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»***

ДОМАШНЯЯ РАБОТА №1

«Первичная обработка данных»

ДИСЦИПЛИНА: «Методы обработки информации»

Выполнил: студент гр. ИУК4-72Б ________________ (Карельский М.К.)
(Подпись)

Проверил: ________________ (Никитенко У.В.)
(Подпись)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:
- Оценка:

Калуга, 2023

Цель: формирование у студентов практических навыков обработки статистических данных.

Задачи: моделирование непрерывной СВ, анализ исходных данных, построение оценок плотности вероятности, нахождение точечных и интервальных оценок параметров распределения.

Задание 1:

1. Выполнить статистическое моделирование случайной величины с заданным законом распределения путем генерации отсчетов α1i, i = 1, …, N случайных величин с равномерным распределением в интервале [0, 1] (или, при необходимости нескольких СВ (α1, α2, ..., αk); N=10000. Сформировать соответствующий script-файл в среде MATLAB.
2. Получить гистограмму для закона распределения в соответствии с вариантом задания. Гистограмма может быть получена в среде MATLAB с помощью оператора hist(X1,N), X1 — анализируемая случайная величина, N — число интервалов на гистограмме, которое должно составлять от 100 до 500. Сравнить полученную гистограмму с соответствующим графиком плотности вероятности f(x) в соответствии с заданием.
3. Вычислить:
    - выборочное среднее значение,
    - медиану,
    - нижний и верхний квартиль,
    - выборочную дисперсию и СКО,

    смоделированной случайной величины и сравнить их с теоретическими значениями (мат. ожиданием и дисперсией, медианой, нижним и верхним квартилем).
4. Сделать выводы.

Вариант 11

- Закон распределения: Хи-квадрат
- Алгоритм: E2
- $v = 3$
- $\sigma = 2$

Листинг:

```
import math, random, statistics
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import chi2
v = 3
sig = 2

G = math.pi / 2
```

```
f = lambda x: x**((v - 2) / 2) / (2**(v / 2) * G * sig**v) * math.exp(-x / (2 *
sig**2))

N = 10000
s = [0] * N
for i in range(N):
    for j in range(v):
        x = random.normalvariate(0, sig)
        s[i] += x**2

print("--------------Модель--------------")
print(f"Выборочное среднее значение: {statistics.fmean(s):.3f}")
print(f"Медиана: {statistics.median(s):.3f}")
print(f"Верхний квартиль: {np.percentile(s, 75):.3f}")
print(f"Нижний квартиль: {np.percentile(s, 25):.3f}")
print(f"Выборочная дисперсия: {statistics.variance(s):.3f}")
print(f"СКО: {statistics.stdev(s):.3f}")

print("--------------Теория--------------")
print(f"Выборочное среднее значение: {chi2.mean(v):.3f}")
print(f"Медиана: {chi2.median(v):.3f}")
print(f"Верхний квартиль: {chi2.cdf(0.75, v):.3f}")
print(f"Нижний квартиль: {chi2.cdf(0.25, v):.3f}")
print(f"Выборочная дисперсия: {chi2.var(v):.3f}")
print(f"СКО: {chi2.std(v):.3f}")

X = np.linspace(0, 100, 101)
Y = [f(x) * 10000 for x in X]

plt.plot(X, Y)
plt.hist(s, bins=200)
plt.show()
```

Результат:

