Balanced Accuracy and the F1 Score

Machine Learning - Section 2.1.3

Marc Omar Haddad

Created: 17 January, 2020

Updated: 20 January, 2020

As discussed previously, overall accuracy does not usually provide enough information to adequately assess an algorithm; hence the introduction of Sensitivity and Specificity. Indeed, a better (and more concise) metric than overall accuracy would be to find the **average** of Sensitivity and Specificity. This average is known as the **Balanced Accuracy**.

Because Sensitivity and Specificity are rates, it is more appropriate to compute their *harmonic average*, like so:

$$\frac{1}{\frac{1}{2}\left(\frac{1}{\text{recall}} + \frac{1}{\text{precision}}\right)}$$

Remember: **Recall** $= \frac{TP}{(TP+FN)}$, **Precision** $= \frac{TP}{(TP+FP)}$

The harmonic average above is also known as the **F1 Score**, and can be rewritten (for the sake of simplicity) like so:

$$2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Other Considerations

Different contexts call for the prioritization to minimize some errors over others (e.g. preferring higher Sensitivity at the cost of lower Specificity and vice versa).

Example: When it comes to plane safety, it is more important to maximize Sensitivity over Specificity. We assume **"Faulty"** to be the **"positive"** outcome:

- The cost of grounding a plane that is *predicted* to be faulty, but is *actually* not faulty (i.e. **False Positive**), is much less costly than *predicting* a plane to be not faulty, but which is *actually* faulty and results in a crash (i.e. **False Negative**).

Conversely, when it comes to capital murder cases, it is more important to maximize Specificity over Sensitivity. We assume **"Guilty"** to be the **"positive"** outcome:

- When determining guiltiness, the cost of *predicting* a person to be not guilty when in fact they are *actually* guilty (i.e. **False Negative**), is much less costly than *predicting* a person to be guilty and sentencing them to death despite *actually* being not guilty (i.e. **False Positive**).

Weighted F1 Score and `F_meas`

Depending on the context, the F1 Score can be weighed accordingly. We use β to denote how much more important Sensitivity is to Specificity. Our weighted F1 Score formula looks like this:

$$\frac{1}{\frac{\beta^2}{1+\beta^2}\frac{1}{\text{recall}} + \frac{1}{1+\beta^2}\frac{1}{\text{precision}}}$$

