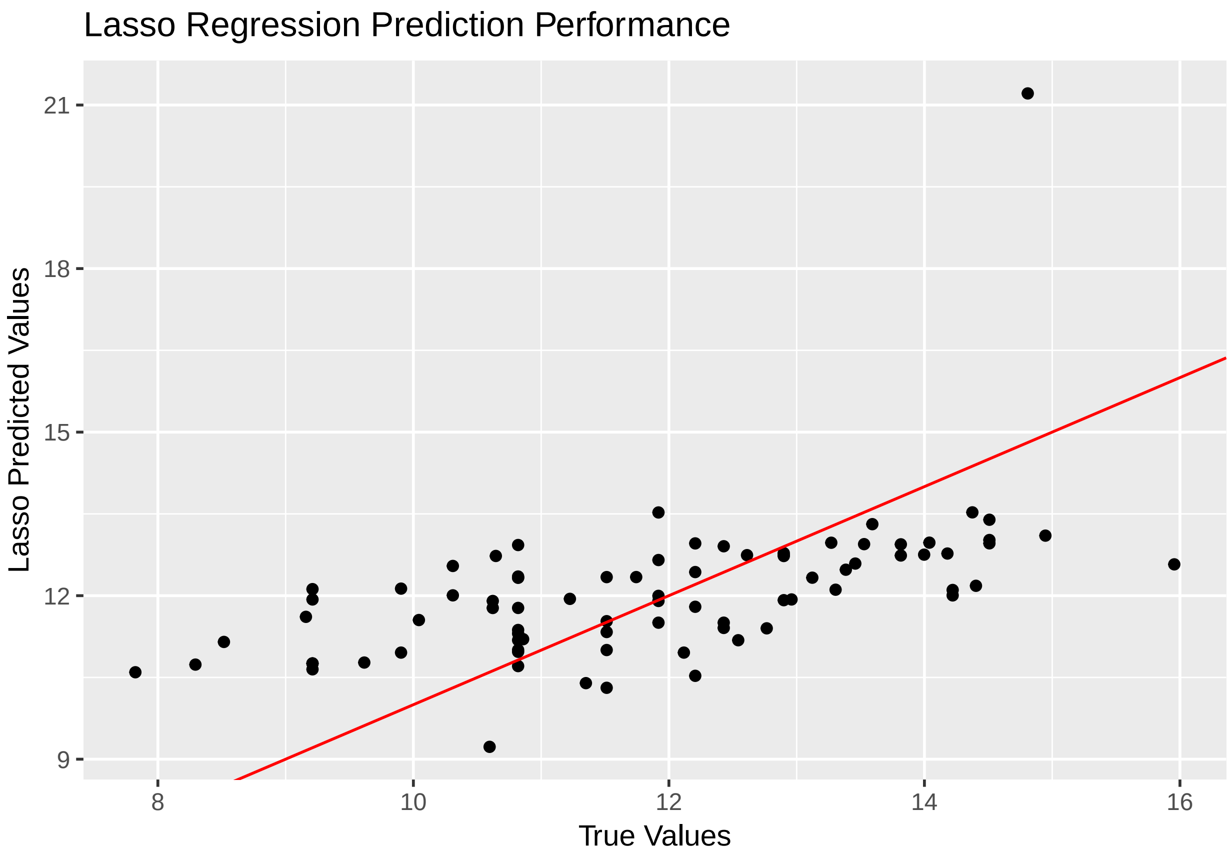

```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)         8.185143e+00
## (Intercept)         .
## GENDER1             5.320118e-01
## AGE                 .
## MARSTAT1            8.235937e-01
## MARSTAT2            .
## EDUCATION           1.990976e-01
## NUMHH2             -9.749511e-01
## NUMHH3              .
## NUMHH5              .
## NUMHH6              5.237853e-01
## NUMHH7              .
## NUMHH8              .
## NUMHH9             -3.483771e-01
## INCOME              .
## TOTINCOME           1.340543e-08
## CHARITY             1.496887e-05
## FACECVLIFEPOLICIES  3.761012e-08
## CASHCVLIFEPOLICIES  1.169839e-06

## [1] "Lasso Regression Test MSE"

## [1] 2.403266

## [1] "Lasso Regression Predictions Average Percent Error"

## [1] 0.03115868
```



