mlp

April 19, 2022

0.0.1 MLP with Confusion Matrix

[1]:
```
import pandas as pd
from sklearn.datasets import load_breast_cancer
```

[2]:
```
cancer_dataset = load_breast_cancer()
```

[4]:
```
from sklearn.model_selection import train_test_split

#split the dataset using train_test_split function, pass train data, labels,␣
 ↪and test data ratio
X_train, X_test, y_train, y_test = train_test_split(
    cancer_dataset.data,cancer_dataset.target,test_size=0.25)
```

[5]:
```
#Importing MLPClassifier
from sklearn.neural_network import MLPClassifier
#Initializing the MLPClassifier
#solver is for weight optimization
#max_iter sets limit till convergence or this value
#activation is the activation function for the hidden layer
classifier = MLPClassifier(hidden_layer_sizes=(150,100,50), max_iter=300,
 activation = 'relu',solver='adam',random_state=1)
classifier.fit(X_train, y_train)
```

[5]:
```
MLPClassifier(hidden_layer_sizes=(150, 100, 50), max_iter=300, random_state=1)
```

[6]:
```
y_pred = classifier.predict(X_test)
```

[7]:
```
from sklearn.metrics import confusion_matrix
#Get the confusion matrix
cf_matrix = confusion_matrix(y_test, y_pred)
print(cf_matrix)
```

```
[[51  0]
 [19 73]]
```

[ ]:
```
Perform the same experiment on the breast cancer dataset
using Logistic Regression and Decision Tree Classification models.
```

```
In each case, identify the confusion matrix and the TP,TN,FP & FN.
Finally, discuss which classifier is best suited for deploying as a
real-time smart cancer diagnosis system.
```

[ ]:

[8]:
```
import numpy as np
cf_matrix=np.transpose(np.transpose(cf_matrix) / cf_matrix.astype(float).
↪sum(axis=1))
```

[9]:
```
cf_matrix
```

[9]:
```
array([[1.        , 0.        ],
       [0.20652174, 0.79347826]])
```

[ ]: