**Logistic Regression**

Logistic Regression is a popular supervised learning algorithm used in machine learning for binary classification tasks. It is mainly employed when the target variable is categorical with two possible classes or labels, typically represented as 0 and 1 or “Yes” and “No”.

For example

1. Fraud or Not
2. Email spam or not Spam
3. Infected with Disease or not Disease.

There are two types of approaches to compute logistic regression Graphical and Probabilistic approach.

**Graphical:**

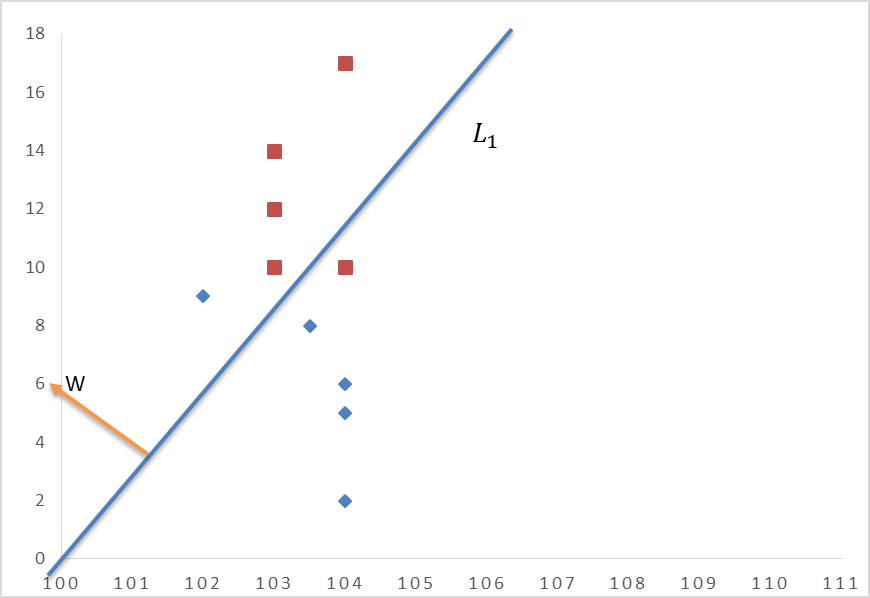
In the graphical approach,

* a best-fit line is established to distinguish between data points.
* After the creation of this best-fit line, the next step involves determining whether data points lie on the positive or negative side of the line.
* Then we decide if the point is correctly classified or not.
* Compute the cost Using Sigmoid function.
* Line with highest value to be considered the final line.

In this context, we consider the upper portion of the line as the positive side, and the lower portion of the line as the negative side.

Assume that below is the line best-fit line created using

y = mx + c

****

The below equation can be used to determine exactly where the point lies, either on the +Ve or –Ve side of the line.

Only for the equation simplicity, we will Consider the line is passing through the origin hence will become 0 and the above equation can be rewritten as below.

If is unit normal then we can use the below equation

If the output of the equation is +ve then the point lies on +ve side of the line and vice-versa.

**How to know if the point is correctly classified or not?**

To identify if the line has predicted the correct point or not multiply the “actual points” and the “predicted points” and if the result of the same is +ve then we can consider that the particular point is correctly classified and if the result of the same is –ve then consider that the point is incorrectly classified.

Let us compute the Graphical approach in the below example.

