|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  | | --- | --- | --- | |  |  |  | | 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»

Тема: \_\_\_\_\_\_\_Первичная обработка выборки из\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_дискретной генеральной совокупности\_\_\_\_\_\_\_\_\_\_

Выполнил:

Студент 3-го курса

Маргулис А.П.

Группа: КМБО-01-17

МОСКВА 2020

**Задание**

**Задание 1.** Получить выборку, сгенерировав 100 псевдослучайных чисел распределенных по биномиальному закону с параметрами n и p .

**Задание 2.** Получить выборку, сгенерировав 100 псевдослучайных чисел распределенных по геометрическому закону с параметром p .

**Задание 3.** Получить выборку, сгенерировав 100 псевдослучайных чисел распределенных по закону Пуассона с параметром λ .

Следуя указаниям для всех выборок построить:

1) статистический ряд;

2) полигон относительных частот;

3) эмпирическую функцию распределения.

Найти:

1) выборочное среднее;

2) выборочную дисперсию;

3) выборочное среднее квадратическое отклонение;

4) моду;

5) медиану;

6) выборочный коэффициент асимметрии;

7) выборочный коэффициент эксцесса.

Все вычисления проводить с точностью до 0,00001 .

**Краткие теоретические сведения**

**Биномиальное распределение:**

* ряд распределения:
* математическое ожидание (среднее значение): *np*
* дисперсия: *npq*
* среднее квадратичное отклонение:
* мода: *[(n+1)p]*, если *(n+1)p* – дробное; *(n+1)p*–*0,5* , если *(n+1)p* – целое
* медиана: *Round(np)*
* коэффициент асимметрии:
* коэффициент эксцесса:

**Геометрическое распределение:**

* ряд распределения:
* математическое ожидание (среднее значение): , где
* дисперсия: , где
* среднее квадратичное отклонение:
* мода: *0*
* медиана: [, если – дробное; , если – целое
* коэффициент асимметрии:
* коэффициент эксцесса:

**Распределение Пуассона:**

* ряд распределения:
* математическое ожидание (среднее значение):
* дисперсия:
* среднее квадратичное отклонение:
* мода:
* медиана:
* коэффициент асимметрии:
* коэффициент эксцесса:

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

* **Эмпирическая функция распределения**
* **Выборочное среднее**
* **Выборочная дисперсия**
* **Выборочный момент k-ого порядка**
* **Выборочный центральный момент k-ого порядка**
* **Выборочное среднее кврадратическое отклонение**
* **Выборочная медиана**
* **Выборочная мода** – это значение , которому соответствует максимальная частота.
* **Выборочный коэффициент асимметрии**
* **Выборочный коэффициент эксцесса**

**Средства языка Python**

В программе расчёта используются следующие средства языка Octave:

• функция binomial(n, p, size) - возвращает матрицу случайных значений из биномиального распределения с параметрами n и p , где n есть число испытаний, p – вероятность успеха, size – количество элементов.

• функция geometric(p, size) - возвращает матрицу случайных значений из геометрического распределения с параметром р, size – количество элементов.

• poisson(λ, size) - возвращает матрицу случайных значений из распределения Пуассона с параметром λ, size – количество элементов.

• sorted(*x*) - возвращает копию *х* с элементами, расположенными в порядке возрастания.

• Counter(x) – возвращает словарь, где ключи – это элементы, а значения – их количество.

• plot(x,y) - построение графика по координатам x,y

**Результаты расчетов**

**Задание 1 (биномиальное распределение)**

n = 20, p = 0,375.

Неупорядоченная выборка (200 чисел):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 11 | 8 | 10 | 10 | 3 | 8 | 10 | 5 | 9 |
| 6 | 8 | 5 | 7 | 6 | 3 | 8 | 7 | 8 | 12 |
| 11 | 4 | 3 | 4 | 11 | 7 | 6 | 10 | 7 | 11 |
| 12 | 9 | 5 | 6 | 4 | 8 | 10 | 5 | 9 | 9 |
| 6 | 10 | 10 | 4 | 6 | 7 | 11 | 7 | 5 | 6 |
| 9 | 5 | 6 | 5 | 6 | 9 | 7 | 6 | 10 | 10 |
| 10 | 8 | 7 | 9 | 8 | 6 | 8 | 9 | 11 | 9 |
| 8 | 4 | 8 | 9 | 9 | 9 | 9 | 8 | 8 | 6 |
| 7 | 8 | 7 | 8 | 13 | 6 | 5 | 8 | 11 | 6 |
| 7 | 6 | 9 | 10 | 8 | 11 | 10 | 10 | 7 | 11 |
| 11 | 8 | 5 | 12 | 6 | 6 | 5 | 11 | 8 | 7 |
| 7 | 10 | 9 | 7 | 6 | 4 | 6 | 4 | 8 | 7 |
| 10 | 6 | 10 | 5 | 7 | 8 | 10 | 4 | 10 | 10 |
| 15 | 8 | 8 | 6 | 6 | 7 | 6 | 6 | 9 | 6 |
| 7 | 9 | 5 | 9 | 5 | 8 | 6 | 1 | 8 | 12 |
| 6 | 8 | 8 | 7 | 5 | 5 | 7 | 8 | 13 | 7 |
| 7 | 8 | 9 | 6 | 5 | 9 | 8 | 9 | 7 | 8 |
| 11 | 8 | 9 | 5 | 8 | 9 | 2 | 7 | 8 | 10 |
| 9 | 6 | 3 | 8 | 10 | 11 | 7 | 9 | 11 | 6 |
| 6 | 9 | 10 | 5 | 8 | 6 | 7 | 2 | 13 | 7 |

