```
from google.colab import files

uploaded = files.upload()

for fn in uploaded.keys():
  print('User uploaded file "{name}" with length {length} bytes'.format(
      name=fn, length=len(uploaded[fn])))
```

Choose Files datasets_18...8_data.csv

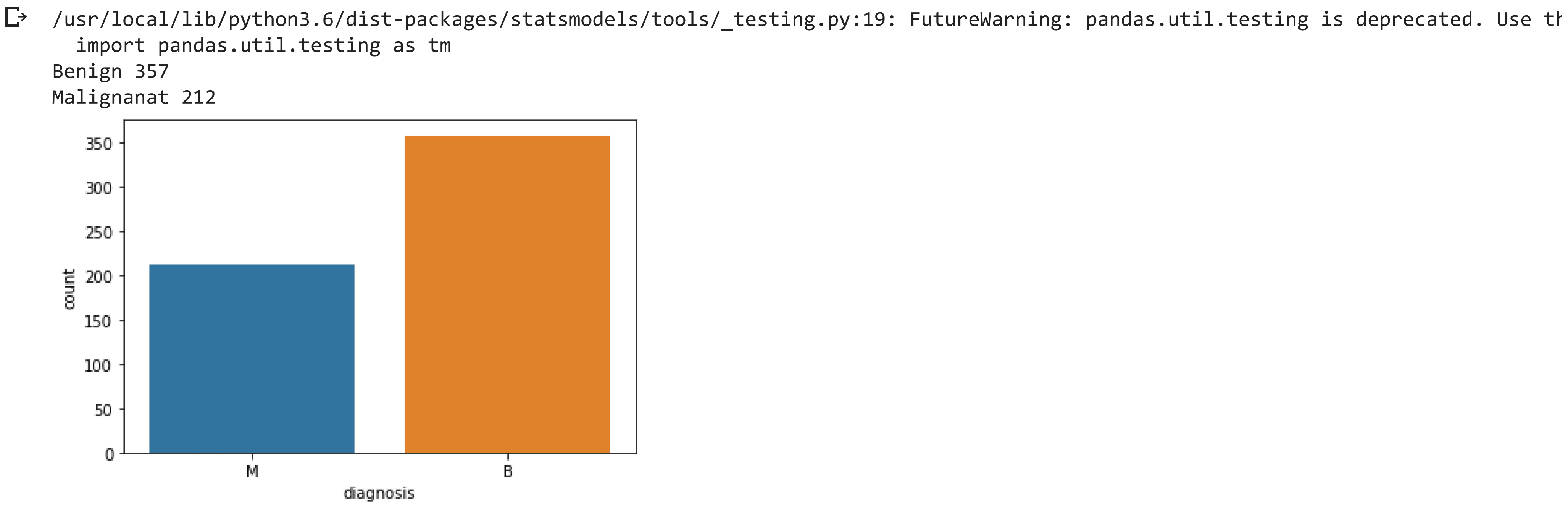
- **datasets_180_408_data.csv**(application/vnd.ms-excel) - 125204 bytes, last modified: 10/09/2020 - 100% done

```
Saving datasets_180_408_data.csv to datasets_180_408_data.csv
User uploaded file "datasets_180_408_data.csv" with length 125204 bytes
```

```
import pandas as pd
import io
data = pd.read_csv('datasets_180_408_data.csv')
data.head
```

```
<bound method NDFrame.head of            id diagnosis  ...  fractal_dimension_worst  Unnamed: 32
0      842302         M  ...                  0.11890          NaN
1      842517         M  ...                  0.08902          NaN
2    84300903         M  ...                  0.08758          NaN
3    84348301         M  ...                  0.17300          NaN
4    84358402         M  ...                  0.07678          NaN
..        ...       ...  ...                      ...          ...
564    926424         M  ...                  0.07115          NaN
565    926682         M  ...                  0.06637          NaN
566    926954         M  ...                  0.07820          NaN
567    927241         M  ...                  0.12400          NaN
568     92751         B  ...                  0.07039          NaN

[569 rows x 33 columns]>
```

```
import seaborn as sns
ax = sns.countplot(data['diagnosis'],label = 'count')
B,M = data['diagnosis'].value_counts()
print('Benign',B)
print('Malignanat',M)
```



```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testing is deprecated. Use th
  import pandas.util.testing as tm
Benign 357
Malignanat 212
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Deleting NaN column
del data['Unnamed: 32']


X = data.iloc[:,2:].values
y = data.iloc[:,1].values

# Encoding categorical data
from sklearn.preprocessing import LabelEncoder
labelencoder_X_1= LabelEncoder()
y = labelencoder_X_1.fit_transform(y)





#Splittting 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)
```

```
X_train
```

```
array([[-1.15036482, -0.39064196, -1.12855021, ..., -0.75798367,
        -0.01614761, -0.38503402],
       [-0.93798972,  0.68051405, -0.94820146, ..., -0.60687023,
         0.09669004, -0.38615797],
       [ 0.574121  , -1.03333557,  0.51394098, ..., -0.02371948,
        -0.20050207, -0.75144254],
       ...,
       [-1.32422924, -0.20048168, -1.31754581, ..., -0.97974953,
        -0.71542314, -0.11978123],
       [-1.24380987, -0.2245526 , -1.28007609, ..., -1.75401433,
        -1.58157125, -1.00601779],
       [-0.73694129,  1.14989702, -0.71226578, ..., -0.27460457,
        -1.25895095,  0.21515662]])
```

```
X_test
```

```
     array([[-0.20175604,  0.3290786 , -0.13086754, ...,  1.3893291 ,
              1.08203284,  1.54029664],
```

```
###Install Keras
```

```
            [-0.02619262, -0.8407682 , -0.09175081, ..., -0.49483785,
```

```
#!pip install keras
```

```
            [ 1.71811488,  0.09318356,  1.7286186 , ...,  1.57630515,
```

```
import keras
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout


#adding the input and first hidden layer

classifier = Sequential()
classifier.add(Dense(kernel_initializer = 'uniform', activation = 'relu',input_dim = 30,units = 16   ))  #output_dim(units) = input /o
classifier.add(Dropout(rate=0.9))

#adding the second hidden layer
classifier.add(Dense(kernel_initializer = 'uniform', activation = 'relu',units = 16   ))


#adding the output layer
classifier.add(Dense(kernel_initializer = 'uniform', activation = 'relu',units = 1   ))



classifier.compile(optimizer="Adam",loss="binary_crossentropy",metrics=['accuracy'])


classifier.fit(X_train,y_train,batch_size=100,epochs=150)
```

```
Epoch 1/150
5/5 [==============================] - 0s 2ms/step - loss: 2.5731 - accuracy: 0.6374
Epoch 2/150
5/5 [==============================] - 0s 1ms/step - loss: 1.7970 - accuracy: 0.6374
Epoch 3/150
5/5 [==============================] - 0s 1ms/step - loss: 1.5458 - accuracy: 0.6374
Epoch 4/150
5/5 [==============================] - 0s 1ms/step - loss: 1.4057 - accuracy: 0.6374
Epoch 5/150
5/5 [==============================] - 0s 1ms/step - loss: 1.2919 - accuracy: 0.6374
Epoch 6/150
5/5 [==============================] - 0s 1ms/step - loss: 1.2219 - accuracy: 0.6374
Epoch 7/150
5/5 [==============================] - 0s 2ms/step - loss: 1.1836 - accuracy: 0.6374
Epoch 8/150
5/5 [==============================] - 0s 2ms/step - loss: 1.0653 - accuracy: 0.6374
Epoch 9/150
5/5 [==============================] - 0s 2ms/step - loss: 0.9619 - accuracy: 0.6396
Epoch 10/150
5/5 [==============================] - 0s 2ms/step - loss: 0.9112 - accuracy: 0.6505
Epoch 11/150
5/5 [==============================] - 0s 1ms/step - loss: 0.8395 - accuracy: 0.6681
Epoch 12/150
5/5 [==============================] - 0s 2ms/step - loss: 0.7576 - accuracy: 0.6813
Epoch 13/150
5/5 [==============================] - 0s 2ms/step - loss: 0.7596 - accuracy: 0.6901
Epoch 14/150
5/5 [==============================] - 0s 2ms/step - loss: 0.6876 - accuracy: 0.7143
Epoch 15/150
5/5 [==============================] - 0s 1ms/step - loss: 0.6589 - accuracy: 0.7407
Epoch 16/150
5/5 [==============================] - 0s 2ms/step - loss: 0.5755 - accuracy: 0.7538
Epoch 17/150
5/5 [==============================] - 0s 2ms/step - loss: 0.5447 - accuracy: 0.7692
Epoch 18/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4956 - accuracy: 0.8066
Epoch 19/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4804 - accuracy: 0.7978
Epoch 20/150
5/5 [==============================] - 0s 1ms/step - loss: 0.4933 - accuracy: 0.8022
Epoch 21/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4281 - accuracy: 0.8242
Epoch 22/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4592 - accuracy: 0.8242
Epoch 23/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4442 - accuracy: 0.8374
```

```
5/5 [==============================] - 0s 2ms/step - loss: 0.4442 - accuracy: 0.8374
Epoch 24/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4301 - accuracy: 0.8418
Epoch 25/150
5/5 [==============================] - 0s 3ms/step - loss: 0.3558 - accuracy: 0.8725
Epoch 26/150
5/5 [==============================] - 0s 2ms/step - loss: 0.4113 - accuracy: 0.8484
Epoch 27/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3809 - accuracy: 0.8549
Epoch 28/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3744 - accuracy: 0.8637
Epoch 29/150
5/5 [==============================] - 0s 1ms/step - loss: 0.3685 - accuracy: 0.8593
Epoch 30/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3657 - accuracy: 0.8549
Epoch 31/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3171 - accuracy: 0.8791
Epoch 32/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3773 - accuracy: 0.8571
Epoch 33/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2989 - accuracy: 0.8923
Epoch 34/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3135 - accuracy: 0.8791
Epoch 35/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3154 - accuracy: 0.8901
Epoch 36/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3258 - accuracy: 0.8681
Epoch 37/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3352 - accuracy: 0.8747
Epoch 38/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3183 - accuracy: 0.8945
Epoch 39/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3248 - accuracy: 0.8857
Epoch 40/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2938 - accuracy: 0.8945
Epoch 41/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3520 - accuracy: 0.8769
Epoch 42/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3233 - accuracy: 0.8879
Epoch 43/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3635 - accuracy: 0.8769
Epoch 44/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3156 - accuracy: 0.8923
Epoch 45/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3548 - accuracy: 0.8879
Epoch 46/150
```

```
5/5 [==============================] - 0s 1ms/step - loss: 0.2961 - accuracy: 0.8879
Epoch 47/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2972 - accuracy: 0.9011
Epoch 48/150
5/5 [==============================] - 0s 1ms/step - loss: 0.3199 - accuracy: 0.8813
Epoch 49/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2822 - accuracy: 0.9099
Epoch 50/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3106 - accuracy: 0.8879
Epoch 51/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2491 - accuracy: 0.9187
Epoch 52/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3158 - accuracy: 0.9033
Epoch 53/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2972 - accuracy: 0.8923
Epoch 54/150
5/5 [==============================] - 0s 3ms/step - loss: 0.2799 - accuracy: 0.9033
Epoch 55/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2730 - accuracy: 0.9077
Epoch 56/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2561 - accuracy: 0.9143
Epoch 57/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2517 - accuracy: 0.9253
Epoch 58/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2386 - accuracy: 0.9253
Epoch 59/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2417 - accuracy: 0.9231
Epoch 60/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2717 - accuracy: 0.9099
Epoch 61/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2423 - accuracy: 0.9209
Epoch 62/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2852 - accuracy: 0.9077
Epoch 63/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2717 - accuracy: 0.9033
Epoch 64/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2289 - accuracy: 0.9275
Epoch 65/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2687 - accuracy: 0.9231
