Confusion Matrix

Research · October 2021

Zohreh Karimi
Kharazmi University

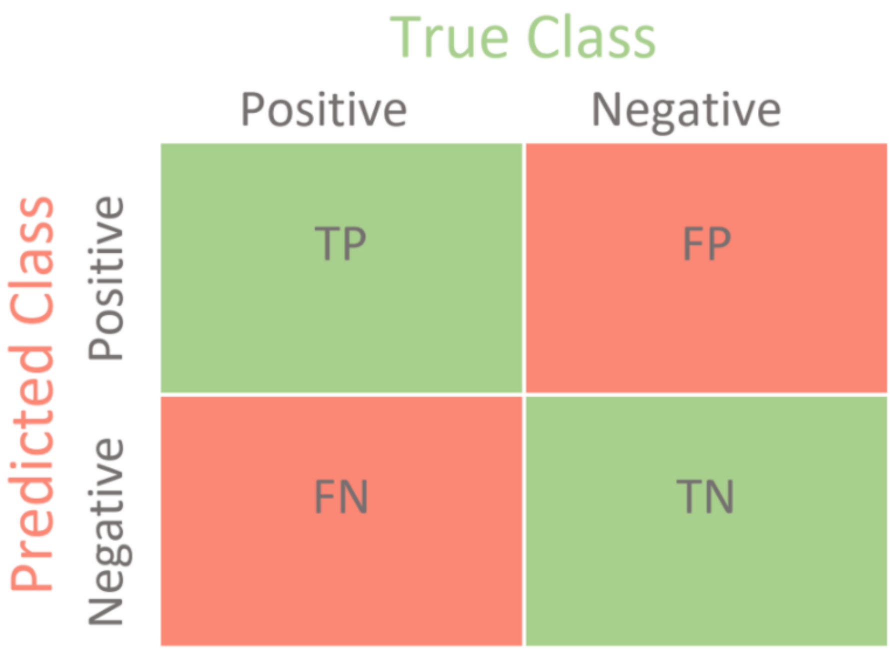
Confusion Matrix

* The confusion matrix is a tool for predictive analysis In machine learning. In order to check the performance of a classification based machine learning model, the confusion matix is deployed.

* Also we can say Confusion matrix is a summerized table of the number of correct and incorrect predictions yielded by a classifier (or a classification model) for binary classification tasks.

* A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes.

By visualzing the confusion matrix, an individual could determine the accuracy of the model by observing the diagonal values for measuring the number of accurate classification.



The confusion matrix is in the form of a square matrix whre the column represents the actual values and the row depicts the predicted value of the model and vica versa.

- **TP: True Positive:** The actual value was positive and the model predicted a positive value
- **FP: False Positive**: Your prediction is positive, and it is false. (Also known as the **Type 1 error**)
- **FN: False Negative:** Your prediction is negative, and result it is also false. (Also known as the **Type 2 error**)
- **TN: True Negative:** The actual value was negative and the model predicted a negative value

1- Accuracy:

Accuracy is a measure for how many correct predictions your model

made for the complete test dataset. Accuracy is a good basic metric to measure the performance of the model. In unbalanced datasets, accuracy becomes a poor metric

$$\text{accuracy}=\frac{TP+TN}{TP+TN+FP+FN}$$

2- Misclassification:

$$\text{misclassification}=\frac{FP+FN}{TP+TN+FP+FN}$$

3- Precison

Precison tells us how many of the correctly predicted case actually turned out to be positive. This would determine whether our model is reliable or not.

Precison is a useful metric in case where False Pasitive is a higher concern than False Negative.

$$\text{Precision}= \frac{TP}{TP+TN}$$

4- Recall(sensivity)

Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

Recall is a useful metric in cases where False Negative trumps False Positive.

$$\text{Recall}= \frac{TP}{TP+FN}$$

A higher recall means that most of the positive cases (TP+FN) will be labeled as positive(TP). This wil lead to a higher number of FP measurement and a lower overall accuracy.

A low recall means that you have a high number of FN (should have been positive but labeled as negative). This means that you have more certainty if you found a positive case, this is like to be a true positive.

5- F1-Score

When we try to increase the precision of model , the recall grows down and vice-versa.

F1-Score is a harmonic mean of Precison and Recall and so it gives a combined idea about these two metrics. It is maximum when precision is equal to recall.

$$\text{F1-Score}= \frac{1}{\frac{1}{recall}+\frac{1}{precision}}$$

The interpretability of the F1-Score is poor, it means that we don't know what our classifier is maximising- precision or recall. So, we use it in combination with other evaluation metrics which gives us a complete picture of the result

6-False Positive rate

False positive rate is a measure for how many results get predicted as positive out of all the negative cases.

$$\text{FPR}= \frac{FP}{TN+FP}$$

How to get these metrics in scikitlearn

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

 # Accuracy

accuracy_score(y_test, y_predicted)

 # Precision

precision_score(y_test, y_predicted)

 # Recall

recall_score(y_test, y_predicted)

 # F1-score

f1_score(y_test, y_predicted)