

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»
Институт компьютерных наук и технологий
Высшая школа искусственного интеллекта



ПОЛИТЕХ
Санкт-Петербургский
политехнический университет
Петра Великого

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине «Элементы теории вероятности и линейной алгебры»

Вариант №11

Выполнил
Студент 3540201/10301 группы

Ф.М. Титов

Руководитель
доцент, к.т.н.

А.В. Востров

Санкт-Петербург
2021 г.

Оглавление

Постановка задачи.....	3
Теоретическая часть.....	4
Реализация.....	7
Результаты.....	10
Заключение	13

Постановка задачи

Найти оценки параметров линейной регрессии y на x , доверительные интервалы для параметров и линии регрессии и проверить согласие линейной регрессии с результатами наблюдений. Принять уровень доверительной вероятности равным 0.90.

Значения для варианта 11:

x	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55
y	2.43	2.67	2.71	3.15	3.47	3.76	3.91	4.46	4.76	5.15	5.54	5.61

Теоретическая часть

Линейная регрессионная модель имеет вид:

$$y_i = \alpha + \beta x_i + \varepsilon_i, \quad i = \overline{1, n},$$

где ε_i - ошибки измерений переменной y предполагаются независимыми случайными величинами, распределенными нормально: $\varepsilon_i \in N(0, D_\varepsilon)$. Наша задача состоит в том, чтобы по наблюдениям найти оценки $\alpha = a^*$, $\beta = b^*$ и $s^2 = D^*$ для параметров α , β и D соответственно.

Для оценки параметров необходимо переписать уравнение регрессии в виде:

$$y = \alpha + \beta(x - \bar{x}),$$

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

Эта прямая называется теоретической линией регрессии или прямой отклика.

Уравнение, определяющее кривую, которая является оценкой для прямой регрессии, имеет вид:

$$\hat{y} = a + b(x - \bar{x})$$

Суть метода наименьших квадратов состоит в выборе таких оценок a и b , которые бы минимизировали сумму квадратов отклонений наблюдаемых значений y_i от прогнозируемых величин y_i^* . Уравнения для нахождения оценок:

$$a = \hat{a} = \frac{1}{n} \sum_{i=1}^n y_i = \bar{y}, \quad b = \hat{b} = \frac{\sum_{i=1}^n (x_i - \bar{x}) y_i}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad b = \frac{\sum_{i=1}^n [(x_i - \bar{x})(y_i - \bar{y})]}{\sum_{i=1}^n (x_i - \bar{x})^2}.$$

Для нахождения интервальных оценок необходима формула:

$$\hat{D} = D^* = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 :$$

Доверительный интервал для α :

$$\varepsilon = t_{\beta', n-2} \sqrt{\frac{\widehat{D}}{n-2}} \text{ и } I_{\alpha} = \left(a - t_{\beta', n-2} \sqrt{\frac{\widehat{D}}{n-2}}, a + t_{\beta', n-2} \sqrt{\frac{\widehat{D}}{n-2}} \right)$$

Доверительный интервал для β :

$$\varepsilon = \frac{t_{\beta'}}{d} = t_{\beta', n-2} \sqrt{\frac{\widehat{D}}{n-2}} \sqrt{1 + \frac{n(x - \bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}}, \quad I_y = (\widehat{y} - \varepsilon, \widehat{y} + \varepsilon)$$

Доверительный интервал для любого конкретного x :

$$\varepsilon = t_{\beta', n-2} \sqrt{\frac{n}{n-2} \frac{\widehat{D}}{\sum_{i=1}^n (x_i - \bar{x})^2}} \text{ и } I_{\beta} = \left(\varepsilon - t_{\beta', n-2} \sqrt{\frac{n}{n-2} \frac{\widehat{D}}{\sum_{i=1}^n (x_i - \bar{x})^2}}, \varepsilon + t_{\beta', n-2} \sqrt{\frac{n}{n-2} \frac{\widehat{D}}{\sum_{i=1}^n (x_i - \bar{x})^2}} \right).$$

