from sklearn.preprocessing import LabelEncoder  
from sklearn.model\_selection import cross\_val\_score  
from sklearn.model\_selection import KFold  
from sklearn.pipeline import Pipeline  
from sklearn.neural\_network import MLPClassifier  
  
import pandas as pd  
from keras.models import Sequential  
from keras.layers import Dense  
from sklearn.neural\_network import MLPClassifier  
  
  
from tensorflow.keras.utils import to\_categorical  
from sklearn.preprocessing import LabelEncoder  
from sklearn.model\_selection import cross\_val\_score  
from sklearn.model\_selection import KFold  
from sklearn.pipeline import Pipeline  
from sklearn.neural\_network import MLPClassifier  
  
import pandas as pd  
from keras.models import Sequential  
from keras.layers import Dense  
from tensorflow.keras.utils import to\_categorical  
  
df = pd.read\_csv('/Users/hemantbodhare/PycharmProjects/pythonProject2/flowers.csv')  
print(df)  
x = df.iloc[:, 0:4].astype(float)  
y = df.iloc[:, 4]  
encoder = LabelEncoder()  
encoder.fit(y)  
encoder\_y = encoder.transform(y)  
print(encoder\_y)  
dummy = to\_categorical(encoder\_y)  
print(dummy)  
  
def baseline\_model():  
 model = Sequential()  
 model.add(Dense(8, input\_dim=4, activation='relu'))  
 model.add(Dense(3, activation='softmax'))  
 model.compile(loss='categorical\_crossentropy', optimizer='adam', metrics=['accuracy'])  
 return model  
  
estimator = baseline\_model()  
estimator.fit(x, dummy, epochs=100, shuffle=True)  
action = estimator.predict(x)  
for i in range(25):  
 print(dummy[i])  
print('^^^^^^^^^^^^^^^^^^')  
  
for i in range(25):  
 print(action[i])