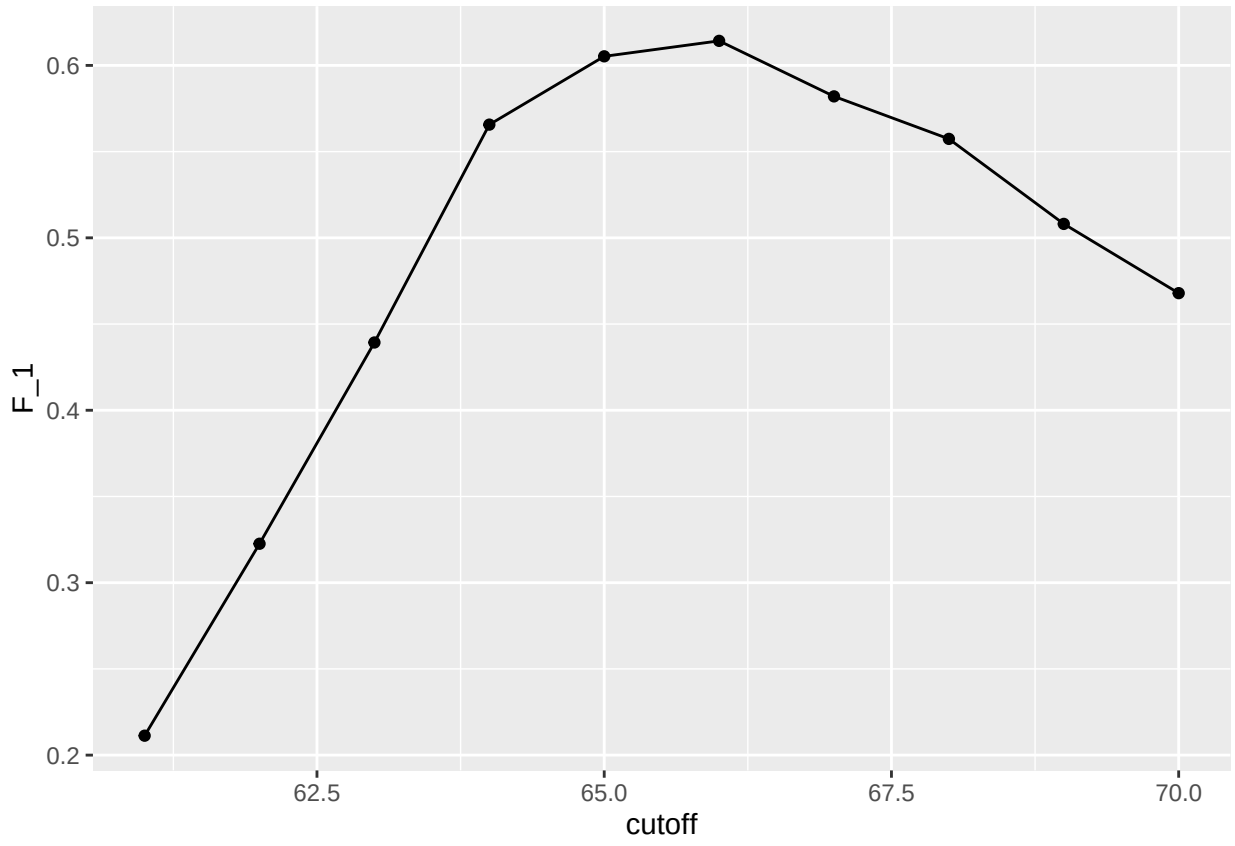
In the `caret` package, the function `F_meas` returns a summary of the F1 Score, with β defaulting to 1.

Knowing this, we can now reconstruct our earlier predictive algorithm (see Sections 2.1.1 and 2.1.2) to **maximize our F1 Score**:

```
cutoff = seq(61, 70)
F_1 = map_dbl(cutoff, function(x) {
  y_hat = ifelse(train_set$height > x, "Male", "Female") %>%
    factor(levels = levels(test_set$sex))
  F_meas(data = y_hat, reference = train_set$sex)
})
F_1
```

```
##  [1] 0.211 0.323 0.439 0.566 0.605 0.614 0.582 0.557 0.508 0.468
```

Let us now compare our F1 Score values with their respective cutoffs:



```
max(F_1)
```

```
## [1] 0.614
```

```
best_cutoff = cutoff[which.max(F_1)]
best_cutoff
```

```
## [1] 66
```

The maximum F1 Score is 0.614, and is achieved by using a cutoff of 66 inches. This is a more reasonable estimate than our previously obtained best cutoff of 64 inches. Furthermore, if we analyze our new confusion matrix we can see that our Sensitivity and Specificity values are more balanced:

```
y_hat = ifelse(test_set$height > best_cutoff, "Male", "Female") %>%
  factor(levels = levels(test_set$sex))
confusionMatrix(data = y_hat, reference = test_set$sex)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Female Male
##     Female     81   67
##     Male       38  339
##
##                Accuracy : 0.8
##                  95% CI : (0.763, 0.833)
##     No Information Rate : 0.773
##     P-Value [Acc > NIR] : 0.07819
##
##                   Kappa : 0.475
##
##  Mcnemar's Test P-Value : 0.00629
##
##             Sensitivity : 0.681
##             Specificity : 0.835
##          Pos Pred Value : 0.547
##          Neg Pred Value : 0.899
##              Prevalence : 0.227
##          Detection Rate : 0.154
##    Detection Prevalence : 0.282
##       Balanced Accuracy : 0.758
##
##        'Positive' Class : Female
##
```