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

Лабораторная работа №2 по дисциплине “Статистика и анализ данных”

Семестр 2

Выполнили студенты:

Косарев Илья,
гр. J3110, ИСУ 466304

Капустина Юлия,
гр. J3110, ИСУ 466110

Кащеев Максим,
гр. J3111, ИСУ 466147

Отчет сдан:
19.05.2025

Санкт-Петербург,
2025 г.

Содержание

Цели и задачи

Цель работы: Исследовать дискретные и непрерывные случайные распределения на практике.

Задачи:

1. Сгенерировать дискретное и непрерывное распределение(биномиальное и экспоненциальное).
2. Вычислить основные характеристики для данных расппределений.
3. Построить графики и сравнить теоретические данные и результаты полученные в ходе эксперемента.
4. Добавить выбросы в непрерывное распределенияч для исследования устойчивости характеристик.
5. Сделать общий вывод по результатам проделанной работы.

Теоретическая часть

Для выполнения данной лабораторной работы мы выбрали два распределения: биномиальное и экспоненциальное.

Биномиальное распределение

Биномиальное распределение описывает количество успехов в серии из n независимых испытаний Бернулли с постоянной вероятностью успеха p.

Функция вероятности задаётся формулой:

$$P(X = k) = C_n^k p^k (1-p)^{n-k}, \quad k = 0, 1, 2, \ldots, n$$

где:

- $C_n^k = \binom{n}{k} = \frac{n!}{k!(n-k)!}$ - биномиальный коэффициент
- n - количество испытаний
- p - вероятность успеха в одном испытании
- k - количество успехов

Функция распределения(CDF) имеет вид:

$$F(k) = P(X \leq k) = \sum_{i=0}^{\lfloor k \rfloor} C_n^i p^i (1-p)^{n-i}$$

Экспоненциальное распределение

Экспоненциальное распределение - это непрерывное распределение, моделирующее время между событиями в пуассоновском процессе (где события происходят независимо с постоянной средней интенсивностью).

Случайная величина X имеет экспоненциальное распределение с параметром $\lambda > 0$ (интенсивность), если её плотность вероятности (PDF) задаётся:

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0. \end{cases}$$

Функция распределения (CDF):

$$F(x) = P(X \leq x) = 1 - e^{-\lambda x}, \quad x \geq 0.$$

Практическая часть

Ссылка на репозиторий с кодом: Лабораторная работа
Результаты

Результаты для биномиального распределения:

- **Меры центральной тенденции**:
  - Первый квартиль (Q1) = 2.0000
  - Медиана (Q2) = 3.0000
  - Третий квартиль (Q3) = 4.0000
  - Среднее значение = 3.0480
  - Мода = 3 (встречается раз: 3)

  По мерам центральной тенденции можно сделать вывод о том, что данные симметричны (первый и третий квартиль совпадают, а медиана близка к моде).
- **Меры вариабельности**:
  - Размах = 8.0000
  - Интерквартильный размах (IQR) = 2.0000
  - Дисперсия = 2.0197
  - Стандартное отклонение = 1.4212
  - Коэффициент вариации (CV) = 46.6029%
  - Среднее абсолютное отклонение (MAD) = 1.0994

  Средние 50% данных сосредоточены в диапазоне шириной 2 единиц(всего размах 8), следовательно основная масса данных сконцентрирована вокруг медианы. Стандартное отклонение составляет 46.6% от среднего, что означает широкую относительную вариабельность.
- **Меры формы распределения**:
  - Коэффициент асимметрии = 0.2084
  - Коэффициент эксцесса = -0.0687

  Коэффицент асимметрии больше нуля, следовательно у распределения правая ассиметрия(большинство знаяений находятся слева от среднего, более длинном правом хвосте справа), однако смещение несильное. Коэффициент эксцесса близок к нулю(отрицательный), что говорит о

том что пик распределения близок к нормальному, но немного более пологий.

Результаты для экспоненциального распределения:

Меры центральной тенденции:

- Первый квартиль (Q1) = 0.5704
- Медиана (Q2) = 1.3640
- Третий квартиль (Q3) = 2.6793
- Среднее значение = 1.9696
- Мода = 0.0000

По квантилям уже можно сказать что данные смещен

Меры вариабельности:

- Размах = 15.4977
- Интерквартильный размах (IQR) = 2.1089
- Дисперсия = 3.9715
- Стандартное отклонение = 1.9929
- Коэффициент вариации (CV) = 101.1312%
- Среднее абсолютное отклонение (MAD) = 1.4600

50% данных в [0.5704, 2.6793], что значит что большая часть данных снова находится в небольшом промежутке. Большой размах говорит о наличие редких экстремальных значений в правом хвосте. Коэфицент ковариации превышает 100%, что говорит об очень большом разбросе данных.

Меры формы распределения:

- Коэффициент асимметрии = 1.9887
- Коэффициент эксцесса = 5.5493

Коэффициент асимметрии положительный, что говорит о сильном перекос вправо(большинство значений меньше среднего, длинный хвост справа).Коэффициент эксцесса положительный и сильно превышает нуль, следовательно пик гораздо острее чем у нормального распределения.

В ходе работы были построенны следущие графики:

Для визуализации данных были посторенны графики на рисунках 1 и 2.

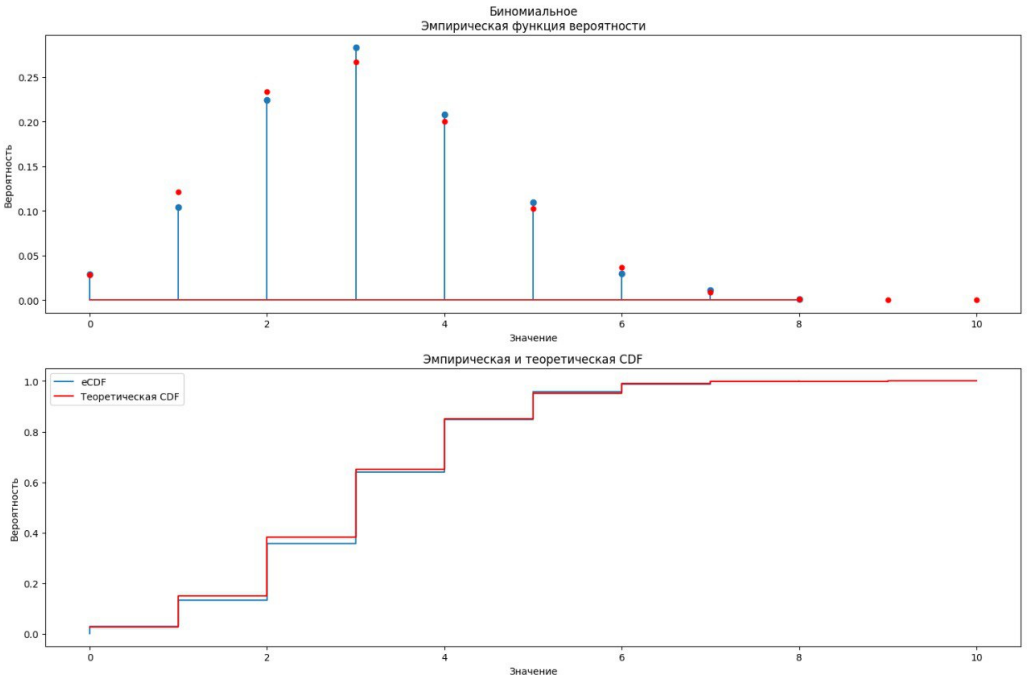

Рис. 0.1: График биномильаного распределения

На графикке изменения мер вариабельности(рис.3) виден рост всех характеристик с увеличением числа выбросов. Особенно неустойчив к выбрасам размах, так как он быстро растет даже при небольшом проценте выбросов. На остальные характеристики также замтно значительное влияние выбросов.

