



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Расчётно-графическая работы №2
«Проверка статистических гипотез.
Линейные статистические модели»
по дисциплине «Математическая статистика»
Вариант 4

Выполнили:

студенты группы Р3213

Поленов К.А.

Пименова Е.А.

Проверила:

Милованович Е.В.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1. Цель работы.....	3
2. Задачи работы	3
ЗАДАНИЕ 1.....	4
1. Текст задания	4
2. Выполнение задания	4
Код задания 1.....	8
ЗАДАНИЕ 2.....	11
1. Текст задания	11
2. Выполнение задания	11
Код задания 2	13
ЗАКЛЮЧЕНИЕ.....	15
ПРИЛОЖЕНИЕ 1	Ошибка! Закладка не определена.

ВВЕДЕНИЕ

1. Цель работы

Целью данной расчётно-графической работы является изучение методов построения доверительных интервалов для параметров распределений, а также исследование их свойств при различных объёмах выборок с использованием численных экспериментов и программных инструментов.

2. Задачи работы

Задачами данной расчётно-графической работы являются:

- 1) Построение доверительного интервала для заданного параметра распределения с использованием теоретических функций;
- 2) Проведение численного эксперимента для оценки покрытия распределения;
- 3) Разработка асимптотического доверительного интервала для параметра однопараметрического распределения;
- 4) Анализ поведения распределения в аналогичных экспериментах;
- 5) Реализация алгоритмов на языке программирования Python с применением библиотек математической статистики и визуализации данных;
- 6) Интерпретация результатов программной реализации алгоритмов.

ЗАДАНИЕ 1

1. Текст задания

Предъявить доверительный интервал уровня $1 - \alpha$ для указанного параметра при данных предположениях (с математическими объяснениями). Сгенерировать 2 выборки объёма 25 и посчитать доверительный интервал. Повторить 1000 раз. Посчитать, сколько раз 95-процентный доверительный интервал покрывает реальное значение параметра. То же самое сделать для объёма выборки 10000. Как изменился результат? Как объяснить? Что изменяется при росте объёмов выборок?

Задача представлена в варианте ниже. Даны две независимые выборки X_1 , X_2 из нормальных распределений $\mathcal{N}(\mu_1, \sigma_1^2)$, $\mathcal{N}(\mu_2, \sigma_2^2)$ объёмов n_1 , n_2 соответственно. Сначала указывается оцениваемая функция, потом данные об остальных параметрах, затем параметры эксперимента и подсказки:

$$\tau = \frac{\sigma_1^2}{\sigma_2^2}; \mu_1 = 0, \mu_2 = 0, \sigma_1^2 = 2, \sigma_2^2 = 1$$

Воспользоваться функцией:

$$\frac{n_2 \sum_{i=1}^{n_1} (X_{1,i} - \mu_1)^2}{n_1 \sum_{i=1}^{n_2} (X_{2,i} - \mu_2)^2} * \frac{\sigma_2^2}{\sigma_1^2}$$

2. Выполнение задания

В задании нужно построить доверительный интервал для отношения дисперсий двух нормальных выборок. Для этого используется распределение Фишера (F-распределение).

При $X_1 \sim N(\mu_1, \sigma_1^2)$ и $X_2 \sim N(\mu_2, \sigma_2^2)$, получаем $\sum_{i=1}^{n_1} (X_{1,i} - \mu_1)^2 \sim \sigma_1^2 \chi^2(n_1)$ и $\sum_{i=1}^{n_2} (X_{2,i} - \mu_2)^2 \sim \sigma_2^2 \chi^2(n_2)$, где $\chi^2(n)$ – распределение хи-квадрат с n степенями свободы.

