**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по лабораторной работе №5**

**по дисциплине «Статистические методы обработки экспериментальных данных»**

Тема: Элементы регрессионного анализа. Выборочные прямые среднеквадратической регрессии. Корреляционное отношение.

|  |  |  |
| --- | --- | --- |
| Студентка гр. 7381 |  | Алясова А.Н. |
| Студент гр. 7381 |  | Кортев Ю.В. |
| Преподаватель |  | Середа А.-В.И. |

Санкт-Петербург

2021

**Цель работы.**

Ознакомление с основными положениями метода наименьших квадратов (МНК), со статистическими свойствами МНК оценок, с понятием функции регрессии и роли МНК в регрессионном анализе, с корреляционным отношением, как мерой тесноты произвольной (в том числе и линейной) корреляционной связи.

**Основные теоретические положения.**

Метод наименьших квадратов (МНК) — метод, основанный на поиске минимума суммы квадратов отклонений значений некоторых функций от заданного множества значений. МНК является одним из основных методов регрессионного анализа и применяется для оценки параметров регрессионных моделей на основе выборочных данных.

Регрессионный анализ – это [статистический метод](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%D1%8B) исследования влияния одной или нескольких [независимых переменных](https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%B0%D1%8F_%D0%B8_%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%B0%D1%8F_%D0%BF%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D1%8B%D0%B5) на [зависимую переменную](https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%B0%D1%8F_%D0%B8_%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%B0%D1%8F_%D0%BF%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D1%8B%D0%B5) .

Линейные функции выборочной среднеквадратической регрессии:

где - статистические оценки математических ожиданий выборок , соответственно, – статистическая оценка коэффициента корреляции, и – статистические оценки среднеквадратических отклонений для выборок , .

Для оценки корреляционной зависимости между случайными величинами в общем, а не только линейной, может быть использовано так называемое корреляционной отношение.

Чтобы рассчитать выборочное корреляционное отношение нужно рассчитать внутригрупповую и межгрупповую дисперсии. Оценку общей дисперсии можно представить, как сумму:

Чтобы рассчитать выборочное корреляционное отношение к нужно рассчитать внутригрупповую и межгрупповую дисперсии.

Внутригрупповая дисперсия вычисляется по формуле:

где – объём выборки, – количество интервалов интервального ряда , – абсолютная частота для -ого интервала интервального ряда , – групповая дисперсия элементов выборки на -ом интервале интервального ряда .

Межгрупповая дисперсия вычисляется по формуле:

где – объём выборки, – количество интервалов интервального ряда , – абсолютная частота для i-ого интервала интервального ряда , 𝑖 – групповое математическое ожидание элементов выборки на -ом интервале интервального ряда , – статистическая оценка математического ожидания .

Выборочное корреляционное отношение к определяется в соответствии с выражением:

Где , – выборочные значения СКВО и соответственно. Аналогично определяется выборочное корреляционное отношение к .

Для расчёта выборочного корреляционного отношения к необходимо рассчитать те же величины по следующим формулам (меняем местами и ):

Выборочное уравнение регрессии на :

Значения коэффициентов определим с помощью МНК, что приводит к необходимости решать систему линейных уравнений 3го порядка:

Решив данную систему, найдём коэффициенты квадратичной функции выборочной среднеквадратической регрессии.

**Постановка задачи.**

Для заданной двумерной выборки построить уравнения выборочных прямых среднеквадратической регрессии. Полученные линейные функции регрессии отобразить графически. Найти выборочное корреляционное отношение. Полученные результаты содержательно проинтерпретировать.

**Выполнение работы.**

1-2) Отобразим двумерную выборку на графике и для заданной выборки построим уравнения средней квадратичной регрессии на и на соответственно. Построим полученные прямые на множестве выборки.

Линейная функция среднеквадратической регрессии 𝑦(𝑥) для заданной выборки:

;

.

Линейная функция среднеквадратической регрессии для заданной выборки:

Двумерная выборка и графики линейной функции выборочной среднеквадратической регрессии и представлены на рисунке 1.

