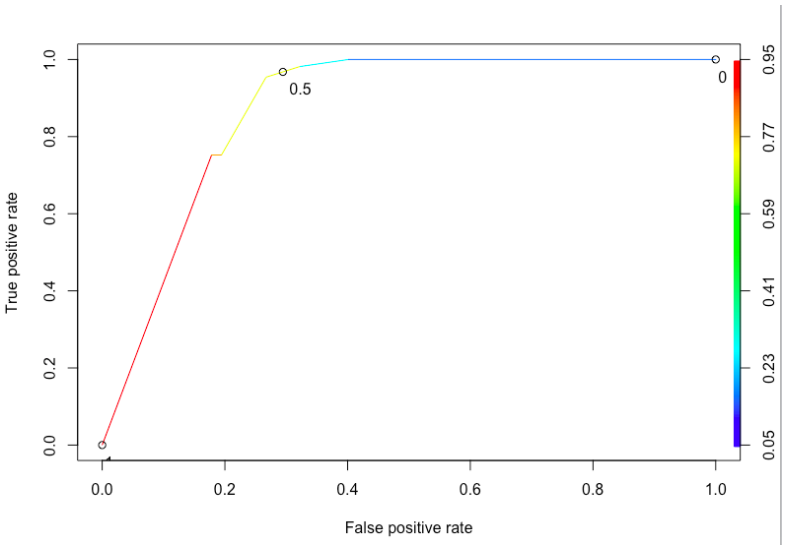
APPENDIX

```
> cor(train_data[,10:16])
                num_25_mean num_50_mean num_75_mean num_985_mean num_100_mean
num_25_mean      1.00000000  0.64467751   0.6524860    0.4786702   0.04031162
num_50_mean      0.64467751  1.00000000   0.8576572    0.5021165   0.03731589
num_75_mean      0.65248597  0.85765724   1.0000000    0.6543823   0.10455331
num_985_mean     0.47867024  0.50211650   0.6543823    1.0000000   0.19044775
num_100_mean     0.04031162  0.03731589   0.1045533    0.1904477   1.00000000
num_unq_mean     0.38262441  0.30598170   0.3454604    0.3726383   0.79467163
total_secs_mean  0.08637770  0.09490440   0.1669607    0.2537751   0.97798163
                num_unq_mean total_secs_mean
num_25_mean        0.3826244       0.0863777
num_50_mean        0.3059817       0.0949044
num_75_mean        0.3454604       0.1669607
num_985_mean       0.3726383       0.2537751
num_100_mean       0.7946716       0.9779816
num_unq_mean       1.0000000       0.8133865
total_secs_mean    0.8133865       1.0000000
> cor(train_data[,17:23])
                 num_25_var  num_50_var  num_75_var num_985_var  num_100_var
num_25_var      1.000000000  0.30674015 0.162909151  0.08048652  1.000000000
num_50_var      0.306740152  1.00000000 0.452915745  0.07959494  0.306740152
num_75_var      0.162909151  0.45291575 1.000000000  0.17682953  0.162909151
num_985_var     0.080486520  0.07959494 0.176829528  1.00000000  0.080486520
num_100_var     1.000000000  0.30674015 0.162909151  0.08048652  1.000000000
num_unq_var     0.368902232  0.28728911 0.234489076  0.20584717  0.368902232
total_secs_var -0.003940434 -0.00452676 0.002032034  0.00585705 -0.003940434
               num_unq_var total_secs_var
num_25_var       0.3689022   -0.003940434
num_50_var       0.2872891   -0.004526760
num_75_var       0.2344891    0.002032034
num_985_var      0.2058472    0.005857050
num_100_var      0.3689022   -0.003940434
num_unq_var      1.0000000    0.040502799
total_secs_var   0.0405028    1.000000000
```

```
num_25_mean         -4.587e-02  2.468e-02  -1.859 0.063041 .
