Predicting Bank Customer’s Subscription to Term Deposits

Project 2 - Applied Statistics: Inference & Modeling (DS 6372)

Summer 2021

Eli Kravez, Halle Purdom, Brandon Sucrese

# Introduction

In order to evaluate the success of bank telemarketing, a Portuguese banking institution collected data on their customers and whether or not a customer subscribed to a term deposit. These variables included factors dependent on the individual customer’s telemarketing experience and unrelated factors such as marital status and education. In order to analyze the factors and their influence in determining the outcome of subscribing to a term deposit, many statistical models were built in this study that predict whether or not a particular customer will subscribe to a term deposit. Two main objectives are achieved through these methods including building a simple logistic regression model aimed at interpretation and analysis of the variables and building more advanced models including advanced logistic regression, quadratic discriminant analysis, decision tree model, and a random forest model aimed at finding the best model for predicting whether or not a customer will subscribe to a term deposit.

# 

# Data Description

The Bank Marketing Data Set includes a variety of variables of client information including monthly income, marital status and education. The majority of the variables in the data set are categorical with a handful of continuous variables. A full list of the variables and their descriptions can be found in [Appendix I.A. Attribute Information](#_ve6trtjrxztw). The minimal invalid measurements in the dataset have been classified as NA and have been exempted from model building. The Bank Marketing Data Set includes 41,188 observations collected between May 2008 and November 2010. This data set is provided by UCI, and was originally used in the paper, “A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems.” The data set was collected from a Portuguese banking institution running marketing campaigns via phone calls. The primary goal of this data set is to determine if a client will subscribe to a term deposit based on their profile.

# 

# Exploratory Analysis

## Data Quality Evaluation

Twelve duplicate rows were found in the dataset. We removed these in order to prevent redundancy.

There were also no NA values (missing) in the data; however, there were a significant amount of ***unknown*** values in the data. These values are visualized in [Appendix I.C. Missing Values](#_6kdh0cs9utwv). Our decision was to treat unknowns as missing values.

One particular column in the dataset, the **default column**, has over 20% unknowns. The rest 99% of the values are “no” and only 3 “yes”. We decided to drop this column from our dataset because there was little practical significance in keeping the column.

## Class Distribution

The variable we want to predict in the dataset is a binomial categorical response. This class variable is named “y “ with two values – “yes” and “no”, referring to whether or not a customer subscribed to a term deposit. There are **36,548** rows with “no” and **4,640** of “yes”, as can be seen in [Appendix I.B. Data Distribution](#_km266i8j4pyl). Therefore, we are in a significantly skewed class situation. This impacts how we develop the prediction models in that we cannot use a simple 0.5 cutoff rule.

We addressed this issue in 2 ways depending on the model:

1. By using ROC graphs and finding the best cutoff.
2. Using upSample on the training set so that both classes will have the same number of rows for model training.

To look further into the class distributions among particular variables, see [Appendix I.D Continuous Predictors](#_venot0u84g) and [Appendix I.E. Categorical Predictors](#_tddb7tuxc80).

## Data Transformation

In order to utilize the dataset, all string-type categorical variables were converted to factors. In addition, features which are captured as string in the data but actually numeric include emp.var.rate, cons.price.idx, cons.conf.idx, euribor3m, nr.employed.

## Correlation Analysis

Because multicollinearity needs to be evaluated for parametric models like logistic regression ([see assumptions](#_gpnuk0n4mgjt)), correlations between continuous variables in the dataset were evaluated. From the correlation of the numerical predictors ([Appendix I.F. Correlation](#_xmllk4nkipvj)), we can see very strong correlation between the following (potential multicollinearity issue):

1. Euriborn3m and nr.employed – 0.95 Pearson correlation
2. nr.employed and emp.var.rate – 0.97 Pearson correlation
3. emp.var.rate and Euriborn3m – 0.91 Pearson correlation

## PCA

We performed PCA analysis with the goal to see if a clear separation between PC’s exists – this should provide additional confidence for us that we can separate classes using LDA/QDA tools.

We can see some kind of the separation from the PCA graphs ([Appendix I.G PCA](#_py34pwibf0bp)) – but it is not as strong as we expected. This is most likely because PCA only works with numerical variables, meaning only 5/19 predictors were used in the analysis. This might be the reason we do not see significant separation discovered by PCA.

# Objective 1

To establish a baseline, a simple logistic regression model is built in order to predict if a client will subscribe to a term deposit based on their profile. While this model may not perform the best in terms of prediction, it is the best model to look at for understanding the variables and determining how each affects the outcome of the binary response. Using the LASSO feature selection process, the predictors were selected and the model was evaluated based on the performance metrics AUC, accuracy, sensitivity and specificity. The unbalanced nature of the data set makes having both strong sensitivity and specificity important to ensure the model doesn’t favor one condition over the other. First we will address the underlying assumptions of logistic regression, then move on to the model and its performance metrics, and finally look at the interpretations that can be made about the predictors in the model.

## Assumptions of Logistic Regression

1. Independence - we assume independence assumption is valid. Data was collected from different clients.
2. Response is a two level categorical variable (yes/no).
3. Predictors are both categorical and continuous.

## Simple Logistic Regression Model (LASSO Feature Selection)

The simple logistic regression model chosen by LASSO feature selection is

*= + job + marital + education + contact + month + day\_of\_week + duration + campaign + pdays + previous + poutcome + emp.var.rate + cons.conf.idx + nr.employed*

where p(x) is the probability of whether or not a customer will subscribe to a term deposit and is the parameter estimate for each particular predictor. These values can be found in [Appendix II.B. Coefficient Summary](#_3lyn5u98n5xn). The following R code was used to create the model:

glm(y~ job + marital + education + contact +

month + day\_of\_week + duration + campaign +

pdays + previous + poutcome + emp.var.rate +

cons.conf.idx + nr.employed ,family="binomial",data=data)

| **Model Selection** | **AUC** | **Accuracy** | **Sensitivity** | **Specificity** |
| --- | --- | --- | --- | --- |
| **LASSO** | **0.94** | **0.895** | **0.9116** | **0.7644** |

Above are the prediction metrics for the simple logistic regression model, determined by splitting the data randomly into a 75/25 train/test split. Cutoff for the model was set at 0.2 to get these statistics.

Based on the variance inflation factor (VIF) table ([Appendix II.A. VIF](#_qze6557n8req)), we can see that the LASSO feature selection process removed high correlated features successfully as there were no high values for the VIF.

## Parameter Interpretation

See list of all the parameters with the p-values in the coefficient summary table ([Appendix II.B. Coefficient Summary](#_3lyn5u98n5xn)) .

We will use the odds ratio table ([Appendix II.C. Odds Ratio Table](#_qwjqklbcqfc4)) for the confidence intervals for the parameter interpretation. Because there are 14 variables in the model, we will provide interpretations for both types of variables (numerical and categorical). The same interpretations can be applied to each variable in the model.

Examples of numerical parameter interpretation:

**Duration**: last contact duration, in seconds (numeric). Based on the coefficient summary table in the appendix, the coefficient of the duration is 0.0047359647 . The p\_value for the coefficient of duration is < .005. Therefore, we reject H0 as the coefficient of duration == 0. Therefore, duration impacts the user decision of yes/no.

Interpretation: For every second increase in the call duration, the odds of outcome y to be yes are exp(0.0047359647) = 1.005 (multiplicative) keeping all other predictor variables fixed. 95% of confidence this odd ratio is between [1.004571, 1.004924]

It might be more convenient to think about the call duration in minutes and not seconds increase. In this case this would be the interpretation:

For every minute increase in the call duration, the odds of outcome y to be yes are exp(0.0047359647\*60) = 1.328 (multiplicative) keeping all other predictor variables fixed. This 95% of confidence this odd ratio is between [1.3147, 1.34273]

Examples of factor parameter interpretation:

**Contact:** customers can be contacted by cellular or by telephone. Based on the coefficient summary table in the appendix, coefficient of contact telephone = -0.2643908921. The p\_value for the coefficient of contact by telephone < .005. Therefore, we reject H0 as the coefficient of contact by telephone == 0. Therefore, contact types impact the user decision of yes/no .

Interpretation: the odds ratio estimate for a contact by telephone relative to a cellular is exp(-0.2643908921) = 0.767. That is, the odds of having y=yes for contact by telephone is 0.76 times smaller than a contact by cellular holding all other predictor variables fixed. This odds are between [0.656, 0.898] with 95% of confidence intervals

While various feature selections can be useful, the models tend to struggle in data sets that aren’t balanced. Both basic forward selection and lasso selection have strong accuracy numbers, but lack in specificity. Ultimately this can be fixed by adjusting the cutout to lean towards prioritizing specificity. While adjusting the cutoff can lead to better figures, a more complex model is required to better handle the nature of this problem.

# Objective 2

The selection of more complex models were created to better predict if customers would sign up for term deposits. The models that are being utilized are advanced LASSO, QDA, decision tree and random forest. With more powerful and complex models we expect to see better performance across the board.

To determine the best performing model, AUC, accuracy, sensitivity and specificity were calculated based on the same training and test datasets mentioned in the simple logistic regression model. Sensitivity and specificity were particularly important because of the unbalanced data. Often models would have artificially large accuracy and sensitivity metrics with a low performing specificity. To counteract these issues, adjustments to the models were needed.

Similar to the simple LASSO model, the cut off for classifications was adjusted for both the random forest and advanced LASSO models to prioritize specificity. To make the decision tree model perform better upscaling was utilized.

After individual adjustments were made to each model, the random forest model performed the most effectively (Fig 1). While the random forest model had the overall best metrics, these can fluctuate depending on the test and training sets being used. In this particular work the same testing and training datasets were used for all models in order to consistently compare and evaluate each model. The decision tree model saw the most fluctuations between runs, making it a poor choice in terms of consistency. Even though the random forest model had the best metrics, advanced LASSO or QDA could be used as a replacement still providing similar results.

To compare ROC curves for each of the models, see the respective model sections in the Appendix.

The random forest model’s variable importance feature also allowed some analysis into which variables were most influential when looking at the outcome of whether or not a customer subscribed to a term deposit. The last contact in duration was determined to be the most important factor, see [Appendix VII. Random Forest Model](#_apc2bbp6nhsl) for other variables.

Note: we used QDA as the [Equal Covariance assumptions for LDA](#_7xgcvedncggw) does not hold.

Fig 1.

| **Model** | **AUC** | **Accuracy** | **Sensitivity** | **Specificity** |
| --- | --- | --- | --- | --- |
| **Advanced LASSO** | **0.9322** | **0.8869** | **0.9024** | **0.7635** |
| **QDA** | **0.9** | **0.8711** | **0.8891** | **0.7285** |
| **Decision Tree** | **0.917** | **0.9027** | **0.9644** | **0.4142** |
| **Random Forest** | **0.944** | **0.8782** | **0.8790** | **0.8722** |

# 

# Conclusion

Overall, the complex models performed better than the simple logistic regression model when looking at prediction performance. Random forest is the best performing model based on our metrics, however the results are close enough that no model is particularly better than the others. While the simple LASSO model performed worse, the results are close enough that the simple LASSO model would still be a valid choice for this data set. Even though complex models seem like an easy choice, implementation and interpretation of the models can be more challenging. It is important to understand the data set and its characteristics to truly determine the best models.

In order to determine the best model from a particular dataset, the context surrounding the data and the purpose of the analysis must be considered. In order to use logistic regression and LDA/QDA, prior assumptions for the data must be met in order for the model to be valid. If these assumptions are not met, a nonparametric model like decision trees or random forests are likely to be the best choices. We were also dealing with a highly unbalanced dataset leading to differences in sensitivity and specificity, so if it is in the company’s best interest to have a model that predicts true positives or true negatives with more accuracy, cutoffs and models can be chosen respectively. In this particular case, we chose to evaluate the models overall by the four performance metrics listed above, however this may not always be the best route. The practical significance of the chosen model always needs to be considered.

Some further work into prediction model building based on this dataset could include building more nonparametric models such as k-nearest neighbors. In addition, looking more into the data imbalance problem of upSampling versus cutoff changes could be analyzed. Particularly if one versus the other produces significantly different performance statistics in the models. In the above analysis we used the same training and testing datasets to ensure consistency among the model comparisons, in the future this could be repeated many times over to prevent any variability in the model performance based on a specific test and train set.

# References

1. <https://archive.ics.uci.edu/ml/datasets/Bank+Marketing>

# Appendix

**Code:** <https://github.com/hallepurdom/MSDS-6372-Project-2/blob/main/Proj2%20Applied%20Stats.Rmd>

## EDA

### Attribute Information

**Input variables:**

**# bank client data:**

1 - age (numeric)

2 - job : type of job (categorical: 'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown')

3 - marital : marital status (categorical: 'divorced', 'married', 'single', 'unknown'; note: 'divorced' means divorced or widowed)

4 - education (categorical: 'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown')

5 - default: has credit in default? (categorical: 'no','yes','unknown')

6 - housing: has housing loan? (categorical: 'no','yes','unknown')

7 - loan: has personal loan? (categorical: 'no','yes','unknown')

**# related with the last contact of the current campaign:**

8 - contact: contact communication type (categorical: 'cellular','telephone')

9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')

10 - day\_of\_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')

11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

**# other attributes:**

12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)

13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)

14 - previous: number of contacts performed before this campaign and for this client (numeric)

15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')

**# social and economic context attributes**

16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)

17 - cons.price.idx: consumer price index - monthly indicator (numeric)

18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)

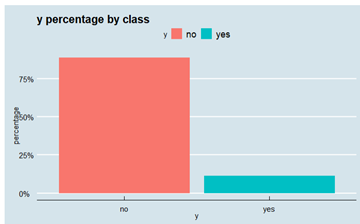
19 - euribor3m: euribor 3 month rate - daily indicator (numeric)

20 - nr.employed: number of employees - quarterly indicator (numeric)

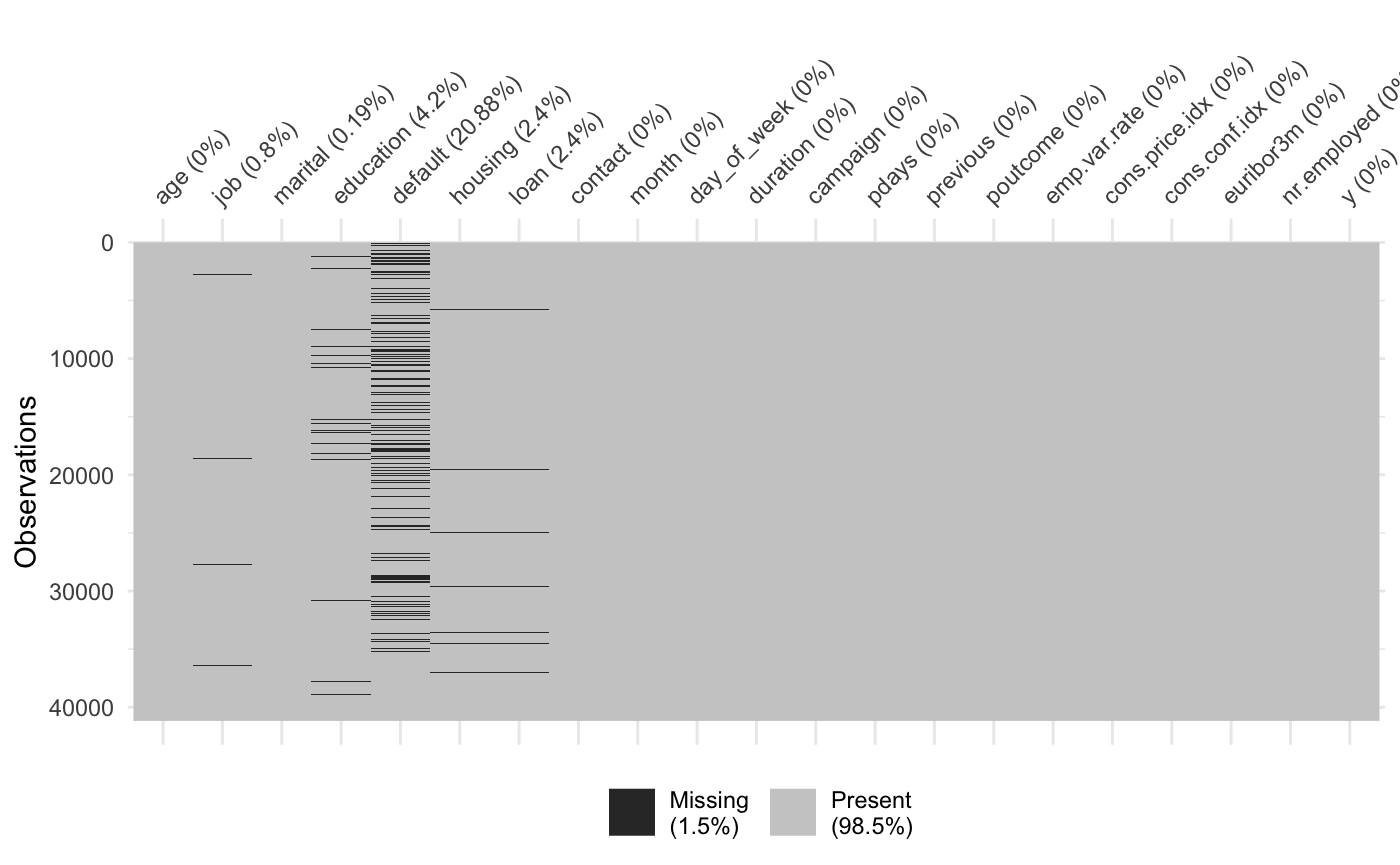
**Output variable (desired target):**

21 - y - has the client subscribed a term deposit? (binary: 'yes','no')

