

# Построение выводов по данным

25 декабря 2018 г.



# Оглавление

<b>1 Краткое введение в теорию вероятностей</b>	<b>7</b>
1.1 Свойства вероятности . . . . .	8
1.2 Числовые характеристики распределений . . . . .	12
1.3 Случайные величины и их распределения . . . . .	14
1.3.1 Дискретные случайные величины . . . . .	14
1.3.2 Непрерывные случайные величины . . . . .	23
<b>2 Статистики</b>	<b>33</b>
2.1 Описательные статистики . . . . .	34
2.2 Интервальные оценки . . . . .	43
2.2.1 Построение доверительного интервала для среднего . . . . .	44
2.2.2 Построение доверительного интервала для доли .	52
2.2.3 Построение доверительных интервалов для разности двух долей для независимых выборок .	53
2.2.4 Построение доверительных интервалов для разности двух долей для связанных выборок . . .	54
2.2.5 Непараметрический доверительный интервал для медианы непрерывного распределения . .	55
2.2.6 Построение доверительных интервалов при помощи бутстрепа . . . . .	56
<b>3 Проверка статистических гипотез</b>	<b>57</b>
3.1 Методика проверки статистических гипотез . . . . .	57

3.2	Ошибки первого и второго рода . . . . .	60
3.3	Типы статистических критериев . . . . .	61
3.4	Связь между проверкой гипотез и доверительными интервалами . . . . .	62
<b>4</b>	<b>Параметрические критерии</b>	<b>63</b>
4.1	Нормальность распределения . . . . .	63
4.2	Критерии предполагающие нормальное распределение . . . . .	66
4.2.1	z-критерий (одновыборочный) . . . . .	66
4.2.2	z-критерий (двухвыборочный) . . . . .	67
4.2.3	Одновыборочный критерий Стьюдента . . . . .	67
4.2.4	Двухвыборочный критерий Стьюдента для независимых выборок . . . . .	68
4.2.5	Двухвыборочный критерий Стьюдента для связанных выборок . . . . .	69
4.2.6	F-критерий Фишера для сравнения двух дисперсий . . . . .	70
4.2.7	Задачи . . . . .	71
4.3	Критерии для долей . . . . .	80
4.3.1	z-критерий для доли . . . . .	80
4.3.2	z-критерий для доли двух независимых выборок . . . . .	80
4.3.3	z-критерий для доли двух связанных выборок . . . . .	81
4.3.4	Задачи . . . . .	82
<b>5</b>	<b>Непараметрические критерии</b>	<b>91</b>
5.1	Критерий знаков . . . . .	91
5.1.1	Критерий знаков для одной выборки . . . . .	92
5.1.2	Критерий знаков для связанных выборок . . . . .	93
5.2	Ранговые критерии . . . . .	94
5.2.1	Критерий ранговых знаков Уилкоксона . . . . .	94
5.2.2	Критерий ранговых знаков для независимых выборок (Критерий Манна - Уитни - Уилкоксона) . . . . .	95

5.2.3	Критерий ранговых знаков для связанных выборок . . . . .	96
5.3	Перестановочные критерии . . . . .	97
5.3.1	Одновыборочный перестановочный критерий . . . . .	97
5.3.2	Двухвыборочный перестановочный критерий для независимых выборок . . . . .	97
5.3.3	Двухвыборочный перестановочный критерий для связанных выборок . . . . .	97
5.4	Задачи . . . . .	99
<b>6</b>	<b>Анализ зависимостей</b>	<b>111</b>
6.1	Непрерывные случайные величины . . . . .	111
6.1.1	Корреляция Пирсона . . . . .	113
6.1.2	Корреляция Спирмена . . . . .	115
6.1.3	Корреляция Кендалла . . . . .	117
6.2	Категориальные признаки . . . . .	118
6.2.1	Критерий $\chi^2$ . . . . .	118
6.2.2	Точный критерий Фишера . . . . .	119
6.2.3	Корреляция Мэттьюса . . . . .	119
6.2.4	Коэффициент V Крамера . . . . .	120
6.3	Пары переменных разных видов . . . . .	120
<b>7</b>	<b>Регрессия</b>	<b>121</b>
7.1	Постановка задачи . . . . .	121
7.2	Категориальные признаки . . . . .	125
7.3	Отбор переменных в модель . . . . .	126
7.4	Задачи . . . . .	129
<b>8</b>	<b>Задача кредитного scoringа</b>	<b>143</b>



# Глава 1

## Краткое введение в теорию вероятностей

Рассмотрим процесс подбрасывания кубика, в котором точное предсказание результата не представляется возможным. Можно отойти от построения сложной физической модели подбрасывания и перейти к рассмотрению некоторого «черного ящика». Он по неизвестным законам генерирует случайные события, соответствующие числам, выпадающим на кубике. В математике такие «черные ящики» называются **случайными величинами**, а генерируемые события — **реализациями случайной величины**. Набор реализаций случайной величины называется **выборкой**.

Если проводить эксперимент со случайной величиной бесконечно, то каждому событию можно будет поставить в соответствие его **вероятность** — долю испытаний, завершившихся наступлением события (определение нестрогое). Вероятность не может быть измерена на практике в силу данного определения.

**Теория вероятностей** изучает модели случайных величин и свойства этих моделей. **Статистика и анализ данных** пытаются по свойствам конечных выборок определить свойства случайной величины, чтобы понять как она будет вести себя в будущем. Осуществить такой переход позволяет **закон больших чисел**: на большой выборке частота события хорошо приближает его вероятность.

## 1.1 Свойства вероятности

Основные свойства вероятности:

- $0 \leq P(A) \leq 1$ . Вероятность любого события лежит на отрезке от нуля до единицы.
- $P(\emptyset) = 0$ . Событие, вероятность которого равна нулю, называется невозможным.
- $P(\bar{A}) + P(A) = 1$ . Для события  $A$  всегда можно определить событие «не  $A$ », которое соответствует событию « $A$  не произошло». Вероятности таких событий в сумме дают единицу.

Для пары событий возможны следующие отношения:

- **Вложенность**  $A \subseteq B$ . Например, эксперимент - стрельба по мишени. Событие  $A$  - попадание в десятку, событие  $B$  - набор более 5 очков (рисунок 1.1).

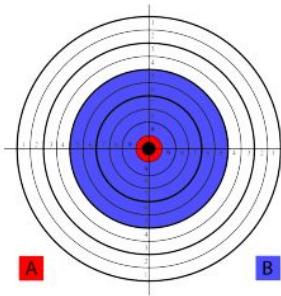


Рис. 1.1: Вложенность событий

$$A \subseteq B \Rightarrow P(A) \leq P(B)$$

- Сумма событий и произведение событий. Пусть событие  $A$  — это попадание в синюю область на мишени, событие  $B$  — в зеленую. Тогда **произведением событий  $A$  и  $B$**  называется событие  $AB$  — произошли оба события одновременно (рисунок 1.2), а **суммой событий  $A$  и  $B$**  называется событие  $A + B$  — произошло хотя бы одно из двух событий (рисунок

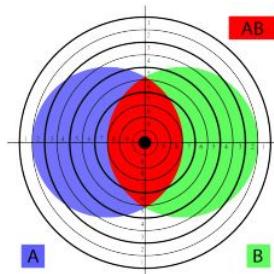


Рис. 1.2: Произведение двух событий

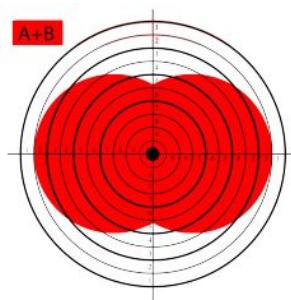


Рис. 1.3: Сумма двух событий

1.3). Определения суммы и произведения событий могут быть применены к произвольному числу событий.

Вероятности суммы и произведения событий связаны следующим отношением:

$$P(A + B) = P(A) + P(B) - P(AB)$$

- **Дополнением  $B \setminus A$ .** события  $A$  до  $B$  называется событие, состоящее в том, что произошло событие  $B$ , но не произошло событие  $A$  (рисунок 1.4).
- **Независимость событий.** Пусть событие  $A$  - попадание в нижнюю часть мишени, событие  $B$  - попадание в правую часть мишени (рисунок 1.5).

События  $A$  и  $B$  являются независимыми, если вероятность их произведения равна произведению их вероятностей:

$$P(AB) = P(A)P(B)$$

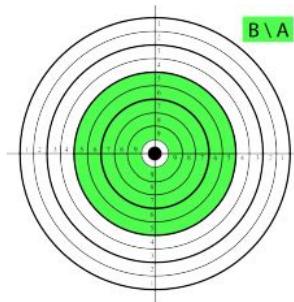


Рис. 1.4: Дополнение

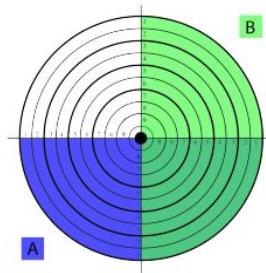


Рис. 1.5: Независимость событий

Для данного случая  $P(A) = 0.5$ ,  $P(B) = 0.5$ ,  $P(AB) = 0.25$ . Следовательно, мы говорим, что события  $A$  и  $B$  являются независимыми.

Пусть событие  $A$  на примере с мишенью — это попадание в «девятку», событие  $B$  — попадание в любое место мишени. Если известно, что событие  $B$  произошло, то вероятность события  $A$  повышается. Условная вероятность события  $A$  при условии, что произошло событие  $B$ , определяется следующим образом:

$$P(A|B) = \frac{P(AB)}{P(B)}$$

Формула полной вероятности:

$$P(A) = P(A|B)P(B) + P(A|\bar{B})P(\bar{B})$$

Условные вероятности двух событий связаны **формулой Байеса**:

$$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)}$$

Условие независимости двух событий  $A$  и  $B$  может быть записано через условную вероятность:

$$P(A|B) = P(A)$$

## 1.2 Числовые характеристики распределений

**Математическое ожидание** - среднее значение случайной величины при стремлении количества её измерений к бесконечности. Пусть дискретная случайная величина принимает значения  $x_1, x_2, \dots, x_n$  с соответствующими вероятностями  $p_1, p_2, \dots, p_n$ :

$$\mathbb{E}X = \sum_{i=1}^n x_i p_i$$

Математическое ожидание для непрерывной случайной величины:

$$\mathbb{E}X = \int x f(x) dx$$

**Дисперсия** - мера разброса значений случайной величины относительно её математического ожидания:

$$\mathbb{D}X = \mathbb{E}((X - \mathbb{E}X)^2)$$

**Квантилем порядка  $\alpha$**  называется такая величина  $X_\alpha$ , что:

$$P(X \leq X_\alpha) \geq \alpha, \quad P(X \geq X_\alpha) \geq 1 - \alpha$$

**Р-м процентилем** называют квантиль порядка  $\alpha = \frac{p}{100}$ . Понятия квантиль и процентиль взаимозаменяемы.

**Медианой** называется такое значение  $medX$  для которого:

$$P(X \leq medX) \geq 0.5, \quad P(X \geq medX) \geq 0.5$$

Альтернативное определение медианы - квантиль порядка 0.5:

$$medX = X_{0.5}$$

**Интерквартильный размах** - разность между 75-м и 25-м процентилем:

$$IQR = X_{0.75} - X_{0.25}$$

**Модой** называется наиболее вероятное значение случайной величины:

$$\text{mode}X = \arg \max_x f(x)$$

**Коэффициент асимметрии:**

$$\gamma_1 = \mathbb{E} \left( \frac{X - \mathbb{E}X}{\sqrt{\mathbb{D}X}} \right)^3$$

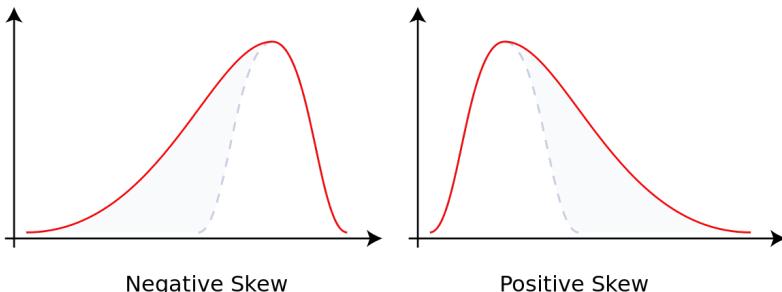


Рис. 1.6: Пример распределений с положительным и отрицательным коэффициентом асимметрии

Если распределение симметрично, следовательно, коэффициент асимметрии равен нулю. Однако обратное неверно.

**Коэффициент эксцесса** - числовая величина, характеризующая степень остроты пика распределения случайной величины:

$$\gamma_2 = \frac{\mathbb{E}(X - \mathbb{E}X)^4}{(\mathbb{D}X)^2} - 3$$

## 1.3 Случайные величины и их распределения

**Случайной величиной** называется величина, которая в результате эксперимента может принять некоторое значение, причём заранее неизвестно какое.

### 1.3.1 Дискретные случайные величины

Случайная величина  $X$  называется **дискретной**, если она принимает значение из счетного множества  $A = \{a_1, a_2, \dots, a_n\}$  с соответствующими вероятностями  $p_1, p_2, \dots, p_n$ , причём

$$p_i \geq 0 \quad \sum_{i=1}^{\infty} p_i = 1 \quad P(X = a_i) = p_i$$

Примеры дискретных случайных величин:

- выпадение «орла» или «решки» при подбрасывании монеты (1 - «орёл», 0 - «решка»)
- количество очков на игральном кубике при броске (1, 2, 3, 4, 5, 6)
- количество попаданий в мишень из 5 выстрелов (1, 2, 3, 4, 5)
- невозврат заемщиком кредита или нет (0 - возврат, 1 - невозврат)

#### Дискретное равномерное распределение

**Дискретное равномерное распределение** — распределение дискретной случайной величины, которая принимает конечное число значений с равными вероятностями.

Примеры:

- при подбрасывании монеты случайная величина принимает значение 1, если выпал «орёл», или 0, если выпала «решка». Вероятность выпадения одного из двух значений равна  $1/2$ ,

которая одинакова для обоих значений, поэтому случайная величина имеет дискретное равномерное распределение.

- при бросании игральной кости случайная величина — число точек на грани, которая принимает одно из 6-и возможных значений:  $\{1, 2, 3, 4, 5, 6\}$ . Вероятность выпадения стороны со значением 1 равна  $1/6$ , одинакова для каждой из сторон, поэтому случайная величина имеет дискретное равномерное распределение.

### Распределение Бернулли

Одной из наиболее часто встречающихся случайных величин является **дискретная случайная величина с двумя исходами** (например, подбрасывание монеты). Пусть выпадение «орла» (успех) - 1, «решки» (неудача) - 0. Если вероятность успеха равна  $p$ .

$$P(X = 1) = p$$

то вероятность неудачи:

$$P(X = 0) = 1 - p$$

Именно так устроена **бернуlliевская случайная величина**. Математическое ожидание и дисперсия:

$$\mathbb{E}X = p \quad \mathbb{D}X = pq$$

Факт того, что случайная величина  $X$  имеет распределение Бернулли, может быть записан следующим образом:

$$X \sim Ber(p)$$

На языке программирования Python выполнить генерацию выборки размера *size*, распределённой по закону Бернулли, можно выполнить способом, представленным на рисунке 1.7.

### Биномиальное распределение

Другим примером является **сумма независимых бинарных случайных величин**. Например, мы выполняем  $n$  бросков в бас-

```
from scipy.stats import bernoulli

# для генерации значений из некоторого распределения
# используется функция rvs - random value sample
size = 10
sample = bernoulli(p=0.5).rvs(size)
print(sample)

[1 0 0 1 1 0 1 0]
```

Рис. 1.7: Генерация выборки, распределенной по закону Бернулли

кетбольное кольцо. Броски являются независимыми событиями (если игрок забросил в прошлый раз, то это никак не влияет на его следующий бросок). Вероятность попадания в кольцо равна  $p \in [0, 1]$ . Случайная величина  $X$  - суммарное количество попаданий, которая будет являться **биномиально распределенной случайной величиной**:

$$X \sim Bin(n, p)$$

Вероятность из  $n$  бросков выполнить  $k$  попаданий:

$$P_n(X = k) = C_n^k p^k (1 - p)^{n-k}$$

Биномиальный коэффициент:

$$C_n^k = \frac{n!}{k!(n - k)!}$$

Если значения  $n$  велики, то непосредственное вычисление вероятностей событий, связанных с данной случайной величиной, технически затруднительно. В этих случаях используют приближения биномиального распределения распределением Пуассона или нормальным распределением (приближение Муавра-Лапласа):

$$P_n(X = k) \approx \Phi\left(\frac{k - np}{\sqrt{npq}}\right)$$

Данное приближение используется если  $n > 20$  и значение  $p$  не слишком близко к нулю или единице.

Математическое ожидание и дисперсия:

$$\mathbb{E}X = np \quad \mathbb{D}X = npq = np(1 - p)$$

Пример генерации биномиальной случайной величины случайной величины представлен на рисунке 1.8.

Функции вероятности биномиального распределения и соответствующие функции распределения представлены на рисунке 1.9.

```
from scipy.stats import binom

# для генерации значений из некоторого распределения
# используется функция rvs - random value sample
size = 10
sample = binom(n=10, p=0.5).rvs(size)
print(sample)

[5 7 4 9 5 3 7 4 3 5]
```

Рис. 1.8: Генерация выборки, распределенной по биномиальному закону

## Геометрическое распределение

**Геометрическое распределение** — распределение дискретной случайной величины, равной количеству испытаний случайного эксперимента до наблюдения первого «успеха».

Математическое ожидание и дисперсия:

$$X \sim \text{Geom}(p) \quad \mathbb{E}X = \frac{q}{p} \quad \mathbb{D}X = \frac{q}{p^2}$$

Функции вероятности геометрического распределения и соответствующие функции распределения представлены на рисунке 1.10.

## Отрицательное биномиальное распределение

**Отрицательное биномиальное распределение (распределение Паскаля)** — распределение дискретной случайной величины, равной количеству произошедших неудач, в последовательности испытаний Бернулли с вероятностью успеха  $p$ , проводимой до  $r$ -го успеха.

Математическое ожидание и дисперсия:

$$X \sim \text{NB}(r, p) \quad \mathbb{E}X = \frac{rq}{p} \quad \mathbb{D}X = \frac{rq}{p^2}$$

Функции вероятности распределения Паскаля представлены на рисунке 1.11 и соответствующие им функции распределения представлены на рисунке 1.12.

### Распределение Пуассона

Еще одним классом дискретных случайных величин являются **счётчики**. Пусть  $X$  — это число использований слова в тексте. Вероятность того, что  $X$  равно  $k$ , можно описать распределением Пуассона:

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}, \quad \lambda > 0, \quad k = 0, 1, 2\dots$$

$$X \sim Pois(\lambda)$$

Распределением Пуассона описывают, например, число автобусов, которые проезжают за час мимо автобусной остановки, или число радиоактивных распадов, которое улавливает счетчик Гейгера.

Распределение Пуассона моделирует случайную величину, представляющую собой число событий, произошедших за фиксированное время, при условии, что данные события происходят с некоторой фиксированной средней интенсивностью и независимо друг от друга.

Математическое ожидание и дисперсия:

$$\mathbb{E}X = \mathbb{D}X = \lambda$$

```

import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import binom

x = np.arange(21)

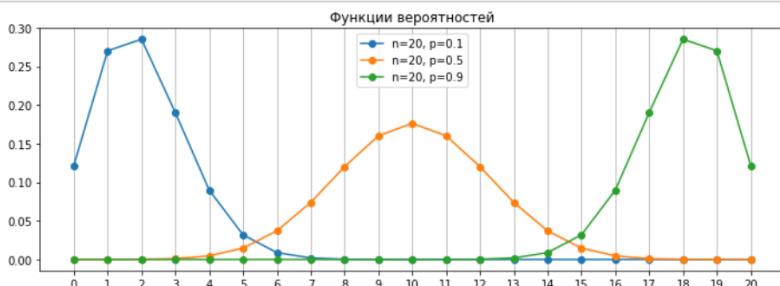
fig = plt.figure(figsize=(12, 4))

for p in [0.1, 0.5, 0.9]:
    plt.plot(x, binom(n=20, p=p).pmf(x), marker="o",
              label="n=20, p=" + str(p))

plt.title("Функции вероятностей")
plt.xticks(x)
plt.legend()
plt.grid(axis="x")

plt.show()

```



```

fig = plt.figure(figsize=(12, 4))

for p in [0.1, 0.5, 0.9]:
    plt.plot(x, binom(n=20, p=p).cdf(x), marker="o",
              label="n=20, p=" + str(p))

plt.title("Функции распределения")
plt.xticks(x)
plt.legend()
plt.grid(axis="x")

plt.show()

```

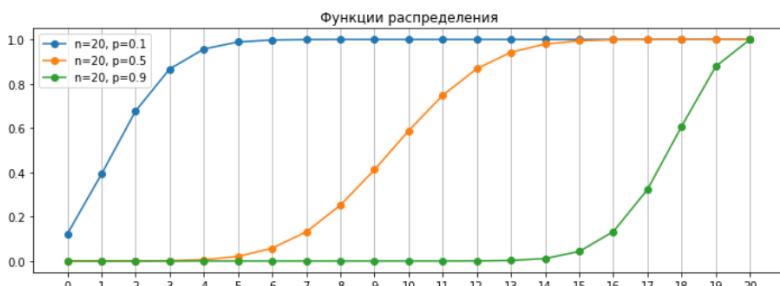


Рис. 1.9: Функции вероятности биномиального распределения и соответствующие функции распределения

```

import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import geom

x = np.arange(10)

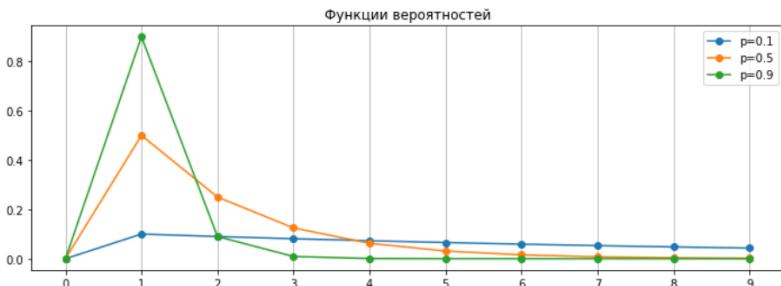
fig = plt.figure(figsize=(12, 4))

for p in [0.1, 0.5, 0.9]:
    plt.plot(x, geom(p=p).pmf(x), marker="o", label="p=" + str(p))

plt.title("Функции вероятностей")
plt.xticks(x)
plt.legend()
plt.grid(axis="x")

plt.show()

```



```

fig = plt.figure(figsize=(12, 4))

for p in [0.1, 0.5, 0.9]:
    plt.plot(x, geom(p=p).cdf(x), marker="o", label="p=" + str(p))

plt.title("Функция распределения")
plt.xticks(x)
plt.legend()
plt.grid(axis="x")

plt.show()

```

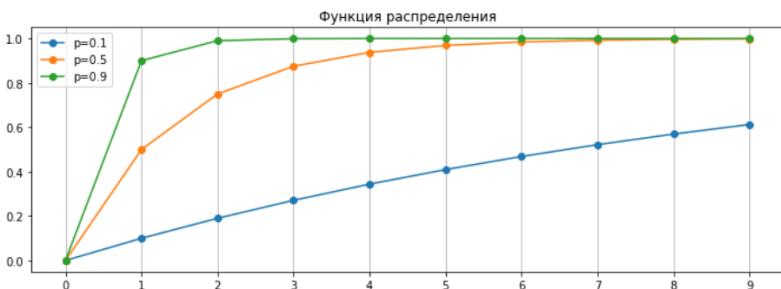


Рис. 1.10: Функции вероятности геометрического распределения и соответствующие функции распределения

```

import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import nbinom

x = np.arange(25)

fig, axes = plt.subplots(3, 3, figsize=(16, 12))
plt.suptitle("Функция вероятности", fontsize=20)

for i, p in enumerate([0.1, 0.5, 0.9]):
    for j, r in enumerate([5, 10, 15]):
        axes[i, j].plot(x, nbinom(n=r, p=p).pmf(x), marker="o",
                          label="p=" + str(p) + ", r=" + str(r))
        axes[i, j].legend()
        axes[i, j].grid(axis="x")

plt.show()

```

Функция вероятности

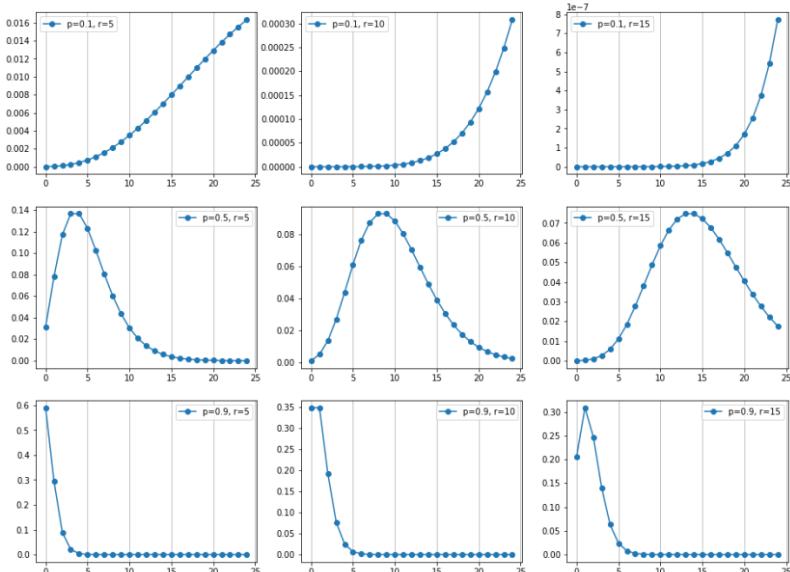


Рис. 1.11: Функции вероятности отрицательного биномиального распределения

```

x = np.arange(25)

fig, axes = plt.subplots(3, 3, figsize=(16, 12))
plt.suptitle("Функция распределения", fontsize=20)

for i, p in enumerate([0.1, 0.5, 0.9]):
    for j, r in enumerate([5, 10, 15]):
        axes[i, j].plot(x, nbinom(n=r, p=p).cdf(x), marker="o",
                          label="p=" + str(p) + ", r=" + str(r))
        axes[i, j].legend()
        axes[i, j].grid(axis="x")

plt.show()

```

Функция распределения

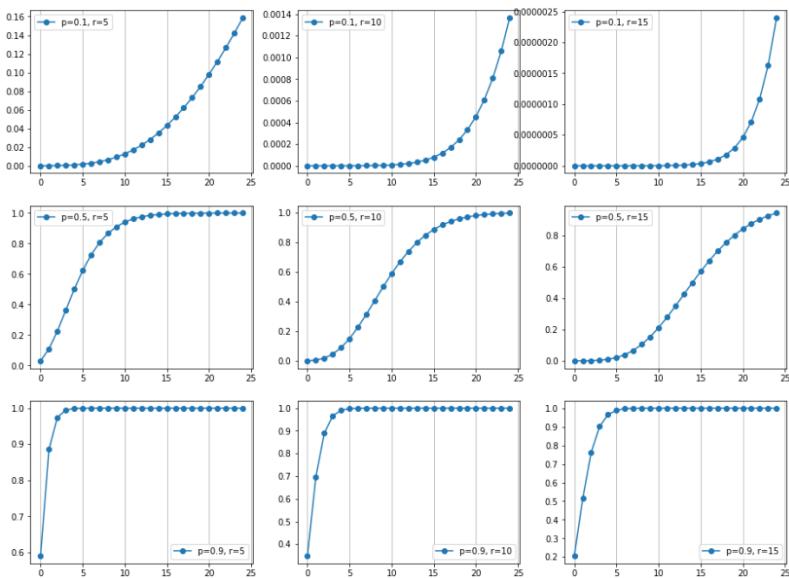


Рис. 1.12: Функции распределения отрицательного биномиального распределения

### 1.3.2 Непрерывные случайные величины

**Непрерывная случайная величина** - случайная величина, которая может принимать любое действительное значение из некоторого промежутка ненулевой длины.

Непрерывные случайные величины можно задать при помощи **функции распределения**:

$$F(x) = P(X \leq x)$$

Примеры функций нормального распределения представлены на рисунке (рисунок 1.13).

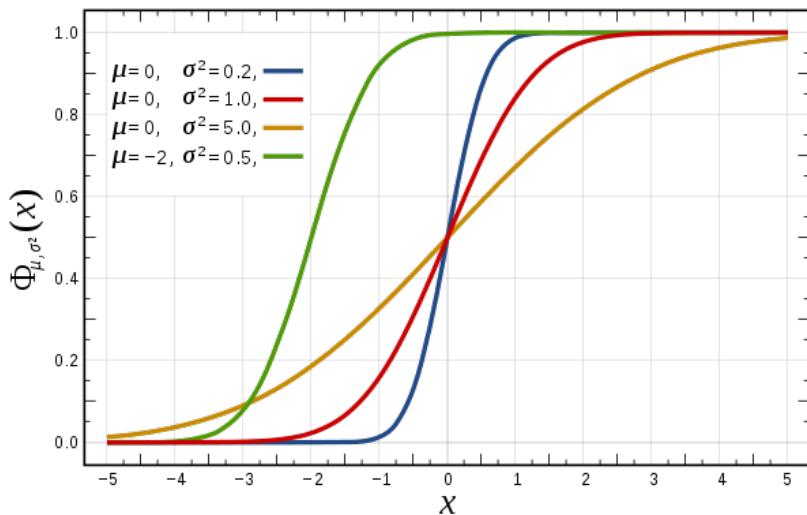


Рис. 1.13: Функция нормального распределения при разных значениях среднего и среднеквадратичного отклонения

Вероятность попадания случайной величины в заданный интервал может быть найдена из определения функции распределения:

$$P(a \leq X \leq b) = F(b) - F(a)$$

Свойства функции распределения:

- $0 \leq F(x) \leq 1$
- $\lim_{x \rightarrow -\infty} F(x) = 0, \lim_{x \rightarrow +\infty} F(x) = 1$
- $F(x_1) \leq F(x_2)$  если  $x_1 \leq x_2$

Другим способом задания вероятностного закона является **функция плотности распределения**.

Пусть имеется некоторая функция  $F(x)$ , которую мы предположим непрерывной и дифференцируемой. Вычислим вероятность попадания случайной величины на участок от  $x$  до  $x + \Delta x$ :

$$P(x \leq X \leq x + \Delta x) = F(x + \Delta x) - F(x)$$

Рассмотрим отношение данной вероятности к длине участка. В пределе получим производную от функции распределения:

$$\lim_{\Delta x \rightarrow 0} \frac{F(x + \Delta x) - F(x)}{\Delta x} = F'(x) = f(x)$$

Функция  $f(x)$  называется функцией плотности распределения случайной величины. Её основные свойства:

- Функция плотности распределения - неотрицательная функция:

$$f(x) \geq 0$$

- Интеграл в бесконечных пределах равен единице:

$$\int_{-\infty}^{\infty} f(x) dx = 1$$

Исходя из определения:

$$P(a \leq X \leq b) = \int_a^b f(x) dx = F(b) - F(a)$$

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

# генерация 100 равномерно распределенных значений в интервале [-3, 3]
x = np.linspace(-3, 3, 100)

# настройка параметров распределения
mu, sigma = 0, 1
dist = norm(loc=mu, scale=sigma)

fig = plt.figure(figsize=(15, 6))

# заголовок графика
plt.title(
    "Стандартное нормальное распределение\n"
    " $\mu =$ " + str(mu) + ",  $\sigma =$ " + str(sigma),
    fontsize=16
)

# cdf - функция распределения
plt.plot(x, dist.cdf(x), label="cdf", linewidth=3)
# pdf - функция плотности распределения
plt.plot(x, dist.pdf(x), label="pdf", linewidth=3)

# подпись оси X
plt.xlabel("X", fontsize=14)
# вывод легенды
plt.legend(fontsize=14)
# сетка
plt.grid()

plt.show()

```

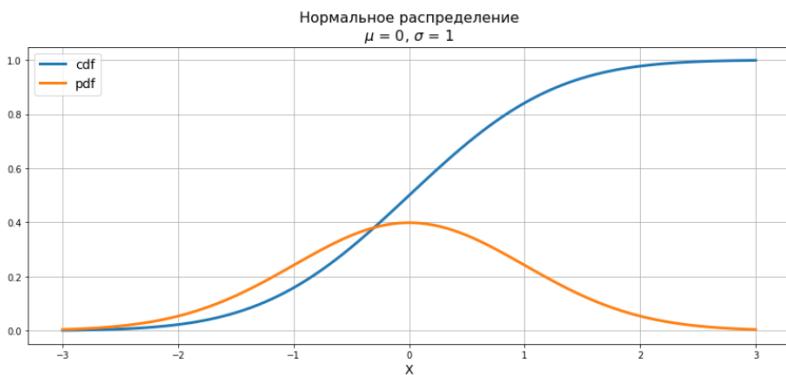


Рис. 1.14: Стандартное нормальное распределение

Пример построения функции стандартного нормального распределения и его плотности представлены на рисунке 1.14.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import uniform

a, b = 1, 3
dist = uniform(loc=a, scale=b-a)
x = np.linspace(a-0.5, b+0.5, 100)

fig = plt.figure(figsize=(15, 6))

plt.title(
    "Равномерное распределение\n"
    "a = " + str(a) + ", b = " + str(b),
    fontsize=16
)

plt.plot(x, dist.cdf(x), label="cdf", linewidth=3)
plt.plot(x, dist.pdf(x), label="pdf", linewidth=3)

plt.xlabel("X", fontsize=14)
plt.legend(fontsize=14)
plt.grid()

plt.show()
```

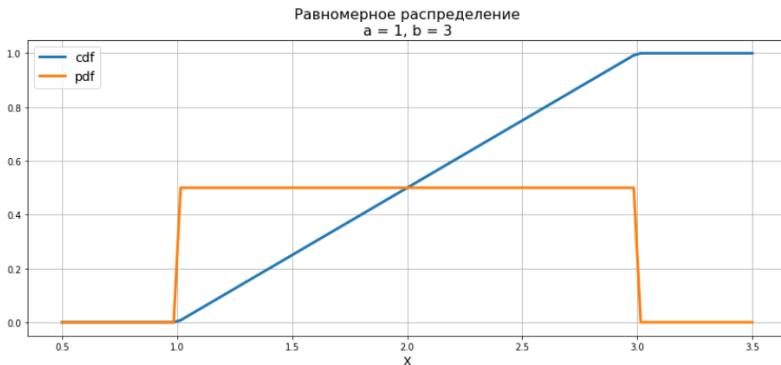


Рис. 1.15: Равномерное распределение

### Равномерная случайная величина

Пусть  $X$  — время ожидания зеленого света на светофоре. Если на нём не установлен счётчик, то нельзя угадать время ожидания зелёного света, которое может быть любым числом в определенном промежутке с нижней границей в нуле. Именно так устроено **равномерное распределение**: случайная величина на отрезке  $[a, b]$  принимает любое значение с одинаковой вероятностью.

$$X \sim U(a, b)$$

$$f(x) = \begin{cases} \frac{1}{b-a} & x \in [a, b] \\ 0 & x \notin [a, b] \end{cases}$$

### Нормальная случайная величина

Другим примером непрерывной случайной величины является **нормальная случайная величина**. Например, время прихода на работу варьируется — иногда мы приходим раньше, иногда опаздываем. Относительно часто мы приходим вовремя. Точное время прихода на работу  $X$  представляет собой результат взаимодействия большого количества слабо зависимых случайных факторов. Именно такие величины хорошо моделируются нормальным (Гауссовым) распределением:

$$X \sim N(\mu, \sigma^2) \quad f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

**Стандартным нормальным распределением** называется нормальное распределение с математическим ожиданием  $\mu = 0$  и отклонением  $\sigma = 1$ .

$$F(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$$

Функция плотности нормального распределения представлены на купюре в 10 немецких марок 1993 года (рисунок 1.16).



Рис. 1.16: Функция плотности нормального распределения

### Распределение $\chi^2$

Распределение  $\chi^2$  (хи-квадрат) с  $df$  степенями свободы — это распределение суммы квадратов  $df$  независимых стандартных нормальных случайных величин:

$$X_1, X_2, \dots, X_{df} \sim N(0, 1), \quad X = \sum_{i=1}^{df} X_i^2 \sim \chi_{df}^2$$

Функции распределения  $\chi^2$  и функции плотности распределения  $\chi^2$  для разных значений  $df$  представлены на рисунке 1.17.

### Распределение Фишера

Имеются две случайные величины  $X_1$  и  $X_2$ , принадлежащие распределению  $\chi^2$ :

$$X_1 \sim \chi_{d_1}^2 \quad X_2 \sim \chi_{d_2}^2$$

Распределение случайной величины

$$X = \frac{X_1/d_1}{X_2/d_2} \sim F(d_1, d_2)$$

является распределением Фишера. Функции распределения и функции плотности распределения Фишера, представлены на рисунке 1.18.

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2

fig, ax = plt.subplots(nrows=2, ncols=1, figsize=(15, 12))

ax[0].set_title("Распределение  $\chi^2$ ", fontsize=16)

x = np.linspace(0, 5, 100)

for k in [1, 2, 3, 4, 6, 9]:
    dist = chi2(df = k)
    ax[0].plot(x, dist.cdf(x), label="df = " + str(k), linewidth=3)
    ax[1].plot(x, dist.pdf(x), label="df = " + str(k), linewidth=3)

ax[1].set_xlim(0, 1)

for i in range(2):
    ax[i].grid()
    ax[i].legend(fontsize=14)

plt.xlabel("X", fontsize=14)

plt.show()

```

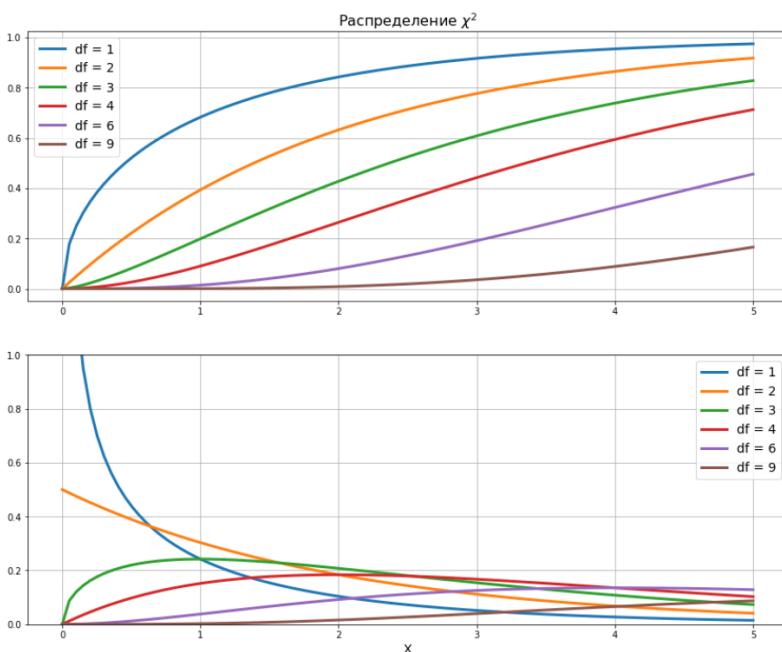


Рис. 1.17: Функция распределения  $\chi^2$  для различных  $df$

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import f

fig, ax = plt.subplots(nrows=2, ncols=1, figsize=(15, 12))

ax[0].set_title("Распределение Фишера", fontsize=16)

x = np.linspace(0, 3, 100)
dfs = [2, 9]

for d1 in dfs:
    for d2 in dfs:
        dist = f(d1, d2)
        ax[0].plot(x, dist.cdf(x),
                    label="d1, d2 = " + str(d1) + ", " + str(d2), linewidth=3)
        ax[1].plot(x, dist.pdf(x),
                    label="d1, d2 = " + str(d1) + ", " + str(d2), linewidth=3)

for i in range(2):
    ax[i].grid()
    ax[i].legend(fontsize=14)

plt.xlabel("X", fontsize=14)

plt.show()

```

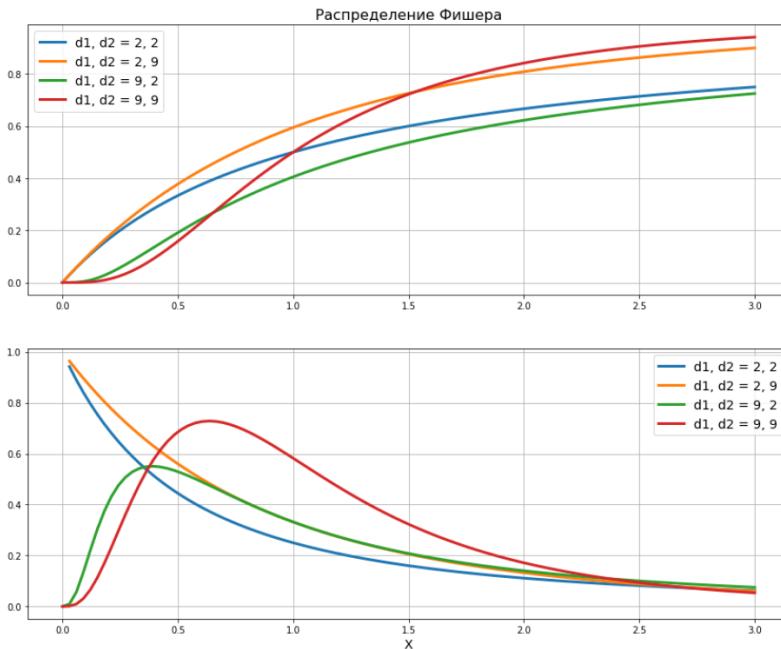


Рис. 1.18: Функция плотности распределения Фишера

### Распределение Стьюдента

Имеется две случайные величины:

$$X_1 \sim N(0, 1), X_2 \sim \chi^2_v$$

Распределение Стьюдента:

$$X = \frac{X_1}{\sqrt{X_2/\nu}} \sim St(v)$$

названо в честь Уильяма Сили Госсета, который первым опубликовал работы, посвящённые этому распределению, под псевдонимом «Стьюдент». Госсету пришлось скрывать свою личность при публикации из-за того, что ранее другой исследователь, работавший на Гиннесс (предприятие пищевой промышленности), опубликовал в своих материалах сведения, составлявшие коммерческую тайну компании, после чего Гиннесс запретил своим работникам публикацию любых материалов, независимо от содержащейся в них информации. Распределение похоже на нормальное распределение, но имеет более тяжелые хвосты (рисунок 1.19).

Данное распределение центрировано в нуле, при больших значениях  $v$  начинает становиться похожим на стандартное нормальное распределение.

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import t

fig, ax = plt.subplots(nrows=2, ncols=1, figsize=(15, 14))

ax[0].set_title("Распределение Стьюдента", fontsize=16)

x = np.linspace(-3, 3, 100)
vs = [1, 2, 5, 30]

for v in vs:
    dist = t(df=v)
    ax[0].plot(x, dist.cdf(x), label="v = " + str(v), linewidth=3)
    ax[1].plot(x, dist.pdf(x), label="v = " + str(v), linewidth=3)

for i in range(2):
    ax[i].grid()
    ax[i].legend(fontsize=14)

plt.xlabel("X", fontsize=14)

plt.show()

```

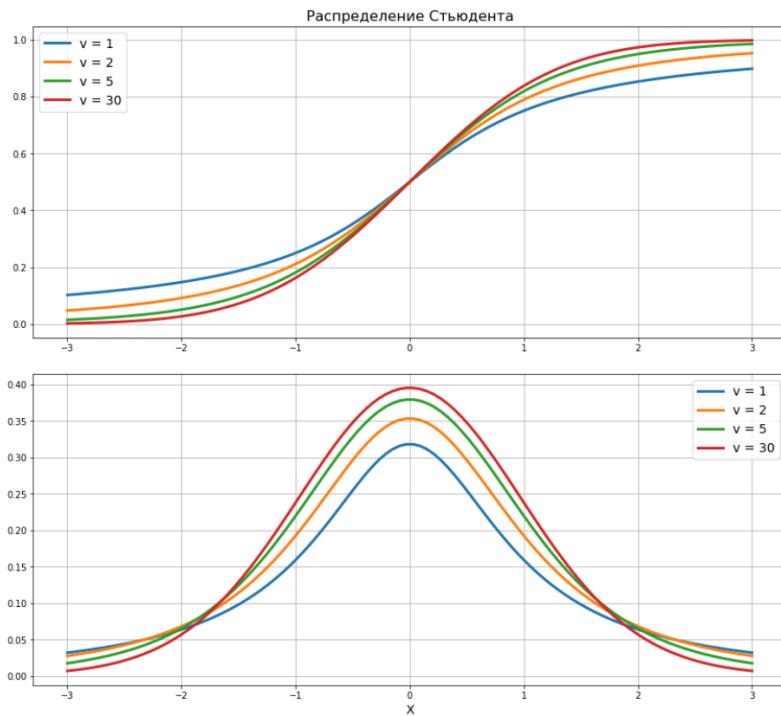


Рис. 1.19: Функция плотности t-распределения Стьюдента

## Глава 2

# Статистики

**Генеральная совокупность** — множество всех объектов, относительно которых делают выводы в рамках исследования некоторой проблемы. Очень часто у нас нет информации о каждом объекте из генеральной совокупности. В исследованиях приходится оперировать выборками.

**Выборка** — некоторое подмножество генеральной совокупности.

$$X^n = (X_1, \dots, X_n)$$

Число  $n$  называют **объемом выборки**.

Для того, чтобы мы могли обобщать результаты исследования на генеральную совокупность, выборка должна обладать свойством репрезентативности. Репрезентативность выборки нередко зависит от способа её формирования. Основные способы:

- Простая случайная выборка - случайным образом отбираются элементы генеральной совокупности.
- Стратифицированная выборка - генеральная совокупность разбивается на подгруппы по некоторым признакам (например, пол) и из каждой группы объекты выбираются случайно.
- Групповая выборка - генеральная совокупность разделяется на несколько групп, которые схожи между собой. Выбирается

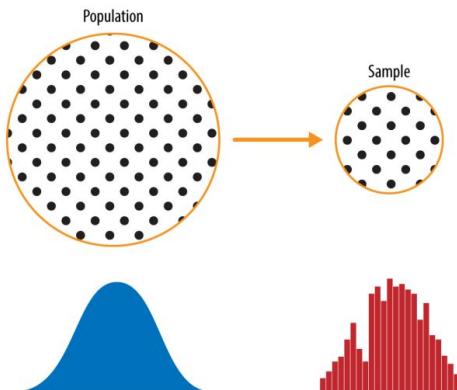


Рис. 2.1: Генеральная совокупность / выборка

$n$  групп. Из каждой группы случайным образом выбирается ряд объектов.

**Статистикой**  $T(X^n)$  называется любая измеримая функция выборки.

## 2.1 Описательные статистики

**Описательная статистика** - некоторая статистика, которая описывает набор данных.

Описательные статистики можно разделить на две группы:

- Меры центральной тенденции неким образом описывают среднее значение (арифметическое среднее, винзоризованное среднее, геометрическое среднее, медиана, мода, усеченное среднее и т. д.)
- Меры изменчивости - описывают меру разброса значений вокруг среднего (дисперсия, стандартное отклонение, коэффициент вариации, асимметрия и т. д.).

**Выборочное среднее** - среднее арифметическое по выборке:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

Очевидно, что выборочное среднее подвержено выбросам (экстремально большим/ маленьеньким значениям).

Для решения проблемы экстремальных значений существует несколько преобразований исходной выборки перед тем, как вычислять среднее арифметическое (усеченное среднее, винзоризованное среднее).

Введём определение вариационного ряда. **Вариационный ряд** — упорядоченная по величине последовательность выборочных значений наблюдаемой случайной величины:

$$X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$$

**Усеченное среднее** (среднее вычисляется после отбрасывания первых  $p$  и последних  $p$  элементов вариационного ряда):

$$\bar{X} = \frac{\sum_{i=p+1}^{n-p} X_{(i)}}{n - 2p}$$

**Винзоризованное среднее.** Среднее арифметическое значение вычисляется после замены  $k\%$  первых элементов вариационного ряда на следующий элемент вариационного ряда,  $k\%$  последних элементов вариационного ряда заменяются на предыдущий элемент вариационного ряда.

**Выборочная медиана** - значение, которое делит упорядоченное множество данных пополам (центральный элемент вариационного ряда):

$$m = \begin{cases} X_{(k+1)} & n = 2k + 1 \\ \frac{X_{(k)} + X_{(k+1)}}{2} & n = 2k \end{cases}$$

Выбор среднего значения может преукрасить исследование. В книге Дарелла Хаффа «Как врать при помощи статистики» использовано изображение к вопросу о средней заработной плате (рисунок 2.3 на странице 38).

**Выборочная мода** - значение, которое встречается в выборке максимальное количество раз.

**Выборочная дисперсия:**

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

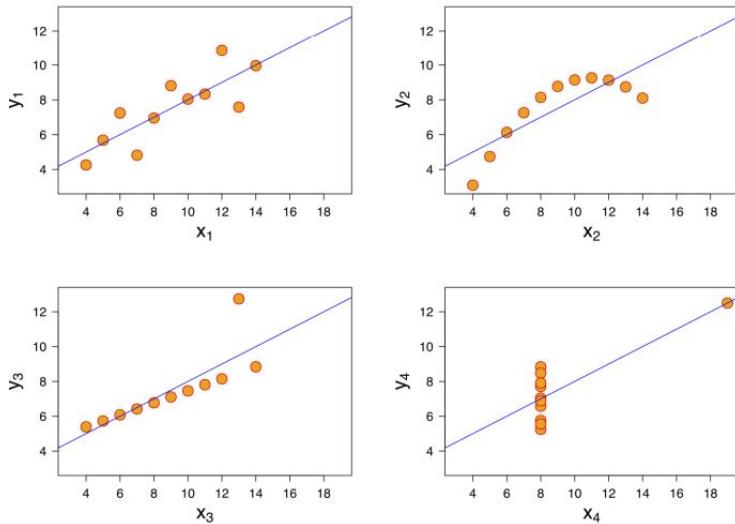


Рис. 2.2: Квартет Энскомба

Выборочный  $\alpha$ -квантиль:

$$X([n\alpha])$$

Выборочный межквартильный размах:

$$IQR = X_{([0.75n])} - X_{([0.25n])}$$

Выборочный коэффициент асимметрии:

$$g_1 = \frac{\sqrt{n} \sum_{i=1}^n (X_i - \bar{X})^3}{\left( \sum_{i=1}^n (X_i - \bar{X})^2 \right)^{\frac{3}{2}}}$$

Выборочный коэффициент эксцесса:

$$g_2 = \frac{n \sum_{i=1}^n (X_i - \bar{X})^4}{\left( \sum_{i=1}^n (X_i - \bar{X})^2 \right)^2} - 3$$

Если распределение симметрично, унимодально (имеет одну моду) и не имеет заметных выбросов, то можно использовать любую меру центральной тенденции.

Если распределение асимметрично, присутствуют заметные выбросы или имеется несколько мод - использование среднего значения может ввести в заблуждение, и поэтому лучше использовать моду или медиану.

Важно отметить тот факт, что не стоит ограничиваться одними статистиками. Любые зависимости лучше визуализировать. Классический пример - квартет Энскомба. Приведенные на рисунке 2.2 выборки имеют одинаковые выборочные средние, выборочные дисперсии; совпадают выборочные коэффициенты корреляции.

В 2016 году Ben Orlin опубликовал статью "Why Not to Trust Statistics?" в которой с помощью карикатур рассказал идею о том, почему не следует слепо доверять статистикам. Карикатуры приведены на страницах 39–42.



Рис. 2.3: Выбор среднего

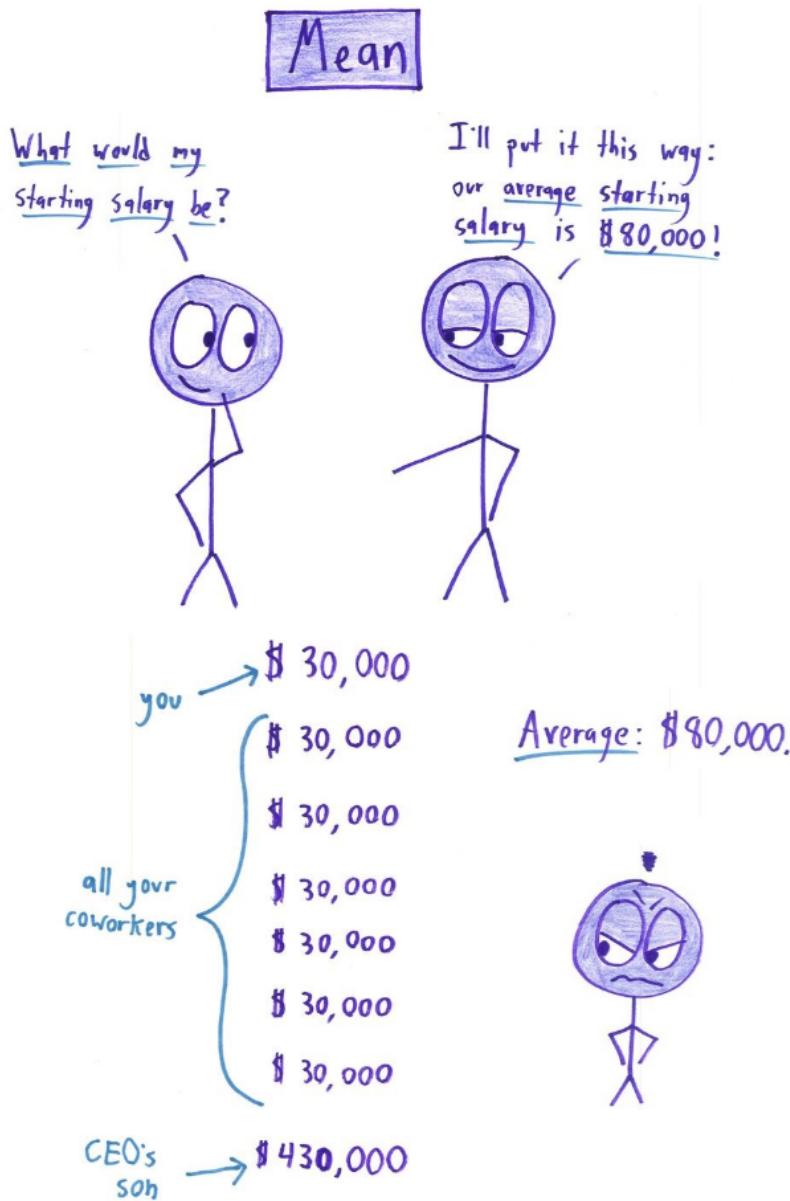


Рис. 2.4: Среднее арифметическое

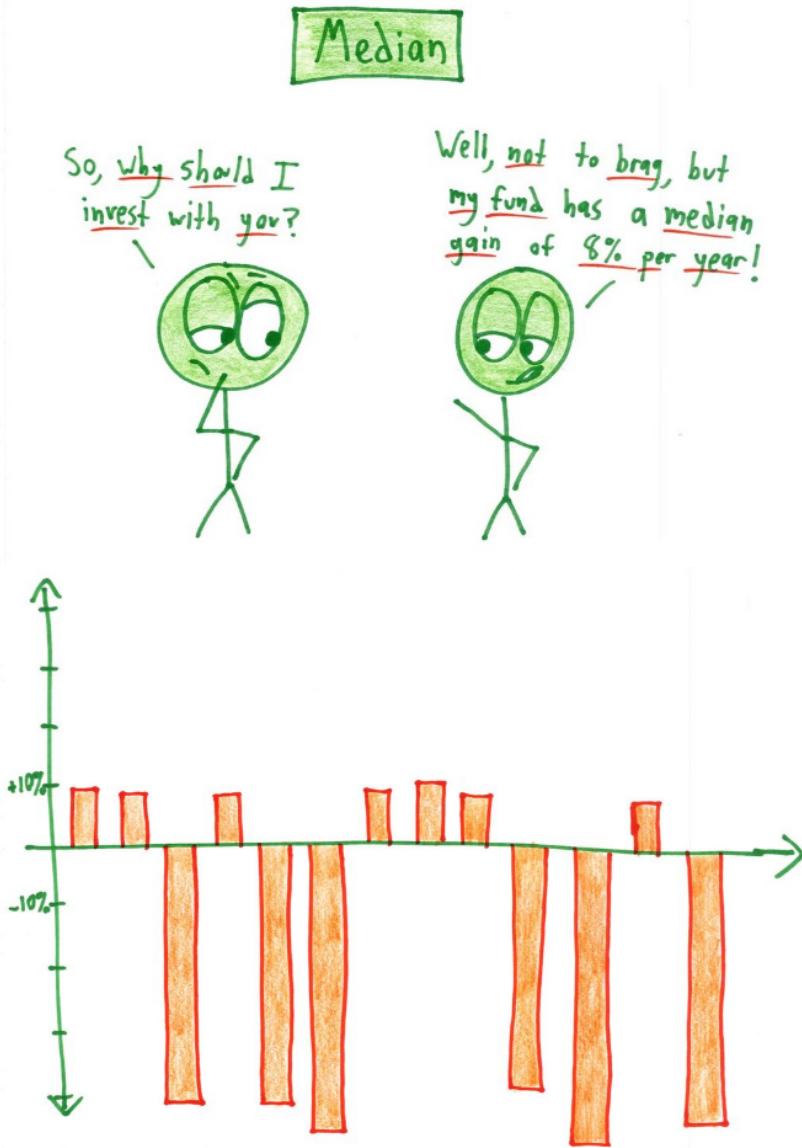


Рис. 2.5: Медиана

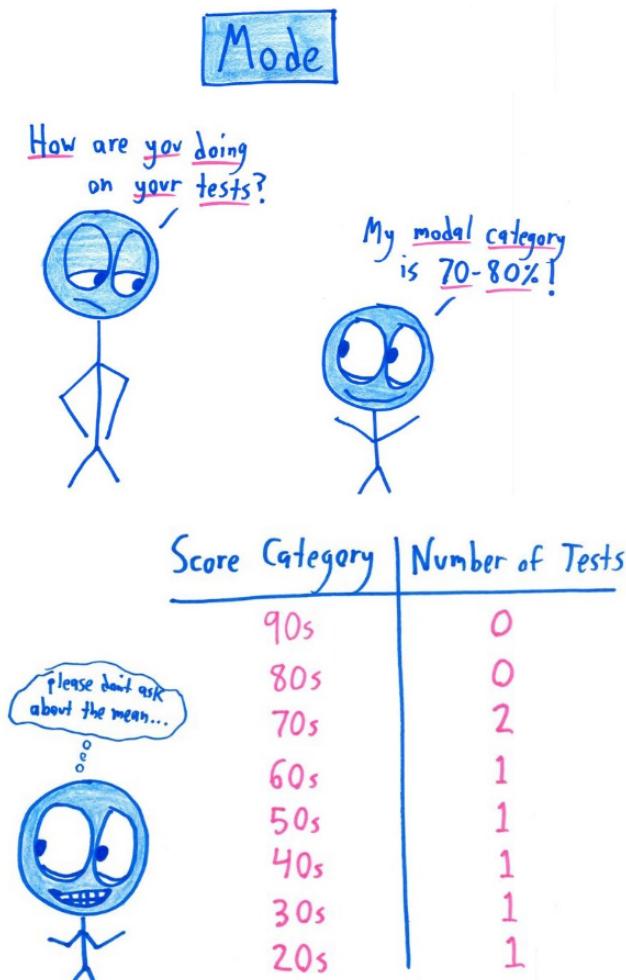


Рис. 2.6: Мода

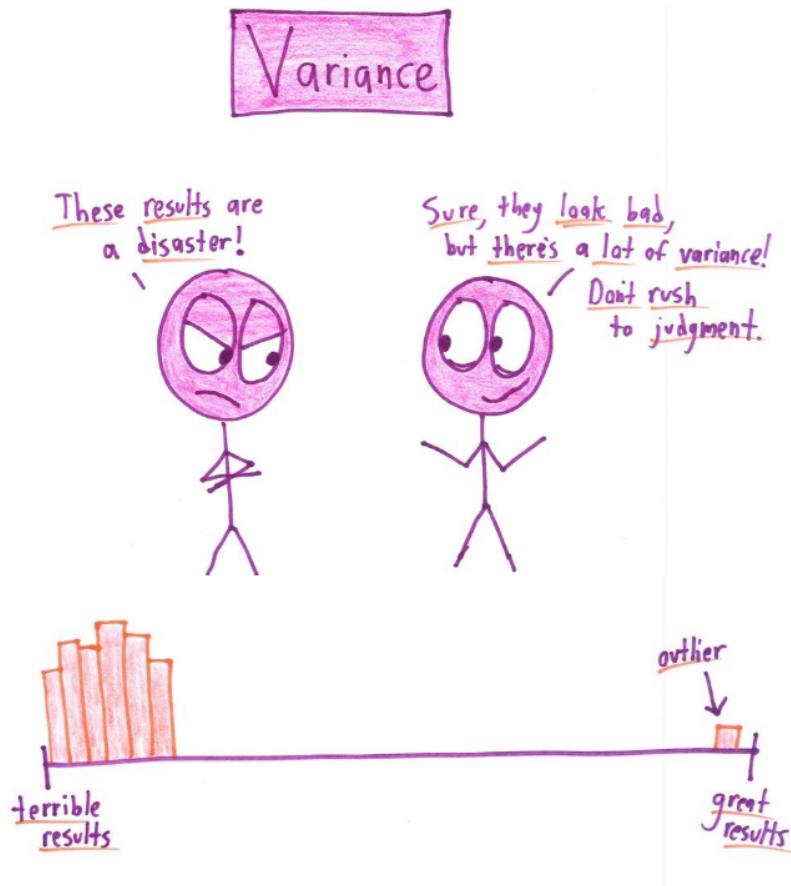


Рис. 2.7: Дисперсия

## 2.2 Интервальные оценки

При решении практических задач, часто нас интересует не только точечная, но и интервальная оценка. Мы провели два опроса об узнаваемости нашего продукта. Участие в первом приняло 10 человек, во втором - 1000. Доля людей, которые узнали наш продукт в первом эксперименте составила 0.5, во втором - 0.42. Очевидно, что второй эксперимент даёт более точную оценку.

### Предсказательный интервал

Отрезок  $[X_{\frac{\alpha}{2}}, X_{1-\frac{\alpha}{2}}]$  называется **предсказательным интервалом**, который оценивает диапазон, в котором лежит случайная величина.

$$P(X_{\frac{\alpha}{2}} \leq X \leq X_{1-\frac{\alpha}{2}}) = 1 - \alpha$$

### Доверительный интервал

Пусть распределение случайной величины  $X \sim F(x, \theta)$  зависит от параметра  $\theta$ .

**Доверительным интервалом для  $\theta$**  называется такая пара статистик  $C_L$  и  $C_U$ , что:

$$P(C_L \leq \theta \leq C_U) \geq 1 - \alpha$$

$\theta$  - оцениваемый параметр,  $1 - \alpha$  - уровень доверия,  $C_L$  - нижний доверительный предел,  $C_U$  - верхний доверительный предел.

Полученную оценку следует интерпритировать следующим образом: при бесконечном повторении процедуры построения доверительного интервала на аналогичных выборках в  $100(1 - \alpha)\%$  случаев он будет содержать истинное значение  $\theta$  (Некорректная интерпритация: неизвестный параметр лежит в пределах построенного доверительного интервала с вероятностью  $1 - \alpha$ ).

### 2.2.1 Построение доверительного интервала для среднего

Оценка среднего для нормального распределения:

$$X \sim N(\mu, \sigma^2)$$

Выборочное среднее  $\bar{X}_n$  является хорошей точечной оценкой математического ожидания  $\mathbb{E}X$ .

**Центральная предельная теорема:** при многократном вычислении среднего значения на выборках из  $X$  их распределение может быть описано нормальным с центром  $\mathbb{E}X$  и дисперсией  $\mathbb{D}X$ .

Для нормального распределения:

$$\bar{X} \sim N\left(\mu, \frac{\sigma^2}{n}\right) \Rightarrow P\left(\mu - z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \bar{X}_n \leq \mu + z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha$$

доверительный интервал для  $\mu$ :

$$P\left(\bar{X}_n - z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X}_n + z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha$$

$z_{1-\frac{\alpha}{2}}$  - квантиль стандартного нормального распределения.

Если  $X^n$  - выборка из  $F(x)$ ,  $F(x)$  не слишком скошено и  $n > 30$ :

$$\bar{X}_n \approx N(\mathbb{E}X, \frac{\mathbb{D}X}{n})$$

доверительный интервал для  $\mathbb{E}X$ :

$$P\left(\mu - z_{1-\frac{\alpha}{2}} \frac{\mathbb{D}X}{\sqrt{n}} \leq \mathbb{E}X \leq \mu + z_{1-\frac{\alpha}{2}} \frac{\mathbb{D}X}{\sqrt{n}}\right) \approx 1 - \alpha$$

Если дисперсия неизвестна:

$$P\left(\mu - t_{n-1, 1-\frac{\alpha}{2}} \frac{s_n}{\sqrt{n}} \leq \mathbb{E}X \leq \mu + t_{n-1, 1-\frac{\alpha}{2}} \frac{s_n}{\sqrt{n}}\right) \approx 1 - \alpha$$

$t_{n-1, 1-\frac{\alpha}{2}}$  - квантиль распределения Стьюдента с  $n - 1$  степенью свободы.

Проверим ЦПТ на различных распределениях (рисунки 2.8 - 2.14).

