|  |
| --- |
|  |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ |

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА – Российский технологический университет»**

ИНСТИТУТ КИБЕРНЕТИКИ КАФЕДРА ВЫСШЕЙ МАТЕМАТИКИ

Лабораторная работа 1

## по курсу «Теория вероятностей и математическая статистика, часть 2»

**ВАРИАНТ 63**

Т е м а : **Первичная обработка выборки из**

**дискретной генеральной совокупности**\_

Выполнила: Студентка 3-го курса Макарчук Н.С.

Группа: КМБО-03-18

М О С К В А – 20 2 1

# Задание

**Задание 1.** Получить выборку, сгенерировав 200 псевдослучайных чисел, распределенных по биномиальному закону с параметрами n и p.

n = 7 + V mod15; p = 0, 2 + 0, 005V

**Задание 2.** Получить выборку, сгенерировав 200 псевдослучайных чисел, распределенных по геометрическому закону с параметром. p = 0, 2 + 0, 005V

**Задание 3.** Получить выборку, сгенерировав 200 псевдослучайных чисел, распределенных по закону Пуассона с параметром. λ = 1 + 0, 02V

Следуя **Указаниям** для всех выборок построить:

1. статистический ряд;
2. полигон относительных частот;
3. график эмпирической функции распределения;

Найти:

1. выборочное среднее;
2. выборочную дисперсию;
3. выборочное среднее квадратическое отклонение;
4. выборочную моду;
5. выборочную медиану;
6. выборочный коэффициент асимметрии;
7. выборочный коэффициент эксцесса.

Провести сравнение рассчитанных характеристик с теоретическими значениями. V - номер варианта. Вычисления проводить с точностью до 0,00001.

# Краткие теоретические сведения

## Биномиальное распределение

* + - Ряд распределения

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | ... | m | ... | n |
|  |  |  |  | ... |  | ... |  |

* + - Математическое ожидание



* + - Дисперсия



* + - Среднее квадратичное отклонение



* + - Мода



* + - Медиана



* + - Коэффициент асимметрии
    - Коэффициент эксцесса

## 2.1 Геометрическое распределение

* + - Ряд распределения

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | ... | m | ... |
|  |  |  |  | ... | *p* | ... |

* + - *Математическое ожидание*



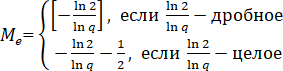
* + - Дисперсия



* + - Среднее квадратичное отклонение



* + - Мода
    - Медиана



* + - Коэффициент асимметрии



* + - Коэффициент эксцесса



## Распределение Пуассона

* + - Ряд распределения

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | ... | m | ... |
|  |  |  |  | ... |  | ... |

* + - Математическое ожидание



* + - Дисперсия



* + - Среднее квадратичное отклонение



* + - Мода



* + - Медиана



* + - Коэффициент асимметрии



* + - Коэффициент эксцесса



* 1. Средства языка программирования

В программе расчёта был использован язык программирования Python.

Работа осуществлялась в среде Jupyter Notebook.

Модуль Numpy:

Функции, которые использовались для генерации выборки

1) random.geometric(p, N) - генерирует N чисел по геометрическому распределения с параметром p — вероятность

2) random.poisson(lambda, N) — генерирует N чисел по распределению Пуассона, с параметромlamda.

3) random.binomial(n, p, N) — функция генерирует N чисел по Биноминальному распределению с параметрами n и p

matplotlib.pyplot.plot(xs, ys) – функция для отрисовки полигона частот:

# Результаты расчётов

## Задание 1

Биноминальная распределение

Параметры: p = 0.515, n = 10

* + - Полученная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | 3 | 2 | 5 | 3 | 6 | 3 | 6 | 6 |
| 5 | 6 | 3 | 6 | 5 | 6 | 6 | 4 | 7 |
| 4 | 5 | 4 | 2 | 3 | 6 | 5 | 3 | 5 |
| 7 | 6 | 5 | 6 | 8 | 3 | 3 | 6 | 6 |
| 3 | 7 | 4 | 6 | 5 | 6 | 4 | 6 | 7 |
| 5 | 4 | 8 | 5 | 4 | 9 | 6 | 3 | 4 |
| 3 | 5 | 5 | 5 | 4 | 5 | 4 | 7 | 7 |
| 4 | 2 | 4 | 4 | 5 | 9 | 5 | 3 | 6 |
| 3 | 5 | 7 | 7 | 8 | 4 | 4 | 6 | 3 |
| 6 | 6 | 6 | 3 | 7 | 7 | 6 | 3 | 8 |
| 6 | 6 | 5 | 5 | 8 | 7 | 7 | 5 | 5 |
| 6 | 5 | 4 | 3 | 6 | 5 | 6 | 6 | 5 |
| 4 | 6 | 7 | 2 | 6 | 6 | 5 | 4 | 8 |
| 5 | 3 | 4 | 5 | 6 | 4 | 3 | 5 | 1 |
| 3 | 4 | 2 | 6 | 8 | 5 | 7 | 4 | 5 |
| 6 | 8 | 4 | 5 | 5 | 7 | 2 | 4 | 3 |
| 6 | 3 | 7 | 5 | 5 | 6 | 2 | 4 | 7 |
| 7 | 6 | 6 | 7 | 4 | 5 | 3 | 5 | 3 |
| 7 | 6 | 5 | 7 | 6 | 5 | 7 | 6 | 5 |
| 2 | 5 | 6 | 4 | 5 | 3 | 4 | 4 | 7 |

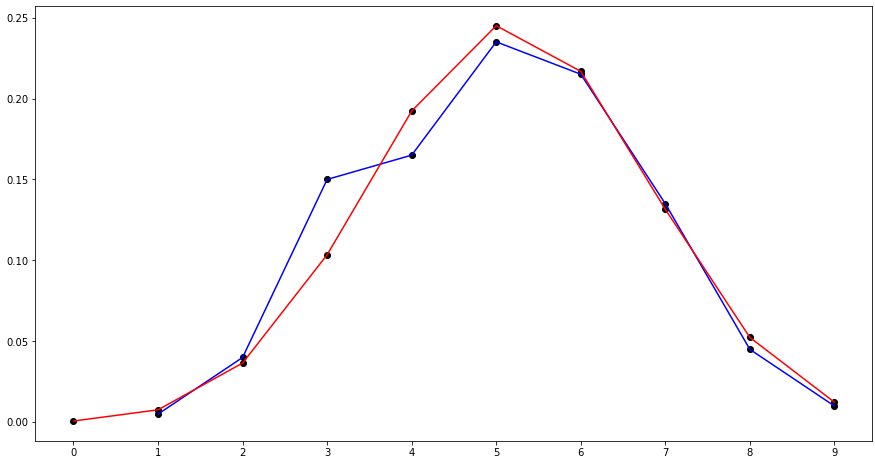
Упорядоченная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 4 | 4 | 5 | 5 | 6 | 6 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 6 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 2 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 7 |
| 3 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 4 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 8 |
| 3 | 3 | 4 | 5 | 5 | 5 | 6 | 7 | 9 |
| 3 | 4 | 4 | 5 | 5 | 6 | 6 | 7 | 9 |

