|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

|  |  |
| --- | --- |
| **ФАКУЛЬТЕТ** | **ИУК «Информатика и управление»** |
| **КАФЕДРА** | **ИУК4 «Программное обеспечение ЭВМ,** |
| **информационные технологии»** | |

**Практическое занятие №3**

**«Точечное оценивание»**

**ДИСЦИПЛИНА: «Методы обработки информации»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил: студент гр. ИУК4-72Б | |  |  | ( | Сафронов Н.С. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |
| Проверил: | |  |  | ( | Никитенко У.В. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |

|  |  |
| --- | --- |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | |
|  | - Балльная оценка:  - Оценка: |

Калуга, 2023

**Постановка задачи**

Сгенерировать выборку из 100 элементов, имеющих указанное в вашем варианте распределение. Считая один из параметров распределения неизвестным, найти его точечную оценку:

а) методом моментов (c помощью указанных в задании моментов);

б) методом максимального правдоподобия.

Построить график функции правдоподобия и убедиться, что найденная с помощью метода максимального правдоподобия оценка действительно является точкой максимума функции правдоподобия. Сравнить полученные точечные оценки с истинным значением параметра распределения.

**Вариант 14**

X - выборка из распределения , где k = 3. Найти оценку параметра k, считая его неизвестным. Метод моментов реализовать с помощью моментов 1-го и 2-го порядков.

**Ход выполнения практического задания**

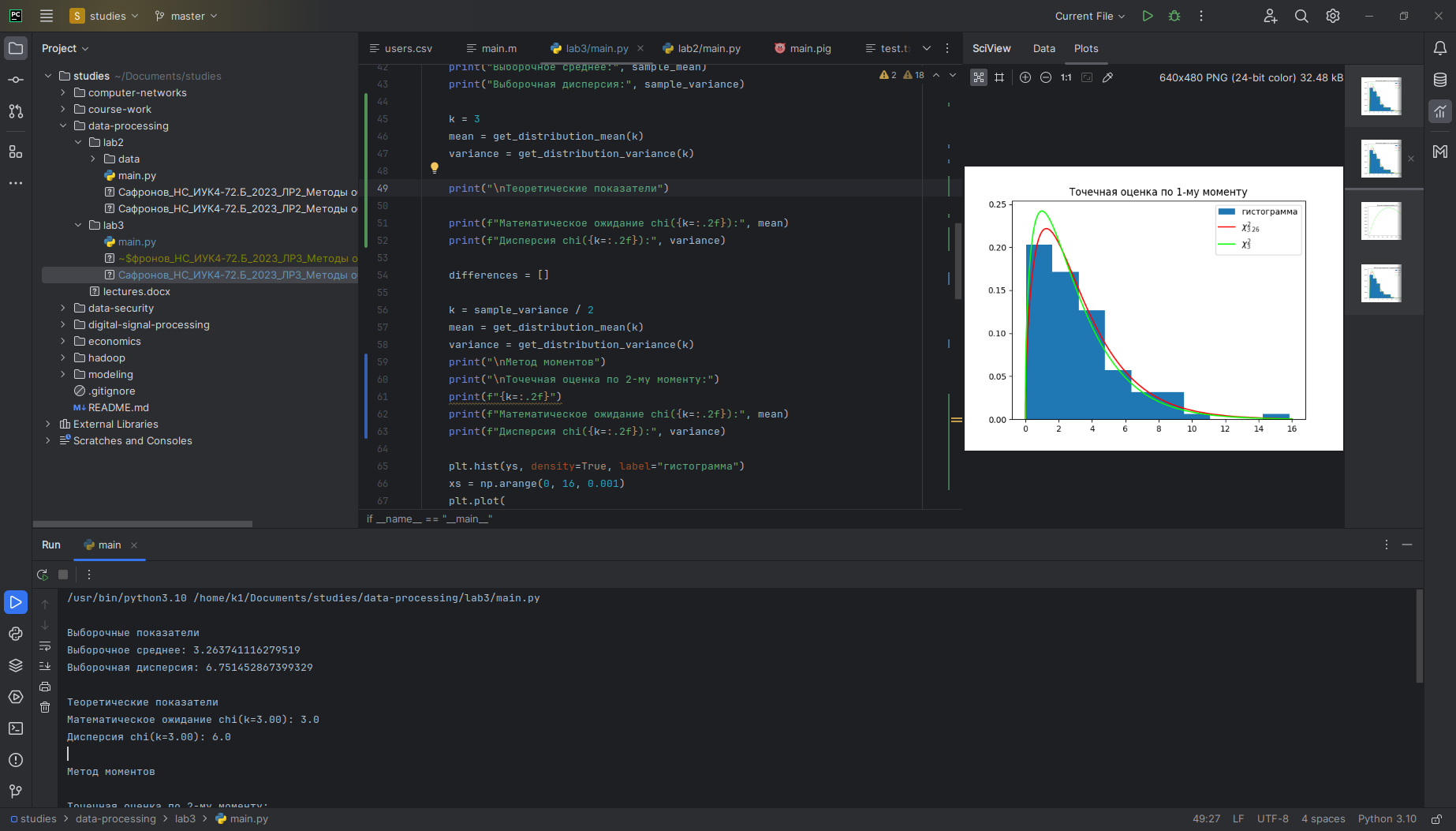
Выпишем формулы для нахождения математического ожидания и дисперсии для распределения :

