|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  | | --- | --- | --- | |  |  |  | | Министерство образования и науки РОССИЙСКОЙ ФЕДЕРАЦИИ | | |  Федеральное государственное бюджетное образовательное учреждение высшего образования  **«МИРЭА – Российский технологический университет»** | |
|  | |
|  | |
|  |  |

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

КАФЕДРА ВЫСШЕЙ МАТЕМАТИКИ

Типовой расчет

**по математической статистике**

**Часть 1**

**ВАРИАНТ 165**

Выполнил:

Студент 3-го курса

Баттур Ц.

Группа: КМБО-07-22

МОСКВА – 2025

Содержание

[Задание 3](#_Toc2897)

[Краткие теоретические сведения 5](#_Toc28770)

[Результаты расчётов 1](#_Toc2879)3

[Список литературы 23](#_Toc29997)

[Приложение 24](#_Приложение)

# Задание

**Задание 1.** Получить выборку объёмом *N=*200, сгенерировав псевдослучайные числа, распределённые по биномиальному закону с параметрами ***n*** и ***p***:

**Задание 2.** Получить выборку объёмом *N=*200, сгенерировав псевдослучайные числа, распределённые по геометрическому закону с параметром ***p***:

В заданиях 1 и 2 построить:

1. статистический ряд;
2. график полигона относительных частот с наложенным на него и выделенным красным цветом график полигона теоретических вероятностей;
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;

составить таблицы:

1. сравнения относительных частот и теоретических вероятностей;
2. сравнения рассчитанных характеристик с теоретическими значениями.

**Задание 3.** Получить выборку объёмом *N*=200, сгенерировав псевдослучайные числа, распределённые по показательному закону с параметром .

В задании 3 построить:

1. график эмпирической функции распределения;
2. интервальный ряд и ассоциированный статистический ряд;
3. гистограмму относительных частот с наложенным на неё и выделенным красным цветом график плотности распределения;

найти:

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

составить таблицы:

1. сравнения относительных частот и теоретических вероятностей попадания в интервалы;
2. сравнения рассчитанных характеристик с теоретическим значениями.

# Краткие теоретические сведения

Выборка объёмом *N=*200 с сгенерированными псевдослучайными числами, распределённые по биномиальному закону с параметрами ***n*** и ***p***:

Выборка объёмом *N=*200 с сгенерированными псевдослучайными числами,

распределённые по геометрическому закону с параметром ***p***:

Полученные выборки упорядочить по возрастанию, определить частоты и относительные частоты , построить статистический ряд.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| … | … | … | … |
|  |  |  |  |
|  |  |  | - |

Таблица 1. Статистический ряд.

Полигон относительных частот - ломаная линия, соединяющая последовательно точки с координатами .

Эмпирическая функция распределения:

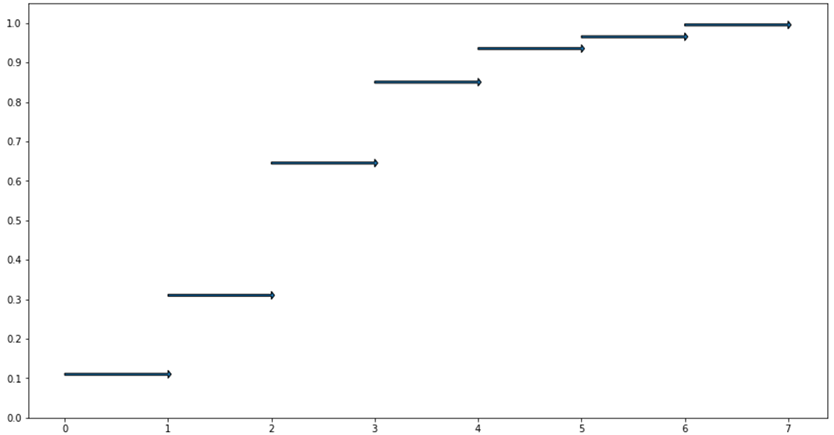


Рисунок 1. Образец графика эмпирической функции распределения.

Выборочное среднее:

Выборочный момент k-ого порядка (выборочный k-ый момент):

Выборочная дисперсия:

Выборочная центральный момент k-ого порядка:

Выборочное среднее квадратическое отклонение:

Выборочный коэффициент асимметрии:

Выборочный коэффициент эксцесса:

Выборочная мода ;

если то ,

если , то *i < k < j*, то - не существует.

Выборочная медиана:

|  |  |
| --- | --- |
| **Биномиальное распределение** | |
| Вероятность |  |
| Математическое ожидание |  |
| Дисперсия |  |
| Среднее квадратическое отклонение |  |
| Мода | , если - дробное;  , если - целое; |
| Медиана |  |
| Коэффициент асимметрии |  |
| Коэффициент эксцесса |  |

Таблица 2. Характеристики биномиального распределения.

|  |  |
| --- | --- |
| **Геометрическое распределение** | |
| Вероятность |  |
| Математическое ожидание |  |
| Дисперсия |  |
| Среднее квадратическое отклонение |  |
| Мода | 0 |
| Медиана |  |
| Коэффициент асимметрии |  |
| Коэффициент эксцесса |  |

Таблица 3. Характеристики геометрического распределения.

В задании 3 полученную выборку псевдослучайных чисел, распределённые по показательному закону, упорядочить по возрастанию, определить интервалы ; число интервалов находится по формуле Стерджеса .

|  |  |  |
| --- | --- | --- |
| Интервалы |  |  |
|  |  |  |
|  |  |  |
| ... | ... | ... |
|  |  |  |
|  |  |  |

Таблица 4. Интервальный ряд.