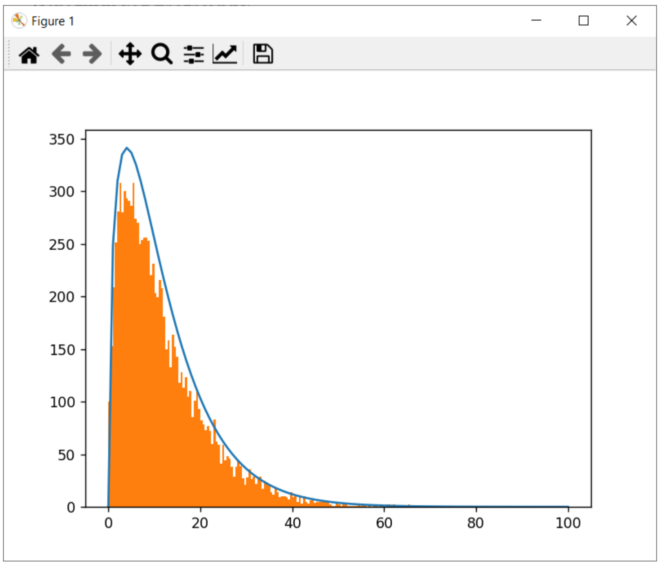

Рис. 1. Графики

```
-------------Модель-------------
Выборочное среднее значение: 3.041
Медиана: 2.430
Верхний квартиль: 0.103
Нижний квартиль: 0.031
Выборочная дисперсия: 5.947
СКО: 2.439
-------------Теория-------------
Выборочное среднее значение: 3.000
Медиана: 2.366
Верхний квартиль: 0.139
Нижний квартиль: 0.031
Выборочная дисперсия: 6.000
СКО: 2.449
```

Рис. 2. Вычисления

Задание 2:

Для обработки преподавателем выдается случайных чисел. Эти числа хранятся в файле TestNN.csv.

1. Выборка подвергается обработке и оформляется в виде таблицы.

№ промежутка	Границы промежутков		n_i	Средняя точка промежутка
	a_{i-1}	a_i		

2. Графические характеристики выборки – строим гистограмму и полигон приведенных частот. Выдвигаем гипотезу о виде плотности вероятности генерального распределения.
3. Находим выборочные характеристики положения и рассеивания.
4. Для сравнения с гистограммой и полигоном приведенных частот на одном чертеже постройте графики гистограммной оценки плотности вероятности $\hat{f}_{\Gamma}$, параметрической оценки плотности вероятности $\hat{f}_{\Pi}$, и усредненную ядерную оценку плотности вероятности $\hat{f}_{\text{УЯ}}$.
5. Значения оценок плотности вероятности в средних точках промежутков группированного статистического ряда оформите в виде таблицы.

z_i		...	Σ
n_i			
$\hat{f}_{\text{г}}(x)$			—
$\hat{f}_{\text{уя}}(x)$			—
$\hat{f}_{\text{п}}(x)$			—
$(\hat{f}_{\text{уя}} - \hat{f}_{\text{г}})^2$			
$(\hat{f}_{\text{п}} - \hat{f}_{\text{г}})^2$			

6. Проанализируйте близость оценок по средним квадратическим отклонениям $\hat{f}_{\text{УЯ}}$ и $\hat{f}_{\Pi}$ от $\hat{f}_{\Gamma}$.

Вариант 7

Листинг:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm, gaussian_kde
import pandas as pd
from prettytable import PrettyTable

data = pd.read_csv('Test7.csv', header=None)
values = data.iloc[:, 0]
num_bins = 8

plt.figure(figsize=(10, 5))
plt.subplot(1, 2, 1)
plt.hist(values, bins=num_bins, density=True, alpha=0.6, label='Гистограмма')

hist, bin_edges = np.histogram(values, bins=num_bins, density=True,
range=(values.min(), values.max()))
bin_centers = (bin_edges[:-1] + bin_edges[1:]) / 2
plt.plot(bin_centers, hist, marker='o', linestyle='-', label='Полигон')
plt.title("Гистограмма и полигон частот")
plt.legend()

edges = [(round(bin_edges[i], 2), round(bin_edges[i + 1], 2)) for i in
range(num_bins)]

table = PrettyTable()
table.add_column("#", [i for i in range(1, num_bins + 1)])
table.add_column("Промежуток", edges)
table.add_column("Максимальное значение в этой точке", list(map(lambda value:
round(value, 4), hist)))
table.add_column("Центр промежутка", list(map(lambda center: round(center, 2),
bin_centers)))
print(table)

