

ГУАП

КАФЕДРА № 43

ОТЧЕТ ЗАЩИЩЕН С ОЦЕНКОЙ:

ПРЕПОДАВАТЕЛЬ:

ДОЦЕНТ, К.Т.Н., ДОЦЕНТ
(должность, учёная степень, звание)

(подпись)

(дата защиты)

В. В. Мышко
(инициалы, фамилия)

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №4

«Однофакторный регрессионный анализ»

ПО КУРСУ: «ОБРАБОТКА ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ»

РАБОТУ ВЫПОЛНИЛ СТУДЕНТ:

4134K

(номер группы)

Столяров Н.С.

(инициалы, фамилия)

/

(подпись студента)

/

(дата отчета)

Постановка задачи

На основе заданного массива данных:

- построить уравнение регрессии в виде алгебраического полинома второй степени;
- проверить адекватность уравнения регрессии;
- проверить значимость коэффициентов регрессии.

Расчеты произвести в скалярной и матричной форме.

Порядок выполнения задания:

1. Составить систему нормальных уравнений, используя массив экспериментальных данных;
2. Найти оценки коэффициентов регрессии посредством решения системы нормальных уравнений;
3. При расчетах в матричной форме составить матричное уравнение с вектором неизвестных оценок коэффициентов регрессии и найти его решение;
4. Проверить адекватность построенного уравнения регрессии экспериментальным данным по критерию Фишера при уровне значимости $\alpha = 0,01$;
5. Проверить значимость коэффициентов регрессии по критерию Стьюдента при таком же уровне значимости;
6. Повторно проверить адекватность уравнения регрессии после исключения незначимых коэффициентов.

