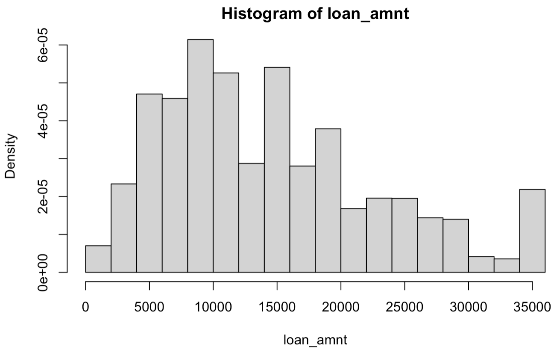
MATH60131: Consumer Credit Risk Modelling Project

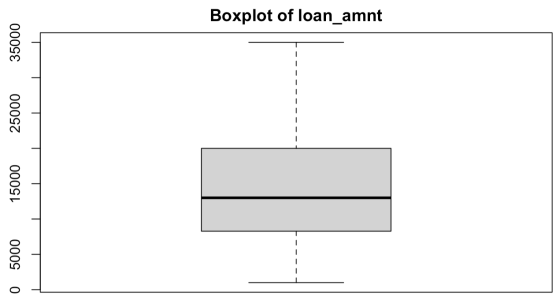
CID: 01938572

1 Q1: Load the data

2 Q2: Analyse the predictor variables: loan_amnt, grade, emp_length_p, term and addr_state

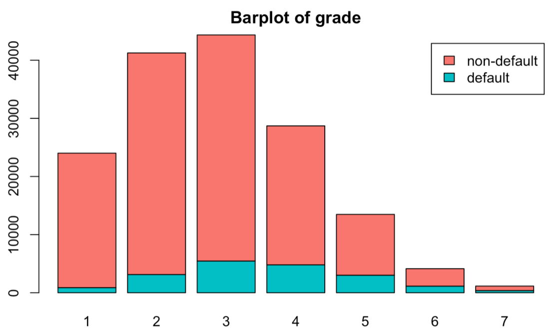
2.1 loan_amnt





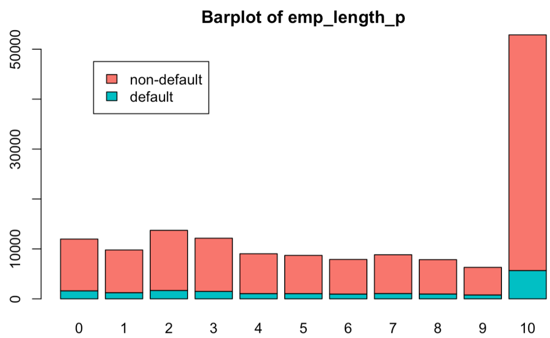
From the figures above, the distribution of loan_amnt only has slight positive skewness and there are no outliers present, thus we will not perform any data transformation on loan_amnt.

2.2 grade



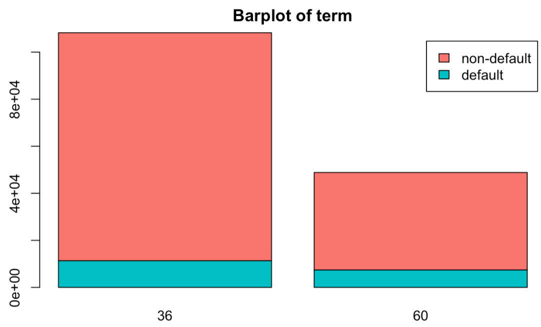
From the barplot above, we notice that the ratio of default to non-default is much higher for higher grades. Thus, we can assume that a customer with higher probability of default will be assigned a higher loan grade by LC.

2.3 emp_length_p



The predictor emp_length_p has 7956 missing values. We will impute the missing values with the mean. From the barplot above, we can notice that the ratio of non-default to default is higher for employment length of 10 or more years.

2.4 term



The predictor term has two values: 36 months or 60 months. We will replace 36 months with 0 and 60 months with 1. From the barplot above, we can observe that the ratio of non-defaut to default is much higher for loan term of 36 months.

2.5 addr_state

```
  AK    AL    AR    AZ    CA    CO    CT    DC
 426  2011  1184  3687 22280  3233  2320   421
  DE    FL    GA    HI    ID    IL    IN    KS
 447 10388  5081   802     1  6436  2820  1400
  KY    LA    MA    MD    ME    MI    MN    MO
1627  1864  3477  3690     1  4122  2894  2475
  MS    MT    NC    NH    NJ    NM    NV    NY
 808   478  4338   747  5928   908  2172 13332
  OH    OK    OR    PA    RI    SC    SD    TN
5343  1370  1927  5592   701  1987   331  2622
  TX    UT    VA    VT    WA    WI    WV    WY
12575 1064  4630   368  3416  2092   878   391
```

The predictor addr_state has 48 unique values, thus it is not feasible to enter them in the model as a series of indicator variables. We will substitute the predictor addr_state with the continuous weights of evidence for each value.

3 Q3: Split the data randomly into a training data set and a test data set

There is a total of 157085 observations in the data set provided. We split the data with respect to the ratio 2:1. There are 104724 observations in the training data set and 52361 observations in the test data set.

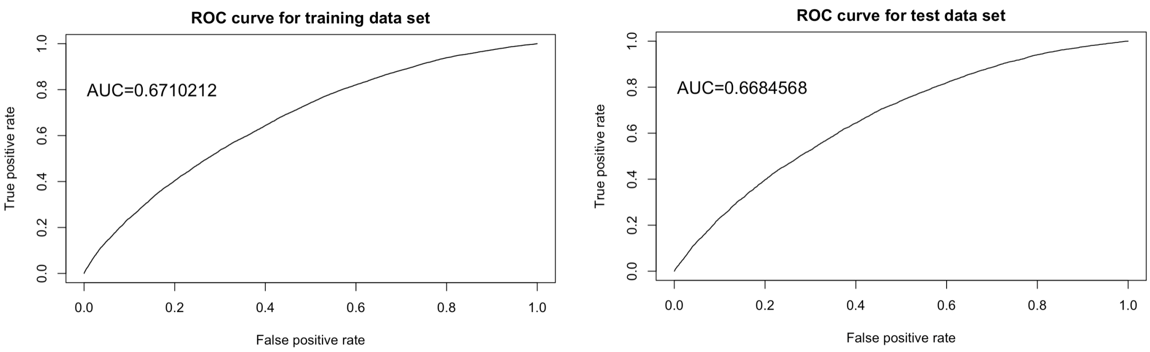
4 Q4: Build a scorecard using a single logistic regression model

```
Call:
glm(formula = def_flag ~ ., family = binomial("logit"), data = D2_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5968   0.3418   0.4304   0.5321   1.2381

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   3.229e+00  3.445e-02  93.729  < 2e-16 ***
loan_amnt     1.474e-06  1.263e-06   1.167    0.243
grade        -4.490e-01  7.856e-03 -57.151  < 2e-16 ***
emp_length_p  2.232e-02  2.737e-03   8.154 3.51e-16 ***
term          1.128e-01  2.461e-02   4.586 4.52e-06 ***
addr_state   -8.722e-01  8.175e-02 -10.670  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 76597  on 104723  degrees of freedom
Residual deviance: 72616  on 104718  degrees of freedom
AIC: 72628

Number of Fisher Scoring iterations: 5
```