8 References

1. https://stats.stackexchange.com/questions/90659/why-is-auc-higher-for-a-classifier-that-is-less-accurate-than-for-one
2. https://www.methodsconsultants.com/tutorial/what-is-the-difference-between-logit-and-probit-models/
3. http://www.physics.csbsju.edu/stats/KS-test.html
4. http://www.stat.ualberta.ca/~kcarrier/STAT562/comp_log_log

9 Acknowledgements

We would like to thank Ian Duncan for continuous support and guidance. We would also like to thank Nhan Huynh for giving us her time, attention and advice in the process.

10 Appendix

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(digits = 4)
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
# Libraries
library(e1071)
library(broom)
library(glmnet)
library(plyr)
library(dplyr)
library(ggplot2)
library(tidyverse)
library(randomForest)   # random forest analysis
library(car)            # vif - variance inflation factor
library(MASS)           # stepAIC
library(ROCR)           # ROC Curves
library(pROC)           # ROC, AUC
library(caret)          # confusionMatrix
library(mice)
library(quantreg)       # quantile regression
# Read in data
#setwd("~/Desktop/Classes/PSTAT 196")
termlife <- read.csv("TermLife.csv")
# Subset Dataset, Delete Variables

# ethnicity factor-levels unclear -> toss this variable
# borrowcvlifepol factor-levels unclear -> toss this variable
# netvalue factor-levels unclear -> toss this variable

termlf <- subset(termlife, select = -c(ETHNICITY, NETVALUE, BORROWCVLIFEPOL))
term <- termlf

# Create agediff variable - only for those with spouse
termlf$AGEDIFF <- ifelse(termlf$MARSTAT==0, NA, termlf$AGE - termlf$SAGE)

# Create edudiff variable - only for those with spouse
termlf$EDUDIFF <- ifelse(termlf$MARSTAT==0, NA, termlf$EDUCATION - termlf$SEDUCATION)

# Set missing spouse ages to NA
termlf$SAGE[termlf$SAGE == 0] <- NA

# Set missing seducation values to NA
termlf$SEDUCATION[termlf$MARSTAT==0] <- NA

# based on education differences and age differences between interviewer and spouse, maybe possible to

str(termlf)
attach(termlf)
#continuous variables
termlife_cont <- data.frame(AGE, EDUCATION, SAGE, AGEDIFF, SEDUCATION, NUMHH, INCOME, TOTINCOME, CHARIT
summary(termlife_cont)
print("Variance of Variables")
```

```r
apply(termlife_cont, 2, var)
catvar <- c("GENDER", "SGENDER", "MARSTAT", "SMARSTAT", "EDUCATION", "SEDUCATION", "EDUDIFF", "NUMHH",
termlife_cat <- termlf[catvar]
termlife_cat <-termlife_cat %>% mutate_each_(funs(factor(.)), catvar) #change into categorical variable
str(termlife_cat)
# Calculate variable correlation values with a few different approaches to measuring correlation
termlf %>% select_at(vars(AGE,EDUCATION,FACE,INCOME,TOTINCOME,NUMHH,CHARITY,AGEDIFF,Term_Flag)) %>%
  cor(method = "pearson", use = "complete")
log_income <- log(INCOME)
log_age <- log(AGE)
log_education <- log(EDUCATION)
log_face <- log(FACE)
pairs(~AGE + FACE + EDUCATION + INCOME, data = termlf, main = "Scatterplot Matrix")
pairs(~log_age + log_face + log_education + log_income, main = "Log-transformed Scatterplot Matrix")
### Graphs of Variable Relationships (Figures 1-4)
termlf %>% ggplot(aes(x = AGEDIFF, y = FACE)) + geom_bar(stat = "identity") + labs(title = "Figure 1")
termlf %>% ggplot(aes(x = AGEDIFF, y = EDUCATION)) + geom_bar(stat = "identity") + labs(title = "Figure
termlf %>% ggplot(aes(x = AGEDIFF, y = INCOME)) + geom_bar(stat = "identity") + labs(title = "Figure 3")
termlf %>% ggplot(aes(x = EDUCATION, y = FACE)) + geom_bar(stat = "identity") + labs(title = "Figure 4")
### Box Plots of Variables vs Term_Flag (Figure 5-8)
termlf$Term_Flag <- factor(Term_Flag, levels = c(0, 1), labels = c("No", "Yes"))
age <- ggplot(termlf, aes(x = Term_Flag, y = AGE, fill = Term_Flag)) + geom_boxplot() + labs(title = "Fi
age

