Table of contents:

1. Introduction with an example
2. Confusion Matrix
3. Precision
4. Recall or Sensitivity
5. Specificity
6. How to remember ?
7. Let us have a small test !

Prerequisites:

Introduction with an example

All of the workers at an industry are undergoing a machine learning, primary diabetes scan.

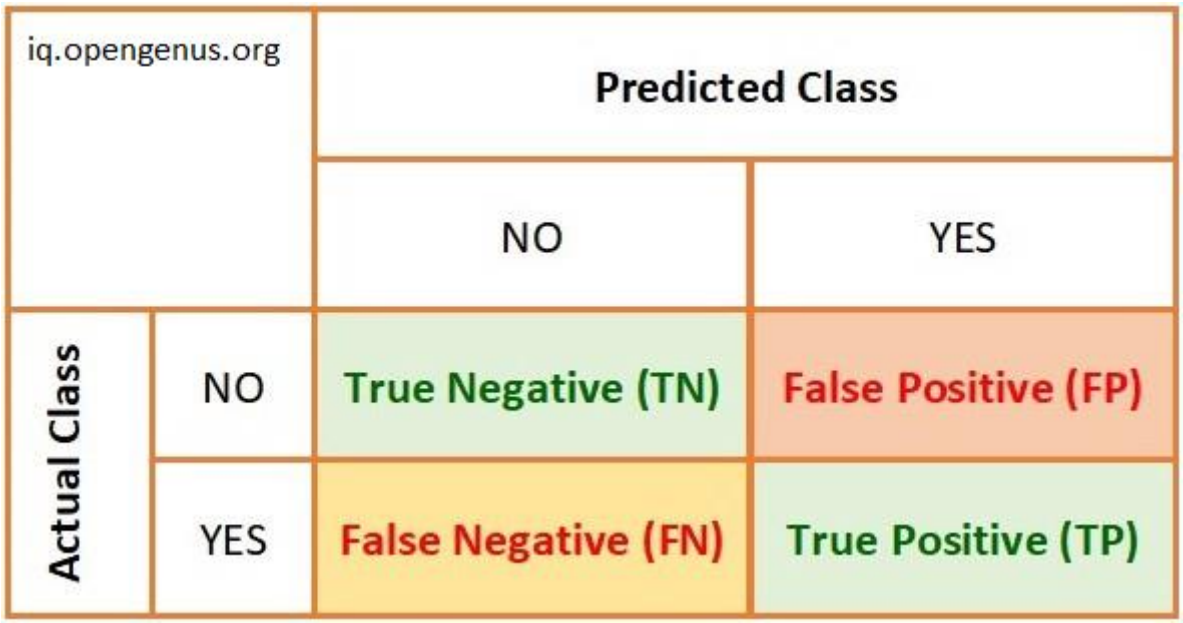
The majority of these tasks are of classification. This is especially true in binary classification. The output is always Boolean, indicating it is either True or False.

The output is either **Diabetic (+ve or True) or healthy (-ve or False).** There are only four possible outcomes for any worker X.

- True positive (TP): Prediction is +ve and X is diabetic, Hit, this is what we desire.
- True negative (TN): Prediction is -ve and X is healthy, Correct Rejection, this is what we desire too.
- False positive (FP): Prediction is +ve and X is healthy, false alarm, bad, Over-Estimation (Type I error).
- False negative (FN): Prediction is -ve and X is diabetic, miss, the worst,Under-Estimation (Type II error).

Confusion Matrix

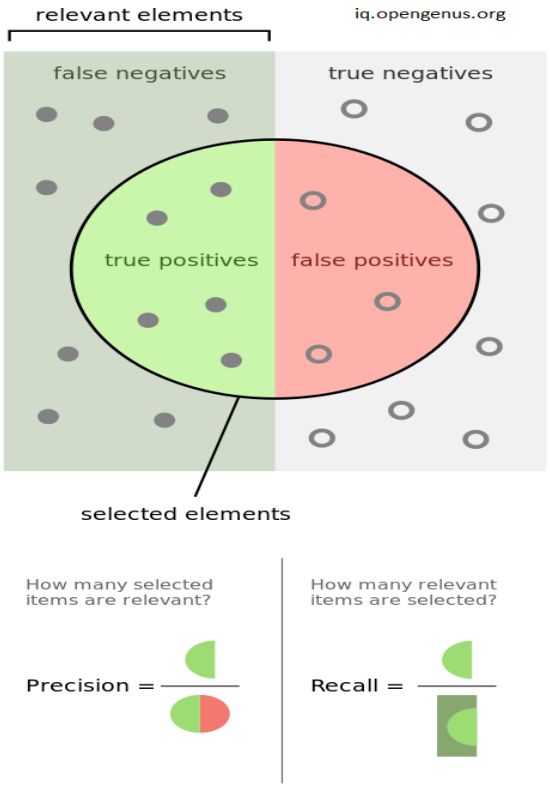
Confusion matrix is an easy-to-understand cross-tab of actual and predicted class values. It contains the total number of data points that fall in each category.



