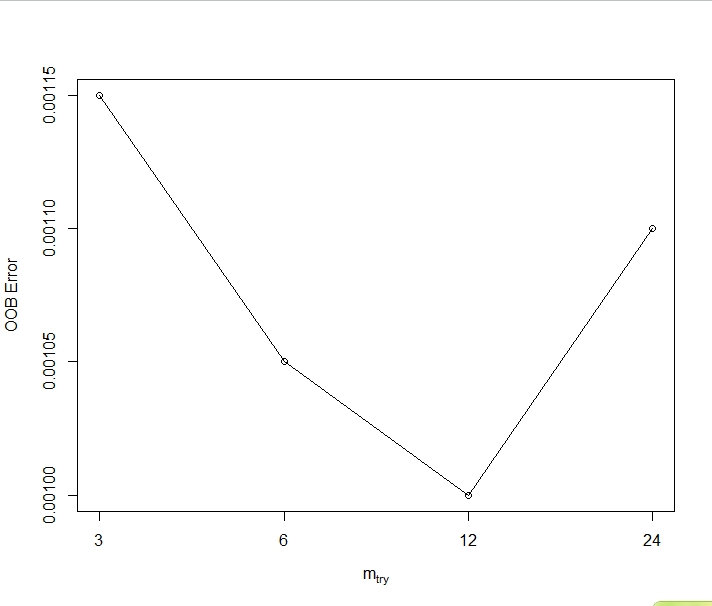
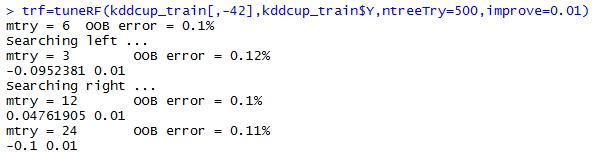
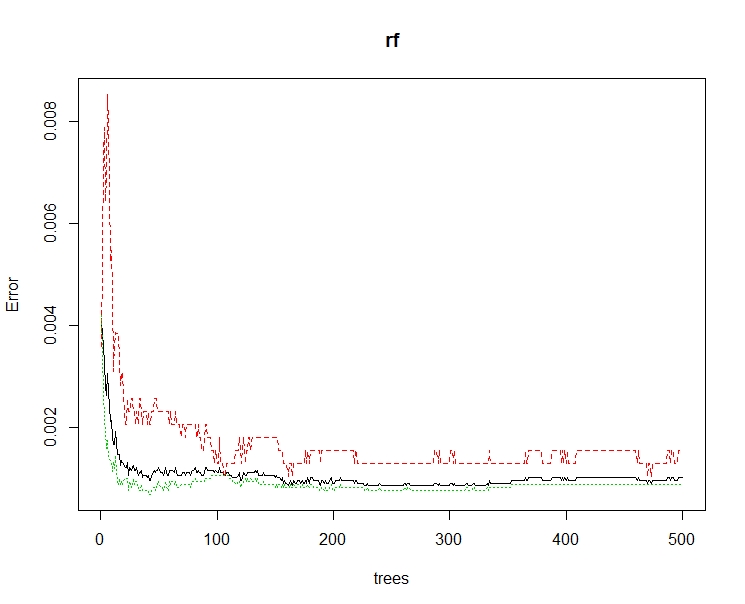
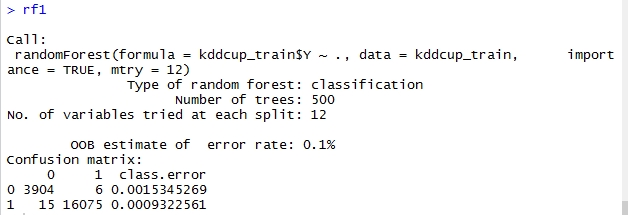
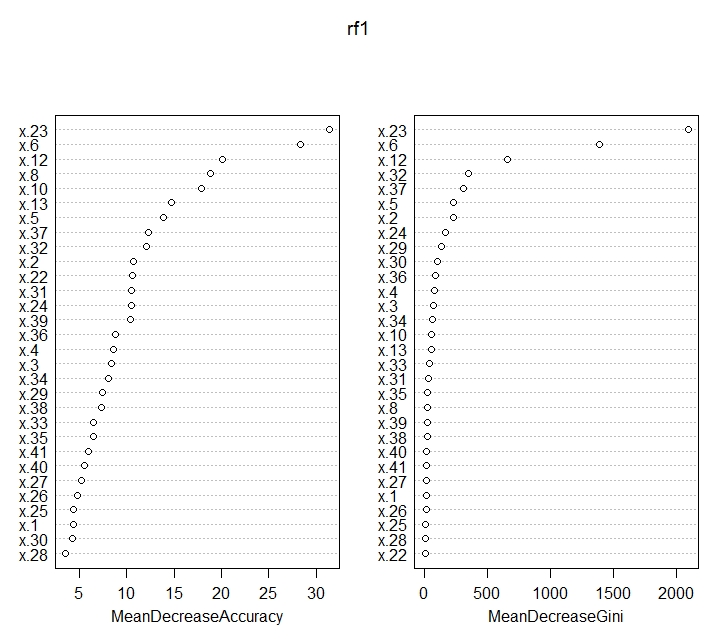
RandomForest classification of Kddcup data











