ПРАКТИЧЕСКАЯ ЧАСТЬ 5 МОДУЛЯ КУРСА «МАТЕМАТИЧЕСКАЯ СТАТИСТИКА»

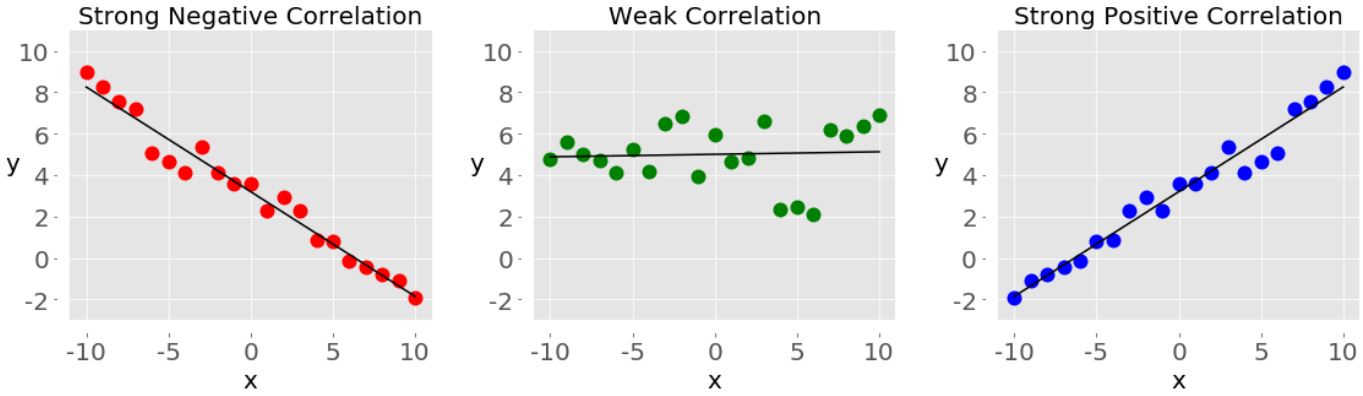
Корреляционный анализ

1. Основные понятия

Итак, мы говорим об исследовании взаимосвязи между данными. Самым первым шагом является выявление наличия (или отсутствия) такой взаимосвязи и некоторое ее численное обозначение для интерпретации направления и существенности взаимосвязи.

Корреляция – измерение, которое используется для количественной оценки взаимосвязи между переменными.

Если увеличение (или уменьшение) одной переменной вызывает соответствующее увеличение (или уменьшение) другой, то говорят, что эти две переменные находятся в прямой корреляции. Аналогично, если увеличение одной вызывает уменьшение другой или наоборот, то говорят, что переменные коррелируют обратно. Если изменение независимой переменной не вызывает изменения зависимой переменной, то они некоррелированы. Таким образом, корреляция может быть положительной (прямая корреляция), отрицательной (обратная корреляция) или нулевой. Эта взаимосвязь определяется коэффициентом корреляции.



Например, на вопрос “Как связаны рост родителей и рост детей?” корреляция отвечает “Дети высоких родителей имеют тенденцию быть выше среднего”.

Численно корреляция показывает, насколько отклонения от средней одной переменной совпадают с отклонениями другой переменной. В результате коэффициент корреляции рассчитывается по отклонениям каждой переменной от ее среднего значения.

При анализе корреляции всегда следует помнить, что корреляция не указывает на причинно-следственную связь. Она количественно определяет силу взаимосвязи между характеристиками набора данных. Иногда связь обусловлена фактором, общим для нескольких интересующих характеристик. Фактически корреляционная зависимость — это статистическая зависимость, проявляющаяся в том, что при изменении одной из величин изменяется среднее значение другой (в отличие от функциональной, когда каждому возможному значению случайной величины X соответствует одно возможное значение случайной величины Y).

Существует несколько методов подсчета коэффициента корреляции.

2. Линейная корреляция Пирсона

Линейная корреляция измеряет близость математической зависимости между переменными или характеристиками набора данных к линейной функции. Если зависимость между двумя характеристиками ближе к некоторой линейной функции, то их линейная корреляция сильнее, а абсолютное значение коэффициента корреляции выше.

Коэффициент корреляции Пирсона (произведение моментов) — это мера линейной зависимости между двумя признаками. Он представляет собой отношение ковариации переменных к произведению их стандартных отклонений (ковариация — это мера взаимосвязи двух случайных величин, измеряющая общее отклонение двух случайных величин от их ожидаемых значений).

Коэффициент корреляции Пирсона по совокупности:

$$\rho_{X,Y} = \frac{E\left[(X-\mu_X)(Y-\mu_Y)\right]}{\sigma_X \sigma_Y} = \frac{E\left[XY\right] - E\left[X\right]E\left[Y\right]}{\sqrt{E\left[X^2\right] - (E\left[X\right])^2} \cdot \sqrt{E\left[Y^2\right] - (E\left[Y\right])^2}}$$

Выборочный коэффициент корреляции Пирсона:

$$r_{X,Y} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2} \cdot \sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

Статистический вывод, основанный на коэффициенте корреляции Пирсона, часто фокусируется на одной из следующих двух целей:

- Одна из целей - проверить нулевую гипотезу о том, что истинный коэффициент корреляции ρ равен 0, на основе значения выборочного коэффициента корреляции r.
- Другая цель - получить доверительный интервал, который при повторной выборке с заданной вероятностью содержит ρ.

Необходимые условия использования коэффициента корреляции Пирсона:

- Обе переменные количественные (вам нужно будет использовать другой метод, если какая-либо из переменных качественная);
- Переменные распределены нормально (вы можете создать гистограмму каждой переменной, чтобы проверить, являются ли распределения приблизительно нормальными, если переменные “немного ненормальны”, также допускается использовать коэффициент Пирсона);
- Данные не имеют выбросов (выбросы — это наблюдения, которые не следуют тем же закономерностям, что и остальные данные, диаграмма рассеяния — один из способов проверки выбросов — ищите точки, которые находятся далеко от других);
- Отношение линейное («Линейное» означает, что отношение между двумя переменными можно достаточно хорошо описать прямой линией, вы можете использовать диаграмму рассеяния, чтобы проверить, является ли отношение между двумя переменными линейным).

Задача 1. Имеются данные средней выработки на одного рабочего *y* (тыс. руб.) и товарооборота *x* (тыс. руб.) в 12 магазинах за квартал. На основе указанных данных требуется определить зависимость (коэффициент корреляции) средней выработки на одного рабочего от товарооборота.

	Магазины											
	1	2	3	4	5	6	7	8	9	10	11	12
x	10	14	21	23	27	32	39	45	55	61	62	68
y	3.8	4.8	5.9	6.1	6.2	6.3	6.6	7.4	8.5	9.7	10.5	12.4

Решение.

Составим расчетную таблицу:

	Магазины												Сумма
	1	2	3	4	5	6	7	8	9	10	11	12	
x	10	14	21	23	27	32	39	45	55	61	62	68	457
y	3.8	4.8	5.9	6.1	6.2	6.3	6.6	7.4	8.5	9.7	10.5	12.4	88.2
x^2	100	196	441	529	729	1024	1521	2025	3025	3721	3844	4624	21779
y^2	14.44	23.04	34.81	37.21	38.44	39.69	43.56	54.76	72.25	94.09	110.25	153.76	716.3
xy	38	67.2	123.9	140.3	167.4	201.6	257.4	333	467.5	591.7	651	843.2	3882.2

Найдем выборочный коэффициент корреляции:

$$r_B = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - \left(\sum x_i\right)^2} \cdot \sqrt{n\sum y_i^2 - \left(\sum y_i\right)^2}} =$$

$$= \frac{12 \cdot 3882,2 - 457 \cdot 88,2}{\sqrt{12 \cdot 21779 - (457)^2} \cdot \sqrt{12 \cdot 716,3 - (88,2)^2}} \approx 0,959.$$

Получили, что связь сильная, прямая.

