To use accuracy\_score, precision\_score, and recall\_score from sklearn.metrics to evaluate your model, you can modify the evaluation section of the example code. Here is how you can integrate these metrics

**Make predictions and evaluate the model**:

After making predictions on the test set, you can compute and print the accuracy, precision, and recall

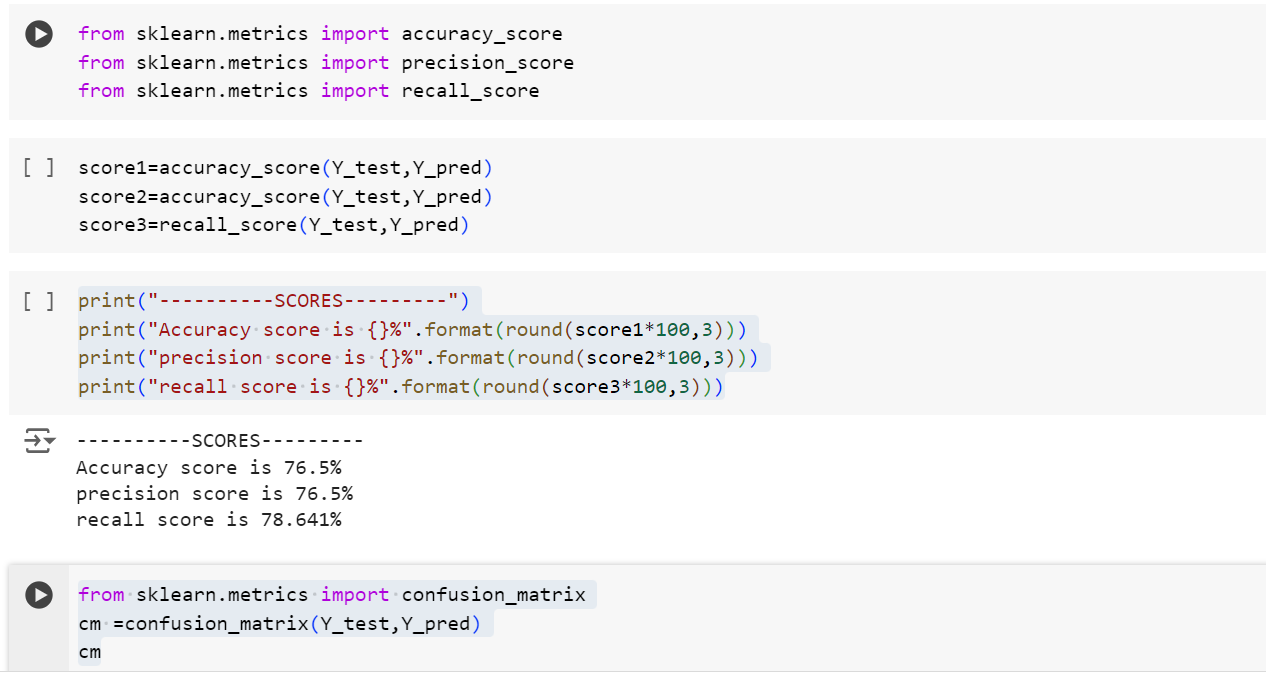
 Train the MultinomialNB model on the training data.

 Make predictions on the test data.

 Calculate the accuracy, precision, and recall scores.

 Print the scores formatted as percentages with three decimal places.

 Print a detailed classification report for further evaluation.



Your code attempts to find the best alpha value for the MultinomialNB classifier by testing different alpha values and recording the accuracy. However, there's a small mistake in the code where it computes the accuracy score; it should use temp\_y\_pred instead of Y\_pred. Additionally, to use np.arange, you need to import NumPy

trained a MultinomialNB classifier with a specified alpha value of 0.2. If you want to check the accuracy of this classifier, you can predict on the test data and evaluate it