Machine Learning Models

1. Load Libraries.

```
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
library(ROCR)
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:randomForest':
##
##     margin
```

```
library(rpart)
```

2. Load test data.

```
setwd("c:/Ryerson University/Semester 4/ProjectCode")
loc<-getwd()
traindata <- read.csv(file="traindata.csv",header=TRUE,sep=",")
testdata1 <- read.csv(file="testdata1.csv",header=TRUE,sep=",")
dim(testdata1)
```

```
## [1] 7575   15
```

```
#####################################################################
#                        LOGISTIC REGRESION
# Regression coefficients represent the mean change in the response variable for one unit of change # i
#####################################################################
m1 <- glm(income ~ age+ workclass+ education+marital.status+ occupation+ sex +hours.per.week, data = tra
summary(m1)
```

```
##
## Call:
## glm(formula = income ~ age + workclass + education + marital.status +
##     occupation + sex + hours.per.week, family = binomial("logit"),
##     data = traindata)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -2.7534  -0.5679  -0.2490  -0.0002   3.5623
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -6.550600   0.234910 -27.886  < 2e-16
## age                              0.028992   0.001836  15.792  < 2e-16
## workclassOther                   0.709291   0.115973   6.116 9.60e-10
## workclassPrivate                 0.313209   0.067394   4.647 3.36e-06
## workclassSelf-Employed           0.083708   0.083127   1.007 0.313943
## workclassWithout-pay           -13.242756 360.993042  -0.037 0.970737
## education11th                   -0.045745   0.241259  -0.190 0.849616
## education12th                    0.470909   0.292592   1.609 0.107521
## education1st-4th                -0.867619   0.552323  -1.571 0.116216
## education5th-6th                -0.415032   0.376937  -1.101 0.270869
## education7th-8th                -0.777786   0.280019  -2.778 0.005476
## education9th                    -0.764471   0.337289  -2.267 0.023420
## educationAssoc-acdm              1.334751   0.196222   6.802 1.03e-11
## educationAssoc-voc               1.327517   0.188998   7.024 2.16e-12
## educationBachelors               1.994109   0.175304  11.375  < 2e-16
## educationDoctorate               2.948809   0.239511  12.312  < 2e-16
## educationHS-grad                 0.809830   0.170801   4.741 2.12e-06
## educationMasters                 2.442595   0.187137  13.052  < 2e-16
## educationPreschool             -12.278239 209.059559  -0.059 0.953167
## educationProf-school             3.105435   0.223687  13.883  < 2e-16
## educationSome-college            1.127201   0.173255   6.506 7.72e-11
## marital.statusMarried-AF-spouse  3.332010   0.615703   5.412 6.24e-08
## marital.statusMarried-civ-spouse 2.102102   0.072379  29.043  < 2e-16
## marital.statusMarried-spouse-absent -0.019878   0.250544  -0.079 0.936762
## marital.statusNever-married     -0.410540   0.088172  -4.656 3.22e-06
## marital.statusSeparated          0.001211   0.167921   0.007 0.994244
## marital.statusWidowed            0.068535   0.164132   0.418 0.676270
## occupationArmed-Forces          -0.900398   1.336716  -0.674 0.500572
## occupationCraft-repair          -0.074268   0.085948  -0.864 0.387529
## occupationExec-managerial        0.795971   0.081752   9.736  < 2e-16
## occupationFarming-fishing       -1.044920   0.149418  -6.993 2.69e-12
## occupationHandlers-cleaners     -0.931525   0.163897  -5.684 1.32e-08
## occupationMachine-op-inspct     -0.421011   0.111677  -3.770 0.000163
## occupationOther-service         -1.087462   0.133580  -8.141 3.92e-16
## occupationPriv-house-serv      -12.533952 127.709038  -0.098 0.921817
## occupationProf-specialty         0.474709   0.086503   5.488 4.07e-08
## occupationProtective-serv        0.502252   0.137626   3.649 0.000263
## occupationSales                  0.250933   0.087195   2.878 0.004004
## occupationTech-support           0.573350   0.119574   4.795 1.63e-06
## occupationTransport-moving      -0.196963   0.108205  -1.820 0.068716
## sexMale                          0.191701   0.057653   3.325 0.000884
## hours.per.week                   0.028375   0.001824  15.553  < 2e-16
##
## (Intercept)                     ***
## age                             ***
## workclassOther                  ***
## workclassPrivate                ***
## workclassSelf-Employed
## workclassWithout-pay
```