- число значений, попавших в *i-*ый интервал; - относительная частота попадания в *i-*ый интервал, .

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
| ... | ... | ... |
|  |  |  |
|  |  |  |

Таблица 5. Ассоциированный статический ряд, где - середина интервала.

Эмпирическая функция распределения:

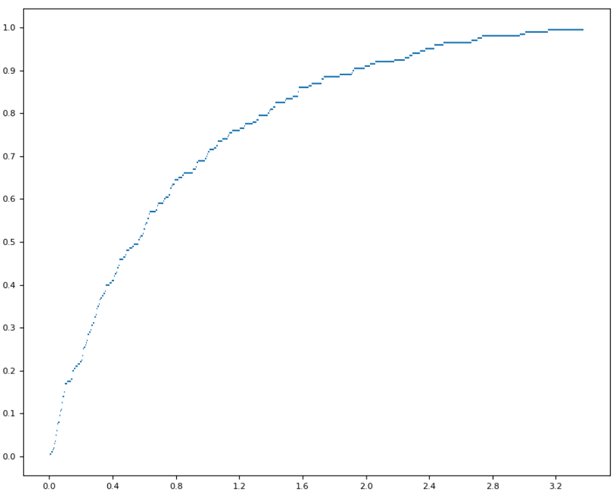


Рисунок 2. Образец графика эмпирической функции распределения.

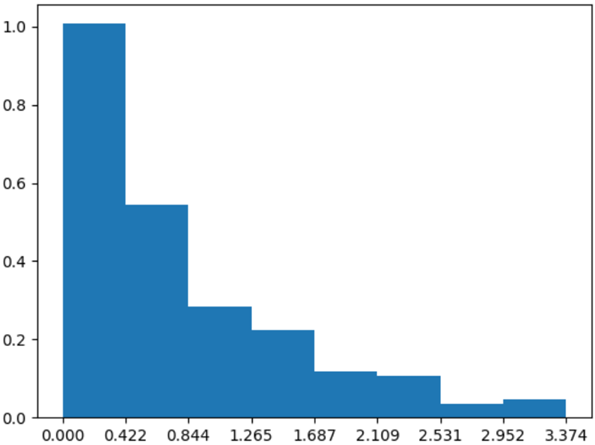


Рисунок 3. Образец гистограммы относительных частот.

Площадь *i*-ого столбца гистограммы равна , а высота .

Выборочное среднее:

Выборочная дисперсия с поправкой Шеппарда:

Выборочное среднее квадратическое отклонение:

Выборочная мода:

Если модальный интервал, на котором высота гистограммы максимальна, один, то , где - левая граница модального интервала ; - правая граница модального интервала ; - относительная частота на модальном интервале; , - относительные частоты интервалов слева и справа от модального интервала.

Если модальных интервалов несколько, и все они идут подряд (т. е. интервалы , ..., - все модальные), то

Если между модальными интервалами находятся немодальные, то считаем, что выборочной моды не существует.

Выборочная медиана:

Выборочный момент *k*-ого порядка:

Выборочный центральный момент *k*-ого порядка:

Выборочный коэффициент асимметрии:

Выборочный коэффициент эксцесса:

|  |  |
| --- | --- |
| **Показательное распределение:** ) | |
| Математическое ожидание |  |
| Дисперсия |  |
| Среднее квадратическое отклонение |  |
| Мода | 0 |
| Медиана |  |
| Коэффициент асимметрии |  |
| Коэффициент эксцесса |  |

# Результаты расчётов

**Задание 1:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 6 | 5 | 3 | 7 | 6 | 6 | 10 | 7 | 7 | 9 |
| 8 | 5 | 8 | 6 | 8 | 7 | 4 | 7 | 9 | 6 |
| 3 | 7 | 7 | 9 | 5 | 9 | 8 | 8 | 6 | 8 |
| 9 | 10 | 6 | 7 | 7 | 4 | 7 | 5 | 7 | 8 |
| 10 | 5 | 6 | 9 | 5 | 6 | 7 | 8 | 8 | 7 |
| 8 | 8 | 8 | 8 | 9 | 7 | 6 | 6 | 5 | 6 |
| 7 | 6 | 5 | 6 | 5 | 8 | 5 | 8 | 10 | 8 |
| 10 | 9 | 7 | 6 | 6 | 7 | 7 | 7 | 6 | 7 |
| 7 | 7 | 7 | 5 | 8 | 8 | 7 | 6 | 9 | 8 |
| 8 | 7 | 7 | 8 | 8 | 9 | 6 | 6 | 4 | 8 |
| 7 | 10 | 9 | 8 | 8 | 5 | 6 | 7 | 9 | 7 |
| 8 | 5 | 8 | 7 | 6 | 7 | 6 | 6 | 6 | 6 |
| 8 | 5 | 9 | 9 | 7 | 6 | 8 | 10 | 9 | 7 |
| 8 | 4 | 6 | 7 | 7 | 7 | 4 | 3 | 9 | 7 |
| 7 | 8 | 8 | 5 | 4 | 9 | 6 | 6 | 7 | 7 |
| 5 | 9 | 9 | 6 | 7 | 9 | 8 | 6 | 10 | 5 |
| 6 | 3 | 5 | 5 | 4 | 6 | 8 | 7 | 5 | 10 |
| 8 | 9 | 10 | 6 | 7 | 8 | 7 | 5 | 7 | 7 |
| 5 | 6 | 5 | 7 | 8 | 9 | 6 | 8 | 7 | 8 |
| 8 | 8 | 10 | 7 | 10 | 7 | 8 | 3 | 7 | 6 |

