**PERFORMANCE METRICS**

Accuracy = (TP+TN)/(TP + TN + FP+FN)

Sensitivity / Recall = TP /(TP+FN) = True Positive Rate = 1- FNR = 1- Type 2 error = %age of actual positives predicted correctly. High sensitivity indicates low FNR or low Type 2 error

When sensitivity is of higher importance, then we want to reduce the threshold to increase the TPR and reduce the FNR

Specificity = TN/(TN + FP) = True Negative Rate = 1-FPR = 1- Type 1 error = %age of actual negatives predicted correctly. High specificity indicates low FPR or low Type 1 error.

When specificity is more important than sensitivity then we are going to increase the cutoff to increase the TNR and reduce the FPR

FNR = same as Type 2 error = %age of actual positives predicted incorrectly

FPR = same as Type 1 error = %age of actual negatives predicted incorrectly

**PRECISION RECALL AND F1 SCORE**

Precision = TP/(TP+FP) – also called as +ve prediction rate – This indicates how precise a model is in it’s positive predictions i.e how many of the positive predictions made are correct. High precision indicates low type 1 error .  
Recall = TP/(TP+FN) – same as sensitivity – high recall indicates low type 2 error.

In both precision and recall, importance has been given to correct prediction of +ves. In addition it talks of which error I am receptive towards? FPR or FNR. In telecom case we want to avoid FNR

F1 score: 2/(1/Precision + 1/Recall)

NPR – Negative prediction Rate = TN/(TN+FN) – i.e what %age of –ve predictions are correct.

**Precision** helps when the costs of false positives are high. So let’s assume the problem involves the detection of skin cancer. If we have a model that has very low precision, then many patients will be told that they have melanoma, and that will include some misdiagnoses. Lots of extra tests and stress are at stake. When false positives are too high, those who monitor the results will learn to ignore them after being bombarded with false alarms.

**Recall** helps when the cost of false negatives is high. What if we need to detect incoming nuclear missiles? A false negative has devastating consequences. Get it wrong and we all die. When false negatives are frequent, you get hit by the thing you want to avoid. A false negative is when you decide to ignore the sound of a twig breaking in a dark forest, and you get eaten by a bear. (A false positive is staying up all night sleepless in your tent in a cold sweat listening to every shuffle in the forest, only to realize the next morning that those sounds were made by a chipmunk. Not fun.) If you had a model that let in nuclear missiles by mistake, you would want to throw it out. If you had a model that kept you awake all night because *chipmunks*, you would want to throw it out, too. If, like most people, you prefer to not get eaten by the bear, and also not stay up all night worried about chipmunk alarms, then you need to optimize for an evaluation metric that’s a combined measure of precision and recall. Enter the F1 score…

**F1 Score**

f1 formula

F1 is an overall measure of a model’s accuracy that combines precision and recall, in that weird way that addition and multiplication just mix two ingredients to make a separate dish altogether. That is, a good F1 score means that you have low false positives and low false negatives, so you’re correctly identifying real threats and you are not disturbed by false alarms. An F1 score is considered perfect when it’s 1, while the model is a total failure when it’s 0.

Remember: All models are wrong, but some are useful. That is, all models will generate some false negatives, some false positives, and possibly both. While you can tune a model to minimize one or the other, you often face a tradeoff, where a decrease in false negatives leads to an increase in false positives, or vice versa. You’ll need to optimize for the performance metrics that are most useful for your specific problem.