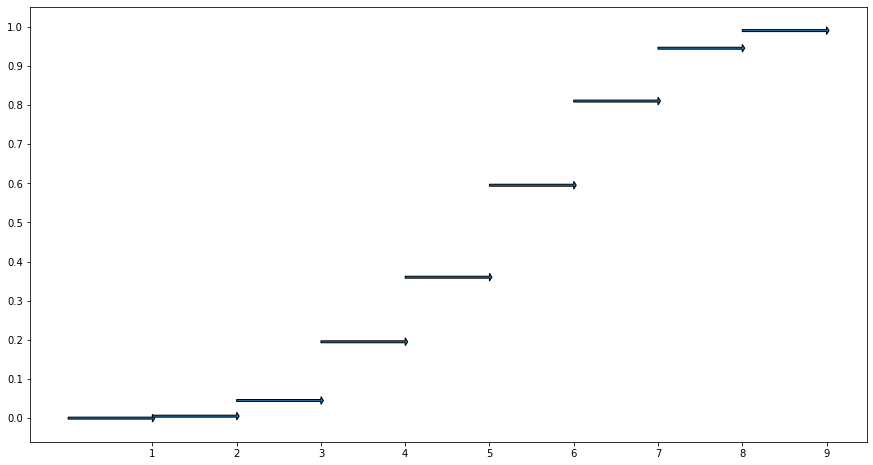
* + - Статистический ряд

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 1 | **1** | **0.005** | **0.005** |
| 2 | **8** | **0.04** | **0.045** |
| 3 | **30** | **0.15** | **0.195** |
| 4 | **33** | **0.165** | **0.36** |
| 5 | **47** | **0.235** | **0.595** |
| 6 | **43** | **0.215** | **0.81** |
| 7 | **27** | **0.135** | **0.945** |
| 8 | **9** | **0.045** | **0.99** |
| 9 | **2** | **0.01** | **1** |
|  | **200** |  |  |

* + - График полигона относительных частот



* + - Эмпирическая функция распределения



* + - Результаты расчетов требуемых характеристик

Выборочное среднее: 5.055

Выборочная дисперсия: 2.52198

Выборочное среднее квадратичное отклонение: 1.58807

Выборочная мода: 5

Выборочная медиана: 5

Выборочный коэффициент асимметрии: -0.00781

Выборочный коэффициент асимметрии: -0.54320

## Задание 2

**Геометрическое распределение**

Полученная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 2 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 2 | 0 | 1 |
| 0 | 0 | 0 | 2 | 0 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| 2 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 2 | 1 | 1 | 0 | 1 | 2 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 4 | 0 | 1 | 0 |
| 2 | 0 | 0 | 4 | 0 | 3 | 1 | 0 | 3 |
| 0 | 0 | 2 | 0 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 2 | 7 | 1 | 3 | 0 | 0 | 1 |
| 3 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 3 |
| 0 | 1 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| 2 | 1 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 5 | 1 | 1 | 0 | 1 | 0 | 2 |
| 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 2 |
| 0 | 3 | 0 | 1 | 0 | 1 | 0 | 0 | 2 |
| 3 | 3 | 4 | 2 | 2 | 0 | 1 | 1 | 0 |
| 0 | 0 | 2 | 0 | 2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 3 | 1 | 0 | 1 | 2 |

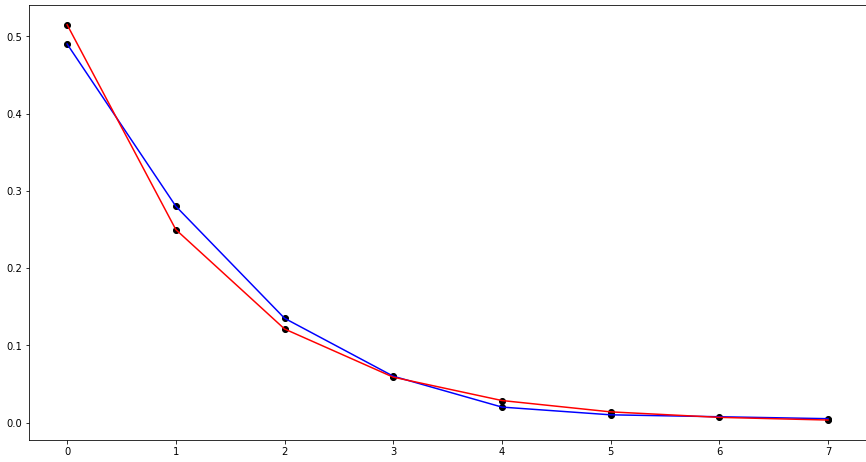
* + - Упорядоченная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 2 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 4 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 4 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 4 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 4 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 5 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 5 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 7 |