AGEDIFFs <- ggplot(termlf, aes(x = Term_Flag, y = AGEDIFF, fill = Term_Flag)) + geom_boxplot() + labs(ti
AGEDIFFs

edu <- ggplot(termlf, aes(x = Term_Flag, y = EDUCATION, fill = Term_Flag)) + geom_boxplot() + labs(title
edu

numh <- ggplot(termlf, aes(x = Term_Flag, y = NUMHH, fill = Term_Flag)) + geom_boxplot() + labs(title =
numh
## Histograms (Figures 9-14)
# AGE Density Histogram
ggplot(termlf) +
  geom_histogram(mapping = aes(x = AGE,y = ..density..),binwidth = 2,na.rm = T) +
  geom_density(mapping = aes(x = AGE, y = ..density..), col="red") +
  labs(title = "Figure 9. Histogram of AGE")


# SAGE Density Histogram
ggplot(termlf) +
  geom_histogram(mapping = aes(x = SAGE,y = ..density..),binwidth = 3,na.rm = T) +
  geom_density(mapping = aes(x = SAGE, y = ..density..), col="red") +
  labs(title = "Figure 10. Histogram of SAGE")

# EDUCATION Density Histogram
ggplot(termlf) +
  geom_histogram(mapping = aes(x = EDUCATION,y = ..density..),binwidth = 2,na.rm = T) +
  geom_density(mapping = aes(x = EDUCATION, y = ..density..), col="red") +
  labs(title = "Figure 11. Histogram of EDUCATION")

# SEDUCATION Density Histogram
```

```
ggplot(termlf) +
  geom_histogram(mapping = aes(x = SEDUCATION,y = ..density..),binwidth = 2,na.rm = T) +
  geom_density(mapping = aes(x = SEDUCATION, y = ..density..), col="red") +
  labs(title = "Figure 12. Histogram of SEDUCATION")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = AGEDIFF,y = ..density..),binwidth = 4,na.rm = T) +
  geom_density(mapping = aes(x = AGEDIFF, y = ..density..), col="red") +
  labs(title = "Figure 13. Histogram of AGEDIFF")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = EDUDIFF,y = ..density..),binwidth = 1,na.rm = T) +
  geom_density(mapping = aes(x = EDUDIFF, y = ..density..), col="red") +
  labs(title = "Figure 14. Histogram of EDUDIFF")

### Zoomed into histograms (Figures 15-21)
ggplot(termlf) +
  geom_histogram(mapping = aes(x = INCOME, y = ..density..)) +
  geom_density(mapping = aes(x = INCOME, y = ..density..), col = "red") +
  xlim(0, 200000) + labs(title = "Figure 16. Histogram of INCOME")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = TOTINCOME, y = ..density..)) +
  geom_density(mapping = aes(x = TOTINCOME, y = ..density..), col = "red") +
  xlim(0, 200000) + ylim(0, 1e-5) + labs(title = "Figure 17. Histogram of TOTINCOME")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = CHARITY, y = ..density..)) +
  geom_density(mapping = aes(x = CHARITY, y = ..density..), col = "red") +
  xlim(0, 50000) + ylim(0, 3e-5) + labs(title = "Figure 18. Histogram of CHARITY")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = FACE, y = ..density..)) +
  geom_density(mapping = aes(x = FACE, y = ..density..), col = "red") +
  xlim(0, 50000) + labs(title = "Figure 19. Histogram of FACE")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = FACECVLIFEPOLICIES, y = ..density..)) +
  geom_density(mapping = aes(x = FACECVLIFEPOLICIES, y = ..density..), col = "red") +
  xlim(0, 100000) + ylim(0, 0.00005) + labs(title = "Figure 20. Histogram of FACECVLIFEPOLICIES")

ggplot(termlf) +
  geom_histogram(mapping = aes(x = CASHCVLIFEPOLICIES, y = ..density..)) +
  geom_density(mapping = aes(x = CASHCVLIFEPOLICIES, y = ..density..), col = "red") +
  xlim(0, 6500) + labs(title = "Figure 21. Histogram of CASHCVLIFEPOLICIES")

#***Note: Warning message "Removed n rows containing non-finite values..." just means that because we z

### Graphs Illustrating Relationships of Term_Flag (variable indicating purchase) and Other Variables
# Subset Dataset, Delete Variables

# ethnicity factor-levels unclear -> toss this variable
# borrowcvlifepol factor-levels unclear -> toss this variable
```

```
# netvalue factor-levels unclear -> toss this variable

termlf <- subset(termlife, select = -c(ETHNICITY, NETVALUE, BORROWCVLIFEPOL))

# based on education differences and age differences between interviewer and spouse, maybe possible to

#continuous variables
termlife_cont <- data.frame(AGE, EDUCATION, SAGE, AGEDIFF, SEDUCATION, NUMHH, INCOME, TOTINCOME, CHARIT
yes_cat <- termlife_cat[which(termlf$Term_Flag == 1),]
EDU_yes <- data.frame(table(yes_cat$EDUCATION))
colnames(EDU_yes) <- c("Education", "Freq")
ggplot(EDU_yes, aes(x=Education, y=Freq, fill=Education)) + geom_bar(width = 1, stat = "identity") + lab

marstat_yes <- data.frame(table(yes_cat$MARSTAT))
colnames(marstat_yes) <- c("Marital_Status", "Freq")
ggplot(marstat_yes, aes(x = Marital_Status, y = Freq, fill = Marital_Status)) + geom_bar(width = 1, sta

gender_cat <- data.frame(table(yes_cat$GENDER))
colnames(gender_cat) <- c("Gender", "Freq")
ggplot(gender_cat, aes(x = Gender, y = Freq, fill = Gender)) + geom_bar(width = 1, stat = "identity") +

numhh_yes <- data.frame(table(yes_cat$NUMHH))
colnames(numhh_yes) <- c("Number_of_Household_Members", "Freq")
ggplot(numhh_yes, aes(x =Number_of_Household_Members, y = Freq), fill = Number_of_Household_Members) + 
# Subset Dataset, Delete Variables

# ethnicity factor-levels unclear -> toss this variable
# borrowcvlifepol factor-levels unclear -> toss this variable
# netvalue factor-levels unclear -> toss this variable

termlf <- subset(termlife, select = -c(ETHNICITY, NETVALUE, BORROWCVLIFEPOL))
catvar <- c("GENDER", "SGENDER", "MARSTAT", "SMARSTAT", "Term_Flag")
term <- term %>% mutate_each_(funs(factor(.)), catvar) #change into categorical variables

# Create agediff variable - only for those with spouse
termlf$AGEDIFF <- ifelse(termlf$MARSTAT==0, NA, termlf$AGE - termlf$SAGE)

# Create edudiff variable - only for those with spouse
termlf$EDUDIFF <- ifelse(termlf$MARSTAT==0, NA, termlf$EDUCATION - termlf$SEDUCATION)

