ROBOFIED

Evaluating Classification Models Part 1

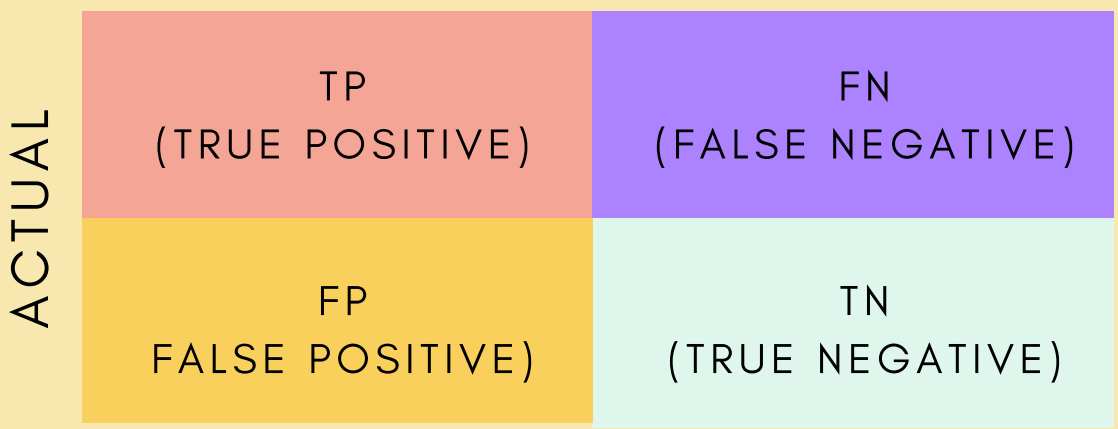
FIXING FUNDAMENTALS
BY ROBOFIED

WWW.ROBOFIED.COM

ROBOFIED

01

Confusion Matrix



From the name we can tell, Confusion Matrix output is a matrix and it is quite ‘confusing’? However, it is actually quite intuitive if you understand the logic. Confusion Matrix is a 2*2 table (for binary class classification)and it is the basis of many other metrics. Assume your classification only has two categories results (1 or 0), a confusion matrix is the combination of your prediction (1 or 0) vs actual value (1 or 0).

ROBOFIED

Confusion Matrix Contd...

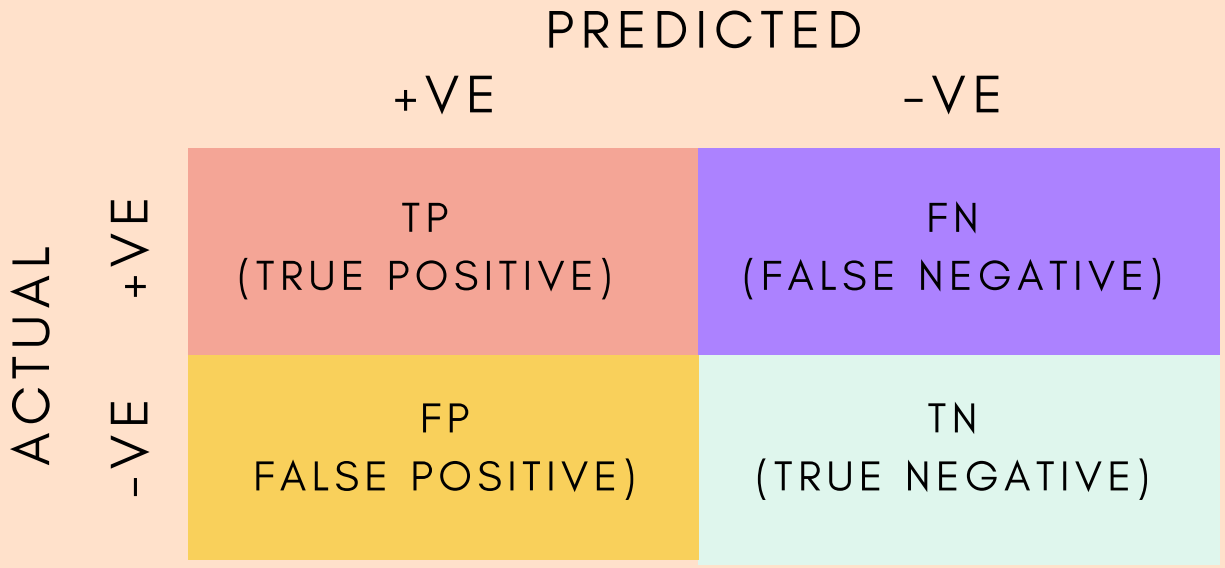
02

True Positive: Case when you correctly predict the positive result.

False Positive: Case when you predict the result to be positive but it is actually negative.

False Negative: Case then you predict the result to be negative but it is actually positive.

True Negative: Case when you correctly predict negative result.



03

ROBOFIED
Code

```
from sklearn.metrics import confusion_matrix
y_true = [2, 0, 2, 2, 0, 1]
y_pred = [0, 0, 2, 2, 0, 2]
confusion_matrix(y_true, y_pred)
>>>array([[2, 0, 0],
        [0, 0, 1],
        [1, 0, 2]])
```

ROBOFIED

04

Accuracy

TP (TRUE POSITIVE)	FN (FALSE NEGATIVE)
FP FALSE POSITIVE)	TN (TRUE NEGATIVE)

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

Accuracy is purely how much percentage of cases that you have predicted correctly on overall.

When to use Accuracy? Accuracy is a good measure of how the overall model performs. However,
it is not telling you the performance in each category and thus you may miss important information if you purely look at accuracy. It is best to used when number of cases in different categories are similar or used together with other metrics.

