MOA – NAVIE BAYES.

JAVA CODE:

import moa.core.InstanceExample;

import moa.core.Example;

import moa.classifiers.Classifier;

import moa.classifiers.bayes.NaiveBayes;

import moa.evaluation.BasicClassificationPerformanceEvaluator;

import moa.streams.ArffFileStream;

import com.yahoo.labs.samoa.instances.Instance;

public class NaiveBayesExample {

public static void main(String[] args) {SssssS

String arffFilePath = "C:\\Program Files\\Weka-3-9-6\\data\\iris.arff";

ArffFileStream stream = new ArffFileStream(arffFilePath, -1);

stream.prepareForUse();

Classifier classifier = new NaiveBayes();

classifier.setModelContext(stream.getHeader());

classifier.prepareForUse();

BasicClassificationPerformanceEvaluator evaluator = new BasicClassificationPerformanceEvaluator();

int numberOfInstances = 150;

for (int i = 0; i < numberOfInstances && stream.hasMoreInstances(); i++) {

Instance instance = stream.nextInstance().getData();

Example<Instance> example = new InstanceExample(instance);

classifier.trainOnInstance(example);

evaluator.addResult(example, classifier.getVotesForInstance(instance));

if (i > 0 && i % 50 == 0) {

System.out.println("Processed " + i + " instances.");

}

}

System.out.println("Accuracy: " + evaluator.getFractionCorrectlyClassified() \* 100 + "%");

System.out.println("Kappa Statistic: " + evaluator.getKappaStatistic());

System.out.println("Kappa Temporal Statistic: " + evaluator.getKappaTemporalStatistic());

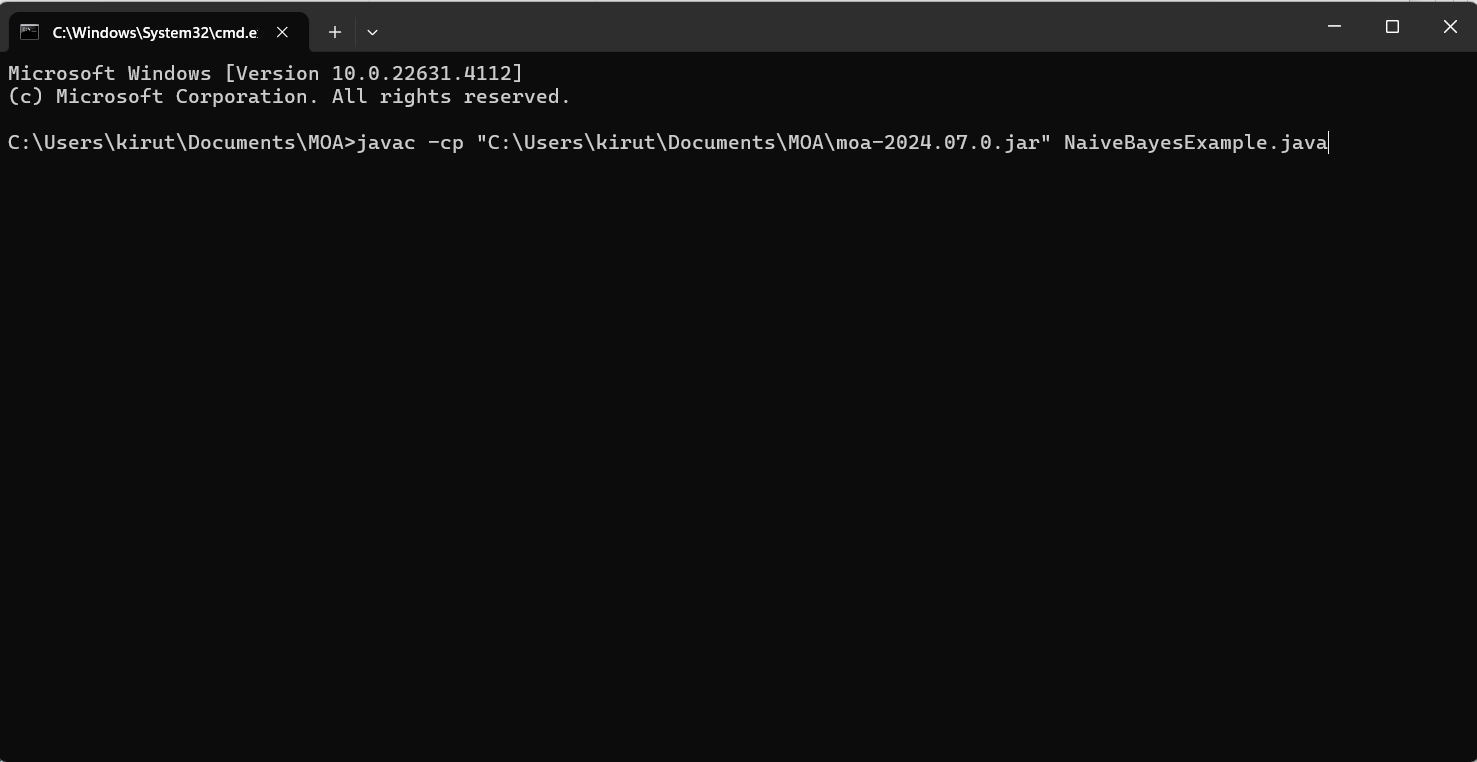
}

}

Output:

To compile:

javac -cp "C:\Users\kirut\Documents\MOA\moa-2024.07.0.jar" NaiveBayesExample.java



To run:

java -cp ".;C:\Users\kirut\Documents\MOA\moa-2024.07.0.jar" NaiveBayesExample

Output:

