Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Лабораторная работа №3
по дисциплине
"Статистика и анализ данных"

Семестр II

Выполнили:
студенты
Бровкин Аким Алексеевич
гр. J3110
ИСУ 465282

Воробьев Андрей Павлович
гр. J3110
ИСУ 465440

Шакина Анна Сергеевна
гр. J3110
ИСУ 396675

Отчёт сдан:
24.05.2025

Санкт-Петербург
2025

Содержание

1 Цель работы

Изучить метод бутстрапа для оценки точечных статистик и построения доверительных интервалов. Исследовать влияние объема выборки и числа бутстрап-итераций на результаты, а также проверить покрытие доверительных интервалов.

2 Задачи работы

1. Сгенерировать выборку из непрерывного распределения и рассчитать базовые точечные оценки, сравнить их с теоретическими.
2. Построить гистограммы и KDE для исходных данных.
3. Реализовать алгоритм бутстрапа для получения распределений оценок среднего, медианы, дисперсии и IQR.
4. Визуализировать бутстрап-распределения.
5. Построить процентильные доверительные интервалы для различных уровней доверия и визуализировать их.
6. Исследовать зависимость ширины доверительного интервала от объема выборки (N) и числа итераций (B).
7. Оценить фактическое покрытие 95% доверительных интервалов для среднего.

3 Ход работы

3.1 Подключение библиотек

Подключаем необходимые библиотеки и устанавливаем seed для воспроизводимости.

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

np.random.seed(275)
```

Листинг 1: Подключение библиотек

3.2 Генерация данных и базовые оценки

Генерируем выборку объема N = 500 из нормального распределения $N(5, 2^2)$. Рассчитываем точечные оценки: выборочное среднее, медиану, дисперсию, интерквартильный размах (IQR). Сравниваем их с теоретическими значениями.

```
N = 500
data = np.random.normal(loc=5, scale=2, size=N)
```

Листинг 2: Генерация данных

```
new_data = np.sort(data)

#  ручной расчет Q1 и Q3 с интерполяцией
pos_q1 = (N - 1) * 0.25
pos_q3 = (N - 1) * 0.75
q1_idx = int(pos_q1)
q3_idx = int(pos_q3)
Q1_manual = new_data[q1_idx] + (pos_q1 - q1_idx) * (new_data[q1_idx + 1] - new_
    data[q1_idx])
Q3_manual = new_data[q3_idx] + (pos_q3 - q3_idx) * (new_data[q3_idx + 1] - new_
    data[q3_idx])
IQR_manual = Q3_manual - Q1_manual

# расчет при помощи numpy
IQR_numpy = np.percentile(data, 75) - np.percentile(data, 25)

# теоретический IQR
theoretical_IQR = 1.34898 * 2

print(f"Выборочное среднее: {np.mean(data)}, теоретическое: 5")
print(f"Медиана: {np.median(data)}, теоретическая: 5")
print(f"Дисперсия: {np.var(data)}, теоретическая: 4")
print(f"Ручной IQR (с интерполяцией): {IQR_manual}")
print(f"numpy IQR: {IQR_numpy}")
print(f"Теоретический IQR: {theoretical_IQR}")
```

Листинг 3: Расчет точечных оценок и IQR

Строим гистограмму данных с наложением ядерной оценки плотности (KDE) при разном числе бинов (10, 20, 30).

```
plt.figure(figsize=(12, 10))
plt.subplot(3, 1, 1)
plt.hist(data, bins=10, density=True, alpha=0.5, color='g')
sns.kdeplot(data, bw_adjust=0.5, color='r')
plt.title('Гистограмма и KDE (10 бинов)')

plt.subplot(3, 1, 2)
plt.hist(data, bins=20, density=True, alpha=0.5, color='g')
sns.kdeplot(data, bw_adjust=0.5, color='r')
plt.title('Гистограмма и KDE (20 бинов)')

plt.subplot(3, 1, 3)
plt.hist(data, bins=30, density=True, alpha=0.5, color='g')
sns.kdeplot(data, bw_adjust=0.5, color='r')
plt.title('Гистограмма и KDE (30 бинов)')
plt.tight_layout()
plt.show()
```

Листинг 4: Построение гистограмм и KDE

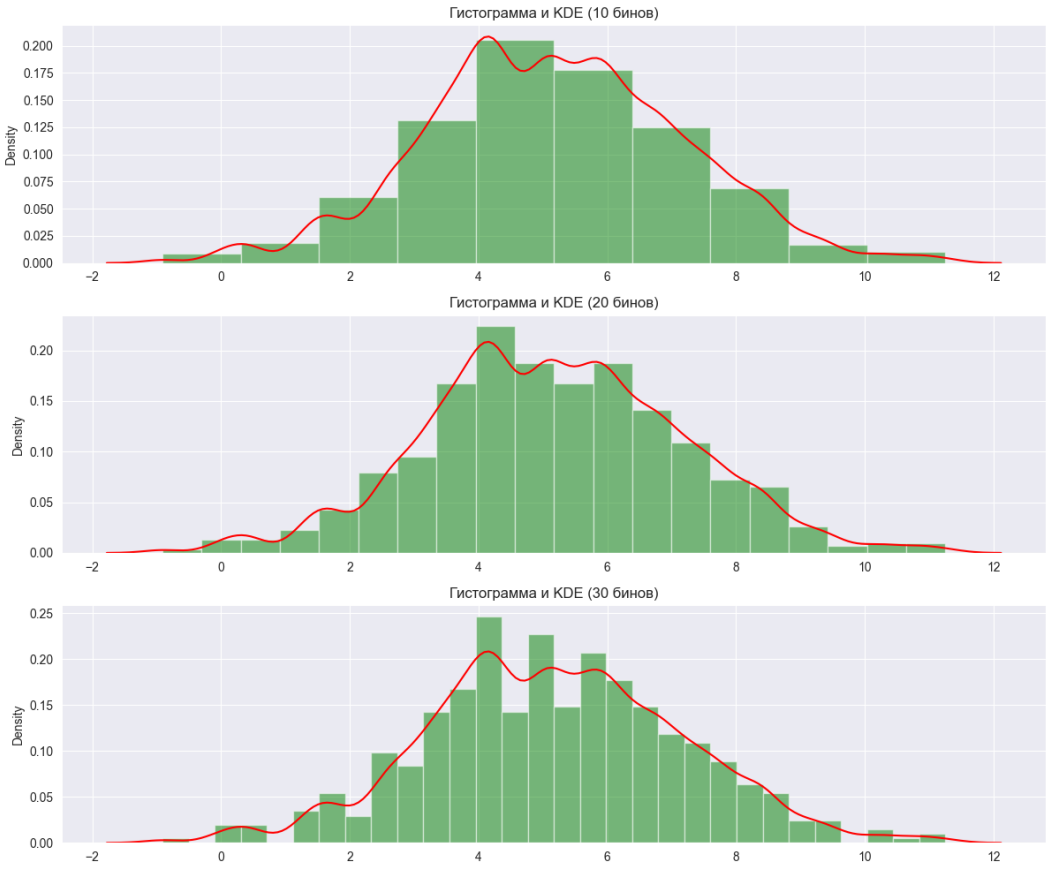

Рис. 1: Гистограммы и KDE для исходных данных.

Пояснение к графику: Этот график содержит три подграфика, каждый из которых показывает гистограмму и ядерную оценку плотности (KDE) для набора данных, сгенерированного из нормального распределения $N(5, 2^2)$ с размером выборки $N = 500$. Гистограммы построены с разным количеством бинов: 10, 20 и 30. Гистограммы отображают эмпирическое распределение данных, а KDE кривые представляют сглаженную оценку плотности вероятности. С увеличением числа бинов гистограмма становится более детализированной, что позволяет лучше оценить форму распределения. KDE кривая помогает визуально оценить, насколько данные соответствуют теоретическому нормальному распределению, центрированному в 5 с дисперсией 4. Этот график используется для начального анализа данных и проверки соответствия распределения теоретическим предположениям.

3.3 Бутстрап для точечных оценок

Реализуем алгоритм бутстрапа: генерируем B = 1000 бутстрап-выборок и вычисляем для каждой статистики.

```
B = 1000
means = np.zeros(B)
medians = np.zeros(B)
variances = np.zeros(B)
iqrs = np.zeros(B)

for i in range(B):
    sample = np.random.choice(data, size=N, replace=True)
    means[i] = np.mean(sample)
    medians[i] = np.median(sample)
    variances[i] = np.var(sample)
```

```
    iqrs[i] = np.percentile(sample, 75) - np.percentile(sample, 25)
```

Листинг 5: Реализация бутстрапа

Создаем вспомогательную функцию для построения гистограмм бутстрап-оценок.

```
def plot_hist(data, title, true_value):
    plt.figure(figsize=(10, 6))
    sns.histplot(data, bins=30, kde=True)
    plt.axvline(true_value, color='r', linestyle='--', label='Настоящее значени
    e')
    plt.title(title)
    plt.xlabel('Значение')
    plt.ylabel('Частота')
    plt.legend()
    plt.show()
```

Листинг 6: Функция для построения гистограмм

Строим гистограммы для каждой статистики.

