USING LOGISTIC REGRESSION TO PREDICT DEFAULT RISK OF CREDIT CARD CUSTOMERS

ST404 – ASSIGNMENT 3 – 1809568

CONTENTS

1. Introduction

Our goal with this analysis was to build a logistic regression model, that can predict the risk of defaulting on credit card bills based on customer data. Our client is a Taiwanese Credit Card company, which hopes to use our model as a component of a system to help them decide whether to give a card to a new customer applying for one or more credit for an existing customer. Our problem involves balancing predictive capabilities of our model with explanatory power, as it is important for our client to be able to justify any decisions which use our model as an input to customers.

The credit card company provided us with two sets of data. A training data set consisting of 5000 observations and a model validation data set, which consists of 2067 observations. The data consists of 24 variables which describe the sex, education, age, and marital status of our customers in addition to variables describing their payment behaviour between April and September of 2015 and whether they ended up defaulting on payments. In our modelling we applied different transformations to our original dataset and used two different variable selection methods to come up with a simple yet robust model to fulfil the needs of our client. We used the provided validation set to evaluate the predictive power of our final model.

2. Technical summary

2.1 Exploratory data analysis: Univariate analysis

2.1.1. Missing values

We explore the univariate distribution of our variables to find out whether there is a need to transform any of the variables. From an initial summary of the data, we notice that two variables, “EDUCATION” and “MARRIAGE” have rows with missing values. We have 57 entries with missing value in the education column and 3 entries with a missing value in the marriage column. We assume that NA in the education column means that the data is missing rather than to denote the lack of education, as this would be then classified under other. The number of rows with missing variables is miniscule in comparison to the total data set, so we decide to remove these entries as even if there was a systematic pattern to the missingness, it would not contribute enough to our model to warrant a deeper analysis.

2.1.2. Distribution of categorical variables

Before going deeper into exploratory data analysis, we note that we performed the same analysis on the validation set and found the distribution to be similar.

We have nine different variables, which are categorical. Level of education, sex of customers, marital status and the repayment status in months between April and September. For the repayment status, we include only one plot, as the distribution is very similar between each month. We note that the data set consists of highly educated customers, which tend to be female, and that we have more married customers than non-married customers in the data set. It also seems that that the customers in the data set generally repay their credit card debts on time, but we have a noticeable minority with delays in repayments. The educational level distribution is skewed towards higher education in comparison with what would be expected of the general population. This is explained by the fact that the dataset consists of customers with a credit limit of 250,000NT$ or more. We expect the bank to set the credit limit based at least somewhat on the income level of the customers, which is correlated with educational level.

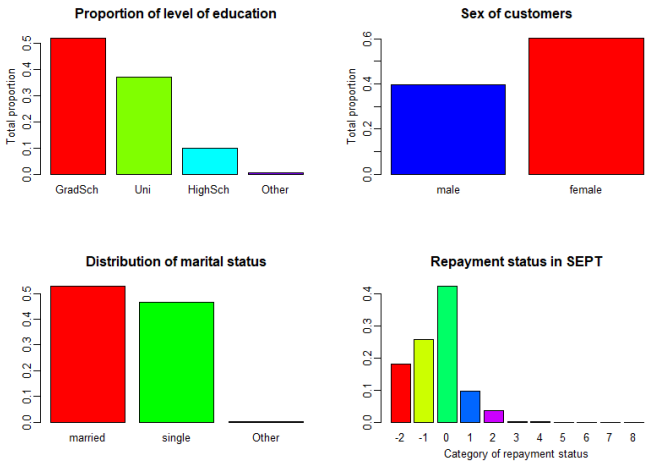

Figure 1: Distribution of categorical variables

2.1.3. Distribution of continuous variables

In Figure 2 we have plotted histograms of our continuous variables, which are the amount of credit limit, amount paid in each month, the amount of the bill statement in each month and age of our customers.

We notice that our monetary variables are heavily skewed. We do not believe these values to be outliers as we expect due to the nature of income distribution, that some people will be spending multiple orders of magnitude larger amounts in comparison with others. This also extends to the credit limit, as people with higher incomes will tend to have higher limits on their credit card. We do notice that the credit limit does not go down continuously, but rather we have very few people with credit limits above 500,000$. This might be due to our company's policies, which might put strict requirements for customers wishing to be allocated a larger credit limit.

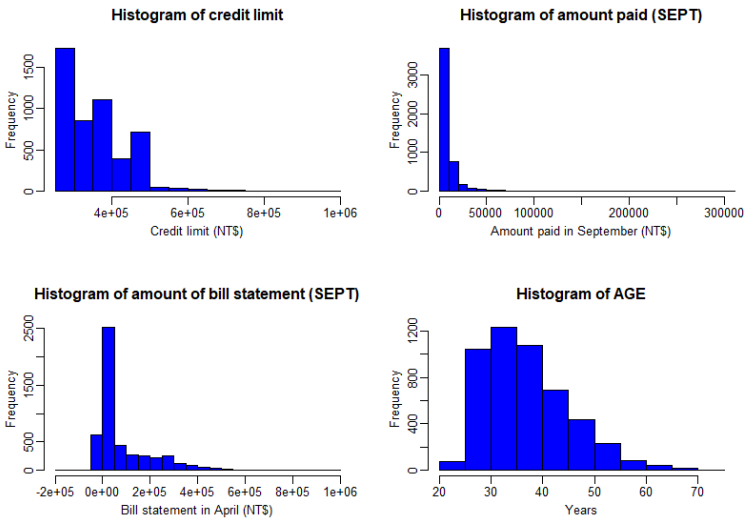

Figure 2: Distribution of continuous variables

There is no reason to expect that the customers on the very right end of our histograms are outliers, due to the aforementioned reasons, but by looking at boxplots of some of the payment variables we note that there are a few customers, who are outliers even with respect to other high spenders.

2.2. Exploratory Data Analysis: Multivariate analysis

2.2.1. Introducing new variables and aggregates

Before going deeper into modelling decisions, we notice that we have multiple variables which capture similar information at different points of time. In figure 3, we notice that these variables are especially multicollinear for the bill-amount each month, with successive months having the highest correlation with each other. With regards to amount paid each month, there is more variability between the months, which is shown in the scatter plots below by the L shape. Some customers choose to pay off their debts in large chunks in one month, while paying a lesser amount in the following month.

Although one will expect that there is some variability in the amounts of bill statements and amounts paid between different months due to one-off purchases or seasonality in consumption, it will help simplify our analysis, and allow us to produce more insightful plots, if we aggregate the data somehow.