Таблица 1: 200 выборок биномиального распределения

Биномиальное распределение

|  |  |  |  |
| --- | --- | --- | --- |
| ***X*** |  |  |  |
| 3 | 5.00000 | 0.02500 | 0.02500 |
| 4 | 7.00000 | 0.03500 | 0.06000 |
| 5 | 23.00000 | 0.11500 | 0.17500 |
| 6 | 38.00000 | 0.19000 | 0.36500 |
| 7 | 51.00000 | 0.25500 | 0.62000 |
| 8 | 42.00000 | 0.21000 | 0.83000 |
| 9 | 22.00000 | 0.11000 | 0.94000 |
| 10 | 12.00000 | 0.06000 | 1.00000 |
| Total | 200 | 1 |  |

Таблица 2: Статический ряд

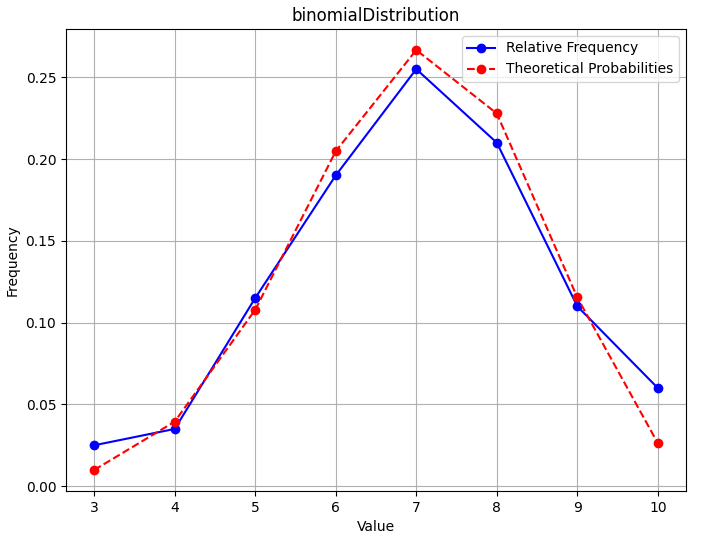


Рисунок 1: Полигон относительных частот

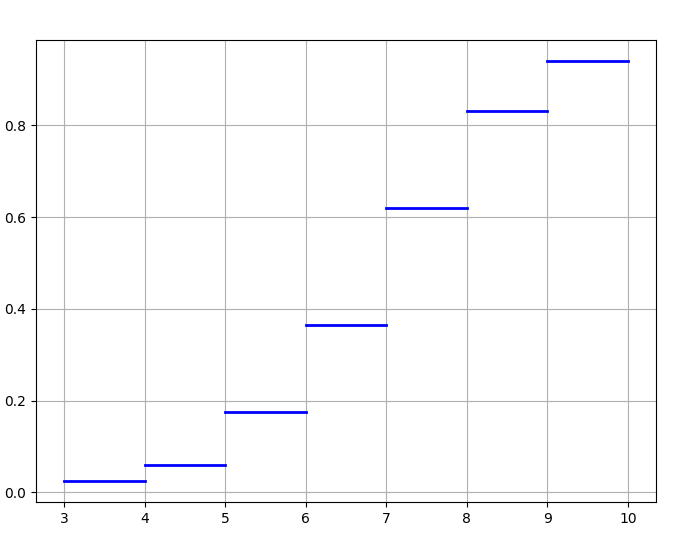


Рисунок 2: График эмпирической функции распеределении

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 3 | 0.02500 | 0.00989 | 0.01511 |
| 4 | 0.03500 | 0.03944 | 0.00444 |
| 5 | 0.11500 | 0.10785 | 0.00715 |
| 6 | 0.19000 | 0.20480 | 0.01480 |
| 7 | 0.25500 | 0.26667 | 0.01167 |
| 8 | 0.21000 | 0.22787 | 0.01787 |
| 9 | 0.11000 | 0.11539 | 0.00539 |
| 10 | 0.06000 | 0.02629 | 0.03371 |
| Total | 1.00000 | 1.00000 | 0.03371 |

Таблица 3: Таблица сравнения относительных частот и теоретических вероятностей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название показателя** | **Выборочное значение** | **Теоретическое значение** | **Абсолютное отклонение** | **Относительное отклонение** |
| Среднее значение | 6.98500 | 6.95000 | 0.03500 | 0.00504 |
| Дисперсия | 2.55478 | 2.11975 | 0.43503 | 0.20522 |
| Среднее квадратичное отклонение | 1.59837 | 1.45594 | 0.14243 | 0.09783 |
| Мода | 7.00000 | 7.00000 | 0.00000 | 0.00000 |
| Медиана | 6.50000 | 6.00000 | 0.50000 | 0.08333 |
| Коэффициент асимметрии | -0.20327 | -0.26787 | 0.06460 | -0.24117 |
| Коэффициент эксцесса | -0.23578 | -0.12825 | 0.10753 | -0.83846 |

Таблица 4: Таблица сранения рассчитанных характеристик с теоретическими значениями