5 Q5: Interpret your scorecard

The predictor loan_amnt does not show evidence of association with defaut, at a 1% significance level. There is sufficient evidence, at 1% significance level, that there is an association with default for the rest of the predictors: grade, emp_length_p, term and addr_state are significant at the significance level of 0.01. The coefficient of grade is negative, thus affirming our assumption above that a higher grade has a negative association with creditworthiness. The coefficient of emp_length_p is positive, thus a longer employment length has positive association with creditworthiness. The coefficient of term is positive, thus a loan term of 60 months has positive association with creditworthiness relative to a loan term of 36 months.

6 Q6: Construct the ROC curve and compute AUC



The AUC of the training data set is slightly greater than the AUC of the test data set. This is as expected because the training data set should in general fit the model better than the test data set as the the model was built using the training data set.

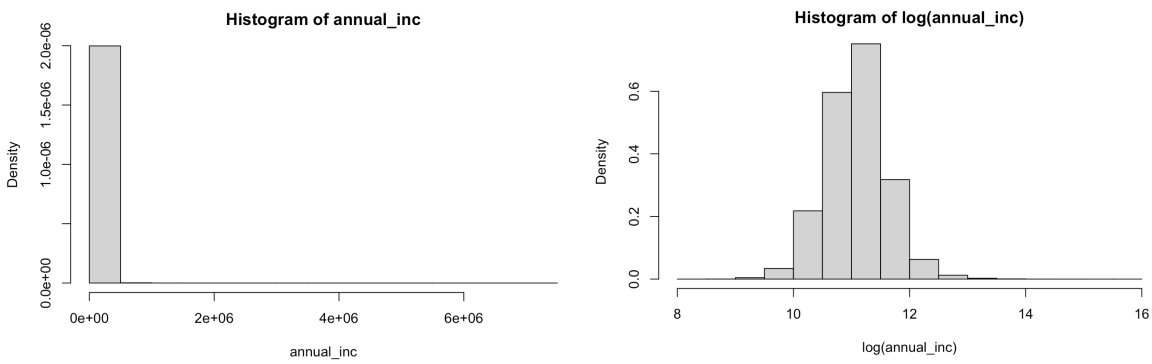
7 Q7: Improve the model

7.1 Data preparation and validation

7.1.1 addr_state

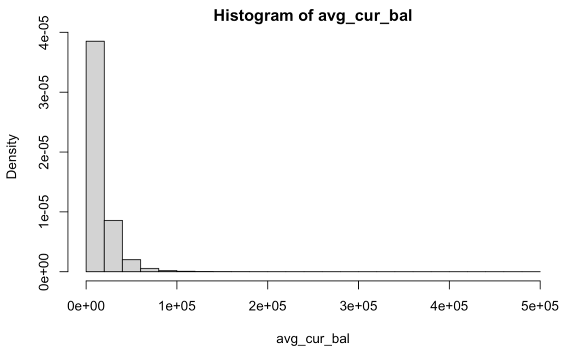
As the previous model, we will substitute the predictor addr_state with the continuous weights of evidence for each value.

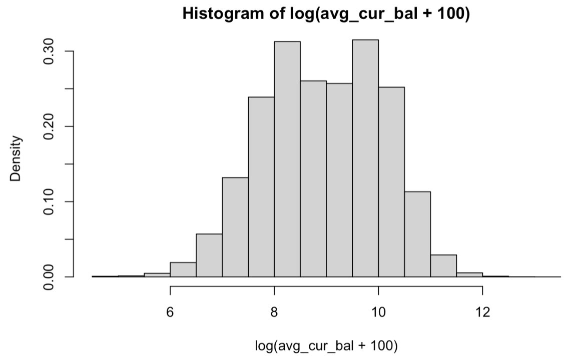
7.1.2 annual_inc



The histogram on the left shows that the distribution of annual_inc is highly positively skewed, thus we will apply logarithm to annual_inc so that it has distribution closer to normal as shown in the histogram on the right.

7.1.3 avg_cur_bal



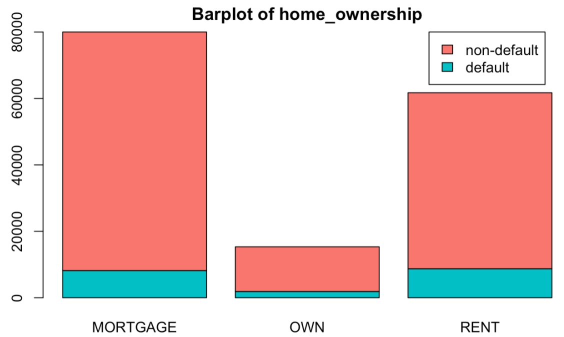


There are 3 missing values for avg_cur_bal and we will impute the missing values with the mean. The histogram on the left shows that the distribution of avg_cur_bal is highly positively skewed, thus we will apply logarithm to avg_cur_bal so that it has distribution closer to normal as shown in the histogram on the right.

7.1.4 emp_length_p

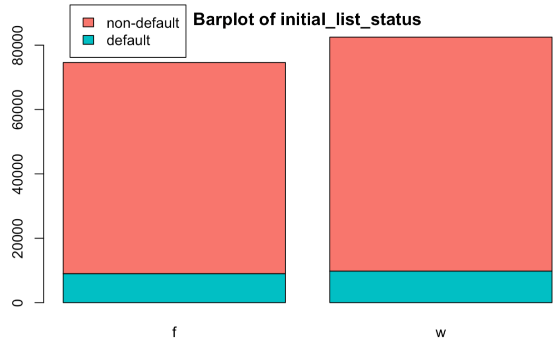
As the previous model, we will impute the 7956 missing values with the mean.