### Data DIstribution

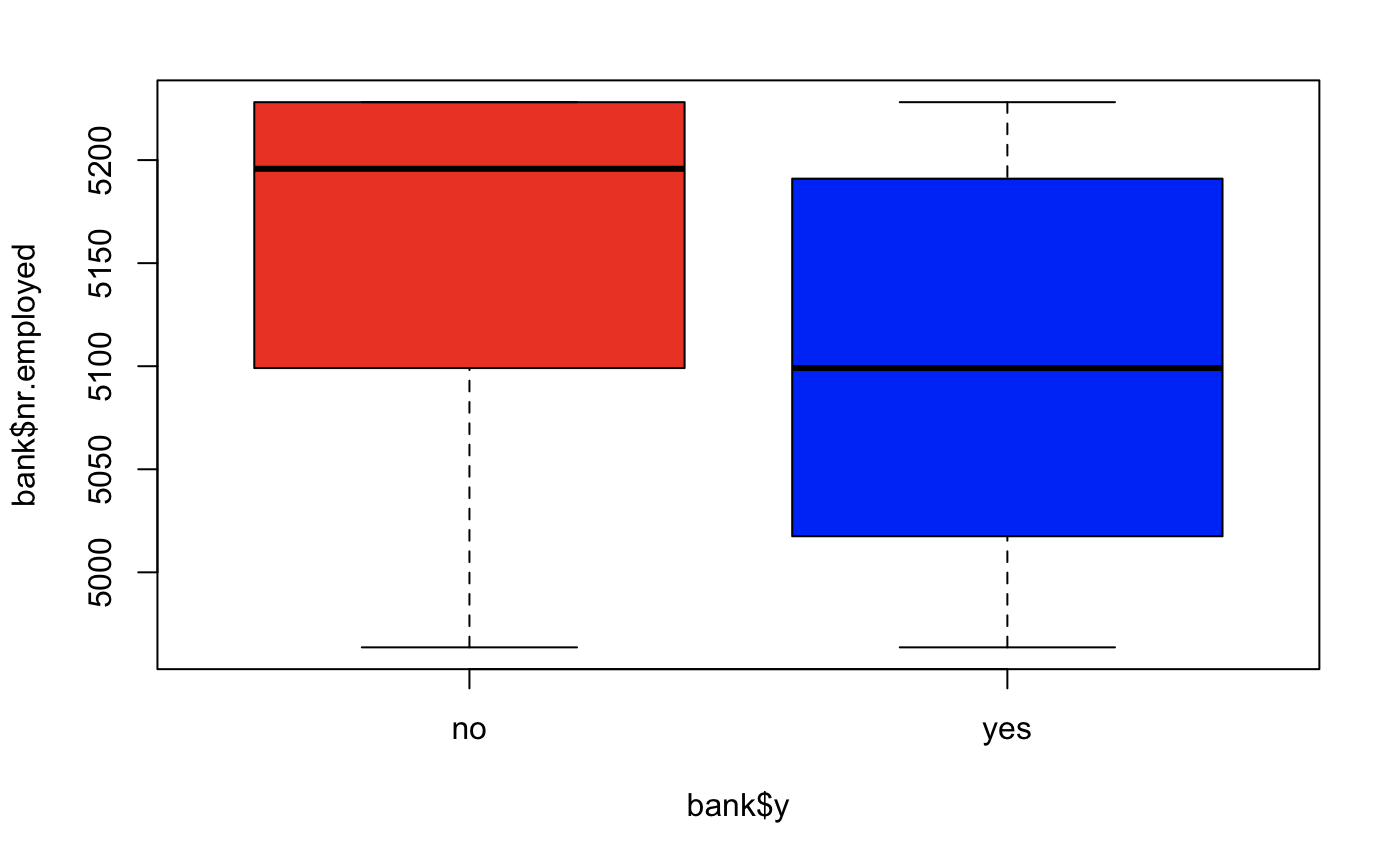
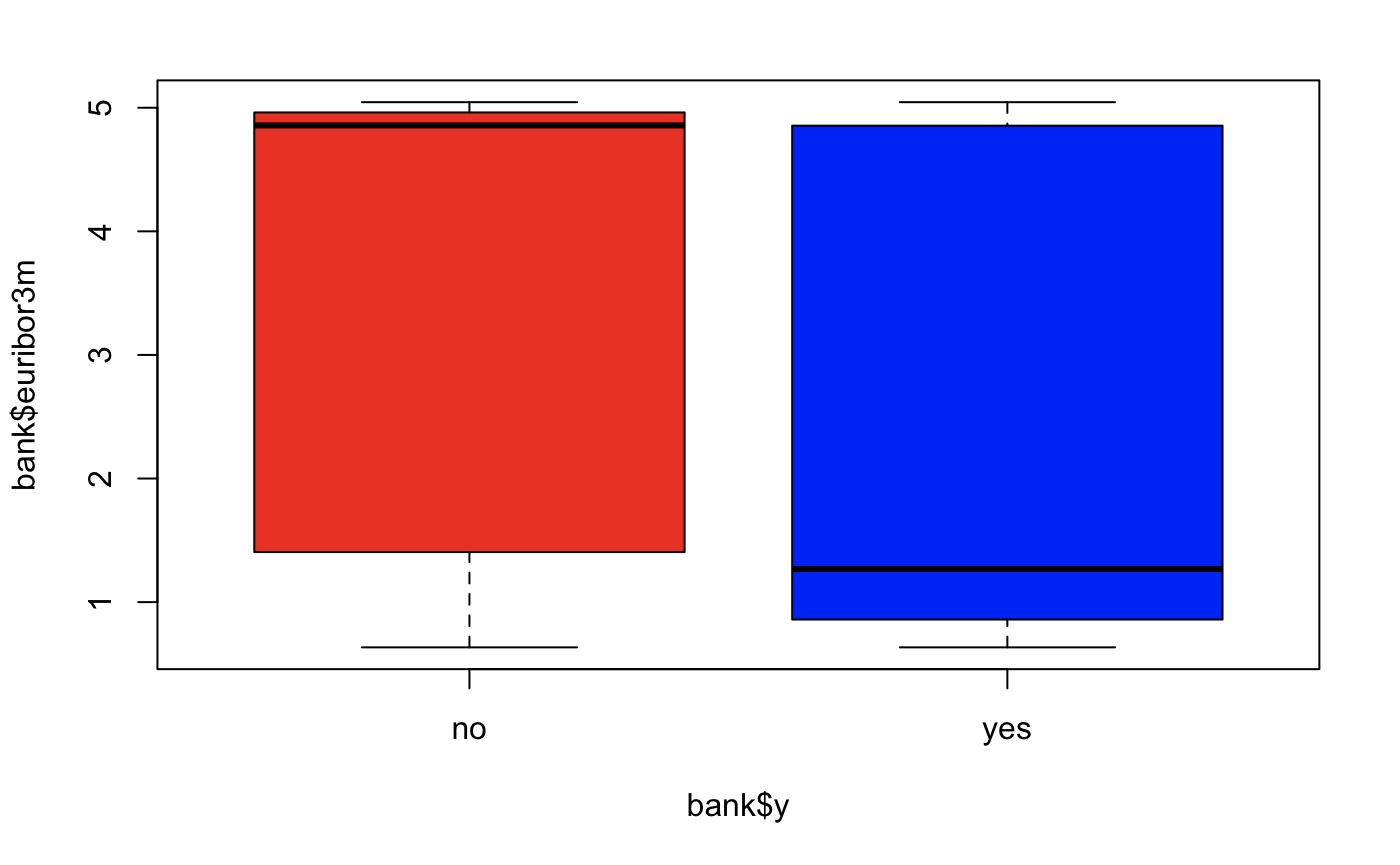
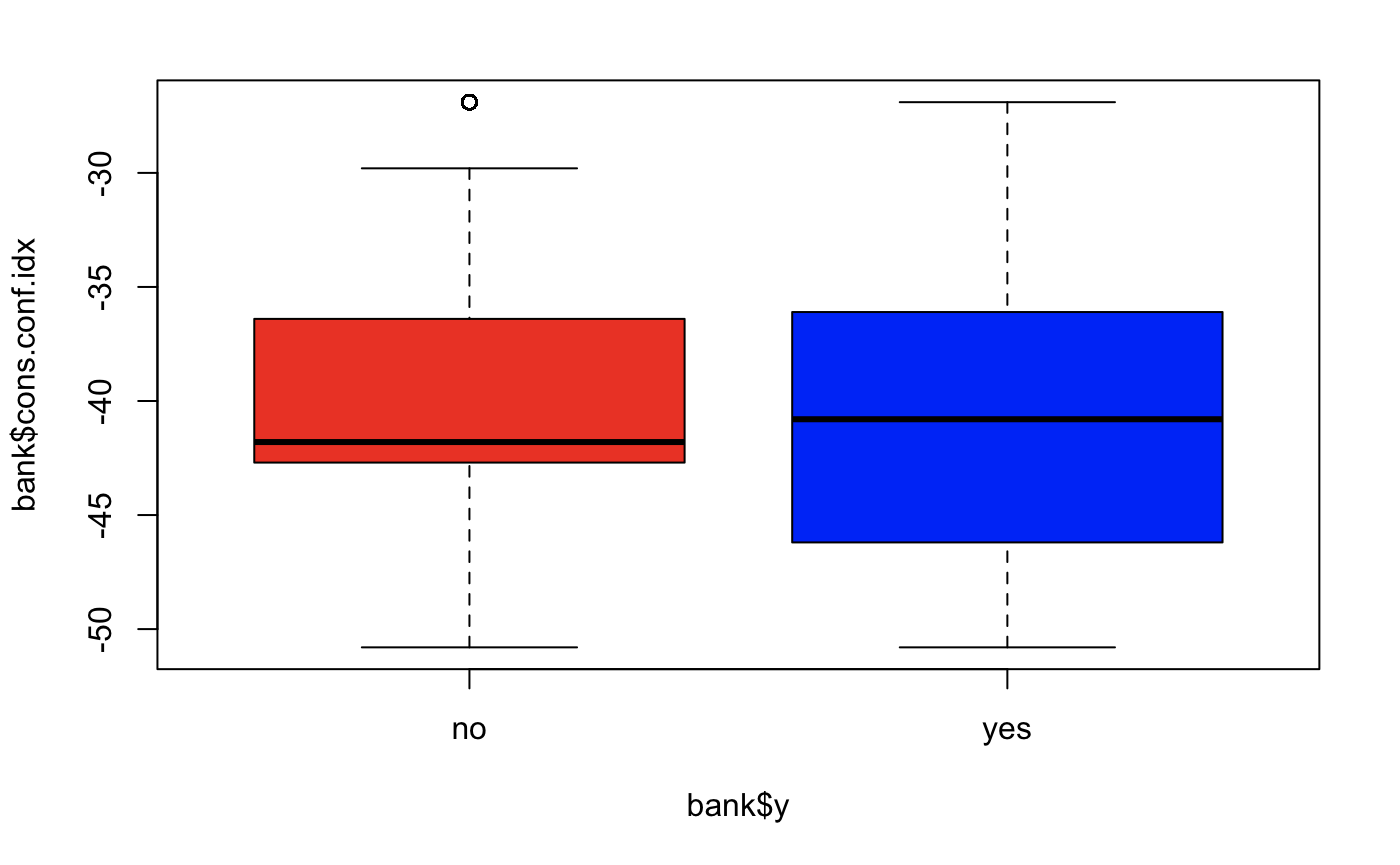
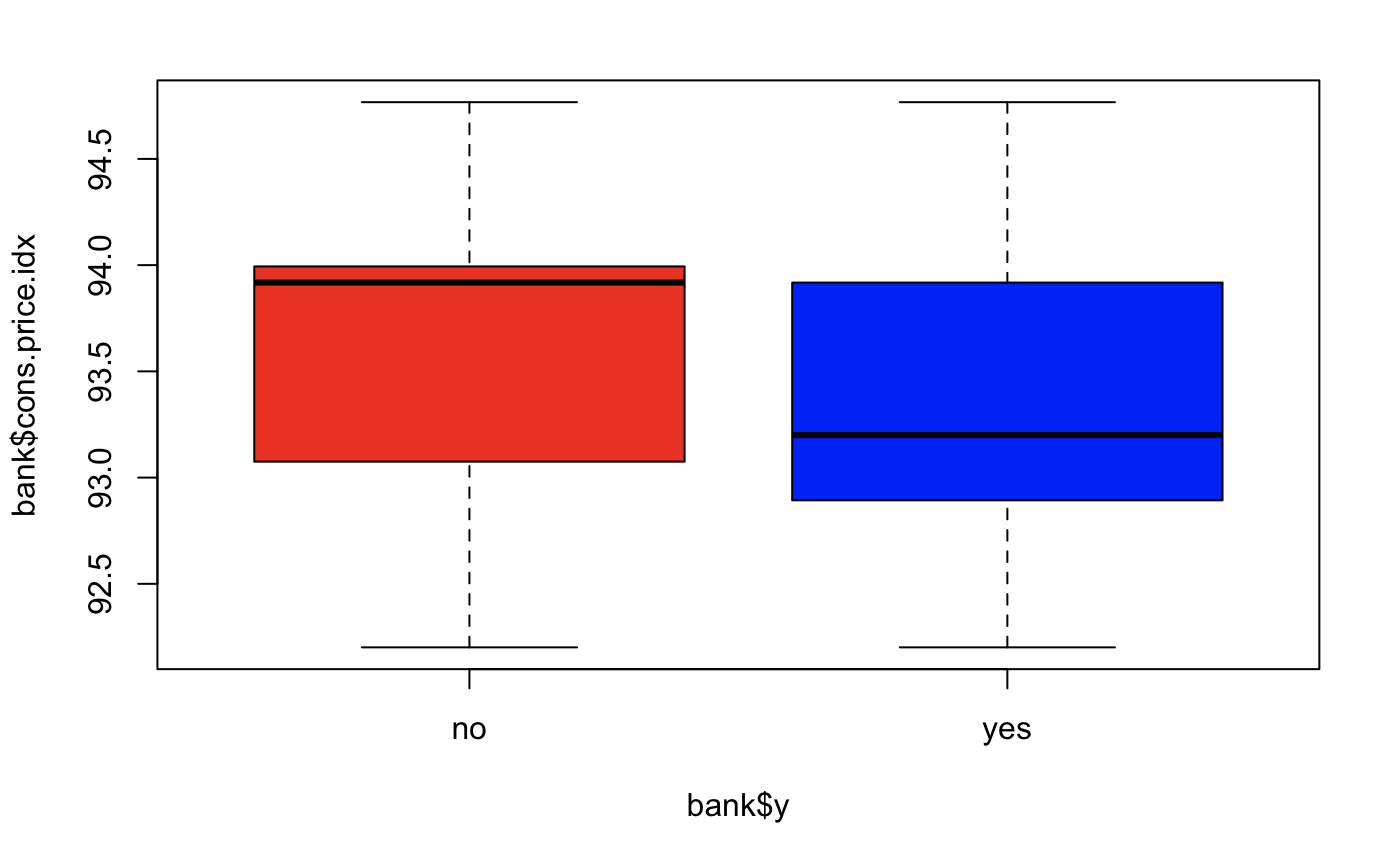
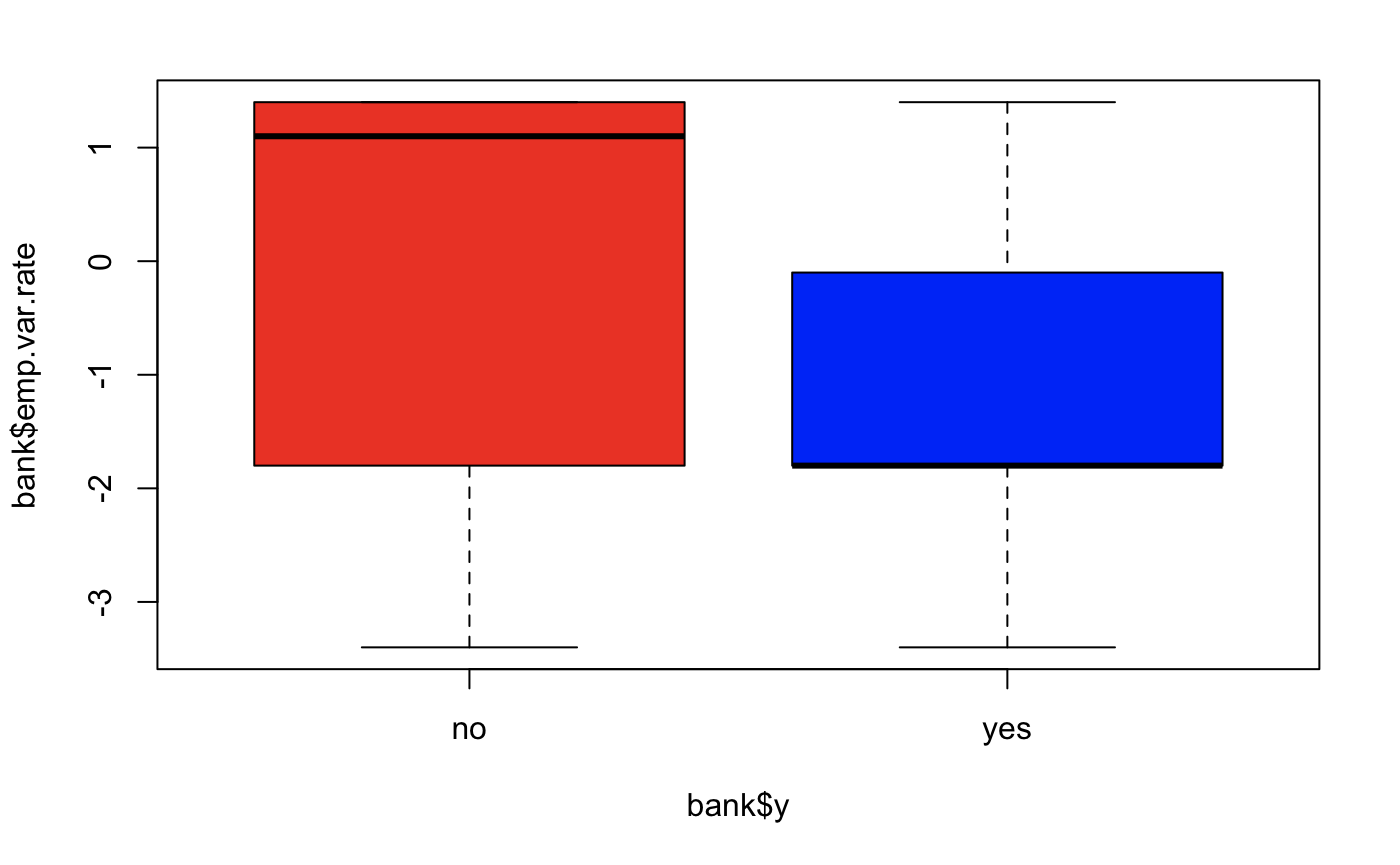