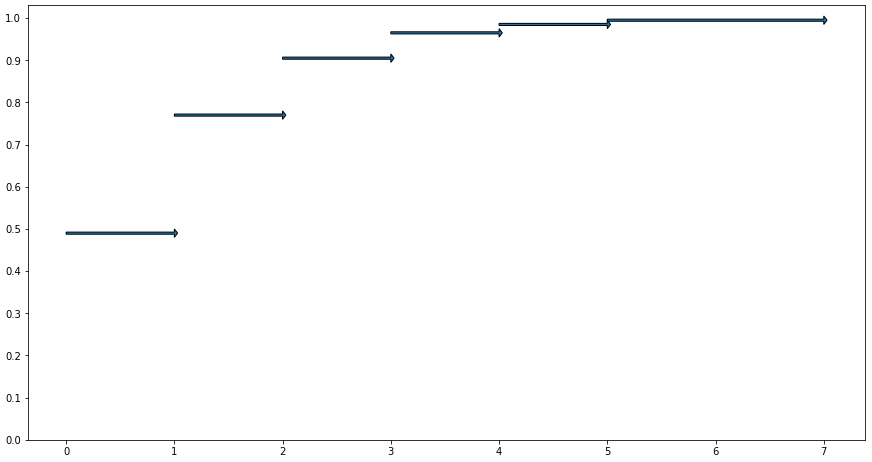
* + - Статистический ряд

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **0** | **98** | **0.49** | **0.49** |
| **1** | **56** | **0.28** | **0.77** |
| **2** | **27** | **0.135** | **0.905** |
| **3** | **12** | **0.06** | **0.965** |
| **4** | **4** | **0.02** | **0.985** |
| **5** | **2** | **0.01** | **0.995** |
| **7** | **1** | **0.005** | **1** |
|  | **200** | **1.0** | **-** |

* + - График полигона относительных частот



* + - Эмпирическая функция распределения



Результаты расчетов требуемых характеристик

Выборочное среднее: 0.895

Выборочная дисперсия: 1.37397

Выборочное среднее квадратичное отклонение: 1.17216

Выборочная мода: 0

Выборочная медиана: 1

Выборочный коэффициент асимметрии: 1.75033

Выборочный коэффициент асимметрии: 4.065314

## Задание 3

* + - Полученная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 4 | 3 | 3 | 1 | 2 | 3 | 1 | 1 |
| 3 | 3 | 1 | 4 | 2 | 2 | 0 | 3 | 2 |
| 1 | 5 | 3 | 0 | 1 | 0 | 0 | 0 | 1 |
| 5 | 0 | 3 | 3 | 2 | 2 | 5 | 2 | 3 |
| 3 | 3 | 0 | 2 | 4 | 3 | 2 | 2 | 3 |
| 3 | 2 | 1 | 4 | 2 | 4 | 0 | 2 | 1 |
| 3 | 2 | 2 | 0 | 3 | 1 | 2 | 1 | 2 |
| 2 | 1 | 0 | 0 | 2 | 3 | 4 | 2 | 2 |
| 1 | 2 | 5 | 3 | 4 | 4 | 1 | 2 | 1 |
| 3 | 2 | 2 | 2 | 1 | 2 | 1 | 1 | 3 |
| 2 | 3 | 2 | 6 | 2 | 0 | 1 | 4 | 4 |
| 4 | 1 | 6 | 3 | 2 | 2 | 2 | 0 | 2 |
| 3 | 1 | 2 | 5 | 2 | 1 | 6 | 7 | 1 |
| 1 | 3 | 0 | 2 | 6 | 5 | 3 | 6 | 2 |
| 1 | 1 | 2 | 0 | 3 | 4 | 1 | 2 | 3 |
| 1 | 2 | 3 | 1 | 2 | 2 | 3 | 0 | 3 |
| 0 | 2 | 4 | 2 | 1 | 4 | 3 | 2 | 3 |
| 2 | 1 | 3 | 0 | 4 | 1 | 1 | 2 | 1 |
| 2 | 0 | 1 | 4 | 3 | 0 | 2 | 3 | 2 |
| 6 | 2 | 1 | 2 | 1 | 2 | 4 | 3 | 3 |

