**Practical: 10**

**Aim: To Implement multiclass classification with neural network on iris flower dataset**

**Code:**

import pandas as pd

import numpy as np

import pandas as pd

import pandas as pd

import numpy as np

from keras.models import Sequential

from keras.layers import Dense

from keras.utils import np\_utils

from sklearn.preprocessing import LabelEncoder

from keras.utils.np\_utils import to\_categorical

from sklearn.utils import shuffle

data = pd.read\_csv('iris.csv')

data = data.drop(['Id'], axis =1)

data = shuffle(data)

i = 8

data\_to\_predict = data[:i].reset\_index(drop = True)

predict\_species = data\_to\_predict.Species

predict\_species = np.array(predict\_species)

prediction = np.array(data\_to\_predict.drop(['Species'],axis= 1))

data = data[i:].reset\_index(drop = True)

X = data.drop(['Species'], axis = 1)

X = np.array(X)

Y = data['Species']

# Transform name species into numerical values

encoder = LabelEncoder()

encoder.fit(Y)

Y = encoder.transform(Y)

Y = np\_utils.to\_categorical(Y)

train\_x, test\_x, train\_y, test\_y = model\_selection.train\_test\_split(X,Y,test\_size = 0.1, random\_state = 0)

input\_dim = len(data.columns) - 1

model = Sequential()

model.add(Dense(8, input\_dim = input\_dim , activation = 'relu'))

model.add(Dense(10, activation = 'relu'))

model.add(Dense(10, activation = 'relu'))

model.add(Dense(10, activation = 'relu'))

model.add(Dense(3, activation = 'softmax'))

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

model.fit(train\_x, train\_y, epochs = 10, batch\_size = 2)

scores = model.evaluate(test\_x, test\_y)

print("\n%s: %.2f%%" % (model.metrics\_names[1], scores[1]\*100))

predictions = model.predict\_classes(prediction)

prediction\_ = np.argmax(to\_categorical(predictions), axis = 1)

prediction\_ = encoder.inverse\_transform(prediction\_)

for i, j in zip(prediction\_ , predict\_species):

print( " Predicted Species: {}, Actual Species: {}".format(i,j))

**Output:**