KARPOV.COURSES >>>
КОНСПЕКТ

> Конспект > 3 урок > Статистики распределений, взаимосвязь случайных величин, показатели корреляции

> Оглавление

>Описательные статистики

Чаще всего, при работе с датасетами невозможно осмотреть все объекты. Сложно оценить, что из себя представляют признаки. А для решения заданий (обучить

модель, посчитать метрики) желательно предварительно изучить датасет. Было бы хорошо, если бы мы могли агрегировать каждую колонку во что-то более короткое.

Основные характеристики случайных величин:

- Математическое ожидание - среднее случайной величины, которое считается как среднее арифметическое с вероятностями в качестве весов

$$\mathbb{E}x = \sum_{i=1}^{N} x_i \mathbb{P}(x_i)$$

- Среднеквадратичное отклонение - показатель разброса случайной величины.

$$\mathbb{D}x = \sum_{i=1}^{N} \mathbb{P}(x_i)(x_1 - \mathbb{E}x)^2$$

Статистика - любая измеримая функция выборки

Мода (mode) – значение измеряемого признака, которое встречается максимально часто. Мод может быть несколько.

```
import pandas as pd
df.column_1.mode()

from scipy import stats
stats.mode(df.column_1)
```

Медиана (median) – значение признака, которое делит упорядоченное множество данных пополам. Берем множество значений признака, сортируем и берем центральное значение (значение ровно посередине выборки).

```
import pandas as pd
df.column_1.median()

import numpy as np
np.median(df.column_1)
```

Среднее (mean, среднее арифметическое) – сумма всех значений измеренного признака, деленная на количество измеренных значений (оценка на

математическое ожидание случайной величины).

$\overline{X} = \frac{x_1+x_2+...+x_n}{n}$

$\overline{X}$ - среднее выборки

M - среднее генеральной совокупности

```
import pandas as pd
df.column_1.mean()

import numpy as np
np.mean(df.column_1)
```

Когда не стоит использовать среднее значение, а лучше брать моду или медиану:

- явная асимметрия
- заметные выбросы
- несколько мод

>Статистики описывающие меры разброса

Дисперсия (variance) – средний квадрат отклонений индивидуальных значений признака от их средней величины.

Для выборки:Для генеральной совокупности:

$$\mathbb{D}x = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N}$$

$$\mathbb{D}x = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}$$

```
import pandas as pd
df.A.var()

import numpy as np
np.var(df.A)
```

a-квантиль - значение выборки, которое больше a части выборки (0.3-квантиль больше 30%)

Размах (range) – разность между максимальным и минимальным значением из распределения

$$R = X_{max} - X_{min}$$

Интерквартильный размах (range) – разность между 0.75 и 0.25 квантилями.

```
import numpy as np
np.percentile(df.A, [0, 100])
```

Чтобы не строить статистики для каждого столбца выборки вручную в pandas есть метод `describe()` . Для числовых данных метод посчитает количество значений, среднее, стандартное откл., минимум, максимум, а также процентили для каждого столбца.

```
data.describe()
```

	distance	consume	speed	temp_inside	temp_outside	AC	rain	
count	388.000000	388.000000	388.000000	376.000000	388.000000	388.000000	388.000000	388.
mean	19.652835	4.912371	41.927835	21.929521	11.358247	0.077320	0.123711	0.08
std	22.667837	1.033172	13.598524	1.010455	6.991542	0.267443	0.329677	0.27
min	1.300000	3.300000	14.000000	19.000000	-5.000000	0.000000	0.000000	0.00
25%	11.800000	4.300000	32.750000	21.500000	7.000000	0.000000	0.000000	0.00
50%	14.600000	4.700000	40.500000	22.000000	10.000000	0.000000	0.000000	0.00
75%	19.000000	5.300000	50.000000	22.500000	16.000000	0.000000	0.000000	0.00
max	216.100000	12.200000	90.000000	25.500000	31.000000	1.000000	1.000000	1.00

