**Confusion Matrix**

**Confusion Matrix**

The confusion matrix is a matrix used to determine the performance of the classification models for a given set of test data. It can only be determined if the true values for test data are known. The matrix itself can be easily understood, but the related terminologies may be confusing. Since it shows the errors in the model performance in the form of a matrix, hence also known as an **error matrix**.

The confusion matrix consists of four basic characteristics (numbers) that are used to define the measurement metrics of the classifier. These four numbers are: TP [ True Positive] FP [ False Positive] TN [ True Negative] FN [False Negative].

For our better understanding, consider a table below which denotes about the healthy tissue and the cancerous tissue. In this table, benign tissue is called healthy and *malignant* tissue is considered cancerous.

**Predicted value**

|  |  |  |
| --- | --- | --- |
| Actual value | Malignant | Benign |
| Malignant | TP | FN |
| Benign | FP | TN |

1.**TP (True Positive):** TP represents the number of patients who have been properly classified to have malignant nodes, meaning they have the disease.

2.**TN (True Negative):** TN represents the number of correctly classified patients who are healthy.

3. **FP (False Positive):** FP represents the number of misclassified patients with the disease but actually they are healthy. FP is also known as a *Type I error*.

4. **FN (False Negative):** FN represents the number of patients misclassified as healthy but actually they are suffering from the disease. FN is also known as a *Type II error*.

**Calculations using Confusion Matrix:**

We can perform various calculations for the model, such as the model's accuracy, using this matrix. These calculations are given below:

* **Classification Accuracy:** It is one of the important parameters to determine the accuracy of the classification problems. It defines how often the model predicts the correct output. It can be calculated as the ratio of the number of correct predictions made by the classifier to all number of predictions made by the classifiers. The formula is given below:  
  Confusion Matrix in Machine Learning
* **Misclassification rate:** It is also termed as Error rate, and it defines how often the model gives the wrong predictions. The value of error rate can be calculated as the number of incorrect predictions to all number of the predictions made by the classifier. The formula is given below:  
  Confusion Matrix in Machine Learning
* **Precision:** It can be defined as the number of correct outputs provided by the model or out of all positive classes that have predicted correctly by the model, how many of them were actually true. It can be calculated using the below formula:  
  Confusion Matrix in Machine Learning
* **Recall:** It is defined as the out of total positive classes, how our model predicted correctly. The recall must be as high as possible.  
  Confusion Matrix in Machine Learning
* **F-measure:** If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the precision. It can be calculated using the below formula:  
  Confusion Matrix in Machine Learning

**Confusion Matrix for Binary Classification**

Let us understand the confusion matrix for a simple binary classification example.

Binary classification has 2 outputs. The inputs for this classification will fall in either of the 2 outputs or classes.

Example: Based on certain inputs, we have to decide whether the person is sick or not, diabetic or not.

Let us see how to construct a confusion matrix and understand its terminologies. Consider we have to model a classifier that classifies 2 kinds of fruits. We have 2 types of fruits – apples and grapes – and we want our machine-learning model to identify or classify the given fruit as an apple or grape.

So we take 15 samples of 2 fruits, out of which 8 belong to Apples, and 7 belong to the Grapes class. Class is nothing but the output, in this example, we have 2 output classes – Apples and Grapes. We will represent Apple as 1 and Grape as 0 class.

The actual class for 8 apples and 7 grapes can be represented as:

Actual = [1,1,1,1,1,1,1,1,0,0,0,0,0,0,0]

The classifier model predicts 1 for Apple and 0 for grape.

Assume that the classifier takes all 15 inputs and makes the following predictions:

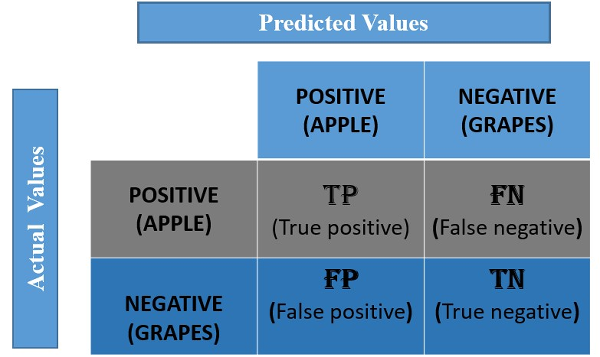
· Out of 8 apples, it will classify 5 correctly as apples and wrongly predict 3 as grapes.

· Out of 7 grapes, it will classify 5 correctly as grapes and wrongly predicts 2 as apples.

The prediction of the classifier may be as follows:

Prediction = [1,0,0,0,1,1,1,1,0,0,0,0,0,1,1]

The confusion matrix for this example can be visualized below.



For our example, the positive value is Apple, and the negative value is Grapes.

**True Positive:**

It means the actual value and also the predicted values are the same. In our case, the actual value is also an apple, and the model prediction is also an apple. If you observe the TP cell, the positive value is the same for Actual and predicted.

