МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»

(БГТУ им. В.Г.Шухова)

**Лабораторная работа № 5**

Дисциплина: Системный анализ и обработка информации

Тема: Уравнение регрессии

Выполнил: студент группы ВТ-31

Ковалёв И. Д.

Проверил:

Полунин А. И.

Белгород 2020

**Цель работы**: получить математическую модель процесса, заданного его числовыми значениями при разных значениях двух аргументов. Числовые значения процесса измеряются со случайной ошибкой, распределенной по нормальному закону, мат. ожидание ошибки равно нулю, дисперсия неизвестна. Процесс описывается алгебраической зависимостью.

**Задание варианта №4**

|  |  |  |
| --- | --- | --- |
| Y | X1 | X2 |
| 434.6056050,  99.4576048,  426.5185028,  219.8754453,  261.1813246,  113.2138322,  293.0785722,  811.3168444,  325.0937345,  209.3121888,  506.5148699,  140.8936827,  185.8041733,  189.7014111,  187.8794151,  190.7171421,  192.7927908,  193.4331172,  185.9415325,  185.9871731,  194.3002881,  191.4144151, | 8.0  7.0  9.0  5.0  8.0  4.0  5.0  10.0  6.0  4.0  5.0  4.0 | 12.0  4.0  11.0  9.0  8.0  6.0  11.0  17.0  11.0  9.0  16.0  7.0 |

**Ход выполнения работы**

Считаем, что аппроксимируемый процесс имеет вид

,

где — оценки неизвестных параметров, — регрессоры.

На основании экспериментальных данных составляем матрицу регрессоров, где — значение j-го регрессора в i-ом эксперименте.

F =

В данном случае:

F =

[[ 13. 121. 143.]

[ 3. 49. 21.]

[ 12. 324. 216.]

[ 2. 121. 22.]

[ 12. 9. 36.]

[ 11. 36. 66.]

[ 5. 49. 35.]

[ 17. 144. 204.]

[ 4. 64. 32.]

[ 11. 4. 22.]

[ 9. 289. 153.]

[ 4. 196. 56.]]

Вектор оцениваем по методу наименьших квадратов

где Y – вектор оценки значений исследуемого процесса, полученный в экспериментах.

Для этого вычислим среднее арифметическое результатов эксперимента

и величины , где — значение модели, полученное при наборе регрессоров, вычисленных для i-го эксперимента.

=

**Результат работы программы:**

x₂² + x₁² + x₁

Дисперсия гауссовского шума по Y: 7.336230648704273

0.9951961894552044 x₂² + 0.9748363702787302 x₁² + 4.50234008958364 x₁

**Вывод:** в процессе выполнения лабораторной работы было изучено уравнение регрессии, получена математическая модель процесса, заданная его числовыми значениями при разных значениях двух аргументов. Числовые значения процесса измеряются со случайной ошибкой, распределенной по нормальному закону, мат. ожидание ошибки равно нулю, дисперсия неизвестна. Процесс описывается алгебраической зависимостью. Было перебрано 10 полиномов второй степени, проанализирована значимость этих коэффициентов.

**Листинг программы:**

x\_vectors = [

(8.0, 12.0),

(7.0, 4.0),

(9.0, 11.0),

(5.0, 9.0),

(8.0, 8.0),

(4.0, 6.0),

(5.0, 11.0),

(10.0, 17.0),

(6.0, 11.0),

(4.0, 9.0),

(5.0, 16.0),

(4.0, 7.0),

]

Y\_vector = [

434.6056050,

99.4576048,

426.5185028,

219.8754453,

261.1813246,

113.2138322,

293.0785722,

811.3168444,

325.0937345,

209.3121888,

506.5148699,

140.8936827,

]

x\_noise\_check = (4.0, 8.0)

y\_noise\_check = [

185.8041733,

189.7014111,

187.8794151,

190.7171421,

192.7927908,

193.4331172,

185.9415325,

185.9871731,

194.3002881,

191.4144151,

]

def f\_x1(x1, x2):

return x1

f\_x1.\_\_name\_\_ = "x₁"

def f\_x2(x1, x2):

return x2

f\_x2.\_\_name\_\_ = "x₂"

def f\_x1x1(x1, x2):

return x1 \*\* 2

f\_x1x1.\_\_name\_\_ = "x₁²"

def f\_x1x2(x1, x2):

return x1 \* x2

f\_x1x2.\_\_name\_\_ = "x₁x₂"

def f\_x2x2(x1, x2):

return x2 \*\* 2

f\_x2x2.\_\_name\_\_ = "x₂²"

def f\_x2x2x1(x1, x2):

return x2 \*\* 2 \*x1

f\_x2x2x1.\_\_name\_\_ = "x₁x₂²"

def f\_x1x1x2(x1, x2):

return x1 \*\* 2 \*x2

f\_x1x1x2.\_\_name\_\_ = "x₁²x₂"

polynom\_parts = [f\_x1, f\_x2, f\_x1x2, f\_x1x1, f\_x2x2]

polynoms = lab5\_regression.get\_polynom\_list(polynom\_parts,3)

print('Список возможных полиномов')

for index,polynom in enumerate(polynoms,start=1):

print('{}. {}'.format(index,' + '.join(lab5\_regression.get\_functions\_labels(polynom))))

F\_matrices = lab5\_regression.get\_all\_F\_matrices(polynoms,x\_vectors)

a\_vectors = lab5\_regression.get\_all\_a\_vectors(F\_matrices,Y\_vector)

best1\_index,best2\_index = lab5\_regression.get\_best\_polynom(F\_matrices,a\_vectors,Y\_vector)

print('Лучшие полиномы:')

print('R² через QR: ',' + '.join(lab5\_regression.get\_functions\_labels(polynoms[best1\_index])))

print('R² через Qstop: ',' + '.join(lab5\_regression.get\_functions\_labels(polynoms[best2\_index])))

print('Дисперсия гауссовского шума по Y: ',np.var(y\_noise\_check))

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

print(lab5\_regression.get\_regression\_equation

(

lab5\_regression.get\_functions\_labels(polynoms[best1\_index])

,a\_vectors[best1\_index])

)

x1\_list,x2\_list = [],[]

for i in x\_vectors:

x1\_list.append(i[0])

x2\_list.append(i[1])

y\_cap\_vector = F\_matrices[best1\_index].dot(a\_vectors[best1\_index])

fig = plt.figure()

axes = Axes3D(fig)

axes.scatter(np.array(x1\_list), np.array(x2\_list), np.array(Y\_vector))

axes.scatter(np.array(x1\_list), np.array(x2\_list), np.array(y\_cap\_vector))

plt.show()

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

print(lab5\_regression.get\_regression\_equation

(

lab5\_regression.get\_functions\_labels(polynoms[best2\_index])

,a\_vectors[best2\_index])

)

x1\_list,x2\_list = [],[]

for i in x\_vectors:

x1\_list.append(i[0])

x2\_list.append(i[1])

y\_cap\_vector = F\_matrices[best2\_index].dot(a\_vectors[best2\_index])

fig = plt.figure()

axes = Axes3D(fig)

axes.scatter(np.array(x1\_list), np.array(x2\_list), np.array(Y\_vector),label = 'Исходные данные')

axes.scatter(np.array(x1\_list), np.array(x2\_list), np.array(y\_cap\_vector),label = 'Полученные значения')

plt.show()