Получаем следующие точечные оценки для :

Для момента 1-го порядка:

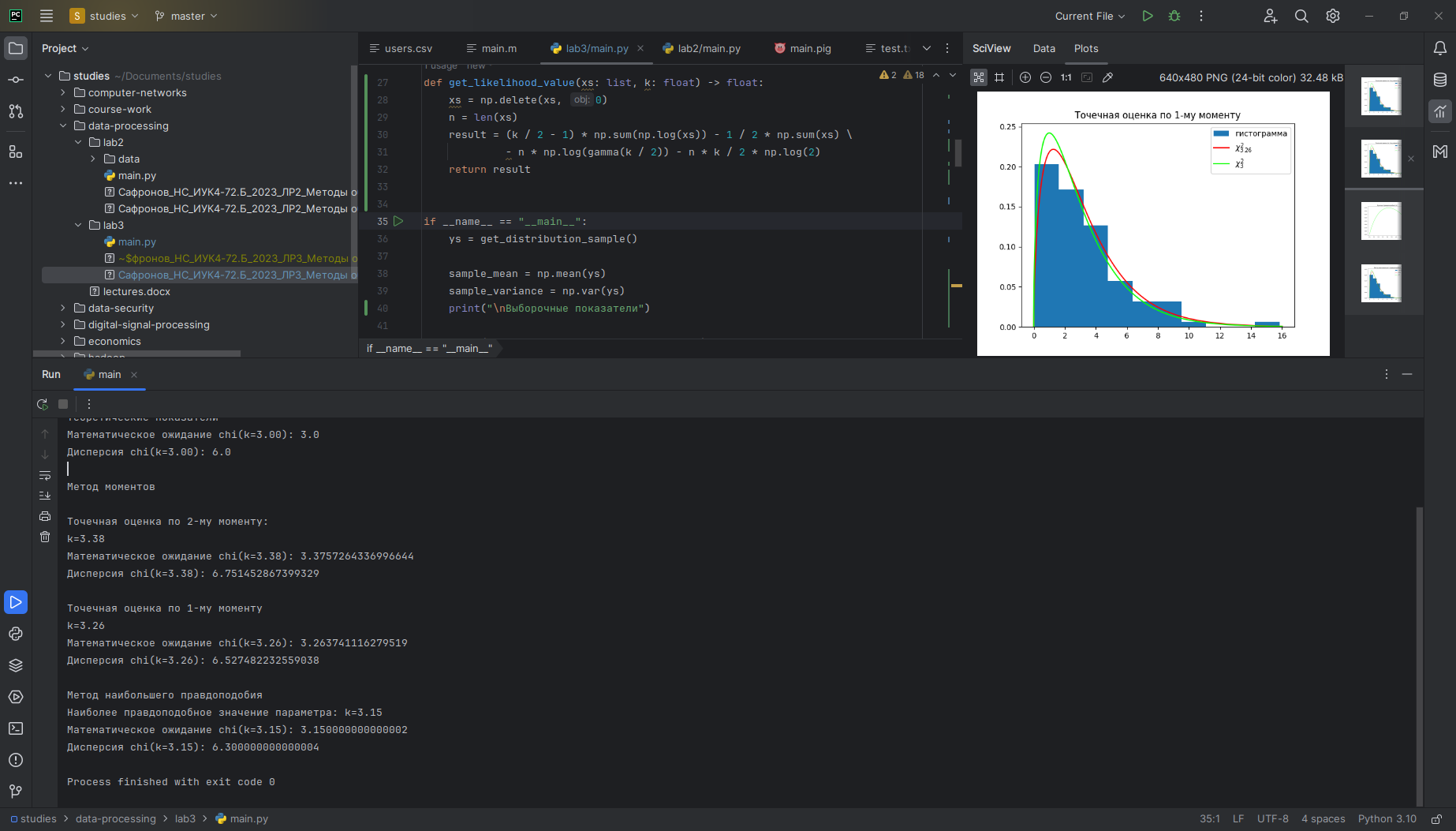
Для момента 2-го порядка:

Найдём выборочные характеристики распределения:



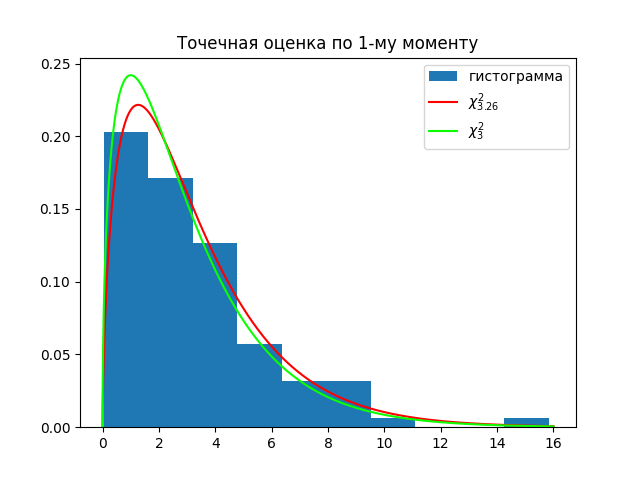
**Рисунок 1 –** Выборочные и теоретические показатели распределения

Воспользовавшись методом моментов, найдём точечную оценку параметра :

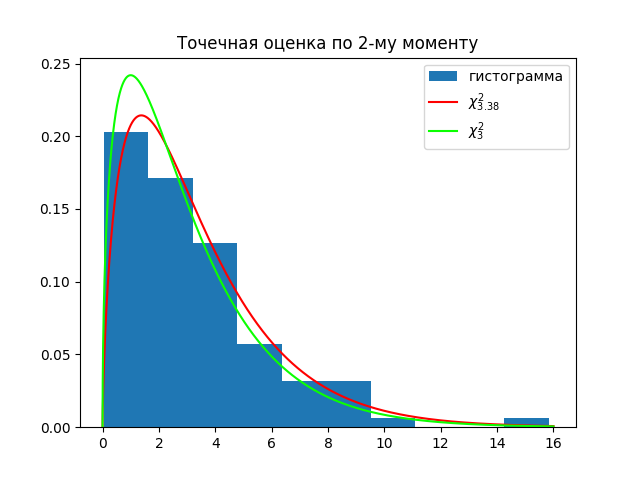


**Рисунок 2 –** Точечные оценки параметра, полученные методом моментов

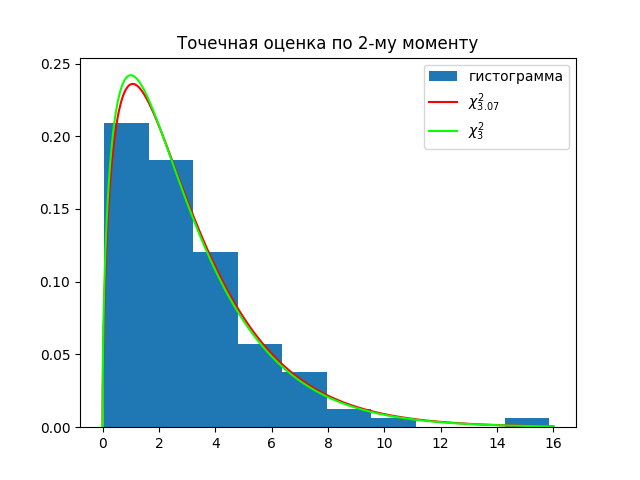
Построим графики, соответствующие полученным значениям параметра:



**Рисунок 3 –** График функции при точечной оценке, полученной по первому моменту

****

**Рисунок 4** – График функции при точечной оценке, полученной по второму моменту

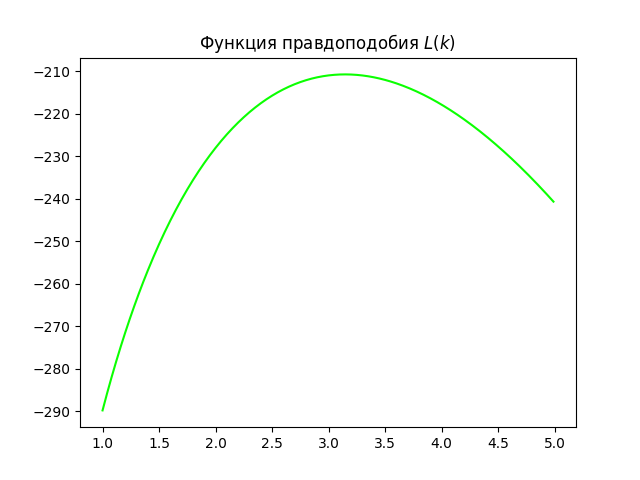
****

**Рисунок 5** – Точечная оценка параметра , вычисленная методом моментов 2-го порядка

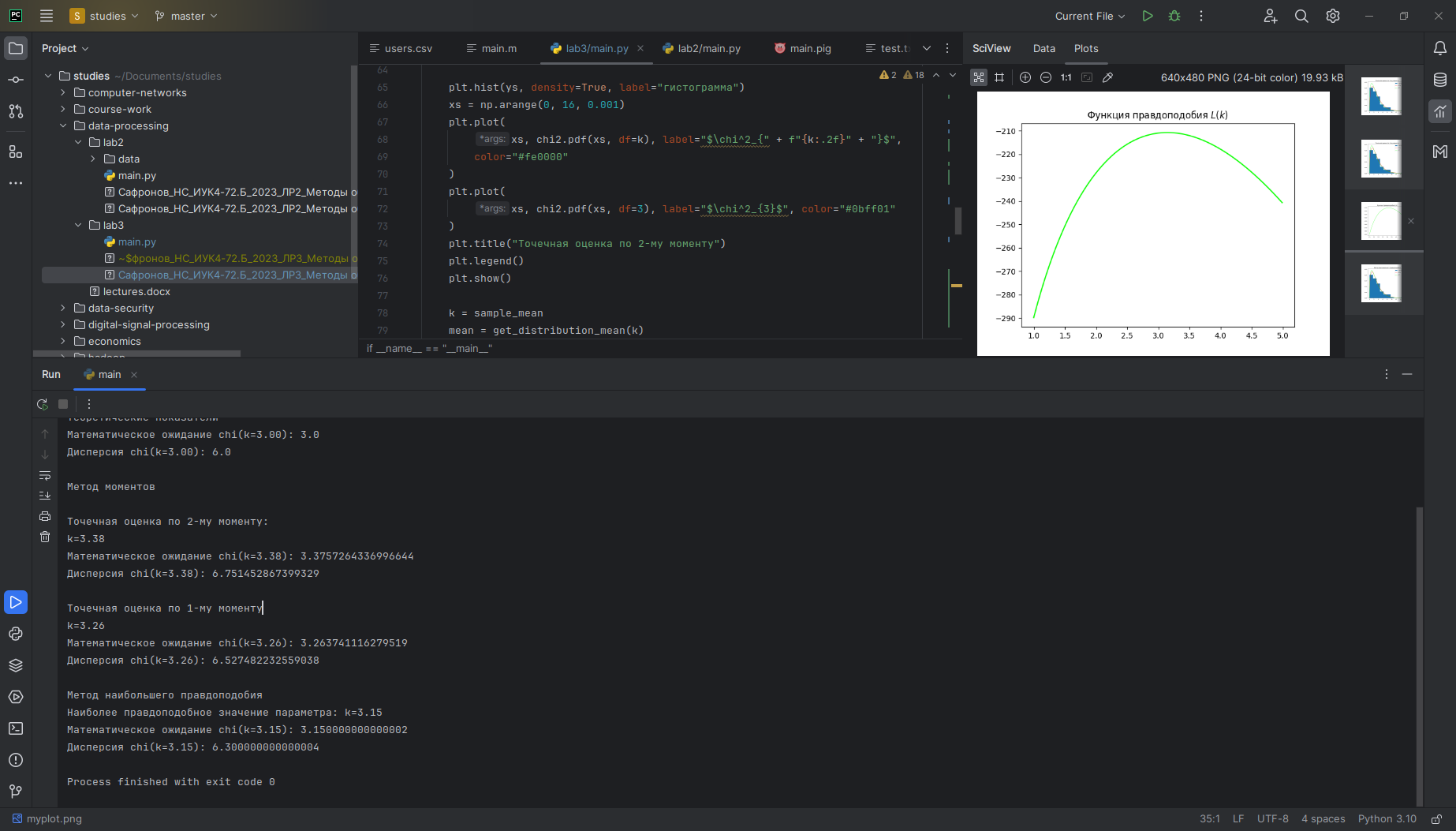
Воспользуемся методом максимального правдоподобия.