num_50_mean          2.251e-01  1.449e-01   1.554 0.120158
num_75_mean         -2.888e-01  2.728e-01  -1.058 0.289834
num_985_mean         1.449e-01  1.260e-01   1.149 0.250355
num_100_mean         3.009e-02  2.762e-02   1.089 0.275970
num_unq_mean         1.617e-02  1.126e-02   1.436 0.150927
total_secs_mean     -1.983e-04  1.260e-04  -1.574 0.115523
num_25_var           2.151e-04  3.393e-04   0.634 0.526132
num_50_var          -6.537e-03  4.158e-03  -1.572 0.115902
num_75_var           1.026e-04  1.176e-02   0.009 0.993038
num_985_var          4.147e-04  4.821e-03   0.086 0.931456
num_100_var                 NA         NA      NA       NA
num_unq_var          1.644e-05  1.416e-04   0.116 0.907592
total_secs_var       4.409e-11  7.173e-11   0.615 0.538742
```

```
> summary(tree_1)
Call:
rpart(formula = is_churn ~ city + bd + gender + registered_via +
    payment_method_id + cancel_number + autorenew_number + total_days +
    num_25_mean + num_50_mean + num_75_mean + num_985_mean +
    num_100_mean + num_unq_mean + total_secs_mean + num_25_var +
    num_50_var + num_75_var + num_985_var + num_unq_var + total_secs_var,
    data = train, method = "class", minbucket = 15)
  n= 3014

          CP nsplit rel error   xerror       xstd
1 0.01574803      0 1.0000000 1.000000 0.06004351
2 0.01000000      5 0.9015748 1.122047 0.06324393

Variable importance
 autorenew_number payment_method_id        total_days       num_25_mean
               42                37                 5                 3
       num_25_var       num_50_mean              city       num_75_mean
                2                 2                 2                 2
               bd      num_985_mean   total_secs_mean
                1                 1                 1


> ct_train_1
   pred_train_1
       0    1
  0 2740   20
  1  209   45

> ct_test_1
   pred_test_1
       0    1
  0 1170   13
  1   87   22

> summary(tree_2)
Call:
rpart(formula = is_churn ~ city + bd + gender + autorenew_number +
    registered_via + payment_method_id + cancel_number + total_days +
    num_25_mean + num_50_mean + num_75_mean + num_985_mean +
    num_100_mean + num_unq_mean + total_secs_mean + num_25_var +
    num_50_var + num_75_var + num_985_var + num_unq_var + total_secs_var,
    data = train_DT, method = "class", minbucket = 15)
  n= 5514

          CP nsplit rel error    xerror       xstd
1 0.51307190      0 1.0000000 1.0406681 0.01347100
2 0.05228758      1 0.4869281 0.4963689 0.01164271
3 0.02069717      3 0.3823529 0.3758170 0.01052842
4 0.01000000      5 0.3409586 0.3449528 0.01018210

Variable importance
 autorenew_number payment_method_id     cancel_number        total_days    registered
_via                bd
               29                22                16                15
8                 8
             city
                2


> ct_train_2
   pred_train_2
       0    1
  0 2029  731
  1  208 2546

> ct_test_2
   pred_test_2
       0   1
  0 867 316
  1   5 104
```

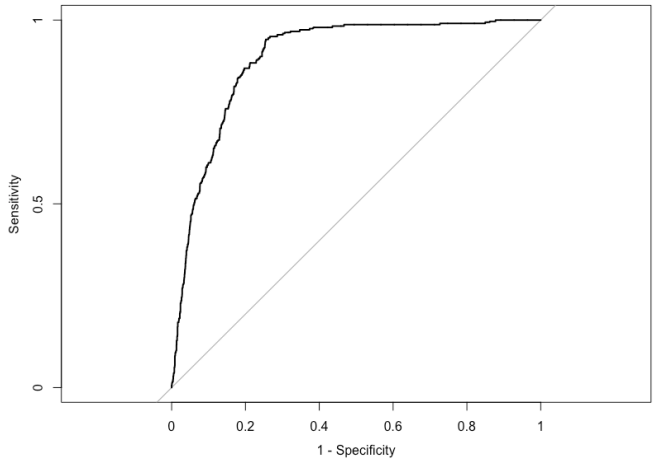



```
> print(randomforest)

Call:
 randomForest(formula = is_churn ~ ., data = train_RF, importance = TRUE,      proxim
ity = FALSE, ntree = 100)
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 4

        OOB estimate of  error rate: 24.01%
Confusion matrix:
    0   1 class.error
0 184  66   0.2640000
1  55 199   0.2165354


> caret::confusionMatrix(pred_rf, test$is_churn)
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 915  16
         1 268  93


> confusionMatrix(data =lr_pred_te:
n,positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 909  29
         1 273  80
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.3826  -0.5310   0.0000   0.5903   2.7599

Coefficients: (1 not defined because of singularities)
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)          4.138e+00  1.108e+00   3.736 0.000187 ***
city3               -2.117e-01  3.878e-01  -0.546 0.585130
city4               -4.308e-01  2.305e-01  -1.869 0.061651 .
