Федеральное государственное бюджетное образовательное учреждение
высшего образования
"МИРЭА – Российский технологический университет"
РТУ МИРЭА

Институт комплексной безопасности и цифровых технологий

Кафедра КБ-14 «Цифровые технологии обработки данных»

Практическая работа

по дисциплине

«Моделирование систем»
наименование дисциплины

Тема практической работы «Основные понятия статистического эксперимента»

Студент группы БСБО-05-21 (учебная группа)
Восоров М.М. (Фамилия И.О.)

Москва 2023 г.

untitled9

October 12, 2023

[2]:
```
#      1       Python
from random import random as r
import random

random.seed(random.randint(-10000, 10000))

def calc_pi(x0, y0, r0, expNmb):
    '''
                      PI                ,
    '''
    positive_res = 0
    r02 = r0**2
    x_max, x_min = x0 - r0, x0 + r0
    y_max, y_min = y0 - r0, y0 + r0
    for k in range(expNmb):
        px, py = r(), r()
        xp = (x_max - x_min) * px + x_min
        yp = (y_max - y_min) * py + y_min
        if (xp-x0) ** 2 + (yp-y0) ** 2 < r02:
            positive_res += 1
    return 4 * positive_res / expNmb
```

[3]:
```
'''
    2
                 1
             ExpNmb = 10**4 (x0 = 1, y0 = 2, r0 = 5).
'''
print(calc_pi(1, 2, 5, 10**4))

'''


        ExpNmb = 104, 105, 106, 107, 108
            seria_1,
                                            seria_2,
seria_3, seria_4, seria_5
'''
def get_serial(seria_number):
```

```
    exp_nmb = 10**4
    x0, y0, r0 = 1, 2, 5
    print(f"Seria number {seria_number} start...")
    res_list = []
    while exp_nmb <= 10**6:
        pi = calc_pi(x0, y0, r0, exp_nmb)
        res_list.append(pi)
        exp_nmb = exp_nmb * 10
    print(f"Seria number {seria_number} end.")
    return res_list

seria_1 = get_serial("1")
seria_2 = get_serial("2")
seria_3 = get_serial("3")
seria_4 = get_serial("4")
seria_5 = get_serial("5")
print(f"seria_1 = {seria_1}")
print(f"seria_1 = {seria_2}")
print(f"seria_1 = {seria_3}")
print(f"seria_1 = {seria_4}")
print(f"seria_1 = {seria_5}")
```

```
3.162
Seria number 1 start…
Seria number 1 end.
Seria number 2 start…
Seria number 2 end.
Seria number 3 start…
Seria number 3 end.
Seria number 4 start…
Seria number 4 end.
Seria number 5 start…
Seria number 5 end.
seria_1 = [3.112, 3.1374, 3.140584]
seria_1 = [3.1588, 3.14332, 3.144952]
seria_1 = [3.1308, 3.14688, 3.145756]
seria_1 = [3.1388, 3.13624, 3.139972]
seria_1 = [3.1652, 3.14324, 3.14138]
```

[5]:
```
'''
    3

math         pi.                          ,

'''
seria_total = [seria_1, seria_2, seria_3, seria_4, seria_5]
from math import pi
```

```python
seria_fault_single = []
for i in range(2):
    fault = []
    for j in range(2):
        fault.append(round(abs(seria_total[i][j] - pi)/pi, 10))
    print(f"Fault {i+1}: {fault}")
    seria_fault_single.append(fault)

seria_fault_total = []
for i in range(2):
    total_fault = 0
    for j in range(2):
        total_fault += seria_fault_single[j] [i]
    total_fault = round(total_fault / 5, 10)
    seria_fault_total.append(total_fault)
print("Total fault: ", seria_fault_total)

'''

'''
```

```
Fault 1: [0.0094196342, 0.0013345631]
Fault 2: [0.0054772685, 0.0005498314]
Total fault:  [0.0029793805, 0.0003768789]
```

Задание 4.

Для начала требуется найти точное значение данного определённого интеграла – оно равно 6. С помощью метода Монте-Карло необходимо приблизиться к данному числу.

Для нахождения примерного значения интеграла воспользуемся формулой $\int_a^b f(x)dx = f(b)(b-a)\frac{m}{exp\_nmb}$, где m – количество успешных попаданий точки под график функции, exp_nmb – общее кол-во попыток. Условие попадания точки под график: $f(xp) > yp$, где xp, yp – координаты случайной точки внутри треугольника.

Исходный код функции представлен на рис. 7.

```python
def calc_area(x0, x1, exp_nmb):
    x_min = x0
    x_max = x1
    y_min, y_max = 0, f(x_max) # предполагается, что функция монотонно возрастает
    positive_res = 0
    for i in range(exp_nmb):
        px, py = r(), r()
        xp = (x_max - x_min) * px + x_min
        yp = (y_max - y_min) * py + y_min
        if f(xp) > yp:
            positive_res += 1
    return (positive_res / exp_nmb) * (x1 - x0) * f(x1)
```

Рис. 7. – исходный код функции.

Проведём 3 серии расчётов для числа экспериментов 10^4, 10^5, 10^6, 10^7 и определим погрешность вычислений для усреднённых по сериям значений. Для нахождения одной серии была использована функция get_area_seria(см. Рис. 8).

```python
def get_area_seria():
    exp_nmb, x0, x1 = 10**4, 0, 2
    res_list = []
    while exp_nmb <= 10**7:
        res = calc_area(x0, x1, exp_nmb)
        res_list.append(res)
        exp_nmb = exp_nmb * 10
    return res_list
```

Рис. 8. – функция get_area_seria.

Исходный код вычисления серий, средних значений и ошибок представлены на рис. 9, результаты вычислений – на рис. 10.

```python
area_seria_avg = []
for i in range(4):
    avg_val = 0
    for j in range(3):
        avg_val += area_seria_total[j][i]
    avg_val = round(avg_val / 3, 8)
    area_seria_avg.append(avg_val)
print("Avg values: ", area_seria_avg)

area_fault = []
for i in range(4):
    area_fault.append(round(abs(6 - area_seria_avg[i]), 8))
print("Area fault: ", area_fault)
```

Рис. 9. – исходный код для вычисления ошибок и средних значений площади.

```
C:\Users\Илья\PycharmProjects\systemsModeling1\venv\Scripts\python.exe
 C:\Users\Илья\PycharmProjects\systemsModeling1\main.py
Area series:  [[5.9976, 5.98554, 5.998446, 6.0006132], [5.949, 6.04152, 5.998572, 5.9982426],
 [6.0678, 5.94198, 6.004548, 6.002226]]
Avg values:  [6.0048, 5.98968, 6.000522, 6.0003606]
Area fault:  [0.0048, 0.01032, 0.000522, 0.0003606]

Process finished with exit code 0
```

Рис. 10. – значения ошибок и средних значений площади.

Вывод.

В данной практической работе были рассмотрены основные понятия статистического эксперимента. С помощью генерации случайных чисел и метода Монте-Карло было найдено значение числа π, найдена погрешность метода для разного числа экспериментов, средние значения числа pi.

Также с помощью метода Монте-Карло была решена задача нахождения значения определённого интеграла функции $f(x) = x^3 + 1$ на интервале от 0 до 2. Для данной функции были проведены 3 серии расчётов для числа экспериментов 10^4, 10^5, 10^6, 10^7, определена погрешность вычислений для усреднённых по сериям значений.