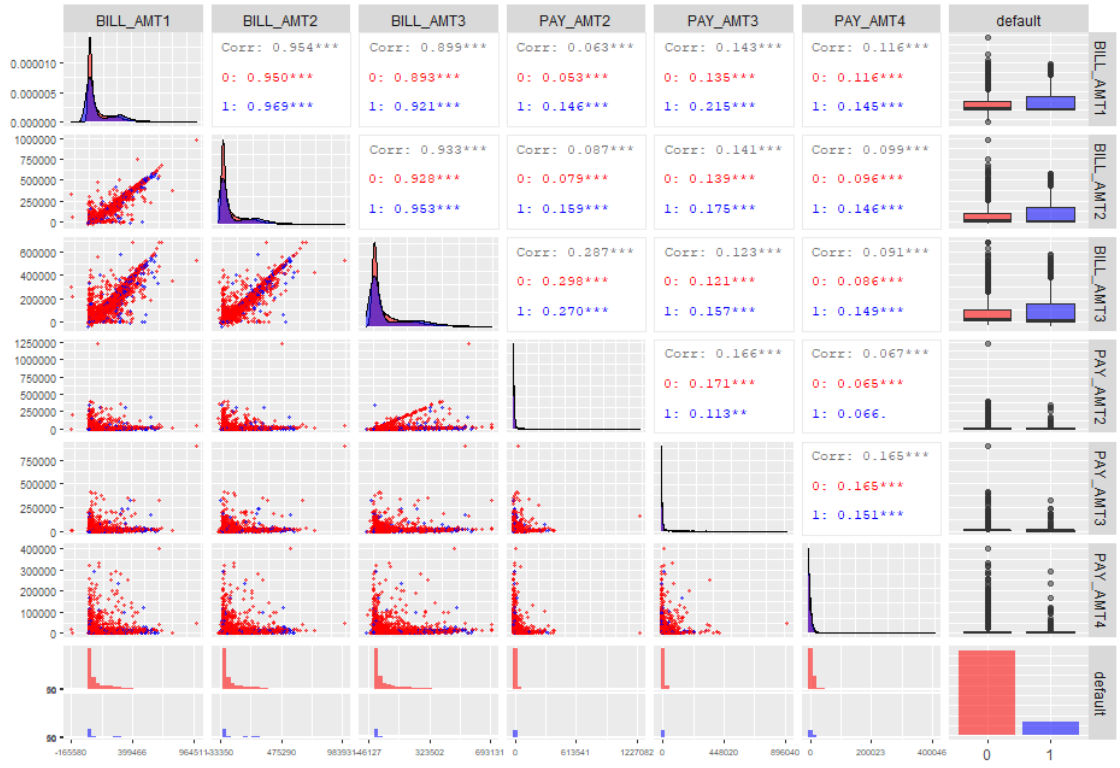

Figure 3: Pair plot of a subset of bill amount and amount paid variables

We want to ensure that these aggregates to capture the underlying mechanism that will affect our dependent variable. We create variables paidMean and dueMean for the means of the amount of bill statement and amount paid each month. These allow us to capture the baseline behaviour of each of our credit card owners from month to another. To compensate for the fact that the credit limit can influence the size of the bills

that our customer can accumulate and therefore also on the payments made each month, we also introduce credit limit normalized versions of the above two variables, paidMeanNormalized and dueMeanNormed. Since these variables will be highly collinear, we will use variable selection procedures in 2.1.3 to decide which variables to include in the model.

We also introduce aggregators for our factor which captures repayment status in each month. We believe that the most common behaviour of our customers between the months is an adequate way to capture the relevant information with regards to the dependent variable. To achieve this, we introduce the paymentMode variable, which indicates the most common level, or mode, of the factor within the 6 month timespan. This factor has four levels, -2,-1,0 and overdue. The factor is set to be overdue for a customer if the most common repayment status was a payment delay of one month or more. We also introduce a indicator, paymentDelay, which captures if there has been a single month where the repayment has been delayed. While paymendMode already captures some of these situations, paymentDelay also flags customers which have only one or two delayed repayments, but otherwise are using their credit card in a normal manner.

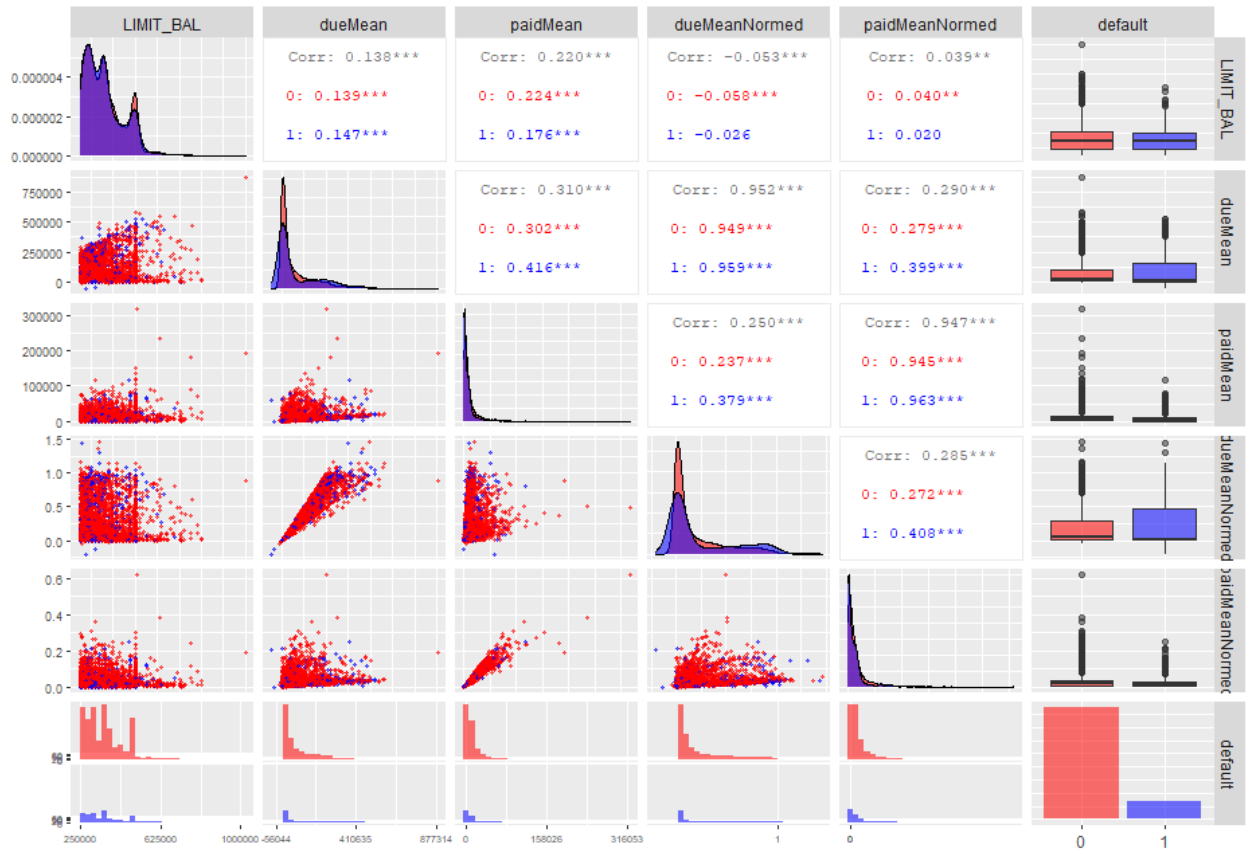

Figure 4: Continuous variables pairwise plot, including default variable

Figure 4 shows that our variable aggregation is looking promising, especially considering the boxplots on the right side, we see that for namely the dueMean variables there is a clear difference in distribution between

customers who defaulted and who did not. This shows that our aggregate variables have some predictive power and that it is sensible to try to build a model with them.

An added benefit of aggregating the time-sensitive variables is that the final model becomes more robust with regards to the data set used. Since it is important for our client to use the model on data can span more than 6 months in a specific year (here 2015), by aggregating the values over the different months into single variables, the client can adjust the timespan on which to base the classification decisions of the model, say by aggregating over the whole of the customers lifetime with the client, which could be longer or shorter than the 6 months available in our data set.

2.2.2. Empirical logit plots

We produce empirical logit plots in order to assess the relationship between our continuous variables and our dependent variable. In a logistic regression setting, we would expect the empirical logit plots to show a linear relationship between the log-odds and our continuous variable. From figure x, we see the initial empirical logit plots of our variables. We can conclude that the plot of age looks quite linear and that the credit limit variable also seems to have some linear, although weak, relationship in the plot. Additionally, our credit limit normalized mean bill statement seems to have a linear relationship with higher % bins having a large leverage. Same can be said to an even larger extent with the credit limit normalized mean amount of bills paid per month.

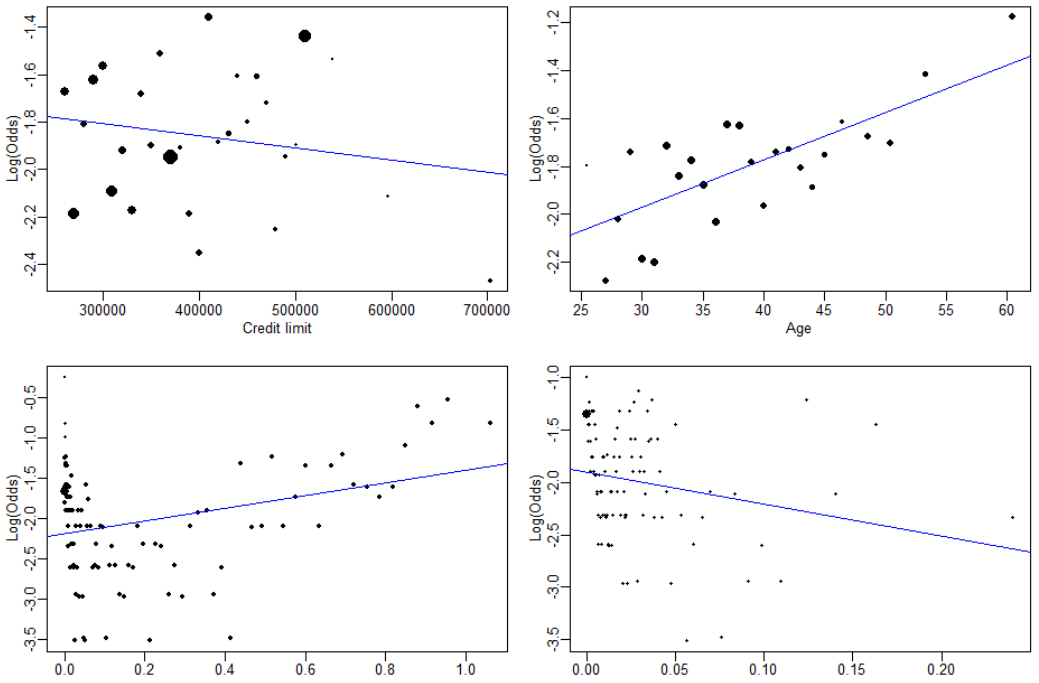

Figure 5: empirical logit plots of continuous variables in the model

Looking at the mean bill statement plot, it seems that the relationship is linear for at least statement amounts larger than 100,000 NT$. For mean amount paid per month we notice the leverage that few high mean points have over the supposedly linear relationship and we therefore apply a log(x+1) transformation to the

paidMean variable to lessen the leverage of these points. We can see that after applying the transformation, the relationship becomes linear with regard to the log-odds.

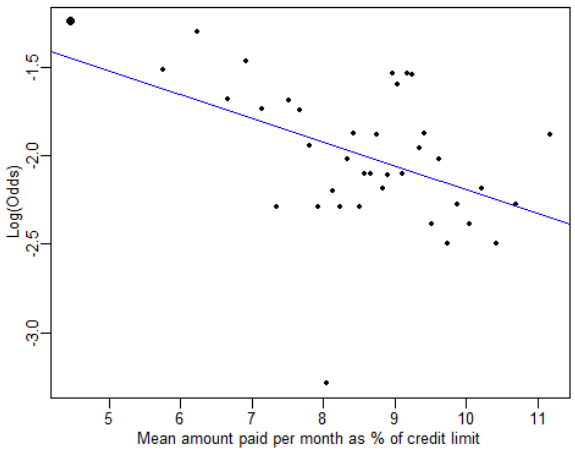

Figure 6: Empirical logit plot of transformed log(paidMean+1)

2.3 Modelling

After confirming that our continuous variables have a linear relationship with regards to the logit of the predictor, we now use two different methods of variable selection to come up with a logistic regression model that can accurately predict whether our customer will default on their credit card debt. We will consider using two different variable selection methods. Firstly, we will manually backwards select the model, starting with a full model, which includes all explanatory variables, we remove insignificant variables one-by-one and arrive at a final model with all terms being statistically significant.

The second variable selection method will be grouped LASSO, which allows us to use LASSO variable selection on a model which includes factor terms. Normal LASSO would not be effective with a model that includes factorized variables, because after converting the factorized variables to dummy variables it might force some coefficients of different values of the same factor to zero, while keeping others non-zero. We either want all the coefficients to be non-zero, which corresponds to including the factor in the model, or all of them zero, to remove the factor from the model.

2.3.1 Backwards Variable Selection

Using backwards variable selection we end up with a model that consists of 5 variables, sex, paymentDelay, paymentMode, log of mean paid amount and credit limit normed mean of bill statement per month.

At the end of our variable selection, we decided between two models, one including sex of the customer and the other including the educational level in addition to the other variable. We decided against the model including education, since the parameter estimate for the dummy variable of education being “Other”, was an order of magnitude more negative than the other estimates. This is explained by the fact that in the training data, all observations with education equal to “other”, we do not have any defaults, thus in our model any observation with education being other would be classified as non-defaulting. This does not make much sense from a logical standpoint. We also note that including sex in the model might be controversial and possibly legally untenable if decisions are to be made using the model. Our final model has the smallest AIC of all the models considered in our selection process.

2.3.2 Grouped LASSO

In figure X we can see the trace plot of our grouped LASSO variable selection. We use 10-fold cross-validation to come up with estimates on the lambda values that minimize our penalized regression objective function. We notice that the minimum value found by cross-validation ends up with a model that has 14 explanatory variables, which is only a reduction of 3 from our original variables. Literature (e.g. Hastie, et al., 2009) presents us with an established method of addressing this is, which is to use a lambda value one standard error from the minimum value. With this we arrive at a model with 5 predictor variables.