# Set missing spouse ages to NA
termlf$SAGE[termlf$SAGE == 0] <- NA

# Set missing seducation values to NA
termlf$SEDUCATION[termlf$MARSTAT==0] <- NA

# Data Frame Preview
str(termlf)

#continuous variables
termlife_cont <- data.frame(AGE, EDUCATION, SAGE, AGEDIFF, SEDUCATION, NUMHH, INCOME, TOTINCOME, CHARIT
                            FACE, FACECVLIFEPOLICIES, CASHCVLIFEPOLICIES)
term <- termlf
```

```r
term$SAGE[is.na(term$SAGE)] <- 0
term$SEDUCATION[is.na(term$SEDUCATION)] <- 0
term$AGEDIFF[is.na(term$AGEDIFF)] <- 0
term$EDUDIFF[is.na(term$EDUDIFF)] <- 0

catvar <- c("GENDER", "SGENDER", "MARSTAT", "SMARSTAT", "Term_Flag")
term <- term %>% mutate_each(funs(factor(.)), catvar) #change into categorical variables

# Split - Purchase, non-Purchase
term_p <- term[ which(term$Term_Flag == 1),]
term_np <- term[ which(term$Term_Flag == 0),]

### Split - Train/Test - purchasers
set.seed(1)
indLR_p <- sample(2, nrow(term_p), replace = T, prob = c(0.7, 0.3))  # split data into test set and tra
traindataLR_p <- term_p[indLR_p == 1,]
testdataLR_p <- term_p[indLR_p == 2,]

### Split - Train/Test - non-purchasers
set.seed(1)
indLR_np <- sample(2, nrow(term_np), replace = T, prob = c(0.7, 0.3))  # split data into test set and t
traindataLR_np <- term_np[indLR_np == 1,]
testdataLR_np <- term_np[indLR_np == 2,]

# Combine Train Sets - purchasers, non-purchasers
train_full <- rbind(traindataLR_p, testdataLR_np)

# Combine Test Sets - purchasers, non-purchasers
test_full <- rbind(testdataLR_p, testdataLR_np)
# Random Forest Model

# Tree functions
varsTree <- Term_Flag ~ . -FACE

# Applying the algorithm
treeRF <- randomForest(varsTree, data = train_full, ntree=100, proximity = T) # importance = T ?

# Importance of each variable
term.imp <- varImpPlot(treeRF, main = "Importance of each variable")
# Variable Importance Measure: Mean Decrease in Gini Index
importance(treeRF)

# Class Prediction Object / ROC Curve
pred.rf = predict(treeRF, test_full, type="prob")
pred.rf = prediction(pred.rf[,2], test_full$Term_Flag)
perf.rf = performance(pred.rf, measure="tpr", x.measure="fpr")
plot(perf.rf, col=2, lwd=3, main="Life Insurance Purchaser: ROC Curve for Random Forest")
abline(0,1)

# AUC
auc.glmRF = performance(pred.rf, "auc")@y.values
auc.glmRF
```

```
# Confusion Matrix
pred.rf = predict(treeRF, test_full, type="response")
confusionMatrix(pred.rf, test_full$Term_Flag)
# Use Alias/VIF to eliminate multicollinearity in predictors

# start here // face correlated with response; agediff, edudiff correlated w/ indep vars
term.lrm.r1 <-glm(Term_Flag ~ . -FACE-AGEDIFF-EDUDIFF, data=train_full, family=binomial(link = "logit")

#vif(term.lrm.r1)    # fails with error - b/c of perfect correlation / alias

alias(term.lrm.r1)  # check problematic variables

# problematic variables removed
term.lrm.r2 <-glm(Term_Flag ~ . -FACE-AGEDIFF-EDUDIFF-SMARSTAT-SGENDER, data=train_full, family=binomia

vif(term.lrm.r2)    # shows us SAGE and SEDUCATION are also too highly correlated

# MODEL 1: base model - removed all spouse vars b/c multicollinearity - FULL MODEL
term.lrm.r3 <-glm(Term_Flag ~ . -FACE-AGEDIFF-EDUDIFF-SMARSTAT-SGENDER-SAGE-SEDUCATION, data=train_full

vif(term.lrm.r3) # in the clear, no further significant multicollinearity

#summary(term.lrm.r3)