Критерий Пирсона

Коэффициент корреляции Пирсона позволяет определить меру линейной корреляции между двумя наборами данных. Он рассчитывается следующим образом:

Given paired data $\{(x_1, y_1), \dots, (x_n, y_n)\}$ consisting of n pairs, r_{xy} is defined as:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (\text{Eq.3})$$

where:

n is sample size

x_i, y_i are the individual sample points indexed with i

$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ (the sample mean); and analogously for \bar{y}

Коэффициент корреляции Пирсона может принимать значения от -1 до 1. Причем чем ближе значение коэффициента по модулю к 1, тем лучше линейное уравнение описывает взаимосвязь между x и y , а все точки лежат на почти прямой линии.

Кроме того, для оценки построенной модели линейной регрессии можно вычислить критерий Пирсона по формуле:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})[y_i - \bar{y} + b(x_i - \bar{x})]}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 * \sum_{i=1}^n [y_i - \bar{y} + b(x_i - \bar{x})]^2}}$$

Реализация

В ходе реализации были использованы программные модули Python: statistics, numpy, prettytable, matplotlib, math.

На рис. 1 представлен программный код, необходимый для построения вспомогательной таблицы №1.

```
x = [0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55]
y = [2.43, 2.67, 2.71, 3.15, 3.47, 3.76, 3.91, 4.46, 4.76, 5.15, 5.54, 5.61]

x_sm = s.mean(x)

xi_xsm = [x[i] - x_sm for i in range(len(x))]
xi_xsm2 = [xi_xsm[i] ** 2 for i in range(len(xi_xsm))]

y_sm = s.mean(y)

yi_ysm = [y[i] - y_sm for i in range(len(y))]
yi_ysm2 = [el ** 2 for el in yi_ysm]
xi_xsm_yi_ysm2 = [xi_xsm[i] * yi_ysm[i] for i in range(len(yi_ysm))]

pt = PrettyTable()
pt.add_column('x', x)
pt.add_column('y', y)
pt.add_column('xi-x_sm', xi_xsm)
pt.add_column('yi-y_sm', yi_ysm)
pt.add_column('(xi-x_sm)^2', xi_xsm2)
pt.add_column('(xi-x_sm)(yi-y_sm)', xi_xsm_yi_ysm2)

print(pt.get_string())
```

Рис. 1. Листинг программы, строящей вспомогательную таблицу №1.

На рис. 2 представлен программный код, необходимый для построения вспомогательной таблицы №2 а также для нахождения уравнения регрессии.

```
alpha = y_sm
beta = sum(xi_xsm_yi_ysm2) / sum(xi_xsm2)

print('Уравнение регрессии: y_r={}-{}*(x-{})'.format(alpha, beta, x_sm))
y_r = [alpha + beta * xi_xsm[i] for i in range(len(xi_xsm))]
yi_yo = [y[i] - y_r[i] for i in range(len(y))]
yi_yo2 = [yi_yo[i] ** 2 for i in range(len(yi_yo))]

pt = PrettyTable()
pt.add_column('x', x)
pt.add_column('y', y)
pt.add_column('y_r-i', y_r)
pt.add_column('yi-y_r-i', yi_yo)
pt.add_column('(yi-y_r-i)**2', yi_yo2)

print(pt.get_string())
```

Рис. 2. Листинг программы, строящей вспомогательную таблицу №2 и уравнение оценки линейной регрессии.

```

fig, ax = plt.subplots()
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.plot(x, y, '-bo', color='blue', label='y')
ax.plot(x, y_r, '-bo', color='orange', label='y_r')
ax.grid()
ax.legend()
plt.show()

```

Рис. 3. Листинг программы построения графика регрессии и исходных данных.

