

# **EVALUATION METRICS FOR DATA CLASSIFICATION EVALUATIONS**

## **1. INTRODUCTION**

In data classification problems, data can be divided into commercial data, texts, DNAs, and images. This paper emphasizes commercial data as the focus of discussion. Furthermore, data classification can be divided into binary, multiclass and multi-labelled classification. In this paper, the study is aimed on binary and multiclass classification which focuses on the evaluation metrics for evaluating the effectiveness of classifiers. In general, the evaluation metric can be described as the measurement tool that measures the performance of classifier. Different metrics evaluate different characteristics of the classifier induced by the classification algorithm.

## **2. REVIEW OF DISCRIMINATOR METRICS**

In a typical data classification problem, the evaluation metric has been employed into two stages, which are training stage (learning process) and testing stage. In training stage, the evaluation metric was used to optimize the classification algorithm. In other words, the evaluation metric was employed as the discriminator to discriminate and to select the optimal solution which can produce a more accurate prediction of future evaluation of a particular classifier. Meanwhile, in the testing stage, the evaluation metric was used as the evaluator to measure the effectiveness of produced classifier when tested with the unseen data.





## Why is it called as confusion matrix:

Well, it is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Confusion Matrix [Image 1]

- It is extremely useful for measuring Recall, Precision, Specificity, Accuracy, and most importantly AUC-ROC curves.
- Let's understand TP, FP, FN, TN in terms of pregnancy analogy.
  1. TP – TRUE POSITIVE
  2. FP – FALSE POSITIVE
  3. FN – FALSE NEGATIVE
  4. TN – TRUE NEGATIVE

		Actual Values	
		1	0
Predicted Values	1	<b>TRUE POSITIVE</b> 	<b>FALSE POSITIVE</b>  <b>TYPE 1 ERROR</b>
	0	<b>FALSE NEGATIVE</b>  <b>TYPE 2 ERROR</b>	<b>TRUE NEGATIVE</b> 

Confusion Matrix [Image 2]

### **True Positive:**

Interpretation: You predicted positive and it's true.

You predicted that a woman is pregnant, and she actually is.

### **True Negative:**

Interpretation: You predicted negative and it's true.

You predicted that a man is not pregnant, and he actually is not.

### **False Positive: (Type 1 Error)**

Interpretation: You predicted positive and it's false.

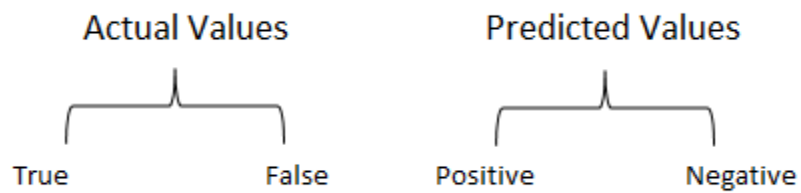
You predicted that a man is pregnant, but he actually is not.

## False Negative: (Type 2 Error)

Interpretation: You predicted negative and it's false.

You predicted that a woman is not pregnant, but she actually is.

Just Remember, we describe predicted values as Positive and Negative and actual values as True and False.



Actual vs Predicted values [Image 3]

## How to Calculate Confusion Matrix for a 2-class classification problem?

Let's understand the confusion matrix through math.

y	y pred	output for threshold 0.6	Recall	Precision	Accuracy
0	0.5	0	<b>1/2</b>	<b>2/3</b>	<b>4/7</b>
1	0.9	1			
0	0.7	1			
1	0.7	1			
1	0.3	0			
0	0.4	0			
1	0.5	0			

Confusion Matrix [Image 4]

### 1. Accuracy(acc):

In general, the accuracy metric measures the ratio of correct predictions over the total number of instances evaluated.

$$(TP + TN) / (TP + TN + FP + FN).$$

## 2. Precision (p):

Precision is used to measure the positive patterns that are correctly predicted from the total predicted patterns in a positive class.

$$TP / (TP + FP)$$

## 3. Recall (r):

Recall is used to measure the fraction of positive patterns that are correctly classified.

$$TP / (TP + FN)$$

## 4.F1-score:

F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall and is often used as a summary metric for classification performance. It is calculated as

$$2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

For our understanding, My hands-on from

```
sklearn.linear_model import LogisticRegression
```

```
from sklearn.metrics import accuracy_score
```

```
#train_model
```

```
lr = LogisticRegression().fit(x_train,y_train)
```

```
y_pred = clf_0.predict(x_val)
```

```
accuracy_score(y_pred,y_val)
```

```
Out[47]: 0.8048780487804879
```

```
from sklearn.metrics import accuracy_score,confusion_matrix
```

```
cm = confusion_matrix(y_val, y_pred)
```

```
sns.heatmap(cm,square = True,annot = True,cbar= False)
```

```
Out[48]:
```