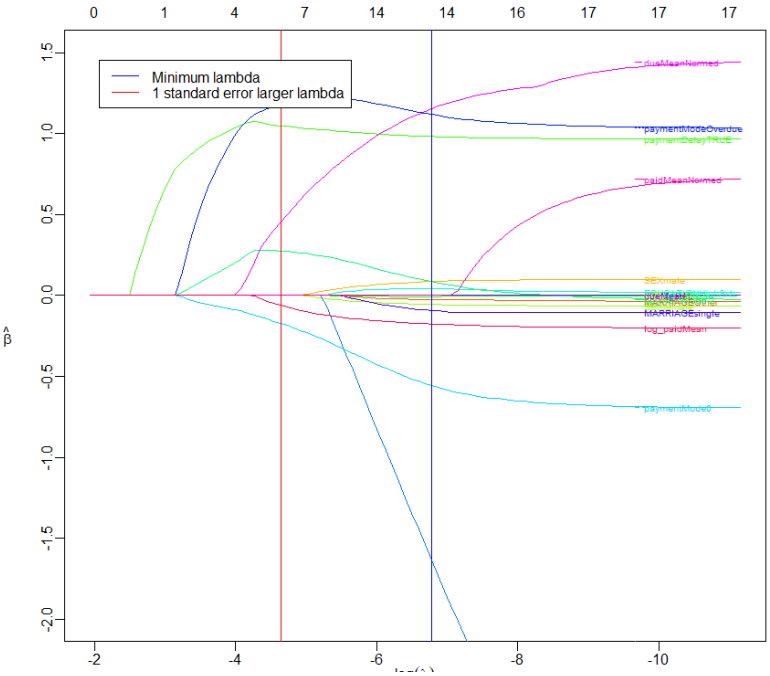

Figure 7: Trace plot of grouped lasso

2.3.3 Final Models

After applying the variable selection methods, we arrive at the following estimates for our logistic regression models.

Table 1: Final model summary

	Intercept	SEXFemale	paymentDelay "TRUE"	Payment Mode "-2"	payment Mode "0"	payment Mode "Overdue"	log(paidMean)	dueMean Normed
Backwards	-0.641	-0.184	0.979	-0.00177	-0.711	1.02	-0.193	1.88
LASSO	-1.78	/	1.05	0.275	-0.172	1.18	-0.0618	0.454

2.4 Model Evaluation

In this part we focus our model diagnostics mainly on the model found with backwards variable selection due to the larger availability of evaluation tools relative to penalized regression models. Our goal is to use diagnostic plots show that our model fits the assumptions of a generalized linear model.

2.4.1 Diagnostic plots

In figure 7, we can see a half normal plot in which we have plotted absolute Pearson residuals against theoretical normal quantiles. Although a binomial model does not require the residuals to be distributed normally, we can still gain confidence in the validity of our model, because the residuals of model lie within the so-called simulated envelope of the half-normal plot.

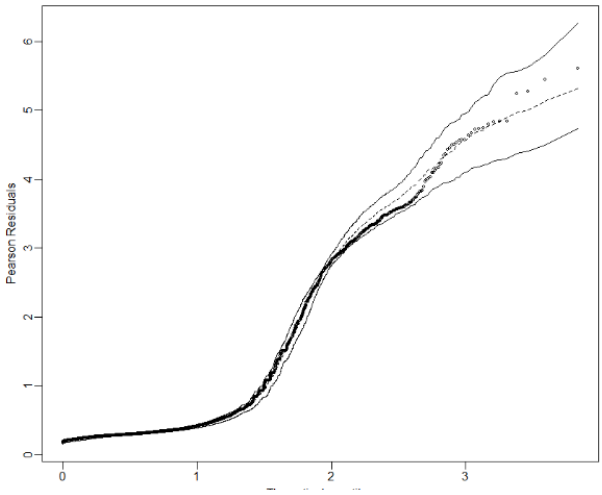

Figure 8: Half normal plot

We have the binned residual plot of standardised Pearson residuals against the expected values of our final model. We see that the Pearson residuals seem to be randomly distributed without a clear trend. After plotting the binned residual plot for studentized Pearson and deviance residuals, there seemed to be a negative bias in the residuals, which we cannot explain considering that the standardized Pearson residuals did not show any trend or bias.

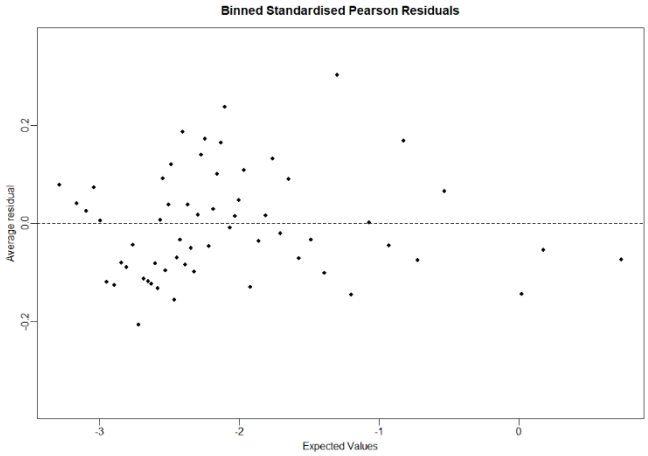

Figure 9: Binned Standardised Pearson residuals

2.4.2 Multicollinearity

We used variance inflation factor to check that our model has no issues of multicollinearity. None of the variables in our model has a high variance inflation factor, so we do not consider multicollinearity an issue of our model.

2.4.3 Outliers

We also looked at the effect of some outliers with especially large values in the log(paidMean) variable, but diagnostics such as the influence index plot and residual vs leverage plot showed that removing these would not affect the model parameters much.

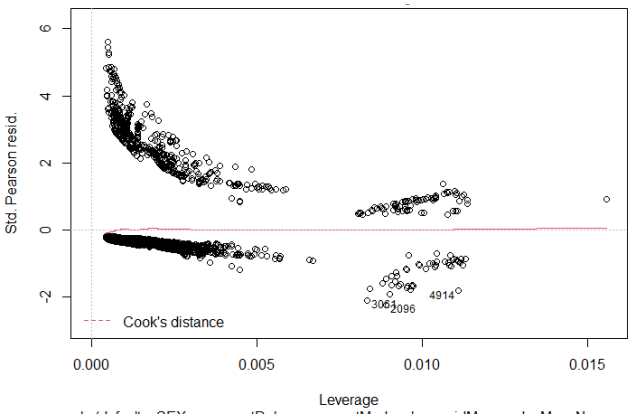

Figure 10: Residuals vs leverage plot

2.5 Model Performance

In the previous part we have shown that our original model sufficiently meets the key assumptions of a generalized linear model, as showcased by the diagnostic plots generated.

In figure 9 we see the output of the probabilities of our model on the training set. Depending on how our model will be used, the importance of balancing between minimizing false positives vs. false negatives will vary. If our model is the only step between giving someone more credit or denying them of more credit, we might want to avoid losses due to customers defaulting on debt and therefore try to minimize the false negative rate as much as possible. If our model is only one part of the system, then we might not care that our false negative rate is higher, if other parts of the credit allocation process will flag possible defaulting customers. This balancing act will also depend on the expected profit and loss for each non-defaulting and defaulting customer. Figure 12 shows the trade-offs between sensitivity (1 – FNR) and specificity (1-FPR).