7.1.5 home_ownership



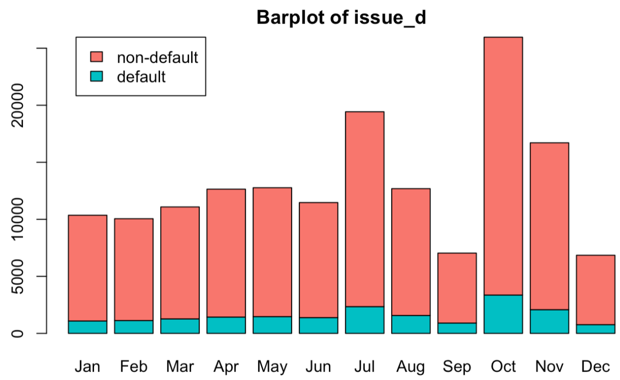
We will include home_ownership as two dummy variables for rent and own, with excluded category mortgage.

7.1.6 initial_list_status



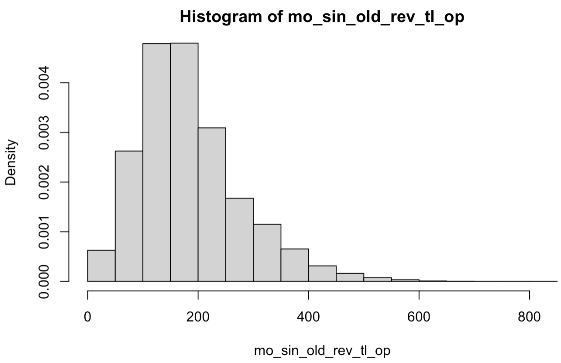
We will replace f which represents fractional loan with 0 and w which represents whole loan with 1.

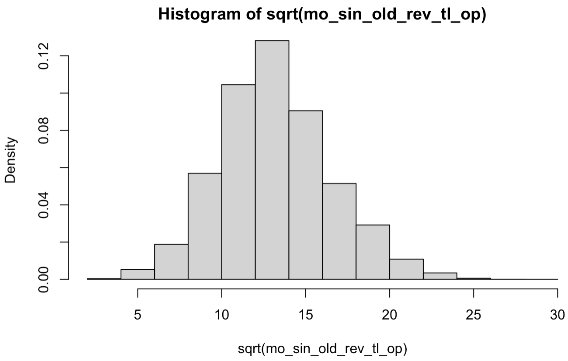
7.1.7 issue_d



We would not know beforehand whether or not a loan would be issued and when would the loan be issued. Including this predictor in will result in data leakage, thus we will remove this predictor.

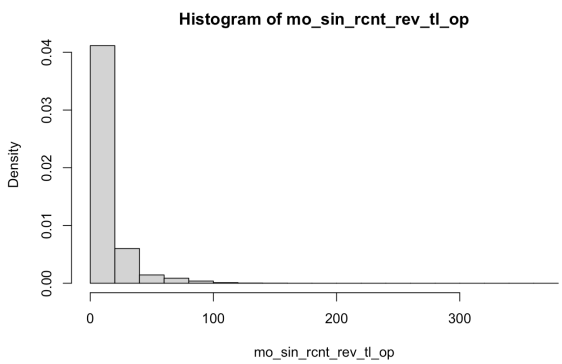
7.1.8 mo_sin_old_rev_tl_op

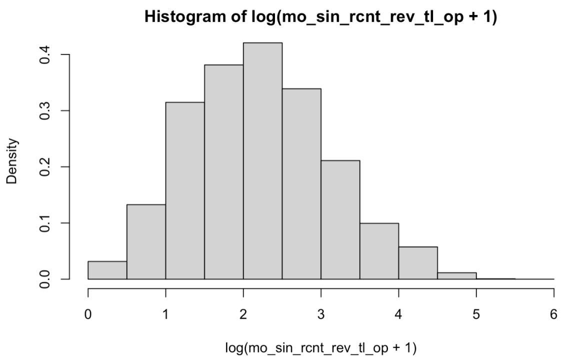




The histogram on the left shows that the distribution of mo_sin_old_rev_tl_op is slightly positively skewed, thus we will apply square root to mo_sin_old_rev_tl_op so that it has distribution closer to normal as shown in the histogram on the right.

7.1.9 mo_sin_rcnt_rev_tl_op





The histogram on the left shows that the distribution of mo_sin_rcnt_rev_tl_op is highly positively skewed, thus we will apply logarithm to mo_sin_old_rcnt_tl_op so that it has distribution closer to normal as shown in the histogram on the right.

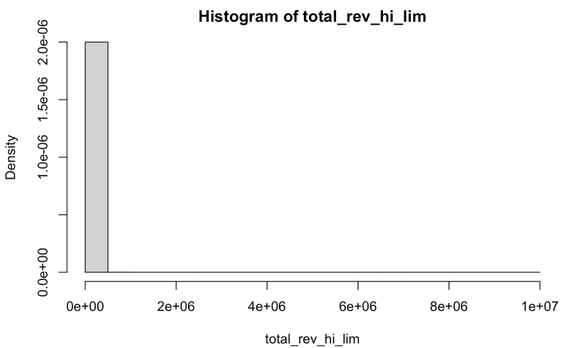
7.1.10 purpose_p

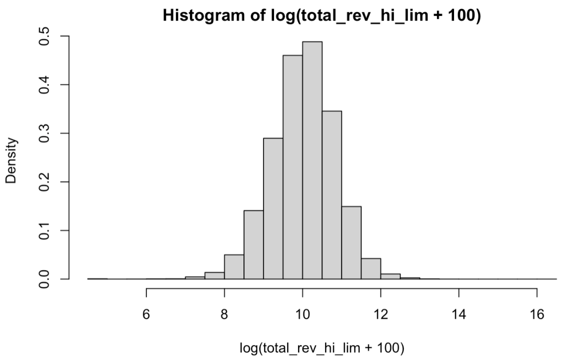
We will include purpose_p as 9 dummy variables for car, credit card, debt consolidation, home improvement, major purchase, medical, moving, small business and vacation, with other as the excluded category.

7.1.11 term

As the previous model, we will replace 36 months with 0 and 60 months with 1.