На рис. 3 представлен программный код, необходимый для построения графика исходных значений и линейной регрессии.

```

e_a = t * math.sqrt(D_o / (n - 2))
e_b = t * math.sqrt((len(x) / (n - 2)) * (D_o / sum(xi_xsm2)))

print('eps_a={}, eps_b={}'.format(e_a, e_b))

def get_eps_y_i(xi_xsm2_i):
    return t * (sigma_o / math.sqrt(n - 2) * math.sqrt(1 + (n * xi_xsm2_i / sum(xi_xsm2))))

x_copy = x.copy()
x_copy[5] = x_sm
s2 = [(el - x_sm) ** 2 for el in x_copy]
e_y = [t * (sigma_o / math.sqrt(n - 2) * math.sqrt(1 + (n * ((el - x_sm) ** 2) / sum(s2)))) for el in x_copy]

pt = PrettyTable()
pt.add_column('x', x)
pt.add_column('eps_y', e_y)

```

Рис. 4. Листинг программы нахождения ε_a , ε_b , ε_y .

На рис. 4 представлен листинг программы, которая находит значения ε_a , ε_b , ε_y . Для нахождения ε_a и ε_b необходимо задать значение параметра t . Для

$n - 2 = 10$, $\beta' = 0.9$, $t(0.9, 10) = 2.228$ (из методички).

```

Ia_left = alpha - e_a
Ia_right = alpha + e_a

c = lin_reg.coef_[0]

Ib_left = beta - e_a
Ib_right = beta + e_a

print('Доверительный интервал Ia: ({}; {})'.format(Ia_left, Ia_right))
print('Доверительный интервал Ib: ({}; {})'.format(Ib_left, Ib_right))

```

Рис. 5. Листинг программы нахождения I_a , I_b .

На рис. 5 представлен листинг программы, которая находит значения доверительных интервалов I_a , I_b .

На рис. 6 представлен листинг программы, которая строит график регрессии и её 90% интервалов.


```

graph_up = [y_r[i] + e_y[i] for i in range(len(y_r))]
graph_down = [y_r[i] - e_y[i] for i in range(len(y_r))]

fig, ax = plt.subplots()
ax.plot(x, y, color='blue', label='y')
ax.plot(x, y_r, color='red', label='y_r')
ax.plot(x, graph_up, ':', color='black', label='U_lim')
ax.plot(x, graph_down, '--', color='black', label='D_lim')
ax.legend()
plt.grid()
plt.show()

```

Рис. 6. Листинг программы построения графика регрессии и её 90% интервалов.

На рис. 7-8 представлен листинг программы, которая находит значение коэффициента корреляции Пирсона двумя способами.

```

# критерий Пирсона
denom = sum([xi_xsm[i] * yi_ysm[i] for i in range(len(xi_xsm))])
sigma_x = math.sqrt(sum([xi_xsm[i] ** 2 for i in range(len(xi_xsm))]))
sigma_y = math.sqrt(sum([yi_ysm[i] ** 2 for i in range(len(xi_xsm))]))
nom = sigma_x * sigma_y
print('Критерий Пирсона:{}'.format(denom/nom))

```

Рис. 7. Листинг программы вычисления критерия корреляции Пирсона первым способом.

```

# критерий Пирсона 2
denom = sum([xi_xsm[i] * (yi_ysm[i] + beta * xi_xsm[i]) for i in range(len(xi_xsm))])
sigma_x = math.sqrt(sum([xi_xsm[i] ** 2 for i in range(len(xi_xsm))]))
sigma_y = math.sqrt(sum([(yi_ysm[i] + beta * xi_xsm[i]) ** 2 for i in range(len(xi_xsm))]))
nom = sigma_x * sigma_y
print('Критерий Пирсона 2:{}'.format(denom / nom))

```

Рис. 8. Листинг программы вычисления критерия корреляции Пирсона вторым способом.