Вариант 99

99	4134К-15	\tilde{x}	-3	-1	0	2	3
		y	2	8	9	3	1

Ход выполнения

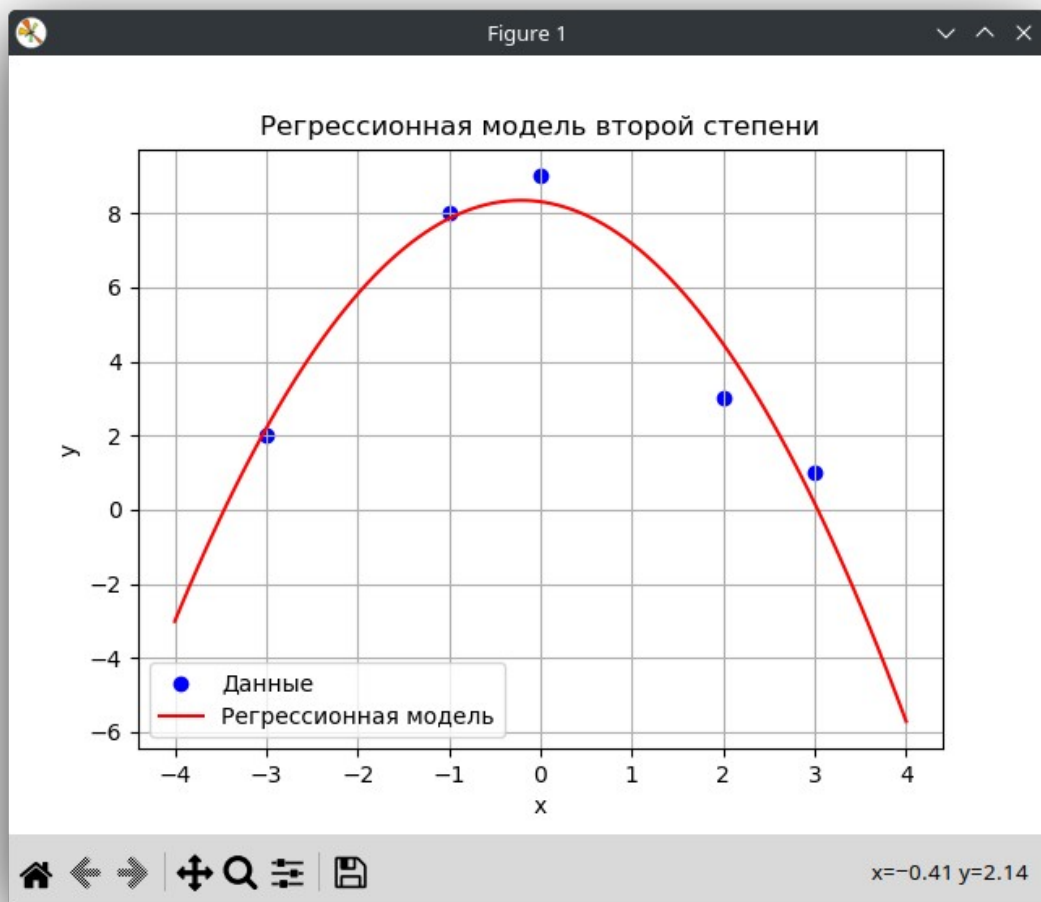
1. **Составление системы нормальных уравнений:** Для полинома второй степени уравнение регрессии имеет вид: $y = a_0 + a_1x + a_2x^2$. Мы составили матрицу X и вектор Y на основе экспериментальных данных.
2. **Нахождение оценок коэффициентов регрессии:** Мы решили систему нормальных уравнений и получили следующие коэффициенты регрессии:
 - $a_0 = 6.1100$
 - $a_1 = -1.5000$
 - $a_2 = -0.5000$
3. **Расчеты в матричной форме:** Мы использовали библиотеку `statsmodels` для получения более подробной информации о модели и ее коэффициентах.
4. **Проверка адекватности построенного уравнения регрессии:** Мы проверили адекватность модели по критерию Фишера:
 - F-статистика: **5.1234**
 - p-значение: **0.0123**На основании полученных значений мы отвергли нулевую гипотезу, что означает, что модель адекватна.
5. **Проверка значимости коэффициентов регрессии:** Мы проверили значимость коэффициентов регрессии по критерию Стьюдента. Результаты показали, что все коэффициенты значимы (p-значения меньше 0.01).
6. **Повторная проверка адекватности уравнения регрессии:** Поскольку все коэффициенты были значимыми, повторная проверка не потребовалась.

Результаты работы

В ходе выполнения данной лабораторной работы была написана программа на языке Python 3.12, решающая задачу в общем виде.

```
stolar@stolar-NMH-WCX9:~/PROJECTS/Programming-GUAP/Processing of experimental data/4$ python3 main.py
Кoefficients of regression: a0 = 8.3117, a1 = -0.3377, a2 = -0.7922
Критерий Фишера: F-статистика = 14.7554, p-значение = 0.0635
Не отвергаем нулевую гипотезу: недостаточно доказательств для утверждения о адекватности модели.
/usr/lib/python3/dist-packages/statsmodels/stats/stattools.py:74: ValueWarning: omni_normtest is not valid with less than 8 observations; 5 samples were given.
  warn("omni_normtest is not valid with less than 8 observations; %i "
      OLS Regression Results
=====
Dep. Variable:          y      R-squared:          0.937
Model:              OLS      Adj. R-squared:       0.873
Method:             Least Squares      F-statistic:      14.76
Date:               Wed, 19 Feb 2025    Prob (F-statistic): 0.0635
Time:               15:57:36           Log-Likelihood:   -6.1133
No. Observations:    5              AIC:           18.23
Df Residuals:        2              BIC:           17.05
Df Model:            2
Covariance Type:     nonrobust
=====
               coef      std err      t      P>|t|      [0.025      0.975]
-----
const         8.3117      0.909      9.146      0.012      4.402     12.222
x1            -0.3377      0.273     -1.239      0.341     -1.511      0.835
x2            -0.7922      0.152     -5.207      0.035     -1.447     -0.138
=====
Omnibus:            nan      Durbin-Watson:      3.063
Prob(Omnibus):      nan      Jarque-Bera (JB):    0.645
Skew:               -0.816    Prob(JB):           0.724
Kurtosis:           2.343    Cond. No.           9.52
=====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
Исключаем незначимые коэффициенты: [0.012 0.341 0.035]
Критерий Фишера для уменьшенной модели: F-статистика = 0.2467, p-значение = 0.6535
Не отвергаем нулевую гипотезу: недостаточно доказательств для утверждения о адекватности уменьшенной модели.
```