city5               -1.084e+00  2.271e-01  -4.773 1.81e-06 ***
city6               -8.824e-01  2.484e-01  -3.552 0.000382 ***
city7               -1.753e+01  4.519e+02  -0.039 0.969052
city8               -7.255e-01  3.147e-01  -2.305 0.021143 *
city9               -2.733e+00  5.473e-01  -4.993 5.95e-07 ***
city10              -3.709e-01  3.641e-01  -1.018 0.308456
city11              -8.233e-01  3.526e-01  -2.335 0.019566 *
city12              -2.511e+00  3.575e-01  -7.023 2.17e-12 ***
city13              -6.244e-01  2.199e-01  -2.839 0.004527 **
city14               7.662e-02  2.575e-01   0.298 0.765993
city15              -6.479e-01  2.366e-01  -2.738 0.006180 **
city16              -1.922e+01  9.182e+02  -0.021 0.983302
city17              -1.681e+01  3.166e+02  -0.053 0.957643
city18              -1.389e+00  4.128e-01  -3.366 0.000764 ***
city19              -1.670e+01  1.052e+03  -0.016 0.987334
city20              -1.594e+01  1.238e+03  -0.013 0.989727
city21              -8.551e-01  4.502e-01  -1.899 0.057549 .
city22              -1.224e+00  2.388e-01  -5.127 2.94e-07 ***
bd                   9.392e-03  4.961e-03   1.893 0.058340 .
genderfemale         2.276e-01  2.392e-01   0.952 0.341204
gendermale           1.287e-01  2.401e-01   0.536 0.592000
registered_via4     -8.959e-02  1.416e-01  -0.633 0.527018
registered_via7     -4.756e-01  3.196e-01  -1.488 0.136722
registered_via9      2.000e-01  1.005e-01   1.990 0.046599 *
registered_via13    -1.624e+01  1.374e+03  -0.012 0.990574
payment_method_id11 -1.674e+01  1.238e+03  -0.014 0.989215
payment_method_id12  1.410e+01  7.588e+02   0.019 0.985172
payment_method_id13 -1.884e+01  1.691e+03  -0.011 0.991106
payment_method_id14 -1.732e+01  1.190e+03  -0.015 0.988385
payment_method_id15  1.408e+01  7.998e+02   0.018 0.985955
payment_method_id16 -2.093e+01  8.738e+02  -0.024 0.980888
payment_method_id17  1.215e+00  1.460e+00   0.832 0.405210
payment_method_id18 -2.164e+01  1.028e+03  -0.021 0.983204
payment_method_id19 -1.693e+01  7.809e+02  -0.022 0.982706
payment_method_id20 -9.133e-01  1.169e+00  -0.781 0.434607
payment_method_id21 -1.671e+01  1.134e+03  -0.015 0.988244
payment_method_id22 -1.842e+00  1.255e+00  -1.467 0.142252
payment_method_id23 -1.938e+01  1.042e+03  -0.019 0.985160
payment_method_id27 -4.406e+00  1.347e+00  -3.271 0.001071 **
payment_method_id28 -1.909e+00  1.107e+00  -1.724 0.084651 .
payment_method_id29 -3.619e+00  1.066e+00  -3.397 0.000682 ***
payment_method_id30 -3.801e+00  1.093e+00  -3.477 0.000506 ***
payment_method_id31 -3.158e+00  1.104e+00  -2.859 0.004246 **
payment_method_id32 -5.628e-01  1.073e+00  -0.524 0.599945
payment_method_id33 -2.363e+00  1.084e+00  -2.181 0.029214 *
payment_method_id34 -2.806e+00  1.082e+00  -2.593 0.009523 **
payment_method_id35 -1.541e+00  1.085e+00  -1.420 0.155694
payment_method_id36 -3.922e+00  1.057e+00  -3.712 0.000205 ***
payment_method_id37 -2.489e+00  1.066e+00  -2.334 0.019602 *
payment_method_id38 -1.540e+00  1.050e+00  -1.467 0.142295
payment_method_id39 -3.043e+00  1.060e+00  -2.871 0.004098 **
payment_method_id40 -3.368e+00  1.057e+00  -3.187 0.001437 **
payment_method_id41 -3.071e+00  1.107e+00  -2.774 0.005536 **
cancel_number        1.965e+00  8.068e-02  24.353  < 2e-16 ***
autorenew_number    -1.202e-01  9.811e-03 -12.256  < 2e-16 ***
total_days          -1.288e-03  2.801e-04  -4.596 4.31e-06 ***
num_25_mean         -5.061e-02  1.378e-02  -3.672 0.000241 ***
num_50_mean          1.734e-01  6.506e-02   2.665 0.007710 **
num_985_mean        -1.764e-02  6.405e-02  -0.275 0.782982
num_100_mean        -5.534e-03  2.536e-03  -2.182 0.029104 *
num_25_var           4.755e-04  1.984e-04   2.397 0.016518 *
num_50_var          -6.484e-03  2.103e-03  -3.083 0.002048 **
num_75_var           7.886e-03  4.150e-03   1.900 0.057420 .