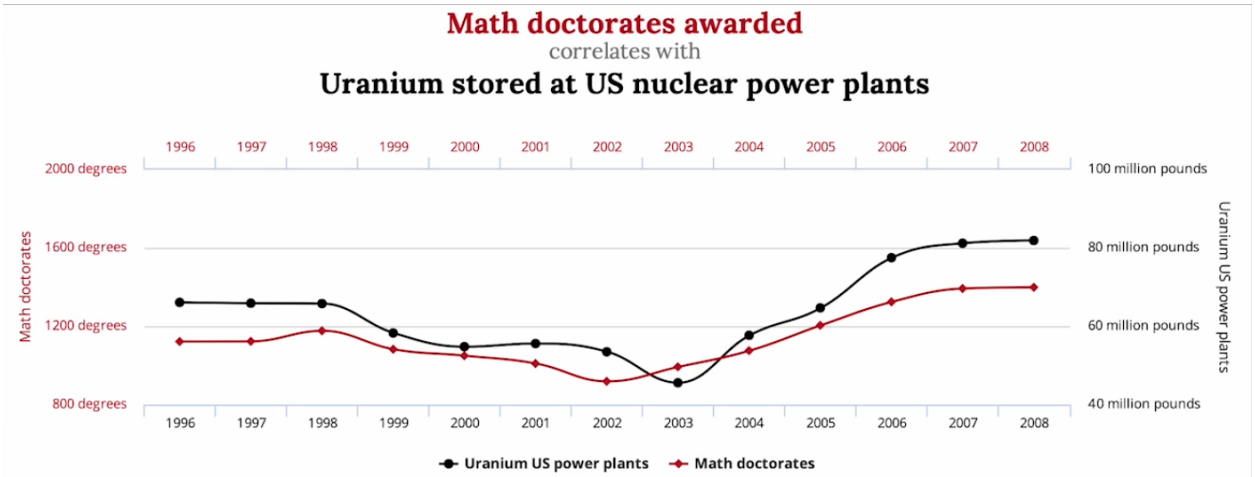
Подробнее

>Исследование взаимосвязи

Не редко, при работе с датасетами нас интересует связь между признаками (или признаков и целевой переменной). Есть два значения признаков X_1 и X_2 на одном наборе объектов и мы хотим узнать связаны ли они между собой.

Корреляция - статистическая взаимосвязь между двумя случайными переменными. Взаимосвязь может быть положительной (когда одна переменная растёт, другая тоже растёт), либо отрицательной (когда одна переменная растёт, другая уменьшается), либо отсутствовать.

Важно заметить, что из корреляции не следует причинно-следственная связь. Всегда можно найти две величины у которых высокая корреляция, но связи между ними нет:



Еще интересные корреляции - https://tylervigen.com/spurious-correlations

Бывает, что две скоррелированные величины зависят от третьей. Например, средняя продолжительность жизни в стране коррелирует с доступом к высшему образованию. Высшее образование не увеличивает продолжительность жизни, на обе величины влияет общее благосостояние страны.

>Корреляция Пирсона

Мера линейной взаимосвязи:

$$r_{x_1,x_2} = \frac{\mathbb{E}((X_1 - \mathbb{E}X_1)(X_2 - \mathbb{E}X_2))}{\sqrt{\mathbb{D}X_1 \mathbb{D}X_2}}$$

Значения от -1 до 1, если близко к 0 - нет взаимосвязи

Выборочный коэффициент Пирсона:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

>Корреляция Спирмена

Корреляция Спирмена вычисляет силу монотонной корреляции. Равна коэффициенту корреляции Пирсона между рангами

$$\hat{\rho}_{X_1,X_2} = \frac{\sum_{i=1}^{n}(rank(X_{1i}) - \frac{n+1}{2})(rank(X_{2i}) - \frac{n+1}{2})}{\frac{1}{12}(n_3 - n)} =$$

$$= 1 - \frac{6}{n^3 - n}\sum_{i=1}^{n}(rank(X_{1i}) - rank(X_{2i}))^2$$

>Корреляция Кендалла

Измеряет меру взаимной упорядоченности, сила монотонной корреляции

$$\hat{\tau}_{X_1,X_2} = \frac{C - D}{C + D} = 1 - \frac{4}{n(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}[[X_{1i} < X_{1j}] \neq [X_{2i} < X_{2j}]]$$

- C - Число согласованных пар. Такая пара двух случайных объектов выборки, где по первому признаку 1 объект больше 2 объекта, то и по второму признаку 1 объект больше 2 объекта
- D - Число несогласованных пар

Корреляции в python:

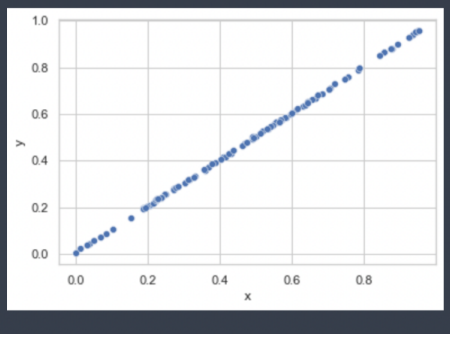
```
numpy.corrcoef(x, y)
scipy.stats.pearsonr(x, y)
scipy.stats.spearmanr(x, y)
scipy.stats.kendalltau(x, y)

# через pandas (сравнение pandas Series)
df.corr()
df.corr(method='spearman')
df.corr(method='kendall')
```

Документация numpy.corrcoef, scipy.stats.pearsonr , scipy.stats.spearmanr, scipy.stats.kendalltau, df.corr

>Как между собой соотносятся разные виды корреляций?

Как корреляции ведут себя на разных выборках? Для начала возьмем датасет, где переменная X равна Y с некоторым шумом.

```
df = pd.DataFrame()
df['x'] = np.random.rand(100)
df['y'] = df['x'] + 0.01 * np.random.rand(100)
sns.scatterplot(data=df, x='x', y='y')
plt.show()
```



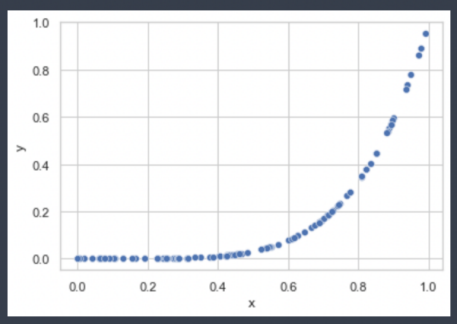
Линейная взаимосвязь очевидна. Все виды корреляций выдают почти единичную корреляцию, т.е есть положительная связь и она очень сильная.

```
print('Пирсон:  ', pearsonr(df['x'], df['y'])[0])
print('Спирман: ', spearmanr(df['x'], df['y'])[0])
print('Кендалл: ', kendalltau(df['x'], df['y'])[0])
```

```
Пирсон:   0.9999422572480603
Спирман:  0.9996399639963995
Кендалл:  0.9911111111111113
```

Теперь, немного изменим датасет: Y будет равен X^5 с некоторым шумом. Взаимосвязь будет не линейной, а степенной. Окажется, что корреляция Пиросона станет меньше и уже не будет близка к 1, а корреляции Спирмана и Кендалла всё еще будут близки к 1. Это связано с тем, что **корреляция Пирсона оценивает линейную взаимосвязь.**

```
sns.scatterplot(data=df, x='x', y='y')
plt.show()
```

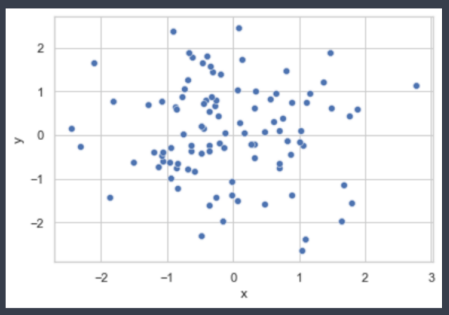



```
print('Пирсон:  ', pearsonr(df['x'], df['y'])[0])
print('Спирман: ', spearmanr(df['x'], df['y'])[0])
print('Кендалл: ', kendalltau(df['x'], df['y'])[0])
```

```
Пирсон:   0.815762248131731
Спирман:  0.9967356735673566
Кендалл:  0.9749494949494951
```

А теперь возьмем случайную выборку, где X от Y никак не зависят. Кажется, что все виды корреляции выдают около нулевое значение и говорят, что взаимосвязи между признаками нет.

```
sns.scatterplot(data=df, x='x', y='y')
plt.show()
```

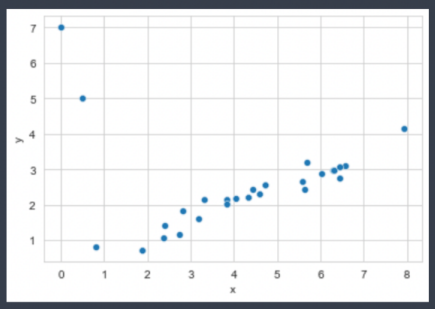



```
print('Пирсон:  ', pearsonr(df['x'], df['y'])[0])
print('Спирман: ', spearmanr(df['x'], df['y'])[0])
print('Кендалл: ', kendalltau(df['x'], df['y'])[0])
```

```
Пирсон:   -0.04109513726816867
Спирман:  0.018037803780378038
Кендалл:  0.012525252525252528
```

Вернемся к линейной взаимосвязи и добавим пару выбросов. Оказывается, тут корреляция Пирсона близка к нулю (хотя линейная взаимосвязь очевидна), а корреляции Спирмана и Кендалла выдают положительную корреляцию. Это означает, что **корреляция Пирсона очень неустойчива к выброс**

```
sns.scatterplot(data=df, x='x', y='y')
plt.show()
```