plt.subplot(1, 2, 2)
x = np.linspace(values.min(), values.max(), 100)
plt.plot(x, norm.pdf(x, loc=values.mean(), scale=values.std()),
label='Параметрическая оценка')
kde = gaussian_kde(values)
plt.plot(x, kde(x), label='Усредненная ядерная оценка')

hist_estimate = hist / (bin_centers[1] - bin_centers[0])
plt.plot(bin_centers, hist_estimate, label='Гистограммная оценка')
plt.legend()
plt.title('Графики оценок плотности вероятности')

mean_value = np.mean(values)
median_value = np.median(values)
variance = np.var(values)
```

```
std_deviation = np.std(values)
print(f"Выборочное среднее: {mean_value:.4f}")
print(f"Выборочная медиана: {median_value:.4f}")
print(f"Выборочная дисперсия: {variance:.4f}")
print(f"Выборочное стандартное отклонение: {std_deviation}")

parametric_estimate = norm.pdf(bin_centers, loc=values.mean(),
scale=values.std())
kde_estimate = kde(bin_centers)
table = PrettyTable()
table.add_column("№", [i for i in range(1, num_bins + 1)])
table.add_column("Центр промежутков", bin_centers)
table.add_column("Гистограммная оценка плотности вероятности", hist_estimate)
table.add_column("Усредненную ядерную оценку плотности вероятности",
kde_estimate)
table.add_column("Параметрическая оценка плотности вероятности",
parametric_estimate)
print("Значения оценок в средних точках:")
print(table)