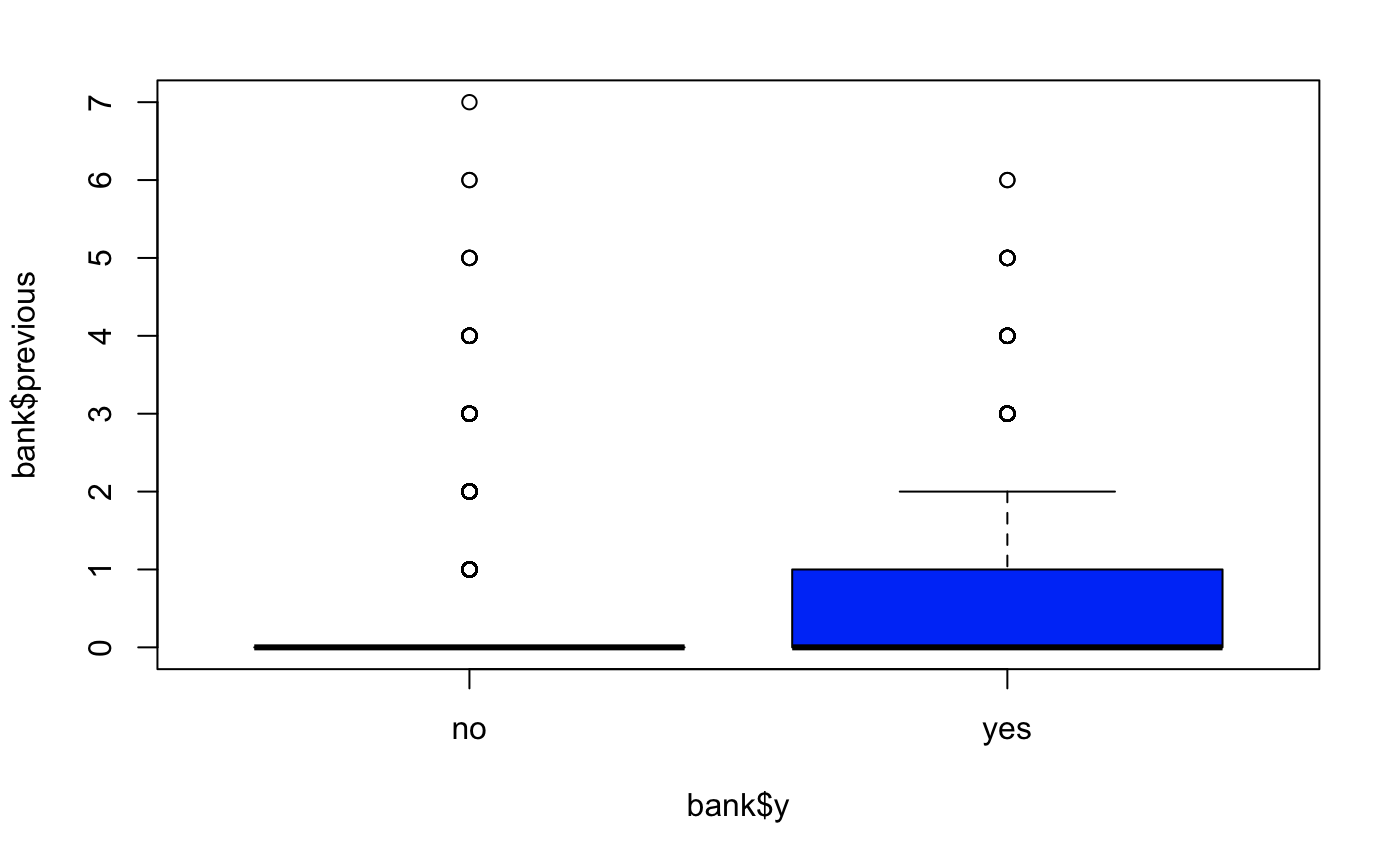
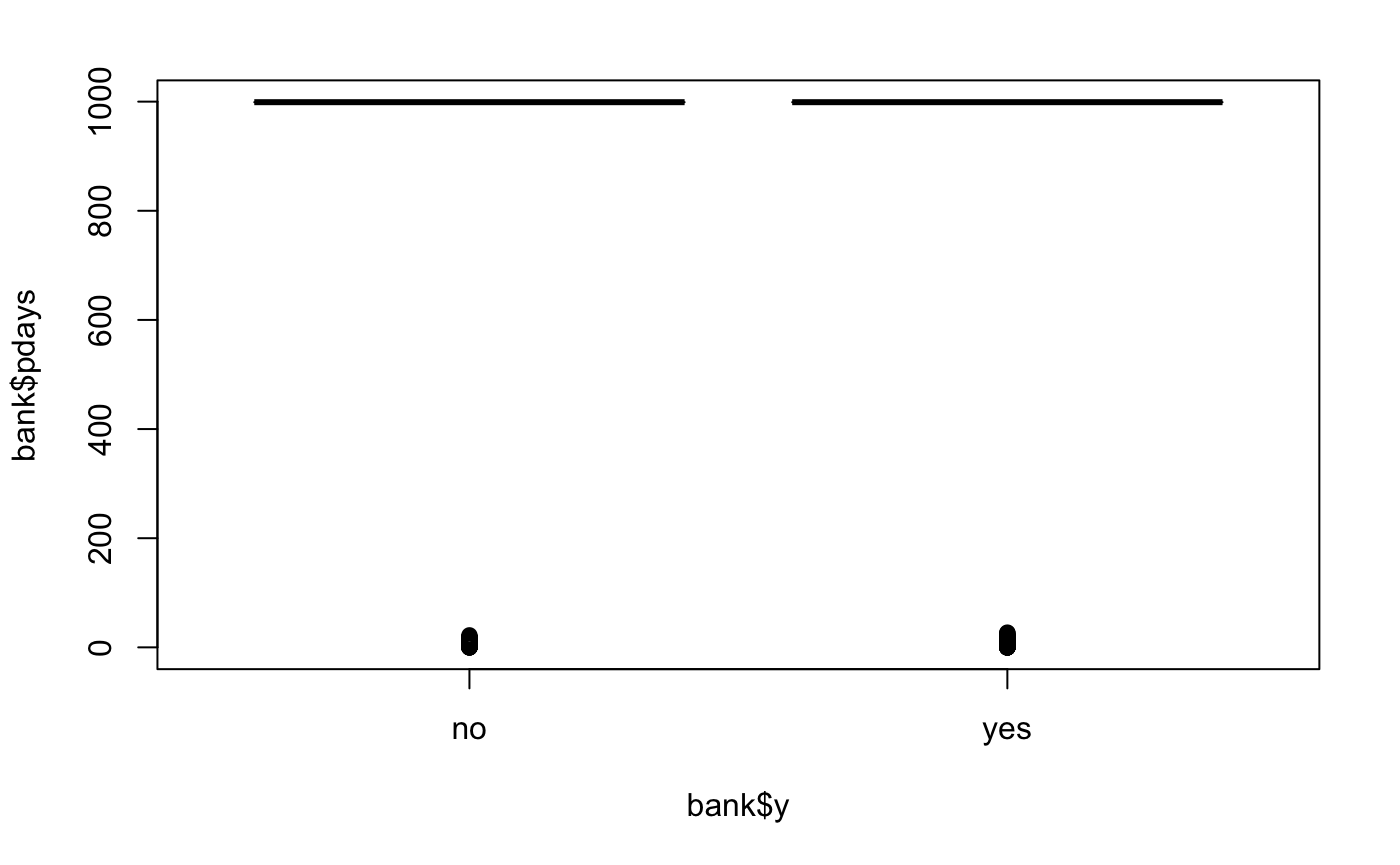
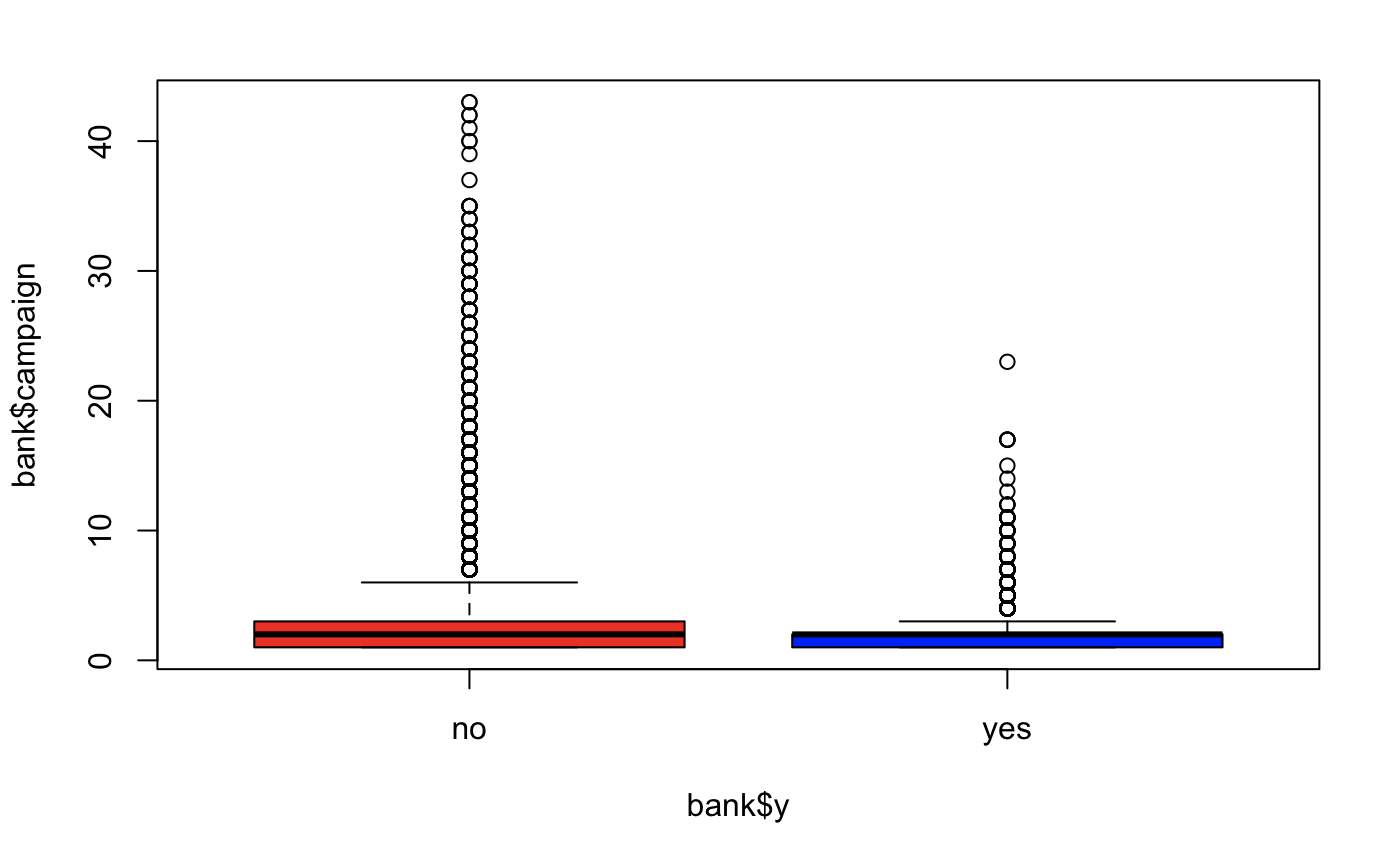
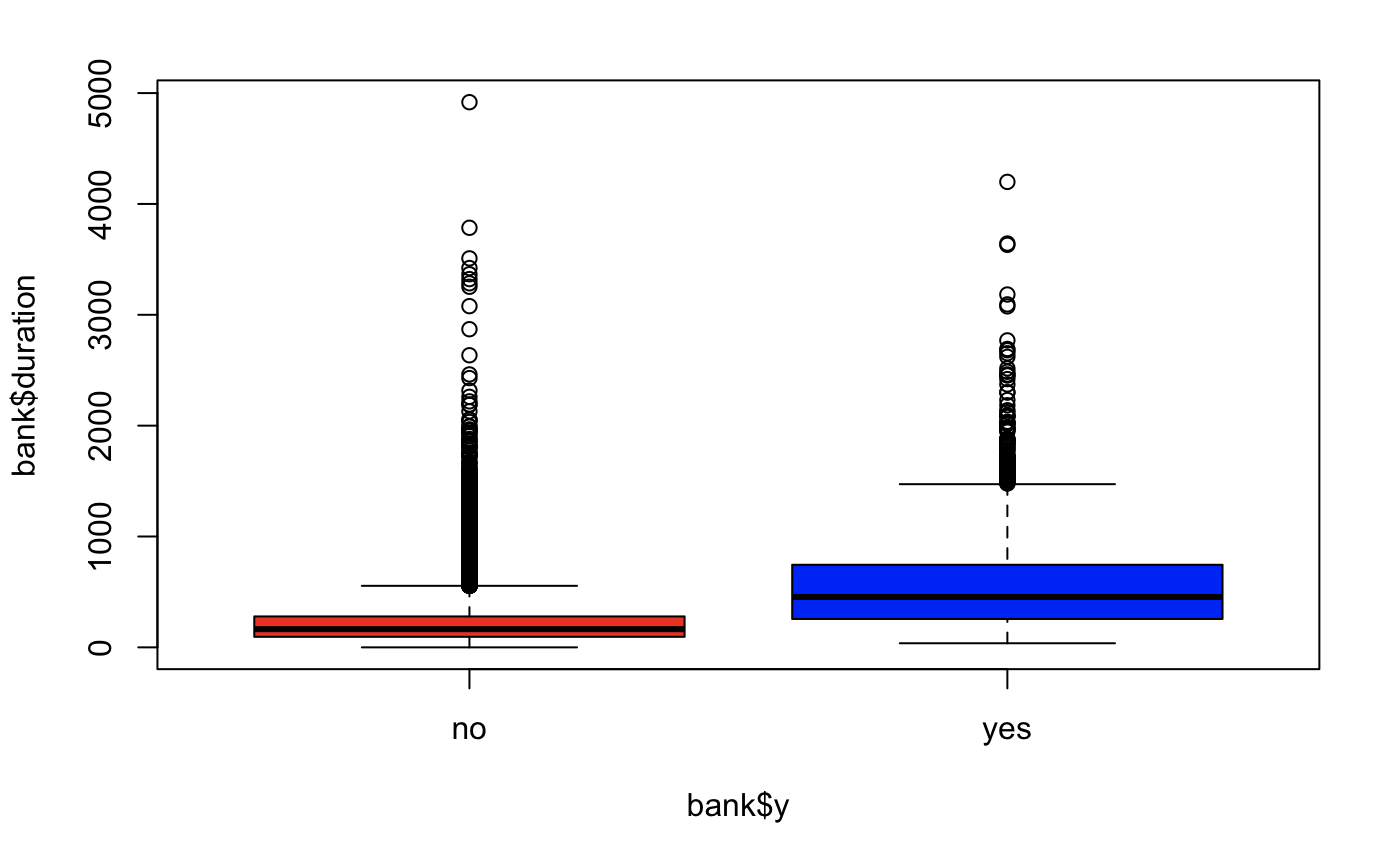
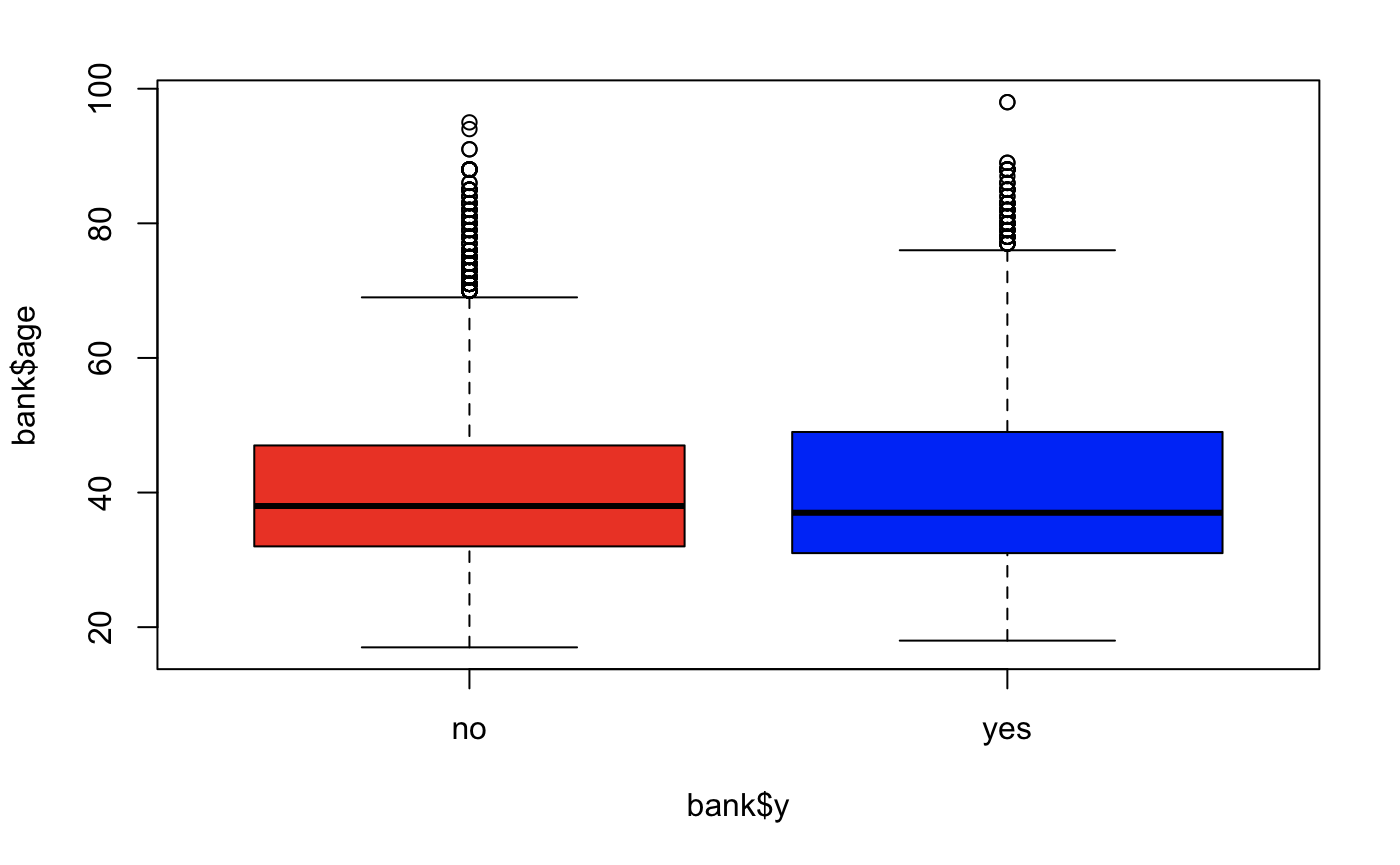


