GeekBrains

Корреляционный анализ

Коэффициент корреляции Пирсона. Ковариация.
Коэффициент корреляции Спирмена.

План курса

1 Расчет вероятностей

2 Дискретные распределения вероятностей

3 Разведочный анализ

4 Непрерывная случайная величина

5 Тестирование гипотез. Параметрические тесты

6 Сравнение долей. Построение доверительного интервала

7 Тестирование гипотез. Непараметрические тесты

8 **Корреляционный анализ**

9 Прошедший урок

10 Сегодняшний урок

11 Будущий урок

12 Будущий урок

13 Прошедший урок

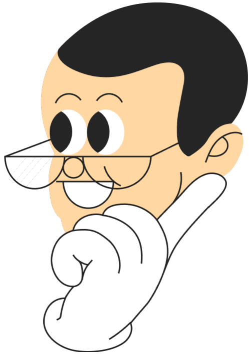
14 Сегодняшний урок

15 Будущий урок

16 Будущий урок

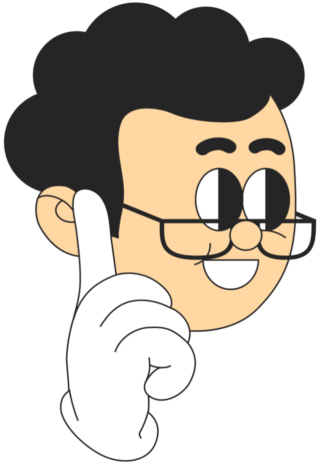
Что будет на уроке сегодня

- 📌 Понятие корреляции
- 📌 Коэффициент корреляции Пирсона
- 📌 Ковариация
- 📌 Коэффициент корреляции Спирмена



Корреляция

Корреляция – это математический показатель, по которому можно судить о наличии статистической взаимосвязи между двумя и более случайными величинами.

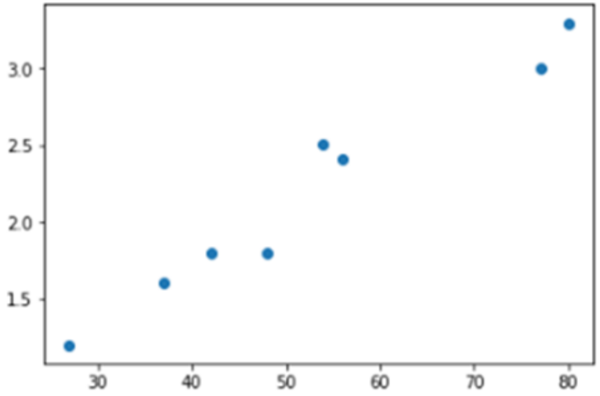


Коэффициент корреляции

Коэффициент корреляции – это коэффициент, показывающий, на сколько велика линейная взаимосвязь

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

s=np.array([27, 37, 42, 48, 57, 56, 77, 80])
s
array([27, 37, 42, 48, 57, 56, 77, 80])

p = np.array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3, 3.3])
p
array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3. , 3.3])

plt.scatter(s,p)
plt.show
```

Площадь	Цена
27	1.2
37	1.6
42	1.8
48	1.8
56	2.6
57	2.5
77	3
80	3.3



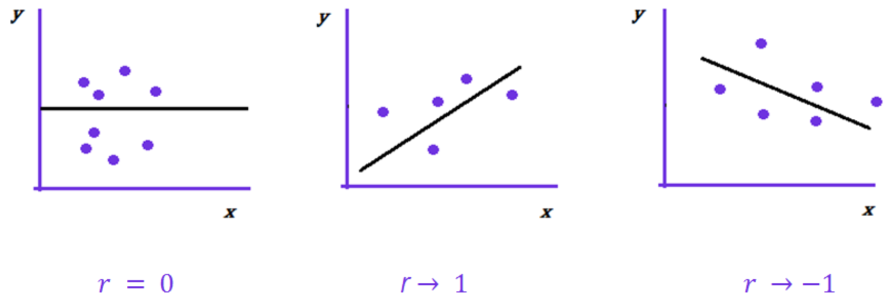
Расчет коэффициента корреляции в Python

```
s=np.array([27, 37, 42, 48, 57, 56, 77, 80])
s
array([27, 37, 42, 48, 57, 56, 77, 80])

p = np.array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3, 3.3])
p
array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3. , 3.3])

np.corrcoef (p,s)
array([[1.        , 0.97857682],
       [0.97857682, 1.        ]])
```

Интерпретация коэффициента корреляции

Коэффициент корреляции обозначается символами R или r

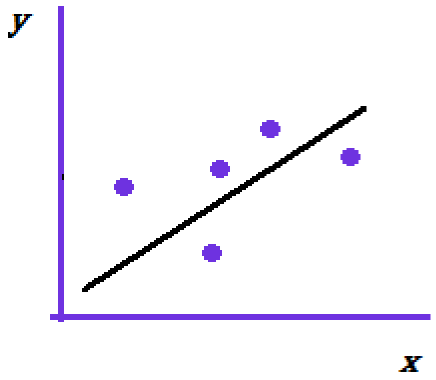
Коэффициент корреляции R может принимать значения ∈ [-1, 1]



Значение r	Интерпретация линейной зависимости
0-0.1	нет линейной зависимости
0.1-0.3	очень слабая
0.3 – 0.5	слабая
0.5 - 0.7	средняя (заметная)
0.7 - 0.9	сильная
0.9 – 1	очень сильная

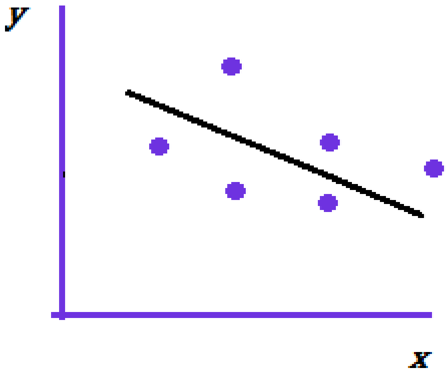
Прямая зависимость

Если коэффициент корреляции близок к 1, то между величинами наблюдается прямая связь: увеличение одной величины сопровождается увеличением другой, а уменьшение одной величины сопровождается уменьшением другой.



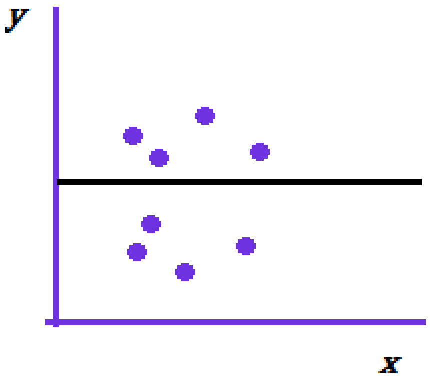
Обратная зависимость

Если же коэффициент корреляции близок к -1, то между величинами есть обратная корреляционная связь: увеличение одной величины сопровождается уменьшением другой и наоборот.



Отсутствие линейной зависимости

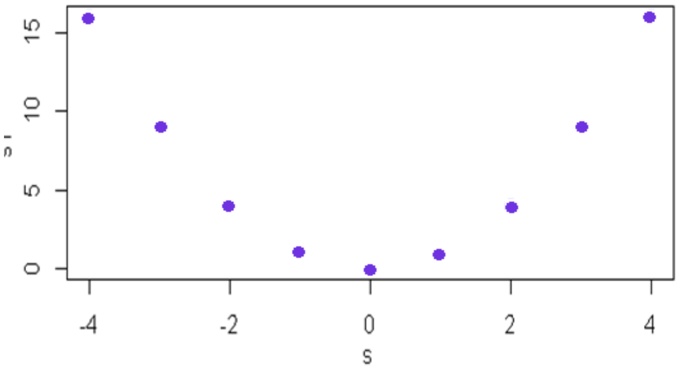
Коэффициент корреляции, равный 0, говорит о том, что между величинами нет линейной связи.



Отсутствие корреляции между двумя величинами еще не говорит о том, что между показателями нет связи.

$r = 0, \qquad y = x^2$

```
x = np.array([0, -1, 1, -2, 2, -3, 3, -4, 4])
x
array([ 0, -1,  1, -2,  2, -3,  3, -4,  4])

y = x**2
y
array([ 0,  1,  1,  4,  4,  9,  9, 16, 16])

np.corrcoef (x,y)
array([[1., 0.],
       [0., 1.]])

plt.scatter(x,y)
plt.show
```



Слабые стороны корреляционного анализа

1. Случайные величины зависимы по случайности