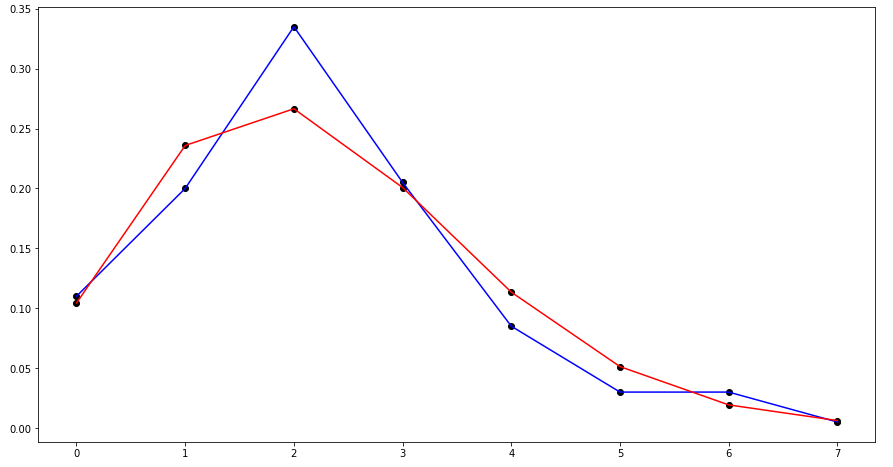
* + - Упорядоченная выборка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 |
| 0 | 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 4 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 5 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 6 |
| 0 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 7 |

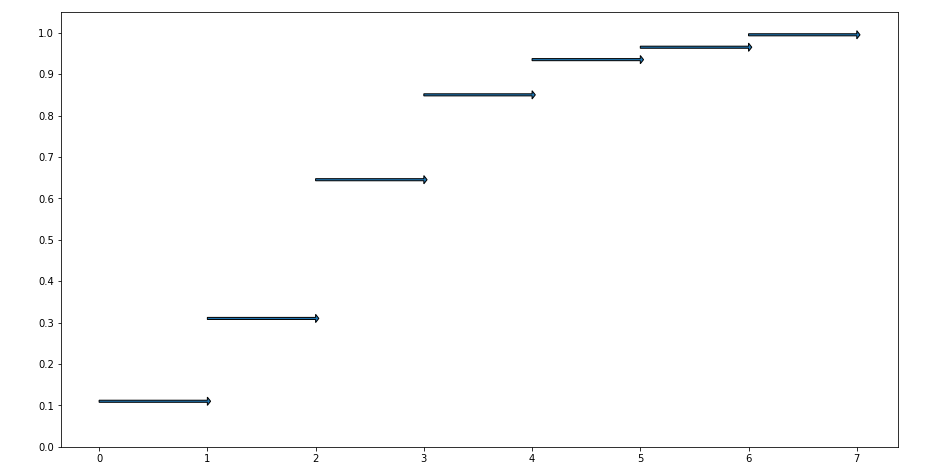
* + - Статистический ряд

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 22 | 0.11 | 0.11 |
| 1 | 40 | 0.2 | 0.31 |
| 2 | 67 | 0.335 | 0.645 |
| 3 | 41 | 0.205 | 0.85 |
| 4 | 17 | 0.085 | 0.935 |
| 5 | 6 | 0.03 | 0.965 |
| 6 | 6 | 0.03 | 0.995 |
| 7 | 1 | 0.005 | 1 |
|  | 200 | 1 | - |

* + - График полигона относительных частот



* + - Эмпирическая функция распределения



Результаты расчетов требуемых характеристик

Выборочное среднее: 2.19

Выборочная дисперсия: 2.02389

Выборочное среднее квадратичное отклонение: 1.422638

Выборочная мода: 2

Выборочная медиана: 2

Выборочный коэффициент асимметрии: 0.694451

Выборочный коэффициент асимметрии: 0.613297

# Анализ результатов и выводы

## Таблица сравнения относительных частот и теоретических вероятностей

* + - Биномиальное распределение

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0 | 0.000720 | 0.00072 |
| 1 | **0.005** | 0.007647 | 0.002647 |
| 2 | **0.04** | 0.036539 | 0.003461 |
| 3 | **0.15** | 0.103465 | 0.046535 |
| 4 | **0.165** | 0.192264 | 0.027264 |
| 5 | **0.235** | 0.244988 | 0.009988 |
| 6 | **0.215** | 0.216785 | 0.001785 |
| 7 | **0.135** | 0.131540 | 0.00346 |
| 8 | **0.045** | 0.052379 | 0.007379 |
| 9 | **0.01** | 0.012360 | 0.00236 |
|  | 1.0 | 0.99797 | 0.046535 |

