Министерство науки и высшего образования Российской Федерации
Калужский филиал федерального государственного автономного образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК Информатика и управление

КАФЕДРА ИУК4 Программное обеспечение ЭВМ, информационные технологии

ПРАКТИЧЕСКАЯ РАБОТА

«Линейные классификаторы и их свойства»

по дисциплине: «Методы машинного обучения»

Выполнил: студент группы ИУК4-72Б _____ Губин Е.В.
(Подпись) (И.О. Фамилия)

Проверил: _____ Семененко М.Г.
(Подпись) (И.О. Фамилия)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:
- Оценка:

Калуга, 2025

1) **Линейная и квадратичная регрессия**

- Построить линейную и квадратичную регрессию по данным.
- Для каждой модели рассчитать предсказанные значения (Линейная Y, Квадратичная Y) для каждого профиля.
- Вычислить среднюю квадратичную ошибку (MSE) и сумму квадратов ошибок (SSE) для каждой модели.
- Построить точечную диаграмму исходных данных с наложением линий регрессии.

2) **Аппроксимация функции Рунге**

- Аппроксимировать функцию полиномами 5-й и 6-й степени.
- Рассчитать предсказанные значения функции для этих полиномов.
- Вычислить среднюю ошибку (MSE) и сумму квадратов ошибок (SSE) для каждой аппроксимации.
- Построить графики функции Рунге и аппроксимационных полиномов с точками.
- Составить таблицу с исходными значениями функции и предсказаниями полиномов.

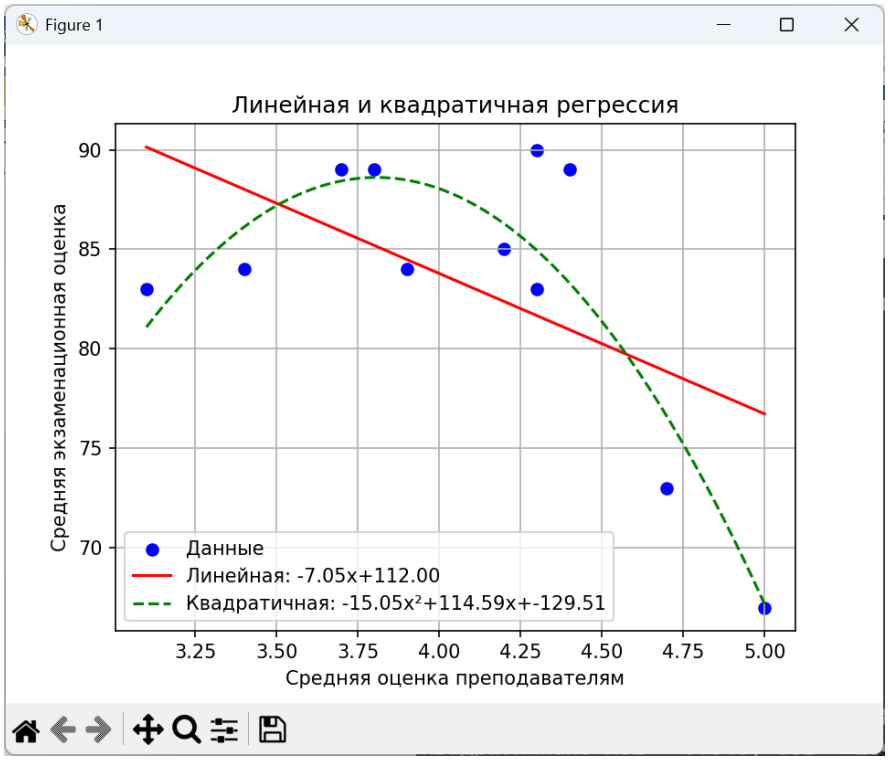
Листинг программы:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

profiles = ["A", "B", "C", "D", "E", "F", "G", "H", "P", "J", "K"]
x = np.array([4.3, 3.9, 5.0, 4.3, 3.7, 4.4, 3.8, 3.1, 4.7, 4.2, 3.4])
y = np.array([90, 84, 67, 83, 89, 89, 89, 83, 73, 85, 84])