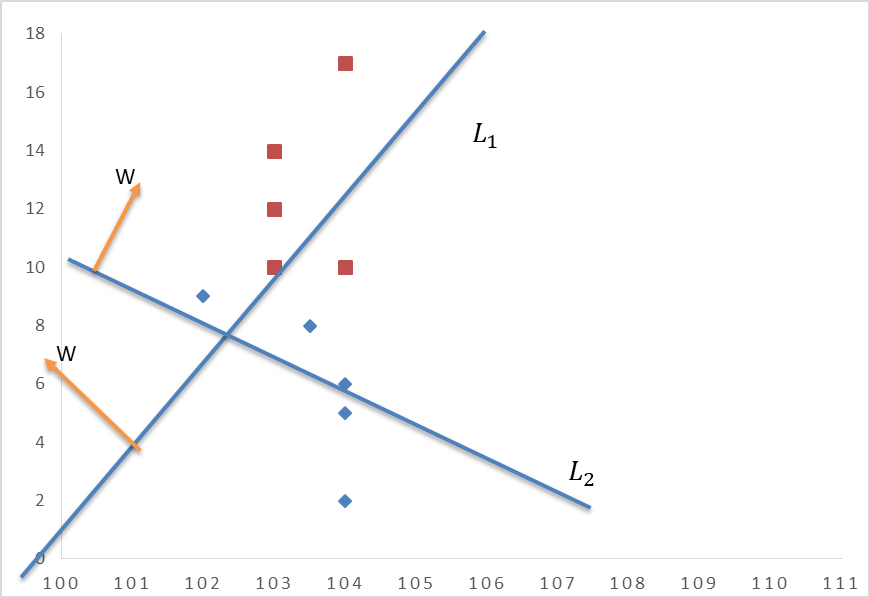
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Reading 01 | Reading 02 | Actual Points(y)  (+1 and -1) | Predicted Points(d)  (+1 and -1) | +VE or -VE Output | Status |
|  |  |  |  |  |  |
| 2 | **17** | **1** | **1** | 1 | Correct |
| 0 | **14** | **1** | **1** | 1 | Correct |
| 0 | **12** | **1** | **1** | 1 | Correct |
| 0 | **10** | **1** | **1** | 1 | Correct |
| 0 | **10** | **1** | **-1** | -1 | Incorrect |
| 9 | **0** | **-1** | **1** | -1 | Incorrect |
| 8 | **0** | **-1** | **-1** | 1 | Correct |
| 6 | **0** | **-1** | **-1** | 1 | Correct |
| 5 | **0** | **-1** | **-1** | 1 | Correct |

Steps of Logistic Regression

* Establish the best fit Linear lines that will clearly separate the data points.
* Know if the point is correctly classified or not.
* Select the line which has a large output.

**Establish the best fit Linear lines that will clearly separate the data points.**

Assume two 𝐿1 AND 𝐿2 Lines are created as shown in the below graph. Now we need to identify which is the best-fit line and separate the maximum point as the correct point for the same We will follow the below process.

****

**Identify the best-fit line which is clearly separating the data point and consider it as a final line.**

The same can be identified by using the below equation.

In this process,

* we will multiply the actual and predicted values for each line and calculate the sum separately for each line.
* This calculation will yield a numerical value. Next, we will compare the calculated values for each line and select the one with the highest output.
* Additionally, we will address any issues related to distance during this evaluation.

We will use the same equation for the L1 and L2 and see which is the best-fit line by using the below equation.

**For L1**

**For L2**

Hence model will select L2 as the final Line, however we still need to find the where will be the maximum. The same can be obtained by the below equation.

Optimization function

**Problem with distance measure() :**

Here we will discuss What is a problem and how to solve the same by using the below example.

|  |  |
| --- | --- |
|  |  |
|  |  |

Certainly, you may have noticed that although the L1 line divides the data points accurately, still it won't be chosen as the final line or the most accurate line. This is because the total outcome of model one (-44) is inferior to that of model two (1).

Hence to overcome such a scenario, we will introduce adjustments to the input approach. Instead of utilizing a projected point distance of 50 for the outlier point in L1, and +2, +3, -2, -3 predicted point distances for the L2 Line, we will now employ a distance of +1 or -1 based on the positions of the respective points. We will then re-compute the values and make a comparison between the results.

|  |  |
| --- | --- |
|  |  |

It's important to recognize that the process of compressing values, such as transforming +50, +2, and +3 to 1, and -2 and -3 to -1, is accomplished through the utilization of a sigmoid function. This sigmoid function effectively scales down large or infinitely positive/negative values to +1 or -1, respectively. As shown in the below image.