table = PrettyTable()
table.add_column("№", [i for i in range(1, num_bins + 1)])
table.add_column("Центр промежутков", bin_centers)
mse_parametric = (parametric_estimate - hist_estimate) ** 2
mse_kde = (kde_estimate - hist_estimate) ** 2
table.add_column("(Усредненная яд. оц. - Гистограммная оц.)^2", mse_parametric)
table.add_column("(Параметрическая оц. - Гистограммная оц.)^2", mse_kde)
table.add_row(['', '', 'sum', 'sum'])
table.add_row(['', '', sum(mse_parametric), sum(mse_kde)])
print(table)
print(f"Среднее квадратичное отклонение параметрической оценки от гистограммной
оценки: {np.sqrt(np.mean(mse_parametric)):.4f}")
print(f"Среднее квадратичное отклонение усредненной ядерной оценки от
гистограммной оценки: {np.sqrt(np.mean(mse_kde)):.4f}")
plt.tight_layout()
plt.show()
```

Результат:

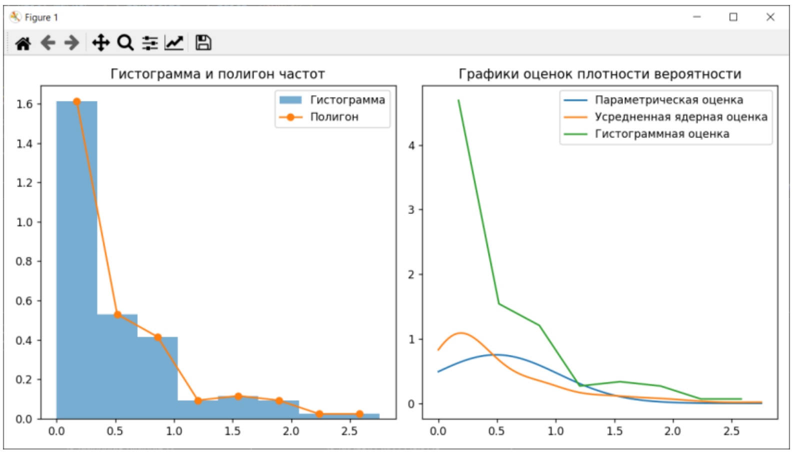

Рис. 1. Графики

Исходя из графиков, можно сделать предположение о геометрическом распределении подборки.

#	Промежуток	Максимальное значение в этой точке	Центр промежутка
1	(0.0, 0.34)	1.6141	0.17
2	(0.34, 0.69)	0.5304	0.52
3	(0.69, 1.03)	0.4151	0.86
4	(1.03, 1.38)	0.0922	1.2
5	(1.38, 1.72)	0.1153	1.55
6	(1.72, 2.07)	0.0922	1.89
7	(2.07, 2.41)	0.0231	2.24
8	(2.41, 2.75)	0.0231	2.58

Выборочное среднее: 0.4909
Выборочная медиана: 0.2969
Выборочная дисперсия: 0.2800
Выборочное стандартное отклонение: 0.5291965435042327
Значения оценок в средних точках:

№	Центр промежутков	Гистограммная оценка плотности вероятности	Усредненную ядерную оценку плотности вероятности	Параметрическая оценка плотности вероятности
1	0.1722783078423276	4.689716443875488	1.0855504750418523	0.6272612967633039
2	0.5164621316149554	1.5409068315590895	0.6688653657173731	0.7500010927376406
3	0.860645955387583	1.2059270855679824	0.34998580554413594	0.5894187904464999
4	1.2048297791602107	0.26798379679288503	0.1664862766001496	0.3044631481913053
5	1.5490136029328387	0.33497974599110625	0.11239666803423407	0.10336991679865919
6	1.8931974267054663	0.26798379679288536	0.07547614751252592	0.023067592767601265
7	2.237381250478094	0.06699594919822122	0.03257938695785676	0.0033834445319016098
8	2.5815650742507215	0.0669959491982213	0.015477309539839066	0.00032618540318292395

№	Центр промежутков	(Усредненная яд. оц. - Гистограммная оц.)^2	(Параметрическая оц. - Гистограммная оц.)^2
1	0.1722783078423276	16.50354182229828	12.9900123308985
2	0.5164621316149554	0.6255318877007019	0.7604563181473694
3	0.860645955387583	0.380082477953597	0.7326354748488608
4	1.2048297791602107	0.0013307430784494283	0.010301746605274734
5	1.5490136029328387	0.053643112978554494	0.04954322659275504
6	1.8931974267054663	0.05998394699415459	0.03705919503144988
7	2.237381250478094	0.004046550749922534	0.0011844997564448804
8	2.5815650742507215	0.00444485740448621	0.002654170232250235
		sum 17.632605399158148	sum 14.583846962112906

Среднее квадратичное отклонение параметрической оценки от гистограммной оценки: 1.4846
Среднее квадратичное отклонение усредненной ядерной оценки от гистограммной оценки: 1.3502

Рис. 2. Вычисления

Задание 3:

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

а) методом моментов (с 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 7

X – выборка из геометрического распределения G_p с параметром p = 0.6. Найти оценку параметра p, считая его неизвестным. Метод моментов реализовать с помощью момента 1-го порядка.

Листинг:

```
import numpy as np
import matplotlib.pyplot as plt

sample_size = 100
```

```
p = 0.6
sample = np.random.geometric(p, size=sample_size)

p_estimate_1 = 1 / np.mean(sample)
print("Оценка параметра p:", p_estimate_1)

def likelihood(p, sample):
    likelihood = np.prod(p * (1-p)**(sample-1))
    return likelihood

grid = np.linspace(0.01, 1, 100)
likelihood_values = [likelihood(p_val, sample) for p_val in grid]
p_estimate_2 = grid[np.argmax(likelihood_values)]
print("Оценка параметра p:", p_estimate_2)

plt.plot(grid, likelihood_values)
plt.axvline(p_estimate_1, color='r', linestyle='--', label='Метод моментов')
plt.axvline(p_estimate_2, color='g', linestyle=':', label='Метод максимального
правдоподобия')
plt.legend()
plt.show()
```

Результат:

```
Оценка параметра p: 0.6097560975609756
Оценка параметра p: 0.61
```

Рис. 1.1. Результат

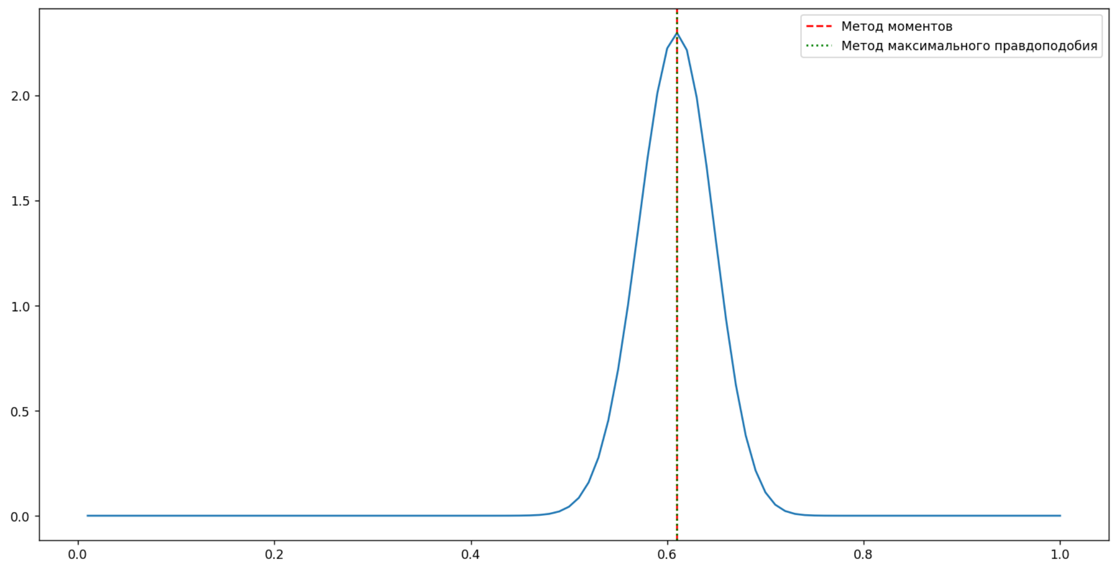

Рис. 1.2. Результат

Задание 4:

Даны две выборки одной случайной величины с нормальным распределением N_{a,σ^2} объема n_1 (малый объем, [8; 12]) и n_2 (в 70 раз больше n_1) соответственно.

Вариант 7

1. Для обеих выборок построить точный доверительный интервал уровня доверия q_0 для параметра a, считая:
   a) σ неизвестным,
   b) σ известным и равным σ_0.
2. В одной системе координат построить графики зависимости длины доверительного интервала от уровня доверия q для всех четырех случаев (объем выборки равен n_1, σ неизвестно; объем выборки равен n_1, σ известно; объем выборки равен n_2, σ неизвестно; объем выборки равен n_2, σ известно). При этом q придать минимум 50 разных значений через равные промежутки.

   Проанализировать взаимное расположение полученных графиков и объяснить его.

- $\sigma_0 = 0.5$
- $q_0 = 0.8$

Листинг:
LW4_1.py

```
import numpy as np
from scipy.stats import t, norm

n1 = 10
n2 = 700
q0 = 0.8
a = 1
sigma0 = 0.5

sample1 = np.random.normal(a, sigma0**2, n1)
sample2 = np.random.normal(a, sigma0**2, n2)

CI1 = t.interval(q0, n1-1, loc=np.mean(sample1), scale=np.std(sample1,
ddof=1)/np.sqrt(n1))
CI2 = t.interval(q0, n2-1, loc=np.mean(sample2), scale=np.std(sample2,
ddof=1)/np.sqrt(n2))

print("Доверительный интервал для выборки 1 (σ неизвестно):", CI1)
print("Доверительный интервал для выборки 2 (σ неизвестно):", CI2)