Найдем оценки остаточной дисперсии для полученных выборочных уравнений регрессии:

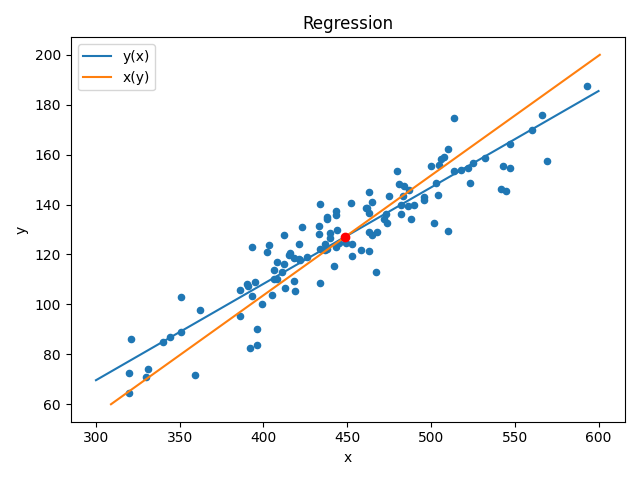
****

Рисунок 1 - Графики линейной функции выборочной среднеквадратической регрессии и

3) Составим корреляционную таблицу для нахождения выборочного корреляционного отношения. Убедимся, что неравенства и выполняются.

Таблица 1 - Корреляционная таблица

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ***72*** | ***87*** | ***102*** | ***117*** | ***132*** | ***147*** | ***162*** | ***177*** | ***192*** |  |  |  |
| ***337*** | 0 | 4 | 4 | 1 | 0 | 0 | 0 | 0 | 0 | 9 | 82 | 100 |
| ***371*** | 1 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 94,5 | 168,750 |
| ***405*** | 0 | 3 | 9 | 14 | 1 | 0 | 0 | 0 | 0 | 27 | 109,222 | 122,840 |
| ***439*** | 0 | 0 | 1 | 10 | 12 | 2 | 0 | 0 | 0 | 25 | 126 | 108 |
| ***473*** | 0 | 0 | 0 | 3 | 12 | 9 | 0 | 0 | 0 | 24 | 135,75 | 98,438 |
| ***507*** | 0 | 0 | 0 | 0 | 2 | 8 | 6 | 1 | 0 | 17 | 152,294 | 130,796 |
| ***541*** | 0 | 0 | 0 | 0 | 0 | 2 | 5 | 0 | 0 | 7 | 157,714 | 45,918 |
| ***575*** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | 0 | 3 | 172 | 50 |
| ***609*** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 192 | 0 |
|  | 5 | 7 | 14 | 27 | 27 | 21 | 12 | 3 | 1 | 117 |  |  |
|  | 343,8 | 366,143 | 395,286 | 425,148 | 457,889 | 489,190 | 526,833 | 552,333 | 609 |  |  |  |
|  | 46,24 | 23,592 | 94,367 | 191,874 | 228,346 | 317,179 | 200,694 | 513,778 | 0 |  |  |  |

Для того, чтобы посчитать выборочное корреляционное отношение, рассчитаем внутригрупповую, межгрупповую, общую дисперсии.

Расчёт выборочного корреляционного отношения к :

Расчёт выборочного корреляционного отношения к :

Проверим выполнение неравенств и :

Неравенства и выполняются.

4.1) Для заданной выборки построим корреляционную кривую параболического вида .

Для определения коэффициентов корреляционной кривой параболического вида была решена следующая система уравнений:

Система была решена с помощью написанной программы на языке Python (код представлен в ПРИЛОЖЕНИИ А).В результате работы программы были получены следующие значения коэффициентов:

Полученное уравнение примет вид:

График квадратичной функции выборочной среднеквадратической регрессии представлен на рис. 2.

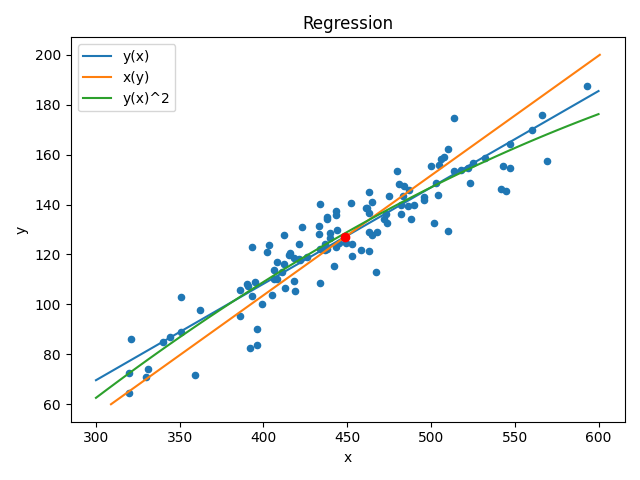


Рисунок 2 - Корреляционная кривая параболического вида

4.2) Построим корреляционную кривую экспоненциальной функции .