Упорядоченная выборка (200 чисел):

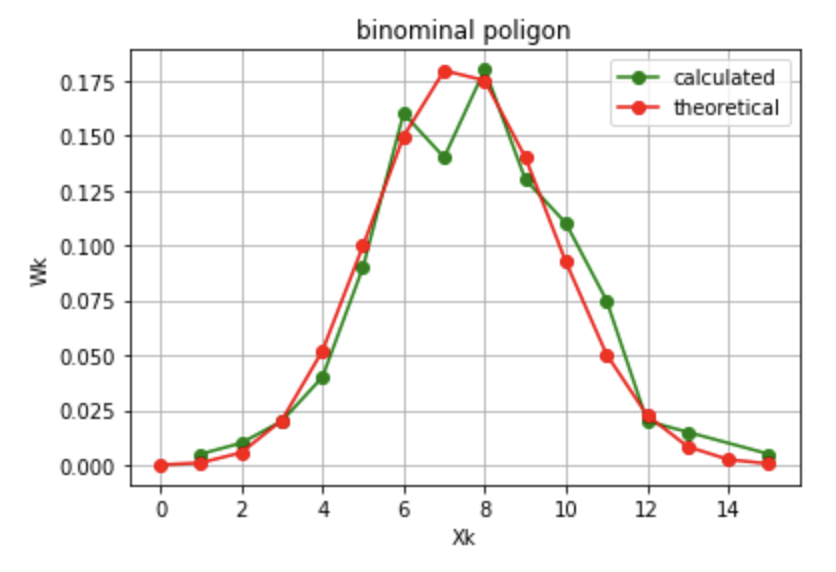
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 2 | 3 | 3 | 3 | 3 | 4 | 4 | 4 |
| 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |
| 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |
| 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |
| 6 | 6 | 6 | 6 | 6 | 7 | 7 | 7 | 7 | 7 |
| 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |
| 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |
| 7 | 7 | 7 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 9 |
| 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| 9 | 9 | 9 | 9 | 9 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 11 | 11 | 11 |
| 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| 11 | 11 | 12 | 12 | 12 | 12 | 13 | 13 | 13 | 15 |

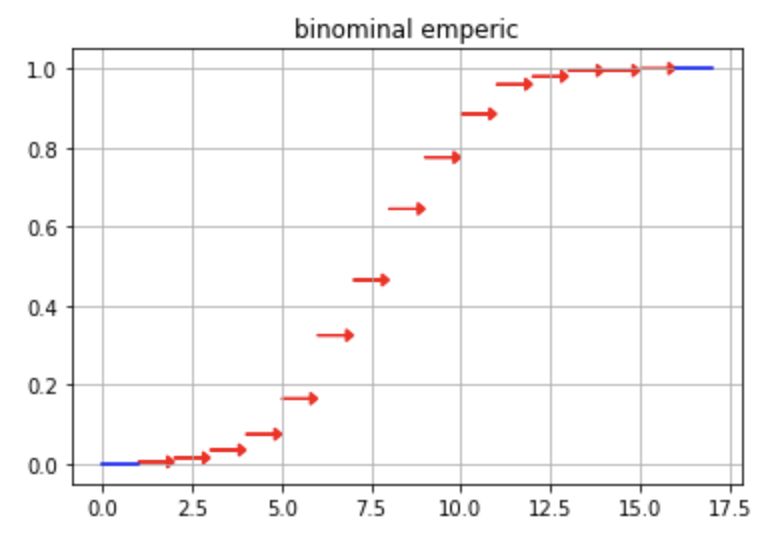
Статистический ряд:

|  |  |  |  |
| --- | --- | --- | --- |
| **Xk** | **Nk** | **Wk** | **Sk** |
| 1 | 1 | 0,005 | 0,005 |
| 2 | 2 | 0,01 | 0,015 |
| 3 | 4 | 0,02 | 0,035 |
| 4 | 8 | 0,04 | 0,075 |
| 5 | 18 | 0,09 | 0,165 |
| 6 | 32 | 0,16 | 0,325 |
| 7 | 28 | 0,14 | 0,465 |
| 8 | 36 | 0,18 | 0,645 |
| 9 | 26 | 0,13 | 0,775 |
| 10 | 22 | 0,11 | 0,885 |
| 11 | 15 | 0,075 | 0,96 |
| 12 | 4 | 0,02 | 0,98 |
| 13 | 3 | 0,015 | 0,995 |
| 15 | 1 | 0,005 | 1 |

Эмпирическая функция распределения и ее график:

=





Выборочное среднее = 7.68

Выборочная дисперсия = 5.417600000000007

Выборочный момент 3-ого порядка = 578.1600000000001

Выборочный момент 4-ого порядка = 5496.320000000001

Выборочное среднее квадратическое отклонение = 2.3275738441561864

Выборочная мода = 8

Выборочная медиана = 8

Выборочный коэффициент асимметрии = 0.0022241805841271612

Выборочный коэффициент эксцесса = -2.896367624820266

**Задание 2 (геометрическое распределение)**

p = 0,375

Неупорядоченная выборка (200 чисел):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 0 | 1 | 0 | 0 | 4 | 0 | 0 | 1 |
| 0 | 0 | 0 | 3 | 0 | 1 | 0 | 0 | 2 | 1 |
| 0 | 0 | 8 | 0 | 3 | 0 | 0 | 2 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 2 | 8 | 0 | 0 | 0 |
| 2 | 0 | 1 | 3 | 2 | 2 | 1 | 0 | 2 | 1 |
| 0 | 5 | 1 | 0 | 0 | 0 | 0 | 0 | 6 | 1 |
| 2 | 2 | 2 | 2 | 0 | 1 | 0 | 2 | 3 | 3 |
| 1 | 0 | 1 | 1 | 0 | 2 | 1 | 5 | 1 | 4 |
| 4 | 2 | 1 | 2 | 1 | 4 | 1 | 1 | 0 | 0 |
| 2 | 7 | 2 | 4 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 8 | 0 | 5 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 2 | 0 | 0 | 0 |
| 0 | 6 | 0 | 2 | 0 | 3 | 2 | 0 | 5 | 1 |
| 1 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 11 | 1 | 4 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 3 | 3 | 2 | 1 | 2 | 0 | 1 | 3 | 1 |
| 1 | 0 | 2 | 1 | 0 | 0 | 3 | 1 | 8 | 5 |
| 2 | 0 | 0 | 0 | 1 | 4 | 0 | 2 | 0 | 1 |
| 4 | 0 | 0 | 4 | 2 | 3 | 4 | 3 | 0 | 1 |
| 1 | 1 | 0 | 3 | 6 | 2 | 0 | 0 | 2 | 1 |

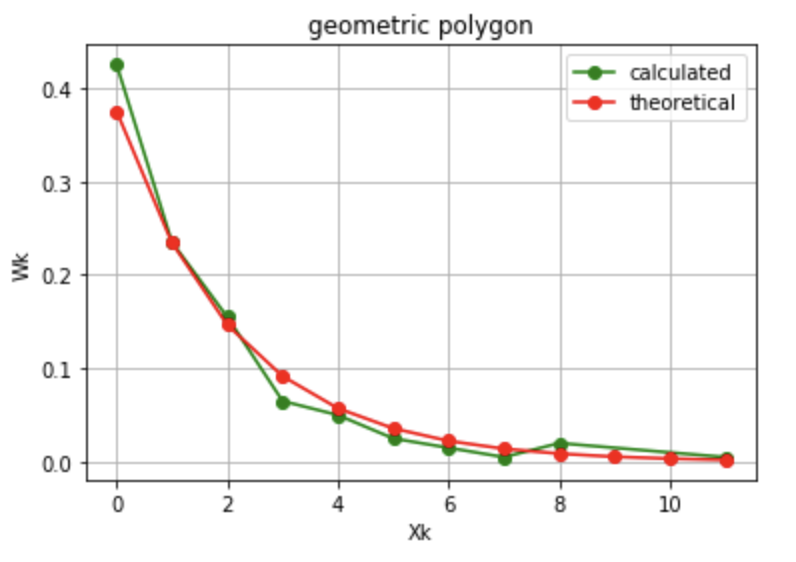
Упорядоченная выборка (200 чисел):

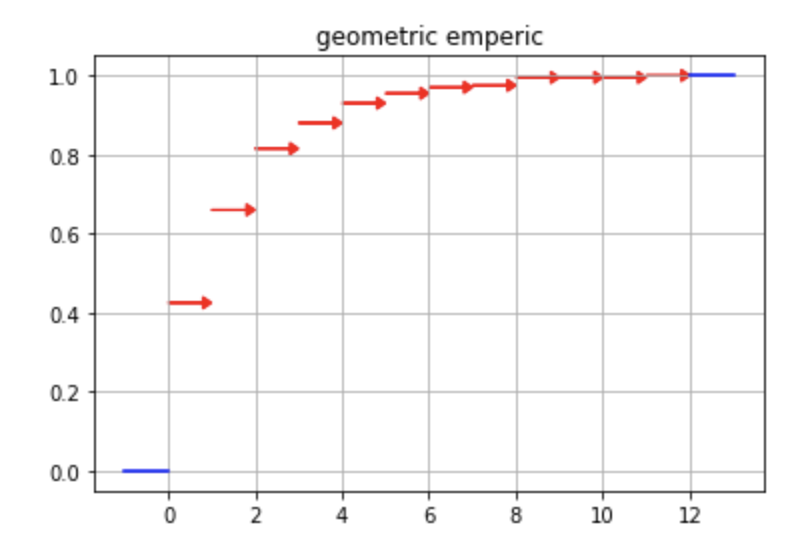
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 |
| 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 |
| 5 | 6 | 6 | 6 | 7 | 8 | 8 | 8 | 8 | 11 |

Статистический ряд:

|  |  |  |  |
| --- | --- | --- | --- |
| **Xk** | **Nk** | **Wk** | **Sk** |
| 0 | 85 | 0,425 | 0,425 |
| 1 | 47 | 0,235 | 0,66 |
| 2 | 31 | 0,155 | 0,815 |
| 3 | 13 | 0,065 | 0,88 |
| 4 | 10 | 0,05 | 0,93 |
| 5 | 5 | 0,025 | 0,955 |
| 6 | 3 | 0,015 | 0,97 |
| 7 | 1 | 0,005 | 0,975 |
| 8 | 4 | 0,02 | 0,995 |
| 11 | 1 | 0,005 | 1 |

Эмпирическая функция распределения и ее график:





Выборочное среднее = 1.4049999999999998

Выборочная дисперсия = 3.560975000000001

Выборочный момент 3-ого порядка = 31.405

Выборочный момент 4-ого порядка = 222.97499999999997

Выборочное среднее квадратическое отклонение = 1.8870545832063261

Выборочная мода = 0

Выборочная медиана = 1

Выборочный коэффициент асимметрии = 0.30167104769736

Выборочный коэффициент эксцесса = -2.3759431804499984

**Задание 3 (распределение Пуассона)**

=0,65

Неупорядоченная выборка (200 чисел):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 2 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 3 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 2 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 2 | 0 |
| 0 | 1 | 2 | 0 | 2 | 0 | 0 | 0 | 0 | 0 |
| 0 | 2 | 0 | 0 | 1 | 2 | 0 | 0 | 0 | 3 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 2 | 0 | 1 | 2 | 0 |
| 1 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 3 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 2 | 0 | 0 | 1 | 0 |
| 1 | 2 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 3 | 1 | 0 | 0 | 3 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 |
| 0 | 1 | 3 | 0 | 2 | 1 | 0 | 1 | 0 | 0 |

