getwd()

data = read.csv("cereals.csv", header=T)

samplesize = 0.60 \* nrow(data)

set.seed(80)

index = sample( seq\_len ( nrow ( data ) ), size = samplesize )

datatrain = data[ index, ]

datatest = data[ -index, ]

max = apply(data , 2 , max)

min = apply(data, 2 , min)

scaled = as.data.frame(scale(data, center = min, scale = max - min))

install.packages("neuralnet")

library(neuralnet)

trainNN = scaled[index , ]

testNN = scaled[-index , ]

set.seed(2)

NN = neuralnet(rating ~ calories + protein + fat + sodium + fiber, trainNN, hidden = 3 , linear.output = T )

plot(NN)

predict\_testNN = compute(NN, testNN[,c(1:5)])

predict\_testNN = (predict\_testNN$net.result \* (max(data$rating) - min(data$rating))) + min(data$rating)

plot(datatest$rating, predict\_testNN, col='blue', pch=16, ylab = "predicted rating NN", xlab = "real rating")

abline(0,1)

RMSE.NN = (sum((datatest$rating - predict\_testNN)^2) / nrow(datatest)) ^ 0.5

install.packages("boot")

install.packages("plyr")

library(boot)

library(plyr)

set.seed(50)

k = 100

RMSE.NN = NULL

List = list( )

for(j in 10:65){

for (i in 1:k) {

index = sample(1:nrow(data),j )

trainNN = scaled[index,]

testNN = scaled[-index,]

datatest = data[-index,]

NN = neuralnet(rating ~ calories + protein + fat + sodium + fiber, trainNN, hidden = 3, linear.output= T)

predict\_testNN = compute(NN,testNN[,c(1:5)])

predict\_testNN = (predict\_testNN$net.result\*(max(data$rating)-min(data$rating)))+min(data$rating)

RMSE.NN [i]<- (sum((datatest$rating - predict\_testNN)^2)/nrow(datatest))^0.5

}

List[[j]] = RMSE.NN

}

Matrix.RMSE = do.call(cbind, List)

boxplot(Matrix.RMSE[,56], ylab = "RMSE", main = "RMSE BoxPlot (length of traning set = 65)")

install.packages("matrixStats")

library(matrixStats)

med = colMedians(Matrix.RMSE)

X = seq(10,65)

plot (med~X, type = "l", xlab = "length of training set", ylab = "median RMSE", main = "Variation of RMSE with length of training set")