**Задание 2:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 3 | 4 | 1 | 1 | 2 | 1 | 1 | 1 | 1 |
| 1 | 2 | 3 | 4 | 1 | 1 | 2 | 1 | 1 | 1 |
| 4 | 1 | 2 | 1 | 2 | 1 | 1 | 3 | 1 | 1 |
| 1 | 4 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 2 |
| 1 | 1 | 1 | 1 | 1 | 1 | 3 | 1 | 3 | 1 |
| 1 | 1 | 2 | 2 | 1 | 2 | 2 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| 1 | 1 | 2 | 2 | 1 | 3 | 1 | 3 | 1 | 1 |
| 1 | 1 | 1 | 1 | 2 | 2 | 1 | 1 | 1 | 2 |
| 1 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 2 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 3 |
| 1 | 1 | 1 | 1 | 1 | 1 | 3 | 2 | 1 | 1 |
| 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| 1 | 1 | 2 | 1 | 1 | 1 | 2 | 1 | 1 | 1 |
| 1 | 1 | 4 | 2 | 1 | 1 | 1 | 3 | 4 | 1 |
| 2 | 1 | 1 | 1 | 3 | 4 | 1 | 2 | 2 | 1 |
| 1 | 2 | 1 | 1 | 2 | 1 | 1 | 1 | 2 | 1 |
| 1 | 1 | 5 | 2 | 2 | 4 | 2 | 1 | 1 | 2 |
| 2 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 2 | 1 |
| 1 | 2 | 1 | 2 | 1 | 1 | 1 | 2 | 1 | 1 |

Таблица 5: 200 выборок геометрического распределения

Геометрическое распределение

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 1 | 137.00000 | 0.68500 | 0.68500 |
| 2 | 44.00000 | 0.22000 | 0.90500 |
| 3 | 11.00000 | 0.05500 | 0.96000 |
| 4 | 7.00000 | 0.03500 | 0.99500 |
| 5 | 1.00000 | 0.00500 | 1.00000 |
| Total | 200 | 1 |  |

Таблица 6: Статистический ряд

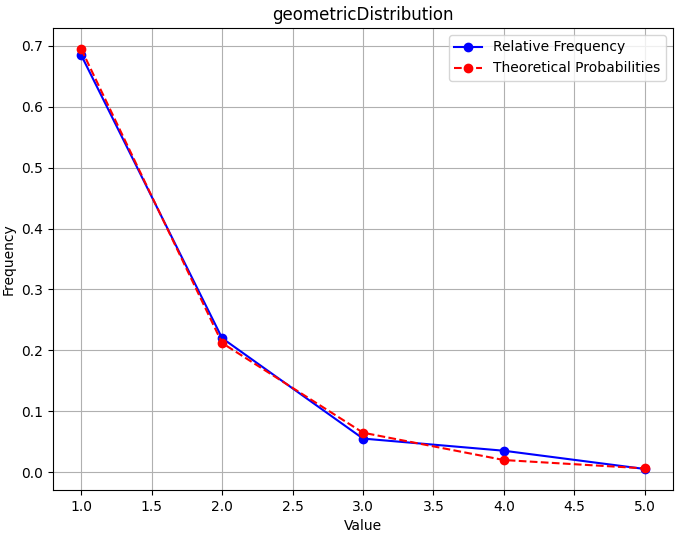


Рисунок 3: Полигон относительных частот

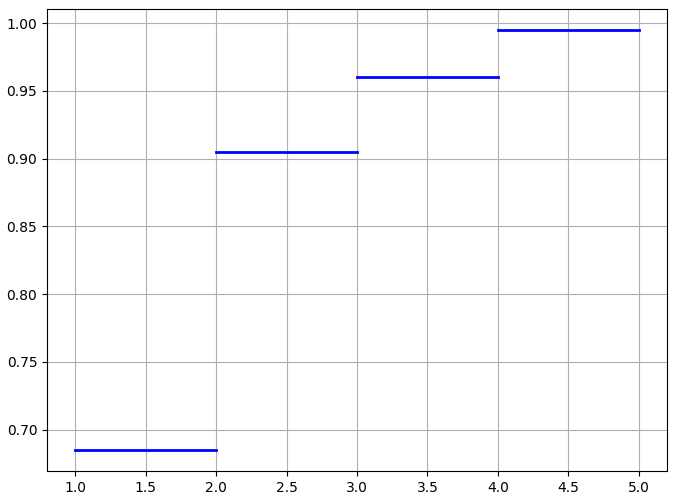


Рисунок 4: График эмпирической функции распределения

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 1 | 0.68500 | 0.69500 | 0.01000 |
| 2 | 0.22000 | 0.21197 | 0.00803 |
| 3 | 0.05500 | 0.06465 | 0.00965 |
| 4 | 0.03500 | 0.01972 | 0.01528 |
| 5 | 0.00500 | 0.00601 | 0.00101 |
| Total | 1.00000 | 1.00000 | 0.01528 |

Таблица 7: Таблица сравнения относительных частот и теоретических вероятностей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название показателя** | **Выборочное значение** | **Теоретическое значение** | **Абсолютное отклонение** | **Относительное отклонение** |
| Среднее значение | 1.45500 | 1.43885 | 0.01615 | 0.01123 |
| Дисперсия | 0.62798 | 0.63144 | 0.00346 | 0.00548 |
| Среднее квадратичное отклонение | 0.79245 | 0.79463 | 0.00218 | 0.00275 |
| Мода | 1.00000 | 1.00000 | 0.00000 | 0.00000 |
| Медиана | 3.00000 | 1.00000 | 2.00000 | 2.00000 |
| Коэффициент асимметрии | 1.95648 | 2.36298 | 0.40651 | 0.17203 |
| Коэффициент эксцесса | 3.64410 | 7.58369 | 3.93959 | 0.51948 |

Таблица 8: Таблица сравнения рассчитанных характеристик с теоретическими значениями