num_985_var          3.538e-03  2.493e-03   1.419 0.155936
num_100_var                 NA         NA      NA       NA
num_unq_var         -1.473e-04  9.189e-05  -1.603 0.108936
total_secs_var      -1.324e-11  4.180e-11  -0.317 0.751511
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7644.0  on 5513  degrees of freedom
Residual deviance: 4488.7  on 5445  degrees of freedom
AIC: 4626.7

Number of Fisher Scoring iterations: 15
```

```
> confusionMatrix(data =lr_pred_t
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 2264  438
         1  496 2316
```

```
> confusionMatrix(data =lr_pred_te
n,positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 909  29
         1 273  80
```



R Code

```
#Read Data

library(data.table)

train_data=fread("/Users/siyang/Desktop/Study/Applied Analytics/5200/Project/train_data.csv")

train_data=na.exclude(train_data)

train_data=train_data[,2:24]

train_data$city=as.factor(train_data$city)

train_data$gender=as.factor(train_data$gender)

train_data$registered_via=as.factor(train_data$registered_via)

train_data$is_churn=as.factor(train_data$is_churn)

train_data$payment_method_id=as.factor(train_data$payment_method_id)


#Detect Multicollinearity

cor(train_data[,10:16])

cor(train_data[,17:23])

model<glm(is_churn~city+bd+gender+registered_via+payment_method_id+cancel_number+autorenew_number+total_days+num_25_mean+num_50_mean+num_75_mean+num_985_mean+num_100_mean+num_unq_mean+total_secs_mean+num_25_var+num_50_var+num_75_var+num_985_var+num_100_var+num_unq_var+total_secs_var,family=binomial(logit),data=train_data)

summary(model)


#Split Data
```

```
library(caTools)

split = sample.split(train_data$is_churn,SplitRatio = 0.7)

train = train_data[split,]

test = train_data[!split,]


#Build Decision Tree - First Try

library(rpart)

library(rpart.plot)


tree_1 = rpart(is_churn ~
city+bd+gender+registered_via+payment_method_id+cancel_number+total_days+num_25_mean+num_50_mean+num_75_mean+num_985_mean+num_100_mean+num_unq_mean+total_secs_mean+num_25_var+num_50_var+num_75_var+num_985_var+num_unq_var+total_secs_var,data=train,method='class',minbucket=15)

rpart.plot(tree_1)

summary(tree_1)


#First Try Performance - Training Data Set

pred_train_1 = predict(tree_1,type='class')

ct_train_1 = table(train$is_churn,pred_train_1)

ct_train_1

accuracy_train_1 = (ct_train_1[1,1]+ct_train_1[2,2])/nrow(train)

accuracy_train_1
```

```
#First Try Performance - Testing Data Set

pred_test_1 = predict(tree_1,newdata=test,type='class')

ct_test_1 = table(test$is_churn,pred_test_1)

ct_test_1

accuracy_test_1 = (ct_test_1[1,1]+ct_test_1[2,2])/nrow(test)

accuracy_test_1


#Deal With Imbalanced Data

train_DT=rbind(train,train[sample(which(train$is_churn=='1'),size=2500,replace=TRUE),])


#Build Decision Tree - Second Try

tree_2 = rpart(is_churn ~ city+bd+gender+autorenew_number+registered_via+payment_method_id+cancel_number+total_days+num_25_mean+num_50_mean+num_75_mean+num_985_mean+num_100_mean+num_unq_mean+total_secs_mean+num_25_var+num_50_var+num_75_var+num_985_var+num_unq_var+total_secs_var,data=train_DT,method='class', minbucket=15)

rpart.plot(tree_2)