Отношение масштабированных сумм: $\frac{\sum_{i=1}^{n_1} \frac{(\chi_{1,i} - \mu_1)^2}{n_1}}{\sum_{i=1}^{n_2} \frac{(\chi_{2,i} - \mu_2)^2}{n_2}} * \frac{\sigma_2^2}{\sigma_1^2} \sim F(n_1, n_2)$, где

$F(n_1, n_2)$ – распределение Фишера со степенями свободы n_1 и n_2

Из определения F-распределения получаем:

$$P\left(F_{\frac{\alpha}{2}}(n_1, n_2) \leq \frac{\sum_{i=1}^{n_1} \frac{(\chi_{1,i} - \mu_1)^2}{n_1}}{\sum_{i=1}^{n_2} \frac{(\chi_{2,i} - \mu_2)^2}{n_2}} * \frac{\sigma_2^2}{\sigma_1^2} \leq F_{1-\frac{\alpha}{2}}(n_1, n_2)\right) = 1 - \alpha$$

Решая неравенство относительно $\tau = \frac{\sigma_2^2}{\sigma_1^2}$ получаем:

$$\tau \in \left[\frac{\sum_{i=1}^{n_1} \frac{(X_{1,i} - \mu_1)^2}{n_1}}{\sum_{i=1}^{n_2} \frac{(X_{2,i} - \mu_2)^2}{n_2}} * \frac{1}{F_{1-\frac{\alpha}{2}}(n_1, n_2)}, \frac{\sum_{i=1}^{n_1} \frac{(X_{1,i} - \mu_1)^2}{n_1}}{\sum_{i=1}^{n_2} \frac{(X_{2,i} - \mu_2)^2}{n_2}} * \frac{1}{F_{\frac{\alpha}{2}}(n_1, n_2)} \right]$$

Теперь сгенерируем 1000 выборок для $n_1 = n_2 = 25$ и $n_1 = n_2 = 10000$ и сравним результаты работы программы. По результатам работы программы можно заметить, что при разных размерах выборки, доля покрытия там и там близка к 95% (с большим n разброс уменьшается и доля покрытия ближе к 95%, но всего на несколько десятых).

Визуализация доверительных интервалов для p при малом объёме выборки при $n = 25$ представлена на рисунке 1.

Boxplot границ дов. инт-а уровня доверия 0.95 для выборки 25

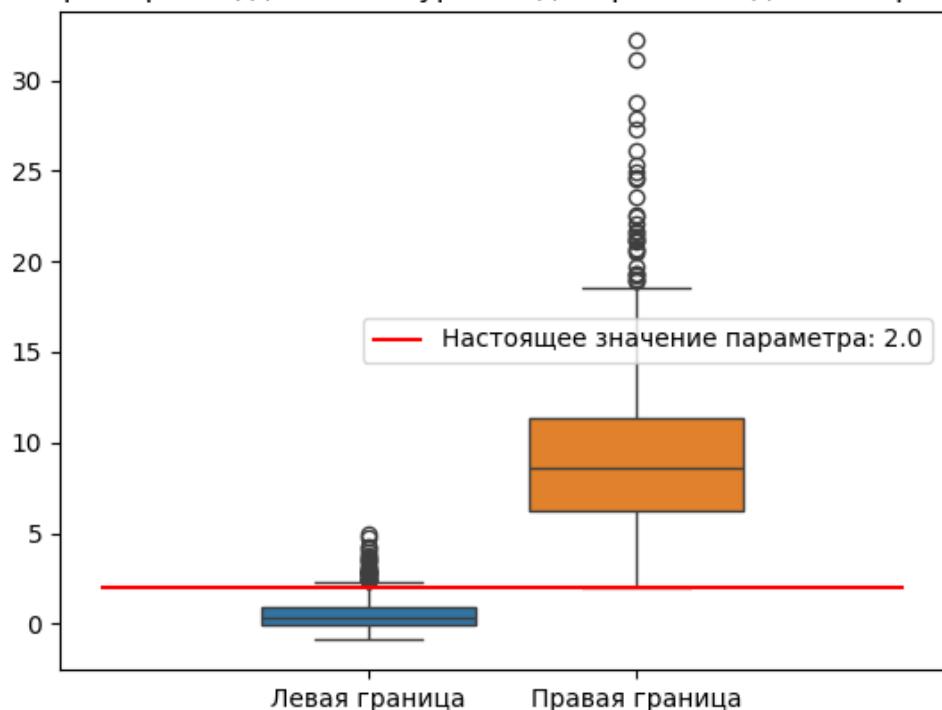


Рис. 1

95% доверительные интервалы для τ ($n=25$)