So, How we determine that our Classification task in good ?

The four categories enable us in determining the classification's quality ->

1. Precision
2. Recall
3. Sensitivity
4. Specificity



Precision

Precision is the Ratio of true positives to total predicted positives.

Precision = TP / (TP + FP)
Numerator: +ve diabetes workers.
Denominator: Our algorithm recognised all +ve diabetes workers, whether or not they are diabetic in reality.

What Precision tells us ?

How many of individuals we forecasted as diabetes are truly diabetic? Precision offers us the answer to this question.

PRECISION	USED	IMPORTANT
When	When the occurrence of false positives is Unacceptable.	When we want to be more confident of your predicted positives.
Example	Spam emails. You'd rather have some spam emails in your inbox than miss out some regular emails that were incorrectly sent to your spam box.	

Recall or Sensitivity

Recall or Sensitivity is the Ratio of true positives to total (actual) positives in the data.

Recall and Sensitivity are one and the same.

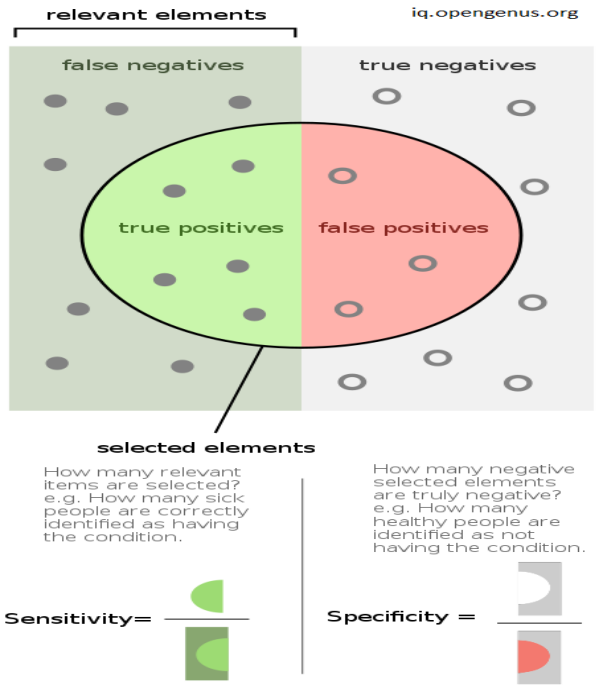
Recall = TP / (TP + FN)
Numerator: +ve labeled diabetic people.
Denominator: All diabetic workers, whether or not our algorithm has identified them.

What Recall or Sensitivity tells us ?

Recall answer to this question, Of all the workers who are diabetic, how many of them did we properly predict?

RECALL	USED	IMPORTANT
When	When the occurrence of false negatives is unacceptable.	When we want to identifying the positives is crucial.
Example	When predicting financial default or a deadly disease or Security checks in airports.	More false positives than fewer false negatives.

Specificity



Specificity is the Ratio of true negatives to total negatives in the data.
Specificity is the correctly -ve labeled by the program to all who are healthy in reality.

Specificity = TN / (TN + FP)
Numerator: -ve labeled healthy worker.
Denominator: All workers who are healthy in actuality, regardless of whether they are classed as +ve or -ve.

What Specificity tells us ?

How many people who are healthy did we accurately predict? Specificity offers us the answer to this question.

RECALL	USED	IMPORTANT
When	When you don't want to frighten people with misleading information.	When you wish to cover all areas where negative categorization is a top priority.
Example	A drug test in which all people who test positive will immediately go to jail or Diagnosing for a health condition before treatment.	

How to remember ?

Many cannot remember the difference between sensitivity, specificity, precision, accuracy, and recall, despite having encountered these phrases multiple times. These are very basic terms, but the names are unintuitive, thus many keep getting them mixed up. What's a decent approach to think about these ideas such that the names make sense?

For precision and recall, each is the true positive (TP) as the numerator divided by a different denominator.

Precision and Recall: focus on True Positives (TP).