CI1_known_sigma = norm.interval(q0, loc=np.mean(sample1),
scale=sigma0/np.sqrt(n1))
CI2_known_sigma = norm.interval(q0, loc=np.mean(sample2),
scale=sigma0/np.sqrt(n2))

print("Доверительный интервал для выборки 1 (σ известно):", CI1_known_sigma)
print("Доверительный интервал для выборки 2 (σ известно):", CI2_known_sigma)
```

LW4_2.py

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import t, norm

n1 = 10
n2 = 700
sigma0 = 0.5

q_values = np.linspace(0.01, 0.99, 50)

length_CI1_unknown_sigma = []
length_CI2_unknown_sigma = []
length_CI1_known_sigma = []
length_CI2_known_sigma = []

for q in q_values:
    CI1_unknown_sigma = t.interval(q, n1-1, loc=0, scale=1)
    CI2_unknown_sigma = t.interval(q, n2-1, loc=0, scale=1)

    CI1_known_sigma = norm.interval(q, loc=0, scale=sigma0 / np.sqrt(n1))
    CI2_known_sigma = norm.interval(q, loc=0, scale=sigma0 / np.sqrt(n2))

    length1_unknown_sigma = CI1_unknown_sigma[1] - CI1_unknown_sigma[0]
    length2_unknown_sigma = CI2_unknown_sigma[1] - CI2_unknown_sigma[0]
    length1_known_sigma = CI1_known_sigma[1] - CI1_known_sigma[0]
    length2_known_sigma = CI2_known_sigma[1] - CI2_known_sigma[0]

    length_CI1_unknown_sigma.append(length1_unknown_sigma)
    length_CI2_unknown_sigma.append(length2_unknown_sigma)
    length_CI1_known_sigma.append(length1_known_sigma)
    length_CI2_known_sigma.append(length2_known_sigma)

plt.plot(q_values, length_CI1_unknown_sigma, label="n1, σ unknown")
plt.plot(q_values, length_CI2_unknown_sigma, label="n2, σ unknown")
plt.plot(q_values, length_CI1_known_sigma, label="n1, σ known")
plt.plot(q_values, length_CI2_known_sigma, label="n2, σ known")

plt.xlabel('Уровень доверия, q')
plt.ylabel('Длина доверительного интервала')
plt.title('График зависимости длины доверительного\n интервала от уровня
доверия')
plt.legend()
plt.show()
```

Результат:

```
Доверительный интервал для выборки 1 (σ неизвестно): (1.0145274964657385, 1.1840208787501476)
Доверительный интервал для выборки 2 (σ неизвестно): (0.9942305075394486, 1.0187014054084993)
Доверительный интервал для выборки 1 (σ известно): (0.8966430933041679, 1.301905281911718)
Доверительный интервал для выборки 2 (σ известно): (0.9822469083687132, 1.0306850045792346)
```