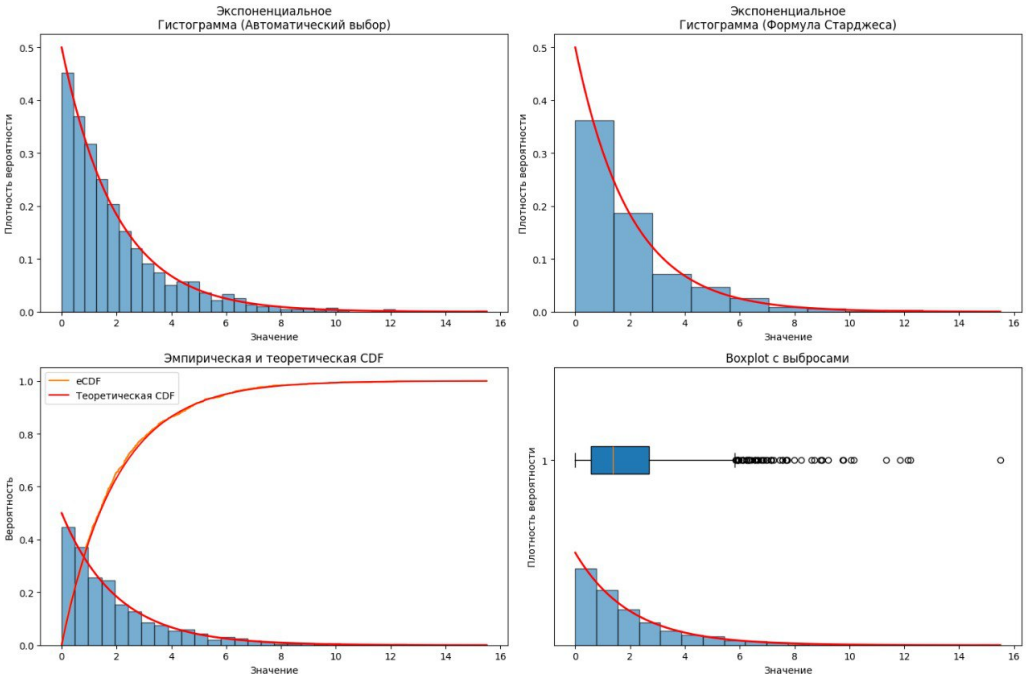

Рис. 0.2: График экспоненциального распределения

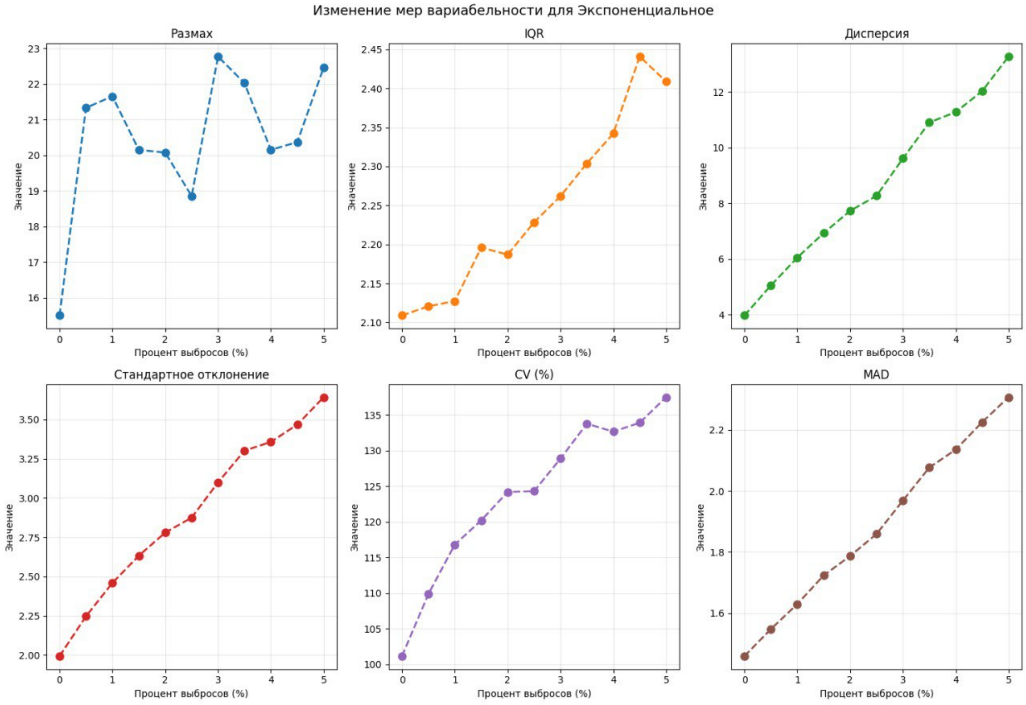

Рис. 0.3: График изменения мер вариабельности

Выводы

В ходе работы была проведена практическая исследовательская деятельность, направленная на изучение дискретных и непрерывных случайных распределений. Были рассмотрены основные виды случайных величин, их свойства и законы распределения, в частности биномиальное распределение как пример дискретного закона и экспоненциального распределение как классический пример непрерывного.

Практические задачи позволили построить функции распределения и графики плотности вероятностей, что дало наглядное представление о поведении случайных величин и их вариативности. Также были проанализированы основные характеристики данных распределений.

Лабораторная работа позволила лучше понять алгоритмы работы с случайными распределениями и применть теоритические знания на практике.

Приложение

0.1. Математические формулы и определения

Квартили

- Q1 (Первый квартиль): Значение, ниже которого находится 25% данных

$$Q1 = x_{(\lceil \frac{n}{4} \rceil)}$$

- Q2 (Медиана): Значение, разделяющее выборку пополам

$$Q2 = \begin{cases} x_{(\frac{n+1}{2})}, & \text{если } n \text{ нечётное} \\ \frac{x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)}}{2}, & \text{если } n \text{ чётное} \end{cases}$$

- Q3 (Третий квартиль): Значение, ниже которого находится 75% данных

$$Q3 = x_{(\lceil \frac{3n}{4} \rceil)}$$

Меры центральной тенденции

- Выборочное среднее:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

- Медиана: см. Q2 выше
- Мода: Наиболее часто встречающееся значение в выборке

Меры вариабельности