- **P**recision: TP / **P**redicted positive
- **R**ecall: TP / **R**eal positive

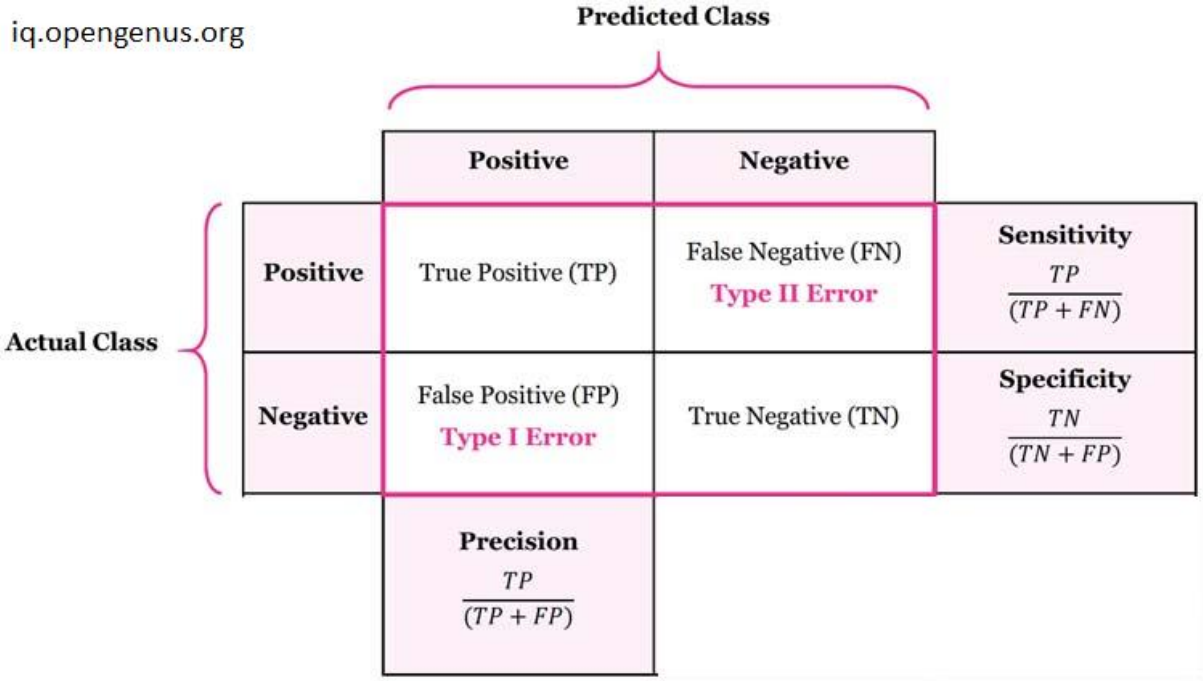
Sensitivity and Specificity: focus on Correct Predictions.
There is one concept viz., **SNIP SPIN.**

- SNIP (**S**e**N**sitivity **I**s **P**ositive): TP / (TP + FN)
- SPIN (**SP**ecificity **I**s **N**egative): TN / (TN + FP)

SNIP refers to Sensitivity.
SPIN refers to Specificity.

Let us Summarize now

iq.opengenus.org



Logistic Regression

Logistic Regression

It is a predictive algorithm using independent variables to predict the dependent variable, just like Linear Regression, but with a difference that the dependent variable should be categorical variable.

Independent variables can be numeric or categorical variables, but the dependent variable will always be categorical

Logistic regression is a statistical model that uses Logistic function to model the conditional probability.

For binary regression, we calculate the conditional probability of the dependent variable Y, given independent variable X

It can be written as ***P(Y=1|X) or P(Y=0|X)***

This is read as the conditional probability of Y=1, given X or conditional probability of Y=0, given X.

P(Y |X) is approximated as a sigmoid function applied to a linear combination of input features

An example of logistic regression can be to find if a person will default their credit card payment or not. The probability of a person

defaulting their credit card payment can be based on the pending credit card balance and income etc.

hence, we can write ***P(default=yes|balance)***

when the ***P(default=yes)≥0.5,*** then we say the person will default their payment.

When the ***P(default=yes)<0.4,*** then we say the person will NOT default their payment.

The probability will always range between 0 and 1. In the case of binary classification, the probability of defaulting payment and not defaulting payment will sum up to 1

$$P(default=yes \mid balance) + P(default=no \mid balance) = 1$$

Logistic Regression can be used for binary classification or multi-class classification.

Binary classification is when we have two possible outcomes like a person is infected with COVID-19 or is not infected with COVID-19. In multi-class classification, we have multiple outcomes like the person may have the flu or an allergy, or cold or COVID-19.

Assumptions for Logistic Regression

- No outliers in the data. An outlier can be identified by analyzing the independent variables

- No correlation (multi-collinearity) between the independent variables.