Запишем выборочное уравнение в виде .

Найдем коэффициенты и с помощью написанной программы на языке Python (код представлен в ПРИЛОЖЕНИИ А), получим следующие коэффициенты:

Корреляционная кривая экспоненциального вида имеет следующий вид:

.

График полученной кривой на множестве выборки представлен на рис. 3.

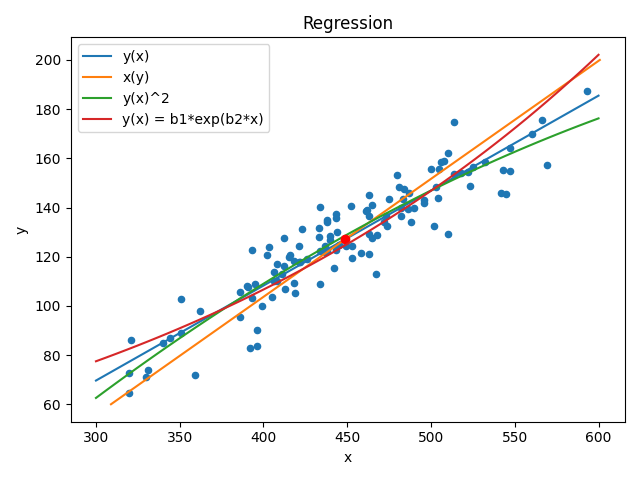


Рисунок 3

**Выводы.**

Таким образом, были получены уравнения прямых среднеквадратической регрессии: и .

Были найдены корреляционные соотношения и . Эти значения близки к 1, что говорит о том, что между и есть сильная зависимость.

Было найдено и построено уравнение выборочных кривых для параболической среднеквадратической регрессии . Была построена корреляционной кривая экспоненциального вида .

ПРИЛОЖЕНИЯ А

КОД ПРОГРАММЫ

from lab4 import df, cor, s\_x, s\_y, mean\_x, mean\_y, means\_x, means\_y, k, cor\_table, nums\_x, nums\_y

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np

def mean\_sq\_regression(s1, s2, mean1, mean2, cor, pref='y(x)='):

def inner\_foo(x, pr=False):

if pr:

print(pref + '{} \* x'.format(cor \* s1 / s2), '{0:+}'.format(mean1 - cor \* s1 / s2 \* mean2))

a1 = cor \* s1 / s2

a2 = mean1 - cor \* s1 / s2 \* mean2

return a2 + a1 \* x

return inner\_foo

msr\_x = mean\_sq\_regression(s\_x, s\_y, mean\_x, mean\_y, cor, 'x(y)=')

msr\_y = mean\_sq\_regression(s\_y, s\_x, mean\_y, mean\_x, cor)

ax = df.plot.scatter(x=0, y=1)

y1 = np.array([60, 200])

x2 = np.array([300, 600])

ax.plot(x2, msr\_y(x2, pr=True), label='y(x)')

ax.plot(msr\_x(y1, pr=True), y1, label='x(y)')

line1 = np.array([[x2[0], msr\_y(x2)[0]], [x2[1], msr\_y(x2)[1]]])

line2 = np.array([[msr\_x(y1)[0], y1[0]], [msr\_x(y1)[1], y1[1]]])

t, s = np.linalg.solve(np.array([line1[1] - line1[0], line2[0] - line2[1]]).T, line2[0] - line1[0])

ax.plot(\*((1 - t) \* line1[0] + t \* line1[1]), 'o', color='red')

ax.set\_title('Regression')

ax.set\_xlabel('x')

ax.set\_ylabel('y')

plt.legend()

plt.show()

print('Остаточная дисперсия y: ', (np.array(means\_y) - msr\_y(np.array(means\_x))).sum() / k)

print('Остаточная дисперсия x: ', (np.array(means\_x) - msr\_x(np.array(means\_y))).sum() / k)

cors = cor\_table.to\_numpy()

rows\_y = []

vars\_y = []

for row in range(len(cor\_table)):

cols = []

var = []

for col in range(len(cor\_table.columns)):

cols.append(cors[row][col] \* means\_y[col])

rows\_y.append(sum(cols) / nums\_x[row])