# MODEL 2: reduced model
term.lrm.r4 <-glm(Term_Flag ~ . -FACE-AGEDIFF-EDUDIFF-SMARSTAT-SGENDER-SAGE-SEDUCATION-FACECVLIFEPOLICI

# MODEL 3: log AGE predictor model
logistic_model <-glm(Term_Flag ~ . -FACE-AGEDIFF-EDUDIFF-SMARSTAT-SGENDER-SAGE-SEDUCATION-FACECVLIFEPOLI
# Logistic ROC Curves
pred.glm1 = predict(term.lrm.r3, test_full, type="response")
predict.glm1 = prediction(pred.glm1, test_full$Term_Flag)
perf.glm1 = performance(predict.glm1, measure="tpr", x.measure="fpr")

pred.glm2 = predict(term.lrm.r4, test_full, type="response")
predict.glm2 = prediction(pred.glm2, test_full$Term_Flag)
perf.glm2 = performance(predict.glm2, measure="tpr", x.measure="fpr")

pred.glm3 = predict(logistic_model, test_full, type="response")
predict.glm3 = prediction(pred.glm3, test_full$Term_Flag)
perf.glm3 = performance(predict.glm3, measure="tpr", x.measure="fpr")

plot(perf.glm1, col="orange", lwd=2, main="ROC Curve Logistic Models")
plot(perf.glm2, col="purple", lwd=2, main="ROC Curve Logistic Models", add="T")
plot(perf.glm3, col = "green", lwd = 2, main = "ROC Curve Logistic Models", add = "T")
legend("bottomright", inset = .05, legend=c("Model 1: Full Model", "Model 2: Reduced Model","Model 3: R
abline(0,1)
test_full = test_full %>%
mutate(Term_Flag=as.factor(ifelse(Term_Flag==0,"No", "Yes")))
pred.logistic1_table <- ifelse(pred.glm1 > 0.5, "Yes", "No")
table(pred=pred.logistic1_table, true=test_full$Term_Flag)
pred.logistic2_table <- ifelse(pred.glm2>0.5, "Yes", "No")
table(pred = pred.logistic2_table, true = test_full$Term_Flag)
```