**Задание 3:**

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

Таблица 9: 200 выборок распределения Пуассона

Распределение Пуассона

|  |  |  |
| --- | --- | --- |
| **Интервалы** |  |  |
| (0.00000, 0.57845) | 126.00000 | 0.63000 |
| (0.57845, 1.15689) | 57.00000 | 0.28500 |
| (1.15689, 1.73534) | 0.00000 | 0.00000 |
| (1.73534, 2.31378) | 14.00000 | 0.07000 |
| (2.31378, 2.89223) | 0.00000 | 0.00000 |
| (2.89223, 3.47067) | 1.00000 | 0.00500 |
| (3.47067, 4.04912) | 1.00000 | 0.00500 |
| (4.04912, 5.00000) | 0.00000 | 0.00000 |
|  | 200 | 1 |

Таблица 10: Интервальный Ряд

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0.28922 | 126.00000 | 0.63000 |
| 0.86767 | 57.00000 | 0.28500 |
| 1.44611 | 0.00000 | 0.00000 |
| 2.02456 | 14.00000 | 0.07000 |
| 2.60300 | 0.00000 | 0.00000 |
| 3.18145 | 1.00000 | 0.00500 |
| 3.75990 | 1.00000 | 0.00500 |
| 4.52456 | 0.00000 | 0.00000 |
|  | 200 | 1 |

Таблица 11: Ассоциированный статистический ряд

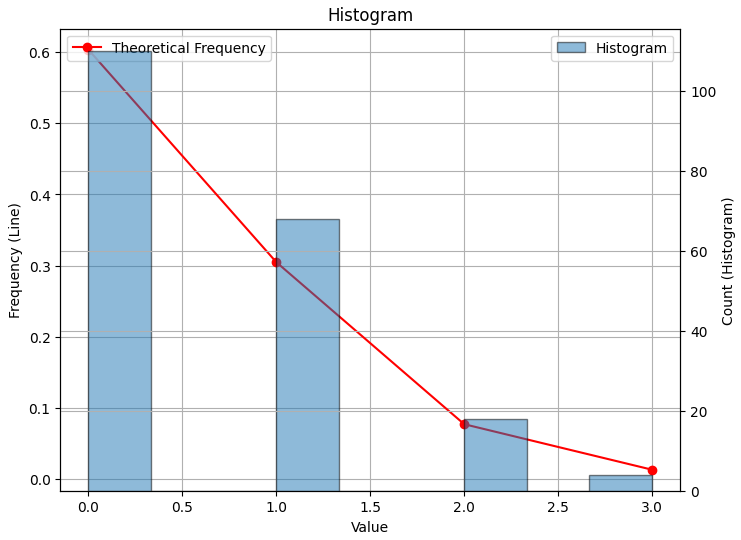


Рисунок 5: Гистограмма относительных частот с наложенным на нее и выделенным красным цветом графиком плотности показательного распределения

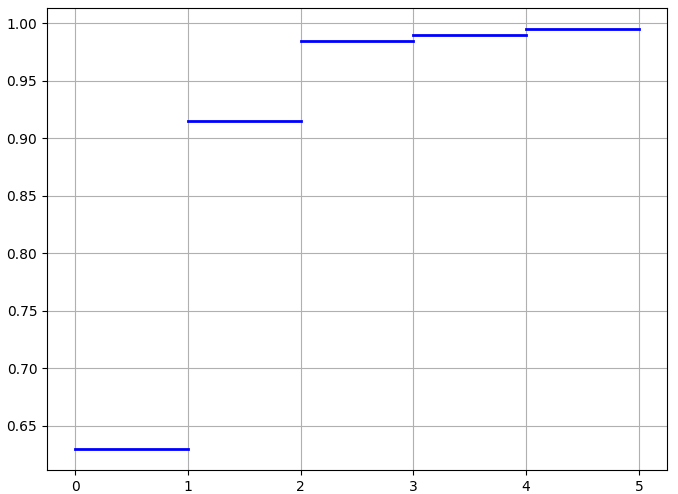


Рисунок 6: График эмфирической функции распределения

|  |  |  |  |
| --- | --- | --- | --- |
| **Интервалы:** |  |  |  |
| (0.00000, 0.57845) | 0.63000 | 0.60351 | 0.02649 |
| (0.57845, 1.15689) | 0.28500 | 0.30477 | 0.01977 |
| (1.15689, 1.73534) | 0.00000 | 0.00000 | 0.00000 |
| (1.73534, 2.31378) | 0.07000 | 0.07695 | 0.00695 |
| (2.31378, 2.89223) | 0.00000 | 0.00000 | 0.00000 |
| (2.89223, 3.47067) | 0.00500 | 0.01295 | 0.00795 |
| (3.47067, 4.04912) | 0.00500 | 0.00164 | 0.00336 |
| (4.04912, 5.00000) | 0.00000 | 0.00000 | 0.00000 |
|  | 1 | 1 | 0.026494 |

Таблица 12: Таблица сравнения относительных частот и теоретических вероятностей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название показателя** | **Выборочное значение** | **Теоретическое значение** | **Абсолютное отклонение** | **Относительное отклонение** |
| Среднее значение | 0.48500 | 0.50500 | 0.02000 | 0.03960 |
| Дисперсия | 0.57978 | 0.50500 | 0.07478 | 0.14807 |
| Среднее квадратичное отклонение | 0.76143 | 0.71063 | 0.05080 | 0.07148 |
| Мода | 0.00000 | 0.00000 | 0.00000 | nan |
| Медиана | 2.50000 | 0.00000 | 2.50000 | nan |
| Коэффициент асимметрии | 2.19125 | 1.40720 | 0.78405 | 0.55717 |
| Коэффициент эксцесса | 7.30751 | 1.98020 | 5.32732 | 2.69029 |