### Missing Values

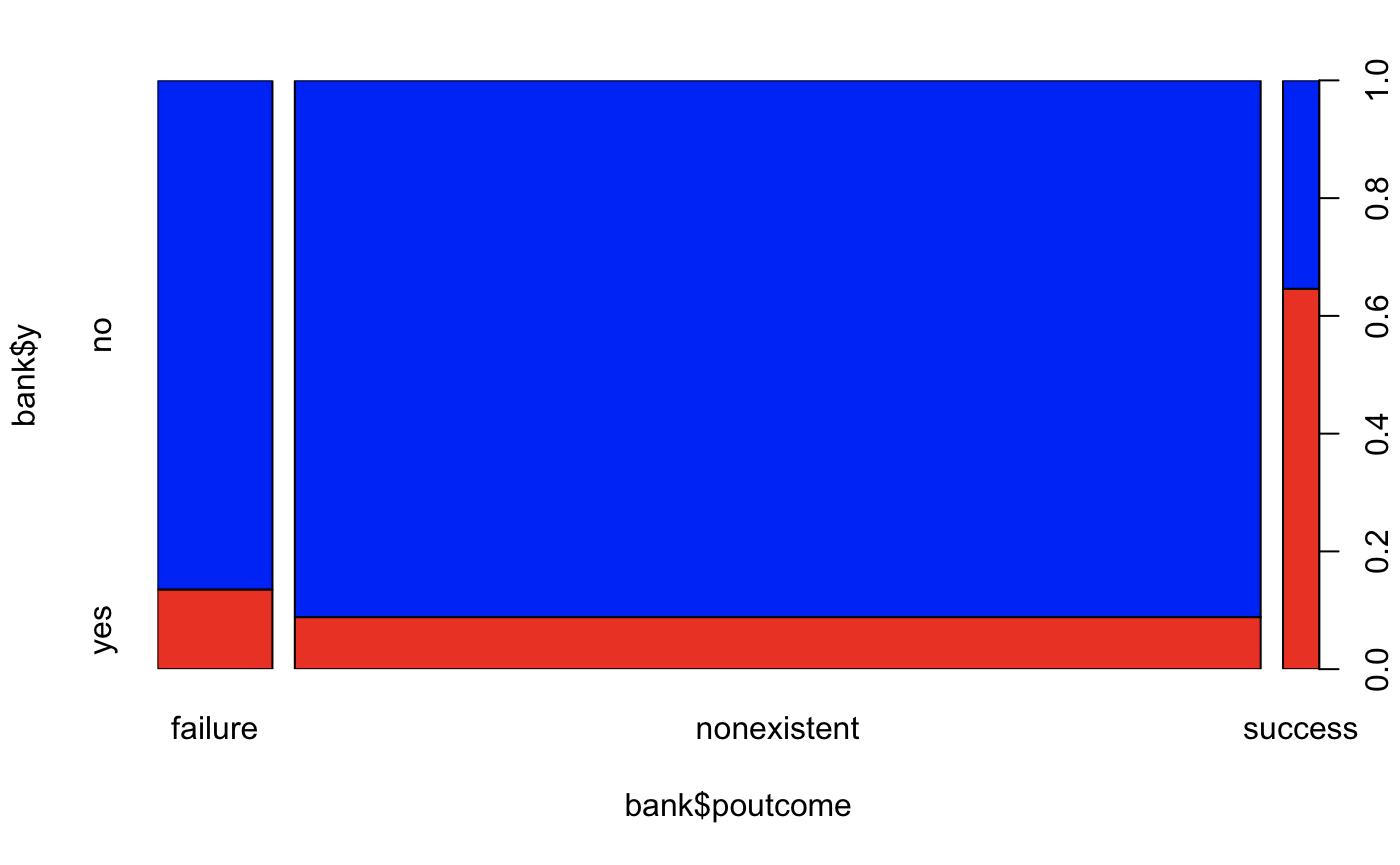
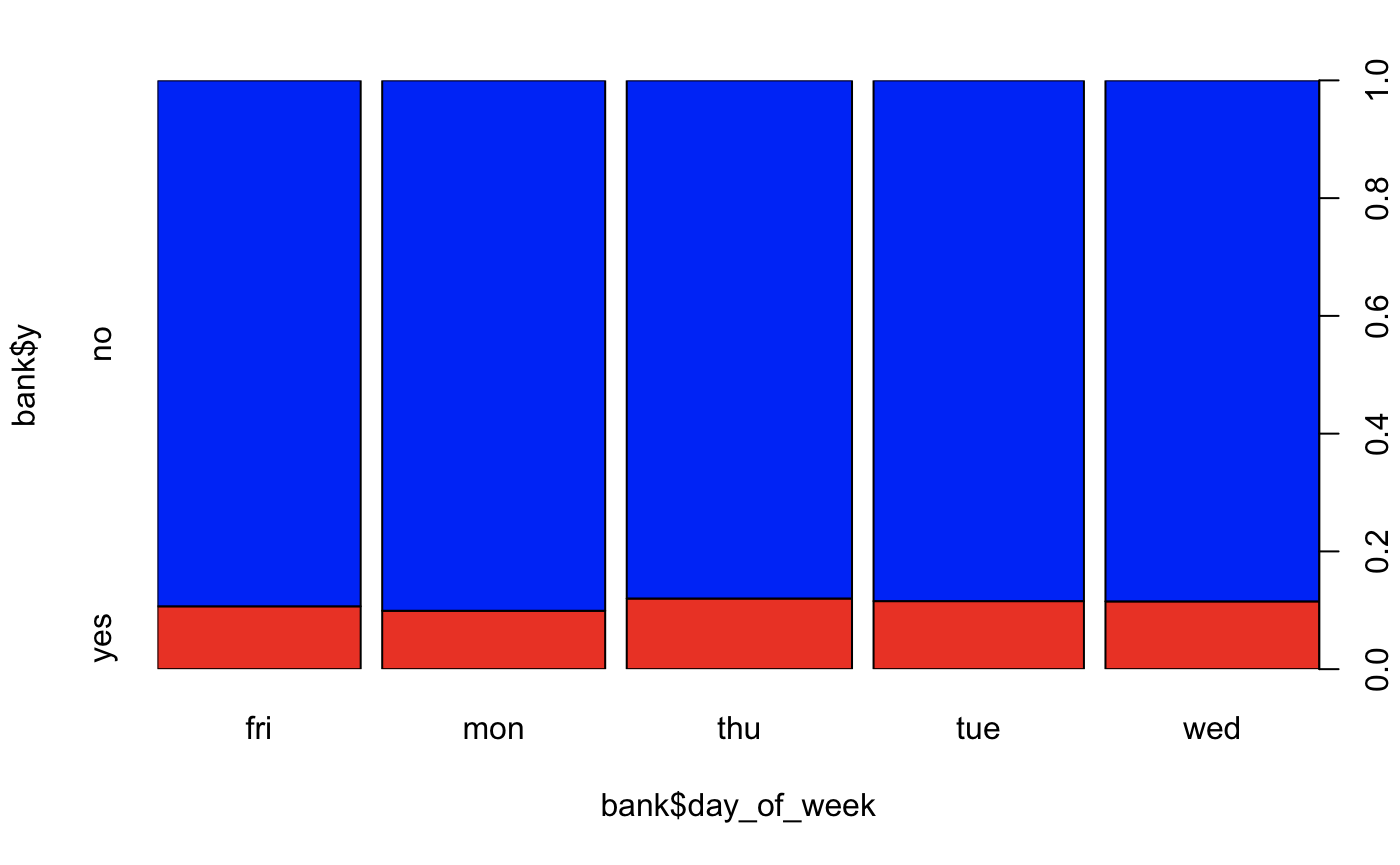
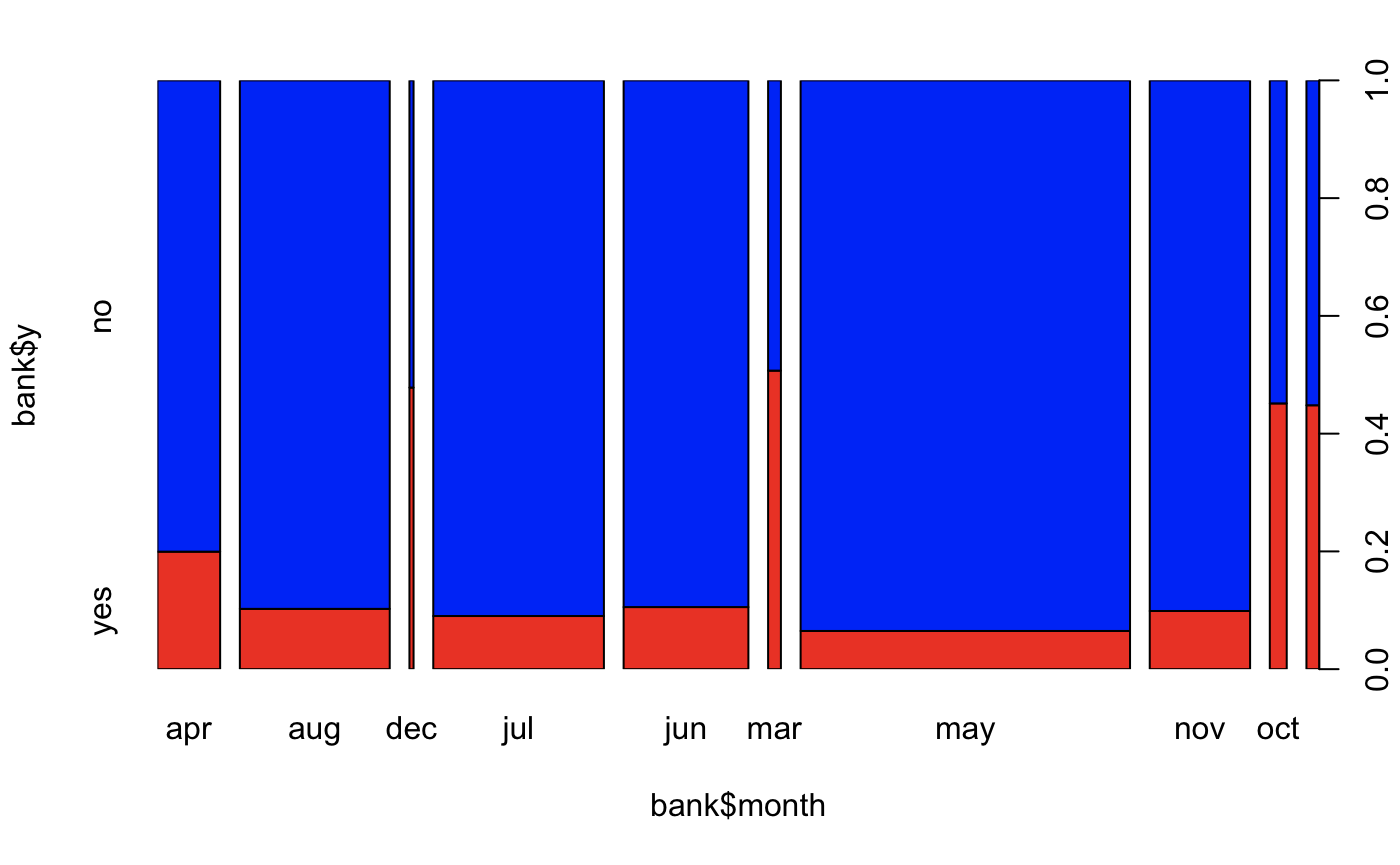
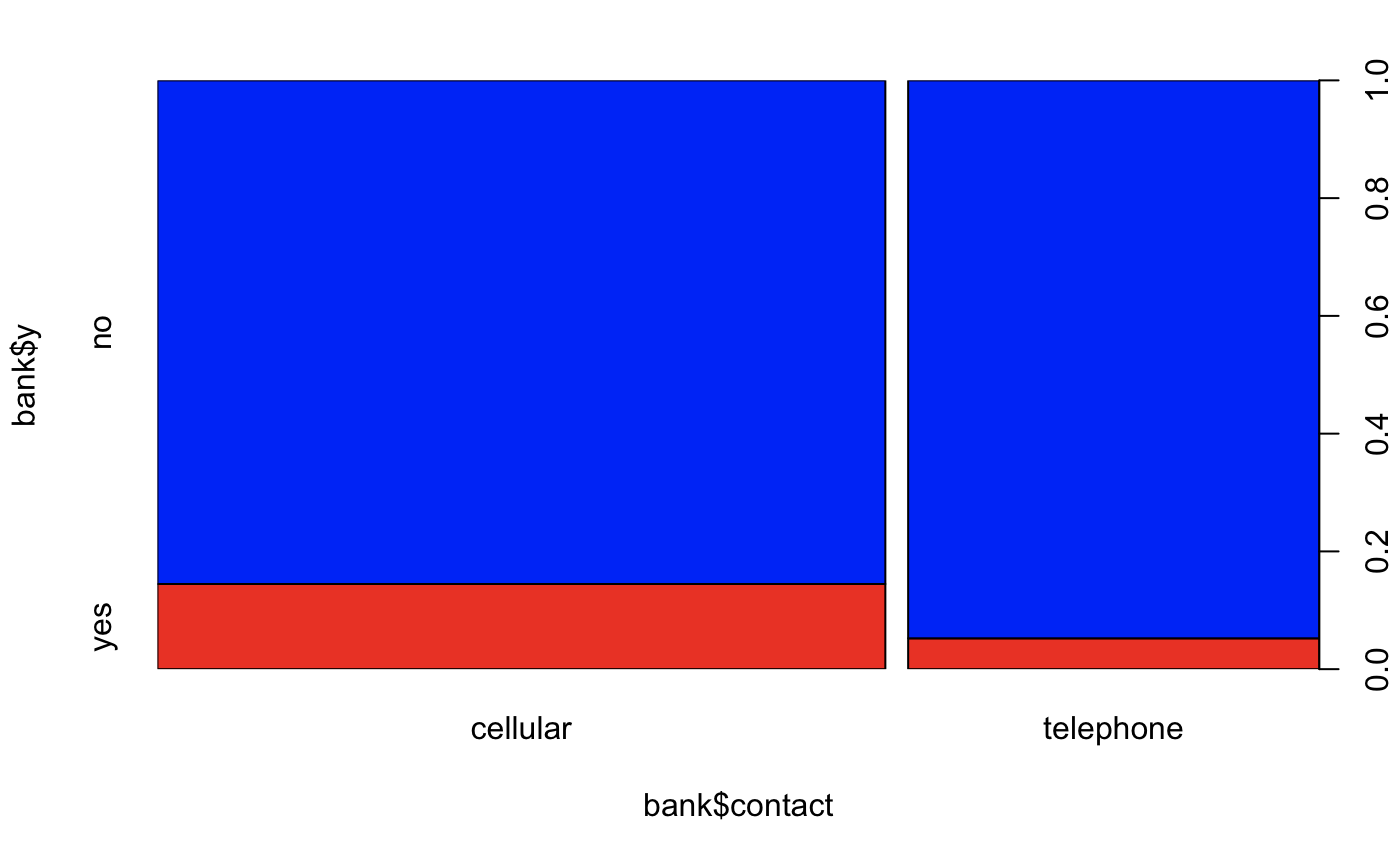