**The sigmoid function:**

The sigmoid function has the following formula:

Applying log to the above function and considering

The above function is the loss function for logistic regression in the geometric approach.

Going further if then the above equation can rewritten as below

Here we need to consider that if the output is +ve value then the output will be the minimum

If then the above equation can rewritten below

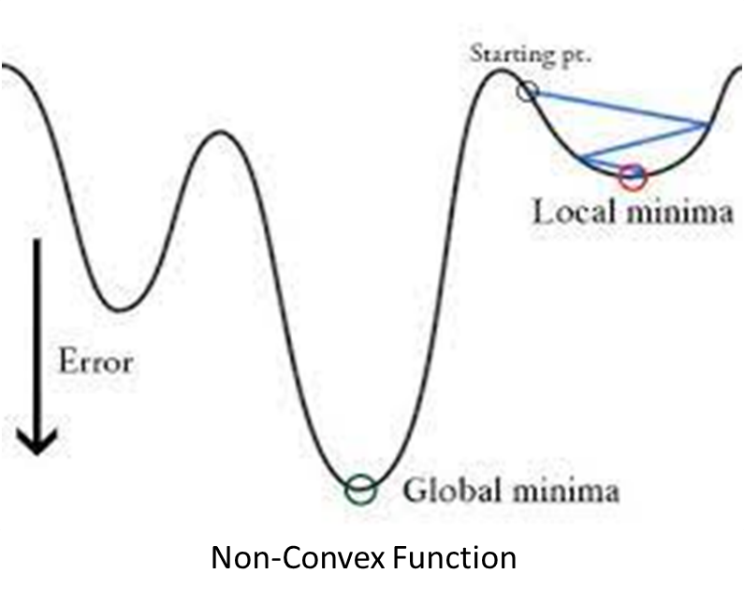
By employing the equation mentioned earlier, the resulting value will be notably greater.

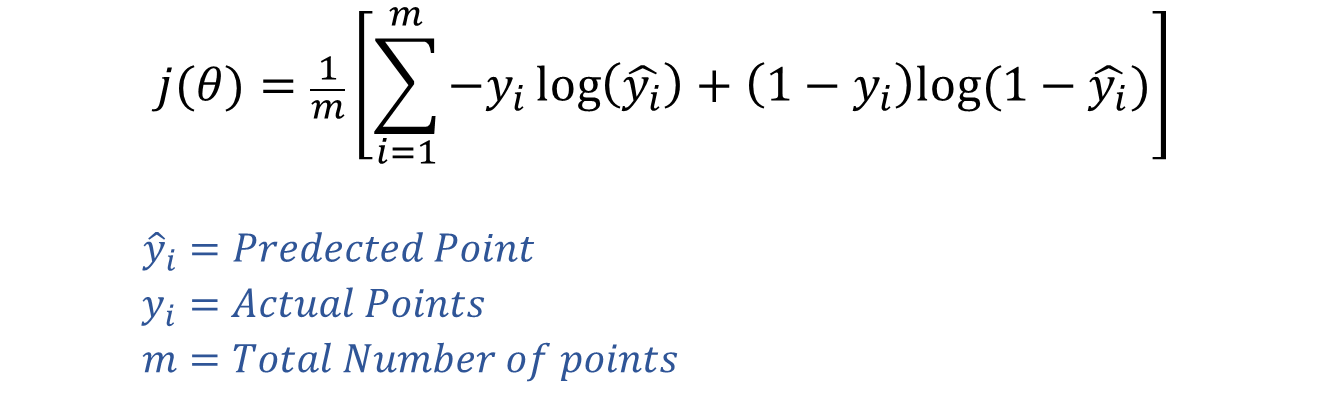
This procedure outlined above must be iterated across all the data points, with their respective outcomes being aggregated. Through this process, we come to comprehend that if the summation is minimized, it indicates that the model is performing optimally. Conversely, if the aggregated value is substantial, it signifies that the model is yielding a higher number of incorrectly classified points, thereby implying its inadequacy as a model.

