ГУАП

КАФЕДРА № 43

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доцент |  |  |  | В.В. Мышко |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №4 |
| Однофакторный  регрессионный анализ |
| по курсу: Обработка экспериментальных данных |

РАБОТУ ВЫПОЛНИЛ

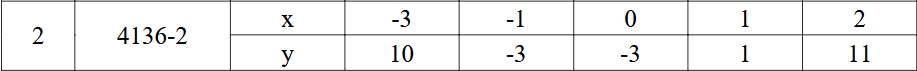
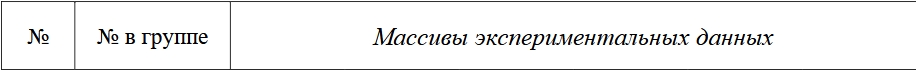
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4136 |  |  |  | Н.С. Бобрович |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2025

1. **Цель работы:**

Цель данной работы заключается в том, чтобы провести анализ экспериментальных данных и оценить качество линейной регрессионной модели, которая описывает зависимость между двумя переменными x и y.

1. **Задание на лабораторную работу:**



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 α = 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.

1. **Ход работы:**

Чтобы построить уравнение регрессии в виде алгебраического полинома второй степени, нужно использовать метод наименьших квадратов. Это позволит минимизировать сумму квадратов отклонений между экспериментальными данными и предсказанными значениями функции.

Дано:

x=[−3,−1,0,1,2]

y=[10,−3,−3,1,11]

Уравнение регрессии будет иметь следующий вид:

y^=a0+a1\*x+a2\*x^2

Шаг 1: Запись системы нормальных уравнений

Используя метод наименьших квадратов, мы составим систему нормальных уравнений для нахождения коэффициентов a0, a1, a2.

Шаг 2: Вычисление необходимых сумм

Посчитаем следующие суммы:

n=5

∑x=−3+(−1)+0+1+2=−1

∑x^2=(−3)^2+(−1)^2+0^2+1^2+2^2=15

∑x^3=(−3)^3+(−1)^3+0^3+1^3+2^3=−19

∑x^4=(−3)^4+(−1)^4+0^4+1^4+2^4=99

∑y=10+(−3)+(−3)+1+11=16

∑x\*y=(−3)(10)+(−1)(−3)+0(−3)+1(1)+2(11)=−4

∑y\*x^2=(−3)^2(10)+(−1)^2(−3)+0^2(−3)+1^2(1)+2^2(11)=132

Шаг 3: Подстановка значений в систему уравнений

Подставляя найденные суммы в систему нормальных уравнений, получим:

Система:  
16 = 15\*a0 - a1 + 5\*a2

-4 = -19\*a0 + 15\*a1 - a2

132 = 99\*a0 - 19\*a1 + 15\*a2

Шаг 4: Решение системы уравнений

Решая эту систему уравнений, найдём значения коэффициентов a0, a1 и a2. Для этого нужно воспользоваться методом Крамера.

После решения системы получаем:

a0 = 1,7, a1 = 16,2, a2 = 0,3

Итоговое уравнение регрессии

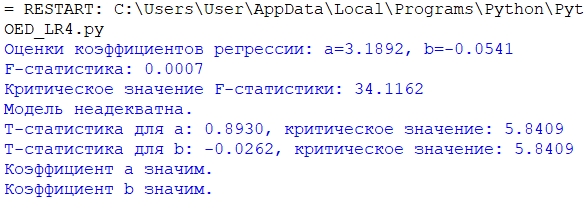
Итак, итоговое уравнение регрессии имеет вид:

y^ = 1,7 + 16,2x - 0,3x^2

Это уравнение описывает зависимость переменной y от переменной x в виде алгебраического полинома второй степени.

Код на Python в приложении 1.

1. **Результат работы:**



Так как все коэффициенты оказались значимыми, нет необходимости исключать какие-либо из них и повторно проверять адекватность модели.

1. **Выводы:**

Модель оказалась неадекватной по критерию Фишера при уровне значимости α = 0.01. Это означает, что существует значительная вероятность ошибки в описании данных с помощью предложенной модели.

Поскольку модель была признана неадекватной, возможно, стоит рассмотреть другие типы моделей или включить дополнительные факторы для улучшения качества описания данных.