Boxplot длин дов. инт-а уровня доверия 0.95 для выборки 25

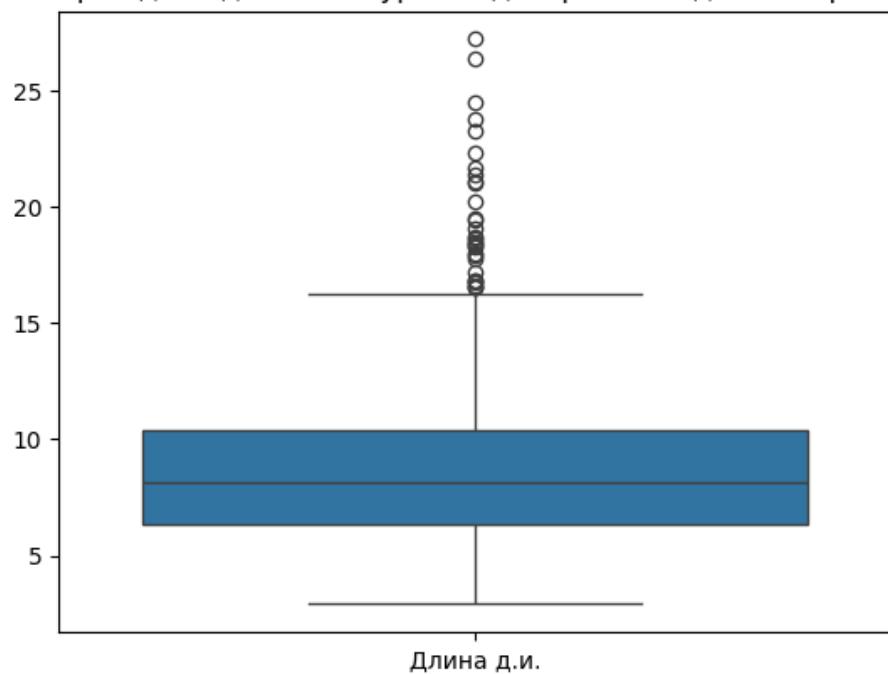


Рис. 2

длины 95% доверительных интервалов для τ ($n=25$)

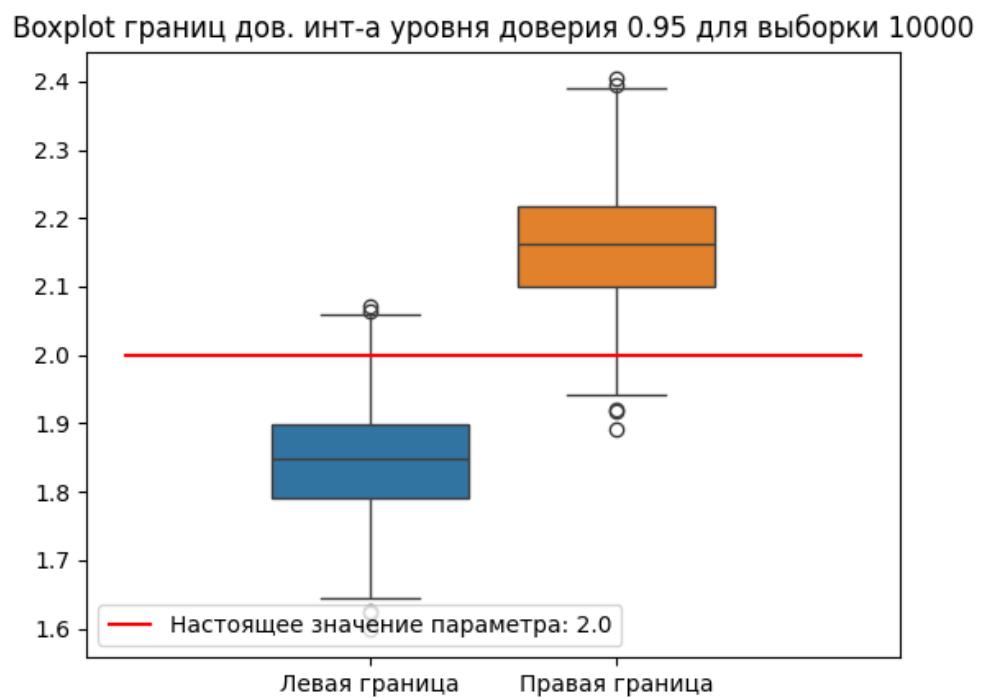


Рис. 3 95% доверительные интервалы для τ ($n=10000$)