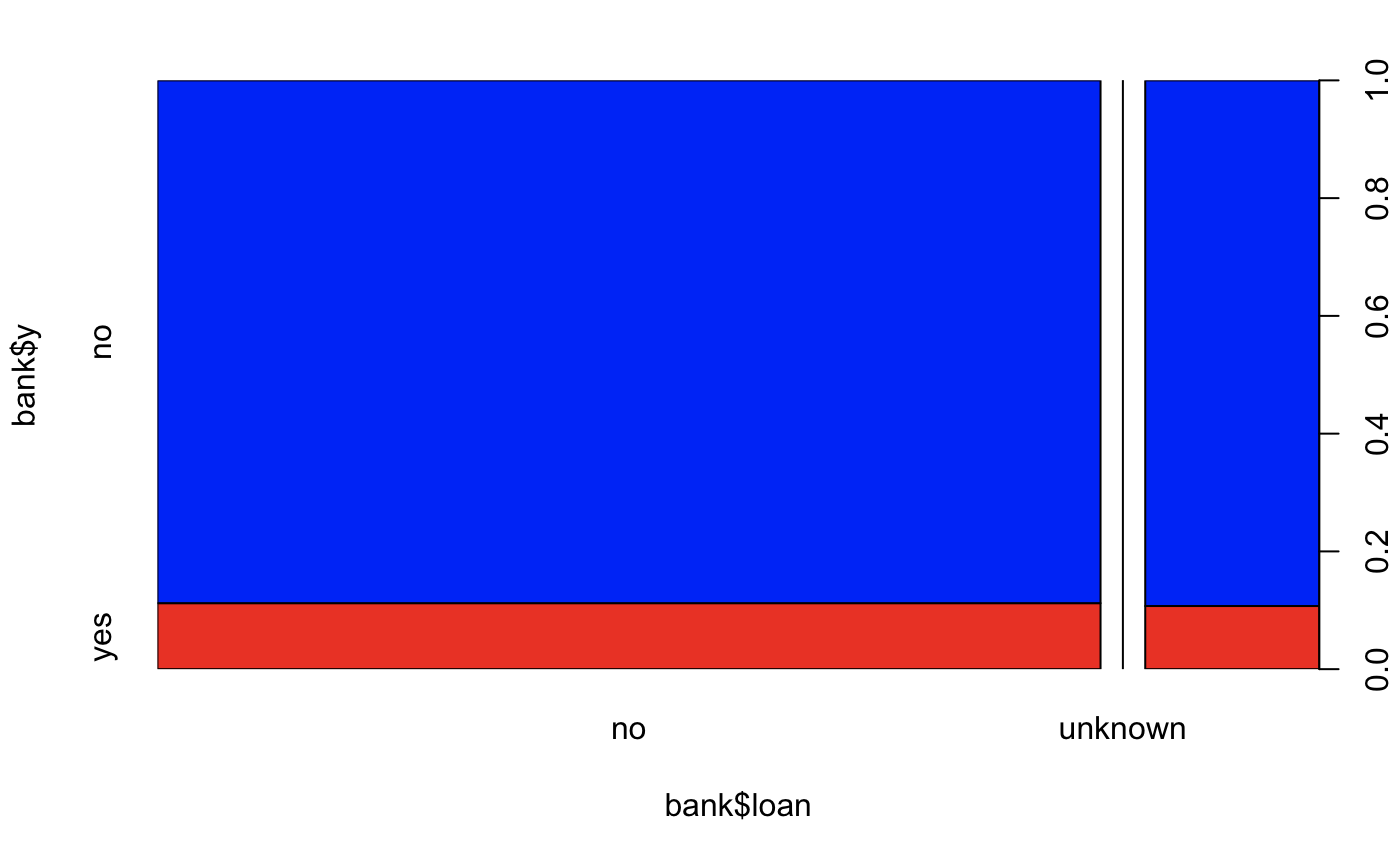
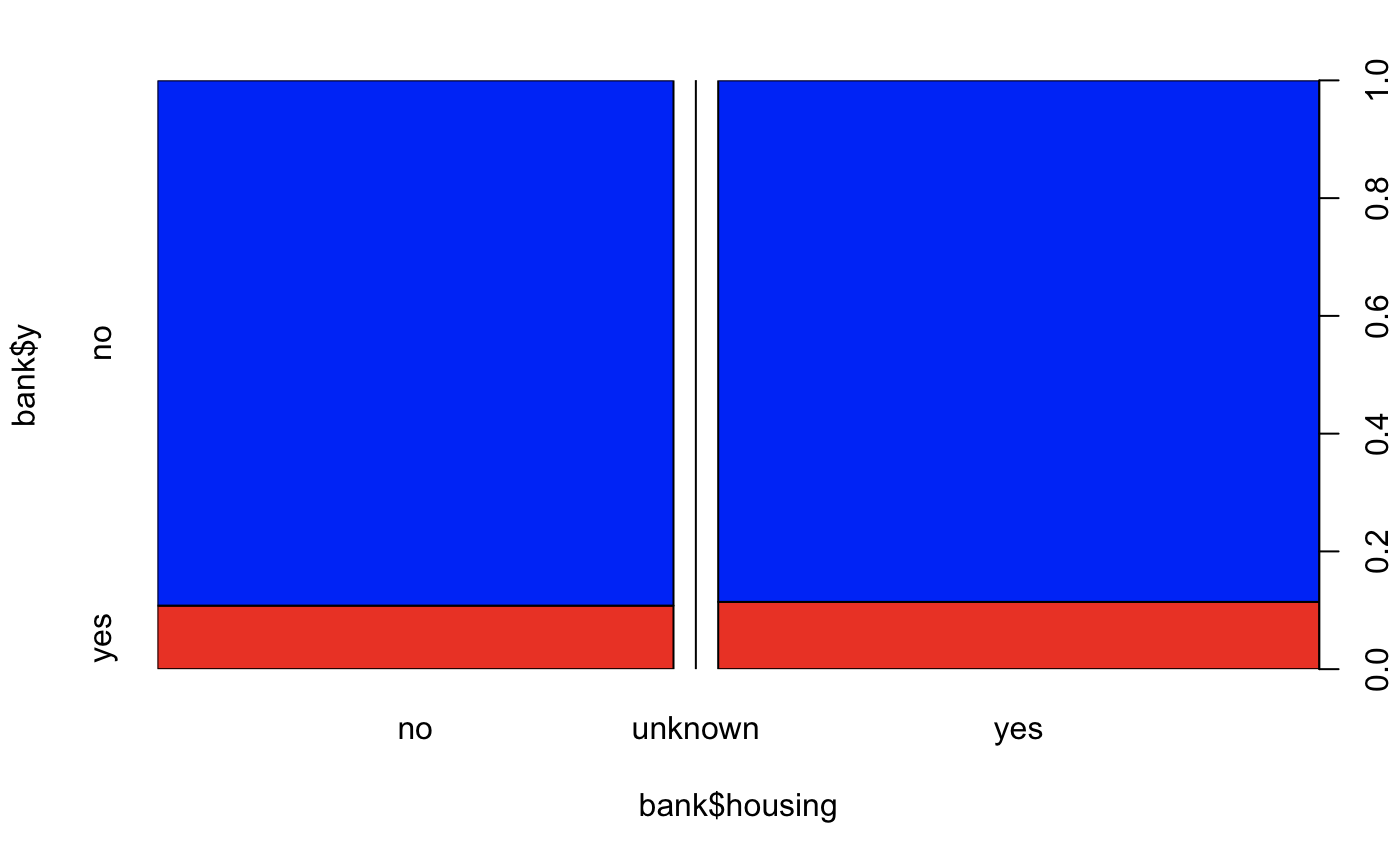
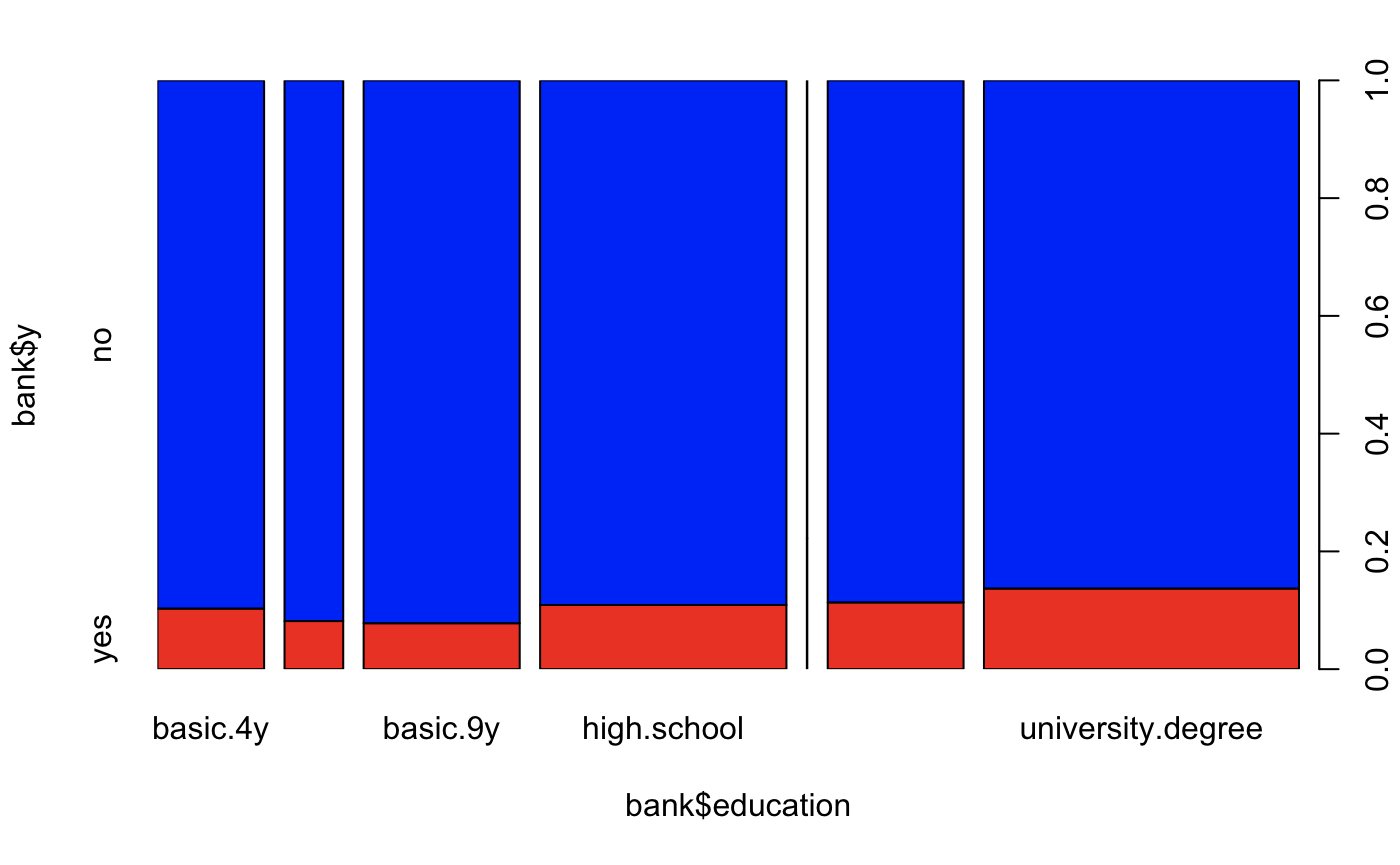
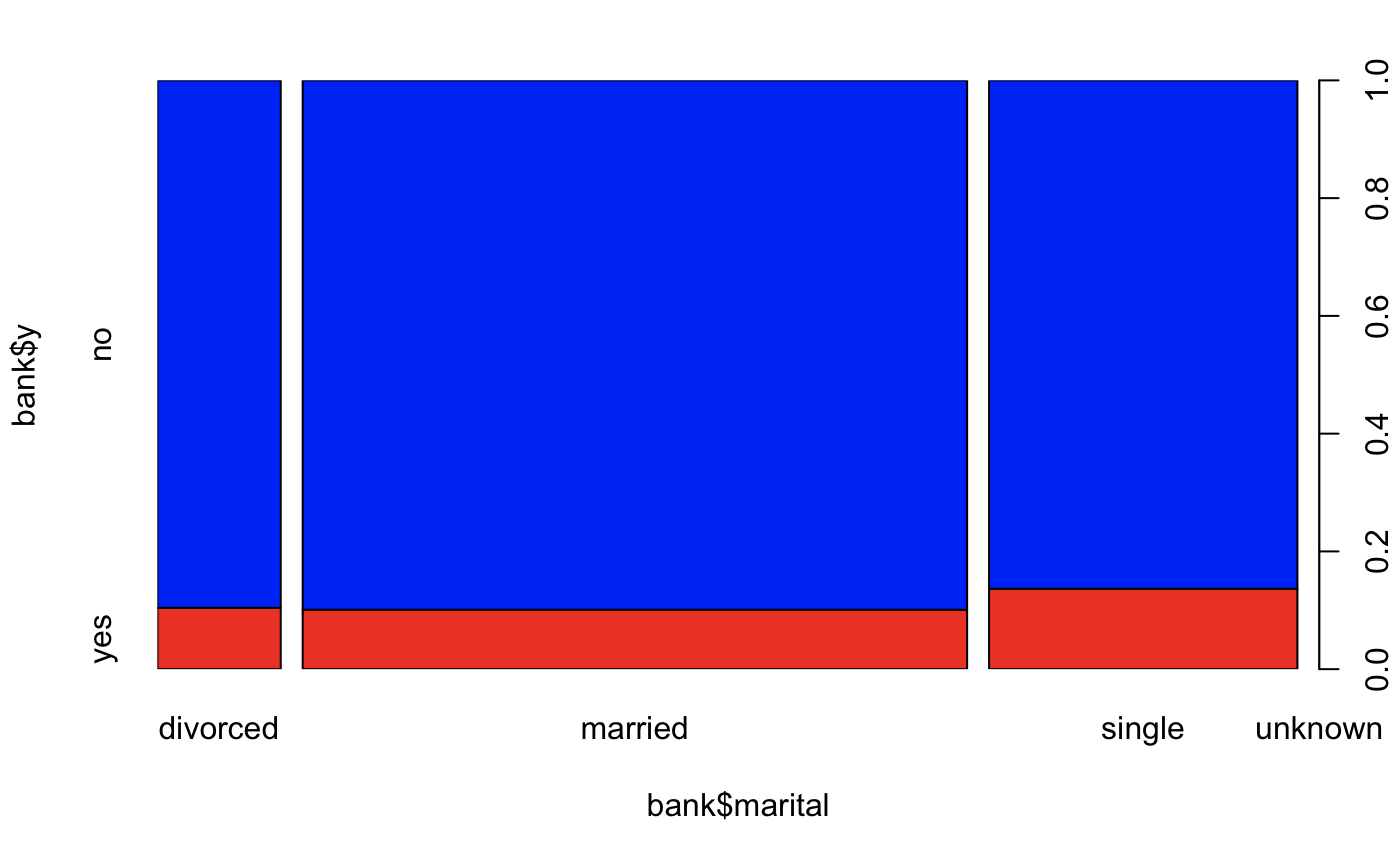
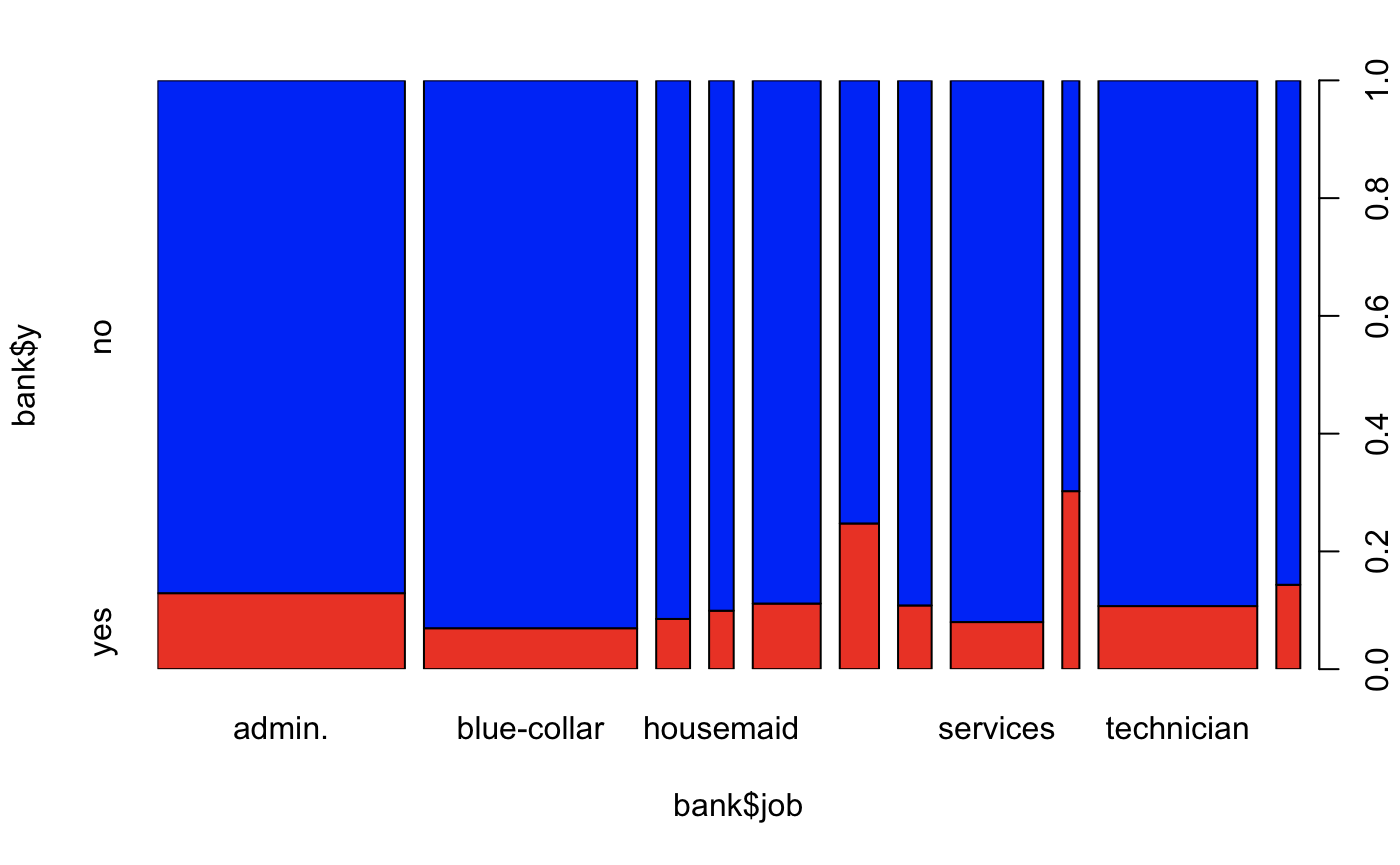
****