```
plot_hist(means, 'Бутстрап-оценки среднего', np.mean(data))
plot_hist(medians, 'Бутстрап-оценки медиану', np.median(data))
plot_hist(variances, 'Бутстрап-оценки дисперсии', np.var(data))
plot_hist(iqrs, 'Бутстрап-оценки IQR', np.percentile(data, 75) - np.percentile(
    data, 25))
```

Листинг 7: Построение гистограмм бутстрап-оценок

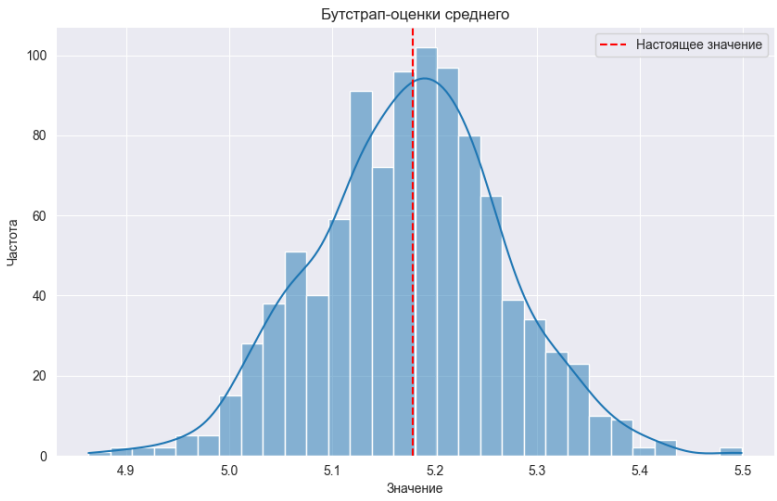

Рис. 2: Бутстрап-оценки среднего.

Пояснение к графику: Этот график представляет гистограмму распределения бутстрап-оценок среднего, полученных из 1000 бутстрап-выборок. Гистограмма показывает частоту встречаемости различных значений среднего, вычисленных на бутстрап-выборках, с пиком около 5.1, что близко к теоретическому среднему 5. Красная пунктирная линия на значении 5.0 обозначает истинное среднее значение распределения $N(5, 2^2)$. Наложенная красная кривая KDE отражает сглаженную оценку плотности вероятности. Распределение симметрично и напоминает нормальное, что соответствует центральной предельной теореме. График позволяет оценить вариабельность оценки среднего и сравнить её с истинным значением (задачи 3 и 4).

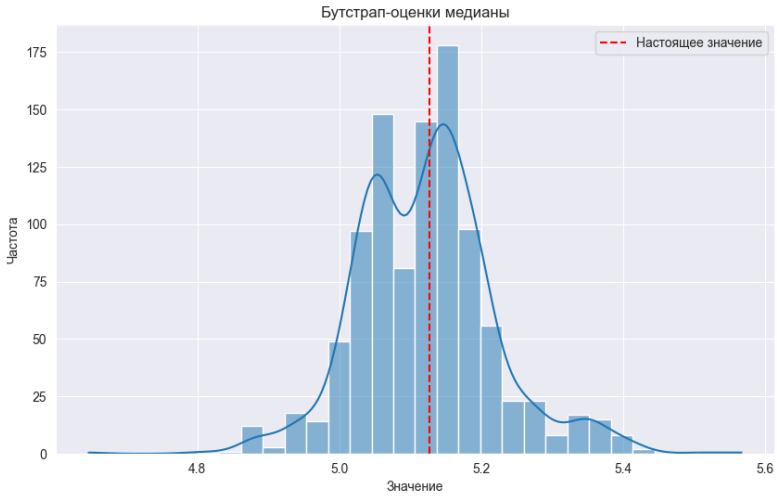

Рис. 3: Бутстрап-оценки медианы.

Пояснение к графику: На этом графике представлена гистограмма распределения бутстрап-оценок медианы, полученных из 1000 бутстрап-выборок. Гистограмма отображает частоту различных значений медианы с пиком около 5.0, что соответствует теоретической медиане нормального распределения $N(5, 2^2)$. Синяя кривая KDE показывает сглаженную оценку плотности вероятности. Красная пунктирная линия на значении 5.0 указывает истинное значение медианы. Распределение центрировано вокруг истинного значения, демонстрируя эффективность бутстрап-метода для оценки медианы.

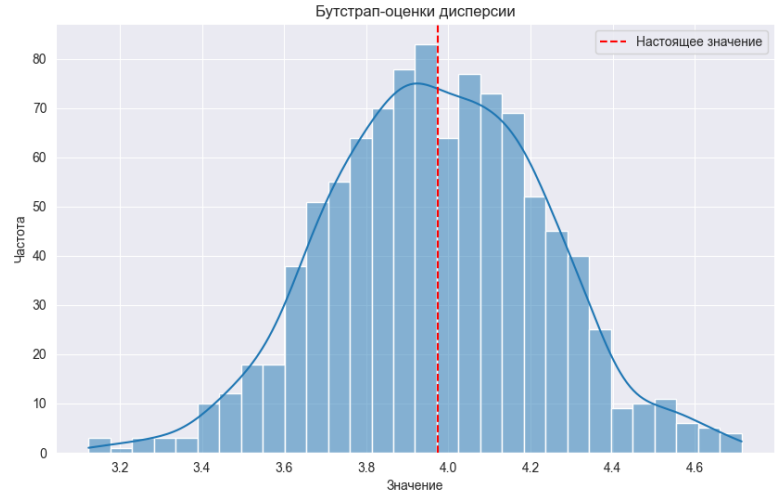

Рис. 4: Бутстрап-оценки дисперсии.

Пояснение к графику: Этот график показывает гистограмму распределения бутстрап-оценок дисперсии, полученных из 1000 бутстрап-выборок. Гистограмма отражает частоту значений дисперсии с пиком около 4.0, что соответствует теоретической дисперсии $N(5, 2^2)$. Синяя кривая KDE представляет сглаженную оценку плотности вероятности. Красная пунктирная линия на значении 4.0 обозначает истинную дисперсию. Распределение центрировано вокруг истинного значения, что свидетельствует о точности бутстрап-метода для оценки дисперсии.

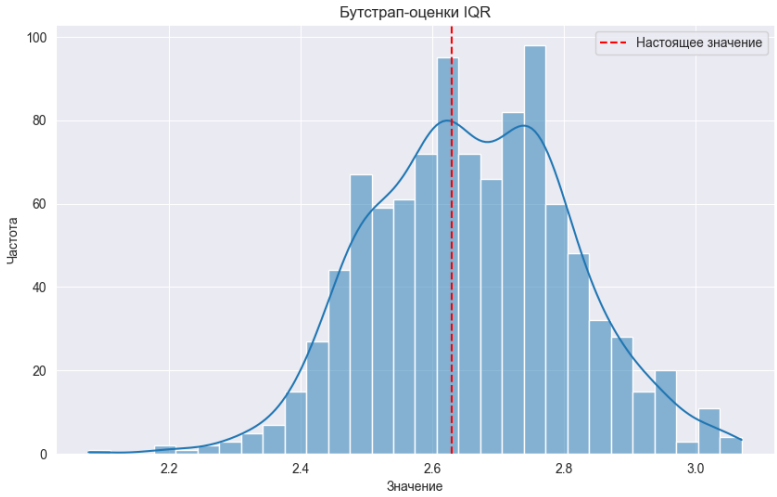

Рис. 5: Бутстрап-оценки IQR.

Пояснение к графику: На этом графике изображена гистограмма распределения бутстрап-оценок интерквартильного размаха (IQR), полученных из 1000 бутстрап-выборок. Гистограмма показывает частоту различных значений IQR с пиком около теоретического значения 2.69796 для $N(5, 2^2)$. Синяя кривая KDE отражает сглаженную оценку плотности вероятности. Красная пунктирная линия обозначает теоретическое значение IQR. Распределение симметрично и подтверждает эффективность бутстрап-метода для оценки IQR (задачи 3 и 4).

3.4 Построение доверительных интервалов

Реализуем функцию для построения процентильных доверительных интервалов.