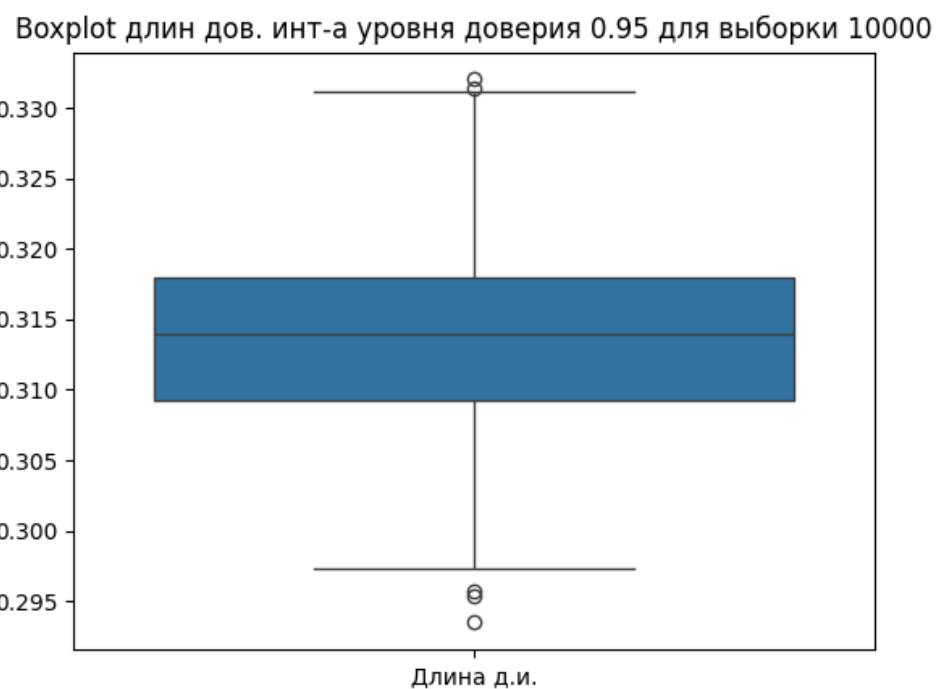


Рис. 4 длины 95% доверительных интервалов для τ ($n=25$)

КОД ЗАДАНИЯ 1

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import scipy
import seaborn as sns

real_mean_1 = 0
real_mean_2 = 0
real_sd_1 = 2
real_sd_2 = 1

sample_count = 1000
sample_size_1 = 25
sample_size_2 = 10_000

alpha = 0.05

sample1_25 = scipy.stats.norm.rvs(loc=real_mean_1, scale=real_sd_1,
size=(sample_count, sample_size_1))
sample2_25 = scipy.stats.norm.rvs(loc=real_mean_2, scale=real_sd_2,
size=(sample_count, sample_size_1))

quantile_left = scipy.stats.f.ppf(alpha / 2, sample_size_1 - 1, sample_size_1 - 1)
quantile_right = scipy.stats.f.ppf(1 - alpha / 2, sample_size_1 - 1, sample_size_1 - 1)

lefts = []
rights = []
for i in range(sample_count):
    sum_n1 = np.sum((sample1_25[i] - real_mean_1) ** 2)
    sum_n2 = np.sum((sample2_25[i] - real_mean_2) ** 2)

    ratio = sum_n1 / sum_n2

    left = (1 / quantile_right) * ratio
    right = (1 / quantile_left) * ratio
    rights.append(right)
    lefts.append(left)

good_intervals = 0
tau_real = real_sd_1 / real_sd_2
for i in range(sample_count):
    if lefts[i] <= tau_real <= rights[i]:
        good_intervals += 1

