

Exercise – 1. Evaluation of Performance metric for a Classification Problem

Write a python code to display confusion matrix and to calculate the performance metrics like Recall, Precesion, F1score and accuracy. You create synthetic Ground truth values and Predictions

In [109]:

```
#initilializing the array

import numpy as np
X = np.array([.50,1.50,2.00,4.25,3.25,5.50], ndmin=2).reshape((6,1))
y = np.array([1,0,0,1,1,1])
X_mean = np.mean(X)
y_mean = np.mean(y)
n = len(X)
```

In [110]:

```
#logistic regression

from sklearn.linear_model import LogisticRegression
lr = LogisticRegression(penalty='none')
lr.fit(X, y)
y_pred = lr.predict(X)
y_pred
```

Out[110]:

```
array([0, 0, 1, 1, 1, 1])
```

In [111]:

```
#confusion matrix

from sklearn.metrics import confusion_matrix
confusion_matrix(y, y_pred)
```

Out[111]:

```
array([[1, 1],
       [1, 3]], dtype=int64)
```

#classification report for precision, recall, f1-score and accuracy

In [112]:

```
#finding precision

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
precision_score(y,y_pred)
```

Out[112]:

```
0.75
```

In [113]:

```
#finding accuracy  
accuracy_score(y,y_pred)
```

Out[113]:

0.6666666666666666

In [114]:

```
#finding recall  
recall_score(y,y_pred)
```

Out[114]:

0.75

In [115]:

```
#finding f1_score  
f1_score(y,y_pred)
```

Out[115]:

0.75

In []: