|  |  |
| --- | --- |
|  | **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»**

**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

**Цель:** изучить применение линейных классификаторов для аппроксимации функции Рунге методом полиномиальной регрессии.

**Задачи:**

1. Сформировать обучающую и контрольную выборки, построить полиномиальные модели разной степени и выбрать оптимальную по функции потерь.
2. Оценить качество выбранной модели на контрольной выборке и визуализировать результат аппроксимации.

**Задание №1:**

Используя функцию Рунге y = 1/(1 + 25x2), сформировать

обучающую выборку по правилу:



Сформировать контрольную выборку по правилу:



Объем выборки *l* = 15.

По обучающей выборке выбрать оптимальную степень полинома

,

соответствующую минимальной квадратичной функции потерь, задавая вручную степень полинома.

Для контрольной выборки построить полином выбранной степени и рассчитать значение квадратичной функции потерь. Построить график.

**Результаты выполнения программы:**

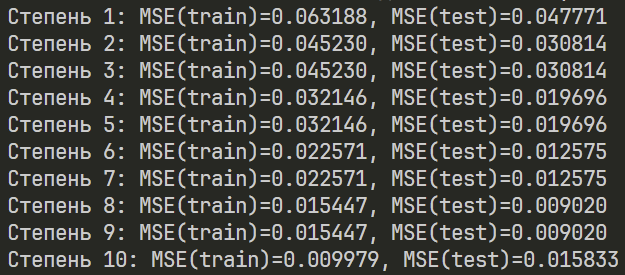


Рисунок 1 Степени полиномов и ошибки по ним

В результате расчётов видно, что минимальную ошибку имеет полином 8-й степени, его и возьмём за самый оптимальный полином. Выполнена аппроксимация функции Рунге полиномом 8-й степени. Результаты аппроксимации отображены на рис. 2.