print(f"Количество экспериментов, в которых доверительный интервал покрывает реальный параметр: {good_intervals}")
print(f"Общее количество экспериментов: {sample_count}")
print(f"Эмпирический процент попадания: {good_intervals / sample_count},\nтеоретический уровень доверия: {1 - alpha}")

combined_data = pd.DataFrame(np.vstack((lefts, rights)).T, columns=["Левая граница",
"Правая граница"])
plot = sns.boxplot(combined_data)
plot.set_title(f"Boxplot границ дов. инт-а уровня доверия {1 - alpha} для выборки {sample_size_1}")
plot.hlines(tau_real, xmin = -1, xmax = 2, colors = ["Red"], label = f"Настоящее значение параметра: {tau_real}")
plt.legend()
plt.show()

ci_length = pd.DataFrame(np.array([rights[i] - lefts[i] for i in
range(sample_count)]), columns=["Длина д.и."])
```

```

print(ci_length.describe())
plot = sns.boxplot(ci_length)
plot.set_title(f"Boxplot длин дов. инт-а уровня доверия {1 - alpha} для выборки
{sample_size_1}")
plt.show()

sample1_10_000 = scipy.stats.norm.rvs(loc=real_mean_1, scale=real_sd_1,
size=(sample_count, sample_size_2))
sample2_10_000 = scipy.stats.norm.rvs(loc=real_mean_2, scale=real_sd_2,
size=(sample_count, sample_size_2))

quantile_left = scipy.stats.f.ppf(alpha / 2, sample_size_2 - 1, sample_size_2 - 1)
quantile_right = scipy.stats.f.ppf(1 - alpha / 2, sample_size_2 - 1, sample_size_2 - 1)

lefts = []
rights = []
for i in range(sample_count):
    sum_n1 = np.sum((sample1_10_000[i] - real_mean_1) ** 2)
    sum_n2 = np.sum((sample2_10_000[i] - real_mean_2) ** 2)

    ratio = sum_n1 / sum_n2

    left = (1 / quantile_right) * ratio
    right = (1 / quantile_left) * ratio
    rights.append(right)
    lefts.append(left)

good_intervals = 0
for i in range(sample_count):
    if lefts[i] <= tau_real <= rights[i]:
        good_intervals += 1

print(f"Количество экспериментов, в которых доверительный интервал покрывает реальный
параметр: {good_intervals}")
print(f"Общее количество экспериментов: {sample_count}")
print(f"Эмпирический процент попадания: {good_intervals / sample_count},
теоретический уровень доверия: {1 - alpha}")

combined_data = pd.DataFrame(np.vstack((lefts, rights)).T, columns=["Левая граница",
"Правая граница"])
plot = sns.boxplot(combined_data)
plot.set_title(f"Boxplot границ дов. инт-а уровня доверия {1 - alpha} для выборки
{sample_size_2}")
plot.hlines(tau_real, xmin = -1, xmax = 2, colors = ["Red"], label = f"Настоящее
значение параметра: {tau_real}")
plt.legend()
plt.show()

ci_length = pd.DataFrame(np.array([rights[i] - lefts[i] for i in
range(sample_count)]), columns=["Длина д.и."])
print(ci_length.describe())
plot = sns.boxplot(ci_length)
plot.set_title(f"Boxplot длин дов. инт-а уровня доверия {1 - alpha} для выборки
{sample_size_2}")
plt.show()

```

результаты работы программы:

```
↑ "E:\python projects\happy_tree\venv\Scripts\python.exe" E:\matstat\lab2\lab2_task1.py
↓ Количество экспериментов, в которых доверительный интервал покрывает реальный параметр: 935
≡ общее количество экспериментов: 1000
≡ Эмпирический процент попадания: 0.935, теоретический уровень доверия: 0.95
≡     Длина д.и.
count 1000.000000
mean    8.807151
std     3.449523
min     2.664105
25%    6.316129
50%    8.146992
75%    10.493832
max     27.570745
Количество экспериментов, в которых доверительный интервал покрывает реальный параметр: 955
общее количество экспериментов: 1000
Эмпирический процент попадания: 0.955, теоретический уровень доверия: 0.95
≡     Длина д.и.
count 1000.000000
mean    0.313818
std     0.006190
min     0.290414
25%    0.309483
50%    0.313834
75%    0.317837
max     0.335280
Process finished with exit code 0
```

ЗАДАНИЕ 2

1. Текст задания

Построить асимптотический доверительный интервал уровня $1 - \alpha$ для указанного параметра. Провести эксперимент по схеме, аналогичной первой задаче.

Вариант задачи представлен ниже. Сначала указывается класс распределений (однопараметрический) и оцениваемый параметр, затем параметры эксперимента:

$$Geom(p); \quad p; \quad p = 0.7$$

2. Выполнение задания

Геометрическое распределение описывает число испытаний до первого успеха в серии независимых экспериментов с фиксированной вероятностью успеха. В данном случае среднее значение случайной величины равно $\frac{1}{p} \approx 1.43$, а дисперсия $\frac{1-p}{p^2} \approx 0.612$.

Для построения интервала использовалась центральная предельная теорема, которая утверждает, что при большом объёме выборки распределение среднего значения стремится к нормальному. Пусть X_1, X_2, \dots, X_n — выборка из $Geom(p)$, тогда выборочное среднее $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$ имеет математическое ожидание $\frac{1}{p}$ и дисперсию $\frac{1-p}{np^2}$. Оценка параметра p задаётся как $\hat{p} = \frac{1}{\bar{X}}$. Применяя асимптотическое приближение, дисперсия \hat{p} оценивается через дельта-метод как $\widehat{Var}(\hat{p}) \approx \frac{1-p}{n\hat{p}^2}$. Таким образом, асимптотический доверительный интервал для p принимает вид $\hat{p} \pm z_{0.975} \sqrt{\frac{1-p}{n\hat{p}^2}}$, где $z_{0.975} = 1.96$ — квантиль стандартного нормального распределения для уровня доверия 95%.

n=25: Генерировалась выборка из 25 элементов, например, $X = \{2, 1, 3, 1, \dots\}$, где каждое значение — число испытаний до успеха при $p = 0.7$. Вычислялось среднее \bar{X} , затем находилась оценка $\hat{p} = \frac{1}{\bar{X}}$, и на её основе строился

интервал $\hat{p} \pm z \sqrt{\frac{1-\hat{p}}{25\hat{p}^2}}$, $z = 1.96$ – квантиль стюдента для 0,95 Процедура повторилась 1000 раз, и подсчитывалась доля случаев, когда интервал включал истинное значение $p = 0.7$. Результат показал покрытие в 94,8% что чуть ниже уровня ожидаемых 95%.

Для $n = 10000$: после 1000 повторений доля покрытий – 94,4%, что соответствует ожидаемому уровню 95%. Увеличение объёма выборки привело к сужению интервала и повышению точности покрытия, что объясняется лучше применимостью центральной предельной теоремы при большом n . Таким образом, при росте объёма выборки асимптотический интервал становится более надёжным и точным.

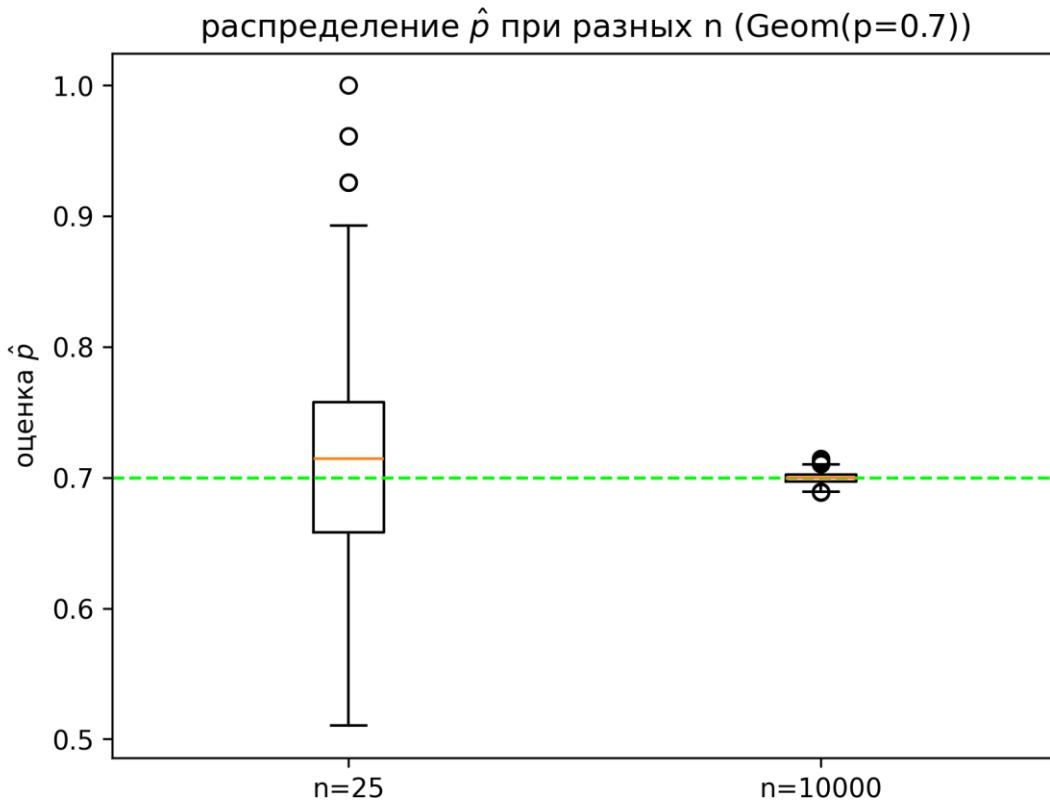


Рис. 2 95% асимптотические доверительные интервалы для $p=0.7$

КОД ЗАДАНИЯ 2