* + - Геометрическое распределение

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0.490 | 0.515000 | 0.025 |
| 1 | 0.280 | 0.249775 | 0.030225 |
| 2 | 0.135 | 0.121141 | 0.013859 |
| 3 | 0.060 | 0.058753 | 0.001247 |
| 4 | 0.020 | 0.028495 | 0.008495 |
| 5 | 0.010 | 0.013820 | 0.00382 |
| 6 | 0 | 0.006703 | 0.006703 |
| 7 | 0.005 | 0.003251 | 0.001749 |
|  | 1.0 | 0.99694 | 0.030225 |

* + - Распределение Пуассона

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0.11 | 0.104350 | 0.00565 |
| 1 | 0.2 | 0.235832 | 0.035832 |
| 2 | 0.335 | 0.266490 | 0.06851 |
| 3 | 0.205 | 0.200756 | 0.004244 |
| 4 | 0.085 | 0.113427 | 0.028427 |
| 5 | 0.03 | 0.051269 | 0.021269 |
| 6 | 0.03 | 0.019311 | 0.010689 |
| 7 | 0.005 | 0.006235 | 0.001235 |
|  | 1.0 | 0.99767 | 0.06851 |

## Таблица сравнения рассчитанных характеристик с теоретическими значениями

* + - Биномиальное распределение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Эксперименталь ное значение | Теоретичес кое значение | Абсолютное значение | Относительно е отклонение |
| Выборочное  среднее | 5.055 | 5.15 | 0.095 | 0.01844 |
| Выборочная  дисперсия | 2.52198 | 2.49775 | 0.02423 | 0.0097 |
| Выборочное среднее квадратично  е отклонение | 1.58807 | 1.58042 | 0.00765 | 0.00484 |
| Выборочная  мода | 5 | 5 | 0 | 0 |
| Выборочная  медиана | 5 | 5 | 0 | 0 |
| Выборочный  коэффициен т ассиметрии | -0.00781 | -0.01898 | 0.01117 | 0.01898 |
| Выборочный коэффициен  т эксцесса | -0.54320 | -0.19963 | 0.34357 | 0.19963 |

* + - Геометрическое распределение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Эксперименталь ное значение | Теоретичес  кое значение | Абсолютное значение | Относительно е отклонение |
| Выборочное  среднее | 0.895 | 0.941747 | 0.046747 | 0.049639 | |
| Выборочная  дисперсия | 1.37397 | 1.82863 | 0.45466 | 0.248634 | |
| Выборочное среднее квадратичное отклонение | 1.1721667 | 1.35227 | 0.180103 | 0.133186 | |
| Выборочная  мода | 0 | 0 | 0 | --- | |
| Выборочная  медиана | 1 | 0 | 1 | --- | |
| Выборочный  коэффициен т ассиметрии | 1.75033 | 2.132335 | 0.382005 | 0.179149 | |
| Выборочный коэффициен  т эксцесса | 4.06531 | 6.546855 | 2.481545 | 0.379044 | |

* + - Распределение Пуассона

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Эксперименталь ное значение | Теоретичес  кое значение | Абсолютное значение | Относительно е отклонение |
| Выборочное  среднее | 2.19 | 2.26 | 0.07 | 0.030973 |
| Выборочная  дисперсия | 2.023899 | 2.26 | 0.236101 | 0.104469 |
| Выборочное среднее квадратично  е отклонение | 1.42263 | 1.50332 | 0.08069 | 0.053675 |
| Выборочная  мода | 2 | 2 | 0 | 0 |
| Выборочная  медиана | 2 | 2 | 0 | 0 |
| Выборочный коэффициен т ассиметрии | 0.69445 | 0.66519 | 0.02926 | 0.043987 |
| Выборочный коэффициен  т эксцесса | 0.61329 | 0.44248 | 0.17081 | 0.386029 |

