Concept of F-Measure

Contingency Matrix

	Correct Values	Incorrect Values
Selected by Algorithm	True Positive	False Positive
Not Selected by Algorithm	False Negative	True Negative

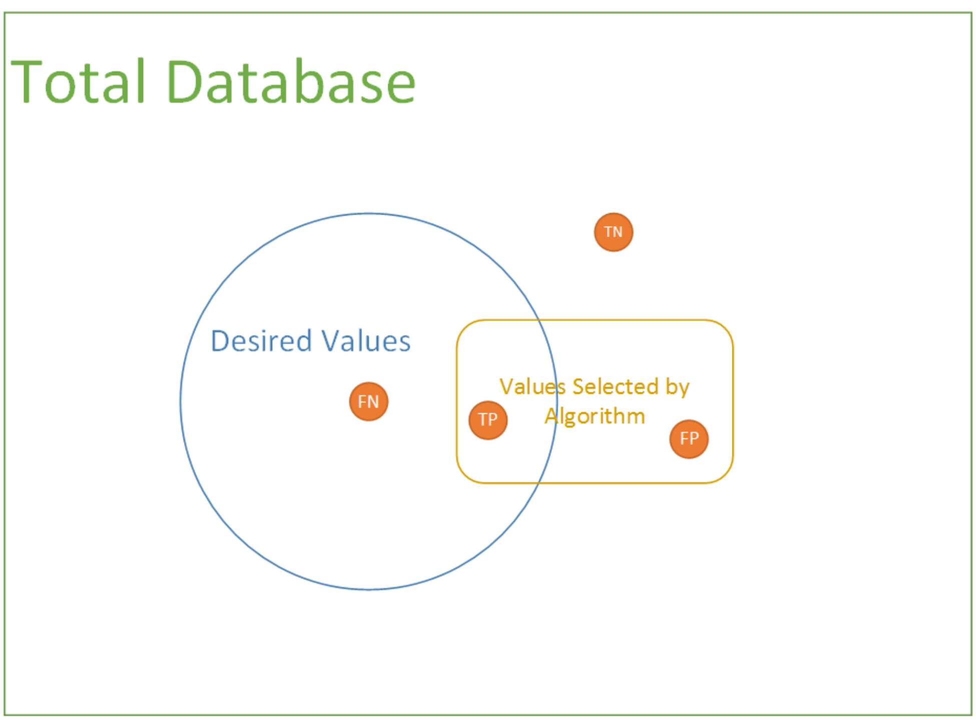
Precision

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

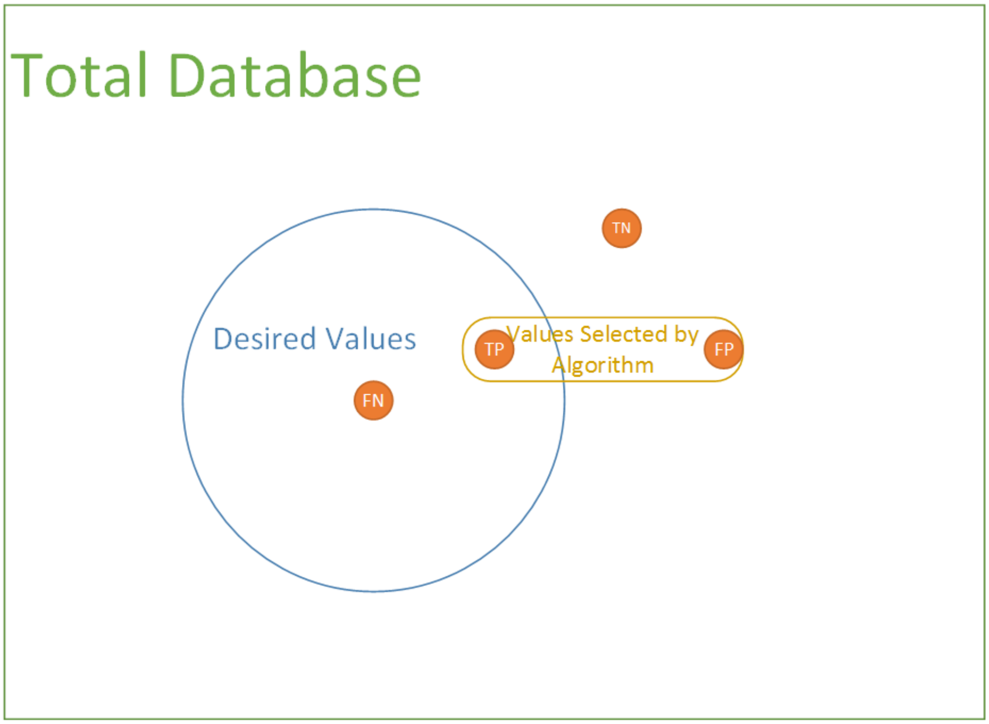
Recall

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

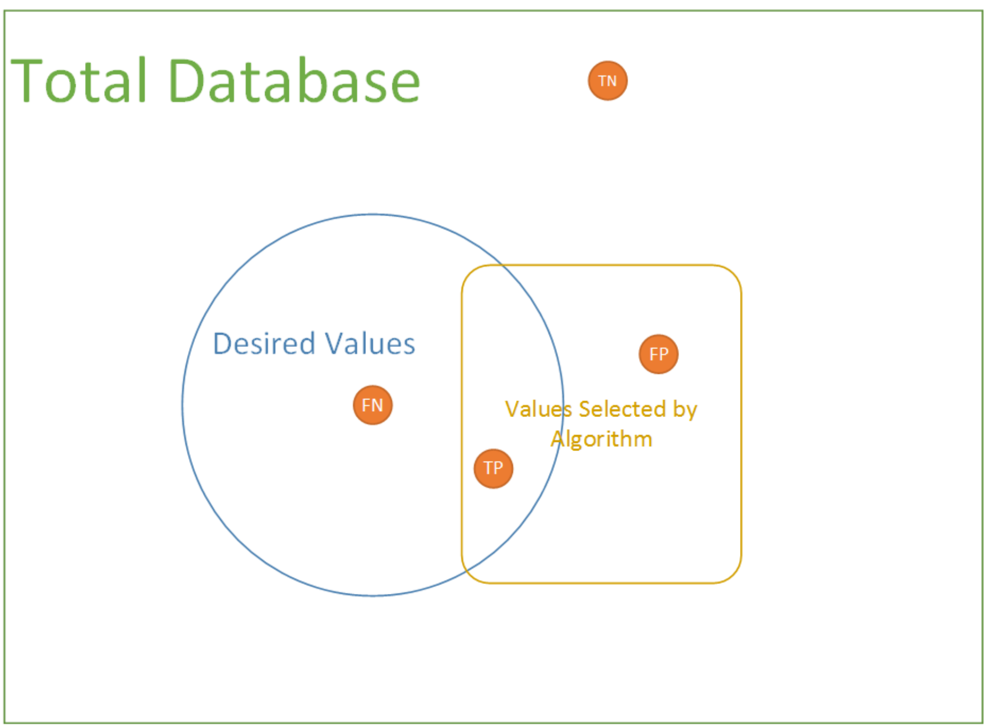
Sample Data Set



By reducing the actual number of values selected by algorithm we reduce the value of False positives to a greater extent than that of false negative thus Recall decreases whereas Precision increases.



On the other hand if the algorithm gathers more values from dataset it ends up gathering more false positive too hence Recall increases at the cost of precision as Precision decreases



F Measure

Some times we need recall to be high and some times we need a higher precision. But one comes at the cost of other.

So we devise a common scale to measure both of them together as one gets traded off for the other when we try to change any one of them.

And that common scale is called F measure.

F measure is simply a weighted harmonic mean of Precision and Recall.

It is given as

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

Where α and β decide the weightage given to Precision and recall.

If we give equal weightage to both α and β

i.e α = ½ and β $= 1$

We get the formula

$$F = \frac{2PR}{(P + R)}$$

Which is also called F1 Measure

Application of F1 Measure in our Algorithm

For applying F1 Measure to our algorithm we consider the following parameters.

We Take our total database to be the total number of matched features.

We assume our desired values to be the features which are exactly present onto the object of consideration.

We take the values selected by the algorithm to be the features that the K means algorithm sorted out to be on the object.