```
## education11th
## education12th
## education1st-4th
## education5th-6th
## education7th-8th                    **
## education9th                        *
## educationAssoc-acdm                 ***
## educationAssoc-voc                  ***
## educationBachelors                  ***
## educationDoctorate                  ***
## educationHS-grad                    ***
## educationMasters                    ***
## educationPreschool
## educationProf-school                ***
## educationSome-college               ***
## marital.statusMarried-AF-spouse     ***
## marital.statusMarried-civ-spouse    ***
## marital.statusMarried-spouse-absent
## marital.statusNever-married         ***
## marital.statusSeparated
## marital.statusWidowed
## occupationArmed-Forces
## occupationCraft-repair
## occupationExec-managerial           ***
## occupationFarming-fishing           ***
## occupationHandlers-cleaners         ***
## occupationMachine-op-inspct         ***
## occupationOther-service             ***
## occupationPriv-house-serv
## occupationProf-specialty            ***
## occupationProtective-serv           ***
## occupationSales                     **
## occupationTech-support              ***
## occupationTransport-moving          .
## sexMale                             ***
## hours.per.week                      ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25373  on 22586  degrees of freedom
## Residual deviance: 16433  on 22545  degrees of freedom
## AIC: 16517
##
## Number of Fisher Scoring iterations: 14
```

```
predictiontrain <- predict(m1,traindata,type='response')
pred1 <- rep('<=50K', length(predictiontrain))
pred1[predictiontrain>=.5] <- '>50K'
tb1 <- table(pred1, traindata$income)
tb1
```

```
##
## pred1   <=50K  >50K
```

```
##   <=50K 15596  2537
##   >50K   1358  3096
```

```
prob <- predict(m1, testdata1, type = 'response')
prediction <- predict(m1,testdata1,type='response')

#############################################################################
# P values shows that Age ,workclass, education, marital status, occupation,
# race, sex, hours per week  are the significant attributes.
#############################################################################
pred <- rep('<=50K', length(prob))
pred[prob>=.5] <- '>50K'
tb <- table(pred, testdata1$income)
tb
```

```
##
## pred    <=50K >50K
##   <=50K  5247  846
##   >50K    453 1029
```

```
# Confusion matrix shows that it has an Accuracy of 83.01%
# misclasification 17%.
```

DECISION TREE

```