**False Negative:**

This means the actual value is positive. In our case, it is apple, but the model has predicted it as negative, i.e., grapes. So the model has given the wrong prediction. It was supposed to give a positive (apple), but it has given a negative (grape). So whatever the negative output we got is false; hence the name False Negative.

**False Positive:**

This means the actual value is negative. In our case, it is grapes, but the model has predicted it as positive, i.e., apple. So the model has given the wrong prediction. It was supposed to give a negative (grape), but it has given a positive (apple), so whatever the positive output we got is false, hence the name False Positive.

**True Negative:**

It means the actual value and also the predicted values are the same. In our case, the actual values are grapes, and the prediction is also Grapes.

The values for the above example are:

TP=5, FN=3, FP=2, TN=5.

**Confusion Matrix for Multi-Class Classification:**

The above example is a binary classification model with only 2 outputs, so we got a 2 X 2 matrix. If the outputs are greater than 2 classes, we need to find confusion matrix for multi class classification

*Confusion Matrix for a 3-Class Classification:*

Let’s try to understand the confusion matrix for 3 classes and the confusion matrix for multiple classes with a popular dataset – the IRIS DATASET.

The dataset has 3 flowers as outputs or classes, Versicolor, Virginia, and Setosa.



With the help of petal length, petal width, sepal length, and sepal width, the model has to classify the given instance as Versicolor or Virginia, or Setosa flower.

Let’s apply a classifier model here. We can use logistic regression, but a decision tree classifier is applied to the above dataset. The dataset has 3 classes; hence we get a 3 X 3 confusion matrix.

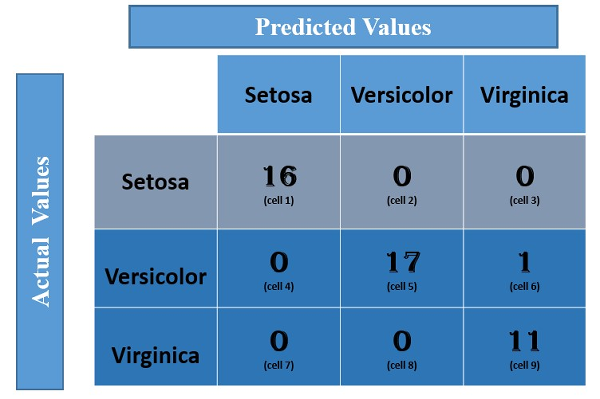
*But how to know the TP, TN, FP, and FN values?*

In the multi-class classification problem, we won’t get TP, TN, FP, and FN values directly as in the binary classification problem. For validation, we need to calculate for each class.

*How to Calculate FN, FP, TN, and TP Values?*

As discussed earlier, FN: The False-negative value for a class will be the sum of values of corresponding rows except for the TP value. FP: The False-positive value for a class will be the sum of values of the corresponding column except for the TP value. TN: The True-negative value for a class will be the sum of the values of all columns and rows except the values of that class that we are calculating the values for. And TP: the True-positive value is where the actual value and predicted value are the same.

The confusion matrix for the IRIS dataset is as below:



Let us calculate the TP, TN, FP, and FN values for the class **Setosa**using the Above tricks:

**TP**: The actual value and predicted value should be the same. So concerning Setosa class, the value of cell 1 is the TP value.

**FN**: The sum of values of corresponding rows except for the TP value

FN = (cell 2 + cell3)

= (0 + 0)

= 0

**FP:**The sum of values of the corresponding column except for the TP value.

FP = (cell 4 + cell 7)

= (0 + 0)

= 0

**TN:**The sum of values of all columns and rows except the values of that class that we are calculating the values for.

TN = (cell 5 + cell 6 + cell 8 + cell 9)

= 17 + 1 +0 + 11

= 29

Similarly, for the **Versicolor** class, the values/metrics are calculated as below:

TP: 17 (cell 5)

FN : 0 + 1 = 1 (cell 4 +cell 6)

FP : 0 + 0 = 0 (cell 2 + cell 8)

TN : 16 +0 +0 + 11 =27 (cell 1 + cell 3 + cell 7 + cell 9).

**Other important terms used in Confusion Matrix:**

* **Null Error rate:** It defines how often our model would be incorrect if it always predicted the majority class. As per the accuracy paradox, it is said that "*the best classifier has a higher error rate than the null error rate.*"
* **ROC Curve:** The ROC is a graph displaying a classifier's performance for all possible thresholds. The graph is plotted between the true positive rate (on the Y-axis) and the false Positive rate (on the x-axis).

**Conclusion**

* The confusion matrix is a very popular measure used while solving classification problems. It can be applied to binary classification as well as to multiclass classification problems.
* The confusion matrix gives a comparison between actual and predicted values. The confusion matrix is a N x N matrix, where N is the number of classes or outputs.