Epoch 66/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3193 - accuracy: 0.9011
Epoch 67/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3085 - accuracy: 0.9011
Epoch 68/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2700 - accuracy: 0.9099
Epoch 69/150
```

```
5/5 [==============================] - 0s 1ms/step - loss: 0.2944 - accuracy: 0.9121
Epoch 70/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2705 - accuracy: 0.9187
Epoch 71/150
5/5 [==============================] - 0s 3ms/step - loss: 0.3024 - accuracy: 0.9055
Epoch 72/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2764 - accuracy: 0.9231
Epoch 73/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2529 - accuracy: 0.9121
Epoch 74/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2963 - accuracy: 0.8989
Epoch 75/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2477 - accuracy: 0.9231
Epoch 76/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2980 - accuracy: 0.8967
Epoch 77/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2836 - accuracy: 0.8945
Epoch 78/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2654 - accuracy: 0.9099
Epoch 79/150
5/5 [==============================] - 0s 1ms/step - loss: 0.3132 - accuracy: 0.9033
Epoch 80/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2478 - accuracy: 0.9253
Epoch 81/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2756 - accuracy: 0.9231
Epoch 82/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3259 - accuracy: 0.8967
Epoch 83/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2584 - accuracy: 0.9187
Epoch 84/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2818 - accuracy: 0.9055
Epoch 85/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2566 - accuracy: 0.9297
Epoch 86/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2965 - accuracy: 0.9187
Epoch 87/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2543 - accuracy: 0.9187
Epoch 88/150
5/5 [==============================] - 0s 1ms/step - loss: 0.3332 - accuracy: 0.8901
Epoch 89/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2734 - accuracy: 0.9055
Epoch 90/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2517 - accuracy: 0.9231
Epoch 91/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3042 - accuracy: 0.9055
Epoch 92/150
Epoch 92/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2764 - accuracy: 0.9209
Epoch 93/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2659 - accuracy: 0.9077
Epoch 94/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2502 - accuracy: 0.9275
Epoch 95/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2494 - accuracy: 0.9165
Epoch 96/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3085 - accuracy: 0.9055
Epoch 97/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2429 - accuracy: 0.9275
Epoch 98/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2923 - accuracy: 0.9165
Epoch 99/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2857 - accuracy: 0.9011
Epoch 100/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3267 - accuracy: 0.9055
Epoch 101/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2571 - accuracy: 0.9187
Epoch 102/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2558 - accuracy: 0.9165
Epoch 103/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3198 - accuracy: 0.9011
Epoch 104/150
5/5 [==============================] - 0s 1ms/step - loss: 0.3021 - accuracy: 0.8945
Epoch 105/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2573 - accuracy: 0.9121
Epoch 106/150
5/5 [==============================] - 0s 3ms/step - loss: 0.2721 - accuracy: 0.9077
Epoch 107/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2293 - accuracy: 0.9253
Epoch 108/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2533 - accuracy: 0.9165
Epoch 109/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2423 - accuracy: 0.9275
Epoch 110/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2618 - accuracy: 0.9165
Epoch 111/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2642 - accuracy: 0.9165
Epoch 112/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3095 - accuracy: 0.9011
Epoch 113/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2803 - accuracy: 0.9209
Epoch 114/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2514 - accuracy: 0.9187
Epoch 115/150