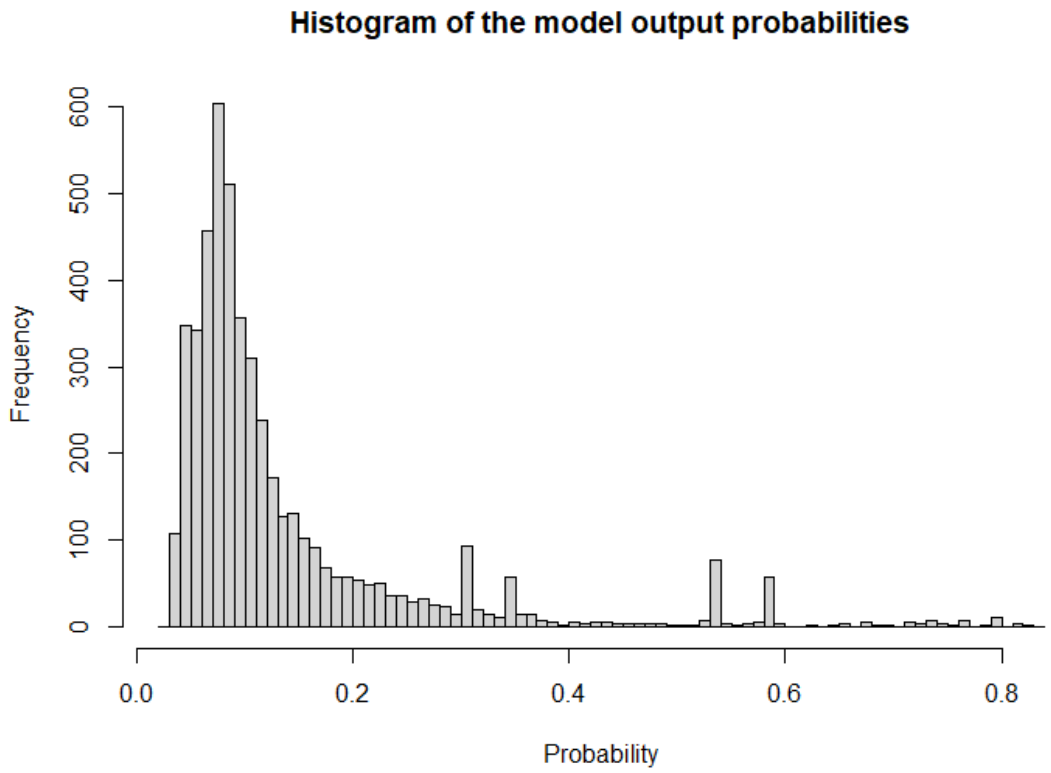

Figure 11: Distribution of output probabilities of our backwards selection model

	Training AUC	Validation AUC	FNR (10% threshold)	FPR	Accuracy (10% threshold)
Backwards Selection	0.749	0.754	0.226	0.391	0.636
LASSO	0.753	0.735			

Table 2: Comparison table of our two models

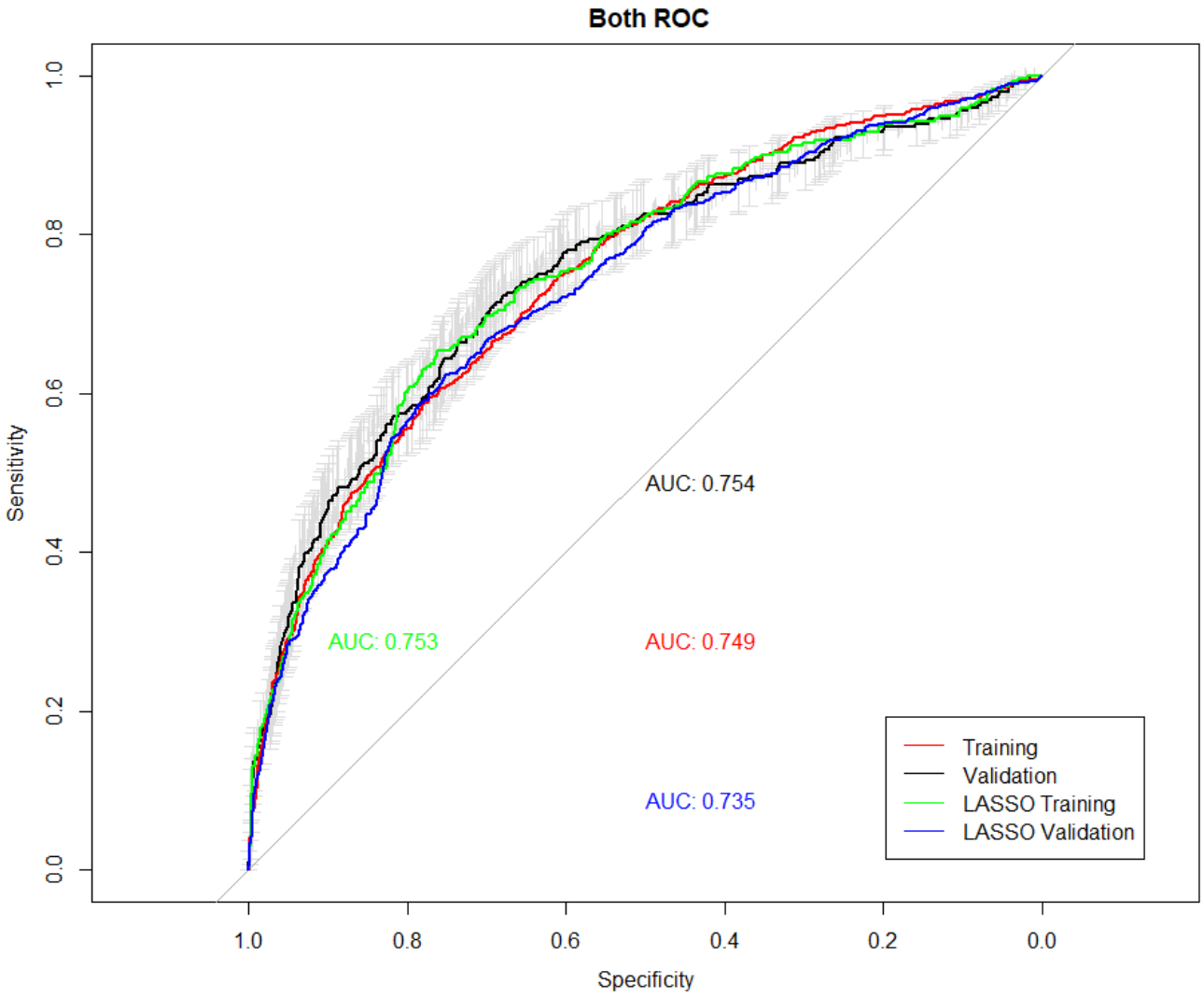

Figure 12: ROC curve with AUC calculation

The AUC values are large enough to say that the model does relatively well in detecting customers who will default on their credit card debt. Additionally, the AUC values calculated from the training and validation set are similar, which is a sign that the model does not have a poor out-of-sample accuracy, which is naturally something we wish from a model, which will be used on new data to inform client decisions. We notice that the grouped LASSO model underperforms the backwards selection model with regards to the validation set. Because we want the model to have as small as possible out-of-sample error, we will choose the final model to be the backwards variable selection one.

3. Results

3.1 Summary of findings and technical methods

By using a statistical technique called logistic regression, we built a model which is able to take in data of credit card customers and output a probability of that customer defaulting on their credit card debt. We first trained, or fine-tuned the model on a set of 5000 observations of customers. After this we were able to

estimate the accuracy of our model on completely unseen data. We found that our model is sufficiently accurate to correctly distinguish in the majority of the cases between customers which are likely to default and those which are not likely to default.