Таблица 13: Таблица сравнения рассчитанных характеристик с теоретическими значениями

# 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.-704с.
3. Гмурман В. Е. Теория вероятностей и математическая статистика.-М.: Юрайт, 2013.-479с.
4. Гмурман В. Е. Руководство к решению задач по теории вероятностей и математической статистике. - М.:Юрайт,2013.-404с.
5. Емельянов Г.В. Скитович В.П. Задачник по теории вероятностей и математической статистике.-СПб.: Лань, 2007.-336с.
6. Кибзун А.И., Горяинова Е.Р., Наумов А.В. Теория вероятностей и математическая статистика. Базовый курс с примерами и задачами. Учебное пособие - М.: ФИЗМАТЛИТ, 2005.-232с.
7. Кобзарь А.И. Прикладная математическая статистика: для инженеров и научных работников - М.: ФИЗМАТЛИТ, 2006.-816с.
8. Монсик В.Б., Скрынников А.А. Вероятность и статистика. - М.: БИНОМ, 2015-384с.
9. Сборник задач по теории вероятностей, математической статистике и теории случайных функций: Учеб. пособие для вузов / Под ред. А. А. Свешникова. - СПб.: Лань, 2012. - 472с.
10. Письменный Д.Т. Конспект лекций по теории вероятностей, математической статистике и случайным процессам: учеб. пособие для вузов.-М.: Айрис-пресс,2013.-288с.

# Приложение

import numpy as np

import matplotlib.pyplot as plt

from scipy import stats

import pandas as pd

import math

N = 200 #Sample

V = 165 #Variant

n = 5 + V % 20

p = 0.2 + 0.003 \* V

lamda = 1 + ((-1) \*\* V) \* (V \* 0.003)

seed = 420

rng = np.random.default\_rng(seed=seed)

def task(X\_distribution, name\_distribution):

  #1

  counts = np.unique(X\_distribution, return\_counts=True)

  X = counts[0]

  freq = counts[1]

  rel\_freq = counts[1] / N

  cum\_freq = np.cumsum(rel\_freq)

  total\_freq = freq.sum(axis=0)

  total\_rel\_freq = rel\_freq.sum(axis=0)

  intervals = None

  if name\_distribution == 'binomial' or name\_distribution == 'geometric' or name\_distribution == 'poisson':

    total\_row = {

      'X': 'Total',

      'Frequency': total\_freq,

      'Relative Frequency': total\_rel\_freq,

      'Cumulative Frequency': ''

    }

    df\_freq = pd.DataFrame({

        'X': counts[0],

        'Frequency': [f"{val:.5f}" for val in freq],

        'Relative Frequency': [f"{val:.5f}" for val in rel\_freq],

        'Cumulative Frequency': [f"{val:.5f}" for val in cum\_freq]

        })

    df\_freq.loc[len(df\_freq)] = total\_row

    print(df\_freq)

    df\_freq.to\_excel(name\_distribution + '\_stat.xlsx')

  if name\_distribution == 'poisson':

        step = 1 + np.log2(N)

        distance\_between\_step = (X.max() - X.min()) / step

        intervals = np.array([X.min() + i \* distance\_between\_step for i in range(int(step))])

        intervals = np.append(intervals, X.max())

        intervals\_freq = np.zeros(len(intervals)-1)

        intervals\_relative\_freq = np.zeros(len(intervals)-1)

        for i in range(len(intervals)-1):

            for j in range(len(X)):

                if intervals[i] <= X[j] < intervals[i+1]:

                    intervals\_freq[i] += freq[j]

                    intervals\_relative\_freq[i] += rel\_freq[j]

        total\_row = {'Frequency': total\_freq, 'Relative Frequency': total\_rel\_freq}

        df\_freq\_intervals = pd.DataFrame({'Interval: ': zip([f"{val:.5f}" for val in intervals[:-1]], [f"{val:.5f}" for val in intervals[1:]]), 'Frequency': [f"{val:.5f}" for val in intervals\_freq], 'Relative Frequency': [f"{val:.5f}" for val in intervals\_relative\_freq]})

        df\_freq\_intervals.loc[len(df\_freq\_intervals)] = total\_row

        print(df\_freq\_intervals)

        df\_freq\_intervals.to\_excel(name\_distribution + '\_intervals.xlsx')

        middle = np.array([(intervals[i] + intervals[i+1]) / 2 for i in range(len(intervals)-1)])

        df\_freq\_middle = pd.DataFrame({'Middle': [f"{val:.5f}" for val in middle], 'Frequency': [f"{val:.5f}" for val in intervals\_freq], 'Relative Frequency': [f"{val:.5f}" for val in intervals\_relative\_freq]})

        df\_freq\_middle.loc[len(df\_freq\_intervals)] = total\_row

        print(df\_freq\_middle)

        df\_freq\_middle.to\_excel(name\_distribution + '\_middle.xlsx')

  #2

  if name\_distribution == 'binomial':

    theo\_freq = np.array([(math.comb(n, k) \* (p \*\* k) \* ((1-p) \*\* (n-k))) for k in X])

  elif name\_distribution == 'geometric':

    theo\_freq = np.array([(p \* ((1-p) \*\* (k-1))) for k in counts[0]])

  elif name\_distribution == 'poisson':

    theo\_freq = np.array([(lamda \*\* k) \* np.exp(-lamda) / math.factorial(k) for k in counts[0]])

    intervals\_theoretical\_relative\_freq = np.zeros(len(intervals)-1)

    for i in range(len(intervals)-1):

        for j in range(len(X)):

            if intervals[i] <= X[j] < intervals[i+1]:

                intervals\_theoretical\_relative\_freq[i] += theo\_freq[j]