# Список литературы

1. Математическая статистика [Электронный ресурс]: метод. указания по выполнению лаб. работ / А.А. Лобузов — М.: МИРЭА, 2017.
2. Боровков А. А. Математическая статистика. — СПб.: Лань, 2010.704 с.
3. Гмурман В.Е. Теория вероятностей и математическая статистика. — М.: Юрайт, 2013. — 479 с.
4. Гмурман В.Е. Руководство к решению задач по теории вероятностей и математической статистике. — М.: Юрайт, 2013. — 404 с.
5. Емельянов Г.В.Скитович В.П. Задачник по теории вероятностей и математической статистике. — СПб.: Лань, 2007. — 336 с.
6. Ивченко Г. И., Медведев Ю. И. Введение в математическую статистику. — М.: Изд-во ЛКИ, 2010. — 599 с.
7. Кибзун А.И., Горяинова Е.Р., Наумов А.В. Теория вероятностей и математическая статистика. Базовый курс с примерами и задачам. Учебное пособие — М.:ФИЗМАТЛИТ, 2005. — 232 с.

# Приложение

import numpy as np

import math

import scipy as sp

import matplotlib.pyplot as plt

import pandas as pd

from decimal import Decimal

#params:

V = 63

EPS = 0.00001

n = 7 + V % 15

p = 0.2 + 0.005\*V

N = 200

lamb = 1 + 0.02\*V

q = 1 - p

distrub\_geom = np.random.geometric(p, N) - 1

distrub\_poisson = np.random.poisson(lamb, N)

distrub\_binom = np.random.binomial(n, p, N)

def get\_frequency\_table(xs):

df = pd.DataFrame({"values" :xs})

df = pd.value\_counts(df["values"]).to\_frame().reset\_index()

df.columns = ['Value','Count']

df = df.sort\_values(by=['Value']) #Сортировка по значению

df.reset\_index(inplace = True, drop = True)

return df

def count\_edf(freqs):

edf = [0]

s = 0

for i in freqs:

s += i

edf.append(round(s, 5))

return edf

def count\_k\_moment(table, k):

return sum(table.Frequency \* table.Value\*\*k)

def count\_emperical\_mean(table):

return count\_k\_moment(table, 1)

def count\_centr\_k\_moment(table, k):

mean = count\_emperical\_mean(table)

return sum((table.Value - mean)\*\*k \* table.Frequency)

def plot\_edf(edf, values):

fig = plt.figure(figsize=(15,8))

ax = fig.add\_subplot(111)

W = 0.005

n = len(values)

if (values[0] != 0):

plt.arrow(values[0] - 1,0,1,0,width=W,head\_width = 0.02)

for i in range(n - 1):

plt.arrow(values[i], edf[i+1], values[i+1] - values[i], 0,width=W, head\_width = 0.02)

ax.set(

xticks = np.arange(values[0], values[n - 1] + 1),

yticks = np.arange(edf[0], edf[-1] + 0.1, 0.1))

def count\_emperical\_dispersion(table):

mean = count\_emperical\_mean(table)

sqr = sum(table.Frequency \* table.Value\*\*2)

return sqr - mean\*\*2

def count\_emperical\_std(table):

return np.sqrt(count\_emperical\_dispersion(table))

def count\_moda(table):

moda = max(table.Count)

indx = table.Count[table.Count == moda].index.tolist()

if len(indx) == 1:

return int(table.Value[indx])

for j in range(1, len(indx)):

if abs(indx[j] - indx[j-1]) > 1:

return "Выборочной моды не существует"

return 1/2\*(table.Value[indx[0]] + table.Value[indx[-1]])

def count\_skewness(table):

std = count\_emperical\_std(table)

num = count\_centr\_k\_moment(table, 3)

return num / (std\*\*3)

def count\_kurtosis(table):

std = count\_emperical\_std(table)

num = count\_centr\_k\_moment(table, 4)

return num/(std\*\*4) – 3

def plot\_table(table, table\_comp):

fig = plt.figure(figsize=(15,8))

ax = fig.add\_subplot(111)

plt.plot(table.Value, table.Frequency, color='b')

plt.plot(table\_comp.Value, table\_comp.P, color="r")

plt.scatter(table.Value, table.Frequency, color="black")

plt.scatter(table\_comp.Value, table\_comp.P, color="black")

ax.set(

xticks = table\_comp.Value)