7.1.12 total_rev_hi_lim



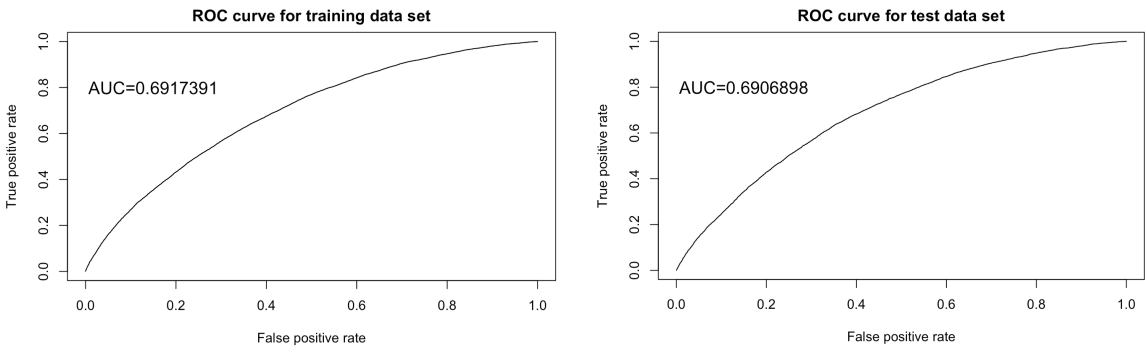


The histogram on the left shows that the distribution of total_rev_hi_lim is highly positively skewed, thus we will apply logarithm to total_rev_hi_lim so that it has distribution closer to normal as shown in the histogram on the right.

7.1.13 verification_status

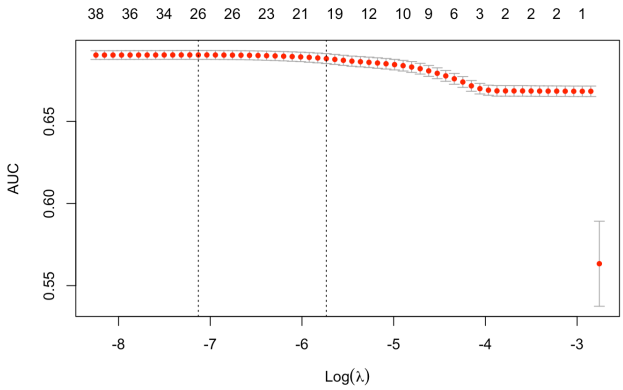
We will include verification_status as 2 dummy variables for income was verified by LC and income source verified, with not verified as the excluded category.

7.2 Model after Data Transformation



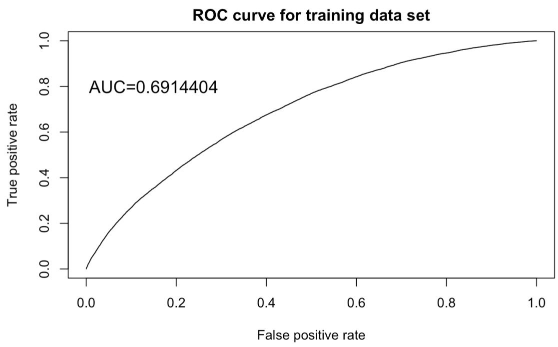
The model after data transformation has 42 predictor variables. The AUCs for both the training data set and test data set have increased after data processing, which implies that the data transformation techniques applied have been effective.

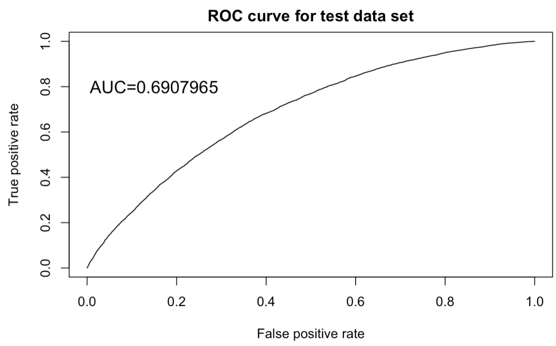
7.3 Variable Selection



We will perform variable selection via Least Absolute Shrinkage and Selection Operator (LASSO). We run cross-validation to obtain the best lambda value which maximises the AUC as 0.0008006879. LASSO regularisation reduces the number of of predictor variables from 42 to 26 variables as shown in the plot of lasso regularisation cross-validation above. The coefficient of the predictor variables: revol_bal, acc_now_delinq, chargeoff_within_12_mths, delinq_amnt, initial_list_status, num_accts_ever_120_pd, num_actv_bc_tl, num_bc_sats, open_acc, pub_rec_bankruptcies, purpose_credit _card, purpose_home_improvement, purpose_major_purchase, purpose_medical, purpose_vacation and own have been shrunk to 0.

7.4 Model after Variable Selection



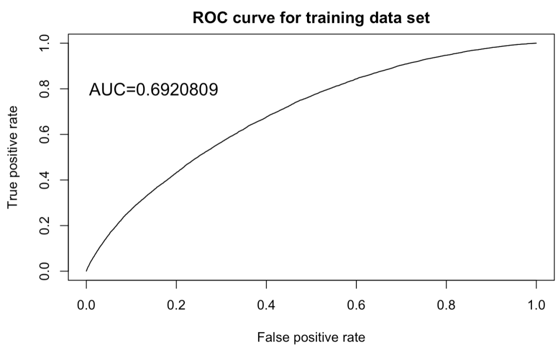


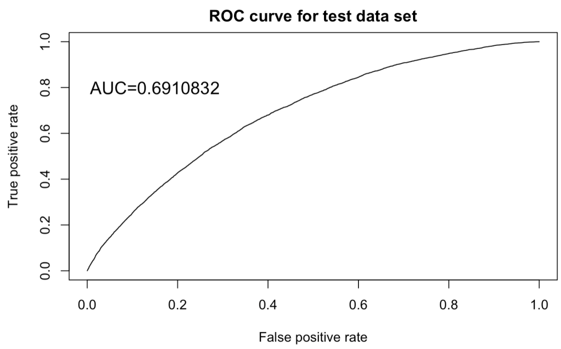
The model after variable selection has 26 variables as mentioned above. The AUC for training data set decreased, but the AUC for test data set has slightly increased. This implies that overfitting of the training data set has been decreased after variable selection.

