Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

По дисциплине: “Основы системного анализа”

Лабораторная работа №5

“Метод Монте-Карло”

Вариант 2

Выполнил:

ст.гр. ИС/б-22

Волобуев Ю. С.

Проверил:

Гончаренко Д. Г.

Севастополь

2019

1 ЦЕЛЬ РАБОТЫ

Углубление теоретических знаний в области системного анализа, ознакомление с методом Монте-Карло.

2 ПОСТАНОВКА ЗАДАЧИ

1. Написать программу на языке программирования python для вычисления площади под кривой методом Монте-Карло.
2. Построить график зависимости точности результата от числа испытаний.
3. Дополнительное задание: написать программу на языке программирования python для визуального отображения результатов решения (см. рисунок 1).



Рисунок 1 – Задание по варианту

3 ХОД РАБОТЫ

1. Код программы на языке программирования python для вычисления площади под кривой методом Монте-Карло*:*

import numpy as np  
import matplotlib.pyplot as plt  
import random  
  
def generate\_point(xstart, xend, ystart, yend):  
 x = random.uniform(xstart, xend)  
 y = random.uniform(ystart, yend)  
 return x, y  
  
def integrate(xstart, xend, ystart, yend, function, points\_number):  
 points\_within = []  
 points\_beyond = []  
 within = 0  
 for i in range(points\_number):  
 x, y = generate\_point(xstart, xend, ystart, yend)  
 if (y <= function(x) and y > 0):  
 points\_within.append((x, y))  
 within += 1  
 elif (y >= function(x) and y < 0):  
 points\_within.append((x, y))  
 within -= 1  
 else:  
 points\_beyond.append((x, y))  
 integral = (xend - xstart) \* (yend - ystart) \* within / points\_number   
  
 return integral, points\_within, points\_beyond  
   
print('Enter interval start:')  
start = float(input())  
  
print('Enter interval end:')  
end = float(input())  
  
print('Enter number of points:')  
points\_number = int(input())  
  
x = np.linspace(start, end, 100000)  
function = np.cos(x)  
max = max(function)  
if min(function) >= 0:  
 min = 0  
else:  
 min = min(function)  
  
integral, points\_below, points\_above = integrate(start, end, min, max, np.cos, points\_number)  
print('integral value: ', integral)  
  
belowx, belowy = zip(\*points\_below)  
abovex, abovey = zip(\*points\_above)  
plt.scatter(belowx, belowy, color='red')  
plt.scatter(abovex, abovey, color='blue')  
plt.plot(x, function)  
plt.grid()  
plt.show()

2. Результат для 10 точек:

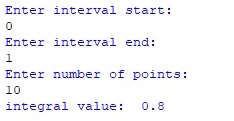


Рисунок 2 – Результат для 10 точек

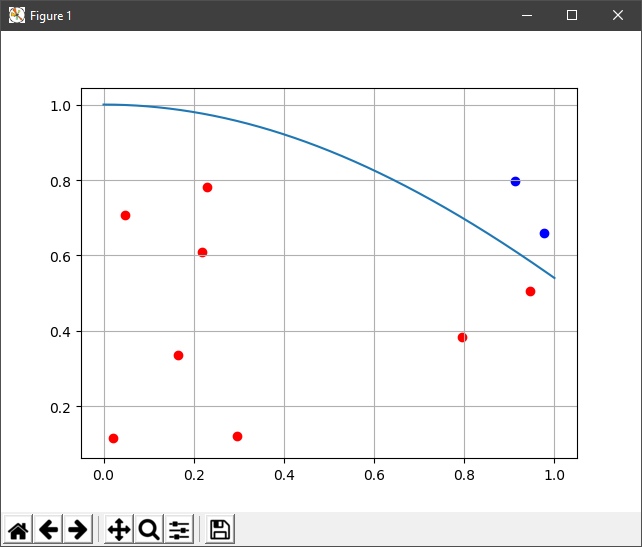


Рисунок 3 – График результатов

Результат для 100 точек:

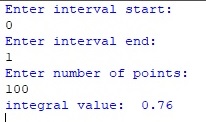


Рисунок 4 – Результат для 100 точек

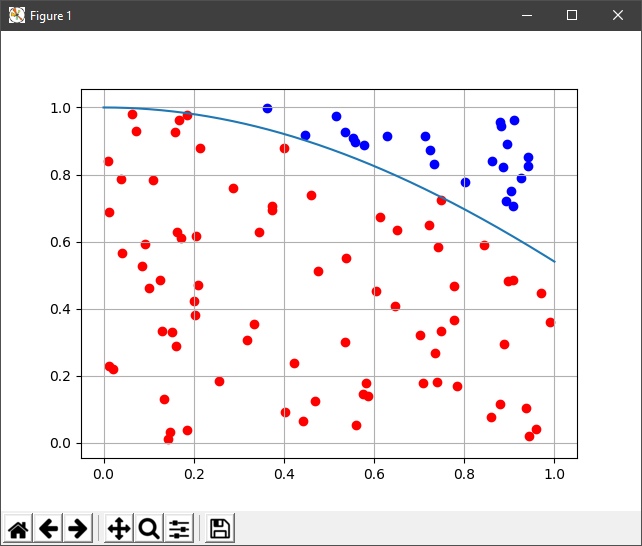


Рисунок 5 – График результатов

Результат для 1000 точек:

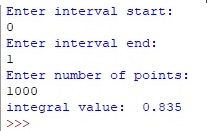


Рисунок 6 – Результат для 1000 точек

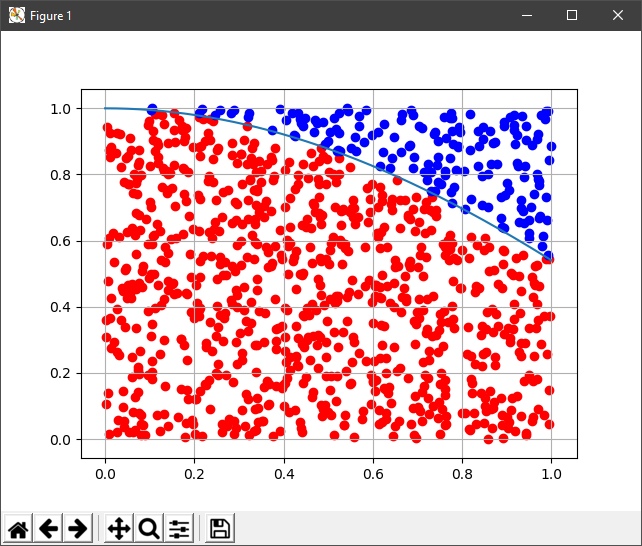


Рисунок 7 – График результатов

Результат для 10000 точек:

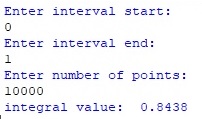


Рисунок 8 – Результат для 10000 точек

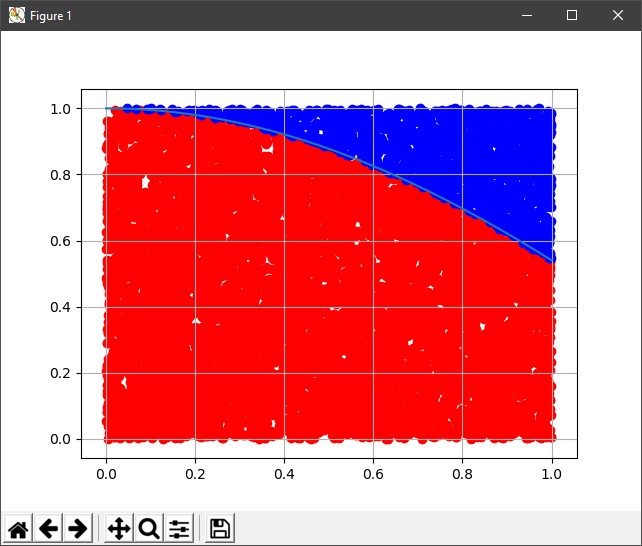


Рисунок 9 – График результатов

ВЫВОД

В ходе выполнения лабораторной работы была написана программа на языке программирования python для вычисления площади под кривой методом Монте-Карло с визуальным отображением результатов решения.