Найденное *r* – это оценка коэффициента корреляции по совокупности ρ. Примем гипотезы:

Нулевая гипотеза (H_0): $\rho = 0$;

Альтернативная гипотеза (H_A): $\rho \neq 0$.

Рассчитаем t-критерий для уровня значимости 0.05:

$$t = \frac{r}{\sqrt{\frac{1-r^2}{n-2}}}$$

t = 10.7. $t_{крит}(10, 0.05) = 2.23$. $t > t_{крит}$, значит связь статистически значима.

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

x = np.array([10, 14, 21, 23, 27, 32, 39, 45, 55, 61, 62, 68])
y = np.array([3.8, 4.8, 5.9, 6.1, 6.2, 6.3, 6.6, 7.4, 8.5, 9.7, 10.5, 12.4])
n = len(x)
alpha = 0.05
r, p_value = stats.pearsonr(x, y)
print(r, p_value)

# Принятие решения на основе критического значения t
t_score = r*(n-2)**0.5 / (1-r**2)**0.5
t_critical = stats.t.ppf(1 - alpha/2, df = n-2)
print('Critical t-Score:',t_critical)
if np.abs(t_score) > t_critical:
    print("Связь статистически значима")
else:
    print("Нельзя отклонить нулевую гипотезу")

# Принятие решения на основе p-значения
p_value = (1 - stats.t.cdf(np.abs(t_score), df = n-2)) * 2
print('P-Value :',p_value)
if p_value < alpha:
    print("Связь статистически значима")
else:
    print("Нельзя отклонить нулевую гипотезу")
```

Задача 2. На основании 18 наблюдений установлено, что на 64% вес кондитерских изделий зависит от их объема Y . Можно ли на уровне значимости α = 0,05 утверждать, что между X и Y существует зависимость?

Решение.

Из условия задачи имеем, что n =18, r = 0,64.

Введем нулевую гипотезу H_0: ρ = 0. Проверим эту гипотезу об отсутствии корреляционной зависимости (о незначимости коэффициента корреляции). Вычислим значение t-критерия: t = 3.33. Вычислим критическое значение t-критерия: $t_{крит}(16, 0.05) = 2.12$. $t > t_{крит}$, значит связь статистически значима, между X и Y существует зависимость.

Задача 3. Исследование 27 семей по среднедушевому доходу (X) и сбережениям (Y) дало результаты: X =82 у.е., S_x=31 у.е., Y =39 у.е., S_Y=29 у.е., XY =3709 (у.е.)^2. При α=0,05 проверить наличие линейной связи между X и Y.

Решение.

Вычислим выборочный коэффициент корреляции:

$$r = \frac{\overline{XY} - \overline{X} \cdot \overline{Y}}{S_X \cdot S_Y} = \frac{3709 - 82 \cdot 39}{31 \cdot 29} \approx 0{,}568 .$$

Проверим гипотезу о значимости коэффициента корреляции. Введем нулевую гипотезу H_0: r = 0 и вычислим наблюдаемое и критическое значения критерия Стьюдента: t = 3.451, $t_{крит}(25, 0.05) = 2.06$. $t > t_{крит}$, значит связь статистически значима, между X и Y существует линейная зависимость (средней силы).

Задача 4.

Найти коэффициент корреляции по заданной корреляционной таблице.

y\x	10	15	20	25	30	35	n_y
30	2	6					8
40		4	4				8
50			7	35	8		50
60			2	10	8		20
70				5	6	3	14
n_x	2	10	13	50	22	3	n = 100

Решение.

Построим ряды распределений для x и уи вычислим их характеристики.

x_i	n_i	$x_i \cdot n_i$	$\bar{x}$	$(x_i - \bar{x})^2 \cdot n_i$	D_x	σ_x
10	2	20		417.605		
15	10	150		893.025		
20	13	260		257.4325		
25	50	1250	24.45	15.125	25.9475	5.09
30	22	660		677.655		
35	3	105		333.9075		
	100	**2445**		**2594.75**		

y_i	n_i	$y_i \cdot n_i$	$\bar{y}$	$(y_i - \bar{y})^2 \cdot n_i$	D_y	σ_y
30	8	240		4014.08		
40	8	320		1230.08		
50	50	2500	52.4	288	110.24	10.5
60	20	1200		1155.2		
70	14	980		4336.64		
	100	**5240**		**11024**		

Коэффициент корреляции вычислим по формуле:

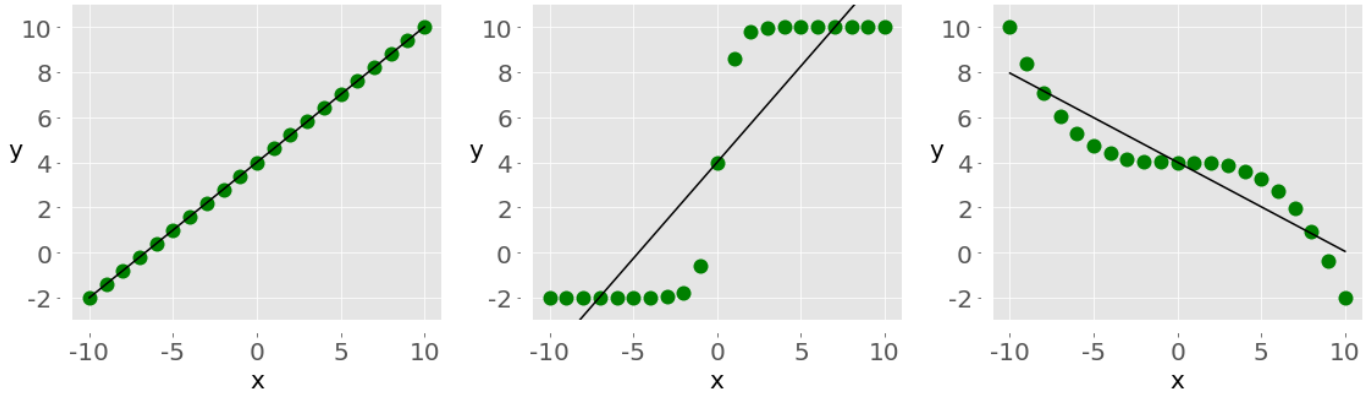
$$r_{\hat{a}} = \frac{\sum n_{xy} x_i y_i - n\overline{x}\,\overline{y}}{n\sigma_x\sigma_y}$$

Получили, что r = 0.754.

Проверим гипотезу о значимости коэффициента корреляции. Введем нулевую гипотезу H_0: r = 0 и вычислим наблюдаемое и критическое значения критерия Стьюдента: t = 11.363, $t_{крит}(98, 0.05) = 1.98$. $t > t_{крит}$, значит связь статистически значима, между X и Y существует линейная зависимость (сильная).

3. *Ранговая корреляция Спирмена*

Ранговая корреляция сравнивает ранги или порядок данных, связанных с двумя переменными или характеристиками набора данных. Если порядок одинаков, то корреляция сильная, положительная и высокая. Однако если порядок близок к противоположному, то корреляция сильная, отрицательная и низкая. Другими словами, ранговая корреляция связана только с порядком значений, а не с конкретными значениями из набора данных.



Коэффициент корреляции Спирмена между двумя признаками — это коэффициент корреляции Пирсона между их ранговыми значениями. Он рассчитывается так же, как и коэффициент корреляции Пирсона, но с учётом рангов, а не значений.

Данный коэффициент - лучший выбор, чем коэффициент корреляции Пирсона, когда справедливо одно или более из следующих положений:

- Переменные являются порядковыми.
- Переменные не распределены нормально.
- Данные включают выбросы.
- Взаимосвязь между переменными является нелинейной и монотонной.

Если все n рангов являются различными целыми числами, коэффициент можно вычислить, используя популярную формулу ($d_i = R(x_i) - R(y_i)$ – разница между двумя рангами каждого наблюдения):

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}$$

Недостатки ранговой корреляции Спирмена:

- Она неприменима в случае сгруппированных данных.
- Она игнорирует немонотонные взаимосвязи между переменными, например, не учитывает другие типы взаимосвязей, такие как криволинейные или нелинейные связи между переменными.
- При преобразовании данных в ранги исходные значения переменных отбрасываются и заменяются соответствующими рангами. Такое преобразование может привести к потере информации в данных, особенно если переменные данных имеют значимые значения или единицы измерения.

Для оценки тесноты связи используют шкалу Чеддока:

Диапазон значений $\lvert r_s \rvert$	Ранговая корреляционная зависимость Y от X
0-0,1	практически отсутствует
0,1-0,3	слабая
0,3-0,5	умеренная
0,5-0,7	заметная
0,7-0,9	сильная
0,9-0,99	очень сильная
0,99-1	практически линейная зависимость рангов

Задача 1. Тринадцать цветных полос расположены в порядке убывания окраски от темной к светлой и каждой полосе присвоен ранг – порядковый номер A. При проверке способности различать оттенки цветов испытуемый расположил полосы в порядке B. Оцените качество цветного зрения испытуемого.

A	1	2	3	4	5	6	7	8	9	10	11	12	13
B	6	3	4	2	1	10	7	8	9	5	11	13	12

Решение.

x_i	1	2	3	4	5	6	7	8	9	10	11	12	13
y_i	6	3	4	2	1	10	7	8	9	5	11	13	12
$d_i=x_i-y_i$	-5	-1	-1	2	4	-4	0	0	0	5	0	-1	1
d_i^2	25	1	1	4	16	16	0	0	0	25	0	1	1
Σd_i^2	90												
r	0.753												

Проверим значимость коэффициента. Вычислим значение t-критерия: t = 3.792, $t_{крит}(11, 0.05) = 2.2$. $t > t_{крит}$, значит связь статистически значима, между X и Y существует монотонная зависимость (сильная).

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

x = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13])
y = np.array([6, 3, 4, 2, 1, 10, 7, 8, 9, 5, 11, 13, 12])
n = len(x)
alpha = 0.05
r, p_value = stats.spearmanr(x, y)
print(r, p_value)

# Принятие решения на основе критического значения t
t_score = r*(n-2)**0.5 / (1-r**2)**0.5
print('t-score:',t_score)
t_critical = stats.t.ppf(1 - alpha/2, df = n-2)
print('Critical t-Score:',t_critical)
if np.abs(t_score) > t_critical:
    print("Связь статистически значима")
else:
```

```
    print("Нельзя отклонить нулевую гипотезу")

# Принятие решения на основе p-значения
p_value = (1 - stats.t.cdf(np.abs(t_score), df = n-2)) * 2
print('P-Value :',p_value)
if p_value < alpha:
    print("Связь статистически значима")
else:
    print("Нельзя отклонить нулевую гипотезу")
```

Задача 2. Три арбитра оценили мастерство 10 спортсменов, в итоге были получены три последовательности рангов (в первой строке приведены ранги арбитра A, во второй – ранги арбитра B, в третьей – ранги арбитра C). Определить пару арбитров, оценки которых наиболее согласуются.

A	1	2	3	4	5	6	7	8	9	10
B	3	10	7	2	8	5	6	9	1	4
C	6	2	1	3	9	4	5	7	10	8

Решение.

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

x = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
y = np.array([3, 10, 7, 2, 8, 5, 6, 9, 1, 4])
z = np.array([6, 2, 1, 3, 9, 4, 5, 7, 10, 8])
r_xy, p_value_xy = stats.spearmanr(x, y)
r_yz, p_value_yz = stats.spearmanr(y, z)
r_zx, p_value_zx = stats.spearmanr(z, x)
print(f'r_xy = {r_xy}, r_yz = {r_yz}, r_zx = {r_zx}')
```

Задача 3. Имеются выборочные данные по студентам: количество прогулов за некоторый период времени и суммарная успеваемость за этот период. Проверить наличие взаимосвязи между факторами.

X	12	9	8	14	15	11	10	15
Y	42	107	100	60	78	79	90	54

Решение.

Ранжируем значения:

X	12	9	8	14	15	11	10	15
R_X	5	2	1	6	7.5	4	3	7.5
Y	42	107	100	60	78	79	90	54
R_Y	1	8	7	3	4	5	6	2

В данном случае есть дробные ранги (это означает, что есть одинаковые значения в выборке). В том случае, если точность вычислений не критична и дробных рангов не так много, можно пользоваться той же формулой, что и для уникальных рангов, но она будет давать приближённый результат. Но если вам необходимы абсолютно точные расчёты, то нужно искать линейный коэффициент корреляции Пирсона, только не между значениями факторов, а между их рангами.

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

x = np.array([12, 9, 8, 14, 15, 11, 10, 15])
y = np.array([42, 107, 100, 60, 78, 79, 90, 54])
n = len(x)
df = pd.DataFrame()
```

```
df['x'] = x
df['y'] = y
print(df)
r = df.corr()['x']['y']
print(f'Корреляция по Пирсону для значений: {r}, t_score = {r*(n-2)**0.5 / (1-r**2)**0.5}')
r = df.corr(method = 'spearman')['x']['y']
print(f'Корреляция по Спирмену для значений: {r}, t_score = {r*(n-2)**0.5 / (1-r**2)**0.5}')
df_ranked = df.rank(ascending=True)
print(df_ranked)
r = df_ranked.corr()['x']['y']
print(f'Корреляция по Пирсону для рангов: {r}, t_score = {r*(n-2)**0.5 / (1-r**2)**0.5}')
r = df_ranked.corr(method = 'spearman')['x']['y']
print(f'Корреляция по Спирмену для рангов: {r}, t_score = {r*(n-2)**0.5 / (1-r**2)**0.5}')

alpha = 0.03
t_critical = stats.t.ppf(1 - alpha/2, df = n-2)
print('Critical t-Score:',t_critical)

r, p_value = stats.spearmanr(x, y)
print(r, p_value)
```

4. Корреляция Фехнера «да-нет»

Коэффициент корреляции Фехнера тоже является ранговым коэффициентом, но в отличие от коэффициента Спирмена использует лишь два ранга: «да-нет», «больше / меньше среднего».

Формула расчета коэффициента корреляции Фехнера (n_a – число совпадений знаков отклонений индивидуальных величин от средней, $n_н$ – число несовпадений знаков отклонений):

$$r = \frac{n_a - n_b}{n_a + n_b}$$

Значимость оценивается с помощью той же формулы t-критерия, как в предыдущих случаях.

Задача 1. Среди 50 семейных пар был проведён опрос: «Как вы относитесь к кошке в доме?». Допустимы лишь два ответа: положительный либо отрицательный; нейтральная позиция засчитывается положительно («ничего не имею против»). В результате исследования выяснилось, что в 37 парах мнения супругов совпадают, а в 13 парах – нет (т. е. кто-то «за», а кто-то «против»). Оцените зависимость семейного счастья от согласия по этому вопросу.

Решение.

Коэффициент корреляции Фехнера: (37-13)/(37+13) = 0.48. Расчет t-статистики: 3.79

$t_{крит}(48, 0.05) = 2.01$. $t > t_{крит}$, значит связь статистически значима, между X и Y существует зависимость (умеренная).

Задача 2. Имеются выборочные данные по студентам: количество прогулов за некоторый период времени и суммарная успеваемость за этот период. Проверить наличие взаимосвязи между факторами.

X	12	9	8	14	15	11	10	15
Y	42	107	100	60	78	79	90	54

Решение.

Сначала найдём средние значения: $\bar{x} = 11.75, \bar{y} = 76.25$. После этого каждому значению присваиваем свой ранг:

– если больше либо равно среднему, то ставим «плюс»;

– если меньше, чем среднее, то «минус».

Теперь в каждой паре нужно сравнить ранги: если знаки совпадают, то ставим единичку, а если не совпадают – то ноль. В чём логика такого ранжирования и сравнения? Если студент прогуливает меньше среднего, то по идее, его успеваемость должна быть выше средней. И наоборот, если много прогуливает – то и успевает хуже.

X	12	9	8	14	15	11	10	15
Y	42	107	100	60	78	79	90	54

X_{ranked}	+	-	-	+	+	-	-	+
Y_{ranked}	-	+	+	-	+	+	+	-
	0	0	0	0	1	0	0	0

Коэффициент корреляции Фехнера: (1-7)/(1+7) = -0.75.

Таким образом, существует сильная обратная корреляционная зависимость суммарной успеваемости от количества прогулов.

5. Корреляция и причинность

Понимание разницы между корреляцией и причинно-следственной связью может иметь огромное значение, особенно когда кто-то принимает решение о чём-то, что может быть ошибочным.

Корреляция — это взаимосвязь между переменными: когда меняется одна переменная, меняется и другая. Корреляция — это статистическая мера связи между переменными. Эти переменные изменяются одинаково: они коррелируют. Но эта корреляция не обязательно обусловлена внешней или дополнительной причинно-следственной связью.

Причинно-следственная связь означает, что изменения в одной переменной приводят к изменениям в другой; между переменными существует причинно-следственная связь.

Пример. Мой отец только что пожаловался мне: «Во время написания текстового сообщения клавиатура моего телефона зависает». Связана ли эта проблема с причинно-следственной связью или корреляцией?

Ответ. В смартфоне отца одновременно было открыто четыре игровых приложения, а также WhatsApp и YouTube. Отправка текстового сообщения не вызывала зависание, а была вызвана нехваткой оперативной памяти. Но отец мгновенно связал это с последним действием, которое он совершил перед зависанием. Таким образом отец указывал на причинно-следственную связь там, где была только корреляция: **Недостаток оперативной памяти ⇢ Причинно-следственная связь ⇢ Телефон зависает ⇢ Корреляция ⇢ Текстовые сообщения не работают.**

Одним из способов определения причинно-следственной связи в статистике являются тест Грейнджера и конвергентное кросс-сопоставление.

Согласно тесту Грейнджера, временной ряд X является причиной ряда Y, если можно показать, обычно с помощью серии t-тестов и F-тестов, на запаздывающих значениях X (а также с запаздывающими значениями Y), что эти значения X предоставляют статистически значимую информацию о будущих значениях Y. Фактически это означает, что прогнозы значения Y, основанные на её собственных прошлых значениях и на прошлых значениях X, лучше, чем прогнозы Y, основанные только на собственных прошлых значениях Y.

Чтобы проверить нулевую гипотезу о том, что X не является причиной Y по Грейнджеру, сначала нужно найти подходящие запаздывающие значения Y, которые можно включить в одномерную авторегрессию Y:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_m y_{t-m} + \mathrm{error}_t$$

Далее авторегрессия дополняется включением запаздывающих значений X:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_m y_{t-m} + b_p x_{t-p} + \cdots + b_q x_{t-q} + \mathrm{error}_t$$

В этой регрессии сохраняются все запаздывающие значения X, которые являются значимыми по отдельности в соответствии с их t-статистикой, при условии, что в совокупности они повышают объясняющую способность регрессии в соответствии с F-тестом (нулевая гипотеза которого заключается в отсутствии объясняющей способности). В обозначениях приведённой выше расширенной регрессии p — это наименьшая, а q — наибольшая длина запаздывания, при которой запаздывающее значение X является значимым.

Нулевая гипотеза о том, что X не является причиной Y по Грейнджеру, не отвергается, если и только если в регрессии не используются запаздывающие значения X.

Приведённый линейный метод подходит для проверки причинно-следственной связи по Грейнджеру в среднем значении. Однако он не способен выявить причинно-следственную связь по Грейнджеру в более высоких моментах, например, в дисперсии. Непараметрические тесты на причинно-следственную связь по Грейнджеру предназначены для решения этой проблемы. Определение причинно-следственной связи по Грейнджеру в этих тестах является общим и не предполагает каких-либо допущений о моделировании, например, линейной авторегрессионной модели. Непараметрические тесты причинно-следственных связей

по Грейнджеру можно использовать в качестве диагностических инструментов для построения более качественных параметрических моделей, включая моменты более высокого порядка и/или нелинейность. При этом важно помнить, что тесты причинно-следственных связей по Грейнджеру предназначены для работы с парами переменных и могут давать неверные результаты, когда истинная взаимосвязь включает три или более переменных.

Задача. В вашем распоряжении имеются помесячные данные (файл advert.xslx) за период с января 2001 по декабрь 2010 г. о расходах на рекламу и объемах продаж регионального подразделения транснациональной корпорации. Осуществите для рассматриваемых переменных тест Грейнджера на причинно-следственную связь (используйте первый, второй и третий лаги для каждой из переменных). Интерпретируйте результаты теста.

```
import numpy as np
import pandas as pd
import statsmodels.api as sm
from statsmodels.tsa.stattools import grangercausalitytests

df_1 = pd.read_excel("advert.xlsx") # x - расходы фирмы на рекламу, y — объем продаж фирмы
df_1.head(5)

r = df_1.corr(method = 'spearman')['x']['y']
print(r)

df_2 = df_1.copy()
df_2['x']=df_1['y']
df_2['y']=df_1['x']
# grangercausalitytests проверяет, влияет ли временной ряд во втором столбце на временной ряд в первом столбце
gc_res_1 = grangercausalitytests(df_1, 3) # влияет ли объем продаж фирмы на расходы фирмы на рекламу
for lag, result in gc_res_1.items():
    p_values = [round(value[1],4) for key, value in result[0].items()]
    print(f"Lag: {lag} \nF-Statistic: {result[0]['ssr_ftest'][0]} \nP-Value: {p_values[0]}\n")

gc_res_2 = grangercausalitytests(df_2, 3) # влияют ли расходы фирмы на рекламу на объем продаж фирмы
for lag, result in gc_res_2.items():
    p_values = [round(value[1],4) for key, value in result[0].items()]
    print(f"Lag: {lag} \nF-Statistic: {result[0]['ssr_ftest'][0]} \nP-Value: {p_values[0]}\n")

# Если значение p меньше вашего уровня значимости (обычно 0,05), вам следует отклонить нулевую гипотезу и сделать вывод,
# что временной ряд во втором столбце по Грейнджеру влияет на временной ряд в первом столбце.
```

Регрессионный анализ

1. Основные понятия

Регрессия - тип задачи машинного обучения, направленная на предсказание неизвестной величины по имеющимся данным.

Допущения при использовании регрессионного анализа:

- Переменные имеют распределение, близкое к нормальному (проверяется тестом Шапиро-Уилка);
- Переменные приведены в единую измерительную систему, несущую смысл (безразмерные величины, единицы СИ);
- Поскольку мы рассматриваем конкретный простейший тип регрессии - линейную, то переменные должны быть связаны линейной функцией;
- Отсутствие мультиколлинеарности, то есть известные переменные не зависят друг от друга;
- Отсутствие автокорреляции, то есть отсутствие независимости остатков (выявляется с помощью теста Дурбина-Уотсона);
- Гомоскедастичность (равенство дисперсий остатков для каждого из значений, можно проверить на диаграмме рассеяния или тестом Гольдфельда — Квандта).

Довольно часто при описании аппроксимирующей функции ограничиваются простым видом полиноминальной зависимости, полагая ее линейной, т.е. в виде уравнения прямой $y = b_0 + b_1x$. Здесь свободный член b_0 характеризует сдвиг и равен тому значению y, которое получается при $x = 0$, а коэффициент b_1 определяет наклон линии.

Поиск коэффициентов b_0 и b_1 осуществляется по методу наименьших квадратов (МНК).

В соответствии с МНК полагаем, что искомая прямая будет наилучшей, если сумма квадратов всех расстояний $(b_0 + b_1x_i - y_i)^2 = \varepsilon_i^2$ окажется наименьшей, то есть сумма квадратов отклонений ε_i экспериментальных точек от кривой по вертикальному направлению должна быть наименьшей.

Особенности МНК:

1. Этот метод не дает ответа на вопрос о том, какого вида функция лучше всего аппроксимирует конкретные экспериментальные точки. Вид интересующей нас функции должен быть задан на основе каких-то физических или экономических соображений (либо специальным образом отыскан). МНК позволяет лишь выбрать, какая из прямых (парабол, экспонент) является лучшей прямой (параболой, экспонентой) для прогнозирования.

2. Вычисления по МНК являются достаточно громоздкими, поэтому основная нагрузка − на компьютерные программы.

3. МНК является достаточно точным приемом и позволяет получить вполне надежные результаты. Одновременно он является интерполяционным методом, поскольку обеспечивает с определенной вероятностью предсказание любых значений y_i в интервале изученных значений x_i, в отличие от экстраполяционного метода, который дает возможность предсказывать результаты за пределами изученной области.

После того как уравнение регрессии найдено, необходимо определить его статистическую пригодность, т.е. выяснить, насколько оно верно (надежно) предсказывает в интервале $x_1; x_2; \ldots x_n$ экспериментальные результаты для y. Подобную оценку принято называть проверкой на значимость или адекватность.

2. Парная линейная регрессия

Задача 1. В таблице приведены данные зависимости потребления Y (усл. ед.) от дохода X (усл. ед.) для некоторых домашних хозяйств.

1. В предположении, что между Y и X существует линейная зависимость, найдите точечные оценки коэффициентов линейной регрессии.

2. В предположении нормальности случайной составляющей регрессионной модели проверьте гипотезу об отсутствии линейной зависимости между Y и X .

3. Каково ожидаемое потребление Y_0 домашнего хозяйства с доходом $X_0 = 7$ усл. ед.? Найдите доверительный интервал для прогноза.

Дайте интерпретацию полученных результатов. Уровень значимости считать равным $\alpha = 0{,}05$.

X	2.2	2.8	3.0	3.5	3.9	4.2	4.4	4.8	5.1	5.5
Y	0.13	0.21	0.28	0.26	0.23	0.29	0.28	0.24	0.31	0.33

Решение.

По результатам исследования ниже можно сделать вывод, что между доходами и объемом потребления существует прямая линейная связь, которая является достаточно тесной (коэффициент корреляции 0,787). Доля дисперсии признака Y в общей дисперсии Y, объясненную регрессией Y по X, которая выражается через коэффициент детерминации 0,619, говорит о том, что линейная модель адекватна статистике на 61,9%.

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

x = np.array([2.2, 2.8, 3.0, 3.5, 3.9, 4.2, 4.4, 4.8, 5.1, 5.5])
y = np.array([0.13, 0.21, 0.28, 0.26, 0.23, 0.29, 0.28, 0.24, 0.31, 0.33])
fig, ax = plt.subplots(figsize=(6, 4))
ax.scatter(x, y)
plt.show()

# Расчет корреляции
alpha = 0.05
r, p_value = stats.pearsonr(x, y)
print(f'Коэффициент корреляции = {r}, p_value = {p_value}')
if p_value < alpha:
    print("Связь статистически значима")
else:
    print("Нельзя отклонить нулевую гипотезу")
# существует линейная зависимость между доходами и объемом потребления, коэффициент корреляции статистически значим

# расчет линейной регрессии
res = stats.linregress(x, y)
print(f'Коэффициент наклона {res.slope}, сдвиг {res.intercept}')
print(f'Коэффициент корреляции {res.rvalue}, коэффициент детерминации {res.rvalue**2}, p-value (из теста Вальда в предположении, что наклон равен нулю) {res.pvalue}')
print(f'Ошибка наклона {res.stderr}, ошибка сдвига {res.intercept_stderr}')
print(f' Y = {round(res.intercept, 3)} + {round(res.slope, 3)}*X')

# визуализация прогноза
y_pred = res.intercept + res.slope*x
fig, ax = plt.subplots(figsize=(6, 4))
ax.scatter(x, y)
ax.scatter(x, y_pred)
plt.show()

# расчет доверительных интервалов для параметров и прогноза
from scipy.stats import t
t_coef = abs(t.ppf(alpha/2, len(x)-2))
t_real = abs(t.ppf(res.pvalue/2, len(x)-2))
print(f'Расчетное значение t-критерия: {t_real}, критическое значение t-критерия: {t_coef}')
print(f"slope (95%): {res.slope:.3f} +/- {t_coef*res.stderr:.3f}")
print(f"intercept (95%): {res.intercept:.3f} +/- {t_coef*res.intercept_stderr:.3f}")