  #plot

  if name\_distribution == 'binomial' or name\_distribution == 'geometric':

    plt.figure(figsize = (8,6))

    plt.plot(X, rel\_freq, marker = 'o', linestyle = '-', color = 'b', label = 'Relative Frequency')

    plt.plot(X, theo\_freq, marker = 'o', linestyle = '--', color = 'red', label = 'Theoretical Probabilities')

    plt.xlabel("Value")

    plt.ylabel("Frequency")

    plt.title(name\_distribution + "Distribution")

    plt.legend()

    plt.grid(True)

    plt.show()

    #3

    plt.figure(figsize = (8,6))

    for i in range(0, len(counts[0])-1):

      plt.plot([counts[0][i], counts[0][i+1]], [cum\_freq[i], cum\_freq[i]], color='b', linestyle='-', linewidth=2)

    plt.grid(True)

    plt.show()

  if name\_distribution == 'poisson':

    fig, ax1 = plt.subplots(figsize=(8, 6))

    #ax1.plot(counts[0], relative\_freq, marker='o', linestyle='-', color='b', label='Relative Frequency')

    ax1.plot(counts[0], theo\_freq, marker='o', linestyle='-', color='r', label='Theoretical Frequency')

    ax1.set\_xlabel('Value')

    ax1.set\_ylabel('Frequency (Line)')

    ax1.set\_title('Histogram')

    ax1.legend(loc='upper left')

    ax1.grid(True)

    ax2 = ax1.twinx()

    ax2.hist(X\_distribution, bins='sturges', edgecolor='black', alpha=0.5, label='Histogram')

    ax2.set\_ylabel('Count (Histogram)')

    ax2.legend(loc='upper right')

    plt.show()

    plt.figure(figsize = (8,6))

    for i in range(0, len(counts[0])-1):

      plt.plot([counts[0][i], counts[0][i+1]], [cum\_freq[i], cum\_freq[i]], color='b', linestyle='-', linewidth=2)

    plt.grid(True)

    plt.show()

  #4

  mean = np.array([X[i] \* rel\_freq[i] for i in range(len(X))]).sum()

  print("Sample mean = ", f"{mean: .5f}")

  #5

  variance = np.array([((X[i] - mean) \*\* 2) \* rel\_freq[i] for i in range(len(X))]).sum()

  print("Sample variance = ", f"{variance: .5f}")

  #6

  deviation = np.sqrt(variance)

  print("Sample standard deviation = ", f"{deviation: .5f}")

  #7

  modes = X[np.argwhere(freq == np.amax(freq))].flatten().tolist()

  mode = (modes[0] + modes[len(modes) - 1]) / 2

  print("Mode = ", f"{mode: .5f}")

  #8

  if len(X) % 2 == 0:

      median = (X[int(len(X)/2)] + X[int(len(X)/2) - 1]) / 2

  else:

      if 0.5 in X.tolist():

          median = X[cum\_freq.tolist().index(0.5)]

      else:

          left = -1

          for element in cum\_freq:

              if element < 0.5:

                  left += 1

              else:

                  right = left + 1

                  break

          median = (X[left] + X[right]) / 2

  print("Median = ", f"{median: .5f}")

  #9

  def sample\_k\_moment\_around\_mean(k, mean):

      return np.array([(X[i] - mean) \*\* k \* rel\_freq[i] for i in range(len(X))]).sum()

  sample\_skeness = sample\_k\_moment\_around\_mean(3, mean) / deviation \*\* 3

  print("Sample skewness = ", f"{sample\_skeness: .5f}")

  sample\_kurtosis = sample\_k\_moment\_around\_mean(4, mean) / deviation \*\* 4 - 3

  print("Sample kurtosis = ", f"{sample\_kurtosis: .5f}")

  abs\_diff\_freq = np.abs(theo\_freq - rel\_freq)

  freq\_compare = pd.DataFrame({

      'X': X,

      'Relative Frequency': [f"{val:.5f}" for val in rel\_freq],

      'Theoretical Frequency': [f"{val:.5f}" for val in theo\_freq],

      'Absolute Difference': [f"{val:.5f}" for val in abs\_diff\_freq]

      })

  if name\_distribution == 'poisson':

    abs\_diff\_freq\_intervals = np.abs(intervals\_theoretical\_relative\_freq - intervals\_relative\_freq)

    freq\_compare\_intervals = pd.DataFrame({

        'Interval: ': zip([f"{val:.5f}" for val in intervals[:-1]], [f"{val:.5f}" for val in intervals[1:]]),

        'Relative Frequency': [f"{val:.5f}" for val in intervals\_relative\_freq],

        'Theoretical Frequency': [f"{val:.5f}" for val in intervals\_theoretical\_relative\_freq],

        'Absolute Difference': [f"{val:.5f}" for val in abs\_diff\_freq\_intervals]

        })

    total\_row\_compare\_intervals = {

        'Relative Frequency': total\_rel\_freq,

        'Theoretical Frequency': 1,

        'Absolute Difference': np.max(abs\_diff\_freq)