Logistic Regression function

Logistic regression uses logit function, also referred to as log-odds; it is the logarithm of odds. The odds ratio is the ratio of odds of an event A in the presence of the event B and the odds of event A in the absence of event B.

$$\ln\left(\frac{P}{1-P}\right) = \theta_1 + \theta_2 x$$

$$\frac{P}{1-P} = e^{\theta_1+\theta_2 x}$$

$$P = \frac{1}{1+e^{-}}(\theta_1 + \theta_2 x)$$

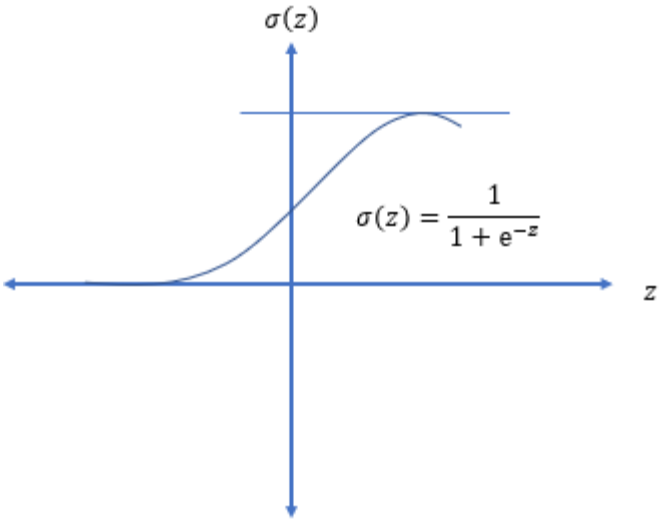
$$\sigma(z) = \frac{1}{1+e^{-z}} \quad \text{Where} \quad z = \theta^T x$$

$$\theta^T \mathbf{x} = \sum_{i=1}^{m} \theta_i x_i = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_m x_m$$

logit or logistic function

- P is the probability that event Y occurs. P(Y=1)
- P/(1-P) is the odds ratio
- θ is a parameters of length m

Logit function estimates probabilities between 0 and 1, and hence logistic regression is a non-linear transformation that looks like S-function shown below.



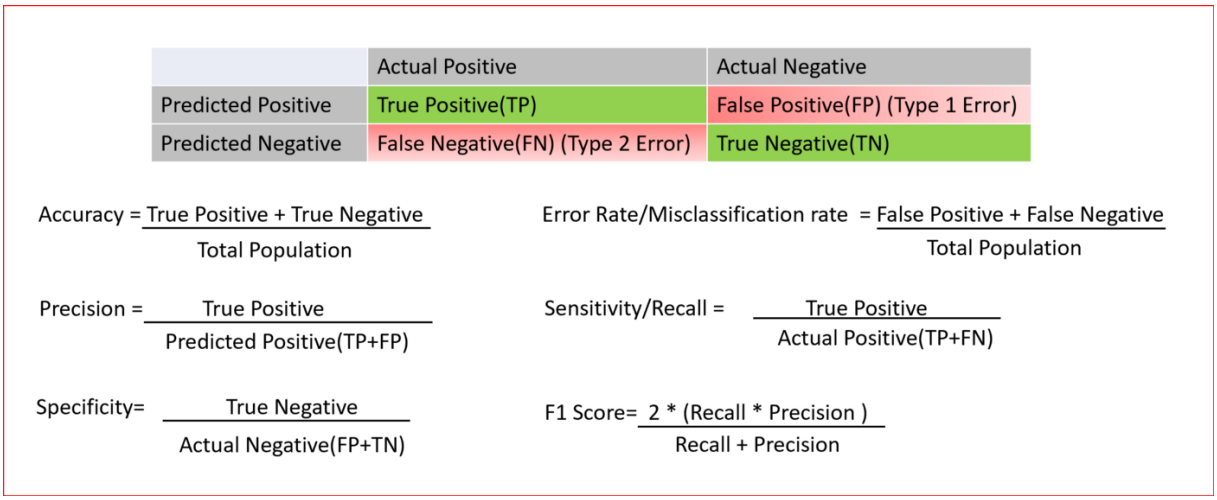
Logistic Regression function

The parameter “θ” of the logistic function can be estimated using the maximum likelihood estimation(MLE) framework. MLE searches for the parameters that best fit the joint probability if the independent variables X.

MLE will give us values for parameter “θ” that would maximize the probability close to 1 for the person who would default their payment and a number close to 0 for all individuals who would not default their payment.

Confusion Matrix to evaluating the performance of binary classification

A confusion matrix is a table that tells us how many actual values and predicted values exist for different classes predicted by the model. Also referred to as the **Error matrix.**



Confusion Matrix or Error Matrix along with different metrics