Epoch 115/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2336 - accuracy: 0.9297
Epoch 116/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3129 - accuracy: 0.8989
Epoch 117/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2740 - accuracy: 0.9055
Epoch 118/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2826 - accuracy: 0.9033
Epoch 119/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2749 - accuracy: 0.9209
Epoch 120/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3139 - accuracy: 0.9099
Epoch 121/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2551 - accuracy: 0.9187
Epoch 122/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2282 - accuracy: 0.9275
Epoch 123/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2688 - accuracy: 0.9055
Epoch 124/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2443 - accuracy: 0.9297
Epoch 125/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2522 - accuracy: 0.9209
Epoch 126/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2415 - accuracy: 0.9297
Epoch 127/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2784 - accuracy: 0.9055
Epoch 128/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2407 - accuracy: 0.9275
Epoch 129/150
5/5 [==============================] - 0s 2ms/step - loss: 0.3102 - accuracy: 0.8989
Epoch 130/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2759 - accuracy: 0.9055
Epoch 131/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2478 - accuracy: 0.9231
Epoch 132/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2768 - accuracy: 0.9077
Epoch 133/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2586 - accuracy: 0.9165
Epoch 134/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2918 - accuracy: 0.9011
Epoch 135/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2799 - accuracy: 0.9055
Epoch 136/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2535 - accuracy: 0.9165
Epoch 137/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2931 - accuracy: 0.9011
```

```
Epoch 138/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2500 - accuracy: 0.9209
Epoch 139/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2515 - accuracy: 0.9231
Epoch 140/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2437 - accuracy: 0.9209
Epoch 141/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2560 - accuracy: 0.9187
Epoch 142/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2903 - accuracy: 0.9033
Epoch 143/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2748 - accuracy: 0.9055
Epoch 144/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2920 - accuracy: 0.9055
Epoch 145/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2593 - accuracy: 0.9187
Epoch 146/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2778 - accuracy: 0.9209
Epoch 147/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2524 - accuracy: 0.9165
Epoch 148/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2450 - accuracy: 0.9209
Epoch 149/150
5/5 [==============================] - 0s 1ms/step - loss: 0.2371 - accuracy: 0.9253
Epoch 150/150
5/5 [==============================] - 0s 2ms/step - loss: 0.2687 - accuracy: 0.9231
<tensorflow.python.keras.callbacks.History at 0x7fc814e1add8>
```

```
X_test
```

```
     array([[-0.20175604,  0.3290786 , -0.13086754, ...,  1.3893291 ,
              1.08203284,  1.54029664],
            [-0.25555773,  1.46763319, -0.31780437, ..., -0.83369364,
             -0.73131577, -0.87732522],
            [-0.02619262, -0.8407682 , -0.09175081, ..., -0.49483785,
             -1.22080864, -0.92115937],
            ...,
            [ 1.71811488,  0.09318356,  1.7286186 , ...,  1.57630515,
              0.20317063, -0.15406178],
            [ 1.18859296,  0.34352115,  1.19333694, ...,  0.56019755,
              0.26991966, -0.27320074],
```

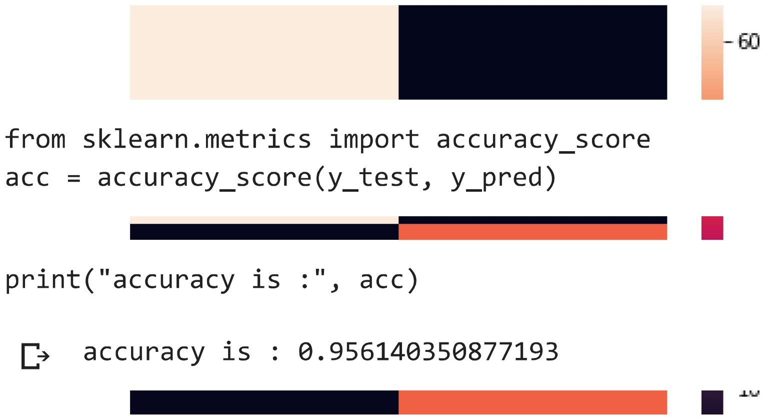
```
#predicting the test set results

y_pred = classifier.predict(X_test)
y_pred = (y_pred > 0.5)
```

```
#making confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,y_pred)
```

```
cm
```

```
     array([[65,  2],
            [ 3, 44]])
```

```
sns.heatmap(cm,annot=True)
plt.savefig("h.png")
```



```
from sklearn.metrics import accuracy_score
acc = accuracy_score(y_test, y_pred)
```

```
print("accuracy is :", acc)
```

```
accuracy is : 0.956140350877193
```