for col in range(len(cor\_table.columns)):

var.append(((means\_y[col] - rows\_y[-1]) \*\* 2) \* cors[row][col])

vars\_y.append(sum(var) / nums\_x[row])

cols\_x = []

vars\_x = []

for col in range(len(cor\_table.columns)):

rows = []

var = []

for row in range(len(cor\_table)):

rows.append(cors[row][col] \* means\_x[row])

cols\_x.append(sum(rows) / nums\_y[col])

for row in range(len(cor\_table)):

var.append(((means\_x[col] - cols\_x[-1]) \*\* 2) \* cors[row][col])

vars\_x.append(sum(var) / nums\_y[col])

cor\_table['n\_y'] = nums\_x

cor\_table['mean\_y\_gr'] = rows\_y

cor\_table['D\_y\_gr'] = vars\_y

cor\_table.to\_csv('Table1.csv')

pd.DataFrame({'n\_x': nums\_y, 'x\_mean\_gr': cols\_x, 'D\_x\_gr': vars\_x}).T.to\_csv('Table1\_last\_rows.csv')

x\_t = pd.DataFrame({'n\_x': nums\_y, 'x\_mean\_gr': cols\_x, 'D\_x\_gr': vars\_x})

D\_ingr\_yx = (cor\_table['D\_y\_gr'].to\_numpy() \* x\_t['n\_x'].to\_numpy()).sum() / len(df)

D\_ingr\_xy = (x\_t['D\_x\_gr'].to\_numpy() \* cor\_table['n\_y'].to\_numpy()).sum() / len(df)

print('D внутригр xy = {}'.format(D\_ingr\_xy))

print('D внутригр yx = {}'.format(D\_ingr\_yx))

D\_betwgr\_yx = (((cor\_table['mean\_y\_gr'] - mean\_y) \*\* 2).to\_numpy() \* x\_t['n\_x'].to\_numpy()).sum() / len(df)

D\_betwgr\_xy = (((x\_t['x\_mean\_gr'] - mean\_x) \*\* 2).to\_numpy() \* cor\_table['n\_y'].to\_numpy()).sum() / len(df)

print('D межгр xy = {}'.format(D\_betwgr\_xy))

print('D межгр yx = {}'.format(D\_betwgr\_yx))

D\_gen\_xy = D\_ingr\_xy + D\_betwgr\_xy

D\_gen\_yx = D\_ingr\_yx + D\_betwgr\_yx

print('D общая xy = {}'.format(D\_gen\_xy))

print('D общая yx = {}'.format(D\_gen\_yx))

n\_xy = np.sqrt(D\_betwgr\_xy / D\_gen\_xy)

n\_yx = np.sqrt(D\_betwgr\_yx / D\_gen\_yx)

print('n xy = {}'.format(n\_xy))

print('n yx = {}'.format(n\_yx))

x = df.iloc[:, 0]

y = df.iloc[:, 1]

system = []

b = []

for i in range(3):

line = []

for j in range(3):

line.append((x \*\* (4 - i - j)).sum())

system.append(line)

b.append((y \* (x \*\* (2 - i))).sum())

res = np.linalg.solve(np.array(system), np.array(b))

print('a={}, b={}, c={}'.format(\*res))

def sq\_regr(x):

return res[0] \* x \*\* 2 + res[1] \* x + res[2]

ax = df.plot.scatter(x=0, y=1)

y1 = np.array([60, 200])

x2 = np.array([300, 600])

ax.plot(x2, msr\_y(x2), label='y(x)')

ax.plot(msr\_x(y1), y1, label='x(y)')

x3 = np.linspace(300, 600)

ax.plot(x3, sq\_regr(x3), label='y(x)^2')

ax.plot(\*((1 - t) \* line1[0] + t \* line1[1]), 'o', color='red')

ax.set\_title('Regression')

ax.set\_xlabel('x')

ax.set\_ylabel('y')

y = df.iloc[:, 1]

x = df.iloc[:, 0]

z = np.log(y)

a1 = (len(df) \* (x\*z).sum() - x.sum() \* z.sum())/(len(df)\*(x\*x).sum()-x.sum()\*\*2)

a0 = z.mean() - a1 \* x.mean()

b = a1

a = np.exp(a0)

ax.plot(x3, a \* np.exp(b\*x3), label='y(x) = b1\*exp(b2\*x)')

plt.legend()

plt.show()