```r
pred.logistic3_table <- ifelse(pred.glm3 > 0.5, "Yes", "No")
table(pred = pred.logistic3_table, true = test_full$Term_Flag)
auc.logistic1 <- performance(predict.glm1,"auc")@y.values[[1]]
auc.logistic2 <- performance(predict.glm2,"auc")@y.values[[1]]
auc.logistic3 <- performance(predict.glm3,"auc")@y.values[[1]]
cat("AUC for logistic regression model 1:", auc.logistic1)
cat("AUC for logistic regression model 2:", auc.logistic2)
cat("AUC for logistic regression model 3:", auc.logistic3)
summary(logistic_model)
probit_model <-glm(Term_Flag ~ . -FACE-SMARSTAT-SGENDER-SAGE-SEDUCATION-FACECVLIFEPOLICIES-TOTINCOME-NU
summary(probit_model)
cloglog_model <-glm(Term_Flag ~ . -FACE-SMARSTAT-SGENDER-SAGE-SEDUCATION-FACECVLIFEPOLICIES-TOTINCOME-NU
summary(cloglog_model)
plot(logistic_model$fitted.values, probit_model$fitted.values,
xlab = "Logit Fitted Values", ylab = "Probit Fitted Values",
main = "Logit vs. Probit Fitted Values", pch=19, cex=0.2)
abline(a=0, b=1, col="red")
plot(logistic_model$fitted.values, cloglog_model$fitted.values,
xlab = "Logit Fitted Values", ylab = "C-Log-Log Fitted Values",
main = "Logit vs. C-Log-Log Fitted Values", pch=19, cex=0.2)
abline(a=0, b=1, col="red")
prob.logistic <- predict(logistic_model, test_full, type = "response")
prediction.logistic <- prediction(prob.logistic, test_full$Term_Flag)
perf.logistic <- performance(prediction.logistic, measure = "tpr", x.measure = "fpr")
prob.probit <- predict(probit_model, test_full, type="response")
prediction.probit <- prediction(prob.probit, test_full$Term_Flag)
perf.probit <- performance(prediction.probit, measure = "tpr", x.measure = "fpr")
prob.cloglog <- predict(cloglog_model, test_full, type="response")
prediction.cloglog <- prediction(prob.cloglog, test_full$Term_Flag)
perf.cloglog <- performance(prediction.cloglog, measure = "tpr", x.measure = "fpr")
plot(perf.logistic, col="blue", lwd=2, main="ROC Curve")
plot(perf.probit, col="red", lwd=2, main="ROC Curve", add="T")
plot(perf.cloglog, col = "green", lwd = 2, main = "ROC Curve", add = "T")
legend("bottomright", inset = .05, legend=c("Logistic Regression", "Probit Regression","C-Log-Log Regres
col=c("blue", "red", "green"), lty=1, cex=1)
abline(0,1)
prediction.logistic_table <- ifelse(prob.logistic > 0.5, "Yes", "No")
table(pred=prediction.logistic_table, true=test_full$Term_Flag)
prediction.probit_table <- ifelse(prob.probit>0.5, "Yes", "No")
table(pred = prediction.probit_table, true = test_full$Term_Flag)
prediction.cloglog_table <- ifelse(prob.cloglog > 0.5, "Yes", "No")
table(pred = prediction.cloglog_table, true = test_full$Term_Flag)
auc.logistic <- performance(prediction.logistic,"auc")@y.values[[1]]
auc.probit <- performance(prediction.probit,"auc")@y.values[[1]]
auc.cloglog <- performance(prediction.cloglog,"auc")@y.values[[1]]
cat("AUC for logistic regression:", auc.logistic)
cat("AUC for probit regression:", auc.probit)
cat("AUC for c-log-log regression:", auc.cloglog)
ks_logistic=max(attr(perf.logistic,'y.values')[[1]]-attr(perf.logistic,'x.values')[[1]])
plot(perf.logistic,main=paste0('Logistic KS=',round(ks_logistic*100,1),'%'))
lines(x = c(0,1),y=c(0,1))
```

```
ks_probit=max(attr(perf.probit,'y.values')[[1]]-attr(perf.probit,'x.values')[[1]])
plot(perf.probit,main=paste0('Probit KS=',round(ks_probit*100,1),'%'))
lines(x = c(0,1),y=c(0,1))


ks_cloglog=max(attr(perf.cloglog,'y.values')[[1]]-attr(perf.cloglog,'x.values')[[1]])
plot(perf.cloglog,main=paste0('C-Log-Log KS=',round(ks_cloglog*100,1),'%'))
lines(x = c(0,1),y=c(0,1))
termlf$FACE[termlf$FACE == 0] <- NA
log_face <- log(termlf$FACE)
taus <- c(0.2,0.4,0.5,0.8)
qr_multiple <- rq(FACE~EDUCATION + AGE + INCOME + TOTINCOME + NUMHH, data = termlf, tau = taus)
qr_multiple_log <- rq(log_face~EDUCATION + AGE + INCOME + TOTINCOME + NUMHH +
                        MARSTAT + SMARSTAT + CHARITY +
                        FACECVLIFEPOLICIES + CASHCVLIFEPOLICIES, data = termlf, tau = taus)
summary(qr_multiple_log)
coef(qr_multiple_log)
QR_education <- rq(log_face~EDUCATION, data = termlf, tau = taus)
QR_AGE <- rq(log_face~AGE, data = termlf, tau = taus)
QR_INCOME <- rq(log_face~INCOME, data = termlf, tau = taus)
QR_TOTINCOME <- rq(log_face~TOTINCOME, data = termlf, tau = taus)
QR_NUMHH <- rq(log_face~NUMHH, data = termlf, tau = taus)
QR_marital_status <- rq(log_face~MARSTAT,data=termlf, tau = taus)
QR_spouse_marital_status <- rq(log_face~SMARSTAT, data = termlf, tau = taus)
QR_charity <- rq(log_face~CHARITY, data = termlf, tau = taus)