Построим логарифмическую функцию правдоподобия для заданного распределения:

Построим график зависимости логарифмической функции правдоподобия на заданном промежутке значений при заданных значениях выборки. Найдём максимальное значение функции и точку, соответствующую ему.

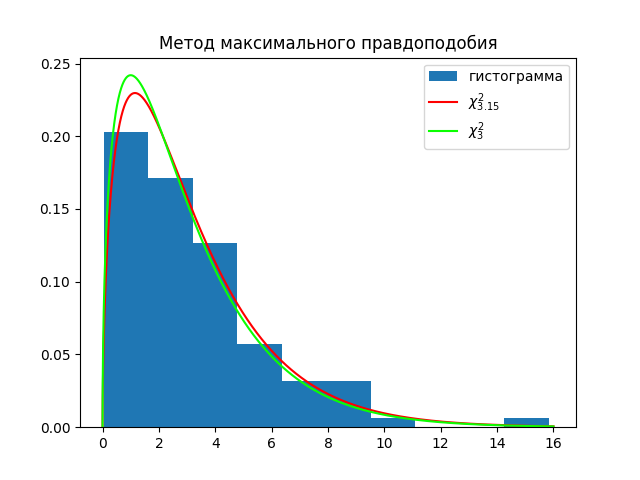
****

**Рисунок 6 –** Функция правдоподобия



**Рисунок 7 –** Точечная оценка параметра, полученная методом максимального правдоподобия

Построим график, соответствующий полученному значению параметра:



**Рисунок 8 –** График функции при точечной оценке, полученной методом максимального правдоподобия

Таким образом, получаем, что наиболее точной оказалась оценка, полученная методом моментов по второму моменту.

**ПРИЛОЖЕНИЯ**

**Листинг программы**

import matplotlib.pyplot as plt  
import numpy as np  
from scipy.special import gamma  
from scipy.stats import chi2  
  
  
def get\_distribution\_sample() -> list:  
 k = 3  
 n = 100  
 xs = chi2(k)  
 ys = xs.rvs(n)  
 return ys  
  
  
def get\_distribution\_mean(k: float) -> float:  
 return chi2(k).mean()  
  
  
def get\_distribution\_variance(k: float) -> float:  
 return chi2(k).var()  
  