```

import numpy as np
import matplotlib.pyplot as plt

from scipy import stats

size = 10000

dists = [stats.uniform(loc=-2, scale=4), stats.t(df=10),
         stats.chi2(df=1), stats.lognorm(s=0.2)]
labels = ["Uniform", "Student", "Chi_square", "Lognormal"]

samples = [dist.rvs(size) for dist in dists]

N_ROW = 2
N_COL = 2

fig, axs = plt.subplots(N_ROW, N_COL, figsize=(18, 12))

dist_index = 0
for i in range(N_ROW):
    for j in range(N_COL):
        axs[i, j].set_title(labels[dist_index], fontsize=14)
        axs[i, j].hist(samples[dist_index], bins=20, color=[0.7, 0.7, 0.7],
                        edgecolor="k")
        dist_index += 1

plt.show()

```

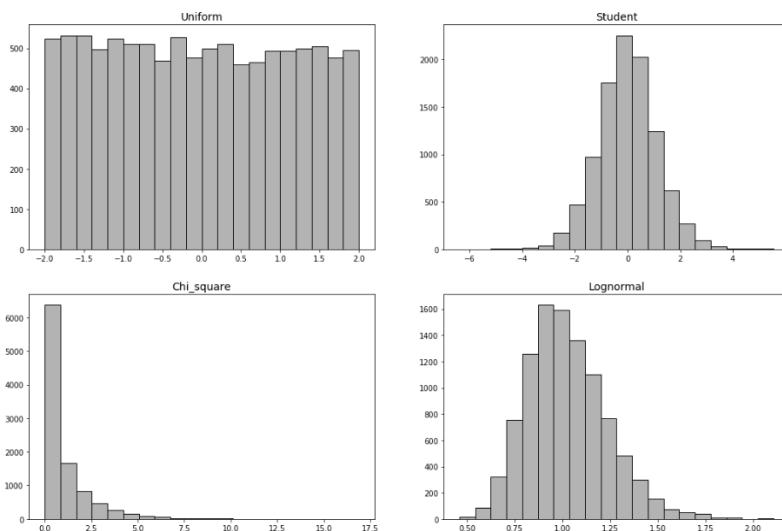


Рис. 2.8: Центральная предельная теорема

Из каждого распределения будем извлекать выборки объемом в 10 раз меньше исходной, вычислять выборочное среднее и строить распределение выборочных средних.

```
%matplotlib inline
from math import ceil

def CPD_plot(sample, dist):
    N_EXPS = [20, 40, 80, 160, 320, 640, 1280, 2560, 5120, 10240]
    N_COL = 2

    for num_of_exp in range(len(N_EXPS)):

        if num_of_exp % N_COL == 0:
            fig, axs = plt.subplots(1, N_COL, figsize=(12, 4))
        n_exp = N_EXPS[num_of_exp]
        sample_means = []
        for i in range(n_exp):
            sample_means.append(np.random.choice(sample,
                                                size=int(size/10)).mean())

        row_index = num_of_exp // N_COL
        col_index = num_of_exp % N_COL
        axs[col_index].hist(sample_means, bins=20, edgecolor="k",
                            color=[0.7, 0.7, 0.7])
        axs[col_index].axvline(x=dist.mean(), color="red",
                               label = "среднее \ngen.совокупности")
        axs[col_index].axvline(x=sample.mean(), color="green",
                               label = "выборочное \n\nc\ преднее")
        axs[col_index].set_title("n = " + str(n_exp))
        axs[col_index].legend(fontsize=12)

    plt.show()
```

## Равномерное распределение

```
CPD_plot(samples[0], dists[0])
```

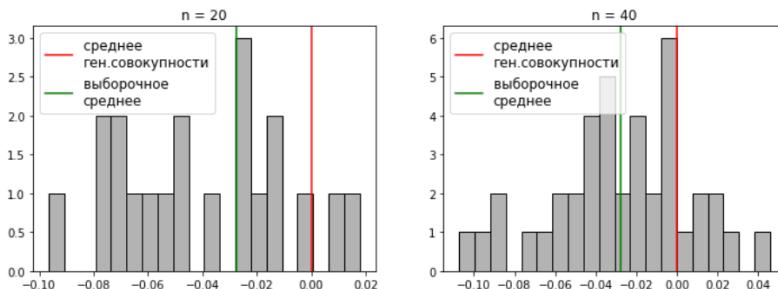


Рис. 2.9: "Центральная предельная теорема (продолжение)"

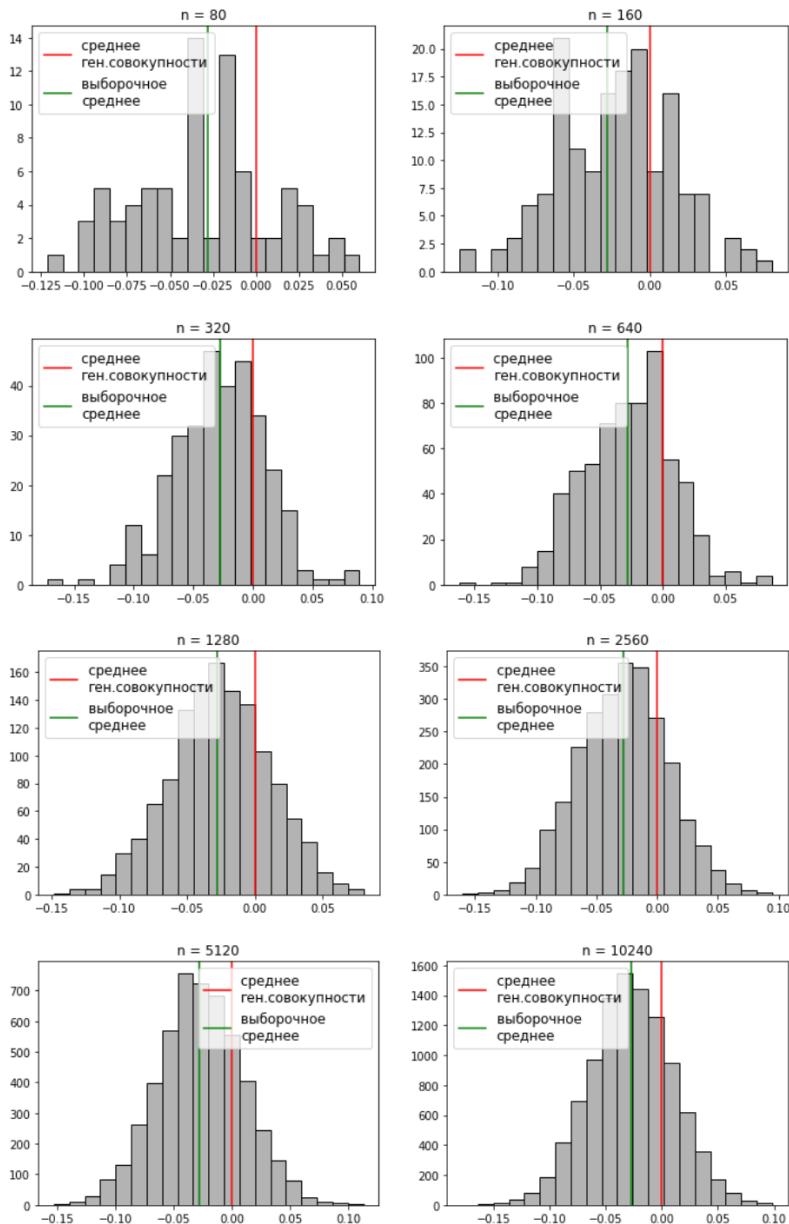


Рис. 2.10: "Центральная предельная теорема (продолжение)"

### Распределение Стьюдента

```
CPD_plot(samples[1], dists[1])
```

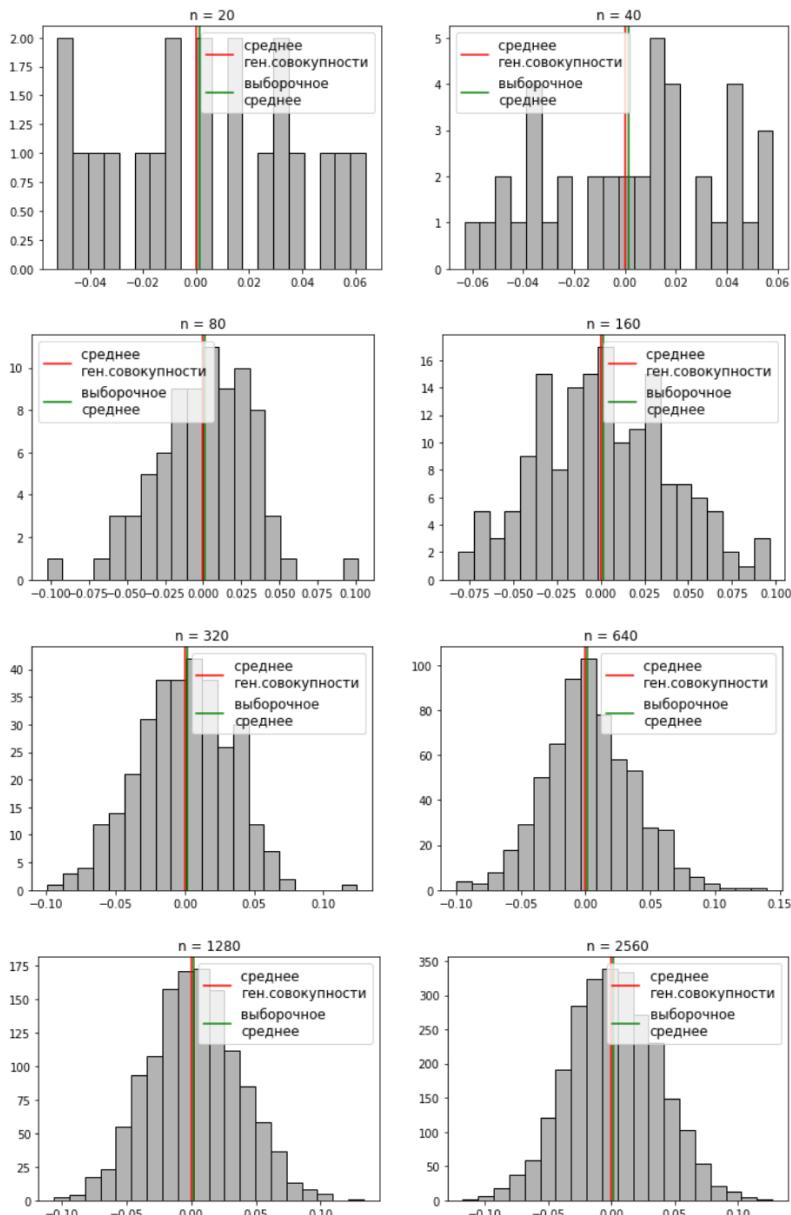
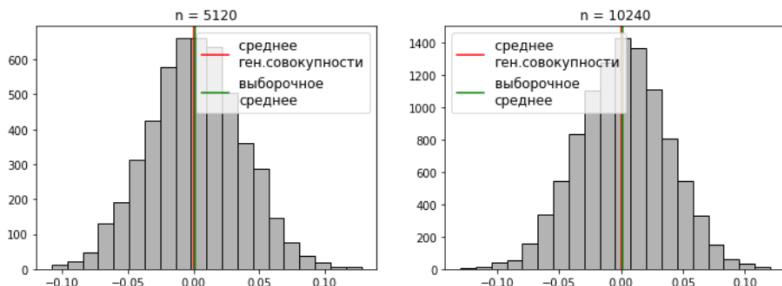


Рис. 2.11: "Центральная предельная теорема (продолжение)"



### Распределение $\chi^2$ -квадрат

```
CPD_plot(samples[2], dists[2])
```

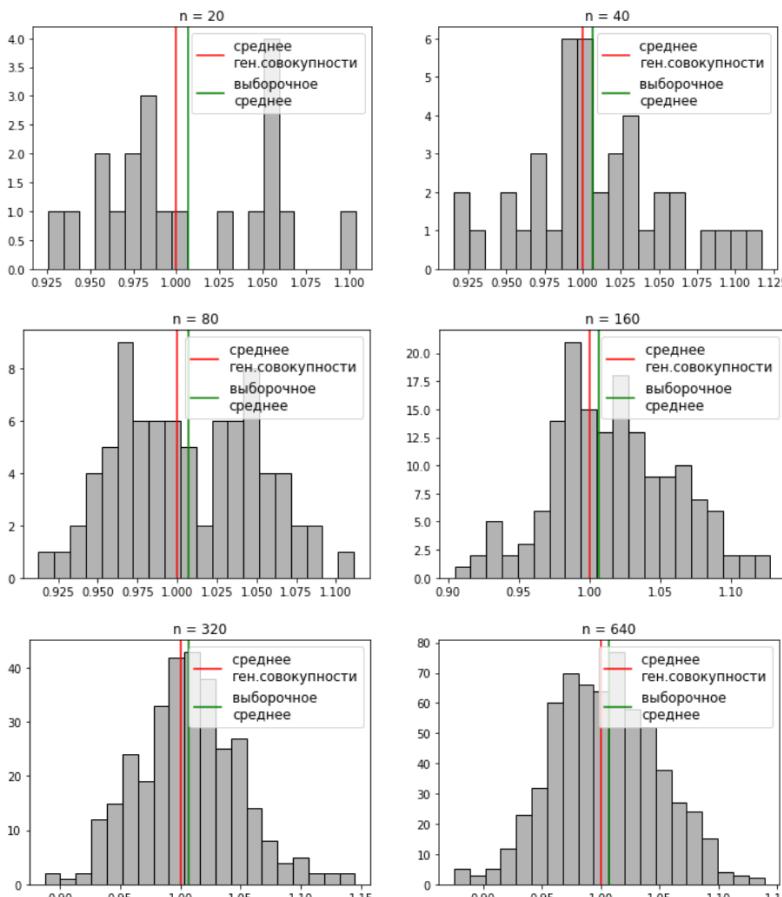
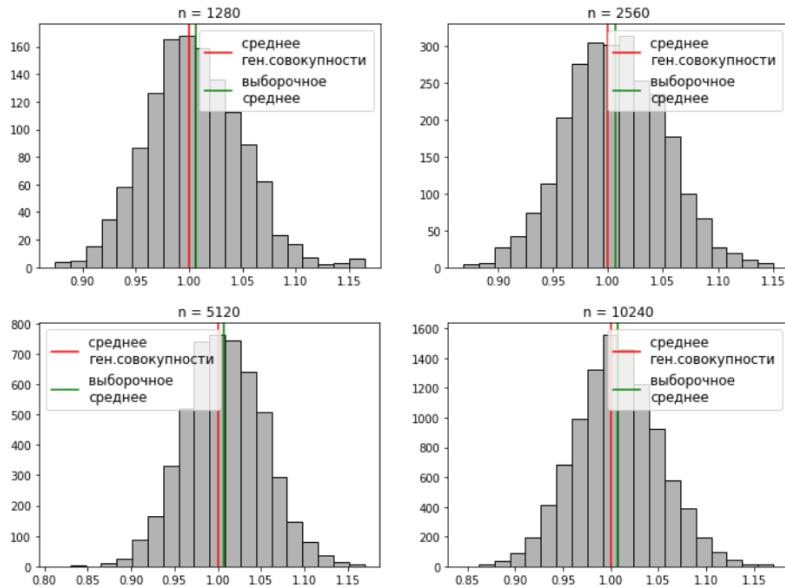


Рис. 2.12: "Центральная предельная теорема (продолжение)"



### Логнормальное распределение