### Continuous Predictors

| > aggregate(age~y, data=bank, summary) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| y | age.Min. | age.1stQu. | age.Median | age.Mean | age.3rdQu. | age.Max. |
| 1 no | 17 | 32 | 38 | 39.75077 | 47 | 95 |
| 2 yes | 18 | 31 | 37 | 40.73573 | 49 | 98 |
| > aggregate(duration~y, data=bank, summary) | | | | | | |
| y | duration.Min. | duration.1stQu. | duration.Median | duration.Mean | duration.3rdQu. | duration.Max. |
| 1 no | 0 | 95 | 164 | 220.9111 | 279 | 4918 |
| 2 yes | 37 | 256 | 454 | 556.1454 | 745 | 4199 |
| > aggregate(campaign~y, data=bank, summary) | | | | | | |
| y | campaign.Min. | campaign.1stQu. | campaign.Median | campaign.Mean | campaign.3rdQu. | campaign.Max. |
| 1 no | 1 | 1 | 2 | 2.630721 | 3 | 43 |
| 2 yes | 1 | 1 | 2 | 2.057787 | 2 | 23 |
| > aggregate(pdays~y, data=bank, summary) | | | | | | |
| y | pdays.Min. | pdays.1stQu. | pdays.Median | pdays.Mean | pdays.3rdQu. | pdays.Max. |
| 1 no | 0 | 999 | 999 | 984.359 | 999 | 999 |
| 2 yes | 0 | 999 | 999 | 797.2079 | 999 | 999 |
| > aggregate(previous~y, data=bank, summary) | | | | | | |
| y | previous.Min. | previous.1stQu. | previous.Median | previous.Mean | previous.3rdQu. | previous.Max. |
| 1 no | 0 | 0 | 0 | 0.1319128 | 0 | 7 |
| 2 yes | 0 | 0 | 0 | 0.4745126 | 1 | 6 |
| > aggregate(emp.var.rate~y, data=bank, summary) | | | | | | |
| y | emp.var.rate.Min. | emp.var.rate.1stQu. | emp.var.rate.Median | emp.var.rate.Mean | emp.var.rate.3rdQu. | emp.var.rate.Max. |
| 1 no | -3.4 | -1.8 | 1.1 | 0.2448951 | 1.4 | 1.4 |
| 2 yes | -3.4 | -1.8 | -1.8 | -1.2097486 | -0.1 | 1.4 |
| > aggregate(cons.price.idx~y, data=bank, summary) | | | | | | |
| y | cons.price.idx.Min. | cons.price.idx.1stQu. | cons.price.idx.Median | cons.price.idx.Mean | cons.price.idx.3rdQu. | cons.price.idx.Max. |
| 1 no | 92.201 | 93.075 | 93.918 | 93.59751 | 93.994 | 94.767 |
| 2 yes | 92.201 | 92.893 | 93.2 | 93.35381 | 93.918 | 94.767 |
| > aggregate(cons.conf.idx~y, data=bank, summary) | | | | | | |
| y | cons.conf.idx.Min. | cons.conf.idx.1stQu. | cons.conf.idx.Median | cons.conf.idx.Mean | cons.conf.idx.3rdQu. | cons.conf.idx.Max. |
| 1 no | -50.8 | -42.7 | -41.8 | -40.62553 | -36.4 | -26.9 |
| 2 yes | -50.8 | -46.2 | -40.8 | -39.87097 | -36.1 | -26.9 |
| > aggregate(euribor3m~y, data=bank, summary) | | | | | | |
| y | euribor3m.Min. | euribor3m.1stQu. | euribor3m.Median | euribor3m.Mean | euribor3m.3rdQu. | euribor3m.Max. |
| 1 no | 0.634 | 1.405 | 4.857 | 3.807399 | 4.962 | 5.045 |
| 2 yes | 0.634 | 0.859 | 1.268 | 2.154251 | 4.855 | 5.045 |
| > aggregate(nr.employed~y, data=bank, summary) | | | | | | |
| y | nr.employed.Min. | nr.employed.1stQu. | nr.employed.Median | nr.employed.Mean | nr.employed.3rdQu. | nr.employed.Max. |
| 1 no | 4963.6 | 5099.1 | 5195.8 | 5176.268 | 5228.1 | 5228.1 |
| 2 yes | 4963.6 | 5017.5 | 5099.1 | 5096.92 | 5191 | 5228.1 |