x_0 = 7
y_0 = res.intercept + res.slope*x_0
y_r = (y_pred-y)**2
s_r = (y_r.sum()/(len(x)-2))**0.5
delta = s_r*(1+1/len(x)+(x_0-x.mean())**2/(len(x)*x.var()))
print(f"При x_0 = 7: y_0 = {y_0:.3f} +/- {t_coef*delta:.3f}")
```

Задача 2. Вычислить коэффициент уравнения регрессии. Определить выборочный коэффициент корреляции между плотностью X древесины маньчжурского ясеня (кг/м^3) и его прочностью Y (МПа).

X	0.69	0.68	0.65	0.74	0.72	0.66	0.72	0.72	0.72	0.75	0.76	0.62	0.69	0.72	0.66	0.69	0.65	0.68	0.71	0.67
Y	53.9	48.9	46.8	52.2	53.5	48.7	49.2	53.7	53.2	57.6	58.3	45.7	49.6	49.4	49.5	51.7	43.6	51.5	46.6	43.8

Решение.

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

x = np.array([0.69, 0.68, 0.65, 0.74, 0.72, 0.66, 0.72, 0.72, 0.72, 0.75, 0.76, 0.62, 0.69, 0.72, 0.66, 0.69, 0.65, 0.68, 0.71, 0.67])
y = np.array([53.9, 48.9, 46.8, 52.2, 53.5, 48.7, 49.2, 53.7, 53.2, 57.6, 58.3, 45.7, 49.6, 49.4, 49.5, 51.7, 43.6, 51.5, 46.6, 43.8])
fig, ax = plt.subplots(figsize=(6, 4))
ax.scatter(x, y)
plt.show()

# Расчет корреляции
alpha = 0.05
r, p_value = stats.pearsonr(x, y)
print(f'Коэффициент корреляции = {r}, p_value = {p_value}')
if p_value < alpha:
    print("Связь статистически значима")
else:
    print("Нельзя отклонить нулевую гипотезу")
# существует линейная зависимость между плотностью дерева и его прочностью, коэффициент корреляции статистически значим

# расчет линейной регрессии
res = stats.linregress(x, y)
print(f'Коэффициент наклона {res.slope}, сдвиг {res.intercept}')
print(f'Коэффициент корреляции {res.rvalue}, коэффициент детерминации {res.rvalue**2}, p-value (из теста Вальда в предположении, что наклон равен нулю) {res.pvalue}')
print(f'Ошибка наклона {res.stderr}, ошибка сдвига {res.intercept_stderr}')
print(f' Y = {round(res.intercept, 3)} + {round(res.slope, 3)}*X')

# визуализация прогноза
y_pred = res.intercept + res.slope*x
fig, ax = plt.subplots(figsize=(6, 4))
ax.scatter(x, y)
ax.scatter(x, y_pred)
plt.show()

# расчет доверительных интервалов для параметров и прогноза
from scipy.stats import t
t_coef = abs(t.ppf(alpha/2, len(x)-2))
t_real = abs(t.ppf(res.pvalue/2, len(x)-2))
print(f'Расчетное значение t-критерия: {t_real}, критическое значение t-критерия: {t_coef}')
print(f"slope (95%): {res.slope:.3f} +/- {t_coef*res.stderr:.3f}")
print(f"intercept (95%): {res.intercept:.3f} +/- {t_coef*res.intercept_stderr:.3f}")
```

3. *Тесты Дурбина-Уотсона, Шапиро-Уилка и Гольдфельда — Квандта*

А) Тест Дурбина-Уотсона

В регрессионном анализе критерий Дурбина-Уотсона (DW) используется для проверки автокорреляции первого порядка (последовательной корреляции):

$$\varepsilon_i = \rho\varepsilon_{i-1} + \delta_i, \quad i = \overline{2,n}$$

$$DW_{calc} = \frac{\sum_{i=2}^{n}(\varepsilon_i - \varepsilon_{i-1})^2}{\sum_{i=1}^{n}{\varepsilon_i}^2}$$

Он анализирует остатки на предмет независимости во времени (автокорреляции). Автокорреляция варьируется от -1 (отрицательная автокорреляция) до 1 (положительная автокорреляция).

Тест Дурбина-Ватсона анализирует следующие гипотезы:

- Нулевая гипотеза (H_0): остатки от регрессии не автокоррелированы (коэффициент $\rho = 0$)
- Альтернативная гипотеза (H_A): остатки от регрессии автокоррелированы (коэффициент автокорреляции $\rho > 0$)

Тестовая статистика Дурбина-Уотсона приблизительно равна 2*(1-r), где r - выборочная автокорреляция остатков. Следовательно, для r = 0 справедливо отсутствие последовательной корреляции, и тестовая статистика равна 2. Значение статистики всегда будет находиться в диапазоне от 0 до 4. Чем ближе значение статистики к 0, тем больше вероятность положительной последовательной корреляции. Чем ближе к 4, тем больше вероятность отрицательной последовательной корреляции.

Правила принятия гипотез по критерию Дурбина-Уотсона выглядят довольно своеобразно - критические значения образуют пять областей различных статистических решений (причем критические границы принятия H_0 и непринятия H_A не совпадают):

Значение DW_{calc}	Принимается гипотеза	Вывод
$0 \le DW_{calc} < d_L$	отвергается H_0 , принимается $H_1 : \rho > 0$	есть положительная автокорреляция
$d_L \le DW_{calc} \le d_U$		неопределенность
$d_U < DW_{calc} < 4 - d_U$	принимается H_0	автокорреляция отсутствует
$4 - d_U \le DW_{calc} \le 4 - d_L$		неопределенность
$4 - d_L < DW_{calc} \le 4$	отвергается H_0 , принимается $H_1 : \rho < 0$	есть отрицательная автокорреляция

Подробные таблицы для критерия Дурбина-Уотсона можно найти в книге Фёрстера Э. и Рёнца Б. «Методы корреляционного и регрессионного анализа» (пер с нем. - М.: Финансы и статистика, 1983. - 302 с.) на стр. 290-292 (m – число параметров регрессионной модели кроме свободного члена, T – размер выборки, нижний d_L и верхний d_U пределы).

Критические значения статистики Дарбина — Уотсона при 5%-ном уровне значимости

T	m									
	1		2		3		4		5	
	$d_н$	$d_в$	$d_н$	$d_в$	$d_н$	$d_в$	$d_н$	$d_в$	$d_н$	$d_в$
15	1,08	1,36	0,95	1,54	0,82	1,75	0,69	1,97	0,56	2,21
16	1,10	1,37	0,98	1,54	0,86	1,73	0,74	1,93	0,62	2,15
17	1,13	1,38	1,02	1,54	0,90	1,71	0,78	1,90	0,67	2,10
18	1,16	1,39	1,05	1,53	0,93	1,69	0,82	1,87	0,71	2,06
19	1,18	1,40	1,08	1,53	0,97	1,68	0,86	1,85	0,75	2,02
20	1,20	1,41	1,10	1,54	1,00	1,68	0,90	1,83	0,79	1,99
21	1,22	1,42	1,13	1,54	1,03	1,67	0,93	1,81	0,83	1,96
22	1,24	1,43	1,15	1,54	1,05	1,66	0,96	1,80	0,86	1,94
23	1,26	1,44	1,17	1,54	1,08	1,66	0,99	1,79	0,90	1,92
24	1,27	1,45	1,19	1,55	1,10	1,66	1,01	1,78	0,93	1,90
25	1,29	1,45	1,21	1,55	1,12	1,66	1,04	1,77	0,95	1,89

Применим критерий Дурбина-Уотсона к задаче №2 из второго пункта.

```
# тест Дурбина-Уотсона на отсутствие независимости остатков
from statsmodels.stats.stattools import durbin_watson
y_r = (y-y_pred)
dw = durbin_watson(y_r)
print(dw) # 1.7344
```

Полученное значение попадает в диапазон, в котором принимается H_0, т.е. автокорреляция остатков отсутствует.

Б) Тест Шапиро-Уилка

Тест Шапиро-Уилка — это тест на нормальность, он определяет, относится ли данная выборка к нормальному распределению или нет. Тест Шапиро-Уилка или тест Шапиро — это тест на нормальность в частотной статистике. Нулевая гипотеза теста Шапиро состоит в том, что популяция распределена нормально:

- H_0: Выборка из нормального распределения.
- H_A: Выборка не соответствует нормальному распределению.

Принцип критерия основан на отношении оптимальной линейной несмещённой оценки дисперсии к её обычной оценке методом максимального правдоподобия.