```
import numpy as np
from math import sqrt
import matplotlib.pyplot as plt

P_TRUE = 0.7
STUDENT_K = 1.96

n1 = 25
n2 = 10000

def solve(n, p):
    success = 0
    phats = []

    for _ in range(1000):
        array = np.random.geometric(p=p, size=n)

        x_mid = np.mean(array)
        p_ = 1 / x_mid

        half_width = STUDENT_K * p_ * sqrt((1 - p_) / n)
        trust_interval_left = p_ - half_width
        trust_interval_right = p_ + half_width

        if trust_interval_left <= p <= trust_interval_right:
            success += 1

        phats.append(p_)

    coverage = success / 1000
    return coverage, phats

print(f"геометрическое распределение для размера выборки 25 и"
      f"параметра p={P_TRUE}:")
success_num, phats_n1 = solve(n1, P_TRUE)
print(f"доля попаданий = {success_num*100}%")
print('-' * 65)

print(f"геометрическое распределение для размера выборки 10 000 и"
      f"параметра p={P_TRUE}:")
success_num, phats_n2 = solve(n2, P_TRUE)
print(f"доля попаданий = {success_num*100}%")

data_for_boxplot = [phats_n1, phats_n2]
labels = [f"n={n1}", f"n={n2}"]

plt.boxplot(data_for_boxplot, labels=labels)
plt.axhline(y=P_TRUE, linestyle='--', label='истинное p',
            linewidth='1.2', color='lime')
```

```
plt.title("распределение  $\hat{p}$  при разных n (Geom(p=0.7))")
plt.ylabel("оценка  $\hat{p}$ ")
plt.savefig("graphic.png", dpi=300, bbox_inches='tight')

plt.legend()
plt.show()
```

результаты работы программы:

геометрическое распределение для размера выборки 25 и параметра p=0.7:
доля попаданий = 94.8%

геометрическое распределение для размера выборки 10 000 и параметра p=0.7:
доля попаданий = 94.3999999999999%

Process finished with exit code 0

ЗАКЛЮЧЕНИЕ

В ходе выполнения расчётно-графической работы были освоены методы построения доверительных интервалов для параметров нормального и геометрического распределений, а также навыки программирования на Python с применением библиотек. Получены умения анализировать влияние объёма выборки на точность и покрытие интервалов, интерпретировать результаты численных экспериментов и визуализировать данные.

Работа, состоящая из двух заданий, выполнена в полном объёме