coeffs_lin = np.polyfit(x, y, 1)
lin_eq = np.poly1d(coeffs_lin)
coeffs_quad = np.polyfit(x, y, 2)
quad_eq = np.poly1d(coeffs_quad)

y_pred_lin = lin_eq(x)
y_pred_quad = quad_eq(x)

mse_lin = np.mean((y - y_pred_lin) ** 2)
mse_quad = np.mean((y - y_pred_quad) ** 2)
sse_lin = np.sum((y - y_pred_lin) ** 2)
sse_quad = np.sum((y - y_pred_quad) ** 2)

results_reg = pd.DataFrame({
    "Profile": profiles,
    "X (Преподаватели)": x,
    "Y (Экзамен)": y,
    "Линейная Y": y_pred_lin.round(2),
    "Квадратичная Y": y_pred_quad.round(2)
})
print("=== Результаты регрессии ===")
print(results_reg)
print("\nОшибки:")
print(f"Линейная: MSE = {mse_lin:.3f}, SSE = {sse_lin:.3f}")
print(f"Квадратичная: MSE = {mse_quad:.3f}, SSE = {sse_quad:.3f}")
```

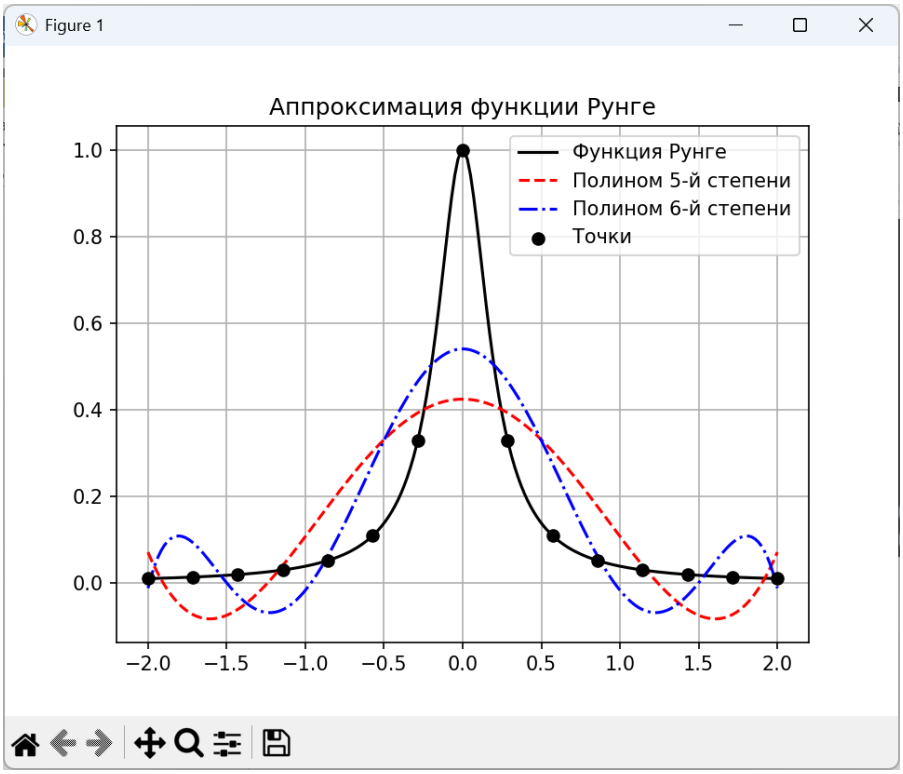
```
plt.scatter(x, y, label="Данные", color="blue")
x_line = np.linspace(min(x), max(x), 200)
plt.plot(x_line, lin_eq(x_line), "r-", label=f"Линейная:
{coeffs_lin[0]:.2f}x+{coeffs_lin[1]:.2f}")
plt.plot(x_line, quad_eq(x_line), "g--", label=f"Квадратичная:
{coeffs_quad[0]:.2f}x²+{coeffs_quad[1]:.2f}x+{coeffs_quad[2]:.2f}")
plt.xlabel("Средняя оценка преподавателям")
plt.ylabel("Средняя экзаменационная оценка")
plt.legend()
plt.title("Линейная и квадратичная регрессия")
plt.grid()
plt.show()

def runge(x):
    return 1 / (1 + 25 * x**2)