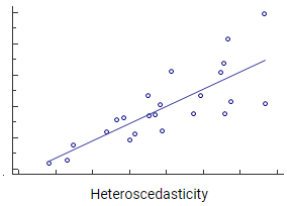
Применим критерий Шапиро-Уилка к задаче №2 из второго пункта.

```
# тест Шапиро-Уилка на нормальность данных
from scipy import stats
stats_x, p_value_x = stats.shapiro(x)
stats_y, p_value_y = stats.shapiro(y)
print(f'Для x: статистика = {stats_x}, p-value = {p_value_x}')

print(f'Для x: статистика = {stats_y}, p-value = {p_value_y}')

# Принятие решения на основе p-значения
if p_value_x < alpha:
    print("Выборка X не нормальна")
else:
    print("Нельзя отклонить нулевую гипотезу о нормальности для X")
if p_value_y < alpha:
    print("Выборка Y не нормальна")
else:
    print("Нельзя отклонить нулевую гипотезу о нормальности для Y")
```

Моделирование методом Монте-Карло показало, что критерий Шапиро-Уилка обладает наибольшей мощностью при заданной значимости среди всех критериев нормальности данных.

В) Тест Гольдфельда — Квандта

Тест Голдфельда-Квандта – это тест для проверки наличия гетероскедастичности в заданных данных. Гетероскедастичность — это ситуация, при которой изменчивость независимой переменной (Y) неравномерна в диапазоне значений объясняющей переменной (X). Если построить диаграмму рассеяния между двумя переменными, она будет иметь конусообразную форму. По мере увеличения значения X разброс переменной Y расширяется или сужается, создавая конусообразную структуру. Например, когда мы пытаемся предсказать годовой доход на основе возраста человека, годовой доход может быть гетероскедастической переменной, поскольку, когда кто-то начинает как новичок, получаемый доход меньше по сравнению с тем, кто имеет опыт за эти годы, и повышение зарплаты может иметь очевидное количество вариаций.



Предположение теста Гольдфельда-Квандта основано на том, что данные распределены нормально.

Нулевая и альтернативная гипотезы теста Гольдфельда-Квандта:

- Нулевая гипотеза H_0: гетероскедастичность отсутствует.
- Альтернативная гипотеза H_A: присутствует гетероскедастичность.

Тест разбивает данные на две группы, исключая определенную долю наблюдений в середине. Затем он выполняет отдельные регрессии для этих двух групп и сравнивает отношение остаточной суммы квадратов (RSS). Если соотношение существенно отличается от единицы, это свидетельствует о том, что отклонения членов ошибки не являются постоянными, и вы можете отклонить нулевую гипотезу гомоскедастичности.

Применим критерий Гольдфельда-Квандта к задаче №2 из второго пункта.

```
# тест Гольдфельда-Квандта на гетероскедастичность
import pandas as pd
import statsmodels.api as sm
from statsmodels.stats.diagnostic import het_goldfeldquandt

alpha = 0.05
x = sm.add_constant(x)
model = sm.OLS(y, x).fit()
gq_test = het_goldfeldquandt(y, x)
print(f'F-статистика = {gq_test[0]}, p-value = {gq_test[1]}')
# Принятие решения на основе p-значения
if gq_test[1] < alpha:
    print("Присутствует гетероскедастичность")
else:
    print("Нельзя отклонить нулевую гипотезу о гомоскедастичности данных")
```

Другой способ проверить гетероскедастичность - визуализировать остатки. Если дисперсия остатков изменяется в зависимости от значений независимой переменной, это может быть признаком гетероскедастичности.

```
# визуализация остатков
import matplotlib.pyplot as plt
import numpy as np

# Calculate the residuals
y_prog = model.predict(x)
y_resud = model.resid - y_prog

# Create a scatter plot
plt.scatter(y_prog, y_resud)
plt.xlabel('Predicted')
plt.ylabel('Residuals')
plt.axhline(y=0, color='r', linestyle='-')
plt.title('Residuals vs. Predicted')
plt.show()
```

Если вы наблюдаете на графике воронкообразную форму (остатки расширяются по мере увеличения прогнозируемого значения), это признак гетероскедастичности.

Если вы обнаружите доказательства гетероскедастичности, вы можете применить несколько стратегий:

- Преобразование зависимой переменной: логарифмические преобразования или преобразования квадратного корня иногда могут помочь стабилизировать дисперсию.
- Использование модели взвешенной регрессии: это придает меньший вес наблюдениям с большей дисперсией.
- Изменение спецификации модели: иногда проблему могут решить нелинейные зависимости или эффекты взаимодействия.
- Использование надежных стандартных ошибок: они разработаны таким образом, чтобы быть допустимыми даже при наличии гетероскедастичности.

4. Множественная линейная регрессия

Задача. Пусть имеются следующие данные (условные) о сменной добыче угля на одного рабочего y (т), мощности пласта x_1 (м) и уровне механизации работ x_2 (%), характеризующие процесс добычи угля в 10 шахтах. Требуется построить уравнение множественной регрессии и исследовать его.

x1	8	11	12	9	8	8	9	9	8	12
x2	5	8	8	5	7	8	6	4	5	7
y	5	10	10	7	5	6	6	5	6	8

Решение.

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
import pandas as pd

x1 = np.array([8, 11, 12, 9, 8, 8, 9, 9, 8, 12])
x2 = np.array([5, 8, 8, 5, 7, 8, 6, 4, 5, 7])
y = np.array([5, 10, 10, 7, 5, 6, 6, 5, 6, 8])

df = pd.DataFrame()
df['x1']=x1
df['x2']=x2
df['y']=y

df.corr()

# решение через statsmodels
import statsmodels.api as sm
import numpy as np
y = [5, 10, 10, 7, 5, 6, 6, 5, 6, 8]
x = [
    [8, 11, 12, 9, 8, 8, 9, 9, 8, 12],
    [5, 8, 8, 5, 7, 8, 6, 4, 5, 7],
]
def reg_m(y, x):
    ones = np.ones(len(x[0]))
    X = sm.add_constant(np.column_stack((x[0], ones)))
    for ele in x[1:]:
        X = sm.add_constant(np.column_stack((ele, X)))
    results = sm.OLS(y, X).fit()
    return results
print(reg_m(y, x).summary())

'''
```

Характеристика уравнения как целого:

1. R-squared (R-квадрат) - индекс (коэффициент) детерминации (множественной корреляции)
2. Adj. R-squared (скорректированный R-квадрат) - индекс (коэффициент) детерминации (множественной корреляции), скорректированный (нормированный) на число степеней свободы остатков
3. F-statistic - F-критерий Фишера для оценки значимости уравнения множественной регрессии в целом
4. Prob (F-statistic) - значение уровня значимости, соответствующее вычисленной величине F-критерия, если он меньше уровня значимости, то построенная регрессия является значимой

Характеристика коэффициентов регрессии:

1. coef - значение коэффициента
2. std err - стандартное отклонение коэффициента
3. t - t-критерий Стьюдента для оценки статистической значимости параметра

4. P>|t| - вероятности событий непревышения расчетной статистикой Стьюдента для соответствующего параметра регрессии табличного значения, если она меньше уровня значимости, то принимается гипотеза о значимости соответствующего коэффициента регрессии

'''

```
# решение через numpy.linalg.lstsq
import numpy as np
y = [5, 10, 10, 7, 5, 6, 6, 5, 6, 8]
x = [
    [8, 11, 12, 9, 8, 8, 9, 9, 8, 12],
    [5, 8, 8, 5, 7, 8, 6, 4, 5, 7],
]
x = np.transpose(x)  # transpose so input vectors
x = np.c_[x, np.ones(x.shape[0])]  # add bias term
linreg = np.linalg.lstsq(x, y, rcond=None)[0]
print(linreg) # Оптимальные значения параметров