    }

    freq\_compare\_intervals.loc[len(freq\_compare\_intervals)] = total\_row\_compare\_intervals

    freq\_compare\_intervals.to\_excel(name\_distribution + '\_intervals\_compare.xlsx')

  total\_row\_compare = {

      'X': 'Total',

      'Relative Frequency': f"{total\_rel\_freq: .5f}",

      'Theoretical Frequency': f"{1: .5f}",

      'Absolute Difference': f"{np.max(abs\_diff\_freq): .5f}"

      }

  freq\_compare.loc[len(freq\_compare)] = total\_row\_compare

  print(freq\_compare)

  freq\_compare.to\_excel(name\_distribution + '\_compare.xlsx')

  #10

  if name\_distribution == 'binomial':

    theo\_mean = n \* p

    theo\_var = n \* p \* (1 - p)

    theo\_deviation = np.sqrt(n \* p \* (1 - p))

    theo\_skewness = ((1-p)-p) / np.sqrt(n \* p \* (1 - p))

    theo\_kurtois = (1 - 6 \* p \* (1 - p)) / (n \* p \* (1 - p))

    theo\_mode = np.floor((n + 1) \* p)

    theo\_median = np.floor(n \* p)

    theo\_values = np.array([theo\_mean, theo\_var, theo\_deviation, theo\_skewness, theo\_kurtois, theo\_mode, theo\_median])

    real\_values = np.array([mean, variance, deviation, sample\_skeness, sample\_kurtosis, mode, median])

    abs\_differences = np.array([abs(mean - theo\_mean), abs(variance - theo\_var), abs(deviation - theo\_deviation), abs(sample\_skeness - theo\_skewness), abs(sample\_kurtosis - theo\_kurtois), abs(mode - theo\_mode), abs(median - theo\_median)])

    rel\_differences = np.array([abs\_differences[i] / theo\_values[i] for i in range(len(theo\_values))])

    char\_combine = pd.DataFrame({'Characteristic': ['Mean', 'Variance', 'Deviation', 'Skewness', 'Kurtosis', 'Mode', 'Median'],

                                'Sample': [f"{val:.5f}" for val in real\_values],

                                'Theoretical': [f"{val:.5f}" for val in theo\_values],

                                'Absolute Difference': [f"{val:.5f}" for val in abs\_differences],

                                'Relative Difference': [f"{val:.5f}" for val in rel\_differences]

                                })

    char\_combine.to\_excel(name\_distribution + '\_char.xlsx')

  elif name\_distribution == 'geometric':

    theo\_mean = 1 / p

    theo\_var = (1 - p) / (p \*\* 2)

    theo\_deviation = np.sqrt((1 - p) / (p \*\* 2))

    theo\_skewness = (2 - p) / np.sqrt(1 - p)

    theo\_kurtois = 6 + p \*\* 2 / (1 - p)

    theo\_mode = 1

    theo\_median = np.round((-1 / np.log2(1 - p)))

    theo\_values = np.array([theo\_mean, theo\_var, theo\_deviation, theo\_skewness, theo\_kurtois, theo\_mode, theo\_median])

    real\_values = np.array([mean, variance, deviation, sample\_skeness, sample\_kurtosis, mode, median])

    abs\_differences = np.array([abs(real\_values[i] - theo\_values[i]) for i in range(len(theo\_values))])

    rel\_differences = np.array([abs\_differences[i] / theo\_values[i] for i in range(len(theo\_values))])

    char\_combine = pd.DataFrame({'Characteristic': ['Mean', 'Variance', 'Deviation', 'Skewness', 'Kurtosis', 'Mode', 'Median'],

                                'Sample': [f"{val:.5f}" for val in real\_values],

                                'Theoretical': [f"{val:.5f}" for val in theo\_values],

                                'Absolute Difference': [f"{val:.5f}" for val in abs\_differences],

                                'Relative Difference': [f"{val:.5f}" for val in rel\_differences]

                                })

    char\_combine.to\_excel(name\_distribution + '\_char.xlsx')

  elif name\_distribution == 'poisson':

    theo\_mean = lamda

    theo\_var= lamda

    theo\_deviation = np.sqrt(lamda)

    theo\_skewness = 1 / np.sqrt(lamda)

    theo\_kurtois = 1 / lamda

    theo\_mode = np.floor(lamda)

    theo\_median = np.floor(lamda + 1/3 - 0.02/lamda)

    theo\_values = np.array([theo\_mean, theo\_var, theo\_deviation, theo\_skewness, theo\_kurtois, theo\_mode, theo\_median])

    real\_values = np.array([mean, variance, deviation, sample\_skeness, sample\_kurtosis, mode, median])

    abs\_differences = np.array([abs(real\_values[i] - theo\_values[i]) for i in range(len(theo\_values))])

    rel\_differences = []

    for i in range(len(theo\_values)):

        if theo\_values[i] == 0:

            rel\_differences.append(np.nan)  # or "N/A"

        else:

            rel\_differences.append(abs\_differences[i] / theo\_values[i])

    rel\_differences = np.array(rel\_differences)

    char\_combine = pd.DataFrame({'Characteristic': ['Mean', 'Variance', 'Deviation', 'Skewness', 'Kurtosis', 'Mode', 'Median'],

                                'Sample': [f"{val:.5f}" for val in real\_values],

                                'Theoretical': [f"{val:.5f}" for val in theo\_values],

                                'Absolute Difference': [f"{val:.5f}" for val in abs\_differences],

                                'Relative Difference': [f"{val:.5f}" for val in rel\_differences]

                                })

    char\_combine.to\_excel(name\_distribution + '\_char.xlsx')

  print(char\_combine)