```
CPD_plot(samples[3], dists[3])
```

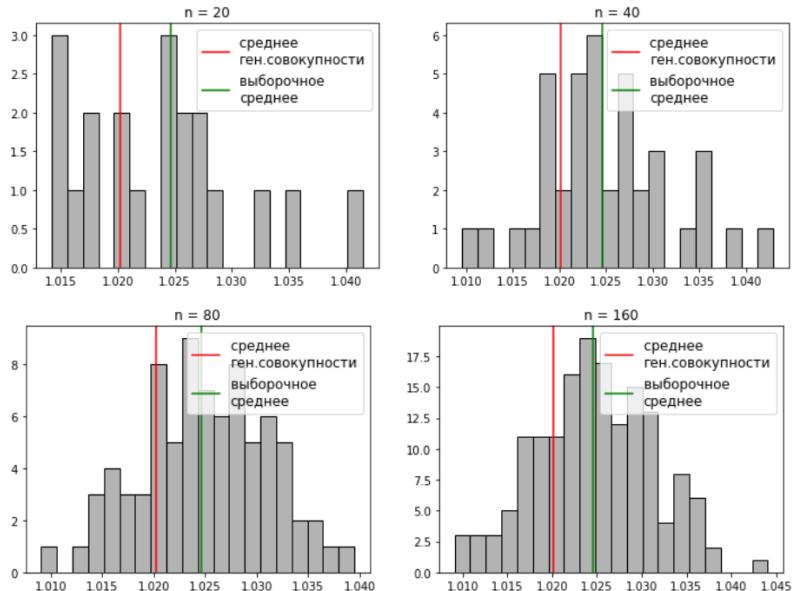


Рис. 2.13: "Центральная предельная теорема (продолжение)"

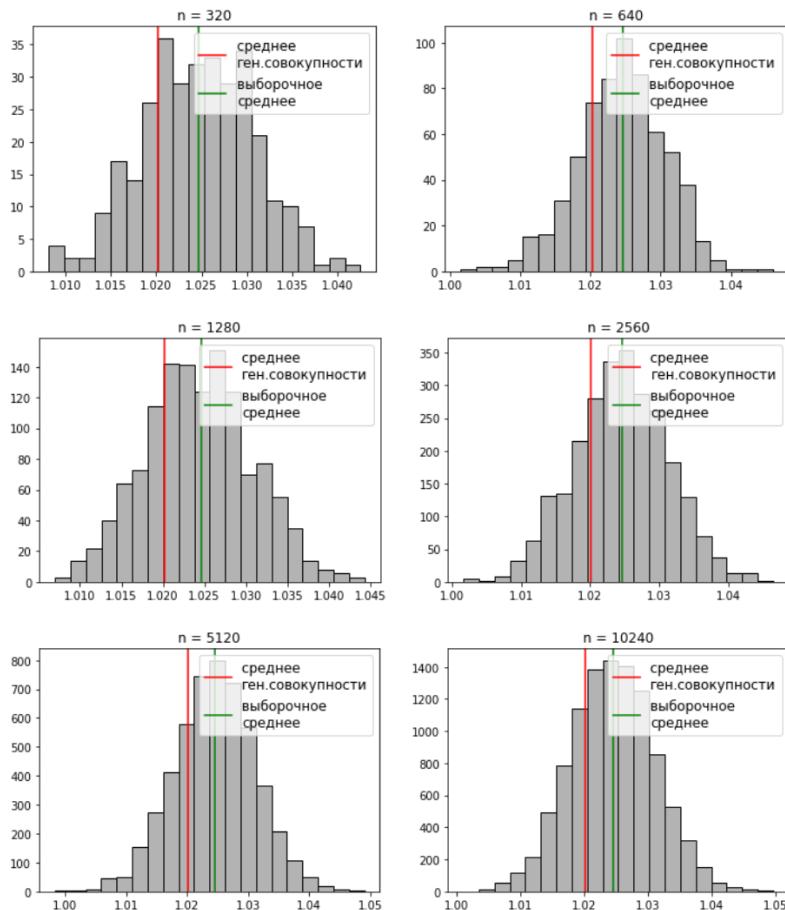


Рис. 2.14: "Центральная предельная теорема (окончание)"

### 2.2.2 Построение доверительного интервала для доли

Вернёмся к задаче об опросах (страница 43):

$$\bar{p}_1 = 0.5, \quad n_1 = 10$$

$$\bar{p}_2 = 0.42, \quad n_2 = 1000$$

Отметим, что наша случайная величина распределена по закону Бернулли (у нас всего два исхода: знает/не знает). Применим центральную предельную теорему:

$$\bar{X} \approx \sim N \left( \mathbb{E}X, \frac{\mathbb{D}X}{n} \right)$$

$$\bar{X} = \bar{p} \quad \bar{p}_n \approx \sim N \left( \mathbb{E}X, \frac{\mathbb{D}X}{n} \right)$$

Математическое ожидание и дисперсия нашей случайной величины:

$$X \sim Ber(p) \Rightarrow \mathbb{E}X = p, \mathbb{D}X = p \cdot (1 - p)$$

Выполним подстановку:

$$\bar{p}_n \approx \sim N \left( p, \frac{p \cdot (1 - p)}{n} \right)$$

Однако, у нас нет значения  $p$ , подставим вместо него  $\bar{p}$

$$\bar{p}_n \approx \sim N \left( \bar{p}, \frac{\bar{p} \cdot (1 - \bar{p})}{n} \right)$$

Применим правило двух сигм:

$$P \left( \bar{p}_n - 2\sqrt{\frac{\bar{p}_n(1 - \bar{p}_n)}{n}} \leq p \leq \bar{p}_n + 2\sqrt{\frac{\bar{p}_n(1 - \bar{p}_n)}{n}} \right) \approx 0.95$$

Выполнив необходимые подстановки мы получим следующее. 95 % доверительный интервал для доли в первом случае - [0.3419, 0.6581], а во втором случае - [0.4044, 0.4356]. Очевидно, что при увеличении количества наблюдений в выборке, мы получим более узкий доверительный интервал.

Важно отметить, что любая оценка, полученная по выборке, не является точной. Доверительный интервал лишь позволяет нам оценить степень неточности.

Помимо центральной предельной теоремы, построить доверительный интервал для бернуллевской случайной величины можно использовать метод Уилсона, который позволяет получить более узкий и более точный промежуток.

$$\frac{1}{1 + \frac{z^2}{n}} \left( \hat{p} + \frac{z^2}{2n} \right) \pm z \sqrt{\frac{\hat{p}(1 - \hat{p})}{n} + \frac{z^2}{4n^2}} \quad z \equiv z_{1 - \frac{\alpha}{2}}$$

### 2.2.3 Построение доверительных интервалов для разности двух долей для независимых выборок

Предположим, что мы выполняем сравнение двух баннеров. Мы собрали две независимые группы (по 100 человек в каждой) и просили оценить каждый баннер отдельно. 10-ти респондентам из первой группы понравился первый баннер, 14-ти респондентам из второй группы понравился второй баннер. Результаты представим в виде таблицы:

	$X_1$	$X_2$
1	$a$	$b$
0	$c$	$d$
$\sum$	$n_1$	$n_2$

В нашем случае:

	$X_1$	$X_2$
1	10	14
0	90	86
$\sum$	100	100

На основании этой таблицы вычисляются параметры  $\hat{p}_1$  и  $\hat{p}_2$

$$\hat{p}_1 = \frac{a}{n_1} \quad \hat{p}_2 = \frac{b}{n_2}$$

Доверительный интервал для разности двух долей  $p_1 - p_2$  оценивается по формуле:

$$\hat{p}_1 - \hat{p}_2 \pm z_{1-\frac{\alpha}{2}} \sqrt{\frac{\hat{p}_1(1-\hat{p}_1)}{n_1} + \frac{\hat{p}_2(1-\hat{p}_2)}{n_2}}$$

Выполним расчеты:

$$\hat{p}_1 = 0.1 \quad \hat{p}_2 = 0.14 \quad z_{0.975} \approx 1.96$$

$$0.1 - 0.14 \pm 1.96 \sqrt{\frac{0.1 \cdot 0.9}{100} + \frac{0.14 \cdot 0.86}{100}}$$

95% доверительный интервал для разности двух долей:  $[-0.13, 0.05]$ . Данный интервал включает 0, следовательно, мы не можем утверждать что существует статистически значимое различие между двумя представленными баннерами.

#### 2.2.4 Построение доверительных интервалов для разности двух долей для связанных выборок

Предположим, что оценка каждого баннера производилась одной и той же группой. Каждый респондент оценивал и первый и второй баннер. Данные выборки являются связанными.

Таблица сопряженности:

		$X_2$		$\Sigma$
		1	0	
$X_1$	1	$e$	$f$	$e + f$
	0	$g$	$h$	$g + h$
$\Sigma$		$e + g$	$f + h$	$n$

Таблица 2.1: Таблица сопряженности

Результаты опроса представлены в таблице 2.2.

Вычисление параметров:

$$\hat{p}_1 = \frac{e + f}{n} \quad \hat{p}_2 = \frac{e + g}{n} \quad \hat{p}_1 - \hat{p}_2 = \frac{f - g}{n}$$

$X_1 \backslash X_2$	1	0	$\Sigma$
1	20	40	60
0	14	26	40
$\Sigma$	34	66	100

Таблица 2.2: Результаты опроса

Доверительный интервал для разности двух долей:

$$\hat{p}_1 - \hat{p}_2 \pm z_{1-\frac{\alpha}{2}} \sqrt{\frac{f+g}{n^2} + \frac{(f-g)^2}{n^3}}$$

Выполним расчеты:

$$\hat{p}_1 = 0.6 \quad \hat{p}_2 = 0.34 \quad \hat{p}_1 - \hat{p}_2 = 0.26 \quad z_{0.975} \approx 1.96$$

Точечная оценка разности двух долей составляет 26%. 95% доверительный интервал для разности двух долей: [0.1072, 0.4127]. Данний интервал не включает в себя ноль, следовательно, мы можем утверждать, что два баннера различаются статистически значимо.

### 2.2.5 Непараметрический доверительный интервал для медианы непрерывного распределения

$$X^n = (X_1, \dots, X_n), \quad X \sim F(x) \Rightarrow$$

$$P(\text{med}X \in [X_r, X_{n-r+1}]) = \frac{1}{2^n} \sum_{i=r}^{n-r+1} C_n^i$$

При  $n > 10$  можно применить нормальную аппроксимацию:

$$P\left(\text{med}X \in \left[X\left(\left\lfloor \frac{n - \sqrt{n}z_{1-\frac{\alpha}{2}}}{2} \right\rfloor\right), X\left(\left\lceil \frac{n + \sqrt{n}z_{1-\frac{\alpha}{2}}}{2} \right\rceil\right)\right]\right) \approx 1 - \alpha$$

## 2.2.6 Построение доверительных интервалов при помощи бутстрепа

Иногда возникает задача построения доверительного интервала для какой-нибудь нетривильной величины. Для её решения может быть использована идея бутстрепа, которая заключается в том, чтобы сгенерировать  $N$  псевдовыборок объема  $n$  из  $X^n$  и вычислить значение интересующей статистики на ней:  $X^{1*}, \dots, X^{N*}$  - бутстреп-псевдовыборки,  $\theta^{1*}, \dots, \theta^{N*}$  - значения статистики на каждой псевдовыборке.

Мы можем построить бутстреп распределение  $F_{\hat{\theta}_n}^{boot}(x)$ , полученное из значений статистики на псевдобрюке. По полученной функции распределения строятся доверительные интервалы для статистики  $\theta$ .

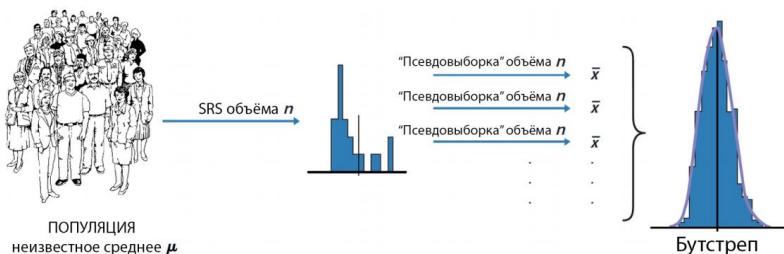


Рис. 2.15: Бутстреп

Можно выделить два вида бутстрепа:

- Базовый бутстреп (выборочные квантили бутстреп-распределения):

$$P \left( \left( F_{\hat{\theta}_n}^{boot}(x) \right)^{-1} \left( \frac{\alpha}{2} \right) \leq \theta \leq \left( F_{\hat{\theta}_n}^{boot}(x) \right)^{-1} \left( 1 - \frac{\alpha}{2} \right) \right) \approx 1 - \alpha$$

- Стьюдентизированный бутстреп:

$$P \left( \hat{\theta}_n - t_{n-1, 1-\frac{\alpha}{2}} S_n^{boot} \leq \theta \leq \hat{\theta}_n + t_{n-1, 1-\frac{\alpha}{2}} S_n^{boot} \right) \approx 1 - \alpha$$

Свойства бутстрепа:

- Простота использования для произвольной статистики
- Плохо работает для статистик, значение которых зависит от небольшого числа элементов выборки

## Глава 3

# Проверка статистических гипотез

Под **статистической гипотезой** будем понимать некоторое предположение о распределении вероятностей, лежащее в основе наблюдаемой выборки данных.

**Проверка статистических гипотез** - процесс принятия решения о том, противоречит рассматриваемая гипотеза наблюдаемой выборке данных или нет.

**Статистический тест** или **статистический критерий** - строгое математическое правило, по которому принимается или отвергается статистическая гипотеза.

### 3.1 Методика проверки статистических гипотез

Пусть задана реализация выборки  $X^m = (X_1, \dots, X_m)$  — последовательность  $m$  объектов из множества  $X$ . Предполагается, что на множестве  $X$  существует некоторая неизвестная вероятностная мера  $\mathbb{P}$ .

1. Формулируется нулевая гипотеза  $H_0$  о распределении вероятностей на множестве  $X$ . Чаще всего рассматриваются две

гипотезы — основная или нулевая  $H_0$  и альтернативная  $H_1$ . Иногда альтернатива не формулируется в явном виде; тогда предполагается, что  $H_1$  означает «не  $H_0$ ».

2. Задаётся некоторая статистика (функция выборки)  $T : X^m \rightarrow \mathbb{R}$ , для которой в условиях справедливости гипотезы  $H_0$  выводится функция распределения  $F(T)$ .
3. Фиксируется уровень значимости — допустимая для данной задачи вероятность ошибки первого рода. Это должно быть достаточно малое число  $\alpha \in [0, 1]$ . На практике часто полагают  $\alpha = 0.05$ ,  $\alpha = 0.01$ ,  $\alpha = 0.005$ ,  $\alpha = 0.001$ .
4. На множестве допустимых значений статистики  $T$  выделяется критическое множество  $\Omega_\alpha$  наименее вероятных значений статистики  $T$ , такое, что  $\mathbb{P}\{T \in \Omega_\alpha | H_0\} = \alpha$ .
5. Статистический тест (статистический критерий) заключается в проверке условия:
  - если  $T(X^m) \in \Omega_\alpha$ , то делается вывод «данные противоречат нулевой гипотезе на уровне значимости  $\alpha$ », нулевая гипотеза отвергается.
  - если  $T(X^m) \notin \Omega_\alpha$ , то делается вывод «данные не противоречат нулевой гипотезе при уровне значимости  $\alpha$ », нулевая гипотеза не отвергается.

Важно отметить тот факт, что если мы не отвергаем нулевую гипотезу, это не означает, что она верна.

**Достигаемый уровень значимости ( $p - value$ )** — наименьшая величина уровня значимости, при которой нулевая гипотеза отвергается для данного значения статистики критерия  $T$ .

$$p(T) = \min\{\alpha : T \in \Omega_\alpha\}$$

где  $\Omega_\alpha$  — критическая область критерия.

Альтернативное определение: **достигаемый уровень значимости ( $p - value$ )** — это вероятность получить такое же значение статистики, как в эксперименте, или еще более экстремальное, при справедливости нулевой гипотезы.  $p - value$  - это не вероятность

**того, что верна нулевая гипотеза; к сожалению, нет способа получить такую оценку.**

$$p = P(T \geq t | H_0) \neq P(H_0) \neq P(H_0 | T \geq t)$$

О выборе альтернативы нужно сделать одно замечание. Иногда получается так, что при двусторонней альтернативе нулевая гипотеза не отвергается на некотором уровне значимости. А при односторонней альтернативе нулевая гипотеза отвергается. Однако стоит отметить, что альтернатива должна выбираться до получения данных. А односторонняя альтернатива может быть использована только тогда, когда заранее известен знак изменения.

Рассмотрим концепцию на примере: Джеймс Бонд говорит, что предпочитает мартини взболтанным, но не смешанным. Проведём слепой тест:  $n$  раз предложим ему пару напитков и выясним, какой из двух он предпочитает.

Выборка: бинарный вектор длины  $n$ . 1 — Джеймс Бонд выберет взболтанный мартини, 0 — смешанный мартини.

Нулевая гипотеза: Джеймс Бонд не различает два вида мартини, т. е., выбирает наугад.

Статистика  $T$ : число единиц в выборке.

Предположим, что  $n = 16$ . Тогда существует  $2^{16} = 65536$  равновероятных варианта. Статистика  $T$  принимает значения от 0 до 16.

Пусть  $H_1$ : Джеймс Бонд предпочитает взболтанный мартини. При справедливости такой альтернативы более вероятны большие значения  $T$  (т.е. большие  $T$  свидетельствуют против  $H_0$  в пользу  $H_1$ ; односторонняя альтернатива (рисунок 3.1)). Вероятность того, что Джеймс Бонд предпочтёт взболтанный мартини в 12 или более случаях из 16 при справедливости  $H_0$ , равна  $\frac{2517}{65536} \approx 0.0384$ .

Пусть  $H_1$ : Джеймс Бонд предпочитает какой-то определенный вид мартини. При справедливости такой альтернативы и большие, и маленькие значения  $T$  свидетельствуют против  $H_0$  в пользу  $H_1$ ; двусторонняя альтернатива (рисунок 3.2)). Вероятность того, что Джеймс Бонд предпочтёт взболтанный мартини в  $\geq 12$  случаях из 16 или в  $\leq 4$  при справедливости  $H_0$ , равна  $\frac{5034}{65536} \approx 0.0768$ .

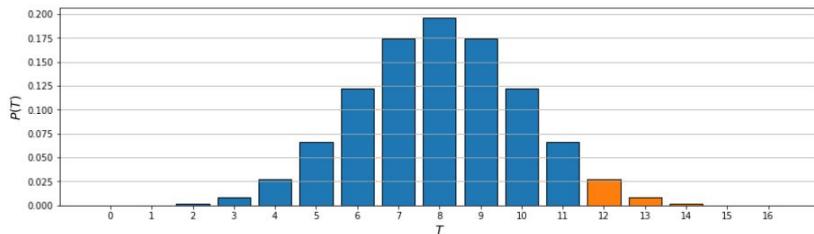


Рис. 3.1: Односторонняя альтернатива

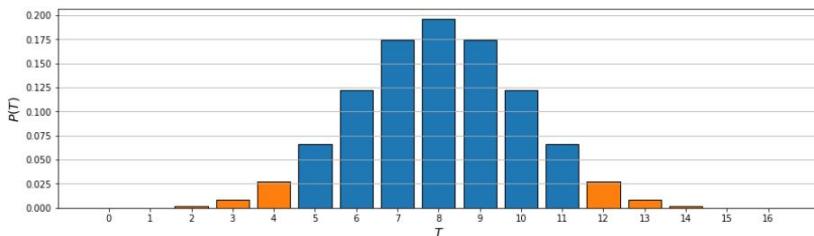


Рис. 3.2: Двусторонняя альтернатива

## 3.2 Ошибки первого и второго рода

**Ошибкой первого рода** называется ситуация, когда нулевая гипотеза была верна, но была отвергнута. **Ошибкой второго рода** называется ситуация, когда нулевая гипотеза неверна, но не была отвергнута (рисунок 3.3) .

	$H_0$ верна	$H_0$ неверна
$H_0$ принимается	$H_0$ верно принята	Ошибка II рода
$H_0$ отвергается	Ошибка I рода	$H_0$ верно отвергнута

Рис. 3.3: Ошибки первого и второго рода

Любой корректный, хорошо построенный критерий имеет вероятность ошибки первого рода не больше, чем  $\alpha$ .

Ошибка второго рода минимизируется по остаточному принципу. Понятие ошибки второго рода связано с понятием мощности статистического критерия.

Под **мощностью статистического критерия** понимается вероятность отвергнуть неверную нулевую гипотезу.

Чтобы найти идеальный критерий для проверки пары «нулевая гипотеза – альтернатива», нужно среди всех корректных критериев выбрать критерий с максимальной мощностью.

Если выборка маленькая (граница условная, около 30 наблюдений) проверить гипотезу по ней удастся, однако вероятность ошибки второго рода будет значительно выше. Некоторые практики повышают  $p$ -уровень значимости при анализе маленьких выборок.

### 3.3 Типы статистических критериев

Статистические критерии могут быть разделены на две группы:

1. **Параметрические критерии** предполагают, что выборка порождена распределением из заданного параметрического семейства. Многие критерии работают с выборками из нормального распределения. Параметрические критерии являются более мощными, чем непараметрические. Но если выборка не удовлетворяет требуемым условиям, вероятность совершить ошибку первого или второго рода резко возрастает. **Не стоит использовать параметрические критерии, если выборка не удовлетворяет необходимым условиям. Критерии согласия** используется для проверки предположений о виде распределений:

$$H_0 : F_n(x) \sim F(x)$$

которые можно разбить на следующие группы:

- критерии, основанные на изучение разницы между теоретической плотностью распределения и эмпирической гистограммой;
- критерии, основанные на расстоянии между теоретической и эмпирической функциями распределения вероятностей;
- корреляционно-регрессионные критерии, которые основаны на изучении корреляционных и регрессионных связей между эмпирическими и порядковыми статистиками.

2. **Непараметрические критерии** не выдвигают предположений о виде распределения.

### Критерий согласия $\chi^2$

Пусть мы выдвигаем предположение о том, что случайная величина имеет некоторое распределение  $F(X)$ . Диапазон измерения экспериментальных данных разбивается на  $k$  интервалов и высчитывается статистика:

$$\chi^2 = \sum_{i=1}^k \frac{(n_i - np_i)^2}{np_i}$$

где  $n$  - число элементов в выборке,  $n_i$  - количество наблюдений случайной величины, которое попало в  $i$ -ый интервал, а  $p_i$  - теоретическая вероятность попадания случайной величины в  $i$ -ый интервал. Замечания:

- Статистика  $\chi^2$  может быть использована, если  $np_i \leq 5$  (допускается невыполнение правила в 20% интервалов).
- Различные источники дают разные рекомендации по выбору  $k$ . Возможное решение:
  - $k \leq \frac{n}{5}$
  - $k = 1 + 3.32 \ln n$  при  $n < 200$
  - $3.78(n-1)^{2/5}$  при  $n \geq 200$
- Для выбора  $p_i$  допустимо следовать рекомендации  $p_i = \frac{1}{k} = const$

## 3.4 Связь между проверкой гипотез и доверительными интервалами

Гипотезы вида  $H_0 : \theta = \theta_0$  можно проверять при помощи доверительных интервалов для  $\theta$ :

- если  $\theta_0$  не попадает в  $100(1 - \alpha) \%$  доверительный интервал для  $\theta$ , то  $H_0$  отвергается на уровне значимости  $\alpha$ ;
- $p-value$  — максимальное  $\alpha$ , при котором  $\theta_0$  попадает в соответствующий доверительный интервал

# Глава 4

## Параметрические критерии

### 4.1 Нормальность распределения

Существует группа критериев, которые являются частными случаями критерия согласия, называемые **критериями нормальности**, которые используются для проверки на нормальность распределения.

Есть относительно много критериев нормальности. Однако основным из используемых является критерий Шапиро-Уилка. Критерий Колмогорова-Смирнова практически вышел из употребления, а критерий Андерсона-Дарлинга имеет высокую вероятность ошибки второго рода.

#### Критерий Шапиро-Уилка

Описание критерия Шапиро-Уилка представлено в таблице 4.1.

Не рекомендуется анализировать выборки объемом больше 5000 значений. Даже при генерации нормально распределенной выборки критерий Шапиро-Уилка может отклонять нулевую гипотезу на довольно низком уровне значимости. Существует рекомендация к использованию графических методов для таких случаев, к которым относится QQ-график.

выборка: нулевая гипотеза: альтернативная гипотеза: статистика:  нулевое распределение	$X^n = (X_1, \dots, X_n)$ $H_0 : X \sim N(\mu, \sigma^2)$ $H_1 : H_0$ неверна $W(X^n) = \frac{\sum_{i=1}^n (a_i X_{(i)})}{\sum_{i=1}^n (X_i - \bar{X})}$ $(a_1, \dots, a_n) = \frac{m^T V^{-1}}{(m^T V^{-1} V^{-1} m)^{1/2}}$ $m = (m_1, \dots, m_n)^T$ - матожидания порядковых статистик $N(0, 1)$ $V$ - их ковариационная матрица табличное
---	--

Таблица 4.1: Описание критерия Шапиро-Уилка

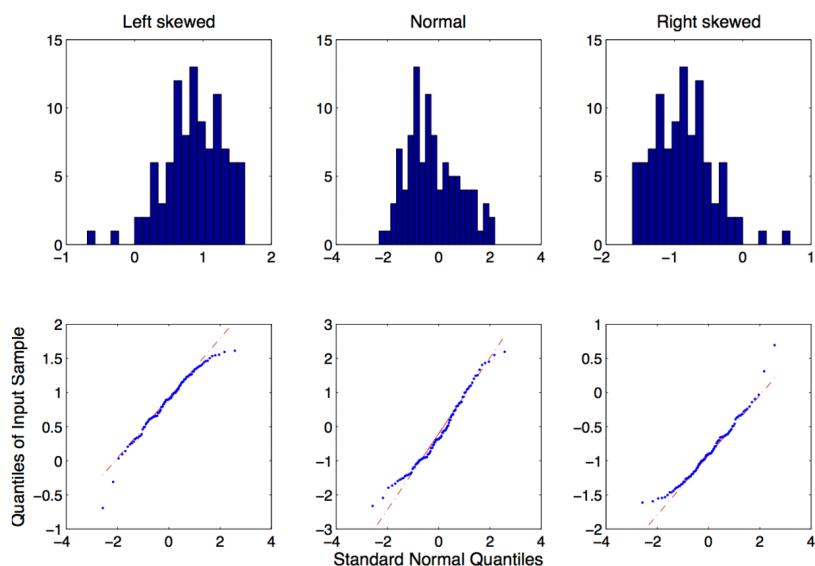


Рис. 4.1: График QQ-plot (различная асимметрия)

### QQ-график (quantile-quantile plot)

Отклонения от некоторого распределения (например, нормального) можно оценить по QQ-графику, который представляет собой график рассеяния со значением квантилей по одной оси и квантилями ожидаемого распределения по другой оси. По близости точек

к прямой мы можем визуально оценить отклонения случайной величины от теоретического распределения.

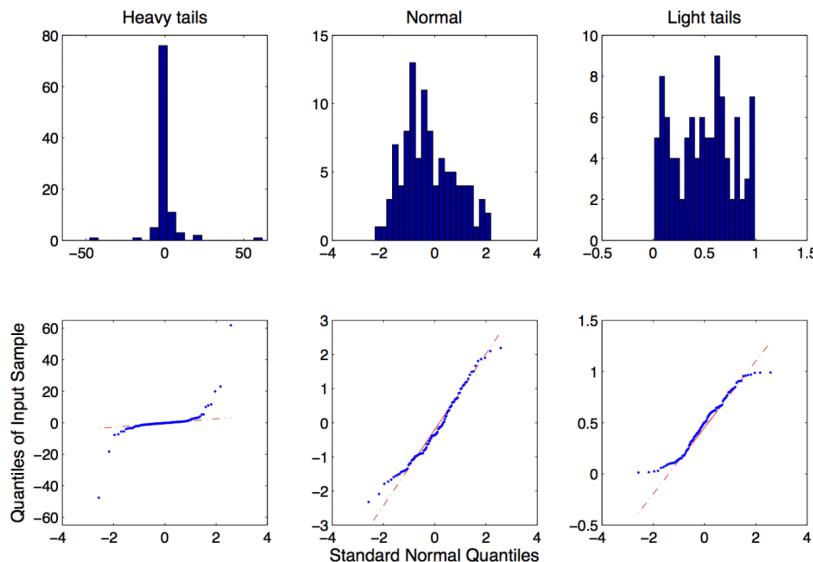


Рис. 4.2: График QQ-plot (хвосты)

В модуле stats библиотеки scipy находится функция `probplot` для построения QQ-plot.

Примеры графиков QQ-plot и соответствующие им гистограммы представлены на рисунках 4.1, 4.2.

## 4.2 Критерии предполагающие нормальное распределение

### 4.2.1 z-критерий (одновыборочный)

Описание одновыборочного z-критерия представлено в таблице 4.2

выборка:	$X^n = (X_1, \dots, X_n)$ , $\bar{X} \sim N(\mu, \sigma^2)$
нулевая гипотеза:	$\mu = \mu_0$
альтернативная гипотеза:	$\mu < \mu_0$ или $\mu \neq \mu_0$ или $\mu > \mu_0$
статистика:	$Z(X^n) = \frac{\bar{X} - \mu_0}{\sigma / \sqrt{n}}$
нулевое распределение	$Z(X^n) \sim N(0, 1)$

Таблица 4.2: Описание одновыборочного z-критерия

Достигаемый уровень значимости:

$$p = \begin{cases} F_{N(0,1)}(Z) & H_1 : \mu < \mu_0 \\ 1 - F_{N(0,1)}(Z) & H_1 : \mu > \mu_0 \\ 2(1 - F_{N(0,1)}(|Z|)) & H_1 : \mu \neq \mu_0 \end{cases}$$

Задача: линия по производству пудры должна обеспечивать средний вес пудры в упаковке 4 грамма, заявленное стандартное отклонение — 1 грамм. В ходе инспекции выбрано 9 упаковок, средний вес продукта в них составляет 4.6 грамма. Необходимо проверить, соответствует ли вес пудры в упаковке норме.

$$\begin{aligned} H_0 : \mu = 4, \quad & H_1 : \mu \neq 4 \\ \mu = 4, \quad & \sigma = 1, \quad n = 9, \quad \bar{X} = 4.6 \end{aligned}$$

Рассчитайте значение статистики  $Z$ ,  $p$ -уровень значимости, постройте 95% доверительный интервал для среднего веса.

Посчитайте  $p$ -уровень значимости при следующих альтернативах:

- $H_1$  : средний вес пудры в упаковке превышает норму.
- $H_1$  : средний вес пудры в упаковке не превышает норму.

Решение задачи представлено на рисунке 4.3 (страница 72).

### 4.2.2 z-критерий (двуихвыборочный)

Данный критерий выполняет сравнение двух средних при известных дисперсиях. Описание критерия представлено в таблице 4.3.

выборки:	$X_1^{n_1} = (X_{11}, \dots, X_{1n}), X_1 \sim N(\mu_1, \sigma_1^2)$ $X_2^{n_2} = (X_{21}, \dots, X_{2n}), X_2 \sim N(\mu_2, \sigma_2^2)$ $\sigma_1, \sigma_2$ известны
нулевая гипотеза:	$\mu_1 = \mu_2$
альтернативная гипотеза:	$\mu_1 < \mu_2$ или $\mu_1 \neq \mu_2$ или $\mu_1 > \mu_2$
статистика:	$Z(X_1^{n_1}, X_2^{n_2}) = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$
нулевое распределение	$Z(X_1^{n_1}, X_2^{n_2}) \sim N(0, 1)$

Таблица 4.3: Описание z-критерия для оценки разности между двумя математическими ожиданиями

### 4.2.3 Одновыборочный критерий Стьюдента

Одновыборочный критерий Стьюдента позволяет проверять гипотезы о математическом ожидании нормальных распределений. Описание критерия представлено в таблице 4.4.

выборка:	$X^n = (X_1, \dots, X_n)$ $X \sim N(\mu, \sigma^2)$ , $\sigma$ неизвестна
нулевая гипотеза:	$H_0 : \mu = \mu_0$
альтернативная гипотеза:	$H_1 : \mu < \mu_0$ или $\mu \neq \mu_0$ или $\mu > \mu_0$
статистика:	$T(X^n) = \frac{\bar{X} - \mu_0}{s/\sqrt{n}}$
нулевое распределение	$T(X^n) \sim St(n - 1)$

Таблица 4.4: Описание одновыборочного критерия Стьюдента

Вычисление р-уровня значимости:

$$p = \begin{cases} F_{St(n-1)}(t) & H_1 : \mu < \mu_0 \\ 1 - F_{St(n-1)}(t) & H_1 : \mu > \mu_0 \\ 2(1 - F_{St(n-1)}(|t|)) & H_1 : \mu \neq \mu_0 \end{cases}$$

Задача: средний вес детей при рождении составляет 3300 г. В то же время, если мать ребёнка живёт за чертой бедности, то средний вес таких детей — 2800 г. С целью увеличить вес тех детей, чьи матери живут за чертой бедности, разработана экспериментальная программа ведения беременности. Чтобы проверить ее эффективность, проводится эксперимент. В нём принимают участие 25 женщин, живущих за чертой бедности. У всех них рождаются дети, и их средний вес составляет 3075 г, выборочное стандартное отклонение — 500 г. Эффективна ли программа?

$$\bar{X} = 3075, \quad s = 500, \quad n = 25$$

Вычислите значение статистики, доверительный интервал для среднего, доверительный интервал для изменения веса.

$$H_0 : \mu = 2800, \quad H_1 : \mu \neq 2800$$

Вычислим значение статистики, нижний предел доверительного интервала для среднего, нижний предел доверительного интервала для изменения веса.

$$H_0 : \mu = 2800, \quad H_1 : \mu < 2800$$

Решение представлено на рисунке 4.4 (страница 73).

Реализация одновыборочного критерия Стьюдента находится в модуле `scipy.stats: ttest_1samp`.

#### 4.2.4 Двухвыборочный критерий Стьюдента для независимых выборок

Двухвыборочный критерий Стьюдента позволяет выполнить сравнение средних значений для двух выборок. Информация о двухвыборочном критерии Стьюдента для двух независимых выборок представлена в таблице 4.5.

выборки: $X_1^{n_1} = (X_{11}, \dots, X_{1n_1})$ , $X_1 \sim N(\mu_1, \sigma_1^2)$ $X_2^{n_2} = (X_{21}, \dots, X_{2n_2})$ , $X_2 \sim N(\mu_2, \sigma_2^2)$ $\sigma_1, \sigma_2$ неизвестны нулевая гипотеза: альтернативная гипотеза: статистика: нулевое распределение	$H_0 : \mu_1 = \mu_2$ $H_1 : \mu_1 < \mu_2$ или $\mu_1 \neq \mu_2$ или $\mu_1 > \mu_2$ $T(X_1^{n_1}, X_2^{n_2}) = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$ $T(X_1^{n_1}, X_2^{n_2}) \approx \sim St(v)$ $v = \frac{\left( \frac{s_1^2}{n_1} + \frac{s_2^2}{n_2} \right)^2}{\frac{s_1^4}{n_1^2(n_1 - 1)} + \frac{s_2^4}{n_2^2(n_2 - 1)}}$
--	---

Таблица 4.5: Описание двухвыборочного критерия Стьюдента для независимых выборок

Стоит сделать несколько важных замечаний:

- нулевое распределение статистики не является точным (проблема Беренса-Фишера: невозможно точно сравнить средние значения в двух выборках, дисперсии которых неизвестны)
- данная аппроксимация достаточно точна если:
  - выборки имеют одинаковый объем
  - если знак неравенства между количеством элементов в выборке такой же, как между дисперсиями.
- если данные условия нарушаются, то вероятность ошибки первого рода начинает превышать  $\alpha$

Реализация двухвыборочного критерия Стьюдента для пары независимых выборок находится в модуле `scipy.stats: ttest_ind`.

#### 4.2.5 Двухвыборочный критерий Стьюдента для связанных выборок

Двухвыборочный критерий Стьюдента для связанных выборок выполняет сравнение средних для двух связанных выборок.

выборки: $X_1^n = (X_{11}, \dots, X_{1n}), X_1 \sim N(\mu_1, \sigma_1^2)$ $X_2^n = (X_{21}, \dots, X_{2n}), X_2 \sim N(\mu_2, \sigma_2^2)$ $\sigma_1, \sigma_2$ неизвестны нулевая гипотеза: $H_0 : \mu_1 = \mu_2$ альтернативная гипотеза: $H_1 : \mu_1 < \mu_2$ или $\mu_1 \neq \mu_2$ или $\mu_1 > \mu_2$ статистика: $T(X_1^n, X_2^n) = \frac{\overline{X_1} - \overline{X_2}}{s/\sqrt{n}}$ $s^2 = \frac{1}{n-1} \sum_{i=1}^n (D_i - \bar{D})^2, D_i = X_{1i} - X_{2i}$ нулевое распределение $T(X_1^n, X_2^n) \sim St(n-1)$
--

Таблица 4.6: Описание двухвыборочного критерия Стьюдента для связанных выборок

Описание критерия представлено в таблице 4.6.

Реализация двухвыборочного критерия Стьюдента для пары связанных выборок находится в модуле `scipy.stats: sts.ttest_rel`.

#### 4.2.6 F-критерий Фишера для сравнения двух дисперсий

С помощью данного критерия можно давать некоторые оценки дисперсиям выборок. Описание критерия представлено в таблице 4.7.

выборки: $X_1^{n_1} = (X_{11}, \dots, X_{1n_1}), X_1 \sim N(\mu_1, \sigma_1^2)$ $X_2^{n_2} = (X_{21}, \dots, X_{2n_2}), X_2 \sim N(\mu_2, \sigma_2^2)$ $H_0 : \sigma_1 = \sigma_2$ нулевая гипотеза: $H_1 : \sigma_1 < \sigma_2$ или $\sigma_1 \neq \sigma_2$ или $\sigma_1 > \sigma_2$ альтернативная гипотеза: $F(X_1^{n_1}, X_2^{n_2}) = \frac{s_1^2}{s_2^2}$ статистика: $F(n_1 - 1, n_2 - 2)$ нулевое распределение
--

Таблица 4.7: F-критерий Фишера для сравнения двух дисперсий

#### 4.2.7 Задачи

Задача №1. В текстовом файле rats.txt записаны результаты исследования, в котором приняло участие 195 крыс. 106 из них держали на строгой диете, оставшиеся 89 — на диете ad libitum. Влияет ли диета на продолжительность жизни?

Гипотезы:  $H_0$ : продолжительность жизни крыс не меняется при ограничении диеты.  $H_1$ : диета ad libitum влияет на среднюю продолжительность жизни крыс. Подробное решение на страницах 74–76.

Задача №2. В текстовом файле ADHD.txt есть информация о 24 испытуемых с СДВГ (Синдром дефицита внимания и гиперактивности), которые проверяли на себе воздействие лекарства метилфенидата и плацебо. Представленные числовые значения в данных - средняя способность к подавлению реакции (чем показатель выше, тем лучше). Требуется определить, помогает ли лекарство с борьбе с СДВГ?

Гипотезы:  $H_0$ : лекарство не помогает  $\mu_1 = \mu_2$ .  $H_1$ : лекарство каким-то образом влияет на больных  $\mu_1 \neq \mu_2$ .  $H_1$ : лекарство помогает в борьбе с СДВГ  $\mu_1 > \mu_2$ . Подробное решение на страницах 77–79.

```

from scipy.stats import norm
from math import sqrt

def z_test(mean, mu, sigma, n, alternative):
    norm_dist = norm(loc=0, scale=1)
    Z = (mean - mu) / (sigma/sqrt(n))

    if alternative in ['two-sided', '2-sided', '2s']:
        p = 2*(1 - norm.cdf(abs(Z)))
    elif alternative in ['larger', 'l']:
        p = 1 - norm.cdf(Z)
    elif alternative in ['smaller', 's']:
        p = norm.cdf(Z)
    else:
        raise ValueError('invalid alternative')

    return Z, p

```

```

def z_confint(mean, sigma, n, alpha):
    norm_dist = norm(loc=0, scale=1)
    z = norm.ppf(1 - alpha/2)
    delta = z*sigma / sqrt(n)

    return (mean - delta, mean + delta)

```

```

z_test(4.6, 4, 1, 9, "2s")
(1.799999999999999, 0.07186063822585176)

```

```

z_test(4.6, 4, 1, 9, "l")
(1.799999999999999, 0.03593031911292588)

```

```

z_test(4.6, 4, 1, 9, "s")
(1.799999999999999, 0.9640696808870741)

```

```

z_confint(4.6, 1, 9, 0.05)
(3.9466786718199818, 5.253321328180018)

```

Рис. 4.3: Решение задачи (z-критерий (одновыборочный))

```
import numpy as np

from math import sqrt
from scipy.stats import t
from statsmodels.stats.weightstats import _tconfint_generic

def t_test(sample_mean, mu, s, n, alternative):
    t_dist = t(df = n - 1)
    t_value = (sample_mean - mu) / (s/sqrt(n))

    if alternative in ['two-sided', '2-sided', '2s']:
        p = 2*(1 - t_dist.cdf(abs(t_value)))
    elif alternative in ['larger', 'l']:
        p = 1 - t_dist.cdf(t_value)
    elif alternative in ['smaller', 's']:
        p = t_dist.cdf(t_value)
    else:
        raise ValueError('invalid alternative')

    return t_value, p
```

```
t_test(3075, 2800, 500, 25, "2s")
```

```
(2.75, 0.011147829812680365)
```

```
_tconfint_generic(3075, 500 / sqrt(25), 25 - 1, 0.05, '2s')
```

```
(2868.610143837198, 3281.389856162802)
```

```
np.array(_tconfint_generic(3075, 500 / sqrt(25), 25 - 1, 0.05, '2s')) - 2800
```

```
array([-68.61014384, 481.38985616])
```

```
t_test(3075, 2800, 500, 25, "l")
```

```
(2.75, 0.005573914906340183)
```

```
_tconfint_generic(3075, 500 / sqrt(25), 25 - 1, 0.05, 'l')
```

```
(2903.911792009057, inf)
```

```
np.array(_tconfint_generic(3075, 500 / sqrt(25), 25 - 1, 0.05, 'l')) - 2800
```

```
array([-103.91179201, inf])
```

Рис. 4.4: Решение задачи (одновыборочный тест Стьюдента)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from seaborn import boxplot
from scipy.stats import shapiro, probplot, ttest_ind
```

```
rats_data = pd.read_csv("datasets/rats.txt", sep="\t")
rats_data.head()
```

	lifespan	diet
0	105	restricted
1	193	restricted
2	211	restricted
3	236	restricted
4	302	restricted

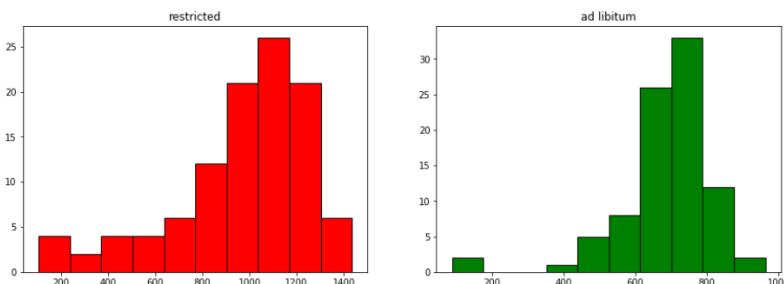
```
restricted_data = rats_data[rats_data.diet == "restricted"]
ad_libitum_data = rats_data[rats_data.diet == "ad libitum"]

plt.figure(figsize=(15, 5))

plt.subplot(121)
plt.hist(restricted_data.lifespan, edgecolor='k', color="red")
plt.title("restricted")

plt.subplot(122)
plt.hist(ad_libitum_data.lifespan, edgecolor='k', color="green")
plt.title("ad libitum")

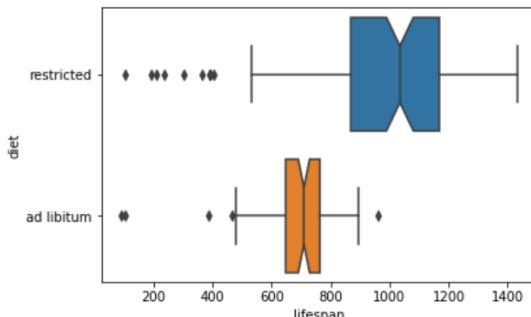
plt.show()
```



Выполним построение графика boxplot. Укажем на графике 95% доверительный интервал для медиан (notch=True).

```
boxplot(x=rats_data.lifespan, y=rats_data.diet, notch=True)
plt.show()
```

Рис. 4.5: Задача о диетах



Перед применением параметрических методов проверим выборки на нормальность для каждой из диет:

```
print(shapiro(restricted_data.lifespan))
print(shapiro(ad_liberum_data.lifespan))

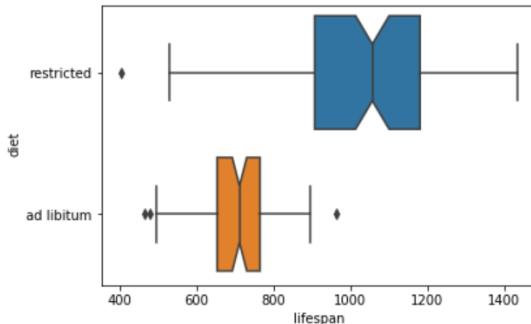
(0.9036345481872559, 1.1565858812900842e-06)
(0.8444006443023682, 3.081509092339729e-08)
```

Мы отвергаем гипотезу о том, что данные две выборки являются нормально распределенными.

