Logistic regression code

setwd("D:\\DS 1\\logistic regression")

data=read.csv("Loan.csv")

summary(data)

sum(is.na(data))

names(data)

attach(data)

# split the data

library(caTools)

split= sample.split(data,SplitRatio = 0.7)

train= subset(data, split=="TRUE")

test= subset(data, split=="FALSE")

# linear model foe checking multicolinearity

mod=lm(PersonalLoan~.-Age,train)

vif(mod)

# logistic model

mod1=glm(PersonalLoan ~ Experience + Income + Family + CCAvg + Education + SecuritiesAccount + CDAccount + Online + CreditCard, family = "binomial", data = train)

summary(mod1)

#prediction

predict= predict(mod1, data=train, type="response")

predict

# necessary libraries

library(lattice)

library(ggplot2)

library(caret)

library(e1071)

# confusion matrix

prediction= ifelse(predict>=0.5,1,0)

confusionMatrix(as.factor(prediction), as.factor(train$PersonalLoan),positive = "1")

##Mcfadden test

library(pscl)

pR2(mod1)

# Concordance Test #

library(InformationValue)

library(caret)

concor <- Concordance(train$PersonalLoan,predict)

concor

# plot ROC curve

library(ROCR)

rocpred= prediction(prediction,train$PersonalLoan)

rocper= performance(rocpred,"tpr","fpr")

plot(rocper,colorize=TRUE, print.cutoffs.at=seq(0.1,by=0.1))

# another way

plotROC(actuals = train$PersonalLoan, predictedScores = as.numeric(fitted(mod1)))

ks\_plot(actuals = train$PersonalLoan, predictedScores = as.numeric(fitted(mod1)))

ks\_stat(actuals = train$PersonalLoan, predictedScores = as.numeric(fitted(mod1)))