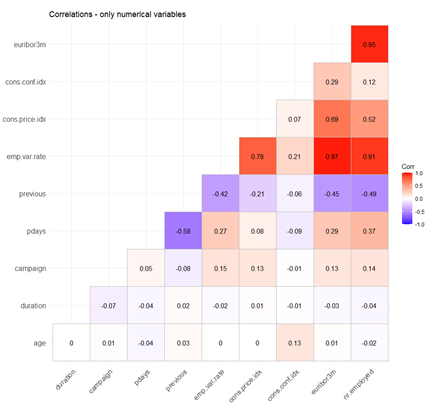
****

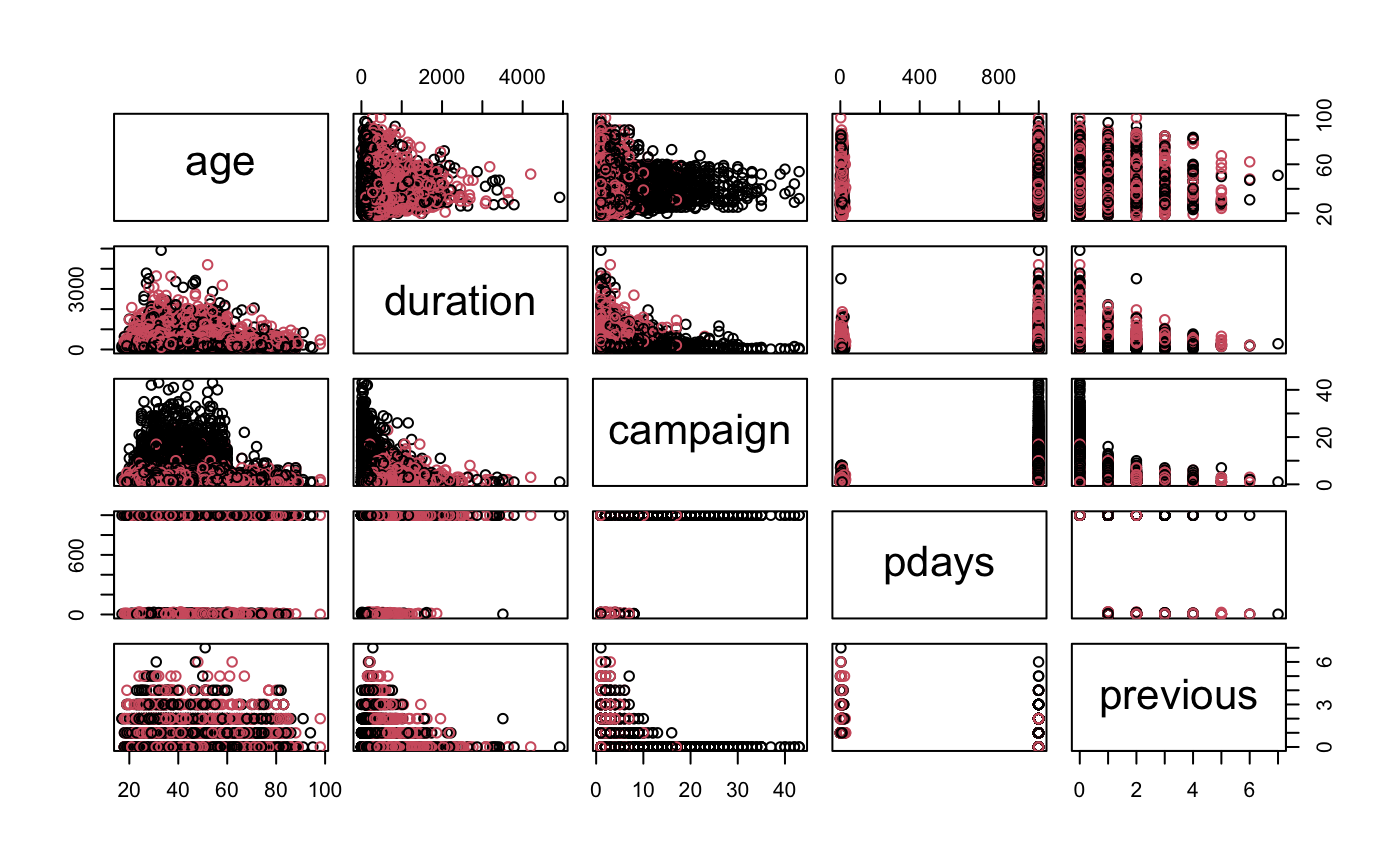
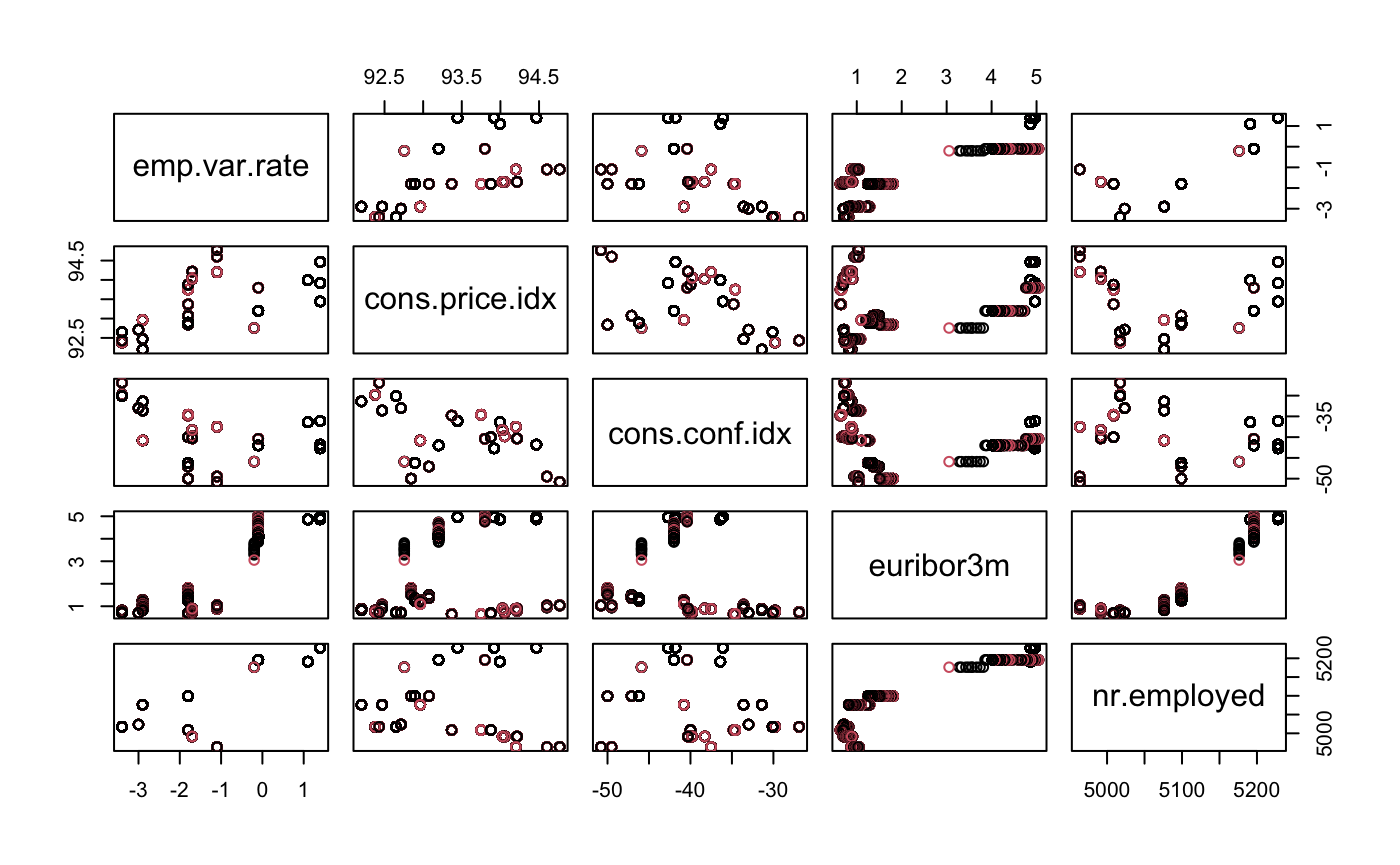
### Categorical Predictors

****

### Correlation

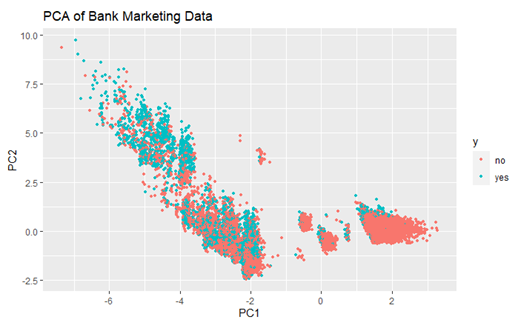
Image 2:



****

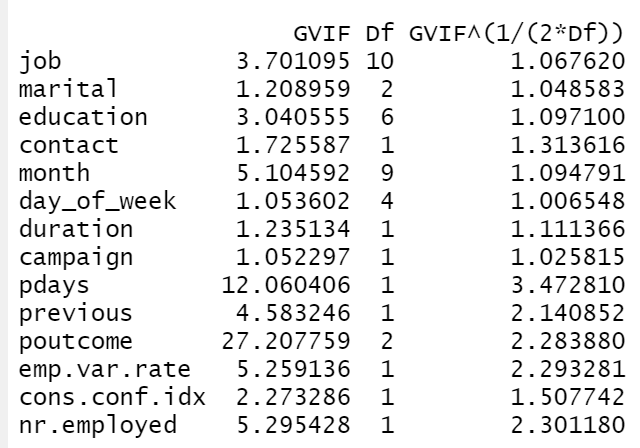
### PCA

Image 3:

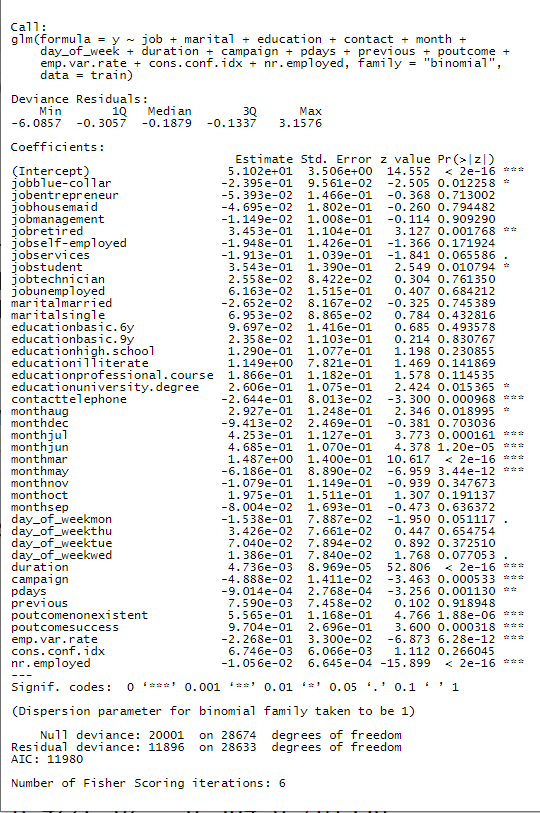


## Simple Lasso Model

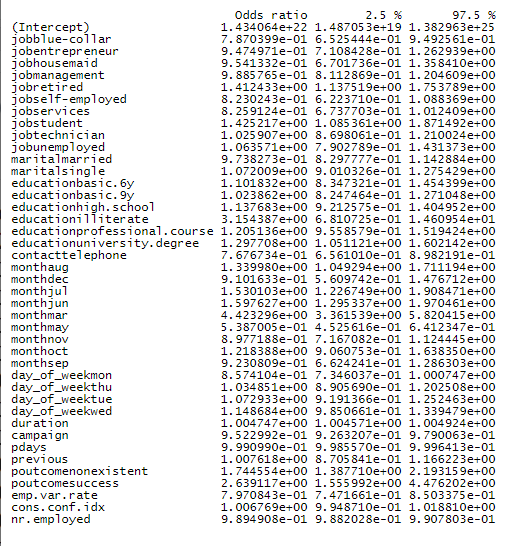
### VIF



### Coefficients summary

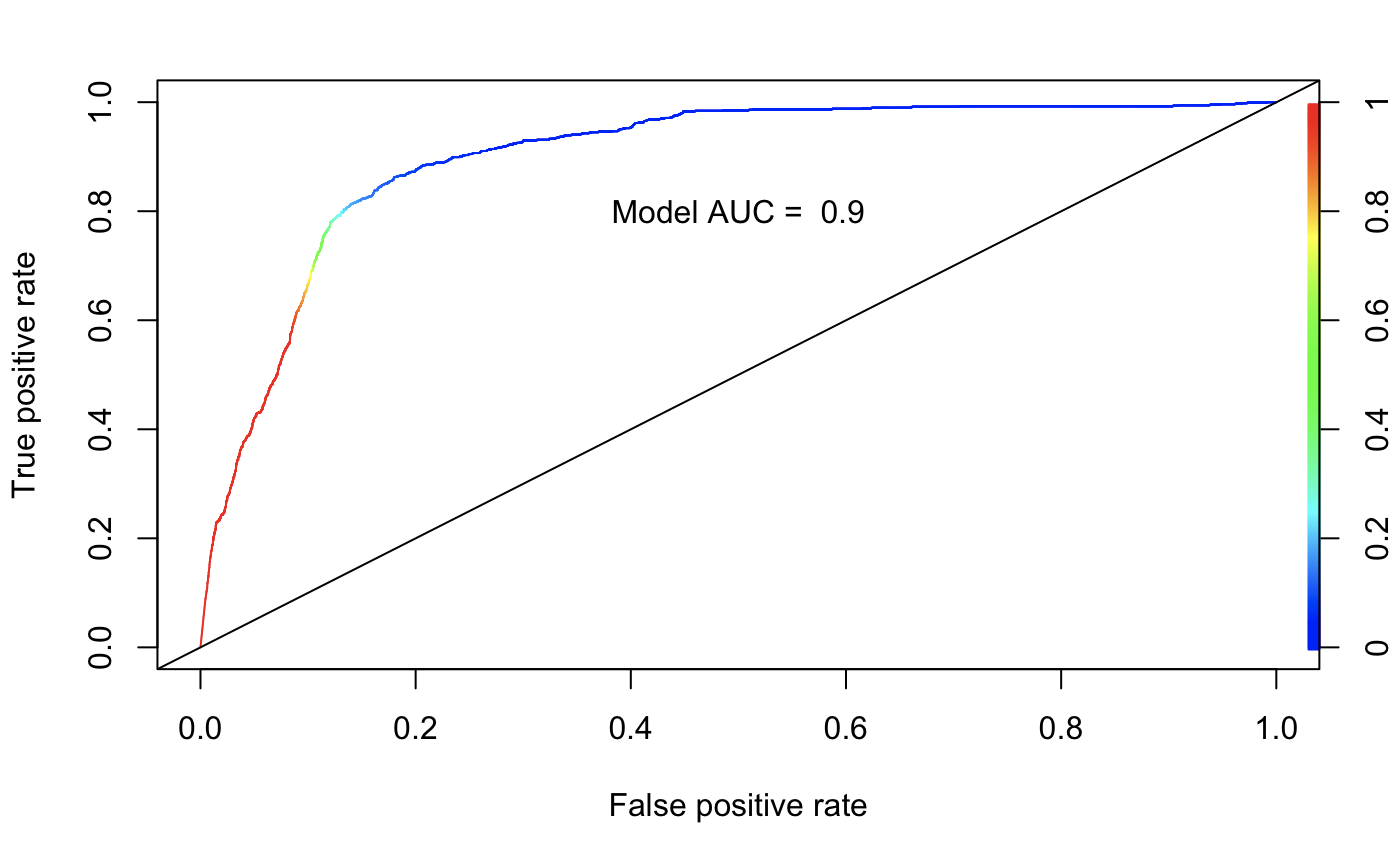


### Odds Ratio Table



## Advanced Lasso Model

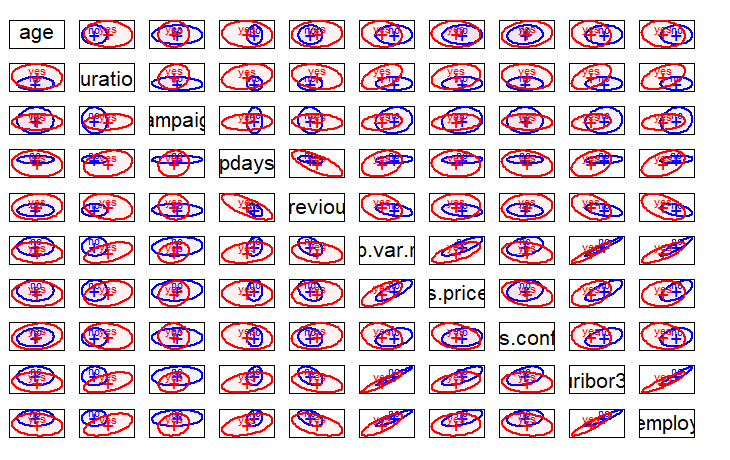
## QDA

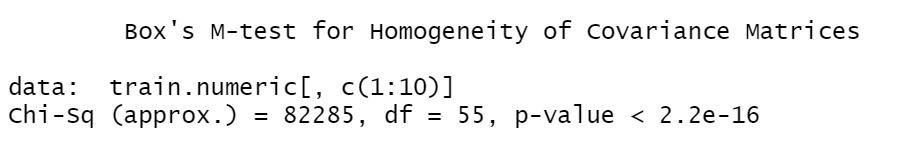
****

## LDA

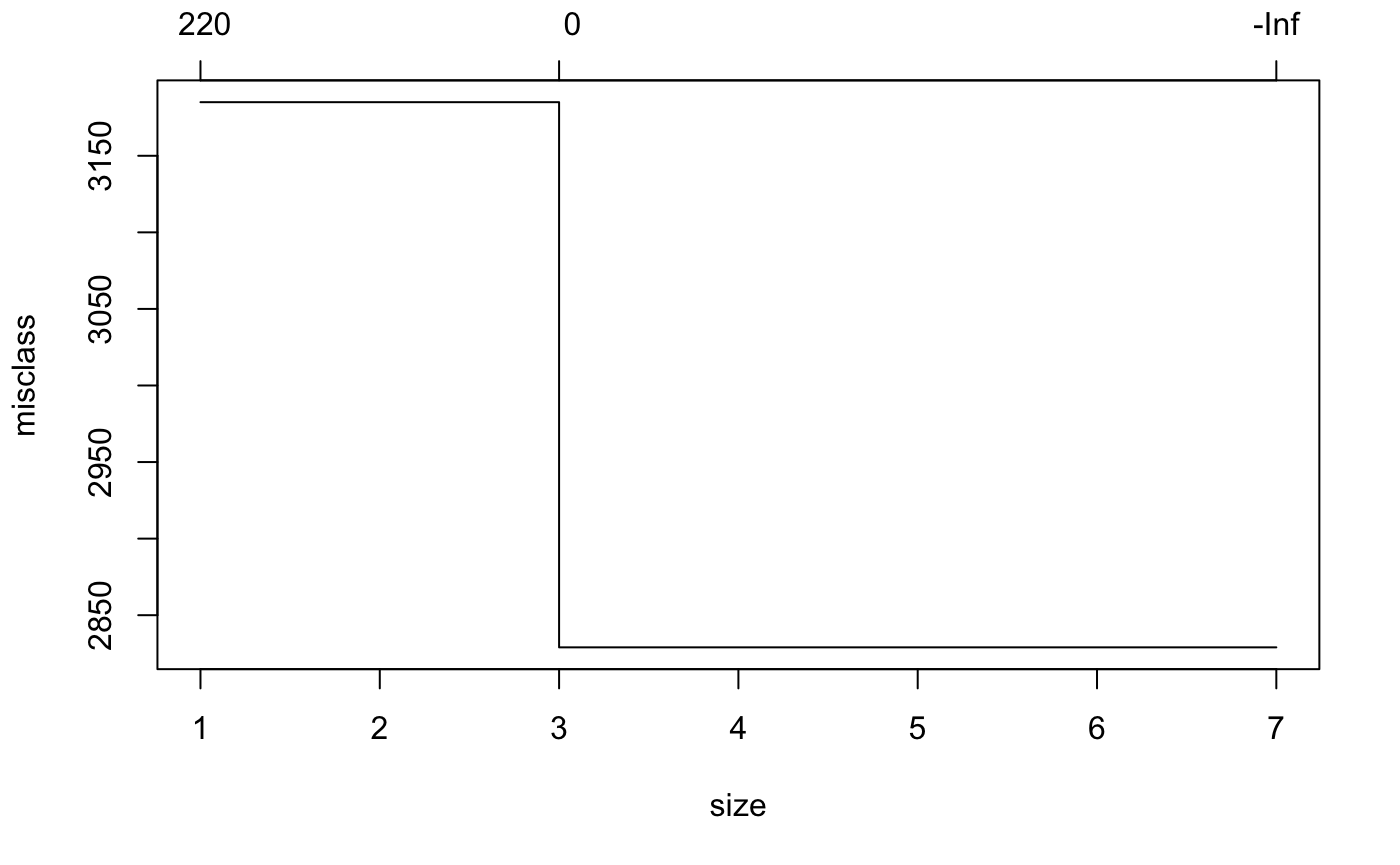
**Homogeneity of Covariance Matrices Test**