**Probabilistic Approach:**

**Log Loss (Cross entropy) Cost Function:**

The main issue with the above equation is that it generates a non-convex function, leading to numerous local minima (as shown in the below image) and hence restricting the model from reaching the global minima. This, in turn, produces an inaccurate model. To tackle this problem, the scientists devised the “**Log Loss Cost Function**” elucidated below.





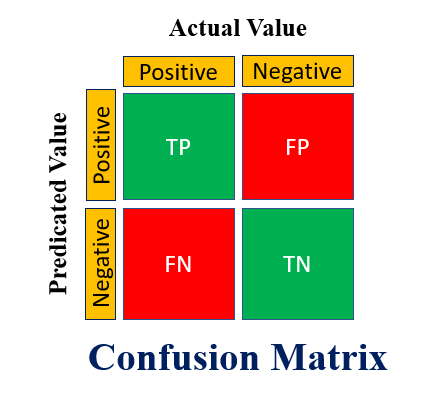
The Log Loss Cost Function is a probabilistic approach, it creates a convex function, which has a single global minimum. This property facilitates the model's ability to reach the global minimum easily, enabling the creation of an accurate model.

**Performance Metrix for classification.**

1. Confusion matrix
2. Accuracy
3. Precision
4. Recall
5. F-Beta Square

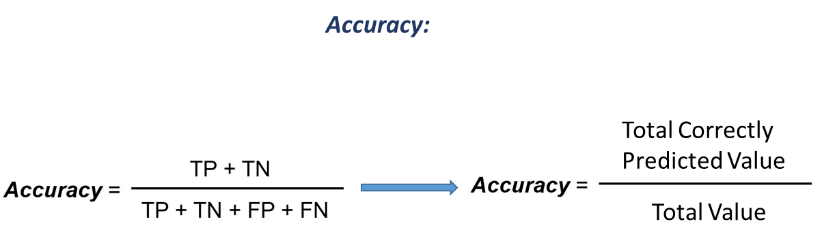
**Confusion matrix :**

The confusion matrix can be explained by studying the graph below.