```
a = np.array([1, 2, 3, 4, 5])
a
array([1, 2, 3, 4, 5])

b = np.array ([7, 4, 6, 9, 0])
b
array([7, 4, 6, 9, 0])

np.corrcoef( a,b)
array([[ 1.        , -0.41602515],
       [-0.41602515,  1.        ]])


b = np.array ([11, 12, 0.8, 9, 0.4])
b
array([11. , 12. ,  0.8,  9. ,  0.4])

np.corrcoef( a,b)
array([[ 1.        , -0.68080746],
       [-0.68080746,  1.        ]])


b = np.array ([0.5, 0.7, 0.9, 0.8, 1])
b
array([0.5, 0.7, 0.9, 0.8, 1. ])

np.corrcoef( a,b)
array([[1.        , 0.90419443],
       [0.90419443, 1.        ]])
```

Слабые стороны корреляционного анализа

2. Высокая корреляция двух величин может свидетельствовать о том, что у них есть общая причина

Наличие корреляции еще не значит, что величины взаимосвязаны, но может подразумевать некую скрытую причину, 3-ю переменную.

Пример : чем больше кафе, тем больше больниц . Прямая корреляция. На самом деле взаимосвязи нет.

Какая третья скрытая переменная?

Слабые стороны корреляционного анализа

3. Можно перепутать причинно - следственную связь.

4. Коэффициент корреляции $\mathrm{r} = 0$, еще не означает отсутствие зависимости между переменными

Ковариация

Ковариация - величина , определяющая зависимость двух случайных величин

$$\mathrm{cov}_{xy} = \mathrm{M}(XY) - \mathrm{M}\,(X) * \mathrm{M}(Y),$$

где M − математическое ожидание

Масштаб ковариации зависит от дисперсии. Поэтому по ковариации нельзя судить о силе взаимосвязи 2х случайных величин. Но ее можно нормировать.

Нормированная ковариация или коэффициент Пирсона

Зная ковариацию и среднее квадратичное отклонение каждого из двух признаков, можно вычислить коэффициент корреляции Пирсона:

$$r_{xy} = \frac{cov_{xy}}{\boldsymbol{\sigma}_x * \boldsymbol{\sigma}_y}$$

Сравним значения ковариации одних и тех же случайных величин

```
p
array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3. , 3.3])
s
array([27, 37, 42, 48, 57, 56, 77, 88])

cov = np.mean(p*s) - np.mean(p) * np.mean (s)

cov
11.662500000000023

np.cov (p,s)
array([[  0.53928571,  13.32857143],
       [ 13.32857143, 344.        ]])
```

Смещенная и несмещенная ковариация

1. Даны две случайные величины площадь и цена

```
p
array([1.2, 1.6, 1.8, 1.8, 2.5, 2.6, 3. , 3.3])
s
array([27, 37, 42, 48, 57, 56, 77, 88])
```

2. Коэффициент корреляции Пирсона

```
np.corrcoef (p,s)
array([[1.        , 0.97857682],
       [0.97857682, 1.        ]])
```

3. Ковариация

```
np.cov (p,s)
array([[  0.53928571,  13.32857143],
       [ 13.32857143, 344.        ]])
```

Несмещенная ковариация

```
np.cov( p,s, ddof = 1)

array([[  0.53928571,  13.32857143],
       [ 13.32857143, 344.        ]])

np.std(p, ddof = 1)
0.7343607521414215

np.std(s, ddof = 1)
18.547236990991408

13.32857143/ (0.7343607521414215 * 18.547236990991408 )

0.9785768206878758
```

Смещенная ковариация

```
np.cov (p,s, ddof = 0 )
array([[  0.471875,   11.6625  ],
       [ 11.6625  , 301.      ]])

np.std(p, ddof = 0)
0.6869315832017042

np.std(s, ddof = 0)
17.349351572897472

11.6625 / (0.6869315832017042 * 17.349351572897472)

0.9785768205829909
```

```
np.corrcoef (p,s)
array([[1.        , 0.97857682],
       [0.97857682, 1.        ]])
```

Плюсы и минусы корреляционного анализа

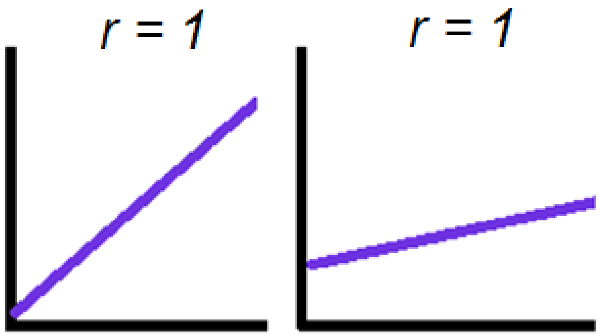
- Плюсы

✓ Простота
✓ Легкость интерпретации
✓ Показывает прямая или обратная линейная взаимосвязь
✓ Показывает, на сколько сильная линейная зависимость.

- Минусы

✓ Случайные величины могут коррелировать по случайности
✓ Есть третья скрытая переменная
✓ Высока вероятность перепутать причину и следствие
✓ Коэффициент корреляции r , равный нули, еще не говорит о том, что зависимости между величинами нет.
✓ Не показывает, как быстро изменяется зависимая величина y при изменении независимой величины x

Коэффициент корреляции не показывает, как быстро изменяется зависимая величина при изменении независимой.



Коэффициент корреляции Спирмена

Коэффициент корреляции Спирмена – это ранговый коэффициент корреляции, также показывает тесноту линейной связи, но в отличии от коэффициента корреляции Пирсона не требует нормальности распределений случайных величин и применяется для порядковых и количественных данных.

Расчет коэффициента корреляции Спирмена в Python