Выполним предположение, что крысы, которые прожили меньше 400 дней умерли от причин, которые не связаны с диетой.

```
rats_data_subset = rats_data[rats_data.lifespan > 400]

boxplot(x=rats_data_subset.lifespan, y=rats_data_subset.diet, notch=True)
plt.show()
```



```
restricted_data_subset = rats_data_subset[rats_data_subset.diet == "restricted"]
ad_liberum_data_subset = rats_data_subset[rats_data_subset.diet == "ad libitum"]
```

```
print(shapiro(restricted_data_subset.lifespan))
print(shapiro(ad_liberum_data_subset.lifespan))
```

Рис. 4.6: Задача о диетах (продолжение)

```
(0.9738006591796875, 0.04921640083193779)
(0.9765744209289551, 0.11983636021614075)
```

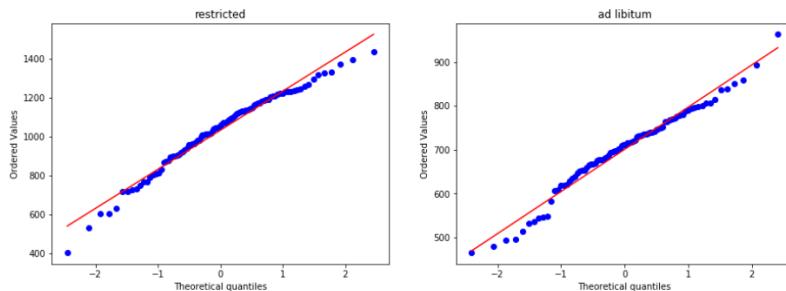
Выполним построение QQ-графика.

```
plt.figure(figsize=(15, 5))

plt.subplot(121)
probplot(restricted_data_subset.lifespan, dist="norm", plot=plt)
plt.title("restricted")

plt.subplot(122)
probplot(ad_libitum_data_subset.lifespan, dist="norm", plot=plt)
plt.title("ad libitum")

plt.show()
```



Согласно QQ-plot мы не замечаем значительных отклонений от нормального распределения. Усечённые выборки можно считать практически нормальными. Отметим, что t-тест Стьюдента допускает незначительные отклонения, следовательно, мы можем его применить.

```
print(len(ad_libitum_data_subset))
print(len(restricted_data_subset))
```

```
86
97
```

```
print(np.std(ad_libitum_data_subset))
print(np.std(restricted_data_subset))
```

```
lifespan      95.12853
dtype: float64
lifespan     198.960155
dtype: float64
```

Отмечаем тот факт, что и количество элементов в первой выборке и дисперсия в первой выборке меньше, чем во второй, следовательно, мы можем применить двухвыборочный тест Стьюдента для независимых выборок.

```
ttest_ind(a=ad_libitum_data_subset.lifespan,
          b=restricted_data_subset.lifespan,
          equal_var=False)
```

```
Ttest_indResult(statistic=-14.527701019046814, pvalue=7.882364548202608e-30)
```

Рис. 4.7: Задача о диетах (окончание)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import shapiro, probplot, ttest_rel
```

```
ADHD_data = pd.read_csv("datasets/ADHD.txt", sep = " ")
ADHD_data.head()
```

	D0	D60
0	57	62
1	27	49
2	32	30
3	31	34
4	34	38

Выполним несколько построений:

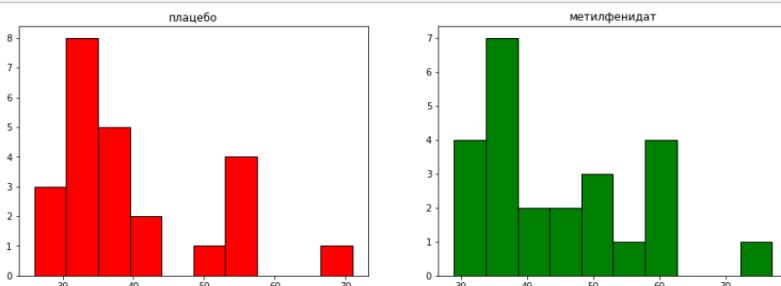
```
D0_data = ADHD_data.D0
D60_data = ADHD_data.D60

plt.figure(figsize=(15, 5))

plt.subplot(121)
plt.hist(D0_data, edgecolor='k', color="red")
plt.title("плацебо")

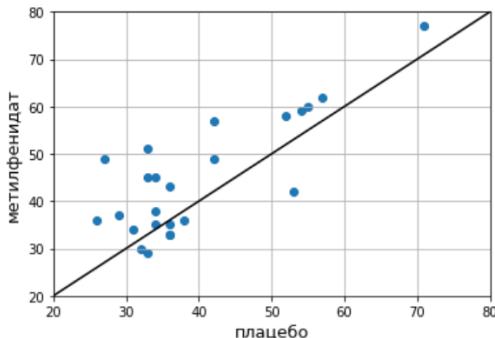
plt.subplot(122)
plt.hist(D60_data, edgecolor='k', color="green")
plt.title("метилфенидат")

plt.show()
```



```
plt.scatter(D0_data, D60_data)
plt.plot(range(100), c='black')
plt.xlim((20, 80))
plt.ylim((20, 80))
plt.xlabel("плацебо", fontsize=13)
plt.ylabel("метилфенидат", fontsize=13)
plt.grid()
plt.show()
```

Рис. 4.8: Задача о метилфенидате



```
print(shapiro(D0_data))
print(shapiro(D60_data))

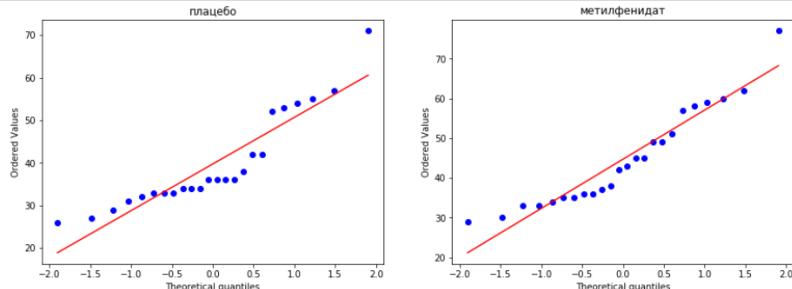
(0.8574873208999634, 0.00302018690854311)
(0.9164453148841858, 0.04876822605729103)
```

```
plt.figure(figsize=(15, 5))

plt.subplot(121)
probplot(D0_data, dist="norm", plot=plt)
plt.title("плацебо")

plt.subplot(122)
probplot(D60_data, dist="norm", plot=plt)
plt.title("метилфенидат")

plt.show()
```



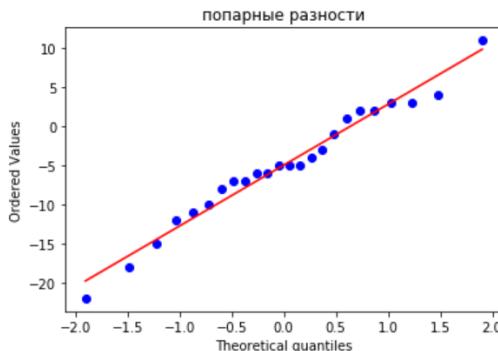
Перед применением t-теста для связанных выборок мы должны проверить разности попарных наблюдений на нормальность:

```
print(shapiro(D0_data - D60_data))

(0.9798052906990051, 0.8922504186630249)

probplot(D0_data - D60_data, dist="norm", plot=plt)
plt.title("парные разности")
plt.show()
```

Рис. 4.9: Задача о метилфенидате (продолжение)



Выборки являются связанными, так как проверка плацебо и лекарства проводилась на одних и тех же людях. Следовательно, для сравнения средних используем двухвыборочный t-тест для связанных выборок

```
ttest_rel(D0_data, D60_data)
```

```
Ttest_relResult(statistic=-3.2223624451230406, pvalue=0.003771488176381471)
```

Очевидно, что если бы мы проверяли одностороннюю гипотезу, то уровень p-value получится бы в два раза меньше. В любом случае, на уровне значимости 0.005 мы можем отвергнуть нулевую гипотезу к каждой из альтернатив.

Предположим, что наши выборки не являлись бы связанными. И мы бы применяли параметрические методы, несмотря на то, что каждая из выборок несколько отклоняется от нормального распределения. Количество элементов в первой и второй выборке одинаково, следовательно, мы можем применить двухвыборочный t-тест для независимых выборок.

```
ttest_ind(D0_data, D60_data)
```

```
Ttest_indResult(statistic=-1.452163501815909, pvalue=0.1532433046938409)
```

В таком случае, мы бы не могли опровергнуть нулевую гипотезу.

Рис. 4.10: Задача о метилфенидате (окончание)

## 4.3 Критерии для долей

### 4.3.1 z-критерий для доли

Описание z-критерия для доли представлено в таблице 4.8.

выборка: нулевая гипотеза: альтернативная гипотеза: статистика: нулевое распределение	$X^n = (X_1, \dots, X_n), X \sim Ber(p)$ $H_0 : p = p_0$ $H_1 : p < p_0$ или $p \neq p_0$ или $p > p_0$ $Z = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}}, \hat{p} = \bar{X}_n$ $Z(X^n) \sim N(0, 1)$
---	--

Таблица 4.8: Описание z-критерия для доли

### 4.3.2 z-критерий для доли двух независимых выборок

Описание z-критерия для доли двух независимых выборок представлено в таблице 4.9.

выборка: нулевая гипотеза: альтернативная гипотеза: статистика: нулевое распределение	$X_1^{n_1} = (X_{11}, \dots, X_{1n_1}), X_1 \sim Ber(p_1)$ $X_2^{n_2} = (X_{21}, \dots, X_{2n_2}), X_2 \sim Ber(p_2)$ $H_0 : p_1 = p_2$ $H_1 : p_1 < p_2$ или $p_1 \neq p_2$ или $p_1 > p_2$ $Z(X_1^{n_1}, X_2^{n_2}) = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{P(1-P)(\frac{1}{n_1} + \frac{1}{n_2})}}$ $P = \frac{\hat{p}_1 n_1 + \hat{p}_2 n_2}{n_1 + n_2}$ $Z(X_1^{n_1}, X_2^{n_2}) \sim N(0, 1)$
---	--

Таблица 4.9: Описание z-критерия для доли двух независимых выборок

### 4.3.3 z-критерий для доли двух связанных выборок

Значения для  $f, g$  представлены в таблице 4.10, а описание критерия для доли двух связанных выборок - в таблице 4.11.

		$X_2$	1	0	$\Sigma$
		$X_1$			
1			$e$	$f$	$e + f$
0			$g$	$h$	$g + h$
$\Sigma$			$e + g$	$f + h$	$n$

Таблица 4.10: Таблица сопряженности

выборки:	$X_1^n = (X_{11}, \dots, X_{1n}), X_1 \sim Ber(p_1)$
	$X_2^n = (X_{21}, \dots, X_{2n}), X_2 \sim Ber(p_2)$
нулевая гипотеза:	$H_0 : p_1 = p_2$
альтернативная гипотеза:	$H_1 : p_1 < p_2$ или $p_1 \neq p_2$ или $p_1 > p_2$
статистика:	$Z(X_1^n, X_2^n) = \frac{f - g}{\sqrt{f + g - \frac{(f-g)^2}{n}}}$
нулевое распределение	$Z(X_1^n, X_2^n) \sim N(0, 1)$

Таблица 4.11: Описание z-критерия для доли двух связанных выборок

#### 4.3.4 Задачи

Задача №1. По данным опроса, 75% работников ресторанов утверждают, что испытывают на работе существенный стресс, оказывавший негативное влияние на их личную жизнь. Крупная ресторанная сеть опрашивает 100 своих работников, чтобы выяснить, отличается ли уровень стресса работников в их ресторанах от среднего. 67 из 100 работников отметили высокий уровень стресса. В другом из ресторанов наблюдалось только 22 работника из 50, испытывающие стресс. Необходимо определить, отличается ли уровень стресса в каждом из ресторанов от среднего (рисунок 4.11).

Задача №2. В текстовом файле banner\_click\_stat хранится информация об оценках двух баннеров опрашиваемыми. Рассмотрите два случая:

- Каждый баннер оценивался своей группой
- Каждый человек оценивал два баннера сразу

Существует ли значимое различие между двумя баннерами в каждом из случаев? (рисунки 4.12 – 4.13).

Задача №3. В одном из выпусков программы «Разрушители легенд» проверялось, действительно ли заразительна зевота. В эксперименте участвовало 50 испытуемых. Каждый из них разговаривал с рекрутером; в конце 34 из 50 бесед рекрутер зевал. Затем испытуемых просили подождать решения рекрутера в соседней пустой комнате.

Во время ожидания 10 из 34 испытуемых экспериментальной группы (в которой рекрутер зевал) и 4 из 16 испытуемых контрольной группы (в которой рекрутер не зевал) начали зевать. Таким образом, разница в доле зевающих людей в этих двух группах составила примерно 4.4%. Ведущие заключили, что миф о заразительности зевоты подтверждён.

Отличаются ли доли зевающих в контрольной и экспериментальной группах статистически значимо (рисунок 4.14)?

Задача №4. В текстовом файле banknotes.txt имеются данные измерений двухсот швейцарских тысячекроновых банкнот, бывших

в обращении в первой половине XX века. Сто из банкнот были настоящими, и сто — поддельными.

Необходимо:

1. Отделить 50 случайных наблюдений в тестовую выборку с помощью функции `sklearn.model_selection.train_test_split`.
2. На оставшихся 150 записях настроить два классификатора поддельности банкнот:
  - (a) логистическую регрессию по признакам  $X_1, X_2, X_3$ ;
  - (b) логистическую регрессию по признакам  $X_4, X_5, X_6$
3. Каждым из классификаторов сделать предсказания меток классов на тестовой выборке.

Однаковы ли доли ошибочных предсказаний двух классификаторов? Проверьте данную гипотезу и вычислите достигаемый уровень значимости (рисунки 4.15 – 4.16).

Задача №5. В 5 серии 13 сезона передачи «Разрушители легенд» проверялась справедливость выражения «know something like the back of one's hand». В эксперименте принимало участие 12 испытуемых. Каждому из них были предъявлены 10 фотографий похожих рук разных людей, среди которых они должны были угадать свою. 11 из 12 испытуемых выбрали свою фотографию.

$H_0$ : испытуемые выбирают фотографии тыльной стороны руки наугад ( $p = 0.1$ ).  $H_1$ : испытуемые выбирают фотографию тыльной стороны своей собственной руки осознанно ( $p > 0.1$ ).

Аналогичный эксперимент был проведён с фотографиями ладоней. 7 из 12 испытуемых угадали свою фотографию.

$H_0$ : испытуемые выбирают фотографии ладони наугад ( $p = 0.1$ ).  $H_1$ : испытуемые выбирают фотографию своей собственной ладони осознанно ( $p > 0.1$ ).

Можно ли утверждать, что тыльную сторону руки люди знают лучше, чем ладонь? Сравните результаты экспериментов. Поскольку это одни и те же испытуемые, выборки связанные, но информации о связности нет, поэтому используйте критерий для независимых выборок (рисунок 4.17).

```

import numpy as np
import matplotlib.pyplot as plt

from scipy import stats

n = 100
prob = 0.75
F_H0 = stats.binom(n, prob)

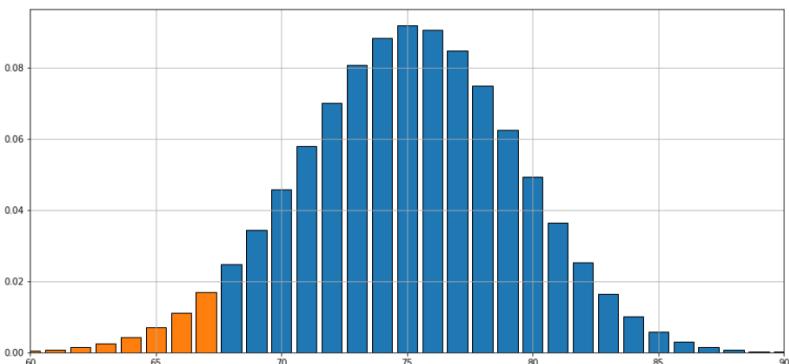
plt.figure(figsize=(15, 7))

x = np.linspace(0, 100, 101)
plt.bar(x, F_H0.pmf(x), align = 'center', edgecolor='k')

x = np.linspace(0, 67, 68)
plt.bar(x, F_H0.pmf(x), align = 'center', edgecolor='k')

plt.xlim(60, 90)
plt.grid()
plt.show()

```



```
print('p-value: %.4f' % stats.binom_test(67, 100, prob, "less"))
```

p-value: 0.0446

```
print('p-value: %.10f' % stats.binom_test(22, 50, prob, "less"))
```

p-value: 0.0000029168

Рис. 4.11: Решение задачи о работниках

```
import pandas as pd
import numpy as np
import scipy

from statsmodels.stats.proportion import proportion_confint

data = pd.read_csv("datasets/banner_click_stat.txt", sep="\t", header=None)
data.columns = ["A", "B"]
data.head()
```

	A	B
0	0	0
1	1	1
2	0	0
3	0	0
4	0	0

```
data.mean()
```

```
A    0.037
B    0.053
dtype: float64
```

Построим доверительные интервалы методом Уилсона:

```
print("A:", proportion_confint(sum(data.A), len(data), method="wilson"))
print("B:", proportion_confint(sum(data.B), len(data), method="wilson"))
```

```
A: (0.026961180875554734, 0.05058239748206931)
B: (0.04074650524859452, 0.06867461683749176)
```

Замечаем, что данные интервалы пересекаются. Если бы интервалы не пересекались, мы бы могли утверждать по результатам нашего эксперимента, что второй баннер лучше первого.

```
def proportions_diff_confint_ind(sample1, sample2, alpha = 0.05):
    z = scipy.stats.norm.ppf(1 - alpha / 2)

    p1 = sum(sample1) / len(sample1)
    p2 = sum(sample2) / len(sample2)

    half_width_of_confint = z * np.sqrt(p1 * (1 - p1) / len(sample1) +
                                         p2 * (1 - p2) / len(sample2))
    delta_p = p1 - p2

    left_boundary = delta_p - half_width_of_confint
    right_boundary = delta_p + half_width_of_confint

    return (left_boundary, right_boundary)
```

Рис. 4.12: Задача о баннерах

```

def proportions_diff_z_stat_ind(sample1, sample2):
    n1 = len(sample1)
    n2 = len(sample2)

    p1 = sum(sample1) / n1
    p2 = sum(sample2) / n2

    P = (p1*n1 + p2*n2) / (n1 + n2)

    return (p1 - p2) / np.sqrt(P * (1 - P) * (1. / n1 + 1. / n2))

def proportions_diff_z_test(z_stat, alternative = 'two-sided'):
    if alternative not in ('two-sided', 'less', 'greater'):
        raise ValueError("alternative not recognized\n"
                         "should be 'two-sided', 'less' or 'greater'")

    if alternative == 'two-sided':
        return 2 * (1 - scipy.stats.norm.cdf(np.abs(z_stat)))

    if alternative == 'less':
        return scipy.stats.norm.cdf(z_stat)

    if alternative == 'greater':
        return 1 - scipy.stats.norm.cdf(z_stat)

print("Доверительный интервал для разности двух долей: [%f, %f]"
      % proportions_diff_confint_ind(data.A, data.B))
print("p-value: %f"
      % proportions_diff_z_test(proportions_diff_z_stat_ind(data.A, data.B)))

```

Доверительный интервал для разности двух долей: [-0.034157, 0.002157]  
p-value: 0.084379

На уровне значимости 0.05 мы не можем отвергнуть нулевую гипотезу о значимости различий между двумя баннерами.

Рис. 4.13: Задача о баннерах (окончание)

```
import numpy as np
import scipy

from statsmodels.stats.proportion import proportion_confint

data_exp = np.array([1 if i < 10 else 0 for i in range(34)])
data_ctrl = np.array([1 if i < 4 else 0 for i in range(16)])

print('Среднее значение в экспериментальной группе: %.4f' % data_exp.mean())
print('Среднее значение в контрольной группе: %.4f' % data_ctrl.mean())
```

Среднее значение в экспериментальной группе: 0.2941  
Среднее значение в контрольной группе: 0.2500

```
conf_interval_banner_exp = proportion_confint(np.sum(data_exp),
                                               len(data_exp),
                                               method = 'wilson')
conf_interval_banner_ctrl = proportion_confint(np.sum(data_ctrl),
                                                len(data_ctrl),
                                                method = 'wilson')
```

```
print('95% доверительный интервал для экспериментальной группы: [%f, %f]' %
      conf_interval_banner_exp)
print('95% доверительный интервал для контрольной группы: [%f, %f]' %
      conf_interval_banner_ctrl)
```

95% доверительный интервал для экспериментальной группы: [0.168346, 0.461689]  
95% доверительный интервал для контрольной группы: [0.101821, 0.494983]

```
print('95% доверительный интервал между разностями двух долей: [%f, %f]' %
      proportions_diff_confint_ind(data_exp, data_ctrl))
print('p-value: %.4f' %
      proportions_diff_z_test(proportions_diff_z_stat_ind(data_exp, data_ctrl),
                              'two-sided'))
```

95% доверительный интервал между разностями двух долей: [-0.2176, 0.3058]  
p-value: 0.7459

Отличия в долях зевающих в контрольной и экспериментальной группах статистически не значимо.

Рис. 4.14: Задача о зевоте

```
import pandas as pd
import numpy as np
import scipy

from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
```

```
banknotes_data = pd.read_table('datasets/banknotes.txt')
banknotes_data.head()
```

	X1	X2	X3	X4	X5	X6	real
0	214.8	131.0	131.1	9.0	9.7	141.0	1
1	214.6	129.7	129.7	8.1	9.5	141.7	1
2	214.8	129.7	129.7	8.7	9.6	142.2	1
3	214.8	129.7	129.6	7.5	10.4	142.0	1
4	215.0	129.6	129.7	10.4	7.7	141.8	1

```
banknotes_train, banknotes_test = train_test_split(banknotes_data,
                                                    test_size=50)
print(banknotes_train.shape, banknotes_test.shape)
```

$$(-150, 7) \quad (50, 7)$$

```
features1 = ["X1", "X2", "X3"]
features2 = ["X4", "X5", "X6"]

logreg1 = LogisticRegression()
logreg1.fit(banknotes_train[features1].values, banknotes_train['real'].values)

logreg2 = LogisticRegression()
logreg2.fit(banknotes_train[features2].values, banknotes_train['real'].values)
```

```

pred1 = logreg1.predict(banknotes_test[features1])
print('Процент ошибок (1): %f'
      % (1 - accuracy_score(banknotes_test['real'].values, pred1)))
err1 = np.array((banknotes_test.real == pred1).astype("int8"))
print(err1)

pred2 = logreg2.predict(banknotes_test[features2])
print('Процент ошибок (2): %f'
      % (1 - accuracy_score(banknotes_test['real'].values, pred2)))
err2 = np.array((banknotes_test.real == pred2).astype("int8"))
print(err2)

```

Рис. 4.15: Задача о банкнотах

```

def proportions_diff_confint_rel(sample1, sample2, alpha = 0.05):
    z = scipy.stats.norm.ppf(1 - alpha / 2)
    n = len(sample1)

    f = sum([1 if (x[0] == 1 and x[1] == 0) else 0
             for x in zip(sample1, sample2)])
    g = sum([1 if (x[0] == 0 and x[1] == 1) else 0
             for x in zip(sample1, sample2)])

    center_of_confint = (f - g) / n
    half_width_of_confint = z * np.sqrt((f + g) / n**2 - (f - g)**2 / n**3)

    left_boundary = center_of_confint - half_width_of_confint
    right_boundary = center_of_confint + half_width_of_confint

    return left_boundary, right_boundary

def proportions_diff_z_stat_rel(sample1, sample2):
    n = len(sample1)

    f = sum([1 if (x[0] == 1 and x[1] == 0) else 0
             for x in zip(sample1, sample2)])
    g = sum([1 if (x[0] == 0 and x[1] == 1) else 0
             for x in zip(sample1, sample2)])

    return (f - g) / np.sqrt(f + g - ((f - g)**2) / n)

def proportions_diff_z_test(z_stat, alternative = 'two-sided'):
    if alternative not in ('two-sided', 'less', 'greater'):
        raise ValueError("alternative not recognized\n"
                         "should be 'two-sided', 'less' or 'greater'")

    if alternative == 'two-sided':
        return 2 * (1 - scipy.stats.norm.cdf(np.abs(z_stat)))

    if alternative == 'less':
        return scipy.stats.norm.cdf(z_stat)

    if alternative == 'greater':
        return 1 - scipy.stats.norm.cdf(z_stat)

print('95% доверительный интервал для разности двух долей ' +
      'в связанных выборках: \n[%4f, %4f]' %
      proportions_diff_confint_rel(err1, err2))
print('p-value: %f' %
      proportions_diff_z_test(proportions_diff_z_stat_rel(err1, err2)))

```

95% доверительный интервал для разности двух долей в связанных выборках:  
[-0.2510, -0.0290]  
p-value: 0.013444

Рис. 4.16: Задача о банкнотах (окончание)

```

from scipy import stats
from statsmodels.stats.proportion import proportion_confint

prob = 0.1
k = 11
n = 12

print('p-value: %.12f' % stats.binom_test(k, n, prob, alternative="greater"))

p-value: 0.00000000109

conf_interval = proportion_confint(k, n, method = 'wilson')
print(conf_interval)

(0.646120088858883, 0.9851349055950829)

k = 7
print('p-value: %.8f' % stats.binom_test(k, n, prob, alternative="greater"))

p-value: 0.00005018

A = np.array([1 if i < 11 else 0 for i in range(12)])
B = np.array([1 if i < 7 else 0 for i in range(12)])
print("Доверительный интервал для разности двух долей: [%f, %f]"
      % proportions_diff_confint_ind(A, B))
print("p-value: %f"
      % proportions_diff_z_test(proportions_diff_z_stat_ind(A, B)))

```

Доверительный интервал для разности двух долей: [0.013551, 0.653116]  
p-value: 0.059346

На уровне значимости 0.05 мы не можем утверждать, что люди знают тыльную сторону ладони лучше, чем ладонь.

Рис. 4.17: Задача о ладонях

# Глава 5

## Непараметрические критерии

Непараметрические критерии не ожидают, что выборка имеет некоторое определенное распределение. Данные критерии работают с произвольным распределением случайных величин.

$$X^n = (X_1, \dots, X_n), \quad X \sim F(x)$$

Основные подходы непараметрических методов:

- Выполнить преобразование исходного распределения  $F(x)$  в некоторое другое, с которым мы можем работать.
- Выполнить некоторые предположения над  $F(x)$ , которые позволяют применить метод.

### 5.1 Критерий знаков

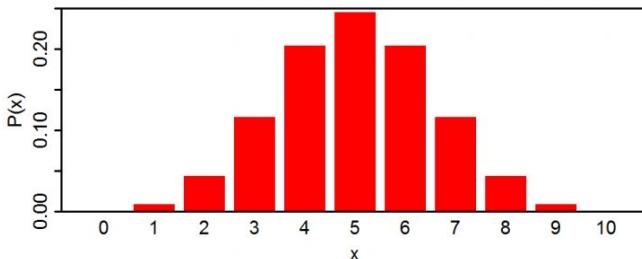
Критерий знаков - самый простой непараметрический метод, который затрагивает вопросы среднего. Главный недостаток - отбрасывание большого количества информации из выборки. Выборка трансформируется в бинарный вектор.

выборка: нулевая гипотеза: альтернативная гипотеза:  статистика:  нулевое распределение	$X^n = (X_1, \dots, X_n), X_i \neq m_0$ $H_0 : med(X) = m_0$ $H_1 : med(X) < m_0$ или $med(X) \neq m_0$ или $med(X) > m_0$ $T(X^n) = \sum_{i=1}^n [X_i > m_0]$ $T(X^n) \sim Bin(n, \frac{1}{2})$
---	---

Таблица 5.1: Описание критерия знаков для одной выборки

### 5.1.1 Критерий знаков для одной выборки

Идея данного критерия очень проста. Из определения медианы следует, что она делит выборку на две части: все значения из первой половины меньше или равны медиане, значения из второй половины - больше или равны ей. Количество элементов в каждой половине либо равно, либо отличается на единицу. Если  $m_0$  - медиана, то элемент выборки равновероятно может реализоваться как в первой половине ( $\leq m_0$ ), так и во второй ( $\geq m_0$ ). Количество элементов, реализовавшихся по одному из сторон, будет представлять значение статистики. Очевидно, что биномиальное распределение является нулевым распределением данного критерия (рисунок 5.1). Описание критерия представлено в таблице 5.1.

Рис. 5.1: Биномиальное распределение  $Bin(10, 0.5)$ 

В модуле `statsmodels.stats.descriptivestats` описана функция `sign_test`.

### 5.1.2 Критерий знаков для связанных выборок

выборки:	$X_1^n = (X_{11}, \dots, X_{1n})$
	$X_2^n = (X_{21}, \dots, X_{2n})$
	$X_{1i} \neq X_{2i}$
нулевая гипотеза:	$H_0 : P(X_1 > X_2) = \frac{1}{2}$
альтернативная гипотеза:	$H_1 : P(X_1 > X_2) < \neq > \frac{1}{2}$
статистика:	$T(X_1^n, X_2^n) = \sum_{i=1}^n [X_{1i} > X_{2i}]$
нулевое распределение	$T(X_1^n, X_2^n) \sim Bin(n, \frac{1}{2})$

Таблица 5.2: Описание критерия знаков для двух связанных выборок

Важно сказать, что под «средним» непараметрические критерии часто понимают несколько разные вещи. Так, одновыборочный критерий знаков под средним понимает медиану. Двухвыборочный критерий знаков гипотезу о средних формулирует в представленном выше экзотическом виде.

Реализация данного метода может быть выполнена через критерий знаков для одной выборки: `sign_test(X1 - X2, mu0=0)`.

Описание критерия представлено в таблице 5.2.

## 5.2 Ранговые критерии

Одним из недостатков знаковых критериев является то, что большая часть исходной информации откидывается. Ранговые критерии сохраняют больше информации. Вначале мы преобразуем выборку в вариационный ряд (по неубыванию). Очевидно, что могут попасться элементы с одинаковым значением. Равные элементы вариационного ряда образуют связку:

$$X_{(1)}, \leq \dots < \underbrace{X_{k_1} = \dots = X_{k_2}}_{k_2 - k_1 + 1} < \dots \leq X_{(n)}$$

**Рангом наблюдения**  $rank(X_i)$  называется позиция наблюдения в вариационном ряду.

Если  $X_i$  оказывается в связке  $X_{(k_1)}, \dots, X_{(k_2)}$ , то все элементы в связке получат одинаковый средний ранг:

$$rank(X_i) = \frac{k_1 + k_2}{2}$$

иначе позиция элемента в вариационном ряду и будет являться его рангом.

### 5.2.1 Критерий ранговых знаков Уилкоксона

Описание рангового критерия Уилкоксона представлено в таблице 5.3.

Реализация критерия имеется в модуле `scipy.stats: wilcoxon( $X_1 - m_0$ )`.

выборка: нулевая гипотеза: альтернативная гипотеза:  статистика:  нулевое распределение	$X^n = (X_1, \dots, X_n), X_i \neq m_0$ $F_X$ симметрично относительно медианы $H_0 : med(X) = m_0$ $H_1 : med(X) < m_0$ или $med(X) \neq m_0$ или $med(X) > m_0$ $W(X^n) = \sum_{i=1}^n rank( X_i - m_0 ) \cdot sign(X_i - m_0)$ табличное для выборки размера $n > 20$ может быть аппроксимировано: $W \approx N \left( 0, \frac{n(n+1)(2n+1)}{6} \right)$
---	---

Таблица 5.3: Описание критерия ранговых знаков Уилкоксона

### 5.2.2 Критерий ранговых знаков для независимых выборок (Критерий Манна - Уитни - Уилкоксона)

Описание критерия Манна-Уитни представлено в таблице 5.4.

выборки: нулевая гипотеза: альтернативная гипотеза:  статистика:  нулевое распределение	$X_1^{n_1} = (X_{11}, \dots, X_{1n_1})$ $X_2^{n_2} = (X_{21}, \dots, X_{2n_2})$ $H_0 : F_{X_1}(x) = F_{X_2}(x)$ $H_1 : F_{X_1}(x) = F_{X_2}(x + \Delta), \Delta <= 0$ $X_{(1)} \leq \dots \leq X_{(n_1+n_2)}$ вариационный ряд объединенной выборки $R_1(X_1^{n_1}, X_2^{n_2}) = \sum_{i=1}^{n_1} rank(X_{1i})$ табличное
---	---

Таблица 5.4: Описание критерия ранговых знаков для связанных выборок

Реализация критерия в модуле `scipy.stats: mannwhitneyu( $X_1, X_2$ )`.

### 5.2.3 Критерий ранговых знаков для связанных выборок

Идея, которая лежит в основе критерия ранговых знаков для связанных выборок проста: необходимо из каждого элемента первой выборки вычесть соответствующий элемент второй выборки, и для полученной выборки применить критерий ранговых знаков Уилкоксона с проверкой на то, что медиана полученной выборки равна нулю.

выборки:	$X_1^n = (X_{11} : \dots, X_{1n})$ $X_2^n = (X_{21} : \dots, X_{2n})$ $X_{1i} \neq X_{1j}$
нулевая гипотеза:	$H_0 : med(X_1 - X_2) = 0$
альтернативная гипотеза:	$H_1 : med(X_1 - X_2) < 0$ или $med(X_1 - X_2) \neq 0$ или $med(X_1 - X_2) > 0$
статистика:	$W(X^n) = \sum_{i=1}^n rank( X_{1i} - X_{2i} ) \cdot sign(X_{1i} - X_{2i})$
нулевое распределение	табличное

Таблица 5.5: Описание критерия ранговых знаков для связанных выборок

Описание критерия - в таблице 5.5.

Реализация двухвыборочного критерия через одновыборочный: `scipy.stats.wilcoxon( $X_2 - X_1$ )`.

## 5.3 Перестановочные критерии

Перестановочные критерии работают аналогично ранговым, однако не делают никаких преобразований исходных значений в ранги. Стоит отметить, что если множество перестановок  $G$  (из которого формируется нулевое распределение) слишком велико, для оценки нулевого распределения  $T$  достаточно взять случайное его подмножество.

### 5.3.1 Одновыборочный перестановочный критерий

Описание критерия представлено в таблице 5.6.

выборка:	$X^n = (X_1, \dots, X_n)$
	$F_X$ симметрично относительно матожидания
нулевая гипотеза:	$H_0 : \mathbb{E}X = m_0$
альтернативная гипотеза:	$H_1 : \mathbb{E}(X) < m_0$ или $\mathbb{E}(X) \neq m_0$
статистика:	или $\mathbb{E}(X) > m_0$
нулевое распределение	$T(X^n) = \sum_{i=1}^n (X_i - m_0)$ порождается перебором $2^n$ знаков перед слагаемыми $X_i - m_0$

Таблица 5.6: Описание одновыборочного перестановочного критерия

### 5.3.2 Двухвыборочный перестановочный критерий для независимых выборок

Описание критерия представлено в таблице 5.7.

### 5.3.3 Двухвыборочный перестановочный критерий для связанных выборок

Описание критерия представлено в таблице 5.8.

выборки:	$X_1^{n_1} = (X_{11}, \dots, X_{1n_1})$
нулевая гипотеза:	$X_2^{n_2} = (X_{21}, \dots, X_{2n_2})$
альтернативная гипотеза:	$H_0 : F_{X_1}(x) = F_{X_2}(x)$
статистика:	$H_1 : F_{X_1}(x) = F_{X_2}(x + \Delta), \Delta <= 0$
нулевое распределение:	$T(X_1^{n_1}, X_2^{n_2}) = \frac{1}{n_1} \sum_{i=1}^{n_1} X_{1i} - \frac{1}{n_2} \sum_{i=1}^{n_2} X_{2i}$ порождается перебором $C_{n_1+n_2}^{n_1}$ размещений объединенной выборки

Таблица 5.7: Описание двухвыборочного перестановочного критерия для независимых выборок

выборки:	$X_1^n = (X_{11}, \dots, X_{1n})$
нулевая гипотеза:	$X_2^n = (X_{21}, \dots, X_{2n})$
альтернативная гипотеза:	выборки связанные, распределение попарных разностей симметрично
статистика:	$H_0 : \mathbb{E}(X_1 - X_2) = 0$
нулевое распределение	$H_1 : \mathbb{E}(X_1 - X_2) < 0$ или $\mathbb{E}(X_1 - X_2) \neq 0$ или $\mathbb{E}(X_1 - X_2) > 0$
	$D_i = X_{1i} - X_{2i}$
	$T(X_1^n, X_2^n) = \sum_{i=1}^n D_i$
	порождается перебором $2^n$ знаков перед слагаемыми $D_i$

Таблица 5.8: Описание двухвыборочного перестановочного критерия для связанных выборок

## 5.4 Задачи

Задача №1. В текстовом файле mirror\_mouses.txt хранится информация о 16 лабораторных мышах, которые были помещены в двухкомнатные клетки, в одной из комнат висело зеркало. Измерялось отношение между временем, которое мыши проводили в комнате с зеркалами к общему времени. Требовалось установить, существуют ли у мышей какие-то предпочтения по поводу зеркал. Решение на рисунке 5.2.

Задача №2. В файле weight.txt имеется информация об исследовании, в котором оценивается эффективность поведенческой терапии для лечения анорексии. Для 50 пациентов известен вес до начала терапии и по её окончании. Была ли терапия эффективной? Решение на рисунках 5.3 – 5.4.

Задача №3. В файле seattle.txt имеются данные о продажной стоимости недвижимости в Сиэтле для 50 сделок в 2001 году и 50 в 2002. Изменились ли в среднем цены? Решение на рисунках 5.5 – 5.7.

Задача №4. Есть данные о выживаемости пациентов с лейкоцитарной лимфомой:

$$49, 58, 75, 110, 112, 132, 151, 276, 281, 362*$$

Измерено остаточное время жизни с момента начала наблюдения (в неделях); звёздочка обозначает цензурирование сверху — исследование длилось 7 лет, и остаточное время жизни одного пациента, который дожил до конца наблюдения, неизвестно.

$$H_0 : \text{med}X = 200$$

Критерием знаковых рангов необходимо проверить гипотезу  $H_0$  против двусторонней альтернативы:

$$H_1 : \text{med}X \neq 200$$

Подробное решение на рисунке 5.8.

Задача №5. В ходе исследования влияния лесозаготовки на биоразнообразие лесов острова Борнео собраны данные о количестве видов деревьев в 12 лесах, где вырубка не ведётся:

22, 22, 15, 13, 19, 19, 18, 20, 21, 13, 13, 15

и в 9 лесах, где идёт вырубка:

17, 18, 18, 15, 12, 4, 14, 15, 10

Необходимо проверить гипотезу о равенстве среднего количества видов в двух типах лесов против односторонней альтернативы о снижении биоразнообразия в вырубаемых лесах посредством рангового критерия. Решение на рисунке 5.9.

Задача №6. 28 января 1986 года космический шаттл «Челленджер» взорвался при взлёте. Семь астронавтов, находившихся на борту, погибли. В ходе расследования причин катастрофы основной версией была неполадка с резиновыми уплотнительными кольцами в соединении с ракетными ускорителями. Для 23 предшествовавших катастрофе полётов «Челленджера» известны температура воздуха и появление повреждений хотя бы у одного из уплотнительных колец.

С помощью бутстрэпа постройте 95% доверительный интервал для разности средних температур воздуха при запусках, когда уплотнительные кольца повреждались, и запусках, когда повреждений не было. Необходимо проверить гипотезу об одинаковой средней температуре воздуха в дни, когда уплотнительный кольца повреждались, и в дни, когда повреждений не было. Решение на рисунках 5.10 – 5.11.

Задача №7. Предполагается, что стоимость материала, получаемого при переработке строительной конструкции, составляет в среднем 0.28 долларов. Взята случайная выборка из 10 конструкций, все они переработаны; стоимость в долларах полученного из каждой конструкции материала составила

0.28, 0.18, 0.24, 0.30, 0.40, 0.36, 0.15, 0.42, 0.23, 0.48

Изменилось ли средняя стоимость? Решение на рисунке 5.12.

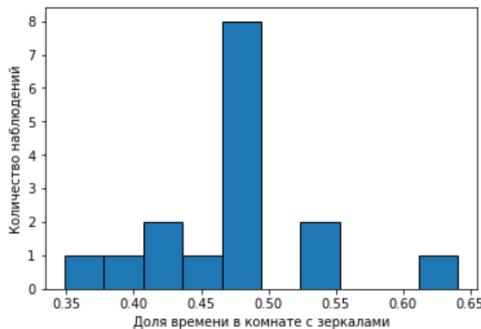
```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as stats
from statsmodels.stats.descriptivestats import sign_test

mouses_data = pd.read_csv("datasets/mirror_mouses.txt", header=None)
mouses_data.columns = ["time"]
mouses_data.head()
```

♦ time ♦

0	0.348471
1	0.640620
2	0.549818
3	0.537454
4	0.400444