05

ROBOFIED Recall

Recall measures how much percentage of real positive cases are correctly identified.

When to use Recall? It is used when the aim is to capture maximum number of positive cases.
For example, during the recent Covid-19 situation, the government wants to track all the infected cases in the community. If you have built an model to predict whether the person is infected with Covid-19 based on symptoms, Recall will be an important metric to be measured.

TP (TRUE POSITIVE)	FN (FALSE NEGATIVE)
FP FALSE POSITIVE)	TN (TRUE NEGATIVE)

$$recall = \frac{TP}{TP + FN}$$

ROBOFIED

06

Precision

TP (TRUE POSITIVE)	FN (FALSE NEGATIVE)
FP FALSE POSITIVE)	TN (TRUE NEGATIVE)

$$precision = \frac{TP}{TP + FP}$$

Precision measures that among the cases predicted to be positive, how much percentage of them are really positive.

When to use Precision? Precision is used when you want to be very accurate in measuring positive cases. It always conflicts with Recall because the more positive cases you want to capture, the lower the standard you will put to be classified as positive cases.
In the Covid-19 cases example, if you want to capture more infected cases, you may want to include all cases that have slight symptoms that could just be because of normal flu.

07

ROBOFIED
Code

```python
#Calculate Accuracy
from sklearn.metrics import accuracy_score
accuracy_score(y_test,y_predicted)
# Accuracy is 0.8812499999999998

#Calculate Recall
from sklearn.metrics import recall_score
recall_score(y_test, y_predicted)
#Recall is 0.8410958904109589

#Calculate Precision
from sklearn.metrics import precision_score
precision_score(y_test, y_predicted)
# Precision is 0.8457300275482093
```

ROBOFIED

08

F-1 Score

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$F1 - Score = 2 \cdot \frac{(precision \cdot recall)}{(precision + recall)}$$

Considering the conflicting feature between precision and recall, F1 Score is created to have a balanced metric between recall and precision. It is the Harmonic mean of recall and precision. You may want to ask: why we do not simply average Precision and Recall?

Well, if the distribution of positive and negative cases are very uneven, the average may not be a good representation of model performance.

09

ROBOFIED
Code

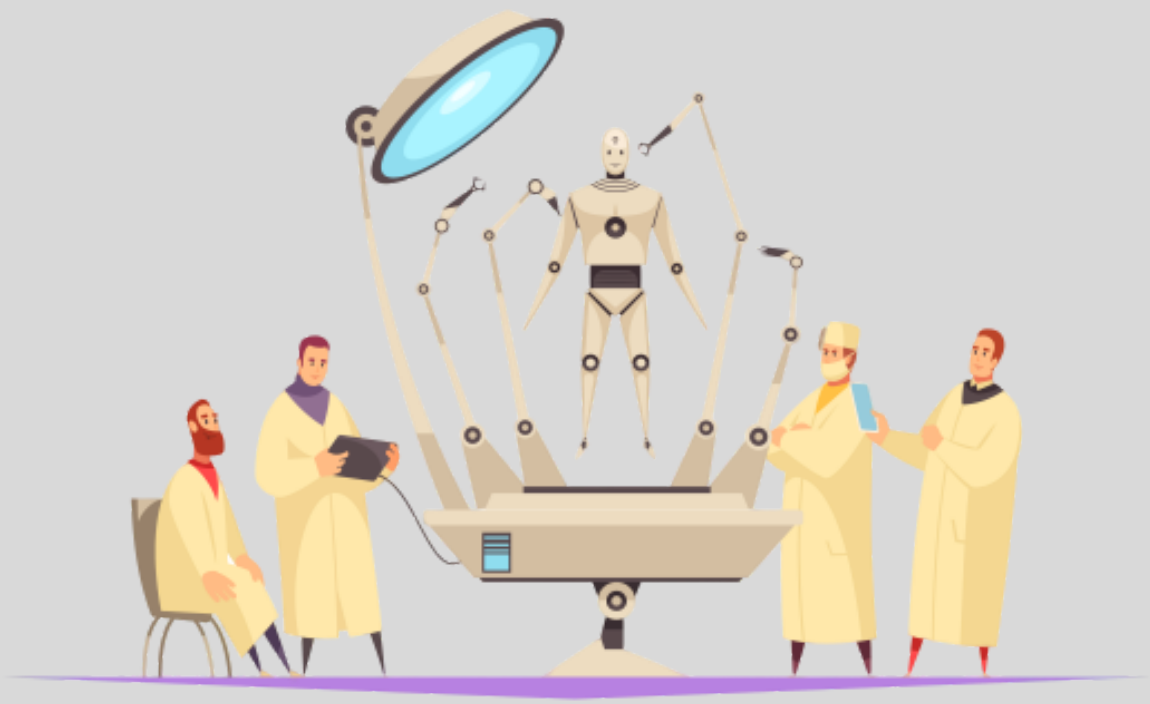
```
#calculate F1 score
from sklearn.metrics import f1_score
f1_score(y_test,y_predicted)
# F1 score is 0.8434065934065941
```

ROBOFIED

Other Resources

10

1. Machine Learning Mastery Blogs by Jason Brownlee
2. CS229 Machine Learning @Stanford by Andrew Ng
3. The Elements of Statistical Learning Book by Trevor Hastie, Robert Tibshirani, Jerome H. Friedman



WWW.ROBOFIED.COM

ROBOFIED

At **Robofied,**
we aspire to democratize **AI**
Education

Join our channels:

www.robofied.com
https://instagram.com/robofied
https://twitter.com/Robofied