Our final model predicts the probability of a customer defaulting by considering 5 different variables: the sex of the customer, whether the customer has had a payment delay, most common status of repayment, the mean bill statement amount normalized by the credit limit given to the customer and the mean amount paid in bills over the time in the data set (April to September). One advantage of our model is its use of aggregated variables, such as means and modes of the variables in the training data set, which allows for a more robust deployment to data sets with different time scales.

Its accuracy in prediction heavily depends on how weary the user of the model is of making false predictions on whether a customer defaults or not. In some situations, it might be beneficial to try to minimize the amount of false negatives, where the model predicts a customer will not default, where they actually will. This way, the user will avoid credit losses, but this comes with the trade-off that the model will be more prone to wrongly estimating that a customer will default where they will not.

3.1 Model analysis

We now move to interpreting the parameters of our logistic regression model. In figure 13 we see the explanatory variables, which determine the probability of a customer defaulting on their debt in our model. We note that having a singular delay in repayment or having the most common repayment status being overdue, will increase the odds of defaulting by approximately the same factor. An increase in the mean amount of bill statement normalized with the credit limit of the customer will tend to increase the odds of defaulting on debt. Being female and using revolving credit decrease the odds of defaulting, as does an

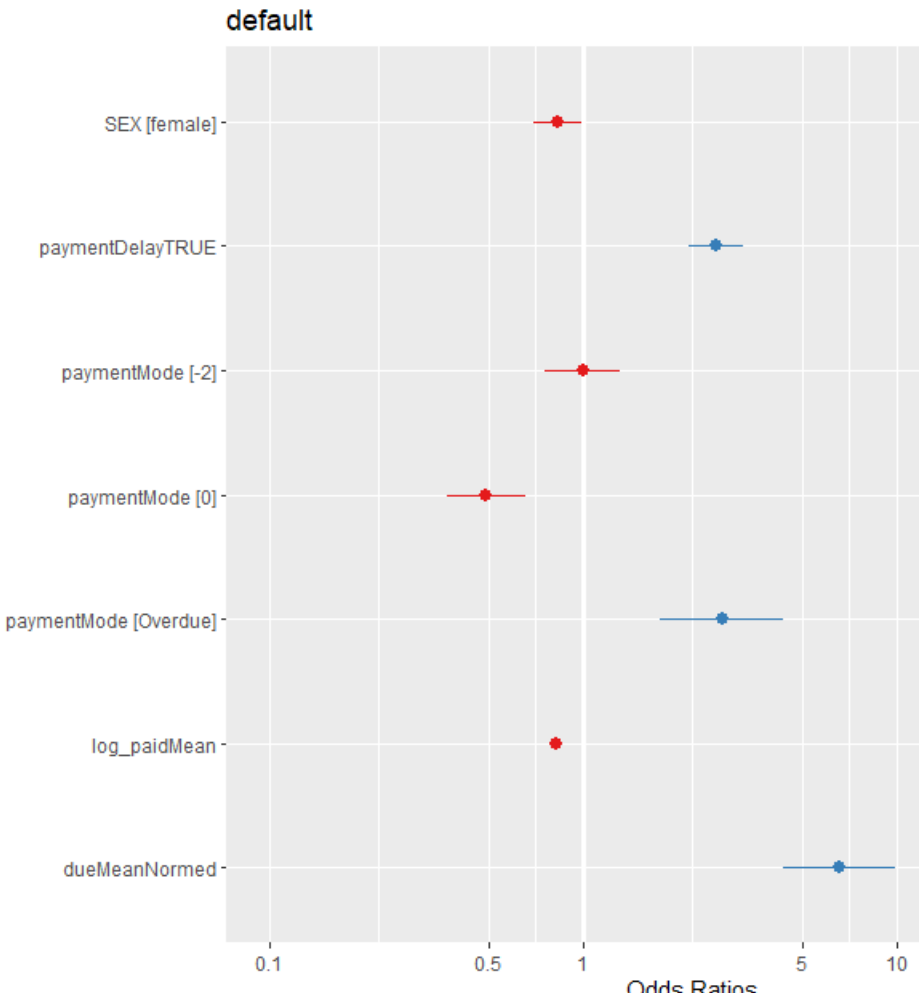

Figure 13: Plot of the odds ratios of our logistic regression model

increase in the mean amount the customer pays bills each month. A standard deviation increase in the dueMeanNormed variable will increase the odds of defaulting to not defaulting by a factor of 1.68 on average, when keeping all other variables fixed. Similarly, an standard deviation increase in log_paidMean corresponds to an 0.63 factor decrease in the odds of defaulting to not defaulting. An unit increase in the log_paidMean variable also corresponds to a similar decrease of the odds of defaulting to not as being a female does, according to our model. Figure 13 also includes the 95% confidence intervals for the odds ratios of these parameters, and give an idea on the ranges of values that the parameters can take.

3.2. Model limitations

The original data set is quite limited and certainly does not contain all the possible data that a credit card company has available on their customers or is able to collect from them as a prerequisite for doing business.

Although our AUC accuracy is relatively high, a naïve model which would classify all the observations as not defaulting would have an accuracy of 85.8% in the validation set, which would outperform our model, depending on what cut-off we choose for the classification probability. It is likely that we do not have enough variables to get much more accuracy with our current method of logistic regression. We might need to explore using other machine learning methods, such as neural networks or random forests to maximize our accuracy, although these are often prone to overfitting.

We feel that the model is not yet robust enough to solely make decisions on customers creditworthiness. It can be used as a guide to shift through customers quickly but still requiring that any final decision will be made based on more data than what was available for the creation of our model. Some other variables which might improve the capability of our model might be the income of our customer and total existing debt. Using these we could calculate debt to credit and debt to income ratios, both of which are used by banks to calculate existing creditworthiness measures. Knowing the interest level of the debts owed might also be useful to set the cut-off for the model, since then we could estimate our profitability and expected losses from defaults.

We also note that some of the diagnostic plots showed a bias in the residual plots. The inclusion of a logarithmically transformed variable also reduces the understandability of the model and thus the explanatory power.

4. Bibliography

- Brunsdon, Teresa & Plummer Martyn, ST404 Applied Statistical Modelling course material.
- Cannon, Ann, George Cobb, Bradley Hartlaub, Julie Legler, Robin Lock, Thomas Moore, Alan Rossman, and Jeffrey Witmer. 2019. “Package 'Stat2Data'.
- Dai , B. [. et al., 2018. oem: Orthogonalizing EM: Penalized Regression for Big Tall Data. [Online]
- de Andrade Moral, Rafael [aut, cre], John [aut] Hinde, and Clarice [aut] Garcia Borges Demetrio. 2018. Half-Normal Plots with Simulation Envelopes
- Fox, John [aut, cre], Sanford [aut] Weisberg, Brad [aut] Price, Michael [aut], Friendly, Jangman [aut] Hong , Robert [ctb] Andersen, David [ctb] Firth, Steve [ctb] Taylor, and R Core Team [ctb]. 2020. Package ‘effects’.
- Friedman, J. et al., 2019. glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models.
- Gelman , Andrew [aut], Yu-Sung [aut, cre] Su , Masanao [ctb] Yajima , Jennifer [ctb] Hill , Maria [ctb] Grazia Pittau , Jouni [ctb] Kerman, Tian [ctb] Zheng , and Vincent [ctb] Dorie. 2018. “Package ‘arm’.
- Robin, Xavier [cre, aut], Natacha [aut] Turck, Alexandre [aut] Hainard, Natalia [aut], Tiberti, Frédérique [aut] Lisacek, Jean-Charles [aut] Sanchez, Markus [aut] Müller, Stefan [ctb] Siegert , and Matthias [ctb] Doering. 2021. Package ‘pROC’
- Schloerke, Barret [aut, cre], Di [aut, ths] Cook, Joseph [aut] Larmarange, Francois [aut] Briatte, Moritz [aut] Marbach, Edwin [aut] Thoen, Amos [aut] Elberg, et al. 2021. GGally: Extension to 'ggplot2'.
- Sing, Tobias [aut], Oliver [aut] Sander, Niko [aut] Beerenwinkel, Thomas [aut] Lengauer , Thomas [ctb] Unterthiner , and Felix [cre] G. M. Ernst. 2020. Package ‘ROCR’.
- Wickham, Hadley. 2016. ggplot2: Elegant Graphics for Data Analysis. New York: Springer-Verlag.
- Wickham, Hadley. 2011. “The Split-Apply-Combine Strategy for Data Analysis.” Journal of Statistical Software 40 (1): 1-29