plot(summary(QR_education), parm = "EDUCATION")
plot(summary(QR_AGE), parm = "AGE")
plot(summary(QR_INCOME), parm = "INCOME")
plot(summary(QR_NUMHH), parm = "NUMHH")
plot(summary(QR_marital_status), parm = "MARSTAT")
plot(summary(QR_spouse_marital_status), parm = "SMARSTAT")
plot(summary(QR_charity), parm = "CHARITY")
purchased <- which(termlf$Term_Flag == 1) #subset out rows that purchased insurance
termlf.p <- termlf[purchased,]
termlf.p <- termlf.p %>% mutate_each_(funs(factor(.)), c("GENDER", "MARSTAT", "NUMHH"))
remove <- c(5:8, 16:18)
termlf.p <- termlf.p[-remove]
colnames(termlf.p)
termlf.p$FACE <- log(termlf.p$FACE)
ggplot(termlf.p) +
  geom_histogram(mapping = aes(x = FACE, y = ..density..)) +
  geom_density(mapping = aes(x = FACE, y = ..density..), col = "red") + labs(title = "Figure 19. Histog
x <- model.matrix(termlf.p$FACE ~., data = termlf.p)[,-9]
y <- termlf.p$FACE
set.seed(45)
train_index <- sample(1:nrow(x), nrow(x) * 0.7)
test_index <- (-train_index)
x_train <- x[train_index,]
y_train <- y[train_index]
y_test <- y[test_index]
x_test <- x[test_index,]
cv_ridge <- cv.glmnet(x_train, y_train, alpha = 0) #perform cross-validation
```

```
ridge_opt_lambda <- cv_ridge$lambda.min
plot(cv_ridge)
abline(v = log(cv_ridge$lambda.min), col="red", lwd=3, lty=2)
ridge_fit <- glmnet(x_train, y_train, alpha = 0, lambda = ridge_opt_lambda)
ridge_pred <- predict(ridge_fit, s = ridge_opt_lambda, newx = x_test)
predict(ridge_fit, type = "coefficients",s = ridge_opt_lambda)[1:19,]
print("Ridge Regression Test MSE")
mean((ridge_pred - y_test)^2) #MSE
print("Ridge Regression Predictions Average Percent Error")
mean((ridge_pred - y_test)/y_test) #average percent error
test_labels <- as.data.frame(y_test)
ridge_pred <- as.data.frame(unlist(ridge_pred))
predtrue <- data.frame(test_labels, ridge_pred)
str(predtrue)
(ggplot(data = predtrue, aes(x = test_labels, y = ridge_pred)) + geom_point() + geom_abline(slope = 1, 
cv_lasso <- cv.glmnet(x_train, y_train, alpha = 1)
plot(cv_lasso)
abline(v = log(cv_lasso$lambda.min), col="red", lwd=3, lty=2)
lasso_opt_lambda <- cv_lasso$lambda.min
lasso_fit <- glmnet(x_train, y_train, alpha = 1, lambda = lasso_opt_lambda)
lasso_pred <- predict(lasso_fit, s = lasso_opt_lambda, newx = x_test)
predict(lasso_fit, type = "coefficients", s = lasso_opt_lambda)
print("Lasso Regression Test MSE")
mean((lasso_pred - y_test)^2) #MSE
print("Lasso Regression Predictions Average Percent Error")
mean((lasso_pred - y_test)/y_test) #average percent error
test_labels <- as.data.frame(y_test)
predtrue <- data.frame(test_labels, lasso_pred)
(ggplot(data = predtrue, aes(x = test_labels, y = lasso_pred)) + geom_point() + geom_abline(slope = 1, 
```