> run\_time

user system elapsed

0.20 0.12 2.39

> error\_rate

[1] 0.0026

## Code Appendix

# random forest classification of the kddcup data

library("randomForest", lib.loc="~/R/win-library/3.2")

kddcup <- read.csv("C:/Users/Christina/Desktop/kddcup.data\_10\_percent\_corrected", header=FALSE)

kddcup=as.matrix(kddcup)

kddcup\_1=sample((nrow(kddcup)),size=20000,replace = FALSE, prob = NULL)

kddcup\_2=sample((nrow(kddcup)),size=5000,replace = FALSE, prob = NULL)

#Process of the data

kddcup\_train=kddcup[kddcup\_1,]

kddcup\_test=kddcup[kddcup\_2,]

for (i in 1:5000){

if (kddcup\_test[i,42]=="normal."){

kddcup\_test[i,42]=0}

else{

kddcup\_test[i,42]=1

}

}

for (i in 1:20000){

if (kddcup\_train[i,42]=="normal."){

kddcup\_train[i,42]=0}

else{

kddcup\_train[i,42]=1

}

}

kddcup\_train=as.data.frame(kddcup\_train)

kddcup\_test=as.data.frame(kddcup\_test)

for (i in 1:41){

kddcup\_train[,i]=as.numeric(kddcup\_train[,i])

kddcup\_test[,i]=as.numeric(kddcup\_test[,i])

}

kddcup\_test[,42]=as.factor(kddcup\_test[,42])

kddcup\_train[,42]=as.factor(kddcup\_train[,42])

colnames(kddcup\_train)=c(paste("x.",1:41,sep=""),"Y")

colnames(kddcup\_test)=c(paste("x.",1:41,sep=""),"Y")

#build the model

rf=randomForest(kddcup\_train$Y~.,kddcup\_train,importance=TRUE)

plot(rf,type="p",cex=0.5)

print(rf)

imp=importance(rf)

varImpPlot(rf)

#find the optimal mtry paraeter

trf=tuneRF(kddcup\_train[,-42],kddcup\_train$Y,ntreeTry=500,improve=0.01)

#we find that the optimal mtry is 12

rf1=randomForest(kddcup\_train$Y~.,kddcup\_train,importance=TRUE,mtry=12)

rf1

varImpPlot(rf1)

start\_time=proc.time()

pre\_test=predict(rf1,kddcup\_test[,-42])

end\_time=proc.time()

run\_time=end\_time-start\_time

error\_rate=sum(pre\_test!=kddcup\_test$Y)/nrow(kddcup\_test)