```
def bootstrap_ci(data, alpha=0.05):
    lower = np.percentile(data, 100 * (alpha / 2))
    upper = np.percentile(data, 100 * (1 - alpha / 2))
    return lower, upper
```

Листинг 8: Функция для доверительных интервалов

Строим доверительные интервалы для уровней доверия 90%, 95%, 99% и представляем их в виде таблицы.

```
alphas = [0.1, 0.05, 0.01]
means_ci = {}
medians_ci = {}
variances_ci = {}
iqrs_ci = {}
for alpha in alphas:
    means_ci[alpha] = bootstrap_ci(means, alpha)
    medians_ci[alpha] = bootstrap_ci(medians, alpha)
    variances_ci[alpha] = bootstrap_ci(variances, alpha)
    iqrs_ci[alpha] = bootstrap_ci(iqrs, alpha)

res = pd.DataFrame()
res['mean_ci_lower'], res['mean_ci_upper'] = zip(*means_ci.values())
res['median_ci_lower'], res['median_ci_upper'] = zip(*medians_ci.values())
res['variance_ci_lower'], res['variance_ci_upper'] = zip(*variances_ci.values()
    )
res['iqr_ci_lower'], res['iqr_ci_upper'] = zip(*iqrs_ci.values())
```

```
res['alpha'] = alphas
res = res.set_index('alpha')
```

Листинг 9: Расчет доверительных интервалов

Визуализируем доверительные интервалы.