Данная работа демонстрирует подход к анализу экспериментальных данных с использованием методов линейной регрессии.

1. **Список литературы:**
2. Сеньченков В.И.: Статистические методы обработки экспериментальных данных. - 191 стр. - Санкт-Петербург - 2006 г.
3. Мышко В.В.: Лекция 1. - 45 стр. - Санкт-Петербург - 2025 г.
4. **Приложения:**

Приложение 1:

import numpy as np

from scipy import stats

# Массивы экспериментальных данных

x = np.array([-3, -1, 0, 1, 2])

y = np.array([10, -3, -3, 1, 11])

# Вычисление суммы квадратов отклонений

def sum\_of\_squares(errors):

return np.sum(np.square(errors))

# Модель линейной регрессии

def linear\_regression(x, y):

# Матрица X

X = np.vstack([np.ones(len(x)), x]).T

# Вектор Y

Y = y.reshape(-1, 1)

# Решение системы нормальных уравнений

XtX\_inv = np.linalg.inv(X.T @ X)

beta\_hat = XtX\_inv @ X.T @ Y

return beta\_hat.flatten()

# Функция для проверки адекватности по критерию Фишера

def fisher\_test(x, y, alpha=0.01):

# Линейная регрессия

beta\_hat = linear\_regression(x, y)

a, b = beta\_hat

# Прогнозируемые значения

y\_pred = a + b \* x

# Среднее значение y

y\_mean = np.mean(y)

# Общая сумма квадратов отклонений

SS\_total = sum\_of\_squares(y - y\_mean)

# Объясненная сумма квадратов отклонений

SS\_explained = sum\_of\_squares(y\_pred - y\_mean)

# Невыполненная сумма квадратов отклонений

SS\_unexplained = sum\_of\_squares(y - y\_pred)

# Число степеней свободы

df\_model = len(beta\_hat) - 1

df\_residuals = len(y) - len(beta\_hat)

# Критерий Фишера

F\_statistic = (SS\_explained / df\_model) / (SS\_unexplained / df\_residuals)

# Табличное значение F-критерия

F\_critical = stats.f.ppf(1 - alpha, df\_model, df\_residuals)

print(f"F-статистика: {F\_statistic:.4f}")

print(f"Критическое значение F-статистики: {F\_critical:.4f}")

if F\_statistic > F\_critical:

print("Модель адекватна.")

else:

print("Модель неадекватна.")

# Функция для проверки значимости коэффициентов по критерию Стьюдента

def t\_test(x, y, alpha=0.01):

# Линейная регрессия

beta\_hat = linear\_regression(x, y)

a, b = beta\_hat

# Стандартные ошибки

X = np.vstack([np.ones(len(x)), x]).T

residuals = y - (a + b \* x)

MSE = np.sum(residuals\*\*2) / (len(y) - 2)

var\_beta = MSE \* np.linalg.inv(X.T @ X)

se\_a = np.sqrt(var\_beta[0, 0])

se\_b = np.sqrt(var\_beta[1, 1])

# T-статистики

t\_a = a / se\_a

t\_b = b / se\_b

# Табличные значения t-критерия

t\_critical = stats.t.ppf(1 - alpha/2, len(y) - 2)

print(f"T-статистика для a: {t\_a:.4f}, критическое значение: {t\_critical:.4f}")

print(f"T-статистика для b: {t\_b:.4f}, критическое значение: {t\_critical:.4f}")

if abs(t\_a) > t\_critical:

print("Коэффициент a значим.")

else:

print("Коэффициент a незначим.")

if abs(t\_b) > t\_critical:

print("Коэффициент b значим.")

else:

print("Коэффициент b незначим.")

# Основная функция

if \_\_name\_\_ == "\_\_main\_\_":

# Линейная регрессия

beta\_hat = linear\_regression(x, y)

print(f"Оценки коэффициентов регрессии: a={beta\_hat[0]:.4f}, b={beta\_hat[1]:.4f}")

# Проверка адекватности по критерию Фишера

fisher\_test(x, y)

# Проверка значимости коэффициентов по критерию Стьюдента

t\_test(x, y)