summary(tree_2)


#Second Try Performance - Training Data Set

pred_train_2 = predict(tree_2,type='class')

ct_train_2 = table(train_DT$is_churn,pred_train_2)
```

```
ct_train_2

accuracy_train_2 = (ct_train_2[1,1]+ct_train_2[2,2])/nrow(train_DT)

accuracy_train_2


#Second Try Performance - Testing Data Set

pred_test_2 = predict(tree_2,newdata=test,type='class')

ct_test_2 = table(test$is_churn,pred_test_2)

ct_test_2

accuracy_test_2 = (ct_test_2[1,1]+ct_test_2[2,2])/nrow(test)

accuracy_test_2


#Plot ROC Curve

library(ROCR)

pred = predict(tree_2,newdata=test)

ROCRpred = prediction(pred[,2],test$is_churn)

as.numeric(performance(ROCRpred,"auc")@y.values)

ROCRperf = performance(ROCRpred,"tpr","fpr")

plot(ROCRperf,colorize=TRUE,print.cutoffs.at=seq(0,1,0.5),text.adj=c(-0.3,2))


#Cross-Validation

library(caret)

trControl = trainControl(method = 'cv',number = 5)
```

```
tuneGrid = expand.grid(.cp=seq(0,0.1,0.001))

trainCV =
train(factor(is_churn)~.,train,method='rpart',trControl=trControl,tuneGrid=tuneGrid)

plot(trainCV)

trainCV$bestTune


#Deal With Imbalanced Data

train_RF=rbind(train[which(train$is_churn=='1'),],train[sample(which(train$is_churn
=='0'),size=250),])


#Build Random Forest

library(randomForest)

randommodel <- randomForest(is_churn ~ ., data=train_RF,importance = TRUE,
proximity = FALSE, ntree = 200)

print(randommodel)


#Determine Number of Trees

plot(randommodel)


#Build New Random Forest

randomeforest<- randomForest(is_churn ~ ., data=train_RF,importance = TRUE,
proximity = FALSE, ntree = 100)

print(randommodel)
```

```
#Test Random Forest

pred_rf <- predict(randomforest, test)

caret::confusionMatrix(pred_rf, test$is_churn)


#Feature Selection

varImpPlot(randomforest, sort=T, n.var = 10, main = 'Top 10 Feature Importance')


#Deal With Imbalanced Data

train_LR=rbind(train,train[sample(which(train$is_churn=='1'),size=2500,replace=TR
UE),])


#Build Logistic Regression

logistic <- glm(is_churn ~
city+bd+gender+registered_via+payment_method_id+cancel_number+autorenew_nu
mber+total_days+num_25_mean+num_50_mean+num_985_mean+num_100_mean+
num_25_var+num_50_var+num_75_var+num_985_var+num_100_var+num_unq_va
r+total_secs_var, family=binomial(logit), data=train_LR)

summary(logistic)


#Goodness of Fit

log.null <- glm(is_churn ~ 1, family=binomial(logit), data=train_LR)

1-logLik(logistic)/logLik(log.null)


#Performance - Training Data Set
```

```
lr_pred_train=data.frame(is_churn=train_LR$is_churn,predvalue=predict(logistic,type="response"))

lr_pred_train$predclass=ifelse(lr_pred_train$predvalue > 0.5, "1","0")
confusionMatrix(data =lr_pred_train$predclass,reference =
lr_pred_train$is_churn,positive = "1")


#Plot ROC Curve

library(pROC)

roc <- roc(response = lr_pred_train$is_churn,predictor = lr_pred_train$predvalue)

plot(roc, legacy.axes = TRUE)


#Performance - Testing Data Set

lr_pred_test=data.frame(is_churn=test$is_churn,predvalue=predict(logistic,newdata=test,type="response"))

lr_pred_test$predclass=ifelse(lr_pred_test$predvalue > 0.5, "1","0")

confusionMatrix(data =lr_pred_test$predclass,reference =
lr_pred_test$is_churn,positive = "1")
```