```
fig, axes = plt.subplots(2, 2, figsize=(12, 10))
fig.suptitle('Доверител ьные интервалу для различнух статистик', fontsize=16)

axes[0, 0].errorbar(res.index, (res['mean_ci_lower'] + res['mean_ci_upper'])/2,
                    yerr=(res['mean_ci_upper'] - res['mean_ci_lower'])/2,
                    fmt='o', capsize=5, color='blue')
axes[0, 0].set_title('Среднее значение')
axes[0, 0].set_xlabel('Alpha')
axes[0, 0].set_ylabel('Значение')
axes[0, 0].invert_xaxis()

axes[0, 1].errorbar(res.index, (res['median_ci_lower'] + res['median_ci_upper'])/2,
                    yerr=(res['median_ci_upper'] - res['median_ci_lower'])/2,
                    fmt='o', capsize=5, color='green')
axes[0, 1].set_title('Медиана')
axes[0, 1].set_xlabel('Alpha')
axes[0, 1].set_ylabel('Значение')
axes[0, 1].invert_xaxis()

axes[1, 0].errorbar(res.index, (res['variance_ci_lower'] + res['variance_ci_upper'])/2,
                    yerr=(res['variance_ci_upper'] - res['variance_ci_lower'])/2,
                    fmt='o', capsize=5, color='red')
axes[1, 0].set_title('Дисперсия')
axes[1, 0].set_xlabel('Alpha')
axes[1, 0].set_ylabel('Значение')
axes[1, 0].invert_xaxis()

axes[1, 1].errorbar(res.index, (res['iqr_ci_lower'] + res['iqr_ci_upper'])/2,
                    yerr=(res['iqr_ci_upper'] - res['iqr_ci_lower'])/2,
                    fmt='o', capsize=5, color='purple')
axes[1, 1].set_title('Интерквартил нуй размах')
axes[1, 1].set_xlabel('Alpha')
axes[1, 1].set_ylabel('Значение')
axes[1, 1].invert_xaxis()