**Accuracy:**

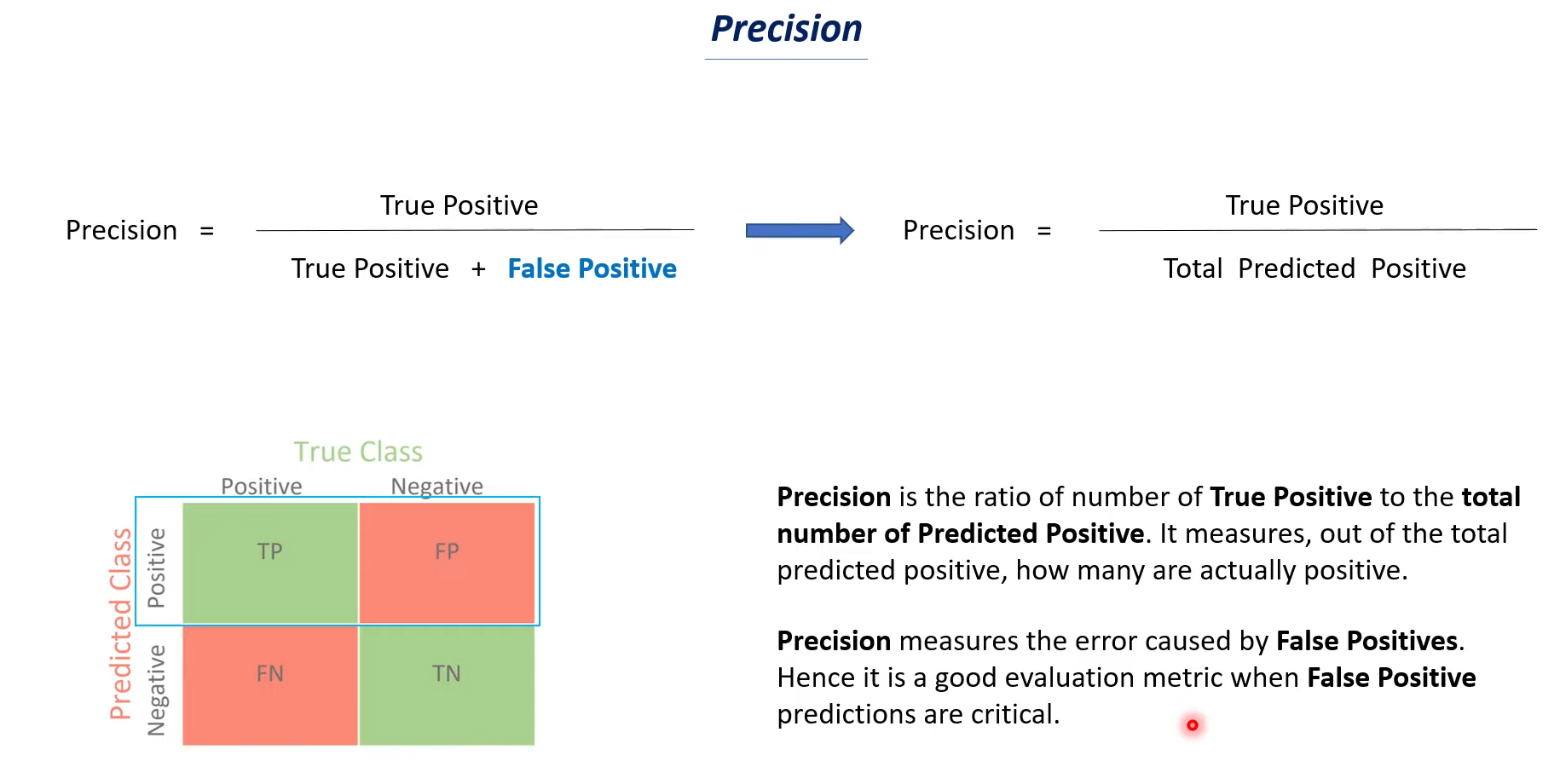
The accuracy function will predict the overall performance of the model.