X_runge = np.linspace(-2, 2, 15)
Y_runge = runge(X_runge)

coeffs_5 = np.polyfit(X_runge, Y_runge, 5)
coeffs_6 = np.polyfit(X_runge, Y_runge, 6)
poly5 = np.poly1d(coeffs_5)
poly6 = np.poly1d(coeffs_6)

Y_pred_5 = poly5(X_runge)
Y_pred_6 = poly6(X_runge)

mse_5 = np.mean((Y_runge - Y_pred_5) ** 2)
mse_6 = np.mean((Y_runge - Y_pred_6) ** 2)
sse_5 = np.sum((Y_runge - Y_pred_5) ** 2)
sse_6 = np.sum((Y_runge - Y_pred_6) ** 2)

results_runge = pd.DataFrame({
    "X": X_runge.round(2),
    "Runge f(x)": Y_runge.round(5),
    "Полином 5 ст.": Y_pred_5.round(5),
    "Полином 6 ст.": Y_pred_6.round(5)
})
print("\n=== Приближение функции Рунге ===")
print(results_runge)
print("\nОшибки для Рунге:")
print(f"Полином 5-й ст.: MSE = {mse_5:.6f}, SSE = {sse_5:.6f}")
print(f"Полином 6-й ст.: MSE = {mse_6:.6f}, SSE = {sse_6:.6f}")

x_plot = np.linspace(-2, 2, 200)
plt.plot(x_plot, runge(x_plot), "k-", label="Функция Рунге")
plt.plot(x_plot, poly5(x_plot), "r--", label="Полином 5-й степени")
plt.plot(x_plot, poly6(x_plot), "b-.", label="Полином 6-й степени")
plt.scatter(X_runge, Y_runge, color="black", zorder=5, label="Точки")
plt.legend()
plt.title("Аппроксимация функции Рунге")
plt.grid()
plt.show()
```

Результаты выполнения программы:

```
=== Результаты регрессии ===
   Profile  X (Преподаватели)  Y (Экзамен)  Линейная Y  Квадратичная Y
0        A                4.3           90       81.67           84.95
1        B                3.9           84       84.49           88.48
2        C                5.0           67       76.73           67.18
3        D                4.3           83       81.67           84.95
4        E                3.7           89       85.90           88.44
5        F                4.4           89       80.96           83.31
6        G                3.8           89       85.20           88.61
7        H                3.1           83       90.13           81.09
8        P                4.7           73       78.85           76.60
9        J                4.2           85       82.38           86.28
10       K                3.4           84       88.02           86.12

Ошибки:
Линейная: MSE = 32.990, SSE = 362.888
Квадратичная: MSE = 9.543, SSE = 104.968
```



```
=== Приближение функции Рунге ===
       X  Runge f(x)  Полином 5 ст.  Полином 6 ст.
0  -2.00     0.00990        0.07087       -0.01213
1  -1.71     0.01343       -0.07321        0.09278
2  -1.43     0.01922       -0.06099       -0.02906
3  -1.14     0.02971        0.04048       -0.06167
4  -0.86     0.05163        0.17632        0.06198
5  -0.57     0.10913        0.30383        0.27481
6  -0.29     0.32886        0.39253        0.46508
7   0.00     1.00000        0.42413        0.54021
8   0.29     0.32886        0.39253        0.46508
9   0.57     0.10913        0.30383        0.27481
10  0.86     0.05163        0.17632        0.06198
11  1.14     0.02971        0.04048       -0.06167
12  1.43     0.01922       -0.06099       -0.02906
13  1.71     0.01343       -0.07321        0.09278
14  2.00     0.00990        0.07087       -0.01213

Ошибки для Рунге:
Полином 5-й ст.: MSE = 0.032146, SSE = 0.482187
Полином 6-й ст.: MSE = 0.022571, SSE = 0.338559
```



Вывод: в ходе лабораторной работы получены навыки по созданию модели, предсказывающую экзаменационные оценки и вычислению ошибок этой модели.