How do you calculate the F1 score in Python, and could you provide an example?

Authored by

stats writer

May 7, 2024

The F1 score is a commonly used metric for evaluating the performance of a machine learning model. It combines precision and recall into a single score, providing a balanced measure of a model's accuracy. In Python, the F1 score can be calculated using the sklearn library's f1_score function, which takes in the model's predicted values and the actual values as parameters. An example of calculating the F1 score in Python would be:

```
from sklearn.metrics import f1_score

predicted =
actual =

f1 = f1_score(actual, predicted)

print(f1) # Output: 0.5
```

In this example, the F1 score is 0.5, indicating that the model's precision and recall are both equally important in evaluating its performance. Overall, the F1 score is a useful tool for assessing the effectiveness of a model and can be easily calculated in Python using the f1_score function.

Calculate F1 Score in Python (Including Example)

When using in machine learning, a common metric that we use to assess the quality of the model is the F1 Score.

This metric is calculated as:

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

where:

Precision: Correct positive predictions relative to total positive predictionsRecall: Correct positive predictions

relative to total actual positives

For example, suppose we use a logistic regression model to predict whether or not 400 different college basketball players get drafted into the NBA.

The following confusion matrix summarizes the predictions made by the model:

		Predicted	
		Drafted = Yes	**Drafted = No**
Actual	**Drafted = Yes**	**120** (True Positive)	**40** (False Negative)
	Drafted = No	**70** (False positive)	**170** (True Negative)

Here is how to calculate the F1 score of the model:

Precision = True Positive / (True Positive + False Positive) = 120/ (120+70) = .63157

Recall = True Positive / (True Positive + False Negative) = 120 / (120+40) = .75

F1 Score = 2 * (.63157 * .75) / (.63157 + .75) = .6857

The following example shows how to calculate the F1

score for this exact model in Python.

Example: Calculating F1 Score in Python

The following code shows how to use the f1_score() function from the sklearn package in Python to calculate the F1 score for a given array of predicted values and actual values.

```
import numpy as np
from sklearn.metricsimport f1_score

#define array of actual classes
actual = np.repeat(, repeats=)

#define array of predicted classes
pred = np.repeat(, repeats=)

#calculate F1 score
f1_score(actual, pred)

0.6857142857142857
```

We can see that the F1 score is 0.6857. This matches the value that we calculated earlier by hand.

Notes on Using F1 Scores

If you use F1 score to compare several models, the model with the highest F1 score represents the model that is best able to classify observations into classes.

For example, if you fit another logistic regression model to the data and that model has an F1 score of 0.75, that model would be considered better since it has a higher F1 score.

How to Calculate Balanced Accuracy in Python