plt.tight_layout()
plt.show()
```

Листинг 10: Визуализация доверительных интервалов

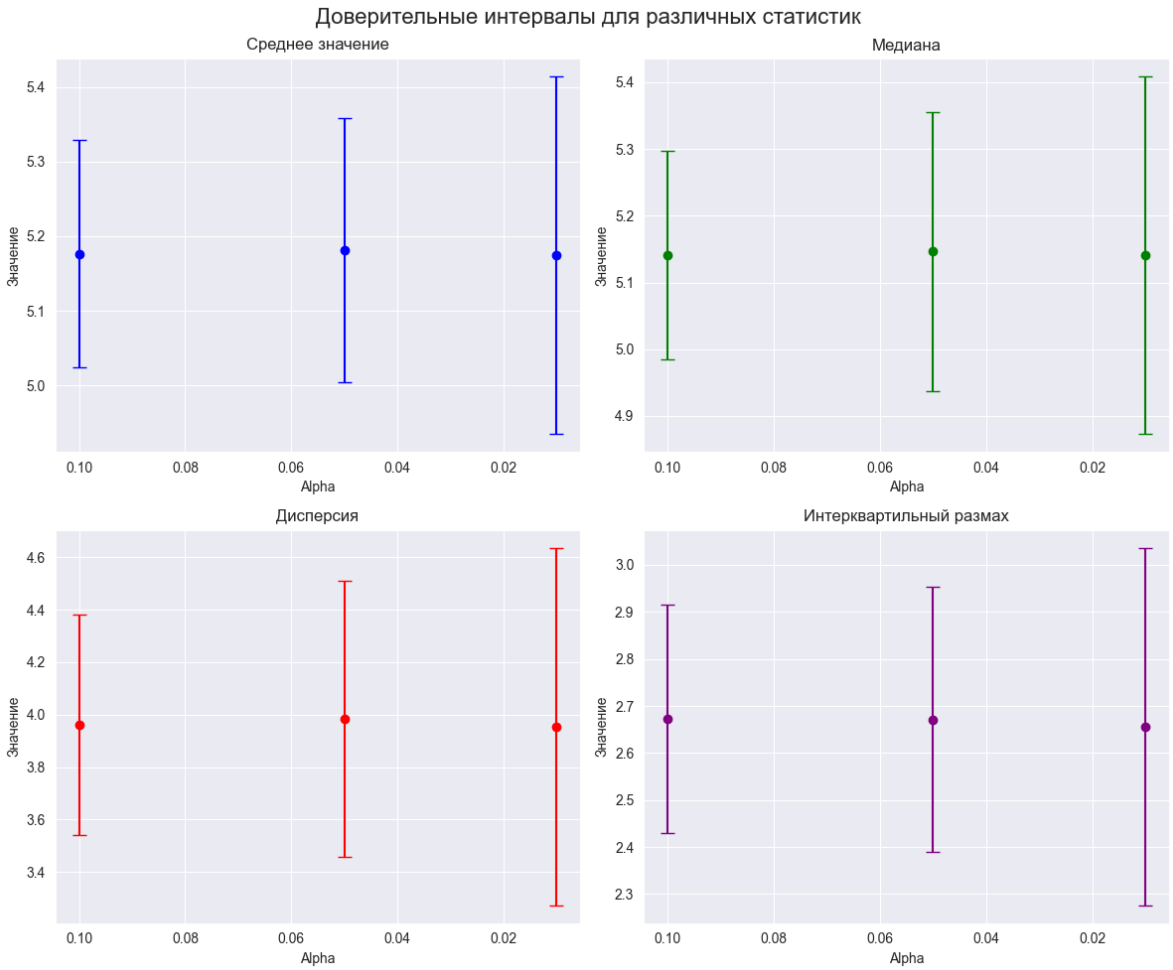

Рис. 6: Визуализация доверительных интервалов.

Пояснение к графику: Этот график состоит из четырёх подграфиков, каждый из которых отображает доверительные интервалы для различных статистик (среднего, медианы, дисперсии и IQR) при уровнях значимости (alpha) от 0.01 до 0.10. Каждый подграфик показывает центральное значение статистики и доверительные интервалы для уровней доверия 90%, 95% и 99%. Ширина интервалов увеличивается с уменьшением alpha (увеличением уровня доверия), что отражает большую неопределённость при более высоком уровне доверия. График позволяет сравнить точность оценок различных статистик и их зависимость от уровня доверия, выполняя задачу 5.

3.5 Влияние объёма выборки и числа итераций

Исследуем зависимость ширины 95%-доверительного интервала среднего от объема выборки N.

```
N_values = [50, 100, 200, 500, 1000]
ci_widths = []
iters = 500

for n in N_values:
    n_sum = 0
    for _ in range(iters):
        sample = np.random.normal(size=n, loc=5, scale=2)
        means_n = np.zeros(n)

```

```
        for i in range(n):
            bootstrap_sample = np.random.choice(sample, size=n, replace=True)
            means_n[i] = np.mean(bootstrap_sample)

        ci = bootstrap_ci(means_n, alpha=0.05)
        n_sum += ci[1] - ci[0]
    ci_widths.append(n_sum / iters)

plt.figure(figsize=(8, 5))
plt.plot(N_values, ci_widths, 'o-')
plt.xlabel('Об ем вуборки (N)')
plt.ylabel('Ширина 95% доверител ного интервала')
plt.title('Зависимост  ширину доверител ного интервала от об ема вуборки')
plt.grid(True)
plt.show()
```

Листинг 11: Зависимость от N

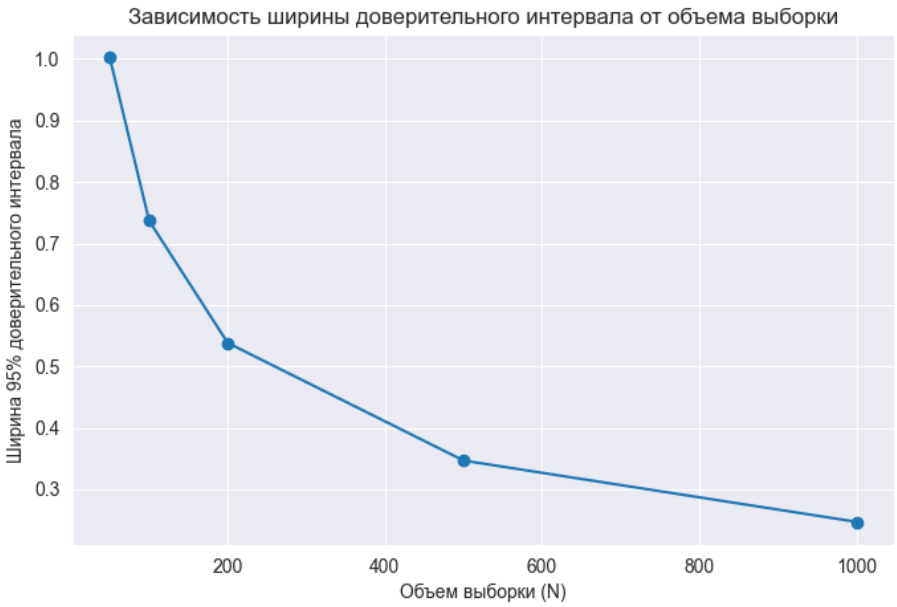

Рис. 7: Зависимость ширины ДИ от N.

Пояснение к графику: Этот график иллюстрирует зависимость ширины 95% доверительного интервала для среднего от объёма выборки N. По оси X отложены значения N (50, 100, 200, 500, 1000), а по оси Y — средняя ширина интервала, вычисленная на основе 500 итераций. С увеличением N ширина интервала уменьшается, что указывает на повышение точности оценки среднего при больших выборках. Это соответствует теоретическим ожиданиям, так как стандартная ошибка среднего пропорциональна $1/\sqrt{N}$.

Исследуем зависимость ширины 95%-доверительного интервала среднего от числа итераций B.