Результаты

x	y	xi-x_sm	yi-y_sm	(xi-x_sm)^2	(xi-x_sm)(yi-y_sm)
0	2.43	-0.275	-1.5383333333333333	0.0756250000000001	0.4230416666666665
0.05	2.67	-0.2250000000000003	-1.2983333333333333	0.0506250000000002	0.292125
0.1	2.71	-0.1750000000000002	-1.2583333333333333	0.0306250000000006	0.2202083333333334
0.15	3.15	-0.1250000000000003	-0.8183333333333334	0.0156250000000007	0.1022916666666667
0.2	3.47	-0.0750000000000001	-0.4983333333333331	0.00562500000000015	0.03737499999999985
0.25	3.76	-0.02500000000000022	-0.20833333333333348	0.00062500000000011	0.00520833333333342
0.3	3.91	0.02499999999999967	-0.058333333333333126	0.00062499999999984	-0.0014583333333333262
0.35	4.46	0.07499999999999996	0.4916666666666667	0.00562499999999994	0.03687499999999998
0.4	4.76	0.125	0.7916666666666665	0.015625	0.09895833333333331
0.45	5.15	0.175	1.1816666666666667	0.03062499999999996	0.20679166666666673
0.5	5.54	0.22499999999999998	1.5716666666666668	0.05062499999999999	0.35362499999999997
0.55	5.61	0.275	1.6416666666666667	0.0756250000000001	0.45145833333333346

Рис. 9. Вспомогательная таблица №1.

На рис. 9 представлена вспомогательная таблица №1 (результат работы программы рис. 1). На рис. 10 представлена вспомогательная таблица №2 и полученное уравнение регрессии.

Уравнение регрессии: $y_r = 3.968333333333333 - 6.2279720279720285 \cdot (x - 0.275)$				
x	y	y_r_i	yi-y_r_i	(yi-y_r_i)**2
0	2.43	2.255641025641025	0.17435897435897507	0.030401051939513726
0.05	2.67	2.5670396270396267	0.10296037296037319	0.010600838400139147
0.1	2.71	2.8784382284382284	-0.16843822843822842	0.02837143679940882
0.15	3.15	3.1898368298368296	-0.03983682983682968	0.0015869730114485233
0.2	3.47	3.5012354312354312	-0.031235431235431044	0.0009756521644633413
0.25	3.76	3.8126340326340324	-0.05263403263403266	0.002770341391320415
0.3	3.91	4.124032634032634	-0.21403263403263395	0.045809968430947416
0.35	4.46	4.435431235431235	0.02456876456876511	0.0006036241924354079
0.4	4.76	4.7468298368298365	0.013170163170163285	0.00017345319792872542
0.45	5.15	5.058228438228438	0.0917715617715622	0.008422019549991657
0.5	5.54	5.36962703962704	0.17037296037296024	0.02902694562624628
0.55	5.61	5.6810256410256414	-0.07102564102564113	0.005044641683103236

Рис. 10. Вспомогательная таблица №2 и полученное уравнение оценки регрессии.

Далее с помощью программы (рис. 3) был построен график, где изображены исходные данные и регрессия (рис. 11).

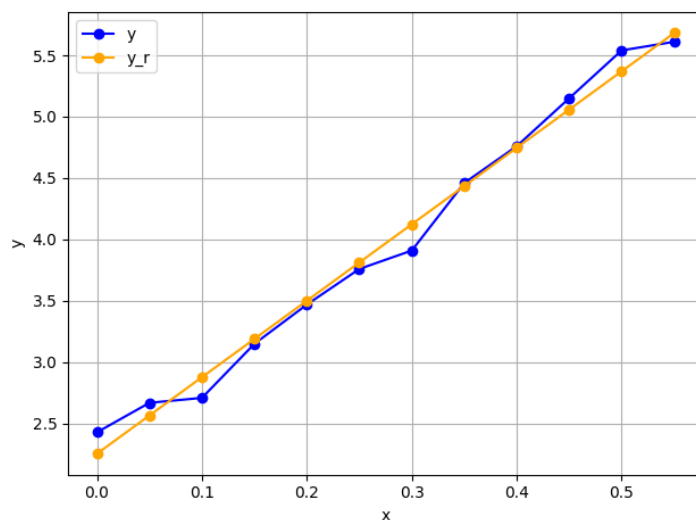


Рис.11. График регрессии и исходных данных.