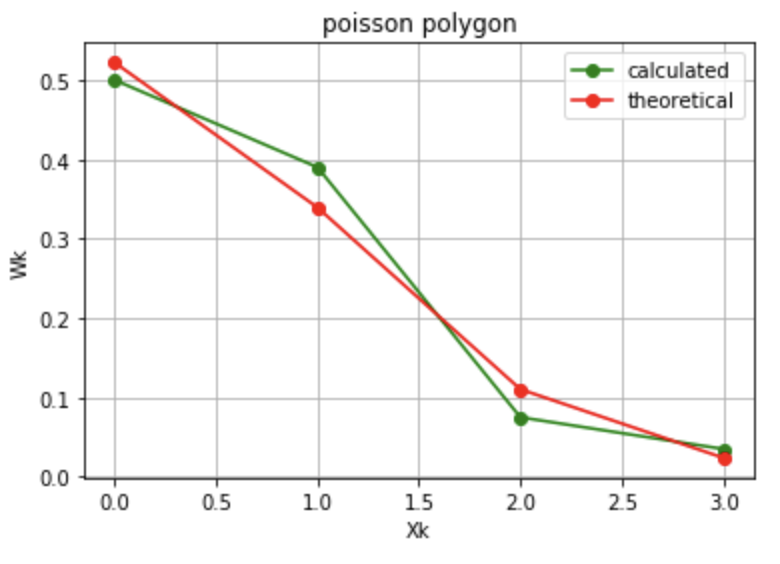
Упорядоченная выборка (200 чисел):

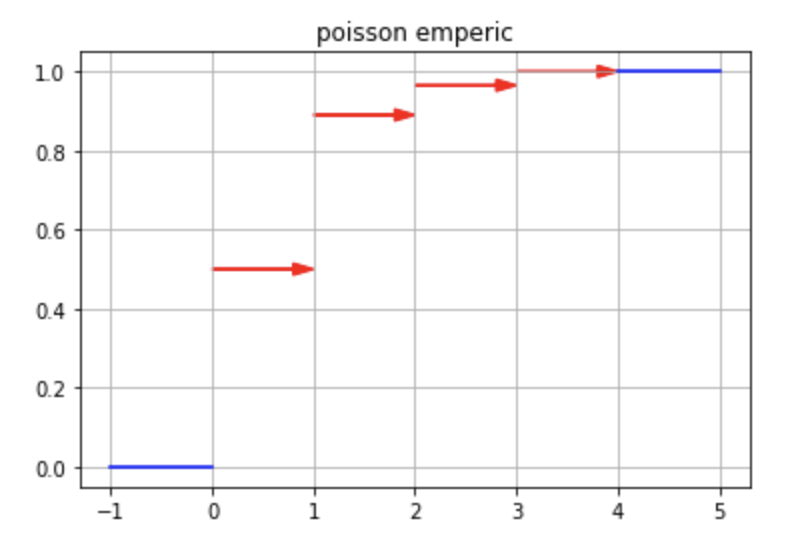
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |

Статистический ряд:

|  |  |  |  |
| --- | --- | --- | --- |
| **Xk** | **Nk** | **Wk** | **Sk** |
| 0 | 100 | 0,5 | 0,5 |
| 1 | 78 | 0,39 | 0,89 |
| 2 | 15 | 0,075 | 0,965 |
| 3 | 7 | 0,035 | 1 |

Эмпирическая функция распределения и ее график:





Выборочное среднее = 0.645

Выборочная дисперсия = 0.5889749999999998

Выборочный момент 3-ого порядка = 1.935

Выборочный момент 4-ого порядка = 4.425000000000001

Выборочное среднее квадратическое отклонение = 0.767447066578536

Выборочная мода = 0

Выборочная медиана = 0.5

Выборочный коэффициент асимметрии = 2.5793944819997057

Выборочный коэффициент эксцесса = 8.817967718961837

**Анализ результатов**

**Задание 1 (биномиальное распределение)**

n = 20, p = 0,375

Таблица сравнения относительных частот и теоретических вероятностей:

|  |  |  |  |
| --- | --- | --- | --- |
| **j** |  |  |  |
| 1 | 0.05 | 0.0009926 | 0.0490074 |
| 2 | 0.01 | 0.0056579 | 0.0043421 |
| 3 | 0.02 | 0.0203684 | 0.0003684 |
| 4 | 0.04 | 0.0519396 | 0.019396 |
| 5 | 0.09 | 0.0997241 | 0.0097241 |
| 6 | 0.16 | 0.1495862 | 0.0104138 |
| 7 | 0.14 | 0.1795034 | 0.0395034 |
| 8 | 0.18 | 0.1750158 | 0.0049842 |
| 9 | 0.13 | 0.1400127 | 0.0100127 |
| 10 | 0.11 | 0.0924083 | 0.0175917 |
| 11 | 0.075 | 0.0504045 | 0.0245955 |
| 12 | 0.02 | 0.0226820 | 0.0026820 |
| 13 | 0.015 | 0.0083749 | 0.0066251 |
| 14 | 0 | 0.0025124 | 0.0025124 |
| 15 | 0.005 | 0.0006029 | 0.0043971 |