```
s
array([27, 37, 42, 48, 56, 57, 77, 80])

p
array([1.2, 1.6, 1.8, 1.8, 2.6, 2.5, 3. , 3.3])


stats.spearmanr(p, s)

SpearmanrResult(correlation=0.9700772721497398,
pvalue=6.548558831120599e-05)
```

Как рассчитывается коэффициент корреляции Спирмена?

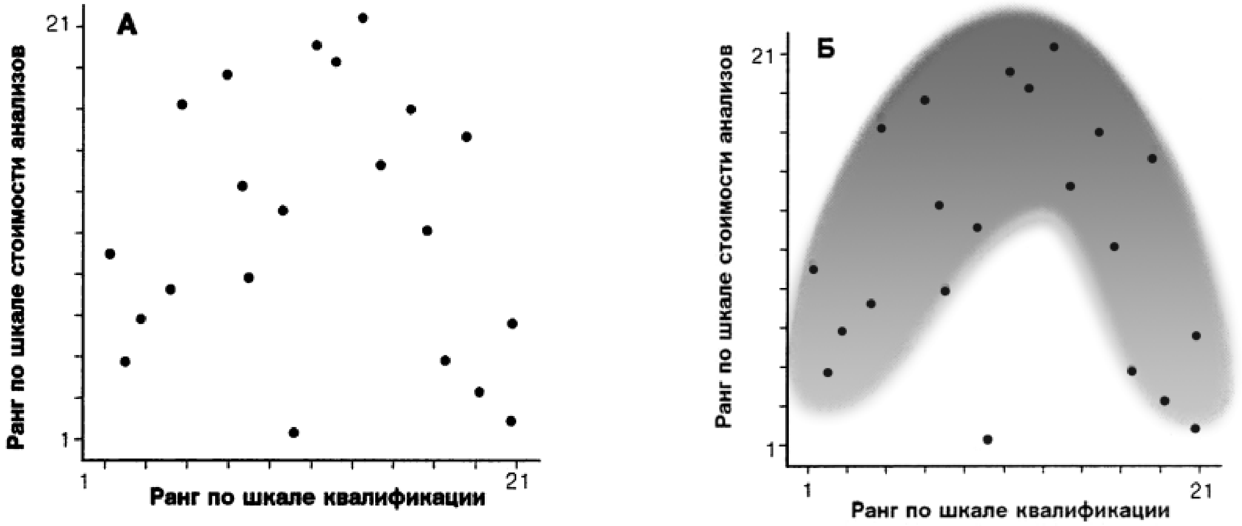
```
s
array([27, 37, 42, 48, 56, 57, 77, 80])

s2 = np.array([1, 2, 3, 4, 6, 5, 7, 8])

p
array([1.2, 1.6, 1.8, 1.8, 2.6, 2.5, 3. , 3.3])

p2= np.array([1, 2, 3.5, 3.5, 5, 6, 7, 8])


np.corrcoef(s2, p2)
array([[1.        , 0.97007727],
       [0.97007727, 1.        ]])

SpearmanrResult(correlation=0.9700772721497398,
pvalue=6.548558831120599e-05)
```

Условия применимости коэффициентов корреляции

Коэффициент корреляции Пирсона	Коэффициент корреляции Спирмена
параметрический метод	непараметрический метод
нормальность	распределение может быть отличным от нормального
количественные данные	количественные и порядковые признаки
сделать проверку на U- образную кривую	

Пример задачи

Найти зависимость между квалификацией врача и затратами на анализы , прописанные врачом, для постановки диагноза



$$r_s = -0.13$$

Стентон Гланц «Медико- биологическая статистика» 1996

Конец