```
plt.hist(mouses_data.time, edgecolor="k")
plt.xlabel("Доля времени в комнате с зеркалами")
plt.ylabel("Количество наблюдений")
plt.show()
```



Критерий знаков:

```
sign_test(mouses_data.time, 0.5)
(-5.0, 0.021270751953125)
```

Критерий ранговых знаков:

```
m0 = 0.5
stats.wilcoxon(mouses_data.time - m0)

WilcoxonResult(statistic=35.0, pvalue=0.08793560714236243)
```

Критерий знаков отвергает гипотезу о равенстве средних, а критерий ранговых знаков не отвергает нулевую гипотезу на уровне 0.05.

Рис. 5.2: Задача о мышах

```

import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as stats
from statsmodels.stats.descriptivestats import sign_test

anorexia_data = pd.read_csv("datasets/weight.txt", sep="\t")
anorexia_data.head()

```

	Before	After
0	80.5	82.2
1	84.9	85.6
2	81.5	81.4
3	82.6	81.9
4	79.9	76.4

Данные выборки являются связанными.

```

fig, ax = plt.subplots(nrows=2, ncols=1, sharex=True, figsize=(11, 6))

ax[0].hist(anorexia_data["Before"], edgecolor="k")
ax[0].set_title("До:")

ax[1].hist(anorexia_data["After"], edgecolor="k")
ax[1].set_title("После:")

plt.show()

```

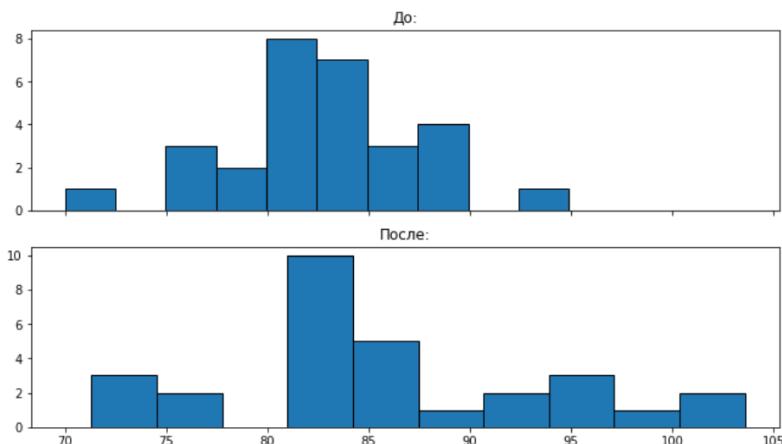


Рис. 5.3: Задача об анорексии

$$H_0 : P(X_1 > X_2) = \frac{1}{2} \quad H_1 : P(X_1 > X_2) \neq \frac{1}{2}$$

```
sign_test(anorexia_data.After - anorexia_data.Before, 0)
(3.5, 0.26493089646101)
```

Критерий знаков не отвергает нулевую гипотезу на уровне значимости 0.05

```
stats.wilcoxon(anorexia_data.After - anorexia_data.Before)
WilcoxonResult(statistic=131.5, pvalue=0.06291972262602667)
```

Критерий ранговых знаков для связанных выборок также не отвергает гипотезу на уровне значимости 0.05.

Рис. 5.4: Задача об анорексии (окончание)

```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as stats
from statsmodels.stats.descriptivestats import sign_test

houses_data = pd.read_csv("datasets/seattle.txt", sep="\t")
houses_data.head()
```

◆ Price ◆ Year ◆

0	142.0	2002
1	232.0	2002
2	132.5	2002
3	200.0	2002
4	362.0	2002

```
price2001 = houses_data[houses_data.Year == 2001].Price
price2002 = houses_data[houses_data.Year == 2002].Price
```

```
fig, ax = plt.subplots(nrows=2, ncols=1, sharex=True, figsize=(11, 8))

ax[0].hist(price2001, edgecolor="k", bins=20)
ax[0].set_title("До:")

ax[1].hist(price2002, edgecolor="k", bins=20)
ax[1].set_title("После:")

plt.show()
```

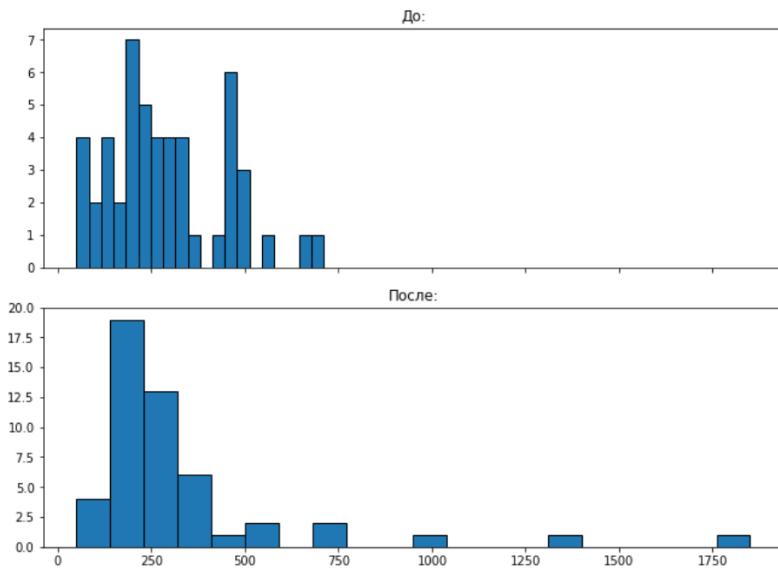


Рис. 5.5: Задача о нежвижимости

Выборки не являются связанными.

Критерий Манна-Уитни:

```
stats.mannwhitneyu(price2001, price2002)
MannwhitneyuResult(statistic=1204.5, pvalue=0.3781936337850874)
```

Получим нулевое распределение статистики:

```
def permutation_t_stat_ind(sample1, sample2):
    return np.mean(sample1) - np.mean(sample2)

def get_random_combinations(n1, n2, max_combinations):
    index = list(range(n1 + n2))
    indices = set([tuple(index)])
    for i in range(max_combinations - 1):
        np.random.shuffle(index)
        indices.add(tuple(index))
    return [(index[:n1], index[n1:]) for index in indices]

def permutation_zero_dist_ind(sample1, sample2, max_combinations = None):
    joined_sample = np.hstack((sample1, sample2))
    n1 = len(sample1)
    n = len(joined_sample)

    if max_combinations:
        indices = get_random_combinations(n1, len(sample2), max_combinations)
    else:
        indices = [(list(index), filter(lambda i: i not in index, range(n))) \
                   for index in itertools.combinations(range(n), n1)]

    distr = [joined_sample[list(i[0])].mean() -
              joined_sample[list(i[1])].mean() \
              for i in indices]
    return distr
```

```
plt.hist(permutation_zero_dist_ind(price2001, price2002,
                                    max_combinations = 5000), edgecolor='k')
plt.grid()
plt.show()
```

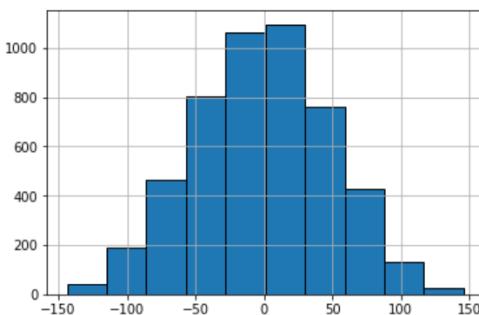


Рис. 5.6: Задача о недвижимости (продолжение)

```

def permutation_test(sample, mean, max_permutations = None, alternative =
                     'two-sided'):
    if alternative not in ('two-sided', 'less', 'greater'):
        raise ValueError("alternative not recognized\n"
                         "should be 'two-sided', 'less' or 'greater'")

    t_stat = permutation_t_stat_ind(sample, mean)

    zero_distr = permutation_zero_dist_ind(sample, mean, max_permutations)

    if alternative == 'two-sided':
        s = sum([1. if abs(x) >= abs(t_stat) else 0. for x in zero_distr])
        return s / len(zero_distr)

    if alternative == 'less':
        s = sum([1. if x <= t_stat else 0. for x in zero_distr])
        return s / len(zero_distr)

    if alternative == 'greater':
        s = sum([1. if x >= t_stat else 0. for x in zero_distr])
        return s / len(zero_distr)

print("p-value =", permutation_test(price2001, price2002,
                                      max_permutations = 10000))

```

p-value = 0.4476

Нулевая гипотеза о равенстве средних в обоих случаях не отвергается.

Рис. 5.7: Задача о недвижимости (окончание)

```

import numpy as np
import scipy.stats as stats

surv_data = np.array([49, 58, 75, 110, 112, 132, 151, 276, 281, 362])

```

Критерием знаковых рангов проверим гипотезу  $H_0$  против двусторонней альтернативы.

$$H_0 : \text{med } X = 200 \quad H_1 : \text{med } X \neq 200$$

```

print("Точечная оценка:", np.median(surv_data))
medX = 200
print(stats.wilcoxon(surv_data - medX))

```

Точечная оценка: 122.0  
WilcoxonResult(statistic=17.0, pvalue=0.2845026979112075)

На уровне значимости 0.05 нулевая гипотеза не отвергается.

Рис. 5.8: Задача о выживаемости

```
import numpy as np
import scipy.stats as stats

forest_not_cut = np.array([22, 22, 15, 13, 19, 19, 18, 20, 21, 13, 13, 15])
print("Медиана по первому лесу (без вырубки):", np.median(forest_not_cut))
forest_cut = np.array([17, 18, 18, 15, 12, 4, 14, 15, 10])
print("Медиана по второму лесу (с вырубкой):", np.median(forest_cut))
```

Медиана по первому лесу (без вырубки): 18.5

Медиана по второму лесу (с вырубкой): 15.0

```
stats.mannwhitneyu(forest_cut, forest_not_cut, alternative='less')
```

```
MannwhitneyResult(statistic=27.0, pvalue=0.02900499272087373)
```

Гипотеза о равенстве средних отвергается на уровне значимости 0.05.

Рис. 5.9: Задача о видовом разнообразии

```
import numpy as np
import scipy.stats as stats

challenger = pd.read_csv('datasets/challenger.txt', delimiter='\t')
challenger.rename(columns={'Unnamed: 0': 'Date'}, inplace=True)
challenger.head()
```

	Date	Temperature	Incident
0	Apr12.81	18.9	0
1	Nov12.81	21.1	1
2	Mar22.82	20.6	0
3	Nov11.82	20.0	0
4	Apr04.83	19.4	0

```
challenger_broken = challenger[challenger.Incident > 0]
challenger_not_broken = challenger[challenger.Incident == 0]

fig, ax = plt.subplots(nrows=2, ncols=1, sharex=True, figsize=(11, 8))

ax[0].hist(challenger_not_broken.Temperature, color="green",
            edgecolor="k", bins=20)
ax[0].set_title("Без повреждений")

ax[1].hist(challenger_broken.Temperature, color="red",
            edgecolor="k", bins=20)
ax[1].set_title("С повреждениями")

plt.show()
```

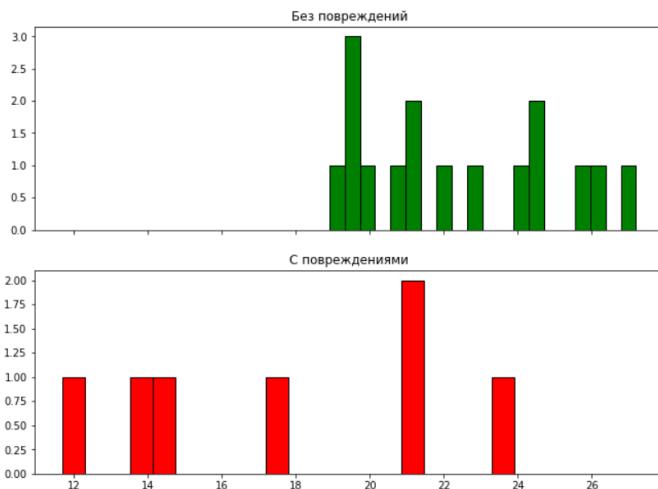


Рис. 5.10: Задача о шаттле

```

def get_bootstrap_samples(data, n_samples):
    indices = np.random.randint(0, len(data), (n_samples, len(data)))
    samples = data[indices]
    return samples

def stat_intervals(stat, alpha):
    boundaries = np.percentile(stat,
                               [100 * alpha / 2., 100 * (1 - alpha / 2.)])
    return boundaries

```

```

challenger_broken_bs_mean = \
    np.array(list(map(np.mean,
                      get_bootstrap_samples(challenger_broken.Temperature.values, 1000)))) 

challenger_not_broken_bs_mean = \
    np.array(list(map(np.mean,
                      get_bootstrap_samples(challenger_not_broken.Temperature.values, 1000)))) 

print('95% доверительный интервал для разности средних температур воздуха при запусках, +' +
      'когда уплотнительные кольца повреждались, и запусках, когда повреждений не было: %s' %
      str(stat_intervals(challenger_broken_bs_mean - challenger_not_broken_bs_mean, 0.05)))

```

95% доверительный интервал для разности средних температур воздуха при запусках, когда уплотнительные кольца повреждались, и запусках, когда повреждений не было: [-8.10417411 -0.99292411]

```

np.random.seed(0)
print('p-value: %.4f' %
      permutation_test(challenger_broken.Temperature.values,
                        challenger_not_broken.Temperature.values,
                        max_permutations=10000))

```

p-value: 0.0057

Рис. 5.11: Задача о шаттле (окончание)

```

import numpy as np
import scipy.stats as stats

X = np.array([0.28, 0.18, 0.24, 0.30, 0.40, 0.36, 0.15, 0.42, 0.23, 0.48])

stats.wilcoxon(X - 0.28)