Таблица сравнения рассчитанных характеристик с теоретическими значениями:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Экспериментальное значение | Теоретическое  значение | Абсолютное отклонение | Относительное отклонение |
| Выборочное среднее | 7.68 | 7.5 | 0.18 | 2.4% |
| Выборочная дисперсия | 5.4176 | 4.6875 | 0.7301 | 15.575466% |
| Выборочное среднее квадратичное отклонение | 2.3275738 | 2.1650635 | 0.1625103 | 7.5060292% |
| Выборочная мода | 8 | 7 | 1 | 14.285714% |
| Выборочная медиана | 8 | 8 | 0 | 0% |
| Выборочный коэффициент асимметрии | 0.0022241 | 0.11547 | 0.1132459 | 0.980738% |
| Выборочный коэффициент эксцесса | -2.896367 | -0.086666 | 2.809701 | 3241.9876% |

**Задание 2 (геометрическое распределение)**

p = 0,375

Таблица сравнения относительных частот и теоретических вероятностей:

|  |  |  |  |
| --- | --- | --- | --- |
| **j** |  |  |  |
| 0 | 0.425 | 0.375 | 0.05 |
| 1 | 0.235 | 0.234375 | 0.000625 |
| 2 | 0.155 | 0.1464843 | 0.0085157 |
| 3 | 0.065 | 0.0915527 | 0.0265527 |
| 4 | 0.05 | 0.0572204 | 0.0072204 |
| 5 | 0.025 | 0.0357627 | 0.0107627 |
| 6 | 0.015 | 0.0223517 | 0.0013002 |
| 7 | 0.005 | 0.0139698 | 0.0089698 |
| 8 | 0.005 | 0.0087311 | 0.0037311 |
| 9 | 0 | 0.0054569 | 0. 0054569 |
| 10 | 0. | 0.0034106 | 0. 0034106 |
| 11 | 0.005 | 0.0021316 | 0.0028684 |

Таблица сравнения рассчитанных характеристик с теоретическими значениями:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Экспериментальное значение | Теоретическое  значение | Абсолютное отклонение | Относительное отклонение |
| Выборочное среднее | 1.404999 | 1.666667 | 0.261668 | 15.700076% |
| Выборочная дисперсия | 3.560974 | 4.444445 | 0.883471 | 19.878095% |
| Выборочное среднее квадратичное отклонение | 1.887054 | 2.1081851 | 0.2211311 | 10.489169% |
| Выборочная мода | 0 | 0 | 0 | - |
| Выборочная медиана | 1 | 1 | 0 | 0% |
| Выборочный коэффициент асимметрии | 0.301671 | 2.055480 | 1.753809 | 85.3235741% |
| Выб. коэфф. эксцесса | -2.375943 | 6.225 | 8.630943 | 137.9847% |

**Задание 3 (распределение Пуассона)**

=0,65

Таблица сравнения относительных частот и теоретических вероятностей:

|  |  |  |  |
| --- | --- | --- | --- |
| **j** |  |  |  |
| 0 | 0.5 | 0.5220457 | 0.0220457 |
| 1 | 0.39 | 0.3393297 | 0.0506703 |
| 2 | 0.075 | 0.1102821 | 0.0352821 |
| 3 | 0.035 | 0.0238944 | 0.0111056 |

Таблица сравнения рассчитанных характеристик с теоретическими значениями:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название показателя | Экспериментальное значение | Теоретическое  значение | Абсолютное отклонение | Относительное отклонение |
| Выборочное среднее | 0.645 | 0,65 | 0.005 | 8.064516% |
| Выборочная дисперсия | 0.58897 | 0,65 | 0.06103 | 9.389230% |
| Выборочное среднее квадратичное отклонение | 0.76744 | 0.80622 | 0.03878 | 4.810101% |
| Выборочная мода | 0 | 0 | 0 | - |
| Выборочная медиана | 0.5 | 0 | 0.5 | - |
| Выборочный коэффициент асимметрии | 2.57939 | 1.24034 | 1.33905 | 107.958302% |
| Выб. коэфф. эксцесса | 8.81796 | 1.53846 | 7.2795 | 473.167973% |

**Вывод**

В ходе лабораторной работы выяснилось, что полученные экспериментальным путем данные соответствуют заданным распределениям, если принимать в расчет отклонения от теоретического значения.

Экспериментальная оценка выборочных показателей может сильно отличаться от теоретического значения, в силу того, что выборки из 200 элементов недостаточно для проведения точных расчетов. С увеличением выборки точность будет улучшаться.

**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. Кетков Ю.Л., Кетков Ю.Л., Шульц М.М. MATLAB 7: программирование, численные методы. − СПб.: БВХ-Петербург, 2005. − 752 с.

**Приложение**

# вариант №15

import matplotlib.pyplot as plt

from numpy.random import binomial, geometric, poisson

import pandas as pd

from collections import Counter

from math import factorial as fac

from math import exp, sqrt, floor, log

V = 15

n = 5+V%16

p = 0.3+0.005\*V

lam = 0.5+0.01\*V

size = 200

print(n, p, lam, size)

# создание df

def df\_create(data, size):

Wk = {}

Sk = {}

Nk = Counter(data)

sum\_Wk = 0

for i in range(n):

Wk[i] = Nk[i]/size

sum\_Wk += Wk[i]

