# Artificial Neural Network

# Installing Theano

# pip install --upgrade --no-deps git+git://github.com/Theano/Theano.git

# Installing Tensorflow

# Install Tensorflow from the website: https://www.tensorflow.org/versions/r0.12/get\_started/os\_setup.html

# Installing Keras

# pip install --upgrade keras

# Part 1 - Data Preprocessing

# Importing the libraries

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

# Importing the dataset

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

X = dataset.iloc[:, 3:13].values

y = dataset.iloc[:, 13].values

# Encoding categorical data

from sklearn.preprocessing import LabelEncoder, OneHotEncoder

labelencoder\_X\_1 = LabelEncoder()

X[:, 1] = labelencoder\_X\_1.fit\_transform(X[:, 1])

labelencoder\_X\_2 = LabelEncoder()

X[:, 2] = labelencoder\_X\_2.fit\_transform(X[:, 2])

onehotencoder = OneHotEncoder(categorical\_features = [1])

X = onehotencoder.fit\_transform(X).toarray()

X = X[:, 1:]

# Splitting the dataset into the Training set and Test set

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, random\_state = 0)

# Feature Scaling

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X\_train = sc.fit\_transform(X\_train)

X\_test = sc.transform(X\_test)

# Part 2 - Now let's make the ANN!

# Importing the Keras libraries and packages

import keras

from keras.models import Sequential

from keras.layers import Dense

#--------------------------------------

#this neural network going to be a classifier so first build classifier

#we will select step function for hidden layer and sigmoid function for output layer

# Initialising the ANN

classifier = Sequential()

# Adding the input layer and the first hidden layer

classifier.add(Dense(output\_dim = 6, init = 'uniform', activation = 'relu', input\_dim = 11))#relu is parameter corresponding rectifier function, 11 is no of input in hidden layer i.e no of independent variable

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

# 6 is no of node we want to add in hidden layer which specify no of input in hidden layer

#to get optimal parameter in model, we generally do parameter tuning but in above problem we will take average no of node as input in model i.e 6.

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

# Adding the second hidden layer

classifier.add(Dense(output\_dim = 6, init = 'uniform', activation = 'relu'))#initlization uniform waight that come from first layer to all

# Adding the output layer

classifier.add(Dense(output\_dim = 1, init = 'uniform', activation = 'sigmoid'))#initlization uniform waight that come from second layer to all

# here we are using sigmoid function in output layer

# Compiling the ANN

classifier.compile(optimizer = 'adam', loss = 'binary\_crossentropy', metrics = ['accuracy'])

#optimizer is algorithum use to find out optimal no of weight

#sum(y-y`)^2 is the loss function in linear regression but in logistic function have different function...in case of one output we have binary\_crossentropy but more then one output we will use (caterorical\_crossvalidation) as loss function

# Fitting the ANN to the Training set

classifier.fit(X\_train, y\_train, batch\_size = 10, nb\_epoch = 100)#100 time train

#we need to do no of experiment to find out batch\_size and nb\_epoch

# Part 3 - Making the predictions and evaluating the model

# RESULT::so finally we rwach the accuracy of 83.5 % in created model

# Predicting the Test set results

y\_pred = classifier.predict(X\_test)# return the probability of customer leave the bank

# RESULT::so first customer have 23% chance to leave the bank

#FOR validation we will seek same accuracy in test data

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

#customer 5 leave the bank

#accuracy is (no of currect/total) i.e (1536+146)/2000 ~84%

# Making the Confusion Matrix

from sklearn.metrics import confusion\_matrix

cm = confusion\_matrix(y\_test, y\_pred)

#we will use data mining technique to find out why customer leave the bank