WilcoxonResult(statistic=17.0, pvalue=0.5146697234497355)

```

Рис. 5.12: Задача о стоимости материала



# Глава 6

## Анализ зависимостей

### 6.1 Непрерывные случайные величины

**Корреляция** — статистическая взаимосвязь двух или более случайных величин, которая сама по себе не является достаточным условием причинно-следственной связи.

При желании, можно найти значимую корреляцию между двумя на первый взгляд несвязанными случайными величинами (рисунки 6.1 – 6.4).

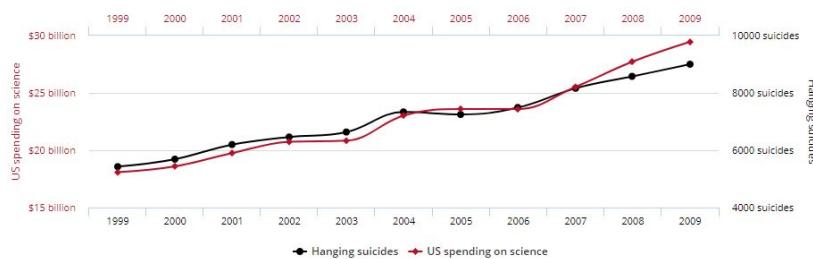


Рис. 6.1: Траты на науку (США) / Число самоубийств (США)

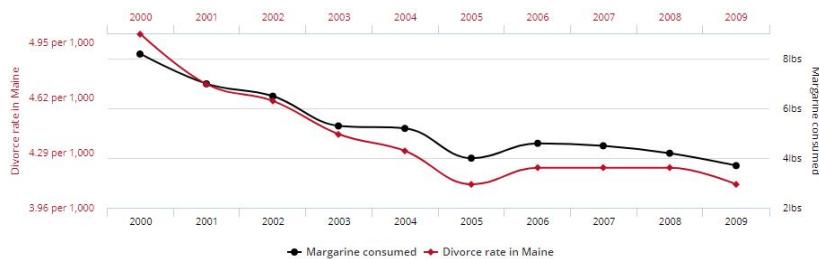


Рис. 6.2: Число разводов в штате Мэн / Потребление маргарина на душу населения

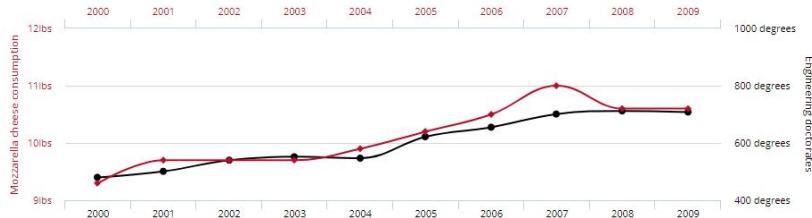


Рис. 6.3: Потребление сыра Моцареллы / Число докторских степеней по инженерии

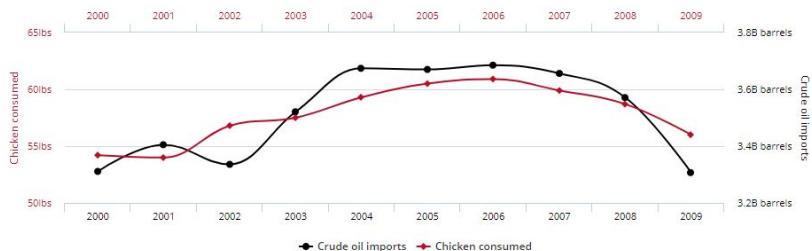


Рис. 6.4: Импорт нефти (США) / Потребление курицы на душу населения

### 6.1.1 Корреляция Пирсона

**Корреляция Пирсона** — это мера силы линейной взаимосвязи между двумя случайными величинами  $X$  и  $Y$ :

$$r_{X,Y} = \frac{\text{cov}(X, Y)}{\sqrt{\mathbb{D}X\mathbb{D}Y}} = \frac{\mathbb{E}((X - \mathbb{E}X)(Y - \mathbb{E}Y))}{\sqrt{\mathbb{D}X\mathbb{D}Y}}, \quad r_{X,Y} \in [-1, 1]$$

Выборочный коэффициент корреляции Пирсона:

$$\hat{r}_{X,Y} = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2 \sum_{i=1}^n (Y_i - \bar{Y})^2}}$$

Значения коэффициента корреляции Пирсона для различных распределений случайных величин представлены на рисунке 6.5.

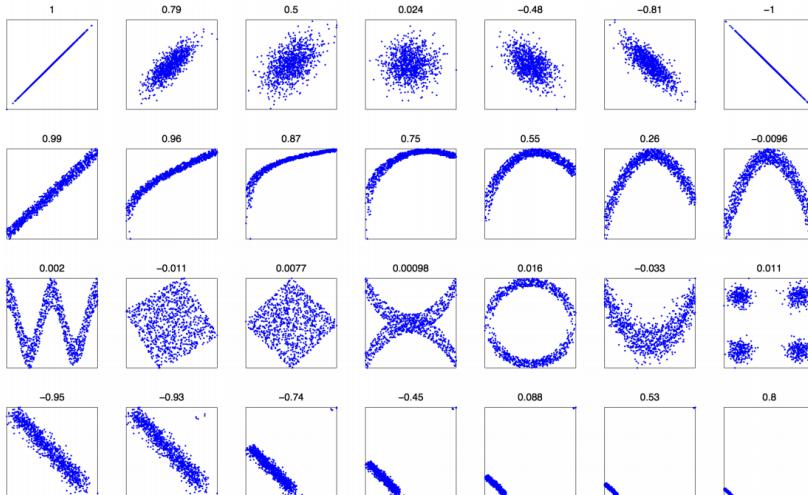


Рис. 6.5: Значения коэффициента корреляции Пирсона для различных распределений случайных величин

Недостатки:

- Служит мерой только линейной взаимосвязи;
- Неустойчив к выбросам.

выборки: нулевая гипотеза: альтернативная гипотеза: статистика: нулевое распределение	$X_1^n = (X_{11}, \dots, X_{1n})$ $X_2^n = (X_{21}, \dots, X_{2n})$ выборки связанные $H_0 : r_{X_1 X_2} = 0$ $H_1 : r_{X_1 X_2} < 0$ или $H_1 : r_{X_1 X_2} \neq 0$ или $H_1 : r_{X_1 X_2} > 0$ $T(X_1^n, X_2^n) = \frac{\hat{r}_{X_1 X_2} \sqrt{n - 2}}{\sqrt{1 - \hat{r}_{X_1 X_2}^2}}$ $T(X_1^n, X_2^n) \sim St(n - 2)$
---	--

Таблица 6.1: Описание критерия Стьюдента (значимость коэффициента корреляции Пирсона)

Для проверки значимости коэффициента корреляции Пирсона используется критерий Стьюдента (таблица 6.1).

Доверительный интервал для коэффициента корреляции Пирсона:

$$\left[ \hat{r}_{X_1 X_2} + \frac{t_{n-2, \alpha/2}(1 - \hat{r}_{X_1 X_2}^2)}{\sqrt{n}}, \hat{r}_{X_1 X_2} - \frac{t_{n-2, \alpha/2}(1 - \hat{r}_{X_1 X_2}^2)}{\sqrt{n}} \right]$$

### 6.1.2 Корреляция Спирмена

**Коэффициент корреляции Спирмена** — это мера силы монотонной взаимосвязи между двумя случайными величинами, который равен коэффициенту корреляции Пирсона между рангами наблюдений.

**Выборочный коэффициент корреляции Спирмена** находится по формуле:

$$\hat{\rho}_{X,Y} = \frac{\left( \text{rank}(X_i) - \frac{n+1}{2} \right) \left( \text{rank}(Y_i) - \frac{n+1}{2} \right)}{\frac{1}{12}(n^3 - n)} = 1 - \frac{6}{n^3 - n} \sum_{i=1}^n (X_i - Y_i)^2$$

где  $X_i$  - ранг наблюдения в вариационном ряду  $X'$ ,  $Y_i$  - ранг наблюдения в вариационном ряду  $Y'$ .

Значение коэффициента корреляции Спирмена также лежит в диапозоне  $[-1, 1]$ .

Значения коэффициента корреляции Спирмена для различных распределений случайных величин представлены на рисунке 6.6.

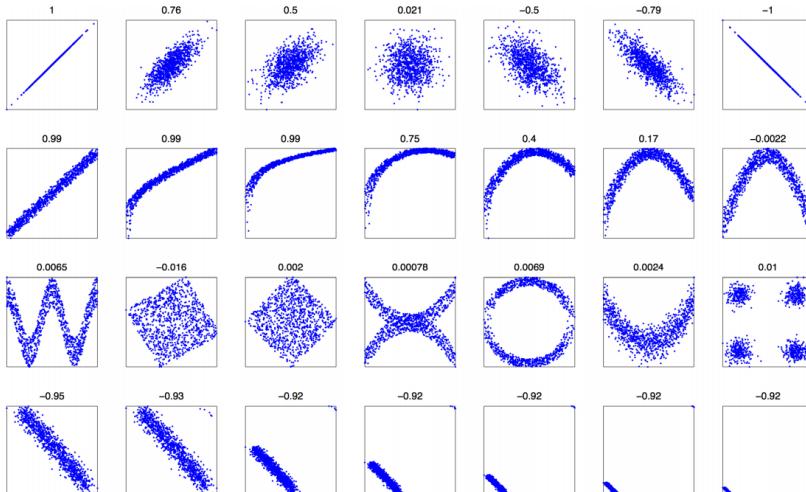


Рис. 6.6: Значения коэффициента корреляции Спирмена на различных распределениях

Для проверки значимости корреляции Спирмена используется

критерий Стьюдента (таблица 6.2).

выборки:	$X_1^n = (X_{11}, \dots, X_{1n})$
	$X_2^n = (X_{21}, \dots, X_{2n})$
	выборки связанные
нулевая гипотеза:	$H_0 : \rho_{X_1 X_2} = 0$
альтернативная гипотеза:	$H_1 : \rho_{X_1 X_2} < 0$ или $H_1 : \rho_{X_1 X_2} \neq 0$ или $H_1 : \rho_{X_1 X_2} > 0$
статистика:	$T(X_1^n, X_2^n) = \frac{\hat{\rho}_{X_1 X_2} \sqrt{n - 2}}{\sqrt{1 - \hat{\rho}_{X_1 X_2}^2}}$
нулевое распределение	$T(X_1^n, X_2^n) \sim St(n - 2)$

Таблица 6.2: Описание критерия Стьюдента (значимость коэффициента корреляции Спирмена)

### 6.1.3 Корреляция Кендалла

**Коэффициент корреляции Кендалла**  $\tau_{XY}$  случайных величин  $X$  и  $Y$  — мера их взаимной неупорядоченности; также оценивает силу монотонной корреляции между величинами:

$$\tau_{XY} = 1 - \frac{4}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=1}^n [[X_i < X_j] \neq [Y_i < Y_j]] = \frac{C - D}{C + D}$$

где  $C$  - число согласованных пар,  $D$  - число несогласованных пар.

Значения коэффициента корреляции Кендалла для различных распределений случайных величин представлены на рисунке 6.7.

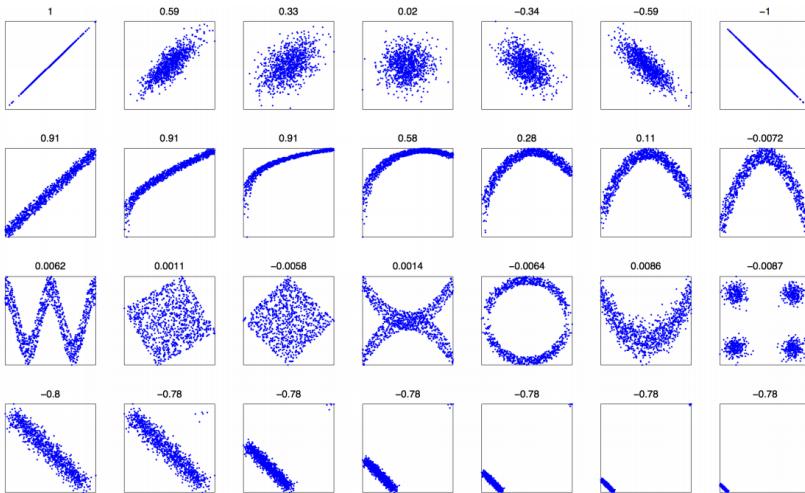


Рис. 6.7: Значения коэффициента корреляции Кендалла на различных распределениях

В сравнении с корреляцией Спирмена, корреляция Кендалла:

- точнее оценивается для выборок небольших объемов;
- обычно меньше по модулю.

## 6.2 Категориальные признаки

Таблица сопряженности для общего случая (таблица 6.3):

$X_1 \backslash X_2$	1	$\dots$	$j$	$\dots$	$K_2$	$\Sigma$
1						
$\vdots$						
i			$n_{ij}$			$n_{i+}$
$\vdots$						
$K_1$						
$\sum$			$n_{+j}$			n

Таблица 6.3: Таблица сопряженности  $K_1 \times K_2$

Введём некоторые обозначения. Пусть первая категориальная переменная принимает значения от 1 до  $K_1$ , вторая - от 1 до  $K_2$ .  $n_{i+}$  - сумма по  $i$ -строке,  $n_{+j}$  - сумма по  $j$ -столбцу.

### 6.2.1 Критерий $\chi^2$

выборки:	$X_1^n = (X_{11}, \dots, X_{1n}), X_1 \in \{1, \dots, K_1\}$
	$X_2^n = (X_{21}, \dots, X_{2n}), X_2 \in \{1, \dots, K_2\}$
нулевая гипотеза:	выборки связанные
альтернативная гипотеза:	$H_0 : X_1$ и $X_2$ независимы
статистика:	$H_1 : H_0$ неверна
нулевое распределение	$\chi^2(X_1^n, X_2^n) = n \left( \sum_{i=1}^{K_1} \sum_{j=1}^{K_2} \frac{n_{ij}^2}{n_{i+} n_{+j}} - 1 \right)$
	$\chi^2(X_1^n, X_2^n) \sim \chi^2_{(K_1-1)(K_2-1)}$

Таблица 6.4: Описание критерия  $\chi^2$

Описание критерия представлено в таблице 6.6.

Условия применимости:

- $n > 40$ ;
- $\frac{n_{i+}n_{+j}}{n} < 5$  не более чем в 20% ячеек.

### 6.2.2 Точный критерий Фишера

Пусть мы имеем следующую таблицу сопряженности (таблица 6.5):

$X_1 \backslash X_2$	0	1	$\sum$
0	$a$	$b$	$a + b$
1	$c$	$d$	$c + d$
$\sum$	$a + c$	$b + d$	n

Таблица 6.5: Таблица сопряженности

выборки: $X_1^n = (X_{11}, \dots, X_{1n})$ , $X_1 \in \{0, 1\}$ $X_2^n = (X_{21}, \dots, X_{2n})$ , $X_2 \in \{0, 1\}$ выборки связанные нулевая гипотеза: $H_0 : X_1$ и $X_2$ независимы альтернативная гипотеза: $H_1 : H_0$ неверна
---

Таблица 6.6: Описание критерия  $\chi^2$

Тогда вероятность появления наблюдаемой таблицы равна:

$$P(X_1^n, X_2^n) = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!}$$

Достигаемый уровень значимости определяется как сумма по всем возможным вариантам таблицы с такими же суммами по строкам и столбцам, имеющим вероятность не более  $P(X_1^n, X_2^n)$ .

### 6.2.3 Корреляция Мэттьюса

**Коэффициент корреляции Мэттьюса** — это мера силы взаимосвязи между двумя бинарными переменными. Таблица сопряженности (таблица 6.5).

$$MCC_{X_1 X_2} = \frac{ad - bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}} \quad MCC_{X_1 X_2} \in [0, 1]$$

Если  $b = c = 0$ , значение коэффициента корреляции равно 1, так как между наборами переменных нет разногласий. В противов-

положном случае  $a = d = 0$ , когда мы наблюдаем одни разногласия, коэффициент корреляции равен  $-1$ .

#### 6.2.4 Коэффициент V Крамера

На основании таблицы сопряженности  $K_1 \times K_2$  можно вычислить **коэффициент V Крамера**.

$$\phi_c(X_1^n, X_2^n) = \sqrt{\frac{\chi^2(X_1^n, X_2^n)}{n(\min K_1, K_2) - 1}}$$

Коэффициент Крамера принимает значения от нуля до единицы. 0 - отсутствие взаимосвязи, 1 - наличие взаимосвязи. Корреляция между двумя категориальными переменными не может быть отрицательной, поскольку уровни категориальных переменных не связаны друг с другом отношениями порядков.

### 6.3 Пары переменных разных видов

Если один признак категориальный, а другой непрерывный - никакой корреляции считать не нужно.

# Глава 7

## Регрессия

### 7.1 Постановка задачи

Пусть имеется некоторый набор из  $n$  объектов, для каждого из которых известно значение  $k$  признаков ( $n > k$ ).  $x_1, \dots, x_k$  - **объясняющие переменные (предикторы, регрессоры, признаки)**. Для каждого из объектов известен некоторый отклик  $y$ . Мы хотели бы найти такую функцию  $f$ , что

$$y \approx f(x_1, \dots, x_k)$$

которая бы минимизировала ошибку прогноза:

$$\arg \min_f \mathbb{E}(y - f(x_1, \dots, x_k))^2 = \mathbb{E}(y|x_1, \dots, x_k)$$

$\mathbb{E}(y|x_1, \dots, x_k) = f(x_1, \dots, x_k)$  - модель регрессии,  $\mathbb{E}(y|x_1, \dots, x_k) = \beta_0 + \sum_{j=1}^k \beta_j x_j$  - модель линейной регрессии.

Задача линейной регрессии заключается в подборе такого вектора  $\beta$ , который бы сводил математическое ожидание ошибки к минимуму.

В матричной форме:

$$X = \begin{pmatrix} x_{10} = 1 & x_{11} & \dots & x_{1k} \\ \vdots & \ddots & & \vdots \\ x_{n0} = 1 & x_{n1} & \dots & x_{nk} \end{pmatrix} \quad y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}$$

Решение задачи методом наименьших квадратов:

$$\|y - X\beta\|_2^2 \rightarrow \min_{\beta}$$

Точное решение может быть найдено аналитически:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

Однако при большом количестве объектов и признаков требуются значительные вычислительные ресурсы, поэтому используются методы численной оптимизации.

Предсказанные значения  $\hat{y}$ :

$$\hat{y} = X(X^T X)^{-1} X^T y$$

Вычислим разброс относительно своего среднего:

$$TSS = \sum_{i=1}^n (y_i - \bar{y})^2$$

Он может быть разделен на две составляющих:

$$TSS = ESS + RSS$$

Одна из частей - **объясненная сумма квадратов**  $ESS$  - это сумма квадратов отклонений предсказанных значений от среднего:

$$ESS = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$$

Вторая часть - **остаточная сумма квадратов**  $RSS$ , — это сумма квадратов отклонений предсказанных  $y$  от их истинных значений:

$$RSS = \sum_{i=1}^n (y_i - \hat{y})^2 = \sum_{i=1}^n \epsilon_i^2$$

Вектор  $\epsilon = y_i - \hat{y}$  называют **вектором ошибки (невязки)**.

Можно составить новую величину, называемой **коэффициентом детерминации**  $R^2$ :

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

Стандартный коэффициент детерминации всегда увеличивается при добавлении регрессоров в модель, поэтому не может быть использован для отбора признаков.

Для сравнения моделей, которые содержат различное число признаков используется **приведенный коэффициент детерминации**:

$$R_a^2 = \frac{ESS/(n - k - 1)}{TSS/(n - 1)} = 1 - (1 - R^2) \frac{n - 1}{n - k - 1}$$

Предположения модели:

- Предполагается, что модель линейна:

$$y = X\beta + \epsilon$$

- Случайность выборки: наблюдения  $(x_i, y_i)$ ,  $i = 1, \dots, n$  независимы.
- Полнота ранга (ни один из признаков не является линейной комбинацией других признаков или константой):

$$\text{rank } X = k + 1$$

- Случайность ошибки:

$$\mathbb{E}(\epsilon|X) = 0$$

- Гомоскетастичность ошибки (разброс ошибок не должен увеличиваться/уменьшаться по продвижению по оси предиктора):

$$\mathbb{D}(\epsilon|x) = \sigma^2$$

Противоположность гомоскетастичности - гетероскетастичность (рисунок 7.1).

- Нормальность распределения ошибок:

$$\epsilon|X \sim N(0, \sigma^2)$$

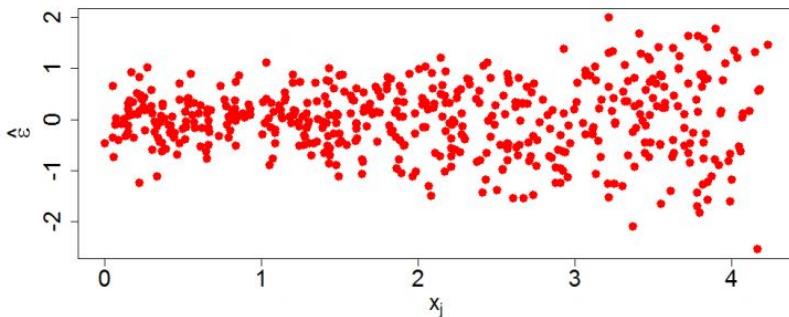


Рис. 7.1: Гетероскетастичность

Проверка предположений:

- Необходимо строить график остатков  $\hat{\epsilon}_i = y_i - \hat{y}_i, i = 1, \dots, n$  от каждого предиктора и оценивать облако точек. Если на графике мы обнаруживаем какую-то функциональную зависимость (например, квадратичную (рисунок 7.2)), то стоит добавить в матрицу  $X$  столбец, соответствующий данной зависимости ( $x_j^2$ ). Если такие зависимости видны, нужно просто добавить в матрицу  $X$  соответствующий столбец.

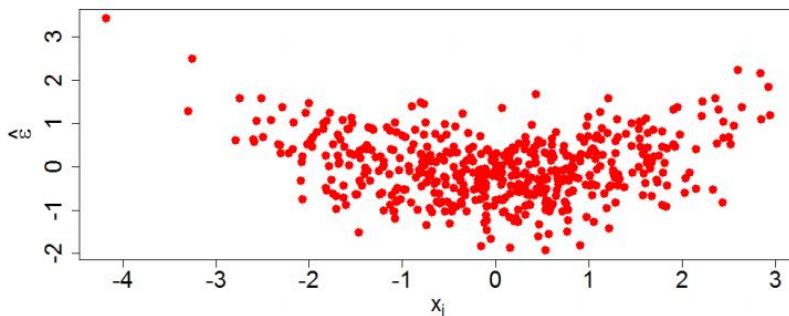


Рис. 7.2: Квадратичная зависимость

- Проверить выполнение о гомоскетастичности данного предположения можно двумя способами:
  - Визуально. Построить графики зависимостей между каждым из предикторов и ошибкой. Если полоса расширяется

ся или сужается с изменением по предиктору, значит мы наблюдаем гетероскедастичность.

- Формально. Критерием Бройша-Пагана.
- Проверка нормальности распределения ошибок может быть проверена критерием Шапиро-Уилка.

## 7.2 Категориальные признаки

Если  $x_j$  - бинарная переменная, она должна быть перекодирована в значения 0 и 1. Например,  $x_j = [\text{пол}=\text{женский}]$ .

Более сложный случай, когда значений больше, чем 2. При построении регрессионной модели нельзя оставлять данных признак в таком виде.

Пусть мы предсказываем уровень заработной платы, в зависимости от фактора «должность», которая является категориальной переменной и принимает 4 значения:

Должность	Значение признака
Грузчик	1
Директор	2
Менеджер	3
Инженер	4

и построили регрессию  $y = \beta_0 + \beta_1 x$ . Тогда значение заработанной платы будет определяться следующим образом:

$$y_{loader} = \beta_0 + \beta_1$$

$$y_{director} = \beta_0 + 2\beta_1$$

$$y_{manager} = \beta_0 + 3\beta_1$$

$$y_{engineer} = \beta_0 + 4\beta_1$$

Если  $\beta_1 > 0$ , тогда инженер будет получать больше, чем директор. Если  $\beta_1 < 0$ , тогда зарплата грузчика будет больше, чем зарплата менеджера или директора. С другой стороны, может прийти идея упорядочить должности согласно здравому смыслу:

Должность	Значение признака
Грузчик	1
Менеджер	2
Инженер	3
Директор	4

Даже если переставить значения таким образом, тогда получится модель, в которой разница между зарплатой менеджера и грузчика будет ровно такой же, как между зарплатой директора и инженера.

Краткий вывод: оставить всё как есть, пытаться переупорядочить - довольно плохая идея, необходимо создавать новые (фиктивные) переменные. Например,

	$x_1$	$x_2$	$x_3$
грузчик	0	0	0
менеджер	1	0	0
инженер	0	1	0
директор	0	0	1

Пусть признак  $x_j$  принимает  $m$  различных значений, тогда для его кодирования необходима  $m - 1$  фиктивная переменная.

### 7.3 Отбор переменных в модель

Есть некоторые статистические критерии, которые позволяют проверить значимость коэффициента линейной регрессии. Описание критерия Стьюдента описано в таблице 7.1.

нулевая гипотеза:	$H_0 : \beta_j = 0$
альтернативная гипотеза:	$H_1 : \beta_j < 0$ или $\beta_j \neq 0$ или $\beta_j > 0$
статистика:	$T = \frac{\hat{\beta}_j}{\sqrt{\frac{RSS}{n - k - 1} (X^T X)_{jj}^{-1}}}$
нулевое распределение	$T \sim St(n - k - 1)$

Таблица 7.1: Описание критерия Стьюдента

Для проверки гипотезы о том, что сразу несколько коэффици-

ентов модели равны 0, используется критерий Фишера. Матрицу объекты-признаки  $X$  нужно поделить на две части:

- В первую матрицу попадают значимые признаки, в том числе и константа.
- Во вторую матрицу заносятся признаки, значимость которых мы хотим проверить.

Описание критерия Фишера представлено в таблице 7.2

нулевая гипотеза: альтернативная гипотеза: статистика:  нулевое распределение	$X_{n \times (k+1)} = \begin{pmatrix} X_1 & X_2 \\ n \times (k+1-k_1) & n \times k_1 \end{pmatrix}$ $\beta^T = \begin{pmatrix} \beta_1^T & \beta_2^T \end{pmatrix}^T$ $(k+1) \times 1 \quad (k+1-k_1) \times 1 \quad k_1 \times 1$ $H_0 : \beta_2 = 0$ $H_1 : H_0 \text{ неверна}$ $RSS_r = \ y - X_1 \beta_1\ _2^2$ $RSS_{ur} = \ y - X \beta\ _2^2$ $F = \frac{(RSS_r - RSS_{ur})/k_1}{RSS_{ur}/(n - k - 1)}$ $F(k_1, n - k - 1)$
---	---

Таблица 7.2: Описание критерия Фишера

Стоит сделать несколько оговорок:

- Если тестировать критерием Стьюдента и Фишера один и тот же признак, они будут достигать одинаковых уровней значимости. При  $k > 1$  могут возникать неоднозначные ситуации:
- Критерий Фишера говорит о значимости признаков группы признаков, при этом критерий Стьюдента не один из признаков не признаёт значимым. Это может быть объяснено двумя способами:
  - отдельные признаки из группы недостаточно хорошо объясняют отклик, но их совокупный эффект при прогнозировании значим.
  - группа признаков может иметь мультиколлинеарные признаки.

- Критерий Фишера не отвергает гипотезу о незначимости группы признаков, а критерий Стьюдента по отдельным компонентам какие-то из гипотез отвергает. То есть все вместе признаки незначимы, а какие-то из них по отдельности оказываются значимыми. Возможны две ситуации:
  - незначимые признаки из группы признаков маскируют влияние значимых
  - значимость отдельных признаков из группы признаков — это результат эффекта множественной проверки гипотез. Критерии Фишера проверяют всего одну гипотезу, а критерии Стьюдента проверяют целую серию из  $k_1$  гипотез, и какие-то из них могут отклониться просто случайно.

Может возникнуть вопрос в возможности построения какой-либо адекватной модели вообще. Для этого используется критерий Фишера (таблица 7.3). Он проверяет гипотезу о равенстве всех коэффициентов регрессии при признаках равным нулю.

нулевая гипотеза:	$H_0 : \beta_1 = \dots = \beta_k = 0$
альтернативная гипотеза:	$H_1 : \text{неверна}$
статистика:	$F = \frac{R^2/k}{(1-R^2)/(n-k-1)}$
нулевое распределение	$F \sim F(k, n - k - 1)$

Таблица 7.3: Описание критерия Фишера

О категориальных признаках. Категориальный предиктор, кодируемый несколькими фиктивными переменными, необходимо включать или исключать целиком. Значимость соответствующих фиктивных переменных лучше проверять в совокупности.

В случае, когда по отдельности какие-то фиктивные переменные не значимы, допустимо объединять уровни категориального предиктора, основываясь на интерпретации.

## 7.4 Задачи

Задача №1. По 1260 опрошенным имеются следующие данные:

- заработка плата за час работы, \$;
- опыт работы, лет;
- образование, лет;
- внешняя привлекательность, в баллах от 1 до 5;
- бинарные признаки: пол, семейное положение, состояние здоровья (хорошее/плохое), членство в профсоюзе, цвет кожи (белый/чёрный), занятость в сфере обслуживания (да/нет).

Требуется оценить влияние внешней привлекательности на уровень заработка с учётом всех остальных факторов.

```

import pandas as pd
import numpy as np
import scipy
import matplotlib.pyplot as plt
import statsmodels
import statsmodels.formula.api as smf
import statsmodels.stats.api as sms

from statsmodels.graphics.regressionplots import plot_leverage_resid2
from pandas.plotting import scatter_matrix

%matplotlib inline

```

Чтение исходных данных:

```

raw = pd.read_csv("datasets/beauty.csv", sep=";", index_col=False)
raw.head()

```

	wage	exper	union	goodhlth	black	female	married	service	educ	looks
0	5.73	30	0	1	0	1	1	1	14	4
1	4.28	28	0	1	0	1	1	0	12	3
2	7.96	35	0	1	0	1	0	0	10	4
3	11.57	38	0	1	0	0	1	1	16	3
4	11.42	27	0	1	0	0	1	0	16	3

Перед построением модели необходимо убедиться в том, что значения каждого из категориальных признаков входят в модель достаточное количество раз.

```

for cat in ["union", "goodhlth", "black", "female", "married", "service",
            "looks"]:
    print(raw[cat].value_counts())

```

```

0    917
1    343
Name: union, dtype: int64
1    1176
0     84
Name: goodhlth, dtype: int64
0    1167
1     93
Name: black, dtype: int64
0    824
1    436
Name: female, dtype: int64
1    871
0    389
Name: married, dtype: int64
0    915
1    345
Name: service, dtype: int64

```

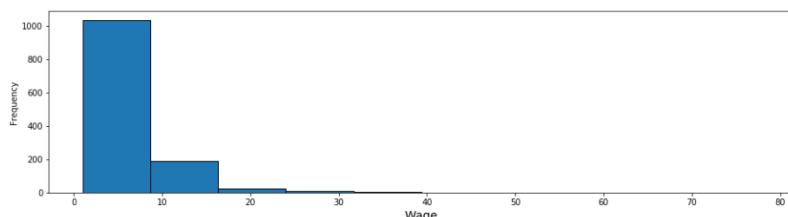
Рис. 7.3: Задача о внешней привлекательности

```
Name: service, dtype: int64
3    722
4    364
2    142
5     19
1     13
Name: looks, dtype: int64
```

Оценим распределение целевого признака.

```
data = raw

plt.figure(figsize=(16,4))
data.wage.plot.hist(edgecolor='k')
plt.xlabel('Wage', fontsize=14)
plt.show()
```



На графике зарплат мы наблюдаем длинный хвост. Вероятно, есть наблюдение с большим значением зарплаты.

```
data.wage.sort_values(ascending=False).head()

602    77.72
269    41.67
415    38.86
69     32.79
290    31.09
Name: wage, dtype: float64
```

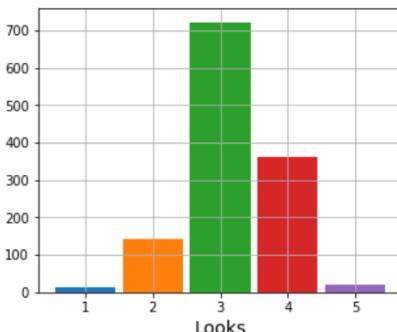
Устраним наблюдение с зарплатой больше 70 (чтобы модель не подстраивалась под это наблюдение):

```
data = data[data.wage < 70]
```

Рассмотрим признак looks (привлекательность)

```
plt.figure(figsize=(5,4))
data.groupby('looks')[['looks']].agg(lambda x: len(x)).plot(kind='bar',
                                                               width=0.9)
plt.xticks(rotation=0)
plt.xlabel('Looks', fontsize=14)
plt.grid()
plt.show()
```

Рис. 7.4: Задача о внешней привлекательности (продолжение)



Перекодируем категориальный признак "привлекательность" на два признака:

- привлекательность ниже средней
- привлекательность выше средней

Создадим данные признаки:

```
data.loc[:, 'belowavg'] = data.looks.apply(lambda x : 1 if x < 3 else 0)
data.loc[:, 'aboveavg'] = data.looks.apply(lambda x : 1 if x > 3 else 0)
```

Устраним признак привлекательность:

```
data = data.drop('looks', axis=1)
data.head()
```

	wage	exper	union	goodhlth	black	female	married	service	educ	belowavg
0	5.73	30	0	1	0	1	1	1	14	
1	4.28	28	0	1	0	1	1	0	12	
2	7.96	35	0	1	0	1	0	0	10	
3	11.57	38	0	1	0	0	1	1	16	
4	11.42	27	0	1	0	0	1	0	16	

Выполним построение модели со всеми предикторами:

```
m1 = smf.ols('wage ~ exper + union + goodhlth + black + female + married + ' +
              'service + educ + belowavg + aboveavg', data=data)
fitted = m1.fit()
fitted.summary()
```

Рис. 7.5: Задача о внешней привлекательности (продолжение)

## OLS Regression Results

<b>Dep. Variable:</b>	wage	<b>R-squared:</b>	0.262			
<b>Model:</b>	OLS	<b>Adj. R-squared:</b>	0.256			
<b>Method:</b>	Least Squares	<b>F-statistic:</b>	44.31			
<b>Date:</b>	Mon, 17 Dec 2018	<b>Prob (F-statistic):</b>	1.42e-75			
<b>Time:</b>	13:33:58	<b>Log-Likelihood:</b>	-3402.9			
<b>No. Observations:</b>	1259	<b>AIC:</b>	6828.			
<b>Df Residuals:</b>	1248	<b>BIC:</b>	6884.			
<b>Df Model:</b>	10					
<b>Covariance Type:</b>	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Intercept	-0.5898	0.743	-0.793	0.428	-2.048	0.869
exper	0.0850	0.009	9.118	0.000	0.067	0.103
union	0.4786	0.234	2.048	0.041	0.020	0.937
goodhlth	-0.0444	0.417	-0.107	0.915	-0.862	0.773
black	-0.6748	0.403	-1.674	0.094	-1.466	0.116
female	-2.3058	0.242	-9.522	0.000	-2.781	-1.831
married	0.4569	0.240	1.905	0.057	-0.014	0.927
service	-0.7303	0.252	-2.896	0.004	-1.225	-0.236
educ	0.4820	0.043	11.272	0.000	0.398	0.566
belowavg	-0.8185	0.323	-2.532	0.011	-1.453	-0.184
aboveavg	-0.0729	0.234	-0.311	0.756	-0.532	0.387
	<b>Omnibus:</b> 898.031		<b>Durbin-Watson:</b> 1.858			
<b>Prob(Omnibus):</b>	0.000	<b>Jarque-Bera (JB):</b> 17969.693				
<b>Skew:</b>	3.076	<b>Prob(JB):</b> 0.00				
<b>Kurtosis:</b>	20.456	<b>Cond. No.</b> 189.				

Оценим распределение остатков:

```
def residual_plots(fitted):
    plt.figure(figsize=(16,4))

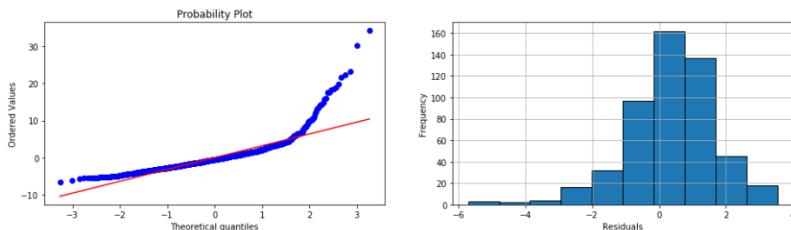
    plt.subplot(121)
    sc.stats.probplot(fitted.resid, dist="norm", plot=plt)

    plt.subplot(122)
    np.log(fitted.resid).plot.hist(edgecolor='k')
    plt.grid()
    plt.xlabel('Residuals')

    plt.show()

residual_plots(fitted)
```

Рис. 7.6: Задача о внешней привлекательности (продолжение)



```
scipy.stats.shapiro(fitted.resid)
```

```
(0.7783821821212769, 3.247975543201691e-38)
```

Если предположение о нормальности ошибки не выполняется, то критерии Стьюдента и Фишера перестают корректно работать. В данном случае, мы можем строить не регрессию исходной модели, а регрессию логарифма предсказываемого признака.

```
m2 = smf.ols('np.log(wage) ~ exper + union + goodhlth + black + female + ' +
              'married + service + educ + belowavg + aboveavg', data=data)
fitted = m2.fit()
fitted.summary()
```

OLS Regression Results

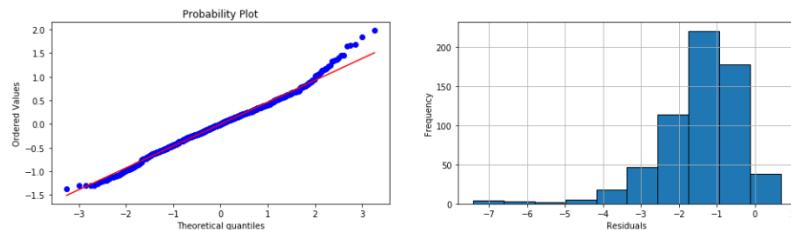
Dep. Variable:	np.log(wage)	R-squared:	0.383
Model:	OLS	Adj. R-squared:	0.379
Method:	Least Squares	F-statistic:	77.63
Date:	Mon, 17 Dec 2018	Prob (F-statistic):	1.18e-123
Time:	13:34:09	Log-Likelihood:	-816.90
No. Observations:	1259	AIC:	1656.
Df Residuals:	1248	BIC:	1712.
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
Intercept	0.4515	0.095	4.737	0.000	0.265	0.639
exper	0.0138	0.001	11.546	0.000	0.011	0.016
union	0.1785	0.030	5.957	0.000	0.120	0.237
goodhlth	0.0785	0.053	1.470	0.142	-0.026	0.183
black	-0.0989	0.052	-1.913	0.056	-0.200	0.003
female	-0.3938	0.031	-12.684	0.000	-0.455	-0.333
married	0.0425	0.031	1.383	0.167	-0.018	0.103
service	-0.1505	0.032	-4.656	0.000	-0.214	-0.087
educ	0.0799	0.005	14.581	0.000	0.069	0.091
belowavg	-0.1305	0.041	-3.148	0.002	-0.212	-0.049
aboveavg	-0.0041	0.030	-0.138	0.890	-0.063	0.055

Рис. 7.7: Задача о внешней привлекательности (продолжение)

Omnibus:	27.318	Durbin-Watson:	1.853
Prob(Omnibus):	0.000	Jarque-Bera (JB):	46.550
Skew:	0.159	Prob(JB):	7.80e-11
Kurtosis:	3.887	Cond. No.	189.

```
residual_plots(fitted)
```



```
scipy.stats.shapiro(fitted.resid)
```

```
(0.9912799000740051, 8.536291034033638e-07)
```

Формально, распределение остатков не стало нормальным, но отклонение от нормальности стало менее заметно визуально.

Оценим изменение остатков регрессионной модели от непрерывных признаков:

```
plt.figure(figsize=(16,4))

plt.subplot(121)
plt.scatter(data['educ'], fitted.resid)
plt.xlabel('Education', fontsize=14)
plt.ylabel('Residuals', fontsize=14)

plt.subplot(122)
plt.scatter(data['exper'], fitted.resid, alpha=0.04)
plt.xlabel('Experience', fontsize=14)
plt.ylabel('Residuals', fontsize=14)

plt.show()
```

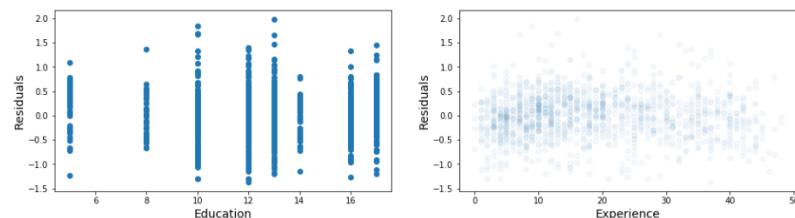


Рис. 7.8: Задача о внешней привлекательности (продолжение)

Мы можем заметить, что на графике опыта работы есть некоторая квадратичная зависимость. Добавим квадрат признака в модель.

```
m3 = smf.ols('np.log(wage) ~ exper + np.power(exper,2) + union + goodhlth +'
              'black + female + married + service + educ + belowavg + aboveavg',
              data=data)
fitted = m3.fit()
fitted.summary()
```

OLS Regression Results

Dep. Variable:	np.log(wage)	R-squared:	0.403			
Model:	OLS	Adj. R-squared:	0.398			
Method:	Least Squares	F-statistic:	76.46			
Date:	Mon, 17 Dec 2018	Prob (F-statistic):	3.19e-131			
Time:	13:34:28	Log-Likelihood:	-796.86			
No. Observations:	1259	AIC:	1618.			
Df Residuals:	1247	BIC:	1679.			
Df Model:	11					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Intercept	0.3424	0.095	3.588	0.000	0.155	0.530
exper	0.0404	0.004	9.290	0.000	0.032	0.049
np.power(exper, 2)	-0.0006	9.63e-05	-6.351	0.000	-0.001	-0.000
union	0.1710	0.030	5.793	0.000	0.113	0.229
goodhlth	0.0716	0.053	1.361	0.174	-0.032	0.175
black	-0.0831	0.051	-1.631	0.103	-0.183	0.017
female	-0.3936	0.031	-12.875	0.000	-0.454	-0.334
married	0.0101	0.031	0.329	0.742	-0.050	0.070
service	-0.1599	0.032	-5.018	0.000	-0.222	-0.097
educ	0.0758	0.005	13.941	0.000	0.065	0.086
belowavg	-0.1352	0.041	-3.313	0.001	-0.215	-0.055
aboveavg	-0.0025	0.030	-0.084	0.933	-0.061	0.056
Omnibus:	30.019	Durbin-Watson:	1.849			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	56.257			
Skew:	0.140	Prob(JB):	6.08e-13			
Kurtosis:	3.997	Cond. No.	5.62e+03			

Рис. 7.9: Задача о внешней привлекательности (продолжение)

```

plt.figure(figsize = (16,4))

plt.subplot(121)
sc.stats.probplot(fitted.resid, dist="norm", plot=plt)

plt.subplot(122)
np.log(fitted.resid).plot.hist(edgecolor='k')
plt.xlabel('Residuals', fontsize=14)
plt.figure(figsize = (16,5))

plt.subplot(131)
plt.scatter(data['educ'],fitted.resid)
plt.xlabel('Education', fontsize=14)
plt.ylabel('Residuals', fontsize=14)

plt.subplot(132)
plt.scatter(data['exper'],fitted.resid)
plt.xlabel('Experience', fontsize=14)
plt.ylabel('Residuals', fontsize=14)

plt.subplot(133)
plt.scatter(data['exper']**2,fitted.resid)
plt.xlabel('Experience^2', fontsize=14)
plt.ylabel('Residuals', fontsize=14)

plt.show()

```

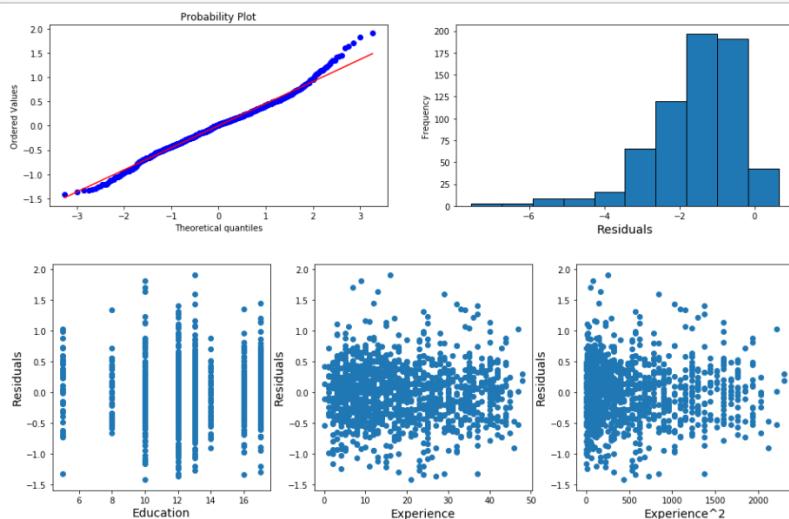


Рис. 7.10: Задача о внешней привлекательности (продолжение)

Используем критерий Брайша-Пагана для проверки гомоскедастичности ошибок:

```
print('Breusch-Pagan test: p = %f'
      % sms.het_breusvhagan(fitted.resid, fitted.model.exog)[1])
```

Breusch-Pagan test: p = 0.000004

Ошибки гетероскедастичны, следовательно, значимость признаков может определяться неверно. Выполним поправку Уайта:

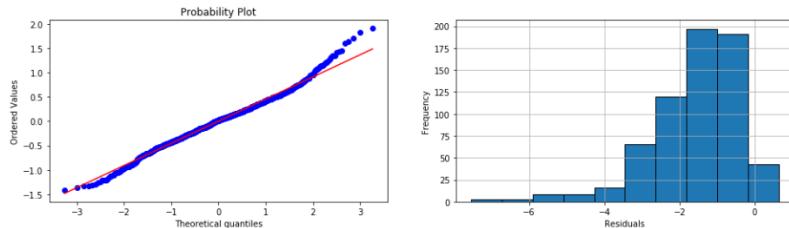
```
m4 = smf.ols('np.log(wage) ~ exper + np.power(exper,2) + union + goodhlth + '
              'black + female + married + service + educ + belowavg + aboveavg',
              data=data)
fitted = m4.fit(cov_type='HC1')
fitted.summary()
```

OLS Regression Results

Dep. Variable:	np.log(wage)		R-squared:	0.403
Model:	OLS		Adj. R-squared:	0.398
Method:	Least Squares		F-statistic:	87.29
Date:	Mon, 17 Dec 2018		Prob (F-statistic):	4.23e-146
Time:	13:34:38		Log-Likelihood:	-796.86
No. Observations:	1259		AIC:	1618.
Df Residuals:	1247		BIC:	1679.
Df Model:	11			
Covariance Type:	HC1			
	coef	std err	z	P> z  [0.025 0.975]
Intercept	0.3424	0.104	3.282	0.001 0.138 0.547
exper	0.0404	0.004	9.511	0.000 0.032 0.049
np.power(exper, 2)	-0.0006	9.46e-05	-6.469	0.000 -0.001 -0.000
union	0.1710	0.026	6.463	0.000 0.119 0.223
goodhlth	0.0716	0.064	1.123	0.262 -0.053 0.197
black	-0.0831	0.052	-1.599	0.110 -0.185 0.019
female	-0.3936	0.031	-12.702	0.000 -0.454 -0.333
married	0.0101	0.030	0.340	0.734 -0.048 0.068
service	-0.1599	0.033	-4.786	0.000 -0.225 -0.094
educ	0.0758	0.006	13.387	0.000 0.065 0.087
belowavg	-0.1352	0.040	-3.384	0.001 -0.214 -0.057
aboveavg	-0.0025	0.030	-0.083	0.934 -0.061 0.056
Omnibus:	30.019	Durbin-Watson:	1.849	
Prob(Omnibus):	0.000	Jarque-Bera (JB):	56.257	
Skew:	0.140	Prob(JB):	6.08e-13	
Kurtosis:	3.997	Cond. No.	5.62e+03	

Рис. 7.11: Задача о внешней привлекательности (продолжение)