Dtree<- rpart(income~ age+ workclass+ education+marital.status+ occupation+ sex +hours.per.week, data =
Dtree.Ptrain <- predict(Dtree,newdata= traindata, type = 'class')
confusionMatrix(traindata$income,Dtree.Ptrain)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50K  >50K
##      <=50K 15658  1296
##      >50K   2370  3263
##
##                Accuracy : 0.8377
##                  95% CI : (0.8328, 0.8425)
##     No Information Rate : 0.7982
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.537
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8685
##             Specificity : 0.7157
##          Pos Pred Value : 0.9236
##          Neg Pred Value : 0.5793
##              Prevalence : 0.7982
##          Detection Rate : 0.6932
##    Detection Prevalence : 0.7506
##       Balanced Accuracy : 0.7921
##
```

```
##        'Positive' Class : <=50K
##
```

```
Dtree.pred.prob <- predict(Dtree, newdata = testdata1, type = 'prob')
Dtree.pred <- predict(Dtree, newdata = testdata1, type = 'class')
confusionMatrix(testdata1$income,Dtree.pred)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50K >50K
##      <=50K  5263  437
##      >50K    833 1042
##
##                Accuracy : 0.8323
##                  95% CI : (0.8237, 0.8407)
##     No Information Rate : 0.8048
##     P-Value [Acc > NIR] : 3.734e-10
##
##                   Kappa : 0.5156
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8634
##             Specificity : 0.7045
##          Pos Pred Value : 0.9233
##          Neg Pred Value : 0.5557
##              Prevalence : 0.8048
##          Detection Rate : 0.6948
##    Detection Prevalence : 0.7525
##       Balanced Accuracy : 0.7839
##
##        'Positive' Class : <=50K
##
```

RANDOM FOREST

```
library(randomForest)
levels(testdata1$workclass) <- levels(traindata$workclass)
rforest <- randomForest(income ~ age+ workclass+ education+marital.status+occupation+ sex+hours.per.week
rforest.pred.prob <- predict(rforest, newdata = testdata1, type = 'prob')
rforest.pred <- predict(rforest, newdata = testdata1, type = 'class')
# confusion matrix
tb3 <- table(rforest.pred, testdata1$income)
tb3
```

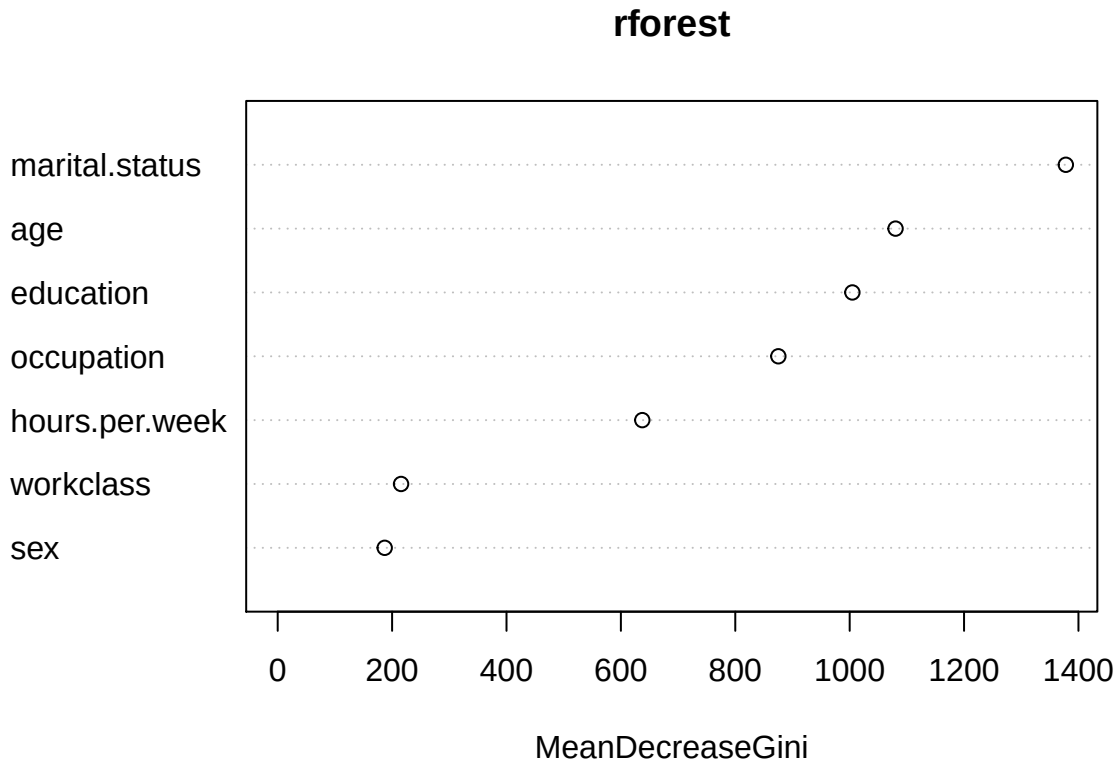
```
##
## rforest.pred <=50K >50K
##        <=50K  5213  753
##        >50K    487 1122
```

```
confusionMatrix(testdata1$income,rforest.pred)
```

```
## Confusion Matrix and Statistics
##
```

```
##           Reference
## Prediction <=50K >50K
##      <=50K  5213  487
##      >50K    753 1122
##
##                Accuracy : 0.8363
##                  95% CI : (0.8278, 0.8446)
##     No Information Rate : 0.7876
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5386