Sk[i] = sum\_Wk

df = pd.DataFrame(sorted(list(Nk.items())), columns=['Xk', 'Nk'])

d\_Wk = list()

for i in range(len(Wk)):

if Wk[i] != 0:

d\_Wk.append(Wk[i])

df['Wk'] = d\_Wk

d\_Sk = list()

for i in sorted(list(Nk.keys())):

d\_Sk.append(Sk[i])

df['Sk'] = d\_Sk

return df

# график знач. теор. бином. распр.

def comb(n, k):

return (fac(n)/fac(k)/fac(n-k))

def binom\_comp(n, k, p):

return (comb(n,k)\*(p\*\*k)\*((1-p)\*\*(n-k)))

def plot\_binom\_theoretical(df):

arr\_x = list(range(0,(df['Xk'][len(df)-1])+1))

arr\_y = list()

for i in range(df['Xk'][len(df)-1]+1):

arr\_y.append(binom\_comp(n, i, p))

plt.plot(arr\_x, arr\_y, marker='o', color='red', label='theoretical')

# график знач. теор. геом. распр.

def plot\_geomm\_theoretical(df):

arr\_x = list(range(df['Xk'][0], df['Xk'][len(df)-1]+1))

arr\_y = list()

for i in range(df['Xk'][0], df['Xk'][len(df)-1]+1):

arr\_y.append(((1-p)\*\*i)\*p)

plt.plot(arr\_x, arr\_y, marker='o', color='red', label='theoretical')

# график знач. теор. Пуассоновского распр.

def plot\_poisson\_theoretical(df):

arr\_x = list(range(df['Xk'][0], df['Xk'][len(df)-1]+1))

arr\_y = list()

for i in range(df['Xk'][0], df['Xk'][len(df)-1]+1):

arr\_y.append((lam\*\*i/fac(i)\*exp(-lam)))

plt.plot(arr\_x, arr\_y, marker='o', color='red', label='theoretical')

# график эмпир. ф-ии распр.

def plot\_emperic(df):

l = len(df)-1

arr\_x = list(range(df['Xk'][0],df['Xk'][l]+1))

lst = list(df['Xk'])

arr\_y = []

k=0

for i in arr\_x:

if i in lst:

arr\_y.append(float(df.loc[df['Xk'] == i]['Sk']))

else:

arr\_y.append(arr\_y[k-1])

k+=1

for i in range(len(arr\_x)):

plt.arrow(arr\_x[i], arr\_y[i], 0.8, 0, head\_width=0.03, head\_length=0.2, color='red')

plt.plot([df['Xk'][0], df['Xk'][0]-1], [0,0], color='blue')

plt.plot([df['Xk'][l]+1, df['Xk'][l]+2], [1,1], color='blue')

plt.grid(True)

# выборочное среднее

def sample\_mean(df):

size = 0

for i in range (len(df)):

size += df['Nk'][i]

x = 0

for i in range(len(df)):

x += (df['Xk'][i])\*df['Wk'][i]

return x

# выборочниый момент К-го порядка

def sample\_moment\_k(df, k):

size = 0

for i in range (len(df)):

size += df['Nk'][i]

x = 0

for i in range(len(df)):

x += (df['Xk'][i]\*\*k)\*df['Wk'][i]

return x

# выборочная дисперсия

def sample\_dispersion(df):

return (sample\_moment\_k(df, 2)-(sample\_mean(df)\*\*2))

# выборочное среднее квадратическое отклонение

def sample\_quadratic\_deviation(df):

return (sqrt(sample\_dispersion(df)))

# выборочный центральный момент k-ого порядка

def sample\_centr\_moment\_k(df, k):

x = 0

sm = sample\_mean(df)

for i in range(len(df)):

x += ((df['Xk'][i]-sm)\*\*k)\*df['Wk'][i]

return x

# выборочная мода

def sample\_fashion(df):

max\_c = max(df['Nk'])

index = list()

for i in range(len(df)):

if df['Nk'][i] == max\_c:

index.append(i)

check = True

if len(index) > 1:

for i in range(1,len(index)):

if index[i]-index[i-1] != 1:

check = False

else:

return (df['Xk'][index[0]])

if check:

return ((df['Xk'][index[0]]+df['Nk'][index[-1]])/2)

else:

return None

# выборочная медиана

def sample\_median(df):

for i in range(len(df)):

if df['Sk'][i] == 0.5 :

return ((df['Xk'][i]+df['Xk'][i+1])/2)

elif df['Sk'][i] > 0.5 :

return (df['Xk'][i])

# выборочный коэффициент асимметрии

def sample\_asymmetry\_coef(df):

return (sample\_centr\_moment\_k(df, 3) / sample\_dispersion(df)\*\*3)

# выборочный коэффициент эксцесса

def sample\_kurtosis\_coef(df):

return ((sample\_centr\_moment\_k(df, 4) / sample\_dispersion(df)\*\*4) - 3)

def print\_all(df):

print('Выборочное среднее =', sample\_mean(df))

print('Выборочная дисперсия =', sample\_dispersion(df))

print('Выборочный момент 3-ого порядка =', sample\_moment\_k(df, 3))

print('Выборочный момент 4-ого порядка =', sample\_moment\_k(df, 4))

print('Выборочное среднее квадратическое отклонение =', sample\_quadratic\_deviation(df))

print('Выборочная мода =', sample\_fashion(df))

print('Выборочная медиана =', sample\_median(df))

print('Выборочный коэффициент асимметрии =', sample\_asymmetry\_coef(df))

print('Выборочный коэффициент эксцесса =', sample\_kurtosis\_coef(df))

def print\_binom\_teor():

print('BINOM teoreric')

print('expected value =', n\*p)

print('dispersion =', n\*p\*(1-p))

print('quadratic deviation =', sqrt(n\*p\*(1-p)))