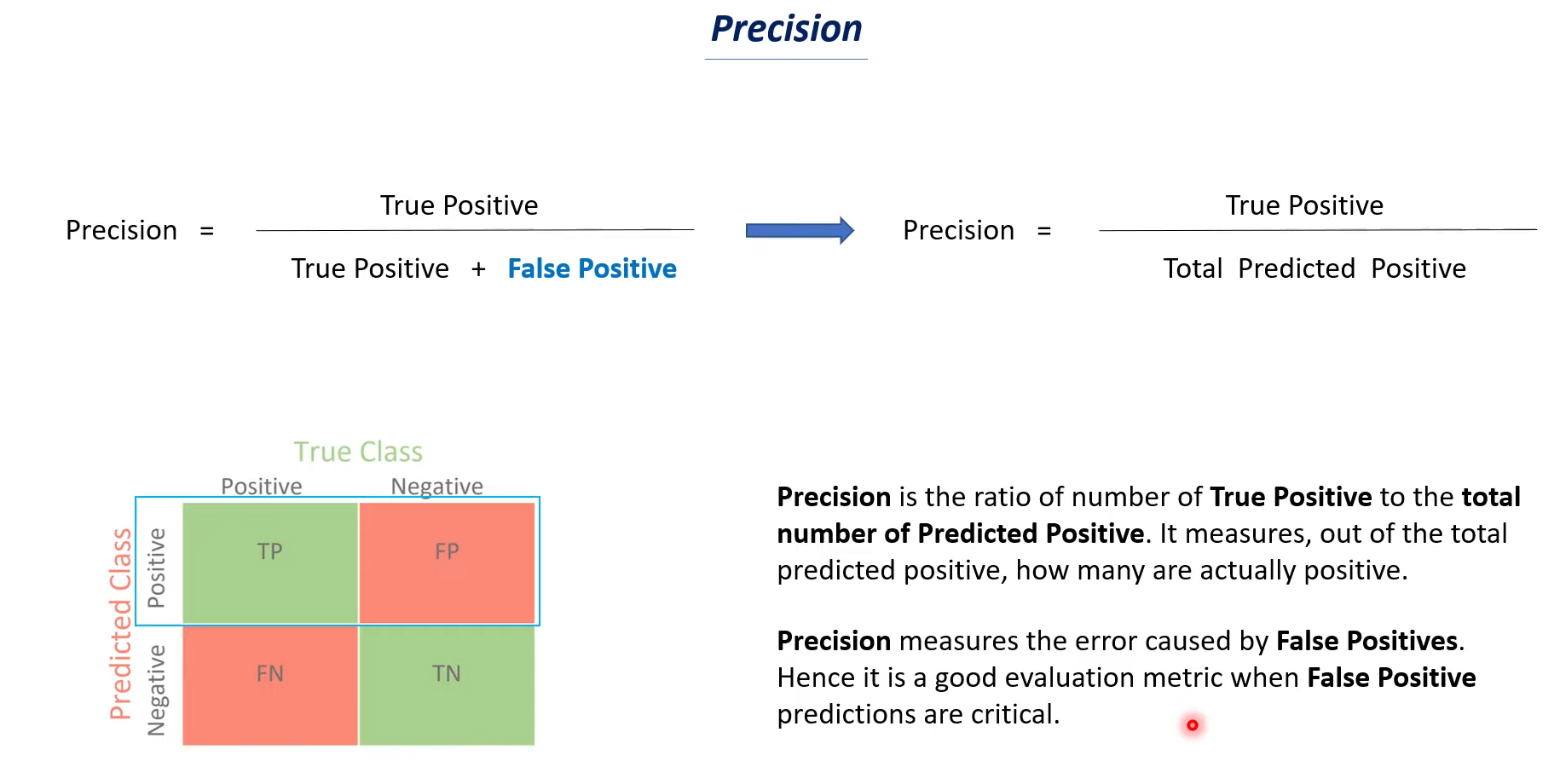


**Precision**

We focus on Precision when It is important to evaluate and **minimize** “**false positives**”, especially in sensitive domains where false negatives are less of a concern compared to false positives.

High Precision means that the model is effective where we want to minimize “**false positives**”.





Example :

**Face recognition:**

If a mobile face detection app incorrectly identifies a person's face as the rightful owner, it may unlock the mobile device and grant unauthorized access to sensitive information. This is a serious concern, as it compromises the security and privacy of the device owner.

In the case of a mobile face detection app, false negatives are less of a concern compared to false positives because, if the rightful person's face is incorrectly identified as negative, they can simply retry or use an alternative authentication method like a PIN to unlock the mobile device.

**Email classification to Spam:**

In this case, let's focus on the false positives. The model predicted 20 instances as spam, but in reality, they were not spam. These could be legitimate emails that were wrongly classified as spam.

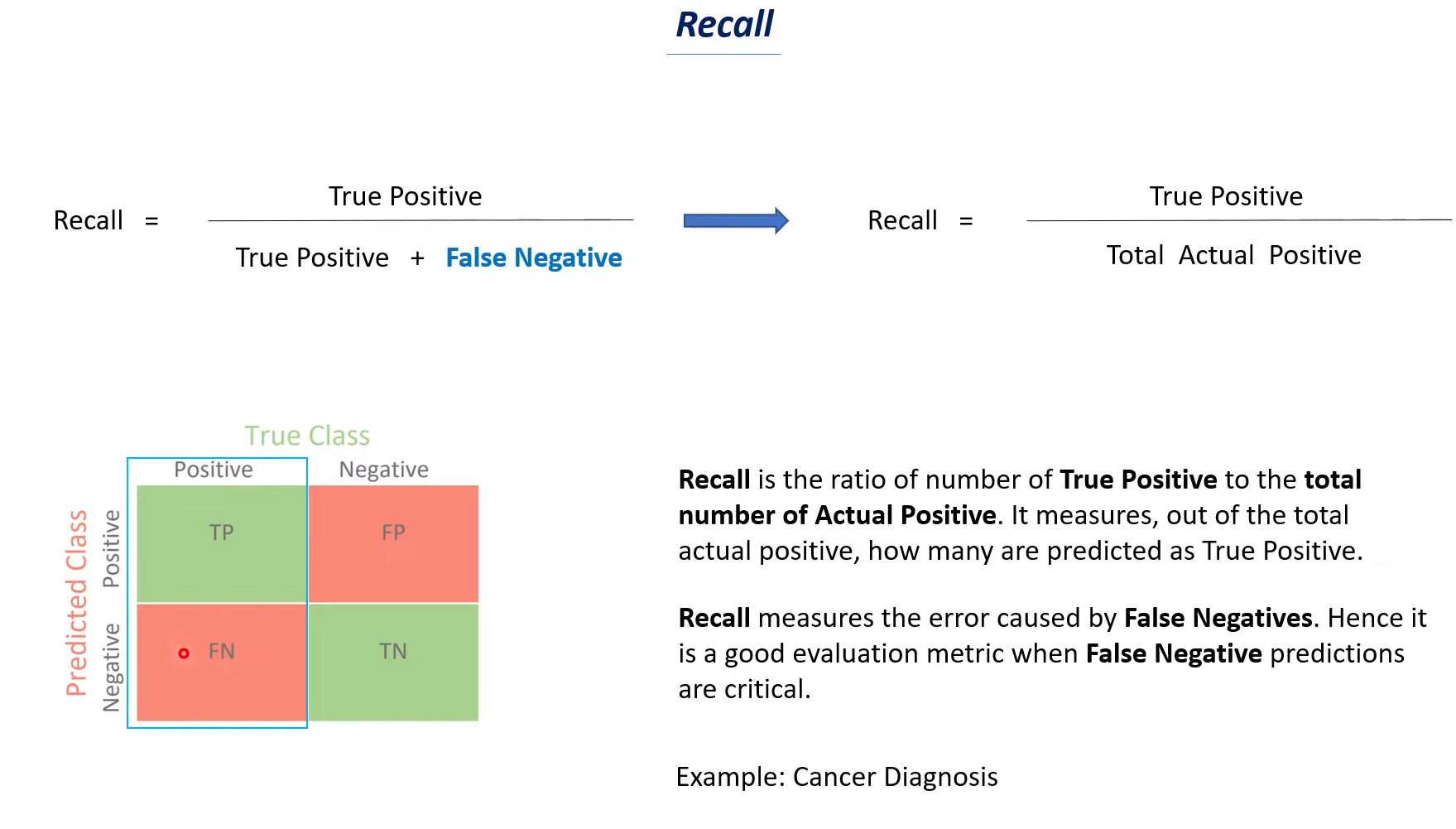
False positives can have negative consequences as they can lead to important emails being sent to the spam folder or filtered out, causing inconvenience or missed opportunities for users.