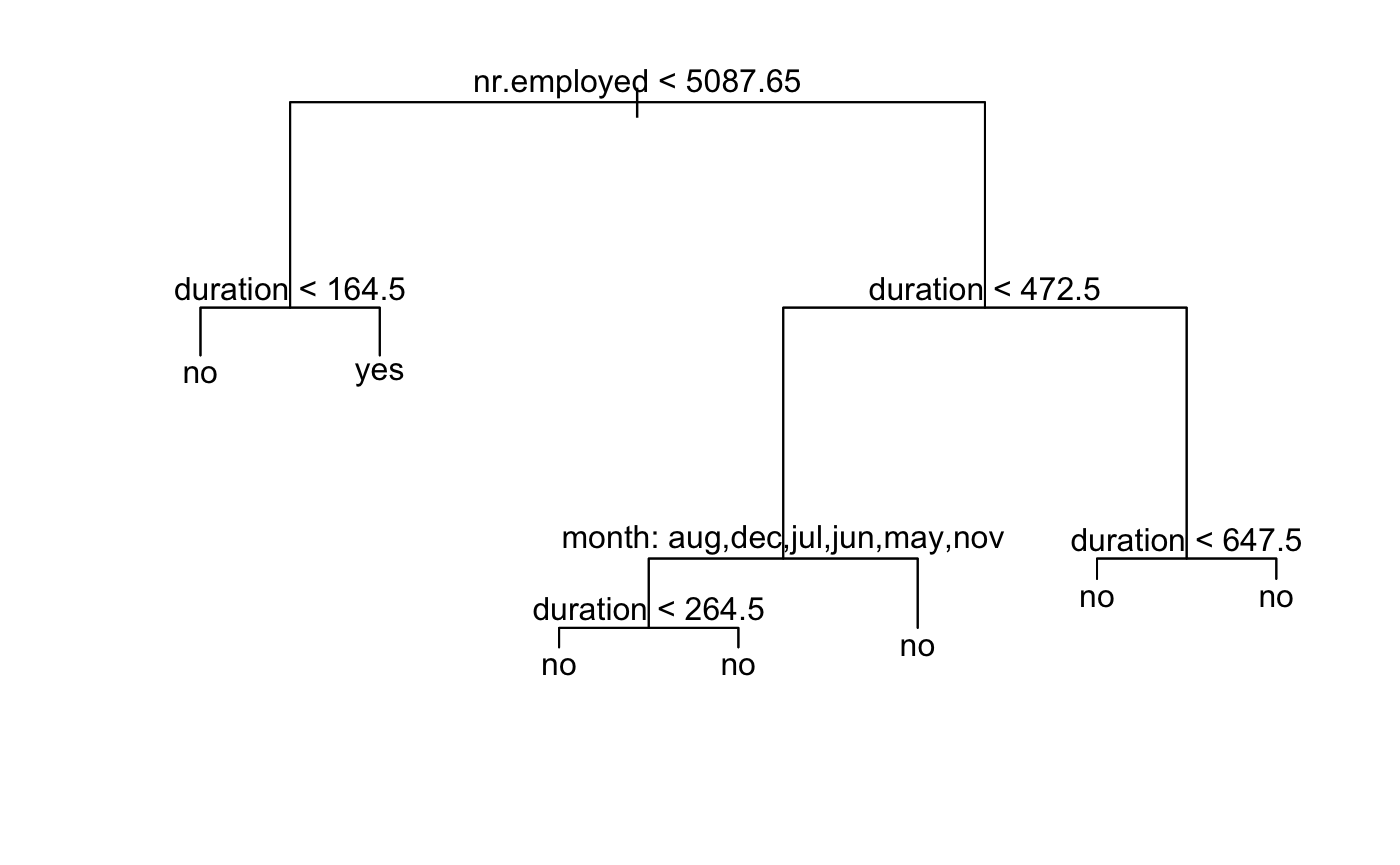
We cannot use LDA as the Covariance Matrix Homogeneity assumption does not holds

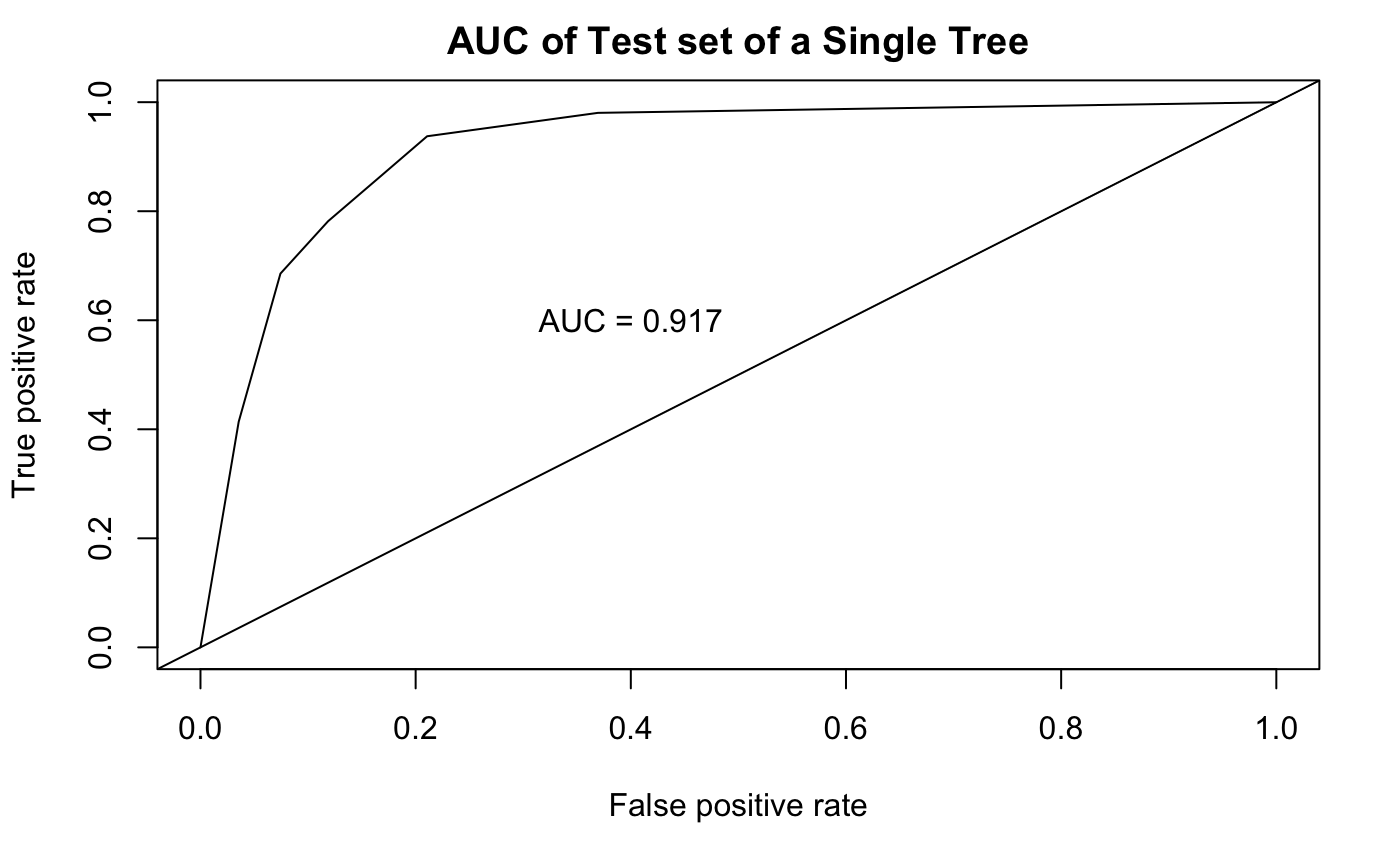




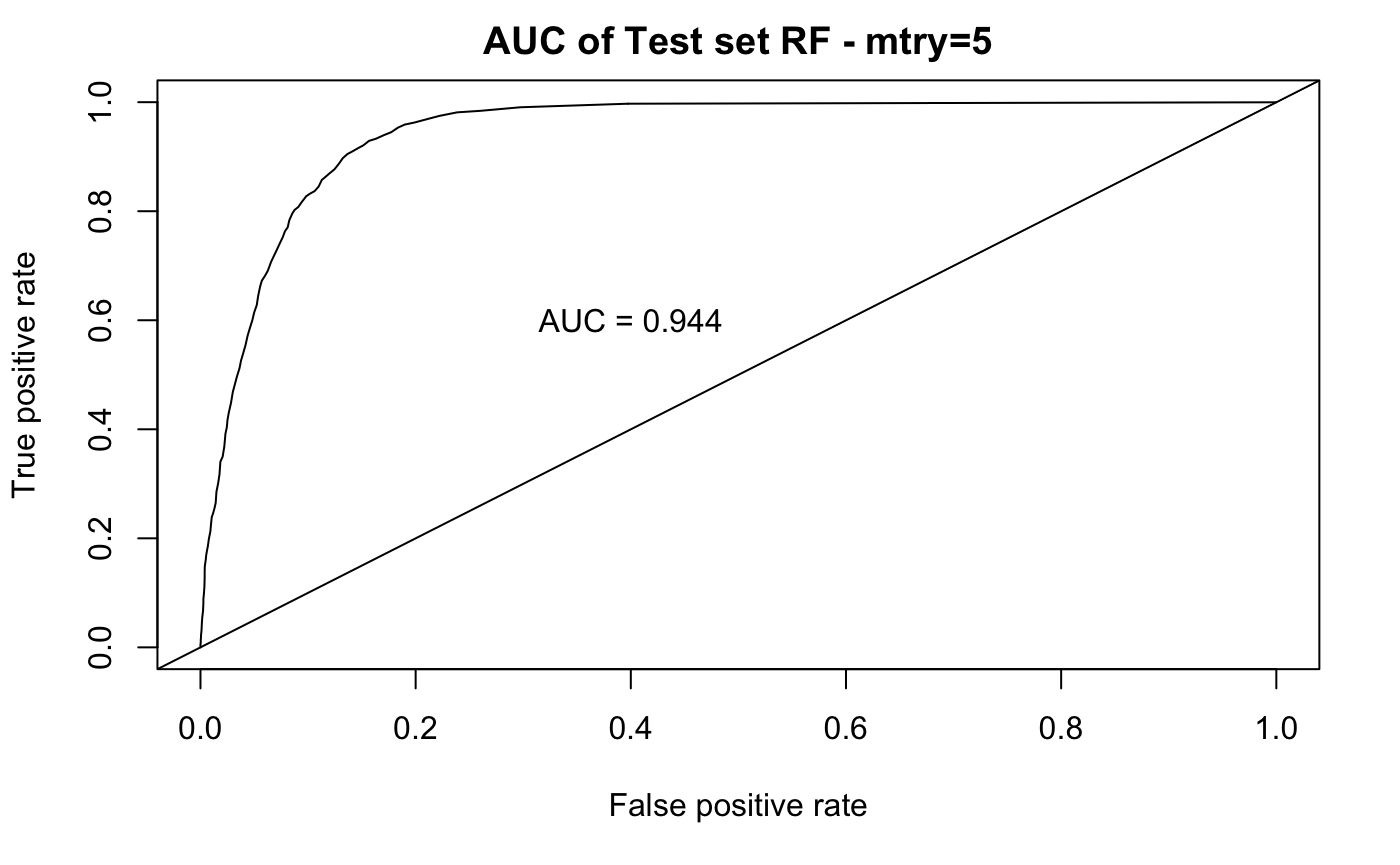
## Decision Tree Model

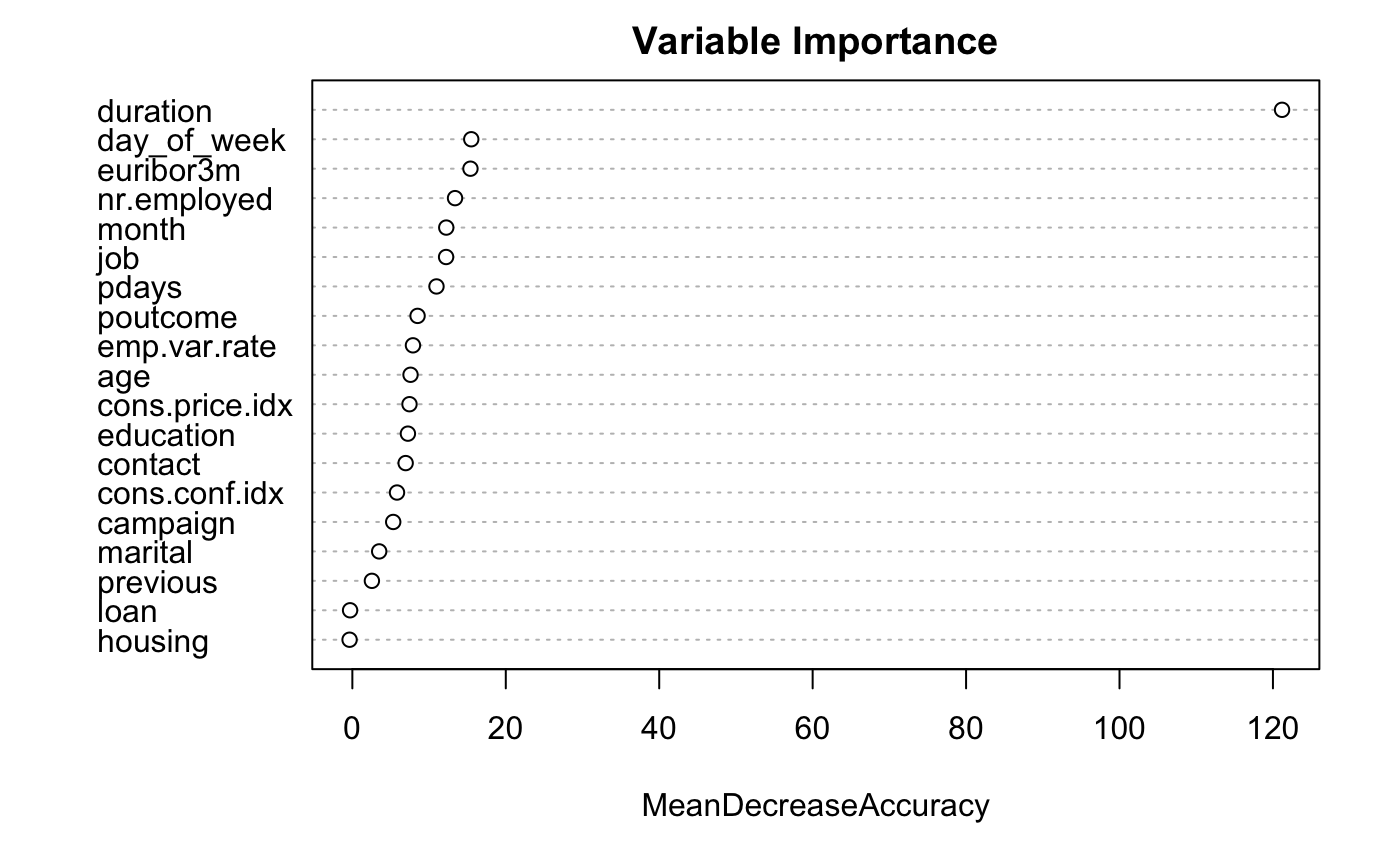






## Random Forest Model





**​​**