

# How can the Matthews Correlation Coefficient be calculated in Python?

Authored by  
**stats writer**

May 13, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can the Matthews Correlation Coefficient be calculated in Python?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=144077>

The Matthews Correlation Coefficient (MCC) is a measure of the performance of binary classification models that takes into consideration the balance between true positives, true negatives, false positives, and false negatives. In order to calculate the MCC in Python, one must first obtain the confusion matrix of the model's predictions. This can be achieved by using the `sklearn.metrics` module and its function `"confusion_matrix"`. Then, the MCC can be calculated using the formula  $(TP*TN - FP*FN)/\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}$ . This calculation can be easily implemented in Python using the `numpy` library. It is important to note that the MCC ranges from -1 to 1, where a value of 1 represents a perfect prediction, 0 represents a random prediction, and -1 represents a completely wrong prediction. By using the MCC, one can evaluate the overall performance of a binary classification model in a more accurate and balanced way.

## Calculate Matthews Correlation Coefficient in Python

**Matthews correlation coefficient (MCC) is a metric we can use to assess the performance of a .**

**It is calculated as:**

$$\text{MCC} = \frac{(TP*TN - FP*FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

**where:**

**TP: Number of true positives**  
**TN: Number of true negatives**  
**FP: Number of false positives**  
**FN: Number of false negatives**

**This metric is particularly useful when the two classes are imbalanced - that is, one class appears much more than the other.**

The value for MCC ranges from -1 to 1 where:

-1 indicates total disagreement between predicted classes and actual classes  
0 is synonymous with completely random guessing  
1 indicates total agreement between predicted classes and actual classes

For example, suppose a sports analyst uses a 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	15 (True Positive)	5 (False Negative)
	Drafted = No	5 (False positive)	375 (True Negative)

To calculate the MCC of the model, we can use the following formula:

$$\text{MCC} = \frac{(\text{TP} * \text{TN}) - (\text{FP} * \text{FN})}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}$$

$$\text{MCC} = \frac{(15 * 375 - 5 * 5)}{\sqrt{(15 + 5)(15 + 5)(375 + 5)(375 + 5)}} = 0.7368$$

**Matthews correlation coefficient turns out to be 0.7368. This value is somewhat close to one, which indicates that the model does a decent job of predicting whether or not players will get drafted.**

**The following example shows how to calculate MCC for this exact scenario using the `matthews_corrcoef()` function from the `sklearn` library in Python.**

**Example: Calculating Matthews Correlation Coefficient in Python**

**The following code shows how to define an array of predicted classes and an array of actual classes, then calculate Matthews correlation coefficient of a model in Python:**

```
import numpy as np
from sklearn.metrics import matthews_corrcoef

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

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

#calculate Matthews correlation coefficient
```

```
matthews_corrcoef(actual, pred)
```

```
0.7368421052631579
```

**The MCC is 0.7368. This matches the value that we calculated earlier by hand.**

**Note: You can find the complete documentation for the `matthews_corrcoef()` function .**

**The following tutorials explain how to calculate other common metrics for classification models in Python:**