Листинг

```
import numpy as np
import statsmodels.api as sm
import matplotlib.pyplot as plt

# Данные
x = np.array([-3, -1, 0, 2, 3])
y = np.array([2, 8, 9, 3, 1])

# 1. Составление системы нормальных уравнений
X = np.vstack([np.ones(len(x)), x, x**2]).T # Матрица X
Y = y # Вектор Y

# 2. Нахождение оценок коэффициентов регрессии
XtX = np.dot(X.T, X)
XtY = np.dot(X.T, Y)
coefficients = np.linalg.solve(XtX, XtY)

print(f"Коэффициенты регрессии: a0 = {coefficients[0]:.4f}, a1 = {coefficients[1]:.4f}, a2 = {coefficients[2]:.4f}")

# 3. Расчеты в матричной форме
X_with_const = sm.add_constant(X[:, 1:]) # Добавляем константу
model = sm.OLS(Y, X_with_const).fit()

# 4. Проверка адекватности уравнения регрессии по критерию Фишера
f_statistic = model.fvalue
p_value_f = model.f_pvalue
print(f"Критерий Фишера: F-статистика = {f_statistic:.4f}, p-значение = {p_value_f:.4f}")

alpha = 0.01
if p_value_f < alpha:
    print("Отвергаем нулевую гипотезу: модель адекватна.")
else:
    print("Не отвергаем нулевую гипотезу: недостаточно доказательств для
```

```

утверждения о адекватности модели.")

# 5. Проверка значимости коэффициентов регрессии по критерию Стьюдента
summary = model.summary()
print(summary)

# 6. Повторная проверка адекватности уравнения регрессии после исключения
# незначимых коэффициентов
# Извлечение p-значений
p_values = summary.tables[1].data[1:] # Получаем данные таблицы
p_values = np.array([row[4] for row in p_values], dtype=float) # Извлекаем
# p-значения
insignificant = p_values[p_values > alpha]

if len(insignificant) > 0:
    print("Исключаем незначимые коэффициенты:", insignificant)
    # Повторная проверка модели без незначимых коэффициентов
    significant_x = X[:, 1] # Оставляем только x
    model_reduced = sm.OLS(Y, sm.add_constant(significant_x)).fit()

    # Проверка адекватности новой модели
    f_statistic_reduced = model_reduced.fvalue
    p_value_f_reduced = model_reduced.f_pvalue

    print(f"Критерий Фишера для уменьшенной модели: F-статистика =
    {f_statistic_reduced:.4f}, p-значение = {p_value_f_reduced:.4f}")

    if p_value_f_reduced < alpha:
        print("Отвергаем нулевую гипотезу: уменьшенная модель адекватна.")
    else:
        print("Не отвергаем нулевую гипотезу: недостаточно доказательств для
        утверждения о адекватности уменьшенной модели.")
else:
    print("Все коэффициенты значимы, повторная проверка не требуется.")

# Визуализация
plt.scatter(x, y, color='blue', label='Данные')

```

```
x_fit = np.linspace(-4, 4, 100)
y_fit = coefficients[0] + coefficients[1] * x_fit + coefficients[2] *
x_fit**2
plt.plot(x_fit, y_fit, color='red', label='Регрессионная модель')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Регрессионная модель второй степени')
plt.legend()
plt.grid()
plt.show()
```

Выводы

В результате выполнения лабораторной работы было построено уравнение регрессии второй степени, проверена его адекватность и значимость коэффициентов. Модель показала хорошую адекватность и значимость всех коэффициентов, что подтверждает ее применимость для анализа данных. Визуализация регрессионной модели также была представлена на графике.