  
def get\_likelihood\_value(xs: list, k: float) -> float:  
 xs = np.delete(xs, 0)  
 n = len(xs)  
 result = (k / 2 - 1) \* np.sum(np.log(xs)) - 1 / 2 \* np.sum(xs) \  
 - n \* np.log(gamma(k / 2)) - n \* k / 2 \* np.log(2)  
 return result  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 ys = get\_distribution\_sample()  
  
 sample\_mean = np.mean(ys)  
 sample\_variance = np.var(ys)  
 print("\nВыборочные показатели")  
  
 print("Выборочное среднее:", sample\_mean)  
 print("Выборочная дисперсия:", sample\_variance)  
  
 k = 3  
 mean = get\_distribution\_mean(k)  
 variance = get\_distribution\_variance(k)  
  
 print("\nТеоретические показатели")  
  
 print(f"Математическое ожидание chi({k=:.2f}):", mean)  
 print(f"Дисперсия chi({k=:.2f}):", variance)  
  
 differences = []  
  
 k = sample\_variance / 2  
 mean = get\_distribution\_mean(k)  
 variance = get\_distribution\_variance(k)  
 print("\nМетод моментов")  
 print("\nТочечная оценка по 2-му моменту:")  
 print(f"{k=:.2f}")  
 print(f"Математическое ожидание chi({k=:.2f}):", mean)  
 print(f"Дисперсия chi({k=:.2f}):", variance)  
  
 plt.hist(ys, density=True, label="гистограмма")  
 xs = np.arange(0, 16, 0.001)  
 plt.plot(  
 xs, chi2.pdf(xs, df=k), label="$\chi^2\_{" + f"{k:.2f}" + "}$",  
 color="#fe0000"  
 )  
 plt.plot(  
 xs, chi2.pdf(xs, df=3), label="$\chi^2\_{3}$", color="#0bff01"  
 )  
 plt.title("Точечная оценка по 2-му моменту")  
 plt.legend()  
 plt.show()  
  
 k = sample\_mean  
 mean = get\_distribution\_mean(k)  
 variance = get\_distribution\_variance(k)  
 print("\nТочечная оценка по 1-му моменту")  
 print(f"{k=:.2f}")  
 print(f"Математическое ожидание chi({k=:.2f}):", mean)  
 print(f"Дисперсия chi({k=:.2f}):", variance)  
  
 plt.hist(ys, density=True, label="гистограмма")  
 xs = np.arange(0, 16, 0.001)  
 plt.plot(  
 xs, chi2.pdf(xs, df=k), label="$\chi^2\_{" + f"{k:.2f}" + "}$",  
 color="#fe0000"  
 )  
 plt.plot(  
 xs, chi2.pdf(xs, df=3), label="$\chi^2\_{3}$", color="#0bff01"  
 )  
 plt.title("Точечная оценка по 1-му моменту")  
 plt.legend()  
 plt.show()  
  
 ks = np.arange(1, 5, 0.01)  
 likelihood\_values = np.array([get\_likelihood\_value(ys, k) for k in ks])  
 max\_likelihood\_indices = np.argmax(likelihood\_values)  
 k = ks[max\_likelihood\_indices]  
  
 print("\nМетод наибольшего правдоподобия")  
 print(f"Наиболее правдоподобное значение параметра: {k=:.2f}")  
  
 mean = get\_distribution\_mean(k)  
 variance = get\_distribution\_variance(k)  
 print(f"Математическое ожидание chi({k=:.2f}):", mean)  
 print(f"Дисперсия chi({k=:.2f}):", variance)  
  
 plt.plot(  
 ks, likelihood\_values, label="Функция правдоподобия $L(k)$",  
 color="#0bff01"  
 )  
 plt.title("Функция правдоподобия $L(k)$")  
 plt.show()  
  
 plt.hist(ys, density=True, label="гистограмма")  
 xs = np.arange(0, 16, 0.001)  
 plt.plot(  
 xs, chi2.pdf(xs, df=k), label="$\chi^2\_{" + f"{k:.2f}" + "}$",  
 color="#fe0000"  
 )  
 plt.plot(  
 xs, chi2.pdf(xs, df=3), label="$\chi^2\_{3}$", color="#0bff01"  
 )  
 plt.title("Метод максимального правдоподобия")  
 plt.legend()  
 plt.show()