Рис. 1.1. Результат

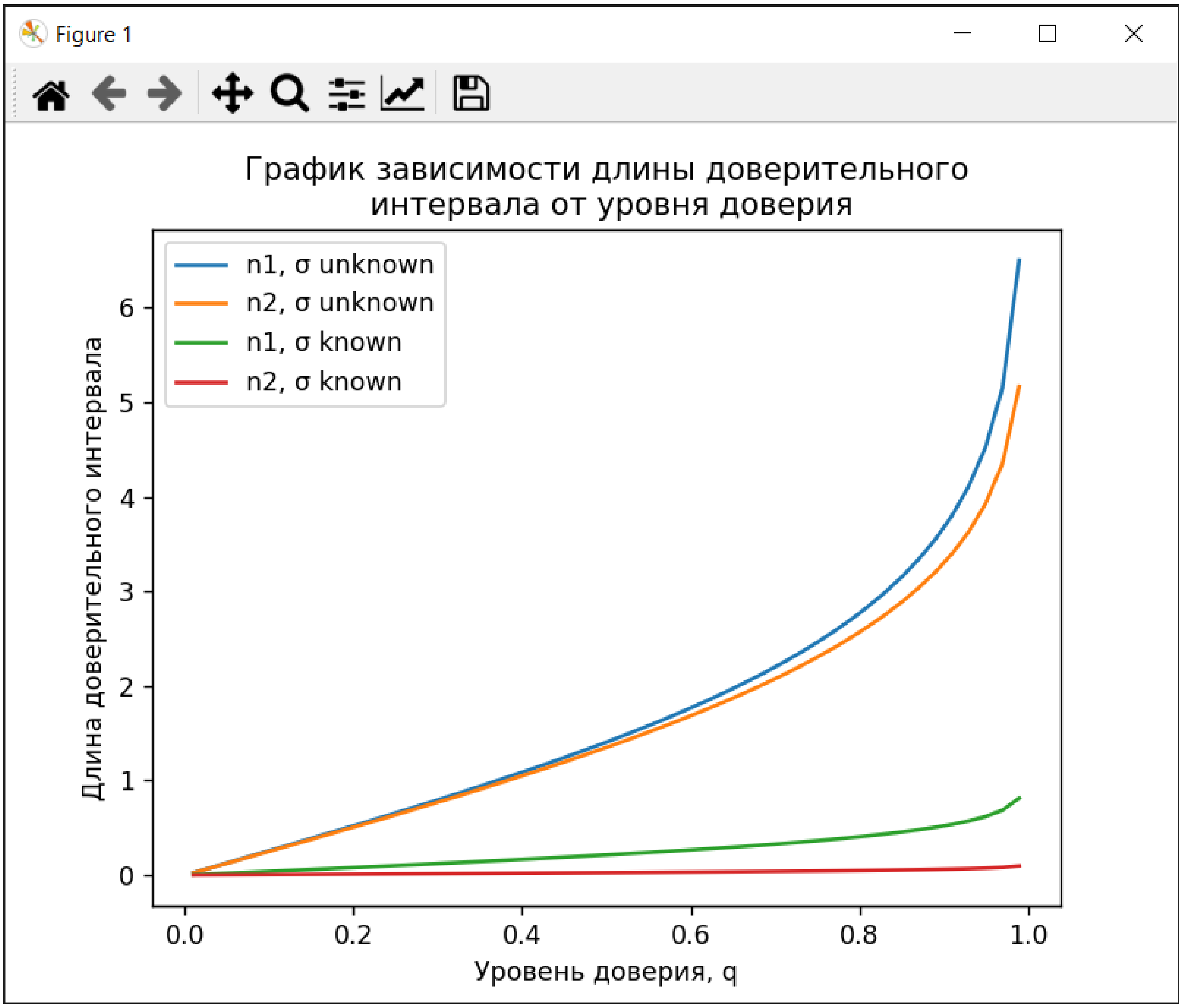

Рис. 1.2. Результат

Графики для случаев с неизвестной σ имеют более пологий и быстрый рост длины интервала по сравнению с графиками для случаев с известной σ. Для малого объема выборки (n_1) доверительный интервал должен быть шире для достижения заданного уровня доверия q. При большом объеме выборки (n_2) можно получить более узкий доверительный интервал при заданном уровне доверия q.

Вывод: в ходе выполнения домашней работы были получены практические навыки обработки статистических данных.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Гмурман, В.Е. Теория вероятностей и математическая статистика: учеб. пособие для вузов/ В.Е. Гмурман. - М.: Юрайт, 2014. – 479 с. 21
2. Гринь, А.Г. Вероятность и статистика [Электронный ресурс]: учебное пособие/ А.Г. Гринь.— Омск: Омский государственный университет им. Ф.М. Достоевского, 2013.— 304 с.— Режим доступа: http://www.iprbookshop.ru/24879.html
3. Кельберт М.Я. Вероятность и статистика в примерах и задачах [Электронный ресурс]/ Кельберт М.Я. Сухов Ю.М.. - М.: МЦНМО, 2010. - Т. 1. Основные понятия теории вероятностей и математической статистики. - 486 с. - URL: //biblioclub.ru/index.php?page=book&id=69109