7.5 Model with Interaction Terms

Interaction terms	$\Pr(>\|z\|)$
int_rate * grade	3.948717×10^{-8}
total_acc * source_verified	3.036798×10^{-4}
mo_sin_rcnt_rev_tl_op * total_rev_hi_lim	7.276919×10^{-4}
annual_inc * dti	1.055499×10^{-3}
annual_inc * purpose_debt_consolidation	1.073842×10^{-3}

We first run a logistic regression model which includes all possible interaction terms between two predictor variables. The table above shows the five interactions with the lowest $\Pr(>|z|)$ values. We will include the interaction terms int_rate * grade in the model since it has much lower $\Pr(>|z|)$ than the other interactions. We will further include two more interactions int_rate * annual_inc and loan_amnt * annual_inc in our model.

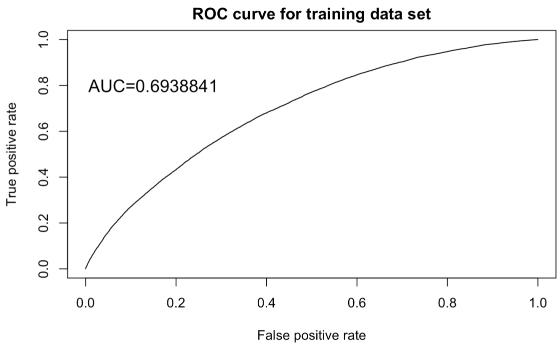


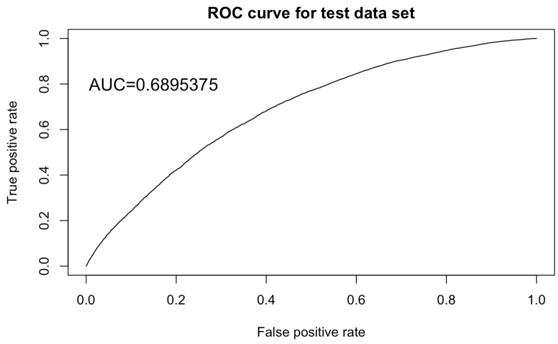


The model with interaction terms has 29 predictor variables. The AUCs of both the training data set and the test data set have increased, implying that the interaction terms are significant.

7.6 Segmented Model

With the assumption that different verification status will have different populations, we will segment out model by values in verification_status and then separate scorecard models built on each separate data segment. We build the segmented model using the data after data transformation. Segment 1 contains observations with income verified by LC, segment 2 contains observations with income source verified and segment 3 contains observations with income not verified. We will apply LASSO regularisation to each segment. The model for segment 1 has 35 predictor variables, the model for segment 2 has 32 predictor variables, and the model for segment 3 has 29 predictor variables. This shows that for each verification status, there are different predictor variables which are significant.





The AUC of the training data set has increased, but the AUC for the test data set has decreased. This implied that there is overfitting after segmenting the model on verification_status.

8 Results

To have a fair comparison between each of the models produced, we set seed as 1 when splitting the data into training data set and test data set to get the same training data set and test data set for each model.

Model	AUC (training)	AUC (test)
Model in step 4	0.6710212	0.6684568
Model after data transformation	0.6917391	0.6906898
Model after variable selection	0.6914401	0.6907965
Model with Interaction terms	0.6920809	0.6910382
Segmented model	0.6938841	0.6895375

The table above summarises the AUCs for both training data set and testing data set of all the models built in this project. The model which performed best is the model with interaction terms as it has the highest AUC for testing data set as compared to all the other models.

```
Call:
glm(formula = def_flag ~ ., family = binomial("logit"), data = D2_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5968   0.3418   0.4304   0.5321   1.2381

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.229e+00  3.445e-02  93.729  < 2e-16 ***
loan_amnt    1.474e-06  1.263e-06   1.167    0.243
grade       -4.490e-01  7.856e-03 -57.151  < 2e-16 ***
emp_length_p  2.232e-02  2.737e-03   8.154 3.51e-16 ***
term         1.128e-01  2.461e-02   4.586 4.52e-06 ***
addr_state  -8.722e-01  8.175e-02 -10.670  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 76597  on 104723  degrees of freedom
Residual deviance: 72616  on 104718  degrees of freedom
AIC: 72628

Number of Fisher Scoring iterations: 5
```

Coefficients of model in step 4

```
Coefficients:
                            Estimate Std. Error z value Pr(>|z|)
(Intercept)               -1.909e-01  8.958e-01  -0.213 0.831286
loan_amnt                 -1.227e-04  2.658e-05  -4.615 3.93e-06 ***
int_rate                  -2.456e-02  5.204e-02  -0.472 0.636964
grade                     -2.560e-01  3.710e-02  -6.900 5.21e-12 ***
emp_length_p               1.190e-02  2.894e-03   4.114 3.88e-05 ***
annual_inc                 2.814e-01  8.139e-02   3.457 0.000546 ***
term                       1.060e-01  2.614e-02   4.054 5.04e-05 ***
delinq_2yrs               -4.121e-02  1.047e-02  -3.936 8.29e-05 ***
avg_cur_bal                4.486e-02  1.356e-02   3.308 0.000938 ***
dti                       -1.314e-02  1.432e-03  -9.174  < 2e-16 ***
inq_last_6mths            -4.377e-02  9.527e-03  -4.594 4.34e-06 ***
mo_sin_old_rev_tl_op       3.181e-02  3.391e-03   9.381  < 2e-16 ***
mo_sin_rcnt_rev_tl_op      1.570e-02  1.504e-02   1.044 0.296322
mo_sin_rcnt_tl             1.468e-02  1.930e-03   7.606 2.83e-14 ***
mort_acc                   1.745e-02  6.297e-03   2.772 0.005579 **
num_actv_rev_tl           -2.036e-02  3.865e-03  -5.268 1.38e-07 ***
total_acc                 -2.577e-03  1.044e-03  -2.469 0.013551 *
total_rev_hi_lim           1.064e-01  1.757e-02   6.057 1.38e-09 ***
pub_rec                    3.527e-02  1.788e-02   1.973 0.048491 *
addr_state                -8.107e-01  8.216e-02  -9.868  < 2e-16 ***
rent                      -1.290e-01  2.457e-02  -5.251 1.51e-07 ***
purpose_car                2.708e-01  1.277e-01   2.120 0.034027 *
purpose_debt_consolidation -4.070e-02  2.120e-02  -1.920 0.054883 .
purpose_moving            -2.901e-01  1.081e-01  -2.684 0.007285 **
purpose_small_business    -3.115e-01  8.279e-02  -3.763 0.000168 ***
verified                  -1.071e-01  2.942e-02  -3.640 0.000273 ***
source_verified           -1.276e-01  2.678e-02  -4.767 1.87e-06 ***
int_rate:grade             1.074e-02  1.392e-03   7.720 1.16e-14 ***
int_rate:annual_inc       -9.486e-03  4.604e-03  -2.060 0.039366 *
loan_amnt:annual_inc       9.439e-06  2.342e-06   4.031 5.55e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coefficients of model with interaction terms

In contrast with model in step 4, in the model with interaction terms, there is sufficient evidence, at 1% significance level, that there is an association with default for the predictor variable loan_amnt. The rest of the predictor variables: grade, emp_length_p, term and addr_state remain as having associations with default at 1% significance level. loan_amnt has a negative coefficient with creditworthiness in the model with interaction terms, which differs from the coefficient of loan_amnt for the model in step 4. The association for the rest of the predictor variables: grade, emp_length_p, term and addr_state with creditworthiness remain the same in the model with interaction terms.

9 Appendix

9.1 Code for Q1