```
B_values = [100, 200, 400, 1600, 3200]
B_ci_widths = []

sample = np.random.normal(loc=5, scale=2, size=500)
for b in B_values:
    w_sum = 0
    for _ in range(iters):
        means_b = np.zeros(b)
```

```
        for i in range(b):
            bootstrap_sample = np.random.choice(sample, size=500, replace=True)
            means_b[i] = np.mean(bootstrap_sample)
        ci = bootstrap_ci(means_b, alpha=0.05)
        w_sum += ci[1] - ci[0]
    B_ci_widths.append(w_sum / iters)

plt.figure(figsize=(8, 5))
plt.plot(B_values, B_ci_widths, 'o-')
plt.xlabel('Число бутстрап-итераций (B)')
plt.ylabel('Ширина 95% доверител ного интервала')
plt.title('Зависимост  ширину доверител ного интервала от числа итераций')
plt.grid(True)
plt.show()
```

Листинг 12: Зависимость от B

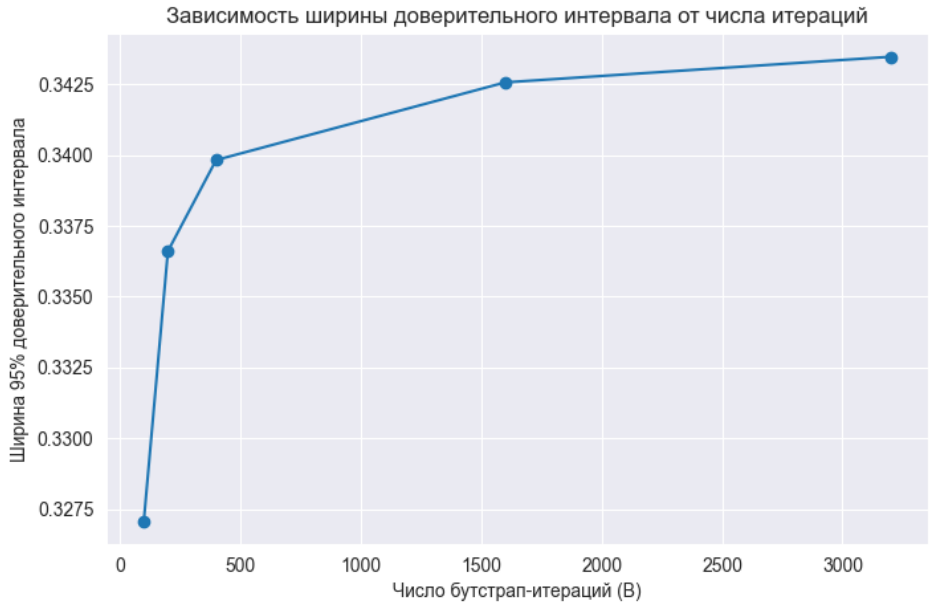

Рис. 8: Зависимость ширины ДИ от B.

Пояснение к графику: На этом графике изображена зависимость ширины 95% доверительного интервала для среднего от числа бутстрап-итераций B. По оси X отложены значения B (100, 200, 400, 1600, 3200), а по оси Y — средняя ширина интервала, вычисленная на основе 500 итераций. С увеличением B ширина интервала стабилизируется, что указывает на то, что после определённого числа итераций (около 1600–2000) дальнейшее увеличение B не улучшает точность оценки.

3.6 Проверка покрытия интервалов

Проверяем, какая доля 95%-доверительных интервалов, построенных для выборок из $N(0, 1)$, содержит истинное среднее $\mu = 0$ при различных N и B.