# решение через scipy.curve_fit()
from scipy.optimize import curve_fit
import scipy
import numpy as np
def function_calc(x, a, b, c):
    return a + b * x[0] + c * x[1]
y = [5, 10, 10, 7, 5, 6, 6, 5, 6, 8]
x = [
    [8, 11, 12, 9, 8, 8, 9, 9, 8, 12],
    [5, 8, 8, 5, 7, 8, 6, 4, 5, 7],
]
popt, pcov = curve_fit(function_calc, x, y)
print(popt) # Оптимальные значения параметров
print(pcov) # Расчетная приблизительная ковариация параметров. Диагонали показывают дисперсию оценки параметра.
print(np.sqrt(np.diag(pcov))) # стандартные отклонения параметров

# расчет стандартных коэффициентов регрессии
x1 = np.array([8, 11, 12, 9, 8, 8, 9, 9, 8, 12])
x2 = np.array([5, 8, 8, 5, 7, 8, 6, 4, 5, 7])
y = np.array([5, 10, 10, 7, 5, 6, 6, 5, 6, 8])
x1 = (x1-x1.mean())/(x1.var()**0.5)
x2 = (x2-x2.mean())/(x2.var()**0.5)
y = (y-y.mean())/(y.var()**0.5)
x = [x1, x2]
def reg_m(y, x):
    ones = np.ones(len(x[0]))
    X = sm.add_constant(np.column_stack((x[0], ones)))
    for ele in x[1:]:
        X = sm.add_constant(np.column_stack((ele, X)))
    results = sm.OLS(y, X).fit()
    return results
print(reg_m(y, x).summary())
```

5. Нелинейная регрессии

Бывают случаи, когда визуально понятно, что зависимость между переменными, хоть и неплохо описывается как линейная, на самом деле таковой не является. В этом случае можно попробовать «угадать» зависимость с тем, чтобы лучше описывать взаимосвязь между переменными.

Задача. Для набора данных предложите вид зависимости Y(X)

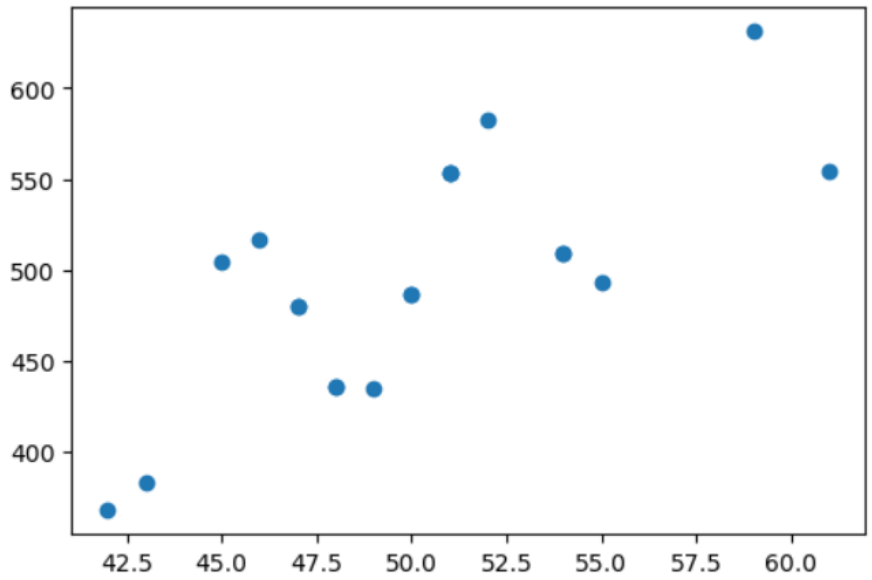
X = [46, 51, 61, 59, 48, 50, 47, 52, 43, 50, 51, 48, 45, 47, 54, 42, 54, 51, 55, 49]

Y = [517, 553, 554, 631, 436, 487, 480, 582, 383, 487, 553, 436, 504, 480, 509, 368, 509, 553, 493, 435]

Решение.

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

x = np.array([46, 51, 61, 59, 48, 50, 47, 52, 43, 50, 51, 48, 45, 47, 54, 42, 54, 51, 55, 49])
y = np.array([517, 553, 554, 631, 436, 487, 480, 582, 383, 487, 553, 436, 504, 480, 509, 368, 509, 553, 493, 435])
alpha = 0.05

# тест Шапиро-Уилка на нормальность данных
stats_x, p_value_x = stats.shapiro(x)
stats_y, p_value_y = stats.shapiro(y)
if p_value_x < alpha:
    print("Выборка X не нормальна")
else:
    print("Нельзя отклонить нулевую гипотезу о нормальности для X")
if p_value_y < alpha:
    print("Выборка Y не нормальна")
else:
    print("Нельзя отклонить нулевую гипотезу о нормальности для Y")

# тест Гольдфельда-Квандта на гетероскедастичность
import pandas as pd
import statsmodels.api as sm
from statsmodels.stats.diagnostic import het_goldfeldquandt
alpha = 0.05
x = sm.add_constant(x)
model = sm.OLS(y, x).fit()
gq_test = het_goldfeldquandt(y, x)
# Принятие решения на основе p-значения
if gq_test[1] < alpha:
    print("Присутствует гетероскедастичность")
else:
    print("Нельзя отклонить нулевую гипотезу о гомоскедастичности данных")

# визуализация
fig, ax = plt.subplots(figsize=(6, 4))
x = np.array([46, 51, 61, 59, 48, 50, 47, 52, 43, 50, 51, 48, 45, 47, 54, 42, 54, 51, 55, 49])
y = np.array([517, 553, 554, 631, 436, 487, 480, 582, 383, 487, 553, 436, 504, 480, 509, 368, 509, 553, 493, 435])
ax.scatter(x, y)
plt.show()

# предположение о линейной регрессии
res = stats.linregress(x, y)
print(f'Коэффициент наклона {res.slope}, сдвиг {res.intercept}')
print(f'Коэффициент корреляции {res.rvalue}, коэффициент детерминации {res.rvalue**2}, p-value (из теста Вальда в предположении, что наклон равен нулю) {res.pvalue}')
print(f'Ошибка наклона {res.stderr}, ошибка сдвига {res.intercept_stderr}')
print(f' Y = {round(res.intercept, 3)} + {round(res.slope, 3)}*X')
```

Глядя на график зависимости Y от X и результаты линейной регрессии понятно, что, хотя регрессия неплоохо описывает данные, улучшения все же необходимы




```
Коэффициент наклона 9.782779084089128, сдвиг 6.893628932930255
Коэффициент корреляции 0.7295829530452735, коэффициент детерминации 0.5322912853742618
p-value (из теста Вальда в предположении, что наклон равен нулю) 0.0002615156106691978
Ошибка наклона 2.1614194079923217, ошибка сдвига 108.8752676354131
 Y = 6.894 + 9.783*X
```

Попробуем изменить функциональную зависимость, фактически создав ситуацию множественной регрессии

```
import statsmodels.api as sm
import numpy as np
y = np.array([517, 553, 554, 631, 436, 487, 480, 582, 383, 487, 553, 436, 504, 480, 509, 368, 509, 553, 493, 435])
x1 = np.array([46, 51, 61, 59, 48, 50, 47, 52, 43, 50, 51, 48, 45, 47, 54, 42, 54, 51, 55, 49])
x2 = np.sin(x1)
x = [x1, x2]
def reg_m(y, x):
    ones = np.ones(len(x[0]))
    X = sm.add_constant(np.column_stack((x[0], ones)))
    for ele in x[1:]:
        X = sm.add_constant(np.column_stack((ele, X)))
    results = sm.OLS(y, X).fit()
    return results
print(reg_m(y, x).summary())
```