5. Appendix

```
#LOAD DATASETS
testSet <- readRDS("CardT.rds")
validationSet <- readRDS("CardV.rds")

attach(testSet)

## REMOVING MISSING VALUES


testSet <- testSet[!is.na(testSet$EDUCATION)&!is.na(testSet$MARRIAGE),]
validationSet <- validationSet[!is.na(validationSet$EDUCATION)&!is.na(validationSet$MARRIAGE),]

# Study the univariate distribution of continuous variables
comparison <- par(mfrow = c(2,2))
par(comparison)
hist(testSet$PAY_AMT1,col="blue",main="Histogram of amount paid in SEPT",xlab="Amount paid in Sep
tember",breaks = 40,xlim=c(0,300000))
hist(testSet$BILL_AMT1,col="blue",main="Histogram of bill statement in SEPT",xlab="Bill statement
 in September",breaks = 40)


## Initial plot of all continuous variables

hist(testSet$LIMIT_BAL,col = "blue", main ="Histogram of credit limit", xlab ="Credit limit (NT$)
",breaks = 30)
hist(testSet$PAY_AMT1,col="blue",main="Histogram of amount paid (SEPT)",xlab ="Amount paid in Sep
tember (NT$)",breaks = 40,xlim=c(0,300000))

## Plot of categorical variables

barplot(prop.table(summary(EDUCATION)),col = rainbow(4),ylab="Total proportion",main = "Proportio
n of level of education")
barplot(prop.table(summary(SEX)),col=c("blue","red"), main = "Sex of customers",ylab = "Total pro
portion")


## Bivariate plots

barplot(prop.table(summary(default)),col = c("green","red"),names.arg = c("No","Yes"),main = "Pro
portion of clients who default")

# Since we know the different months are highly co-linear, we will replace them with mean variabl
es, the mean amount paid per month and maximum paid per month

# Pipeline for transformed credit card information

cards <- testSet

# Remove lagged variables
cards <- cards[,-(12:23)]

#Function to get most common repayment status from data
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

#Helper to check if any of the repayment statuses are overdue
gt <- function(x){
  any(x>0)
}

# Convert factors into integers so that we can take the mode
temp <- cbind(as.numeric(levels(testSet$PAY_1))[testSet$PAY_1],as.numeric(levels(testSet$PAY_2))[
```

```
testSet$PAY_2])
temp <- cbind(temp,as.numeric(levels(testSet$PAY_3))[testSet$PAY_3])
temp <- cbind(temp,as.numeric(levels(testSet$PAY_4))[testSet$PAY_4])
temp <- cbind(temp,as.numeric(levels(testSet$PAY_5))[testSet$PAY_5])
temp <- cbind(temp,as.numeric(levels(testSet$PAY_6))[testSet$PAY_6])
paymentDelay <- apply(temp,1,gt) # Find out whether a row has overdue repayment status

#Calculate other aggregates
dueMean <- rowMeans(testSet[,12:17])
paidMean <- rowMeans(testSet[,18:23])
dueMeanNormed <- dueMean/LIMIT_BAL
paidMeanNormed <- paidMean/LIMIT_BAL

paymentMode <- apply(testSet[,6:11],1,getmode) # Evaluates mode of repayment status
library(forcats)
paymentMode<-fct_collapse(paymentMode, "-1" = c("-1"), "-2" = c("-2"), "0" = c("0"), "Overdue" =
c("1","2","3","4","5","6","7","8","9")) #Simplify the factorisation of mode

#Create new matrix
cards <- cbind(cards,paymentDelay)
cards <- cbind(cards,paymentMode)
cards <- cbind(cards,dueMean)
cards <- cbind(cards,paidMean)
cards <- cbind(cards,dueMeanNormed)
cards <- cbind(cards,paidMeanNormed)

#Remove old values
cards <- cards[,-c(6:11)]
attach(cards)
#Reorder to allow for ggpairs plotting
cards <- cards[,c(1:5,7:12,6)]

# Following functions from "Logistic Regression Pima Indians Example Using R" (Brunsdon 2021)

# Pairs plots of continuous variables

ggpairs(testSet[,c(12:14,19:21,24)],mapping=aes(color=default,alpha=0.4),
        lower = list(continuous= mod_points,combo=mod_bihist),
        diag=list(continuous=modified_density,discrete=mod_bar),
        upper=list(continuous=mod_cor,combo=mod_box))

ggpairs(testSet[,c(18:23,24)],mapping=aes(color=default,alpha=0.4),
        lower = list(continuous= mod_points,combo=mod_bihist),
        diag=list(continuous=modified_density,discrete=mod_bar),
        upper=list(continuous=mod_cor,combo=mod_box))

ggpairs(cards[,c(1,8:12)],mapping=aes(color=default,alpha=0.4),
        lower = list(continuous= mod_points,combo=mod_bihist),
        diag=list(continuous=modified_density,discrete=mod_bar),
        upper=list(continuous=mod_cor,combo=mod_box))

# Empirical logit plots

# Empirical logit function from "Logistic Regression Pima Indians Example Using R" (Brunsdon 2021
)
myemplogit <- function(yvar=y,xvar=x,maxbins=10,sc=1,line=TRUE,...){
  breaks  <<- unique(quantile(xvar, probs=0:maxbins/maxbins))
  levs  <<- (cut(xvar, breaks, include.lowest=FALSE))
  num <<- as.numeric(levs)
  c.tab <- count(num,'levs')
  c.tab$levs <- factor(c.tab$levs, levels = levels(addNA(c.tab$levs)), labels = c(levels(c.tab$le
vs),
                                                                                  paste("[",min(x
var),"]",sep="")), exclude = NULL)
  c.tab <- c.tab[c(nrow(c.tab),1:nrow(c.tab)-1),]
  sc <- (max(c.tab$freq)/min(c.tab$freq)/sc)^2
```

```
  zcex <<- sqrt(c.tab$freq/pi)/sc
  print(c.tab);print(zcex);print(sc)
  emplogitplot1(yvar~xvar,breaks=breaks,cex=zcex,showline=line,...)
}
#Plot empirical logit plots
par(mfrow=c(2,2),mgp=c(1.5,0.5,0),mar=c(3,3,1.2,0.5))
myemplogit(default,(LIMIT_BAL),100,sc=20,xlab="Credit limit",line=TRUE)
myemplogit(default,AGE,30,sc=50,xlab="Age",line=TRUE)
myemplogit(default,dueMeanNormed,100,sc=100,xlab="Mean bill statement per month as % of credit li
mit",line=TRUE)
myemplogit(default,paidMeanNormed,100,sc=5,xlab="Mean amount paid per month as % of credit limit"
,line=TRUE)
myemplogit(default,log((paidMean+1)),40,sc=1,xlab="Mean amount paid per month as % of credit limi
t",line=TRUE)
#Add transformed variable
cards <- transform(cards, log_paidMean = log(paidMean+1))
cards$paidMean <- NULL

# Backwards fitting the model

library(glmnet)

fit1 <- glm(default~.,family=binomial,data = cards)
summary(fit1)
Anova(fit1)

## Backwards elimination of the least significant term

fit2 <- glm(default~. -LIMIT_BAL, family = binomial, data = cards)
Anova(fit2)

# Multiple fit -> check ANOVA later:

fit7 <- glm(default ~ SEX+paymentDelay+paymentMode+log_paidMean+dueMeanNormed, family = binomial,
data = cards)
#DO we remove sex?
fit8 <- glm(default ~ EDUCATION+paymentDelay+paymentMode+log_paidMean+dueMeanNormed,family = bino
mial, data = cards)
anova(fit8,fit7,test="Chisq") #Choose model with SEX, doesnt have a massive coefficient for "EDUC
ATION" other, which might be caused by an anomaly in the training data
Anova(fit7)
summary(fit7)

# Group LASSO

x_train <- model.matrix( ~ .-1, cards[,-12])
group1 <- c(1,2,2,3,3,3,4,4,5,6,7,7,7,8,9,10,11)

lso1 <- oem(x = x_train, y = as.numeric(as.character(cards$default)), family="binomial",
                   penalty = c("lasso", "grp.lasso"),
                   groups = group1)

groupLSO <- cv.oem(x = x_train, y = as.numeric(as.character(cards$default)), family="binomial",
                penalty = c("lasso", "grp.lasso"),
                groups = group1,nfolds = 10)

predict(lso1, s = groupLSO$lambda.min.models[2], which=2,type="coefficients")
predict(lso1, s = groupLSO$lambda.1se.models[2], which=2,type="coefficients")

plot(lso1, which.model = 2, xvar = "loglambda", main="Group lasso", ylim = c(-2,1.5))
abline(v=log(groupLSO$lambda.1se.models[2]),col="red")
abline(v=log(groupLSO$lambda.min.models[2]),col="blue")
legend("topleft",legend=c("Minimum lambda", "1 standard error larger lambda"),lty=c(1,1),col=c("b
lue","red"), ins=0.05)

# Diagnostic plots
```

```
##

finalModel <- fit7

confint(finalModel)

# VIF
vif(finalModel)

# Influence
influenceIndexPlot(finalModel)

residualPlots(finalModel)
crPlots(finalModel)
marginalModelPlots(finalModel)

plot(allEffects(finalModel))

arm::binnedplot(x=predict(finalModel,type="response"),y=rstudent(finalModel,type="pearson"),nclas
s=40,col.int=NA, main ="Binned Student Pearson Residuals") # student residual and 40 bins
arm::binnedplot(predict(finalModel),rstandard(finalModel,type="pearson"),nclass=60,col.int=NA, ma
in ="Binned Standardised Pearson Residuals") # standardised pearson residual and 60 bins
arm::binnedplot(predict(finalModel),rstandard(finalModel,type="deviance"),nclass=40,col.int=NA, m
ain ="Binned Standardised Deviance Residuals") # standardised deviance residual and 40 bins


## HNP

par(mfrow=c(1,1),mgp=c(1.7,0.5,0),mar=c(3.5,3.5,3,0.5))

hnp(finalModel,resid.type="deviance",ylab="Deviance Residuals")
hnp(finalModel,resid.type="pearson", ylab="Pearson Residuals")

## dfbetas
dfbetasPlots(finalModel,intercept = TRUE,id.n=3)


par(mfrow=c(1,1),mgp=c(1.7,0.5,0),mar=c(3.5,3.5,3,0.5))
hist(predict(finalModel))
hist(rstudent(finalModel))

#Validation set pipeline

cardsV <- validationSet
cardsV <- cardsV[,-(12:23)]

# Same preparation for mode taking as before
temp <- cbind(as.numeric(levels(validationSet$PAY_1))[validationSet$PAY_1],as.numeric(levels(vali
dationSet$PAY_2))[validationSet$PAY_2])
temp <- cbind(temp,as.numeric(levels(validationSet$PAY_3))[validationSet$PAY_3])
temp <- cbind(temp,as.numeric(levels(validationSet$PAY_4))[validationSet$PAY_4])
temp <- cbind(temp,as.numeric(levels(validationSet$PAY_5))[validationSet$PAY_5])
temp <- cbind(temp,as.numeric(levels(validationSet$PAY_6))[validationSet$PAY_6])

# Continues in the same way as before, so we do not include the rest here

## Validation

predictedV <- predict(finalModel, type='response',newdata = cardsV)
boxplot(predictedV~cardsV$default, col="blue")

ypred <- predictedV > 0.10
addmargins(table(cardsV$default, ypred))
ylassopred <- predictedV > 0.10
addmargins(table(cardsV$default,ylassopred))
```

```
# Exactly the same margins here, therefore same FPR and FNR

## Accuracy
(1072+233)/2051

#FNR
68/(68+233)
#FPR
688/(1072+688)
x_trainV <- model.matrix( ~ .-1, cardsV[,-6])

predictedLassoValidation <- predict(lso1, s = groupLSO$lambda.1se.models[2], which=2,newx = x_tra
inV,type="response")
predictedLassoTrain <- predict(lso1, s = groupLSO$lambda.1se.models[2], which=2,type="response",n
ewx = x_train)

#PLOT MULTI ROC
plot.roc(cardsV$default,predict(finalModel,type="response",newdata=cardsV),ci=TRUE,of="thresholds
",ci.type="shape",print.auc=TRUE, main=" Both ROC")
plot.roc(cards$default,predict(finalModel, type='response'),add=TRUE,col="red",print.auc=TRUE, pr
int.auc.x= 0.5, print.auc.y=0.3,print.auc.col="red")
plot.roc(cardsV$default,predictedLassoValidation,add=TRUE,col="green",print.auc=TRUE, print.auc.x
= 0.9, print.auc.y=0.3,print.auc.col="green")
plot.roc(cards$default,predictedLassoTrain,add=TRUE,col="blue",print.auc=TRUE, print.auc.x= 0.5,
print.auc.y=0.1,print.auc.col="blue")
legend("bottomright",legend=c("Training","Validation","LASSO Training","LASSO Validation"),col=c(
"red","black","green","blue"),lty=c(1,1),inset=c(0.05,0.05)) # Add a legend
```