#таблица сравнения относительных частот и теоретических вероятностей

def create\_table\_comp(table, type\_):

n = max(table.Value) + 1

table\_comp = pd.DataFrame()

table\_comp['Value'] = np.arange(0, n)

if type\_ == "binom":

table\_comp['P'] = np.zeros(n)

for i in range(n):

table\_comp.P[i] = math.comb(n, i)\*(p\*\*i)\*(q\*\*(n-i))

if type\_ == "geom":

table\_comp['P'] = (q\*\*table\_comp.Value)\*p

if type\_ == "poisson":

table\_comp['P'] = np.zeros(n)

for i in range(n):

table\_comp.P[i] = (lamb\*\*i)/math.factorial(i)\*np.exp(-lamb)

return table\_comp

def main\_count(xs, type\_):

mean = 0

dispersion = 0

std = 0

skewness = 0

kurtosis = 0

moda = 0

median = 0

table = get\_frequency\_table(xs)

table['Frequency'] = table.Count / N

edf = count\_edf(table.Frequency) # Empirical distribution function list[float]

emp\_mean = count\_emperical\_mean(table)

emp\_disp = count\_emperical\_dispersion(table)

emp\_std = count\_emperical\_std(table)

emp\_skewness = count\_skewness(table)

emp\_kurtosis = count\_kurtosis(table)

table\_comp = create\_table\_comp(table, type\_)

plot\_table(table, table\_comp)

print(table\_comp)

if type\_ == "binom":

mean = n\*p

dispersion = n\*p\*q

std = np.sqrt(dispersion)

if int((n+1)\*p) - (n+1)\*p == 0:

moda = (n+1)\*p - 1/2

else:

moda = int((n+1)\*p)

median = int(n\*p)

skewness = (q - p) / np.sqrt(n\*p\*q)

kurtosis = (1 - 6\*p\*q) / (n\*p\*q)

if type\_ == "poisson":

mean = lamb

dispersion = lamb

std = np.sqrt(lamb)

moda = int(lamb)

median = int(lamb + 1/3 - 0.02/lamb)

skewness = 1/np.sqrt(lamb)

kurtosis = 1/lamb

if type\_ == "geom":

mean = q/p

dispersion = q/p/p

std = np.sqrt(q)/p

moda = 0

if int(np.log(2)/np.log(q)) - np.log(2)/np.log(q) == 0:

median = -np.log(2)/np.log(q) - 1/2

else:

median = int(-np.log(2)/np.log(q))

skewness = (2-p)/np.sqrt(q)

kurtosis = 6+p\*p/q

print(table, end='\n\n')

print("EDF : ", edf, end='\n\n')

plot\_edf(edf, table.Value)

print("emperical\_mean :",emp\_mean, end='\n\n' )

print("emperical\_disp :",emp\_disp, end='\n\n' )

print("emperical\_std :",emp\_std, end='\n\n' )

print("emperical moda : ", count\_moda(table), end='\n\n')

print("emp\_skewness :",emp\_skewness, end='\n\n' )

print("emp\_kurtosis: ", emp\_kurtosis, end='\n\n\n\n')

print("theoretical characteristics:", end='\n\n')

print("mean:", mean, end='\n\n')

print("disp:", dispersion, end='\n\n')

print("std:", std, end='\n\n')

print("moda: ", moda, end='\n\n')

print("median: ", median, end='\n\n')

print("skewness :", skewness, end='\n\n' )

print("kurtosis: ", kurtosis, end='\n\n')

def print\_file(xs, type\_):

with open(type\_ + ".txt", 'w') as file1:

for x in xs:

print(x, end='\n', file=file1)

with open("sort" + type\_ + ".txt", 'w') as file2:

for x in np.sort(xs):

print(x, end='\n', file=file2)

print\_file(distrub\_geom, "geom")

print\_file(distrub\_binom, "binom")

print\_file(distrub\_poisson, "poisson")

main\_count(distrub\_geom, "geom")

main\_count(distrub\_poisson, "poisson")

main\_count(distrub\_binom, "binom")