```
load("~/Desktop/Year 3/MATH60131 Consumer Credit Risk Modelling/Coursework/LCdata_2.RData")
```

9.2 Code for Q2

```
D2 <- D1[c("loan_amnt", "grade", "emp_length_p", "term", "addr_state",
↪  "def_flag")]

D2$def_flag <- 1 - as.numeric(D2$def_flag)

# loan_amnt
summary(D2$loan_amnt)
hist(D2$loan_amnt, freq=FALSE, main='Histogram of loan_amnt', xlab='loan_amnt')
boxplot(D2$loan_amnt, range=1.5)
title(main="Boxplot of loan_amnt")

# grade
summary(factor(D2$grade))
barplot(table(D2$def_flag, D2$grade), col=c("#00BFC4", "#F8766D"), legend =
↪  c('default', 'non-default'), main="Barplot of grade")

# emp_length_p

barplot(table(D2$def_flag, D2$emp_length_p), col=c("#00BFC4", "#F8766D"), legend =
↪  c('default', 'non-default'), main="Barplot of emp_length_p", args.legend =
↪  list(x = "topleft",inset = c(0.1, 0.1)))
sum(is.na(D2$emp_length_p))
D2$emp_length_p[is.na(D2$emp_length_p)] <- mean(D2$emp_length_p,na.rm = TRUE)

# term
summary(factor(D2$term))
barplot(table(D2$def_flag, D2$term), col=c("#00BFC4", "#F8766D"),
legend = c('default', 'non-default'), main="Barplot of term")
D2$term[which(D2$term==36)] <- 0
D2$term[which(D2$term==60)] <- 1

# addr_state
woe.tab <- function(x, y) {
  n1 <- sum(y)
  n0 <- sum(1 - y)
  nx0n1 <- tapply(1 - y, x, sum) * n1
  nx1n0 <- tapply(y, x, sum) * n0
  nx0n1[which(nx0n1==0)] <- n1
  nx1n0[which(nx1n0==0)] <- n0
  return(log(nx0n1) - log(nx1n0))
}

woe.assign <- function(wtab, x) {
  w <- rep(0, length(x))
```

```
  ni <- names(wtab)
  for (i in 1:length(ni)) {
    w[which(x==ni[i])] <- wtab[i]
  }
  return(w)
}

summary(factor(D2$addr_state))
D2$addr_state <- woe.assign(woe.tab(D2$addr_state, D2$def_flag), D2$addr_state)
```

9.3 Code for Q3

```
set.seed(1)
ix <- sample(157085, 52361, replace=FALSE)
D2_test <- D2[ix,]
D2_train <- D2[-ix,]
```

9.4 Code for Q4

```
glm1.out <- glm(def_flag ~ ., data = D2_train, family = binomial("logit"))
summary(glm1.out)
```

9.5 Code for Q6

```
# ROC function
roc <- function(y, s){
  yav <- rep(tapply(y, s, mean), table(s))
  rocx <- cumsum(yav)
  rocy <- cumsum(1 - yav)
  area <- sum(yav * (rocy - 0.5 * (1 - yav)))
  x1 <- c(0, rocx) / sum(y)
  y1 <- c(0, rocy) / sum(1 - y)
  auc <- area / (sum(y) * sum(1 - y))
  print(auc)
  plot(x1, y1, "l", xlab="False positive rate", ylab="True positive rate")
}

yp1 <- predict(glm1.out, D2_test, type="response")
roc(D2_train$def_flag, glm1.out$fitted.values)
title(main="ROC curve for training data set")
text(x = 0.15, y = 0.8, "AUC=0.6710212", cex=1.3)
roc(D2_test$def_flag, yp1)
title(main="ROC curve for test data set")
text(x = 0.15, y = 0.8, "AUC=0.6684568", cex=1.3)
```

9.6 Code for Q7

9.6.1 Code for Data preparation and validation

```
D3 <- data.frame(D1)
D3$def_flag <- 1 - as.numeric(D3$def_flag)

# addr_state
D3$addr_state <- woe.assign(woe.tab(D3$addr_state, D3$def_flag), D3$addr_state)

# annual_inc
hist(D3$annual_inc, freq=FALSE, main="Histogram of annual_inc", xlab="annual_inc")
hist(log(D3$annual_inc), freq=FALSE, main="Histogram of log(annual_inc)",
↪  xlab="log(annual_inc)")
```

```
D3$annual_inc <- log(D3$annual_inc)

# avg_cur_bal
D3$avg_cur_bal[is.na(D3$avg_cur_bal)] <- mean(D3$avg_cur_bal,na.rm = TRUE)
hist(D3$avg_cur_bal, freq=FALSE, main="Histogram of avg_cur_bal",
↪  xlab="avg_cur_bal")
hist(log(D3$avg_cur_bal + 100), freq=FALSE, main="Histogram of log(avg_cur_bal +
↪  100)", xlab="log(avg_cur_bal + 100)")
D3$avg_cur_bal <- log(D3$avg_cur_bal + 100)

# emp_length_p
D3$emp_length_p[is.na(D3$emp_length_p)] <- mean(D3$emp_length_p,na.rm = TRUE)

# home_ownership
summary(factor(D3$home_ownership))
barplot(table(D3$def_flag, D3$home_ownership), col=c("#00BFC4", "#F8766D"), legend
↪  = c('default', 'non-default'), args.legend = list(x = "topright",inset =
↪  c(0.05, 0)), main="Barplot of home_ownership")
D3$rent <- as.numeric(D3$home_ownership=='RENT')
D3$own <- as.numeric(D3$home_ownership=='OWN')
D3 <- subset(D3, select = -c(home_ownership))