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

print('fashion =', (n+1)\*p - 0.5)

else:

print('fashion =', floor((n+1)\*p))

print('median =', round(n\*p))

print('asymmetry =', (1 - 2\*p) / sqrt(n\*p\*(1-p)))

print('kurtosis =', (1 - 6\*(1-p)\*p) / (n\*p\*(1-p)))

def print\_geom\_teor():

print('GEOM teoreric')

print('expected value =', (1-p)/p)

print('dispersion =', (1-p)/p\*\*2)

print('quadratic deviation =', sqrt(1-p)/p)

print('fashion =', 0)

if log(2)%log(1-p) == 0:

print('median =', -log(2)/log(1-p) - 0.5)

else:

print('median =', floor(-log(2)/log(1-p)))

print('asymmetry =', (2-p)/sqrt(1-p))

print('kurtosis =', 6+p\*\*2/(1-p))

def print\_poisson\_teor():

print('POISSON teoreric')

print('expected value =', lam)

print('dispersion =', lam)

print('quadratic deviation =', sqrt(lam))

print('fashion =', floor(lam))

print('median =', floor(lam+1/3-0.02/lam))

print('asymmetry =', 1/sqrt(lam))

print('kurtosis =', 1/lam)

# биномиальное

# data\_binom = binomial(n, p, size)

# [11, 11, 8, 10, 10, 3, 8, 10, 5, 9,

# 6, 8, 5, 7, 6, 3, 8, 7, 8, 12,

# 11, 4, 3, 4, 11, 7, 6, 10, 7, 11,

# 12, 9, 5, 6, 4, 8, 10, 5, 9, 9,

# 6, 10, 10, 4, 6, 7, 11, 7, 5, 6,

# 9, 5, 6, 5, 6, 9, 7, 6, 10, 10,

# 10, 8, 7, 9, 8, 6, 8, 9, 11, 9,

# 8, 4, 8, 9, 9, 9, 9, 8, 8, 6,

# 7, 8, 7, 8, 13, 6, 5, 8, 11, 6,

# 7, 6, 9, 10, 8, 11, 10, 10, 7, 11,

# 11, 8, 5, 12, 6, 6, 5, 11, 8, 7,

# 7, 10, 9, 7, 6, 4, 6, 4, 8, 7,

# 10, 6, 10, 5, 7, 8, 10, 4, 10, 10,

# 15, 8, 8, 6, 6, 7, 6, 6, 9, 6,

# 7, 9, 5, 9, 5, 8, 6, 1, 8, 12,

# 6, 8, 8, 7, 5, 5, 7, 8, 13, 7,

# 7, 8, 9, 6, 5, 9, 8, 9, 7, 8,

# 11, 8, 9, 5, 8, 9, 2, 7, 8, 10,

# 9, 6, 3, 8, 10, 11, 7, 9, 11, 6,

# 6, 9, 10, 5, 8, 6, 7, 2, 13, 7]

# sort\_data\_binom = sorted(data\_binom)

# [1, 2, 2, 3, 3, 3, 3, 4, 4, 4,

# 4, 4, 4, 4, 4, 5, 5, 5, 5, 5,

# 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,

# 5, 5, 5, 6, 6, 6, 6, 6, 6, 6,

# 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,

# 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,

# 6, 6, 6, 6, 6, 7, 7, 7, 7, 7,

# 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,

# 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,

# 7, 7, 7, 8, 8, 8, 8, 8, 8, 8,

# 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,

# 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,

# 8, 8, 8, 8, 8, 8, 8, 8, 8, 9,

# 9, 9, 9, 9, 9, 9, 9, 9, 9, 9,

# 9, 9, 9, 9, 9, 9, 9, 9, 9, 9,

# 9, 9, 9, 9, 9, 10, 10, 10, 10, 10,

# 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,

# 10, 10, 10, 10, 10, 10, 10, 11, 11, 11,

# 11, 11, 11, 11, 11, 11, 11, 11, 11, 11,

# 11, 11, 12, 12, 12, 12, 13, 13, 13, 15]

# df\_binom = df\_create(data\_binom, size)

# df\_binom.to\_excel("output\_binom.xlsx", sheet\_name='Sheet\_name\_1', index=False)

df\_binom = pd.read\_excel('output\_binom.xlsx')

df\_binom

df\_binom.plot(style='o-', x='Xk', y='Wk', c='green', label='calculated')

plot\_binom\_theoretical(df\_binom)

plt.legend()

plt.ylabel('Wk')

plt.grid(True)

plt.title('binominal polygon')

plot\_emperic(df\_binom)

plt.title('binominal emperic')

print\_all(df\_binom)

print()

print\_binom\_teor()