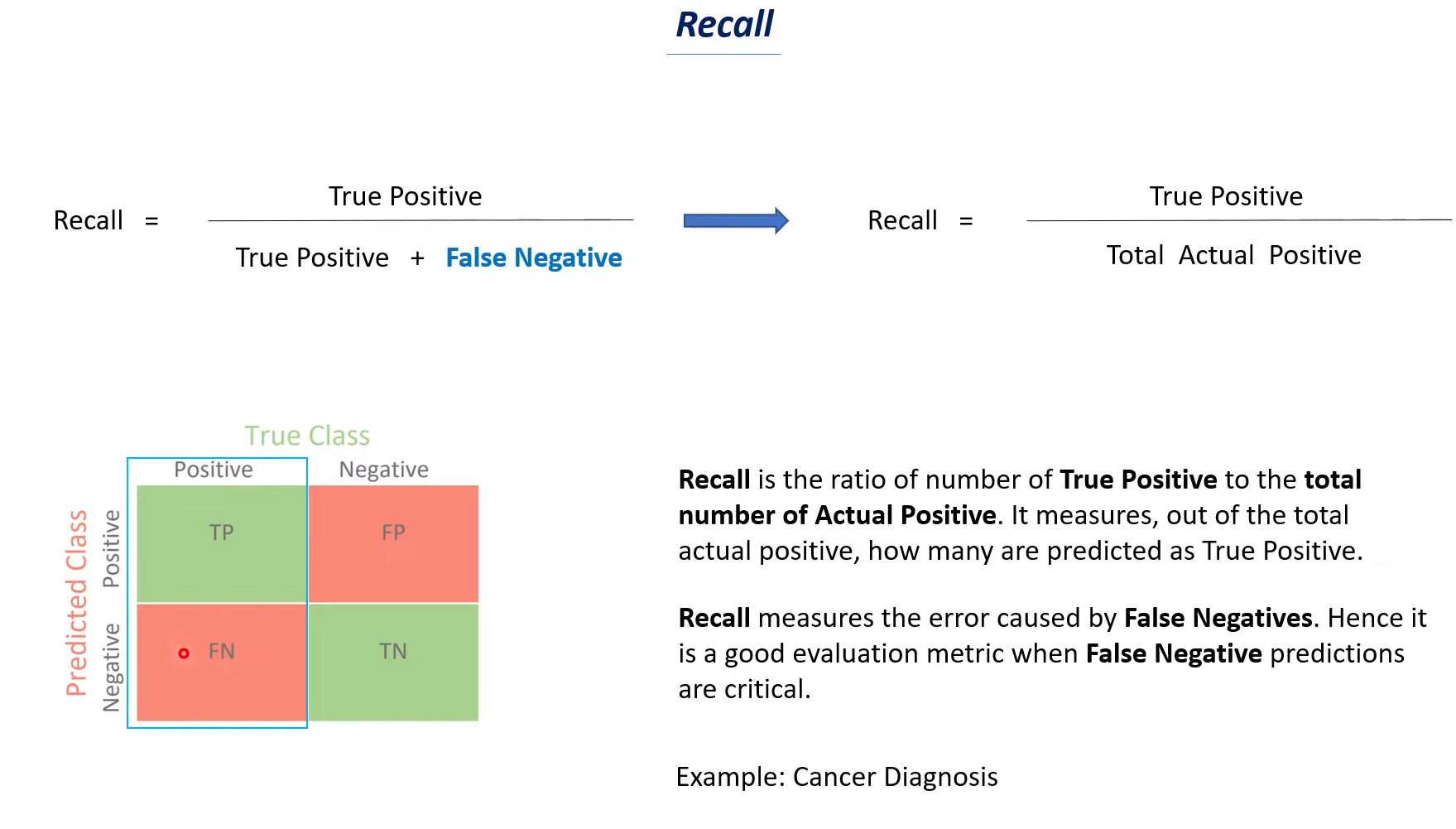
In the case of spam detection, false negatives are generally less of a concern compared to false positives. If a spam message is incorrectly classified as not spam (a false negative), users have the option to manually tag that specific email as spam, thus correcting the misclassification.

**Recall:**

The recall is particularly useful in situations where it is important to **minimize** **“false negatives”**. For example, in medical diagnostics, it is crucial to identify as many true positive cases as possible, even if it results in a higher number of false positives. In such cases, recall serves as a valuable metric to evaluate the model's ability to identify positive instances.

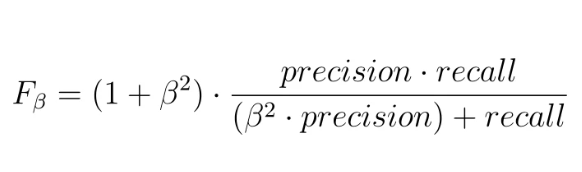
High recall means that the model is effective when we want to minimize **“false negatives”**.





**F-measure:**

If both True positive and False Negative is important.



**Example of Precision Recall and F1 Score:**

