Mini Project - 04

# Using bank-full.csv dataset in R

bankfull = read.csv(file.choose()) # Import data   
# HYPOTHESIS TESTING  
t.test(bankfull$age~bankfull$y) # t-test  
summary(aov(bankfull$age~bankfull$job)) # anova test  
chisq.test(table(bankfull$y, bankfull$job)) # chi-square test  
  
bankfull1=bankfull[-c(10,11)]  
  
#binary logistic regression  
ylogit=glm(y~.,data = bankfull1,family = "binomial")  
summary(ylogit)  
ylogitpredict=predict(ylogit,type="response") #predicting  
table(Actual=bankfull1$y,Predict=ylogitpredict>0.5)  
(39004+1705)/(39004+1705+918+3584) #accuracy=90.04225%  
  
#decision tree #package require"rpart"  
bankfullpart=rpart(y~.,data=bankfull1)  
summary(bankfullpart)  
rpart.plot(bankfullpart)  
text(bankfullpart,cex=.50)  
rpart.plot(bankfullpart)  
bankfullpartpredict=predict(bankfullpart,type = "class")  
table(Actual=bankfull1$y,predict=bankfullpartpredict)  
(38904+1845)/(38904+1845+1081+3444) #accuracy=90.0005%  
  
#random forest(3000) #package="random Forest"  
brandomforest=randomForest(y~.,data=bankfull1,ntree=800,do.trace=100)  
print(brandomforest)  
plot(brandomforest)  
  
#gradient boosting machine #package"gbm"  
bankfull2=bankfull1  
bankfull2$y=ifelse(bankfull2$y=="yes",1,0)  
ygbm=gbm(y~.,data=bankfull2,distribution="bernoulli",n.trees=10,cv.folds=3)  
bestiter=gbm.perf(ygbm,method="cv")  
gbmpredict=predict(ygbm,bankfull2,bestiter)  
table(bankfull2$y,gbmpredict>0.5)  
  
#neural networks-package  
ynnet=nnet(y~.,data=bankfull1,size=10,maxit=3)  
summary(ynnet)  
ynnetpredict=predict(ynnet,type="class")  
table(bankfull1$y,ynnetpredict)