# initial_list_status
summary(factor(D3$initial_list_status))
barplot(table(D3$def_flag, D3$initial_list_status), col=c("#00BFC4", "#F8766D"),
↪  legend = c('default', 'non-default'), args.legend = list(x = "topleft",inset =
↪  c(0.05, -0.12)), main="Barplot of initial_list_status")
D3$initial_list_status <- as.character(D3$initial_list_status)
D3$initial_list_status[which(D3$initial_list_status=='f')] <- 0
D3$initial_list_status[which(D3$initial_list_status=='w')] <- 1

# issue_d
summary(factor(D3$issue_d))
D3$issue_d <- gsub("-2014","",as.character(D3$issue_d))
D3$issue_d <- factor(D3$issue_d, levels=month.abb)
barplot(table(D3$def_flag, D3$issue_d), col=c("#00BFC4", "#F8766D"), legend =
↪  c('default', 'non-default'), args.legend = list(x = "topleft",inset = c(0.05,
↪  0)), main="Barplot of issue_d")
D3 <- subset(D3, select = -c(issue_d))

# mo_sin_old_rev_tl_op
summary(D3$mo_sin_old_rev_tl_op)
hist(D3$mo_sin_old_rev_tl_op, freq=FALSE, main="Histogram of
↪  mo_sin_old_rev_tl_op", xlab="mo_sin_old_rev_tl_op")
hist(sqrt(D3$mo_sin_old_rev_tl_op), freq=FALSE, main="Histogram of
↪  sqrt(mo_sin_old_rev_tl_op)", xlab="sqrt(mo_sin_old_rev_tl_op)")
boxplot(D3$mo_sin_old_rev_tl_op, range=1.5)
D3$mo_sin_old_rev_tl_op <- sqrt(D3$mo_sin_old_rev_tl_op)

# mo_sin_rcnt_rev_tl_op
summary(D3$mo_sin_rcnt_rev_tl_op)
hist(D3$mo_sin_rcnt_rev_tl_op, freq=FALSE, main="Histogram of
↪  mo_sin_rcnt_rev_tl_op", xlab="mo_sin_rcnt_rev_tl_op")
hist(log(D3$mo_sin_rcnt_rev_tl_op+1), freq=FALSE, main="Histogram of
↪  log(mo_sin_rcnt_rev_tl_op + 1)", xlab="log(mo_sin_rcnt_rev_tl_op + 1)")
boxplot(D3$mo_sin_rcnt_rev_tl_op, range=1.5)
D3$mo_sin_rcnt_rev_tl_op <- log(D3$mo_sin_rcnt_rev_tl_op+1)
```

```r
# purpose_p
summary(factor(D3$purpose_p))
barplot(table(D3$def_flag, D3$purpose_p), col=c("#00BFC4", "#F8766D"), legend =
↪  c('default', 'non-default'))
D3$purpose_car <- as.numeric(D3$purpose_p=='car')
D3$purpose_credit_card <- as.numeric(D3$purpose_p=='credit_card')
D3$purpose_debt_consolidation <- as.numeric(D3$purpose_p=='debt_consolidation')
D3$purpose_home_improvement <- as.numeric(D3$purpose_p=='home_improvement')
D3$purpose_major_purchase <- as.numeric(D3$purpose_p=='major_purchase')
D3$purpose_medical <- as.numeric(D3$purpose_p=='medical')
D3$purpose_moving <- as.numeric(D3$purpose_p=='moving')
D3$purpose_small_business <- as.numeric(D3$purpose_p=='small_business')
D3$purpose_vacation <- as.numeric(D3$purpose_p=='vacation')
D3 <- subset(D3, select = -c(purpose_p))

# term
D3$term[which(D3$term==36)] <- 0
D3$term[which(D3$term==60)] <- 1

# total_rev_hi_lim
hist(D3$total_rev_hi_lim, freq=FALSE, main="Histogram of total_rev_hi_lim",
↪  xlab="total_rev_hi_lim")
hist(log(D3$total_rev_hi_lim+100), freq=FALSE, main="Histogram of
↪  log(total_rev_hi_lim + 100)", xlab="log(total_rev_hi_lim + 100)")
D3$total_rev_hi_lim <- log(D3$total_rev_hi_lim+100)

# verification_status
summary(factor(D3$verification_status))
barplot(table(D3$def_flag, D3$verification_status), col=c("#00BFC4", "#F8766D"),
↪  legend = c('default', 'non-default'))
D3$verified <- as.numeric(D3$verification_status=='Verified')
D3$source_verified <- as.numeric(D3$verification_status=='Source Verified')
D3 <- subset(D3, select = -c(verification_status))
```

9.6.2 Code for Model after Data Transformation

```r
# Split the data using the previous seed so that we have the same training data set and test data
set.seed(1)
D3_test <- D3[ix,]
D3_train <- D3[-ix,]

# Model after data transformation
glm2.out <- glm(def_flag ~ ., data = D3_train, family = binomial("logit"))
summary(glm2.out)
yp2 <- predict(glm2.out, D3_test, type="response")
roc(D3_train$def_flag, glm2.out$fitted.values)
title(main="ROC curve for training data set")
text(x = 0.15, y = 0.8, "AUC=0.6917391", cex=1.3)
roc(D3_test$def_flag, yp2)
title(main="ROC curve for test data set")
text(x = 0.15, y = 0.8, "AUC=0.6906898", cex=1.3)
```

9.6.3 Code for Variable Selection

```r
# LASSO
library(glmnet)
X <- data.matrix(D3_train)
```

```
X <- X[,-1]
lasso_cv <- cv.glmnet(X, D3_train$def_flag, type.measure="auc", alpha=1,
↪  family="binomial")
plot(lasso_cv)
best_lambda <- lasso_cv$lambda.min
glm3.out <- glmnet(X, D3_train$def_flag, alpha = 1, family = "binomial",lambda =
↪  best_lambda)
coef(glm3.out)

D4_train <- subset(D3_train, select = -c(revol_bal, acc_now_delinq,
↪  chargeoff_within_12_mths, delinq_amnt, initial_list_status,
↪  num_accts_ever_120_pd, num_actv_bc_tl, num_bc_sats, open_acc,
↪  pub_rec_bankruptcies, purpose_credit_card, purpose_home_improvement,
↪  purpose_major_purchase, purpose_medical, purpose_vacation, own))
D4_test <- subset(D3_test, select = -c(revol_bal, acc_now_delinq,
↪  chargeoff_within_12_mths, delinq_amnt, initial_list_status,
↪  num_accts_ever_120_pd, num_actv_bc_tl, num_bc_sats, open_acc,
↪  pub_rec_bankruptcies, purpose_credit_card, purpose_home_improvement,
↪  purpose_major_purchase, purpose_medical, purpose_vacation, own))