Результат работы программы, которая находит значения ε_a , ε_b , ε_y , представлен на рис. 12.

```
eps_a=0.7618317599021162, eps_b=4.413790076391515
```

x	eps_y
0	1.4339663535575784
0.05	1.2523449903907538
0.1	1.0853823070301525
0.15	0.9409143614097174
0.2	0.830760629926392
0.25	0.7618317599021162
0.3	0.769795374498974
0.35	0.8307606299263919
0.4	0.9409143614097173
0.45	1.0853823070301525
0.5	1.2523449903907535
0.55	1.4339663535575784

Рис. 12. Найденные значения ε_a , ε_b , ε_y .

Результат работы программы, которая находит значения доверительных интервалов для a , b , представлен на рис.13.

Доверительный интервал I_a : (3.206501573431217; 4.730165093235449)
 Доверительный интервал I_b : (5.466140268069912; 6.989803787874145)

Рис. 13. Найденные доверительные интервалы I_a , I_b .

Далее с помощью программы (рис. 6) строим график регрессии и её 90% интервалов (рис. 14).

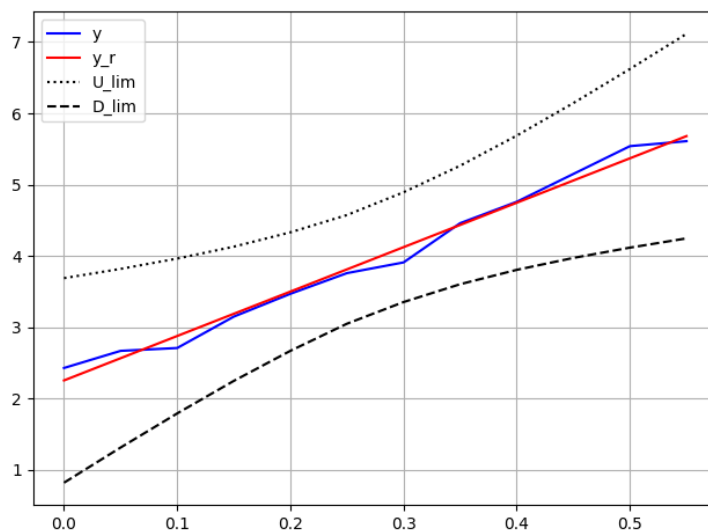
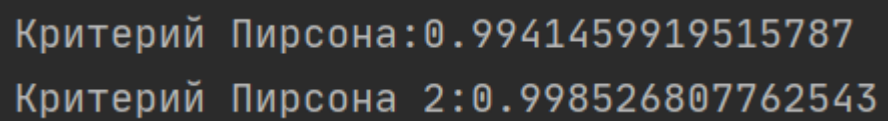


Рис. 14. График регрессии и её 90% интервалов.

A screenshot of a dark background with two lines of text in a light, monospaced font. The first line reads 'Критерий Пирсона:0.9941459919515787' and the second line reads 'Критерий Пирсона 2:0.998526807762543'.

Критерий Пирсона:0.9941459919515787
Критерий Пирсона 2:0.998526807762543

Рис. 15. Полученное значение критерия корреляции Пирсона.

На рис. 15 представлен результат вычисления критерия корреляции Пирсона двумя способами.

Заключение

В результате работы были найдены оценки параметров линейной регрессии y на x и доверительные интервалы для параметров и линии регрессии для заданных наборов значений.

Далее двумя способами был найден коэффициент корреляции Пирсона для x и y , который приблизительно равен 0.994 в первом случае, и 0.999 во втором. Это говорит о том, что линейное уравнение наилучшим образом описывает взаимосвязь x и y , все точки практически лежат на прямой. Близость коэффициента корреляции Пирсона к 1 также говорит о сильной связи между переменными. Таким образом, можно сказать, что построенная линейная регрессионная модель является адекватной исходным данным.