# геометрическое

# data\_geometric = geometric(p, size)

# data\_geometric

# [2, 3, 1, 2, 1, 1, 5, 1, 1, 2,

# 1, 1, 1, 4, 1, 2, 1, 1, 3, 2,

# 1, 1, 9, 1, 4, 1, 1, 3, 1, 1,

# 2, 1, 1, 1, 1, 3, 9, 1, 1, 1,

# 3, 1, 2, 4, 3, 3, 2, 1, 3, 2,

# 1, 6, 2, 1, 1, 1, 1, 1, 7, 2,

# 3, 3, 3, 3, 1, 2, 1, 3, 4, 4,

# 2, 1, 2, 2, 1, 3, 2, 6, 2, 5,

# 5, 3, 2, 3, 2, 5, 2, 2, 1, 1,

# 3, 8, 3, 5, 1, 2, 2, 1, 2, 1,

# 2, 1, 1, 2, 1, 9, 1, 6, 2, 1,

# 1, 2, 2, 1, 1, 2, 3, 1, 1, 1,

# 1, 7, 1, 3, 1, 4, 3, 1, 6, 2,

# 2, 1, 1, 1, 1, 3, 1, 1, 1, 1,

# 12, 2, 5, 3, 1, 1, 1, 2, 1, 1,

# 2, 4, 4, 3, 2, 3, 1, 2, 4, 2,

# 2, 1, 3, 2, 1, 1, 4, 2, 9, 6,

# 3, 1, 1, 1, 2, 5, 1, 3, 1, 2,

# 5, 1, 1, 5, 3, 4, 5, 4, 1, 2,

# 2, 2, 1, 4, 7, 3, 1, 1, 3, 2]

# sort\_data\_geometric = (sorted(data\_geometric))

# print(sort\_data\_geometric)

# [1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 2, 2, 2, 2, 2,

# 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

# 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

# 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

# 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

# 2, 2, 3, 3, 3, 3, 3, 3, 3, 3,

# 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,

# 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,

# 3, 3, 3, 4, 4, 4, 4, 4, 4, 4,

# 4, 4, 4, 4, 4, 4, 5, 5, 5, 5,

# 5, 5, 5, 5, 5, 5, 6, 6, 6, 6,

# 6, 7, 7, 7, 8, 9, 9, 9, 9, 12]

# df\_geometric = df\_create(data\_geometric, size)

# df\_geometric.to\_excel("output\_geometric.xlsx", sheet\_name='Sheet\_name\_1', index=False)

df\_geometric = pd.read\_excel('output\_geometric.xlsx')

df\_geometric

df\_geometric.plot(style='o-', x='Xk', y='Wk', c='green', label='calculated')

plot\_geomm\_theoretical(df\_geometric)

plt.title('geometric polygon')

plt.legend()

plt.ylabel('Wk')

plt.grid(True)

plot\_emperic(df\_geometric)

plt.title('geometric emperic')

print\_all(df\_geometric)

print()

print\_geom\_teor()

# Пуассон

# data\_poisson = poisson(lam, size)

# data\_poisson

# [0, 0, 2, 0, 1, 0, 0, 0, 0, 1,

# 1, 1, 0, 1, 0, 0, 2, 1, 1, 1,

# 1, 0, 0, 1, 3, 0, 0, 0, 0, 0,

# 0, 1, 1, 0, 1, 0, 1, 1, 1, 2,

# 1, 0, 1, 1, 1, 1, 0, 1, 1, 0,

# 1, 1, 1, 0, 0, 0, 0, 1, 2, 0,

# 0, 1, 2, 0, 2, 0, 0, 0, 0, 0,

# 0, 2, 0, 0, 1, 2, 0, 0, 0, 3,

# 0, 0, 1, 1, 1, 1, 1, 1, 1, 0,

# 0, 1, 1, 0, 0, 2, 0, 1, 2, 0,

# 1, 3, 0, 0, 0, 1, 0, 0, 1, 1,

# 0, 0, 0, 3, 1, 0, 0, 0, 1, 0,

# 0, 1, 0, 0, 1, 1, 0, 1, 1, 1,

# 0, 1, 1, 0, 0, 1, 0, 0, 0, 1,

# 1, 0, 1, 1, 1, 2, 0, 0, 1, 0,

# 1, 2, 0, 0, 1, 0, 1, 0, 1, 1,

# 1, 0, 1, 0, 0, 1, 1, 1, 1, 0,

# 3, 1, 0, 0, 3, 0, 0, 1, 0, 0,

# 0, 0, 0, 0, 1, 1, 0, 2, 2, 0,

# 0, 1, 3, 0, 2, 1, 0, 1, 0, 0]

# sort\_data\_poisson = (sorted(data\_poisson))

# print(sort\_data\_poisson)

# [0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,

# 1, 1, 1, 1, 1, 1, 1, 1, 2, 2,

# 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

# 2, 2, 2, 3, 3, 3, 3, 3, 3, 3]

# df\_poisson = df\_create(data\_poisson, size)

# df\_poisson.to\_excel("output\_poisson.xlsx", sheet\_name='Sheet\_name\_1', index=False)

df\_poisson = pd.read\_excel('output\_poisson.xlsx')

df\_poisson

df\_poisson.plot(style='o-', x='Xk', y='Wk', c='green', label='calculated')

plot\_poisson\_theoretical(df\_poisson)

plt.legend()

plt.ylabel('Wk')

plt.grid(True)

plt.title(‘poisson polygon')

plot\_emperic(df\_poisson)

plt.title('poisson emperic')

print\_all(df\_poisson)

print()

print\_poisson\_teor()

# теор. значения распр.

def comb(n, k):

return (fac(n)/fac(k)/fac(n-k))

def binom\_comp(n, k, p):

return (comb(n,k)\*(p\*\*k)\*((1-p)\*\*(n-k)))

arr\_binom\_teor = []

arr\_geom\_teor = []

arr\_poisson\_teor = []

for i in range(df\_binom['Xk'][0], df\_binom['Xk'][len(df\_binom)-1]+1):

arr\_binom\_teor.append(binom\_comp(n, i, p))

print('binom')

for i in range(len(arr\_binom\_teor)):

print(i+1, arr\_binom\_teor[i])

for i in range(df\_geometric['Xk'][0], df\_geometric['Xk'][len(df\_geometric)-1]+1):

arr\_geom\_teor.append(((1-p)\*\*i)\*p)

print('geom')

for i in range(len(arr\_geom\_teor)):

print(i+1, arr\_geom\_teor[i])

for i in range(df\_poisson['Xk'][0], df\_poisson['Xk'][len(df\_poisson)-1]+1):

arr\_poisson\_teor.append((lam\*\*i/fac(i)\*exp(-lam)))

print('poisson')

for i in range(len(arr\_poisson\_teor)):

print(i, arr\_poisson\_teor[i])