**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Лабораторная работа №5

«Реализация алгоритма гауссовского байесовского классификатора на Python»

Выполнил: Проверил:

Студент ИУ5-34Б Преподаватель каф. ИУ5

Бурдуковский И.О. Гапанюк Ю. Е.

Подпись и дата: Подпись и дата:

Москва, 2023 г.

Задание

Реализация алгоритма гауссовского байесовского классификатора на Python.

Код программы

import numpy as np  
from scipy.stats import multivariate\_normal  
  
  
class GaussianBayesianClassifier:  
 def fit(self, X, y):  
 self.classes = np.unique(y)  
 self.class\_priors = {}  
 self.mean\_vectors = {}  
 self.cov\_matrices = {}  
 for c in self.classes:  
 X\_c = X[y == c]  
 self.class\_priors[c] = len(X\_c) / len(X)  
 self.mean\_vectors[c] = np.mean(X\_c, axis=0)  
 self.cov\_matrices[c] = np.cov(X\_c, rowvar=False)  
  
 def predict(self, X):  
 predictions = []  
 for x in X:  
 posteriors = []  
 for c in self.classes:  
 prior = self.class\_priors[c]  
 mean = self.mean\_vectors[c]  
 cov = self.cov\_matrices[c]  
 likelihood = multivariate\_normal(mean=mean, cov=cov).pdf(x)  
 posterior = prior \* likelihood  
 posteriors.append(posterior)  
 predictions.append(np.argmax(posteriors))  
 return np.array(predictions)  
  
  
X\_train = np.array([[1, 2], [2, 3], [3, 4], [4, 5], [1, 3], [2, 4]])  
y\_train = np.array([0, 0, 1, 1, 0, 1])  
  
classifier = GaussianBayesianClassifier()  
classifier.fit(X\_train, y\_train)  
  
X\_test = np.array([[1.5, 2.5], [3.5, 4.5]])  
predictions = classifier.predict(X\_test)  
print(predictions)

Результаты