- Размах выборки:

$$R = x_{\max} - x_{\min}$$

- Интерквартильный размах:

$$IQR = Q3 - Q1$$

- Выборочная дисперсия(несмещенная):

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

– Стандартное отклонение:

$$s = \sqrt{s^2}$$

– Коэффициент вариации:

$$CV = \frac{s}{\bar{x}} \times 100\%$$

– Среднее абсолютное отклонение:

$$MAD = \frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{x}|$$

Меры формы распределения

– Коэффициент асимметрии:

$$\gamma_1 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{s^3}$$

– Коэффициент эксцесса:

$$\gamma_2 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{s^4} - 3$$

Начальные и центральные моменты

– Начальные моменты порядка k:

$$\alpha_k = \frac{1}{n}\sum_{i=1}^{n}x_i^k$$

– Центральные моменты порядка k:

$$\mu_k = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^k$$

0.2. Код работы

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

from scipy import stats
from scipy.stats import variation
from statistics import mode


# Генерация выборки распределения Бернулли
def bernoulli(n, p, size=1000):
    """
    :param n: Количество испытаний в каждом эксперименте
    :param p: Вероятность успеха при каждом испытании
    :param size: Размер выборки
    :return: Массив сгенерированных данных
    """
    distribution = np.random.binomial(n=n, p=p, size=size)
    return distribution


# Генерация выборки эскпоненциального распределения
def exponential(scale, size=1000):
    """
    :param scale: Масштаб (1/lambda)
    :param size: Размер выборки
    :return: Массив сгенерированных данных
    """
    distribution = np.random.exponential(scale=scale,
    ↪  size=size)
    return distribution


# Вычисление квартилей
def quantiles(part, data):
    """
    :param part: Список квантилей, которые хотим рассчитать
    :param data: Данные на вход
    :return: Массив значений полученных квантилей
    """
    q = np.quantile(data, part)
```

```
40      return q
41
42
43  # Вычисление и вывод мер центральной тенденции
44  def central_stats(data, dist_name):
45      """
46
47      :param data: Данные на вход
48      :param dist_name: Название распределения для вывода в
        ↪ print
49      """
50      print(f"\nМеры центральной тенденции для распределения:
        ↪ {dist_name}:")
51
52      # Квартили
53      q1, q2, q3 = np.quantile(data, [0.25, 0.5, 0.75])
54
55      # Среднее значение
56      mean = np.mean(data)
57
58      # Медиана
59      median = np.median(data)
60
61      # Мода
62      moda = mode(data)
63
64
65      # Вывод результатов
66      print(f"• Первый квартиль (Q1) = {q1:.4f}")
67      print(f"• Медиана (Q2) = {q2:.4f}")
68      print(f"• Третий квартиль (Q3) = {q3:.4f}")
69      print(f"• Среднее значение = {mean:.4f}")
70      print(f"• Медиана = {median:.4f}")
71      print(f"• Мода = {moda}" + (f" (встречается раз: {moda}
        ↪ )" if moda else ""))
72
73
74  #Вычисление и вывод мер вариабельности
75  def variabiliyy_stats(data, dist_name):
76      """
77
```

```
78      :param data: Данные на вход
79      :param dist_name: Название распределения для вывода в
        ↪ print
80      """
81      print(f"\nМеры вариабельности для следующего
        ↪ распределения: {dist_name}:")
82
83      # Размах
84      data_range = np.max(data) - np.min(data)
85
86      # Интерквартильный размах (IQR)
87      q1, q3 = np.quantile(data, [0.25, 0.75])
88      iqr = q3 - q1
89
90      # Дисперсия (несмещенная)
91      variance = np.var(data, ddof=1)
92
93      # Стандартное отклонение (несмещенное)
94      std_dev = np.std(data, ddof=1)
95
96      # Коэффициент вариации (в процентах)
97      cv = variation(data) * 100
98
99      # Среднее абсолютное отклонение
100     mad = np.mean(np.abs(data - np.mean(data)))
101
102     # Вывод результатов
103     print(f"• Размах = {data_range:.4f}")
104     print(f"• Интерквартильный размах (IQR) = {iqr:.4f}")
105     print(f"• Дисперсия = {variance:.4f}")
106     print(f"• Стандартное отклонение = {std_dev:.4f}")
107     print(f"• Коэффициент вариации (CV) = {cv:.4f}%")
108     print(f"• Среднее абсолютное отклонение (MAD) =
        ↪ {mad:.4f}")
109
110 # Вычисление и вывод мер форм распределения
111 def distribution_shape_stats(data, dist_name):
112     """
113
114     :param data: Данные на вход
115     :param dist_name:
```

```
116         :return:
117         """
118         print(f"\nМеры формы распределения {dist_name}:")
119 
120         # Коэффициент асимметрии
121         skewness = stats.skew(data)
122         print(f"• Коэффициент асимметрии = {skewness:.4f}")
123 
124         # Коэффициент эксцесса
125         kurtosis = stats.kurtosis(data)
126         print(f"• Коэффициент эксцесса = {kurtosis:.4f}")
127 
128         # Начальные моменты (1-5)
129         print("\nНачальные моменты:")
130         for k in range(1, 6):
131             moment = np.mean(data ** k)
132             print(f"• M_{k} = {moment:.4f}")
133 
134         # Центральные моменты (1-5)
135         print("\nЦентральные моменты:")
136         for k in range(1, 6):
137             moment = stats.moment(data, moment=k)
138             print(f"• _{k} = {moment:.4f}")
139 
140 # Построение графиков распределения
141 def plot_distributions(data, dist_name, dist_type, params):
142         """
143 
144         :param data: Данные на вход
145         :param dist_name: Название распределения для вывода в
            ↪ print
146         :param dist_type: Тип распределения (непрерывный или
            ↪ дискретный соответственно)
147         :param params: Параметры для распределения
148         """
149         plt.figure(figsize=(15, 10))
150 
151         # Для непрерывных распределений
152         if dist_type == "continuous":
153             # Гистограмма с разными бинами
154             bin_methods = [
```

```
155             ('auto', 'Автоматический выбор'),
156             ('sturges', 'Формула Старджеса'),
157             ('sqrt', 'Квадратный корень'),
158             (20, 'Ручной выбор (20 бинов)')
159         ]
160 
161         for i, (bins, title) in enumerate(bin_methods, 1):
162             plt.subplot(2, 2, i)
163             counts, bins, _ = plt.hist(data, bins=bins,
                ↪  density=True, alpha=0.6, edgecolor='black')
164 
165             # Теоретическая PDF
166             x = np.linspace(np.min(data), np.max(data),
                ↪  1000)
167             if dist_name == "Экспоненциальное":
168                 pdf =
                    ↪  stats.expon(scale=params['scale']).pdf(x)
169             plt.plot(x, pdf, 'r-', linewidth=2)
170 
171             plt.title(f"{dist_name}\nГистограмма ({title})")
172             plt.xlabel('Значение')
173             plt.ylabel('Плотность вероятности')
174 
175     # Для дискретных распределений
176     elif dist_type == "discrete":
177         # Многоугольник вероятностей
178         unique, counts = np.unique(data, return_counts=True)
179         probs = counts / len(data)
180         plt.subplot(2, 1, 1)
181         plt.stem(unique, probs, use_line_collection=True)
182 
183         # Теоретическая PMF
184         if dist_name == "Биномиальное":
185             x = np.arange(0, params['n'] + 1)
186             pmf = stats.binom(n=params['n'],
                ↪  p=params['p']).pmf(x)
187             plt.plot(x, pmf, 'ro', markersize=5)
188 
189         plt.title(f"{dist_name}\nЭмпирическая функция
            ↪  вероятности")
190         plt.xlabel('Значение')
```

```
191         plt.ylabel('Вероятность')
192
193     # ECDF и теоретическая CDF
194     plt.subplot(2, 2, 3) if dist_type == "continuous" else
        ↪  plt.subplot(2, 1, 2)
195     x = np.sort(data)
196     y = np.arange(1, len(x) + 1) / len(x)
197     plt.step(x, y, where='post', label='eCDF')
198
199     # Теоретическая CDF
200     if dist_name == "Биномиальное":
201         x_theor = np.arange(0, params['n'] + 1)
202         cdf = stats.binom(n=params['n'],
            ↪  p=params['p']).cdf(x_theor)
203         plt.step(x_theor, cdf, 'r-', where='post',
            ↪  label='Теоретическая CDF')
204     elif dist_name == "Экспоненциальное":
205         x_theor = np.linspace(0, np.max(data), 1000)
206         cdf =
            ↪  stats.expon(scale=params['scale']).cdf(x_theor)
207         plt.plot(x_theor, cdf, 'r-', label='Теоретическая
            ↪  CDF')
208
209     plt.title('Эмпирическая и теоретическая CDF')
210     plt.xlabel('Значение')
211     plt.ylabel('Вероятность')
212     plt.legend()
213
214     # Boxplot
215     plt.subplot(2, 2, 4) if dist_type == "continuous" else
        ↪  None
216     if dist_type == "continuous":
217         plt.boxplot(data, vert=False, patch_artist=True)
218         plt.title('Boxplot с выбросами')
219         plt.xlabel('Значение')
220
221     plt.tight_layout()
222     plt.show()
223
224
225 # Добавление выбросов в данные
```

```
226 def add_outliers(data, outlier_percent, dist_type, params):
227     """
228 
229     :param data: Изначальные данные на вход
230     :param outlier_percent: Процент выбросов
231     :param dist_type: Тип распределения
232     :param params: Параметры распределения
233     :return: Массив данных с выбросами
234     """
235     n = len(data)
236     num_outliers = int(n * outlier_percent / 100)
237 
238     if dist_type == "exponential":
239         threshold = stats.expon.ppf(0.999,
            ↪  scale=params['scale'])
240         outliers = stats.expon.rvs(scale=params['scale'],
            ↪  size=num_outliers) + threshold
241     else:
242         raise ValueError(f"Unsupported distribution type:
            ↪  {dist_type}")
243 
244     # Замена случайных элементы
245     indices = np.random.choice(n, num_outliers,
        ↪  replace=False)
246     data_with_outliers = data.copy()
247     data_with_outliers[indices] = outliers
248 
249     return data_with_outliers
250 
251 # Функция для анализа устойчивости характеристик к выбросам
252 def robustness_analysis(data, dist_name, dist_type, params):
253     """
254 
255     :param data: Данные на вход
256     :param dist_name: Название распределения
257     :param dist_type: Тип распределения
258     :param params: Параметры распределения
259     """
260     # Исходные статистики
261     print("\n" + "=" * 50)
262     print("Исходные статистики:")
```

```
263     central_stats(data, dist_name)
264     variabiliyy_stats(data, dist_name)
265
266     # 5% выбросов
267     data_5perc = add_outliers(data, 5.0, dist_type, params)
268
269     print("\n" + "=" * 50)
270     print("Статистики с 5% выбросов:")
271     central_stats(data_5perc, f"{dist_name} с выбросами")
272     variabiliyy_stats(data_5perc, f"{dist_name} с
        ↪ выбросами")
273
274     # Графики изменения мер вариабельности
275     percentages = np.linspace(0, 5, 11)
276     metrics = {
277         'Размах': [],
278         'IQR': [],
279         'Дисперсия': [],
280         'Стандартное отклонение': [],
281         'CV (%)': [],
282         'MAD': []
283     }
284
285     for p in percentages:
286         data_p = add_outliers(data, float(p), dist_type,
            ↪ params)
287         metrics['Размах'].append(np.max(data_p) -
            ↪ np.min(data_p))
288         q1, q3 = np.quantile(data_p, [0.25, 0.75])
289         metrics['IQR'].append(float(q3 - q1))
290         metrics['Дисперсия'].append(float(np.var(data_p,
            ↪ ddof=1)))
291         metrics['Стандартное
            ↪ отклонение'].append(float(np.std(data_p,
            ↪ ddof=1)))
292         metrics['CV (%)'].append(float(variation(data_p) *
            ↪ 100))
293         metrics['MAD'].append(float(np.mean(np.abs(data_p -
            ↪ np.mean(data_p)))))
294
295     plt.figure(figsize=(15, 10))
```

```
296     colors = ['#1f77b4', '#ff7f0e', '#2ca02c', '#d62728',
        ↪ '#9467bd', '#8c564b']
297
298     for i, (metric, values) in enumerate(metrics.items(),
        ↪ 1):
299         plt.subplot(2, 3, i)
300         plt.plot(percentages, values,
301                  color=colors[i - 1],
302                  marker='o',
303                  linestyle='--',
304                  linewidth=2,
305                  markersize=8)
306         plt.title(metric, fontsize=12)
307         plt.xlabel('Процент выбросов (%)', fontsize=10)
308         plt.ylabel('Значение', fontsize=10)
309         plt.grid(True, alpha=0.3)
310
311     plt.tight_layout()
312     plt.suptitle(f"Изменение мер вариабельности для
        ↪ {dist_name}", y=1.02, fontsize=14)
313     plt.show()
314
315
316 data_exp = np.random.exponential(scale=2.0, size=1000)
317 robustness_analysis(data_exp, "Экспоненциальное",
    ↪ "exponential", {'scale': 2.0})
318
319
320
321
```