# Model after Variable Selection
glm4.out <- glm(def_flag ~ . , data = D4_train, family = binomial("logit"))
summary(glm4.out)
yp4 <- predict(glm4.out, D4_test, type="response")
roc(D4_train$def_flag, glm4.out$fitted.values)
title(main="ROC curve for training data set")
text(x = 0.15, y = 0.8, "AUC=0.6914404", cex=1.3)
roc(D4_test$def_flag, yp4)
title(main="ROC curve for test data set")
text(x = 0.15, y = 0.8, "AUC=0.6907965", cex=1.3)
```

9.6.4 Model with Interaction terms

```
# all possible interaction terms between two predictor variables
glm5.out <- glm(def_flag ~ . ^2, data = D4_train, family = binomial("logit"))
summary(glm5.out)

# model with three interaction terms added
glm6.out <- glm(def_flag ~ . + int_rate*grade + int_rate*annual_inc +
↪  loan_amnt*annual_inc, data = D4_train, family = binomial("logit"))
summary(glm6.out)
yp6 <- predict(glm6.out, D4_test, type="response")
roc(D4_train$def_flag, glm6.out$fitted.values)
title(main="ROC curve for training data set")
text(x = 0.15, y = 0.8, "AUC=0.6920809", cex=1.3)
roc(D4_test$def_flag, yp6)
title(main="ROC curve for test data set")
text(x = 0.15, y = 0.8, "AUC=0.6910832", cex=1.3)
```

9.6.5 Segmented model

```
# Split training and test data into three segments
seg_1_train <- D3_train[(D3_train$verified==1),]
seg_2_train <- D3_train[(D3_train$source_verified==1),]
seg_3_train <- D3_train[(D3_train$verified==0 & D3_train$source_verified==0),]
seg_1_test <- D3_test[(D3_test$verified==1),]
seg_2_test <- D3_test[(D3_test$source_verified==1),]
seg_3_test <- D3_test[(D3_test$verified==0 & D3_test$source_verified==0),]
```

```r
seg_1_train <- subset(seg_1_train, select = -c(verified, source_verified))
seg_2_train <- subset(seg_2_train, select = -c(verified, source_verified))
seg_3_train <- subset(seg_3_train, select = -c(verified, source_verified))
seg_1_test <-  subset(seg_1_test, select = -c(verified, source_verified))
seg_2_test <-  subset(seg_2_test, select = -c(verified, source_verified))
seg_3_test <-  subset(seg_3_test, select = -c(verified, source_verified))

# Model for segment 1
X_seg_train_1 <- data.matrix(seg_1_train)
Y_seg_train_1 <- X_seg_train_1[,1]
X_seg_train_1 <- X_seg_train_1[,-1]
lasso_cv_1 <- cv.glmnet(X_seg_train_1, Y_seg_train_1, type.measure="auc", alpha=1,
↪  family="binomial")
plot(lasso_cv_1)
best_lambda <- lasso_cv_1$lambda.min
glm7.out <- glmnet(X_seg_train_1, Y_seg_train_1, alpha = 1, family =
↪  "binomial",lambda = best_lambda)
coef(glm7.out)
X_seg_test_1 <- data.matrix(seg_1_test)
Y_seg_test_1 <- X_seg_test_1[,1]
X_seg_test_1 <- X_seg_test_1[,-1]
yp7_train <- predict(glm7.out, X_seg_train_1, type="response")
yp7 <- predict(glm7.out, X_seg_test_1, type="response")

# Model for Segment 2
X_seg_train_2 <- data.matrix(seg_2_train)
Y_seg_train_2 <- X_seg_train_2[,1]
X_seg_train_2 <- X_seg_train_2[,-1]
lasso_cv_2 <- cv.glmnet(X_seg_train_2, Y_seg_train_2, type.measure="auc", alpha=1,
↪  family="binomial")
plot(lasso_cv_2)
best_lambda <- lasso_cv_2$lambda.min
glm8.out <- glmnet(X_seg_train_2, Y_seg_train_2, alpha = 1, family =
↪  "binomial",lambda = best_lambda)
coef(glm8.out)
X_seg_test_2 <- data.matrix(seg_2_test)
Y_seg_test_2 <- X_seg_test_2[,1]
X_seg_test_2 <- X_seg_test_2[,-1]
yp8_train <- predict(glm8.out, X_seg_train_2, type="response")
yp8 <- predict(glm8.out, X_seg_test_2, type="response")

# Model for Segment 3
X_seg_train_3 <- data.matrix(seg_3_train)
Y_seg_train_3 <- X_seg_train_3[,1]
X_seg_train_3 <- X_seg_train_3[,-1]
lasso_cv_3 <- cv.glmnet(X_seg_train_3, Y_seg_train_3, type.measure="auc", alpha=1,
↪  family="binomial")
plot(lasso_cv_3)
best_lambda <- lasso_cv_3$lambda.min
glm9.out <- glmnet(X_seg_train_3, Y_seg_train_3, alpha = 1, family =
↪  "binomial",lambda = best_lambda)
coef(glm9.out)
X_seg_test_3 <- data.matrix(seg_3_test)
Y_seg_test_3 <- X_seg_test_3[,1]
X_seg_test_3 <- X_seg_test_3[,-1]
yp9_train <- predict(glm9.out, X_seg_train_3, type="response")
```

```r
yp9 <- predict(glm9.out, X_seg_test_3, type="response")

# Plor ROC curve and calculate AUC
Y_train <- c(Y_seg_train_1, Y_seg_train_2, Y_seg_train_3)
Y_test <- c(Y_seg_test_1, Y_seg_test_2, Y_seg_test_3)
yp_train <- c(yp7_train, yp8_train, yp9_train)
yp_test <- c(yp7, yp8, yp9)
roc(Y_train, yp_train)
title(main="ROC curve for training data set")
text(x = 0.15, y = 0.8, "AUC=0.6938841", cex=1.3)
roc(Y_test, yp_test)
title(main="ROC curve for test data set")
text(x = 0.15, y = 0.8, "AUC=0.6895375", cex=1.3)
```