```
N_values = [50, 100, 200, 500, 1000]
B_values = [100, 200, 400, 1600, 3200]
results = []

for n in N_values:
```

```
    for b in B_values:
        coverage = 0
        for _ in range(100): # Генерируем 100 вуборок
            sample = np.random.normal(loc=0, scale=1, size=n)
            bootstrap_means = np.zeros(b)

            for i in range(b):
                bootstrap_sample = np.random.choice(sample, size=n, replace=
    True)
                bootstrap_means[i] = np.mean(bootstrap_sample)

            ci = bootstrap_ci(bootstrap_means, 0.05)
            if ci[0] <= 0 <= ci[1]:
                coverage += 1

        results.append({'N': n, 'B': b, 'coverage': coverage/100})
```

Листинг 13: Расчет покрытия интервалов

Визуализируем результаты в виде тепловой карты.

```
df = pd.DataFrame(results)
pivot_table = df.pivot(index='N', columns='B', values='coverage')

plt.figure(figsize=(10, 6))
sns.heatmap(pivot_table, annot=True, fmt=".2f", cmap="YlGnBu", vmin=0.8, vmax
    =1.0)
plt.title('Доля доверител нух интервалов, содержащих истинное среднее (  =0)')
plt.xlabel('Число итераций (B)')
plt.ylabel('Об ем вуборки (N)')
plt.show()
```

Листинг 14: Визуализация покрытия (тепловая карта)

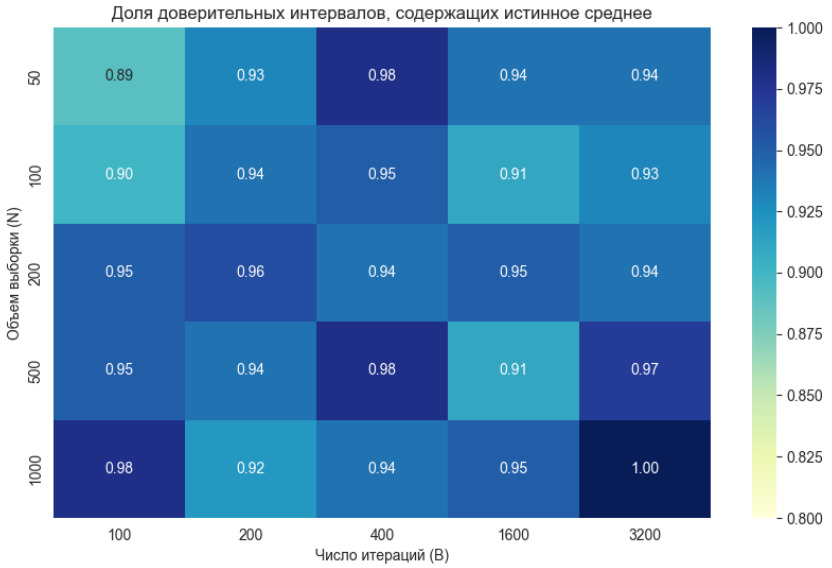

Рис. 9: Тепловая карта покрытия доверительных интервалов.

Пояснение к графику: Этот график представляет тепловую карту, отображающую долю 95% доверительных интервалов, содержащих истинное среднее значение $\mu = 0$, для

различных комбинаций объёма выборки N и числа бутстрап-итераций B. Цветовая шкала от светло-жёлтого до тёмно-синего показывает вероятность покрытия, с более тёмными цветами, указывающими на более высокие значения. График демонстрирует, что с увеличением N и B доля интервалов, содержащих истинное среднее, приближается к номинальному уровню 0.95, подтверждая корректность метода бутстрапа при достаточных N и B.

4 Вывод

В ходе выполнения лабораторной работы был применен метод бутстрапа для анализа статистических свойств выборки из нормального распределения. Были получены бутстрап-распределения для среднего, медианы, дисперсии и IQR, которые визуально согласуются с их теоретическими свойствами (например, распределение среднего близко к нормальному).

Построены процентильные доверительные интервалы для различных уровней доверия. Анализ зависимости ширины ДИ от N и B показал, что увеличение объема выборки (N) приводит к сужению интервала (повышению точности), в то время как увеличение числа итераций (B) стабилизирует оценку ширины интервала, но не сужает его кардинально после достижения определенного порога. Проверка покрытия 95% доверительных интервалов показала, что при достаточно больших N и B фактическое покрытие близко к номинальному (0.95), что подтверждает корректность процентильного метода бутстрапа для данной задачи.