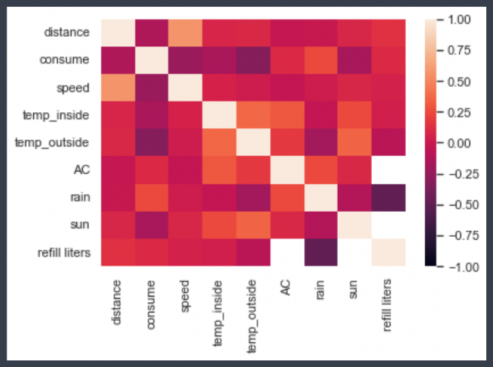

```
print('Пирсон:  ', pearsonr(df['x'], df['y'])[0])
print('Спирман: ', spearmanr(df['x'], df['y'])[0])
print('Кендалл: ', kendalltau(df['x'], df['y'])[0])
```

```
Пирсон:   0.056307464418037884
Спирман:  0.5586080586080585
Кендалл:  0.5897435897435899
```

>Heatmap

Если признаков много, то удобно визуализировать попарные корреляции признаков. Heatmap это график, где каждая ячейка соответствует какой-то паре признаков, а цвет насколько признаки скоррелированы. Для этого можно с помощью pandas вычислять попарно корреляции и визуализировать используя seaborn

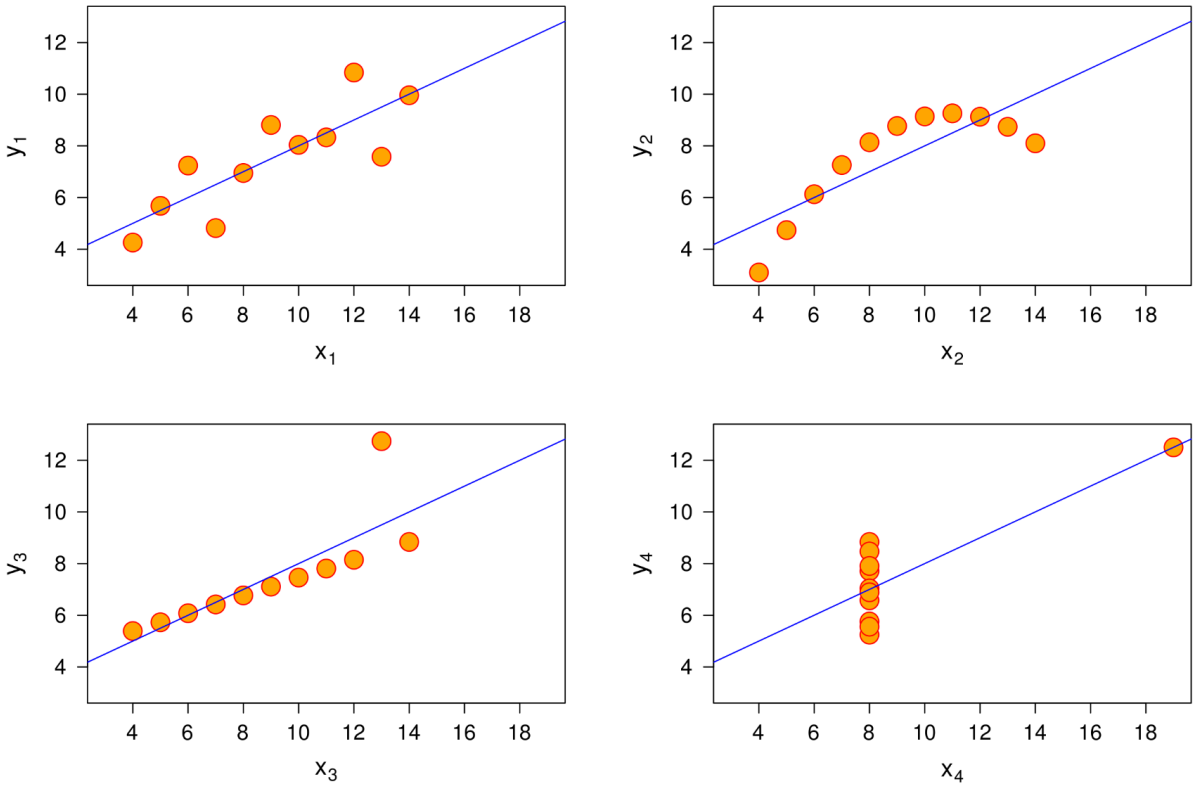
```
sns.heatmap(data.corr(method='pearson'), vmin=-1, vmax=1)
plt.show()
```



>Интерпретация значений

Знак показывает направление — отрицательный коэффициент соответствует «большему значения одного признака соответствует меньшее значение другого признака». Около 0.3 — слабая взаимосвязь, выше 0.7 — сильная.

Стоит аккуратно использовать описательные статистики и коэффициенты корреляции, потому что они не всегда полностью описывают выборку. Например, Квартет Энскомба: есть 4 датасета, и в каждом из них, каждый из признаков имеет одинаковое мат. ожидание и дисперсию, но имеют разную корреляцию.


y1
x1
y2
x2
y3
x3
y4
x4