# Artificial Neural Network

#this is classifier problem because our result will be in yes and no

# Importing the dataset

dataset = read.csv('Churn\_Modelling.csv')

dataset = dataset[4:14]

# Encoding the categorical variables as factors

dataset$Geography = as.numeric(factor(dataset$Geography,

levels = c('France', 'Spain', 'Germany'),

labels = c(1, 2, 3)))

#as.numeric function to set function as numeric

dataset$Gender = as.numeric(factor(dataset$Gender,

levels = c('Female', 'Male'),

labels = c(1, 2)))

# Splitting the dataset into the Training set and Test set

# install.packages('caTools')

library(caTools)

set.seed(123)

split = sample.split(dataset$Exited, SplitRatio = 0.8)

training\_set = subset(dataset, split == TRUE)

test\_set = subset(dataset, split == FALSE)

# Feature Scaling

#there is lot of computation in parallel so need feature scaling and required by package

training\_set[-11] = scale(training\_set[-11])

test\_set[-11] = scale(test\_set[-11])

# Fitting ANN to the Training set

#install.packages('h2o')

#h2o is open source software allow run faster

library(h2o)

h2o.init(nthreads = -1)#allow connect specific server, nthreads -1 take all available code use in your system

model = h2o.deeplearning(y = 'Exited',

training\_frame = as.h2o(training\_set),

activation = 'Rectifier',

hidden = c(6,6),

epochs = 100,

train\_samples\_per\_iteration = -2)

#0,-1,-2...by putting -2 ,this will auto choose ANN

#as.h2o convert training set into training frame

#c(5,5) first parameter is no of hidden layer and second parameter is no of node in each hidden layer

# convenient way to choose the node is take average of (no od independent and dependent variable) i.e 6

#so 11 node in input layer and 1 node in output layer so average((11+1)/2=6) is 6

# Predicting the Test set results

y\_pred = h2o.predict(model, newdata = as.h2o(test\_set[-11]))

y\_pred = (y\_pred > 0.5)# but we neeed yes or no

y\_pred = as.vector(y\_pred)

y\_pred

# Making the Confusion Matrix

cm = table(test\_set[, 11], y\_pred)

cm

#accuracy--(1520+188)/2000 so 85.4% accuracy

#h2o.shutdown() # TO SHUT DOWN h20 its necessary

#Y