##  Mcnemar's Test P-Value : 5.252e-14
##
##             Sensitivity : 0.8738
##             Specificity : 0.6973
##          Pos Pred Value : 0.9146
##          Neg Pred Value : 0.5984
##              Prevalence : 0.7876
##          Detection Rate : 0.6882
##    Detection Prevalence : 0.7525
##       Balanced Accuracy : 0.7856
##
##        'Positive' Class : <=50K
##
```

```
varImpPlot (rforest)
```



```
## LINEAR REGRESION
pr  <- prediction(prob,testdata1$income)
perf <- performance(pr,measure="tpr", x.measure="fpr")
DtFrameReg <- data.frame(FP=perf@x.values[[1]],TP=perf@y.values[[1]])
aucRegresion <- performance(pr,measure='auc')@y.values[[1]]
aucRegresion
```

```
## [1] 0.8814819
```

```
###DECISION TREE
prtree <- prediction(Dtree.pred.prob[,2],testdata1$income)
perftree  <- performance(prtree,measure="tpr",x.measure="fpr")
DTFrametree <- data.frame(FP=perftree@x.values[[1]],TP=perftree@y.values[[1]])
auctree <- performance(prtree, measure='auc')@y.values[[1]]
auctree
```

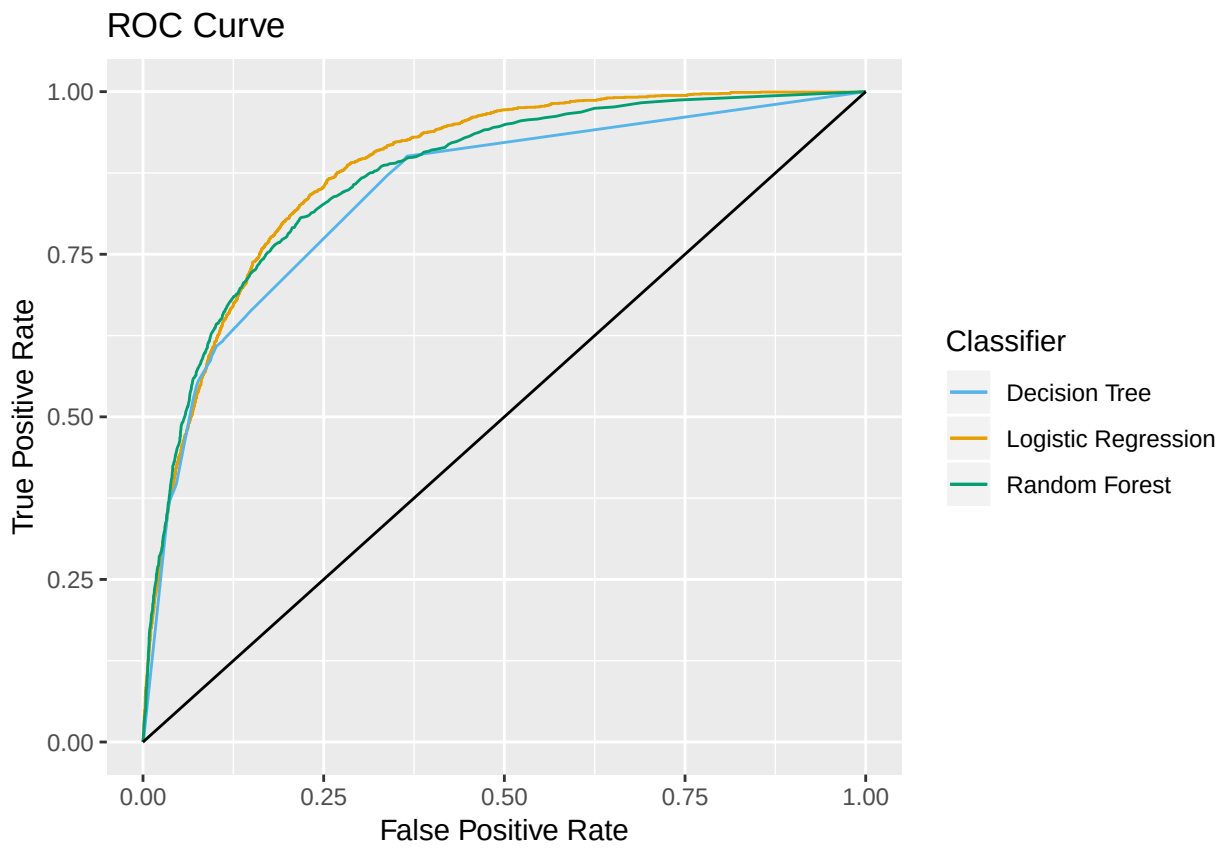
```
## [1] 0.8420762
```

```
###RANDOM FOREST
prRForest <- prediction(rforest.pred.prob[,2],testdata1$income)
perfRForest  <- performance(prRForest,measure="tpr",x.measure="fpr")
DTFrameRForest <- data.frame(FP=perfRForest@x.values[[1]],TP=perfRForest@y.values[[1]])
aucFtree <- performance(prRForest, measure='auc')@y.values[[1]]
aucFtree
```

```
## [1] 0.8699271
```

```
# plot ROC curve for logistic regression
g <- ggplot() +
  geom_line(data = DtFrameReg, aes(x = FP, y = TP, color = 'Logistic Regression')) +
  geom_line(data = DTFrametree, aes(x = FP, y = TP, color = 'Decision Tree')) +
  geom_line(data = DTFrameRForest, aes(x = FP, y = TP, color = 'Random Forest')) +
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1)) +
  ggtitle('ROC Curve') +
  labs(x = 'False Positive Rate', y = 'True Positive Rate')


g +  scale_colour_manual(name = 'Classifier', values = c('Logistic Regression'='#E69F00',
                                                         'Decision Tree'='#56B4E9', 'Random Forest'='#009E73'))
```




```
auc <- rbind(aucRegresion,auctree,aucFtree)
rownames(auc) <- (c('Logistic Regression', 'Decision Tree', 'Random Forest'))
colnames(auc) <- 'Area Under ROC Curve'
round(auc, 4)
```

```
##                     Area Under ROC Curve
## Logistic Regression               0.8815
## Decision Tree                     0.8421
## Random Forest                     0.8699
```