```
residual_plots(fitted)
```



Удалим незначимые признаки. Признак *aboveavg* оставляем в модели, так как является ключевым, согласно условию задачи.

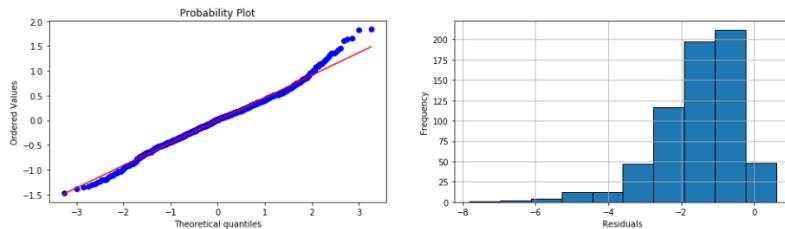
```
m5 = smf.ols('np.log(wage) ~ exper + np.power(exper,2) + union + female + '+
             'service + educ + belowavg + aboveavg', data=data)
fitted = m5.fit(cov_type='HC1')
fitted.summary()
```

OLS Regression Results

Dep. Variable:	np.log(wage)	R-squared:	0.400			
Model:	OLS	Adj. R-squared:	0.397			
Method:	Least Squares	F-statistic:	121.1			
Date:	Mon, 17 Dec 2018	Prob (F-statistic):	6.49e-150			
Time:	13:34:42	Log-Likelihood:	-799.30			
No. Observations:	1259	AIC:	1617.			
Df Residuals:	1250	BIC:	1663.			
Df Model:	8					
Covariance Type:	HC1					
	coef	std err	z	P> z	[0.025	0.975]
Intercept	0.3906	0.084	4.674	0.000	0.227	0.554
exper	0.0410	0.004	9.781	0.000	0.033	0.049
np.power(exper, 2)	-0.0006	9.35e-05	-6.748	0.000	-0.001	-0.000
union	0.1695	0.026	6.414	0.000	0.118	0.221
female	-0.4043	0.030	-13.560	0.000	-0.463	-0.346
service	-0.1600	0.033	-4.785	0.000	-0.225	-0.094
educ	0.0773	0.006	13.549	0.000	0.066	0.089
belowavg	-0.1307	0.040	-3.279	0.001	-0.209	-0.053
aboveavg	-0.0010	0.030	-0.035	0.972	-0.059	0.057
Omnibus:	26.927	Durbin-Watson:	1.842			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	49.409			
Skew:	0.120	Prob(JB):	1.87e-11			
Kurtosis:	3.941	Cond. No.	4.49e+03			

Рис. 7.12: Задача о внешней привлекательности (продолжение)

```
residual_plots(fitted)
```



Посмотрим, не стала ли модель от удаления трёх признаков статистически значимо хуже, с помощью критерия Фишера:

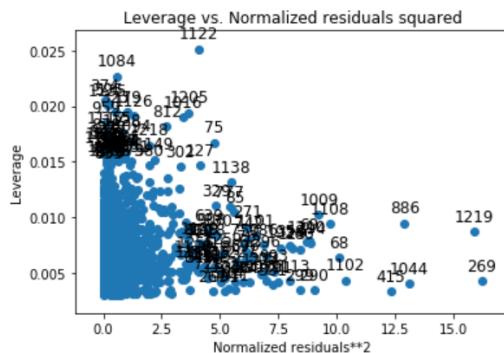
```
print("F = %f, p = %f, k1 = %f" % m4.fit().compare_f_test(m5.fit()))
```

```
F = 1.611478, p = 0.184911, k1 = 3.000000
```

Достигаемый уровень значимости позволяет нам сказать, что удаленные признаки не вносили значимый вклад в модель.

Проверим, нет ли наблюдений, которые слишком сильно влияют на регрессионное уравнение:

```
plt.figure(figsize=(10, 10))
plot_leverage_resid2(fitted)
plt.show()
```



Точка, на которой мы больше всего ошибаемся - 269. Точка с самым большим влиянием - 1122. Рассмотрим данные объекты:

```
data.iloc[[269]]
```

Рис. 7.13: Задача о внешней привлекательности (продолжение)

	wage	exper	union	goodhth	black	female	married	service	educ	be
269	41.67	16	0	0	0	0	1	0	13	

```
data.iloc[[1122]]
```

	wage	exper	union	goodhth	black	female	married	service	educ	be
1123	1.92	8	0	1	0	0	1	0	12	

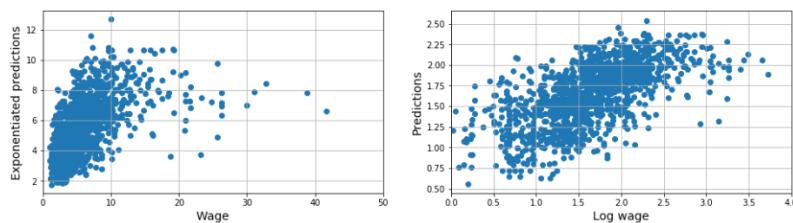
Выводы по построенной модели:

```
plt.figure(figsize=(16, 4))

plt.subplot(121)
plt.scatter(data['wage'], np.exp(fitted.fittedvalues))
plt.xlabel('Wage', fontsize=14)
plt.ylabel('Exponentiated predictions', fontsize=14)
plt.xlim([0,50])
plt.grid()

plt.subplot(122)
plt.scatter(np.log(data['wage']), fitted.fittedvalues)
plt.xlabel('Log wage', fontsize=14)
plt.ylabel('Predictions', fontsize=14)
plt.xlim([0,4])
plt.grid()

plt.show()
```



При интересующих нас факторах привлекательности стоят коэффициенты -0.1307 (привлекательность ниже среднего) и -0.0010 (привлекательность выше среднего).

Поскольку регрессия делалась на логарифм отклика, интерпретировать их можно как прирост в процентах. С учётом дополнительных факторов представители генеральной совокупности, из которой взята выборка, получают в среднем:

- на 13% меньше, если их привлекательность ниже среднего ( $p = 0.001$ , 95% доверительный интервал —  $[-5, -21]\%$ );
- столько же, если их привлекательность выше среднего ( $p = 0.972$ , 95% доверительный интервал —  $[-6, 6]\%$ ).

Рис. 7.14: Задача о внешней привлекательности (окончание)



# Глава 8

## Задача кредитного скоринга

В файле credit\_card\_default\_analysis.csv хранится информация о кредитной истории клиентов одного из банков. Поля имеют следующий смысл:

- LIMIT\_BAL: размер кредитного лимита (в том числе и на семью клиента)
- SEX: пол клиента (1 = мужской, 2 = женский )
- EDUCATION: образование (0 = доктор, 1 = магистр; 2 = бакалавр; 3 = выпускник школы; 4 = начальное образование; 5=прочее; 6 = нет данных ).
- MARRIAGE: (0 = отказываюсь отвечать; 1 = замужем/женат; 2 = холост; 3 = нет данных).
- AGE: возраст в годах
- PAY\_0 - PAY\_6 : История прошлых платежей по кредиту. PAY\_6 - платеж в апреле, ... PAY\_0 - платеж в сентябре. Платеж = (0 = исправный платеж, 1=задержка в один месяц, 2=задержка в 2 месяца ...)
- BILL\_AMT1 - BILL\_AMT6: задолженность, BILL\_AMT6 - на апрель, BILL\_AMT1 - на сентябрь

- PAY\_AMT1 - PAY\_AMT6: сумма уплаченная в PAY\_AMT6 - апреле, ..., PAY\_AMT1 - сентябре
- default - индикатор невозврата денежных средств

Задания:

1. Размер кредитного лимита (**LIMIT\_BAL**). В двух группах тех людей, кто вернул кредит (**default = 0**) и тех, кто его не вернул (**default = 1**) проверьте гипотезы:

- о равенстве медианных значений кредитного лимита с помощью подходящей интервальной оценки
- о равенстве распределений с помощью одного из подходящих непараметрических критериев проверки равенства средних.

Значимы ли полученные результаты с практической точки зрения?

2. Проверьте гипотезу о том, что гендерный состав групп людей, вернувших и не вернувших кредит, отличается.
3. Проверьте гипотезу о том, что образование не влияет на то, вернет ли человек долг. Предложите способ наглядного представления разницы в ожидаемых и наблюдаемых значениях количества человек вернувших и не вернувших долг. Например, составьте таблицу сопряженности "образование"на "возврат долга" где значением ячейки была бы разность между наблюдаемым и ожидаемым количеством человек. Как бы вы предложили модифицировать таблицу так, чтобы привести значения ячеек к одному масштабу не потеряв в интерпретируемости? Наличие какого образования является наилучшим индикатором того, что человек отдаст / не отдаст долг?
4. Проверьте, как связан семейный статус с индикатором дефолта: нужно предложить меру, по которой можно измерить возможную связь этих переменных и посчитать ее значение.
5. Относительно двух групп людей вернувших и не вернувших кредит проверьте следующие гипотезу о равенстве медианных значений возрастов людей.

```

import random
from math import sqrt

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

from scipy import stats
from statsmodels.stats.proportion import proportion_confint
from seaborn import countplot

```

Чтение данных:

```

data = pd.read_csv("datasets/credit_card_default_analysis.csv")
data.head()

```

	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3
0	1	20000	2	2	1	24	2	2	0
1	2	120000	2	2	2	26	0	2	0
2	3	90000	2	2	2	34	0	0	0
3	4	50000	2	2	1	37	0	0	0
4	5	50000	1	2	1	57	0	0	0

5 rows × 25 columns

Задание №1. Выполним построение гистограмм:

```

fig, ax = plt.subplots(nrows=2, ncols=1, sharex=True, figsize=(12, 9))

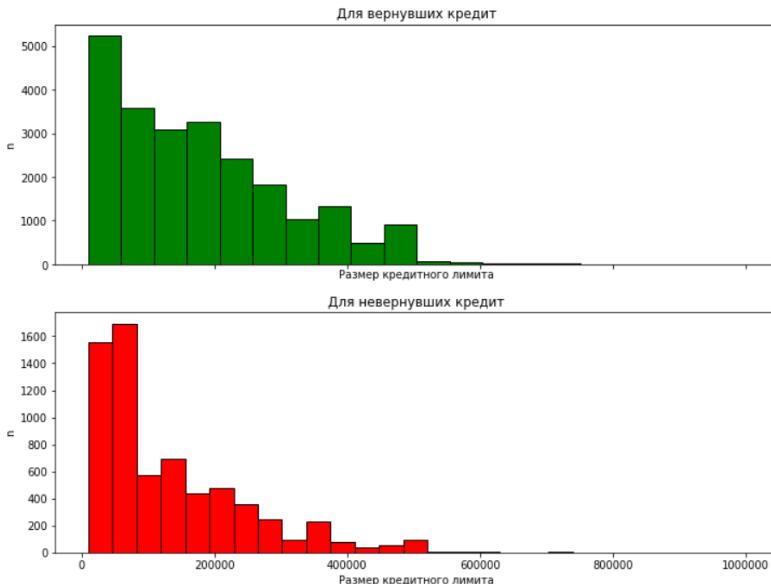
ax[0].hist(data.LIMIT_BAL[data.default == 0],
            color="green", edgecolor="k", bins=20)
ax[0].set_title("Для вернувших кредит")
ax[0].set_xlabel("Размер кредитного лимита")
ax[0].set_ylabel("n")

ax[1].hist(data.LIMIT_BAL[data.default == 1],
            color="red", edgecolor="k", bins=20)
ax[1].set_title("Для невернувших кредит")
ax[1].set_xlabel("Размер кредитного лимита")
ax[1].set_ylabel("n")

plt.show()

```

Рис. 8.1: Задача о кредитах



Распределения точно не являются нормальными. Применим бутстреп метод:

```

def get_bootstrap_samples(data, n_samples):
    indices = np.random.randint(0, len(data), (n_samples, len(data)))
    samples = data[indices]
    return samples

def stat_intervals(stat, alpha):
    boundaries = np.percentile(stat, [100 * alpha / 2., 100 * (1 - alpha / 2.)])
    return boundaries

```

```

random.seed(0)

CLim_def0 = data[data.default == 0].LIMIT_BAL.values
CLim_def1 = data[data.default == 1].LIMIT_BAL.values

CLim_def0_scores = list(map(np.median, get_bootstrap_samples(CLim_def0, 1000)))
CLim_def1_scores = list(map(np.median, get_bootstrap_samples(CLim_def1, 1000)))

print("def = 0; med(LIMIT_BAL) = ", np.median(CLim_def0))
print("def = 1; med(LIMIT_BAL) = ", np.median(CLim_def1))
print("95% доверительный интервал: ", stat_intervals(CLim_def0_scores, 0.05))
print("95% доверительный интервал: ", stat_intervals(CLim_def1_scores, 0.05))

def = 0; med(LIMIT_BAL) =  150000.0
def = 1; med(LIMIT_BAL) =  150000.0
95% доверительный интервал:  [150000. 150000.]
95% доверительный интервал:  [80000. 90000.]

```

Рис. 8.2: Задача о кредитах (продолжение)

```

delta_median_scores = list(map(lambda x: x[1] - x[0],
                               zip(CLim_def0_scores, CLim_def1_scores)))
print("95% доверительный интервал для разности медиан",
      stat_intervals(delta_median_scores, 0.05))

```

95% доверительный интервал для разности медиан [-70000. -60000.]

Медианные значения кредитных лимитов для двух групп (default = 0 / 1) не совпадают на уровне значимости 0.05.

Выполним проверку перестановочным критерием:

```

def permutation_t_stat_ind(sample1, sample2):
    return np.mean(sample1) - np.mean(sample2)

def get_random_combinations(n1, n2, max_combinations):
    index = list(range(n1 + n2))
    indices = set([tuple(index)])
    for i in range(max_combinations - 1):
        np.random.shuffle(index)
        indices.add(tuple(index))
    return [(index[:n1], index[n1:]) for index in indices]

def permutation_zero_dist_ind(sample1, sample2, max_combinations = None):
    joined_sample = np.hstack((sample1, sample2))
    n1 = len(sample1)
    n = len(joined_sample)

    if max_combinations:
        indices = get_random_combinations(n1, len(sample2), max_combinations)
    else:
        indices = [(list(index), filter(lambda i: i not in index, range(n))) \
                   for index in itertools.combinations(range(n), n1)]

    distr = [joined_sample[list(i[0])].mean() -
              joined_sample[list(i[1])].mean()
              for i in indices]
    return distr

def permutation_test(sample, mean, max_permutations = None,
                     alternative = 'two-sided'):

    if alternative not in ('two-sided', 'less', 'greater'):
        raise ValueError("alternative not recognized\n"
                         "should be 'two-sided', 'less' or 'greater'")

    t_stat = permutation_t_stat_ind(sample, mean)
    zero_distr = permutation_zero_dist_ind(sample, mean, max_permutations)
    n = len(zero_distr)

    if alternative == 'two-sided':
        return sum([1 if abs(x) >= abs(t_stat) else 0 for x in zero_distr]) / n

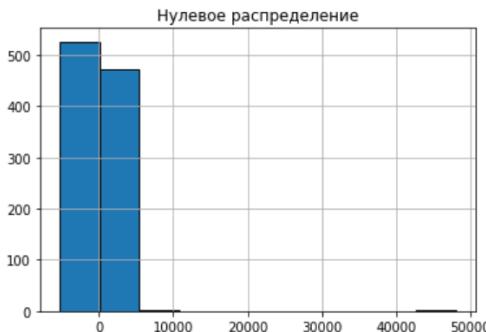
    if alternative == 'less':
        return sum([1 if x <= t_stat else 0 for x in zero_distr]) / n

    if alternative == 'greater':
        return sum([1 if x >= t_stat else 0 for x in zero_distr]) / n

```

Рис. 8.3: Задача о кредитах (продолжение)

```
plt.hist(permuation_zero_dist_ind(CLim_def0, CLim_def1,
                                  max_combinations = 1000), edgecolor="k")
plt.title("Нулевое распределение")
plt.grid()
plt.show()
```



```
print("p-value: %f" % permutation_test(CLim_def0, CLim_def1,
                                         max_permutations = 1000))
```

p-value: 0.001000

На уровне значимости 0.005 мы можем отвергнуть нулевую гипотезу о равенстве средних.

**Задание №2.** Выполним построение таблицы сопряженности:

```
ct = pd.crosstab(data.default, data.SEX)
ct.columns = ["M", "Ж"]
ct.index = ["возврат", "невозврат"]
ct
```

	M	Ж
возврат	9015	14349
невозврат	2873	3763

```
ct / ct.sum()
```

	M	Ж
возврат	0.758328	0.792237
невозврат	0.241672	0.207763

Рис. 8.4: Задача о кредитах (продолжение)

```
def proportions_confint_diff_ind(sample1, sample2, alpha = 0.05):
    z = stats.norm.ppf(1 - alpha / 2.)

    n1 = len(sample1)
    n2 = len(sample2)
    p1 = np.mean(sample1)
    p2 = np.mean(sample2)

    center = p1 - p2
    half_width_of_ci = z * np.sqrt(p1 * (1 - p1) / n1 + p2 * (1 - p2) / n2)

    return center - half_width_of_ci, center + half_width_of_ci
```

```
def_male = data[data.SEX == 1].default.values
def_female = data[data.SEX == 2].default.values
```

```
np.mean(def_male)
```

```
0.2416722745625841
```

```
np.mean(def_female)
```

```
0.20776280918727916
```

```
np.mean(def_male) - np.mean(def_female)
```

```
0.033909465375304954
```

```
print('95% доверительный интервал: ',
      proportions_confint_diff_ind(def_male, def_female))
```

```
95% доверительный интервал: (0.024207372179792706, 0.0436115585708172)
```

Данный интервал не включает ноль, что говорит о **статистической значимости разности двух долей**.

Воспользуемся z-критерием для разности двух долей:

```
def proportions_diff_z_stat_ind(sample1, sample2):
    n1 = len(sample1)
    n2 = len(sample2)

    p1 = float(sum(sample1)) / n1
    p2 = float(sum(sample2)) / n2
    P = float(p1*n1 + p2*n2) / (n1 + n2)

    return (p1 - p2) / np.sqrt(P * (1 - P) * (1. / n1 + 1. / n2))
```

Рис. 8.5: Задача о кредитах (продолжение)

```

def proportions_diff_z_test(z_stat, alternative = 'two-sided'):
    if alternative not in ('two-sided', 'less', 'greater'):
        raise ValueError("alternative not recognized\n"
                         "should be 'two-sided', 'less' or 'greater'")

    if alternative == 'two-sided':
        return 2 * (1 - stats.norm.cdf(np.abs(z_stat)))

    if alternative == 'less':
        return stats.norm.cdf(z_stat)

    if alternative == 'greater':
        return 1 - stats.norm.cdf(z_stat)

print('p-value =',
      proportions_diff_z_test(proportions_diff_z_stat_ind(def_male,
                                                          def_female)))

```

p-value = 4.472866521609831e-12

Гипотеза о том, что гендерный состав групп людей, вернувших и не вернувших кредит, не отличается, отвергается на уровне значимости 0.001.

**Задание №3.** Выполним построение таблицы сопряженности:

```

ct1= pd.crosstab(data.default, data.EDUCATION)
ct1.columns = ["доктор", "магистр", "бакалавр", "выпускник школы",
               "начальное образование", "прочее", "нет данных"]
ct1.index = ["возврат", "невозврат"]
ct1

```

	доктор	магистр	бакалавр	выпускник школы	начальное образование	прочее	нет данных
возврат	14	8549	10700	3680	116	262	43
невозврат	0	2036	3330	1237	7	18	8

```

ct1_proc = ct1 / ct1.sum()
ct1_proc

```

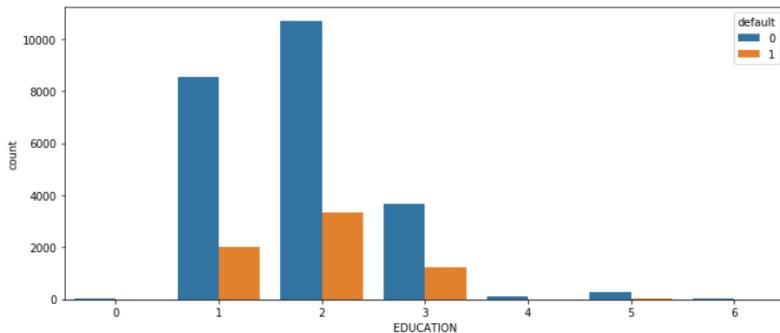
	доктор	магистр	бакалавр	выпускник школы	начальное образование	прочее	нет данных
возврат	1.0	0.807652	0.762651	0.748424	0.943089	0.935714	0.843137
невозврат	0.0	0.192348	0.237349	0.251576	0.056911	0.064286	0.156863

```

plt.figure(figsize=(12, 5))
countplot(x=data.EDUCATION, hue=data.default)
plt.ticks = ["доктор", "магистр", "бакалавр", "выпускник школы",
             "начальное образование", "прочее", "нет данных"]
plt.show()

```

Рис. 8.6: Задача о кредитах (продолжение)



Ожидаемое количество наблюдений в случае справедливости нулевой гипотезы:

```
ct2 = pd.DataFrame(stats.chi2_contingency(ct1)[3])
ct2.columns = ["доктор", "магистр", "бакалавр", "выпускник школы",
               "начальное образование", "прочее", "нет данных"]
ct2.index = ["возврат", "невозврат"]
ct2
```

	доктор	магистр	бакалавр	выпускник школы	начальное образование	прочее	нет данных
возврат	10.9032	8243.598	10926.564	3829.3596	95.7924	218.064	39.7188
невозврат	3.0968	2341.402	3103.436	1087.6404	27.2076	61.936	11.2812

```
ct2_proc = ct2 / ct2.sum()
ct2_proc
```

	доктор	магистр	бакалавр	выпускник школы	начальное образование	прочее	нет данных
возврат	0.7788	0.7788	0.7788	0.7788	0.7788	0.7788	0.7788
невозврат	0.2212	0.2212	0.2212	0.2212	0.2212	0.2212	0.2212

Разность между наблюдаемыми и ожидаемыми значениями в случае справедливости нулевой гипотезы:

```
ct1 - ct2
```

	доктор	магистр	бакалавр	выпускник школы	начальное образование	прочее	нет данных
возврат	3.0968	305.402	-226.564	-149.3596	20.2076	43.936	3.2812
невозврат	-3.0968	-305.402	226.564	149.3596	-20.2076	-43.936	-3.2812

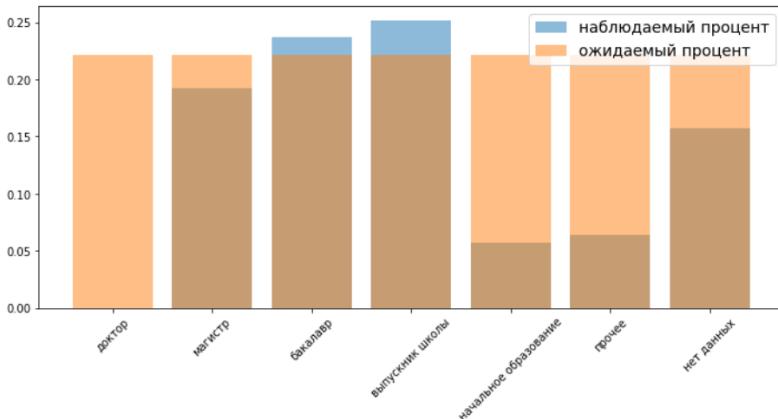
Рис. 8.7: Задача о кредитах (продолжение)

```
print("p-value =", stats.chi2_contingency(ct1)[1])
```

p-value = 1.2332626245415605e-32

Гипотеза о том, что образование не является существенным фактором в возврате кредита отвергается на уровне значимости 0.001.

```
plt.figure(figsize=(12, 5))
plt.bar(x=ctl1_proc.iloc[1].index, height=ctl1_proc.iloc[1], alpha=0.5,
        label="наблюдаемый процент")
plt.bar(x=ctl2_proc.iloc[1].index, height=ctl2_proc.iloc[1], alpha=0.5,
        label="ожидаемый процент")
plt.tick_params(axis='x', rotation=45)
plt.legend(fontsize=14)
plt.show()
```



По последнему графику можем сделать выводы о том, что лучшим индикатором возврата кредита является наличие phd, худшая группа в вопросах возврата кредита - выпускники школ и бакалавры.

**Задание №4.** Выполним построение таблицы сопряженности:

```
ct = pd.crosstab(data.default, data.MARRIAGE)
ct.columns = ['отказ', 'замужем/женат', 'холост', 'нет данных']
ct.index = ["возврат", "невозврат"]
ct
```

	♦ отказ ♦	♦ замужем/женат ♦	♦ холост ♦	♦ нет данных ♦
возврат	49	10453	12623	239
невозврат	5	3206	3341	84

Рис. 8.8: Задача о кредитах (продолжение)

Группы с вариантом "нет данных" и "отказ" являются малочисленными. Исключим их из анализа. Маловероятно, что они могут внести существенный вклад.

```
ct = ct[["замужем/женат", "холост"]]
ct.index = ["возврат", "невозврат"]
ct
```

	♦ замужем/женат ♦ холост ♦	
возврат	10453	12623
невозврат	3206	3341

```
ct_proc = ct / ct.sum()
ct_proc
```

	♦ замужем/женат ♦ холост ♦	
возврат	0.765283	0.790717
невозврат	0.234717	0.209283

```
marred = data[data.MARRIAGE == 1].default.values
single = data[data.MARRIAGE == 2].default.values
```

```
conf_interval_married = proportion_confint(sum(marred),
                                           marred.shape[0],
                                           method = 'wilson')
print("95% доверительный интервал для доли (в браке):",
      conf_interval_married)
```

95% доверительный интервал для доли (в браке): (0.22768464802142566, 0.24189859922313958)

```
conf_interval_single = proportion_confint(sum(single),
                                           single.shape[0],
                                           method = 'wilson')
print("95% доверительный интервал для доли (одинокие):",
      conf_interval_single)
```

95% доверительный интервал для доли (одинокие): (0.20304332502846192, 0.21566332834499138)

Доверительный интервал для разности между долями:

```
proportions_confint_diff_ind(marred, single)
(0.01592898928094534, 0.034938308247285874)
```

Данный интервал не включает в себя ноль, что является достаточным основанием на уровне значимости 0.05 отвернуть нулевую гипотезу о равенстве долей для двух независимых выборок.

Рис. 8.9: Задача о кредитах (продолжение)

Высчитаем корреляцию Метьюса:

```
def MCC(table):
    assert(table.shape[0] == table.shape[1] == 2)
    [[a, b], [c, d]] = table
    return (a*d - b*c) / sqrt((a+b)*(a+c)*(d+b)*(d+c))

print("MCC =", MCC(np.array(ct)))
MCC = -0.030555369920445503
```

Данные переменные практически не коррелируют между собой.

**Задание №5.** Выполним построение таблицы сопряженности:

```
ct = pd.crosstab(data.default, data.AGE)
ct.index = ["возврат", "невозврат"]
ct_proc = ct / ct.sum(axis=0)
ct
```

AGE	21	22	23	24	25	26	27	28	29	30	...	67	68	69	:
возврат	53	391	684	827	884	1003	1164	1123	1292	1121	...	11	4	12	
невозврат	14	169	247	300	302	253	313	286	313	274	...	5	1	3	

2 rows × 56 columns

Визуализируем полученные данные:

```
from seaborn import lineplot

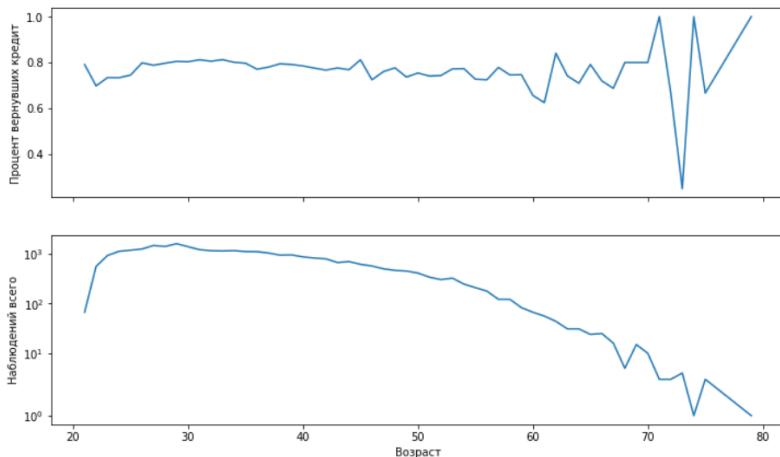
fig, ax = plt.subplots(nrows=2, ncols=1, sharex=True, figsize=(12, 7))

lineplot(x=ct.columns, y=ct_proc.iloc[0].values, ax=ax[0])
ax[0].set_xlabel("Возраст")
ax[0].set_ylabel("Процент вернувших кредит")

lineplot(x=ct.columns, y=(ct.iloc[0] + ct.iloc[1]).values, ax=ax[1])
ax[1].set_yscale('symlog')
ax[1].set_xlabel("Возраст")
ax[1].set_ylabel("Наблюдений всего")

plt.show()
```

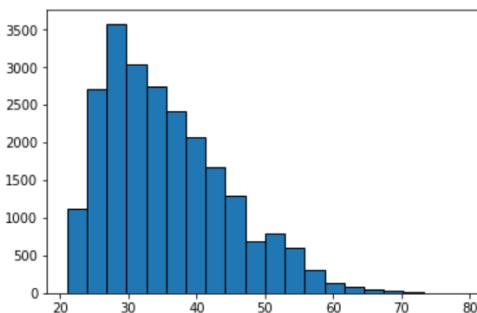
Рис. 8.10: Задача о кредитах (продолжение)



Возврат кредита не зависит от возраста (визуально). Наблюдаемые скачки обусловлены малым количеством наблюдений.

Проверим гипотезу о равенстве возрастов, вернувших и невернувших кредит. Определимся с выбором метода для сравнения средних.

```
plt.hist(x = data.AGE[data.default == 0], bins=20, edgecolor="k")
plt.show()
```



```
stats.shapiro(data.AGE[data.default == 0])
(0.9496142864227295, 0.0)
```

И визуально, и посредством критерия Шапиро-Уилка мы можем сделать вывод о том, что данные не являются нормально распределенными.

Рис. 8.11: Задача о кредитах (продолжение)

```
data[["default", "AGE"]].groupby("default").median()
```

◆ AGE ◆	
default	◆
0	34
1	34

Точечные оценки медиан не отличаются.

Проверим бутстрепом гипотезу о равенстве медиан:

```
age_def0 = data.AGE[data.default == 0].values
age_def1 = data.AGE[data.default == 1].values

age_def0_scores = list(map(np.median, get_bootstrap_samples(age_def0, 1000)))
age_def1_scores = list(map(np.median, get_bootstrap_samples(age_def1, 1000)))

np.median(age_def0), np.median(age_def1)

(34.0, 34.0)

delta_median_scores = list(map(lambda x: x[1] - x[0],
                                zip(age_def0_scores, age_def1_scores)))
print("95% доверительный интервал:",
      stat_intervals(delta_median_scores, 0.05))

95% доверительный интервал: [0. 1.]
```

Интервал включает в себя ноль, следовательно, на уровне значимости 0.05 мы не можем отвергнуть гипотезу о равенстве медиан двух выборок.

Рис. 8.12: Задача о кредитах (окончание)