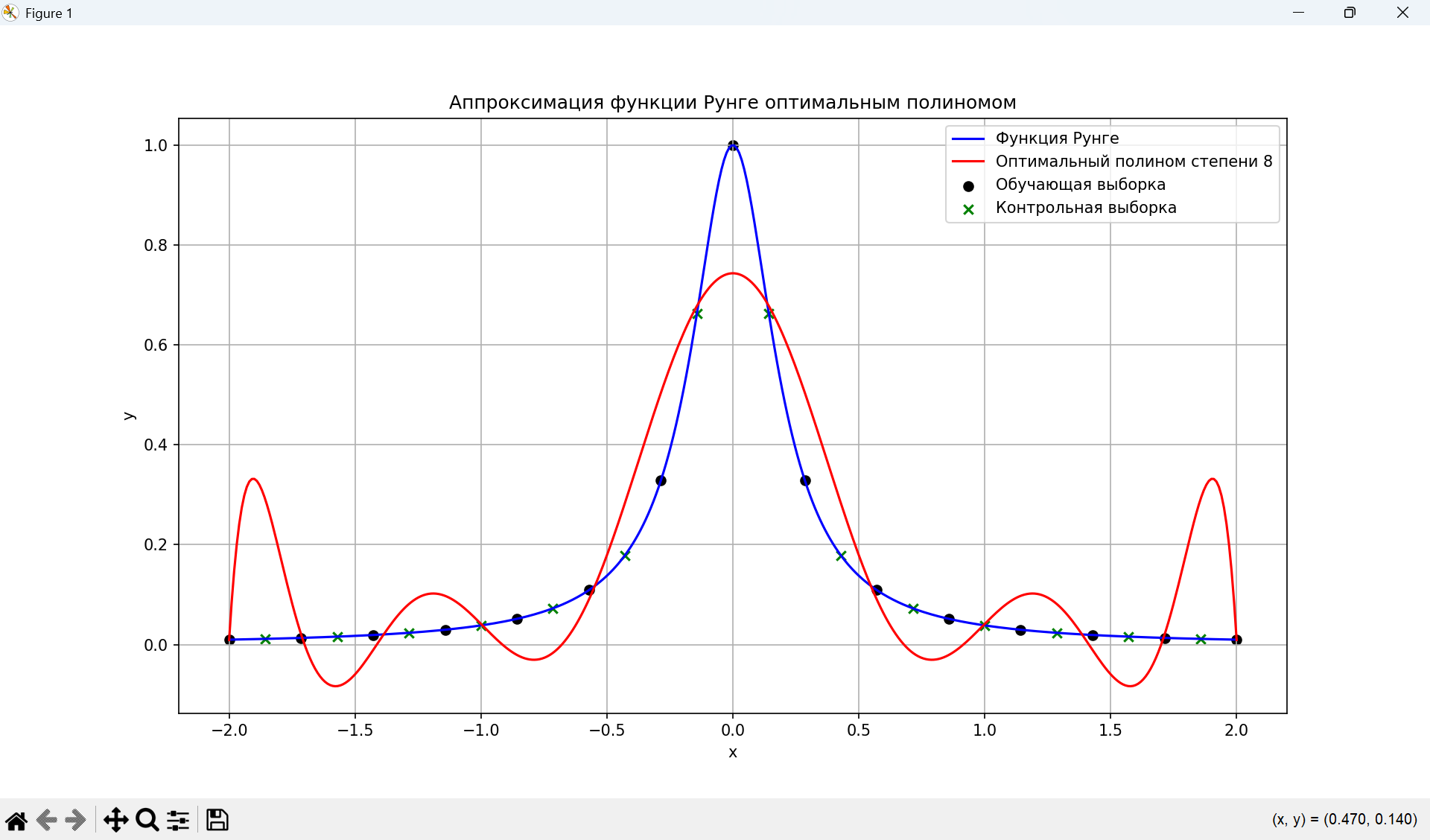


Рисунок 2 Аппроксимация функции Рунге



Рисунок 3 Оптимальная степень полинома и ошибка

**Листинг программы:**

import numpy as np

import matplotlib.pyplot as plt

from sklearn.preprocessing import PolynomialFeatures

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error

def runge\_function(x):

return 1 / (1 + 25 \* x\*\*2)

l = 15

X\_train = np.array([4 \* ((i - 1) / (l - 1)) - 2 for i in range(1, l + 1)])

y\_train = runge\_function(X\_train)

X\_test = np.array([4 \* ((i - 0.5) / (l - 1)) - 2 for i in range(1, l)])

y\_test = runge\_function(X\_test)

max\_degree = 10

results = []

for degree in range(1, max\_degree + 1):

poly = PolynomialFeatures(degree=degree)

X\_poly = poly.fit\_transform(X\_train.reshape(-1, 1))

model = LinearRegression().fit(X\_poly, y\_train)

y\_pred\_train = model.predict(X\_poly)

mse\_train = mean\_squared\_error(y\_train, y\_pred\_train)

X\_test\_poly = poly.transform(X\_test.reshape(-1, 1))

y\_pred\_test = model.predict(X\_test\_poly)

mse\_test = mean\_squared\_error(y\_test, y\_pred\_test)

results.append((degree, mse\_train, mse\_test, model, poly))

print(f"Степень {degree}: MSE(train)={mse\_train:.6f}, MSE(test)={mse\_test:.6f}")

best\_degree, best\_train\_mse, best\_test\_mse, best\_model, best\_poly = min(

results, key=lambda t: t[1] # выбор по ошибке на обучающей выборке

)

print(f"\nОптимальная степень полинома: {best\_degree}")

print(f"MSE(train)={best\_train\_mse:.6f}, MSE(test)={best\_test\_mse:.6f}")

x\_plot = np.linspace(-2, 2, 400)

y\_plot = runge\_function(x\_plot)

y\_poly\_plot = best\_model.predict(best\_poly.transform(x\_plot.reshape(-1, 1)))

plt.figure(figsize=(10, 6))

plt.plot(x\_plot, y\_plot, label="Функция Рунге", color="blue")

plt.plot(x\_plot, y\_poly\_plot, label=f"Оптимальный полином степени {best\_degree}", color="red")

plt.scatter(X\_train, y\_train, color="black", marker="o", label="Обучающая выборка")

plt.scatter(X\_test, y\_test, color="green", marker="x", label="Контрольная выборка")

plt.title("Аппроксимация функции Рунге оптимальным полиномом")

plt.xlabel("x")

plt.ylabel("y")

plt.legend()

plt.grid(True)

plt.show()

**Задание №2 (4 вариант):**

Допустим, что задана решающая функция линейного классификатора *f*(*x*1, *x*2).

Найти координаты и значение функции в точке минимума, используя встроенные функции (пример решения задачи в системе WolframAlpha показан на рисунке). Построить график функции с точкой решения. Вид функции задан в таблице. Номер варианта – номер по списку.

Функция по варианту:



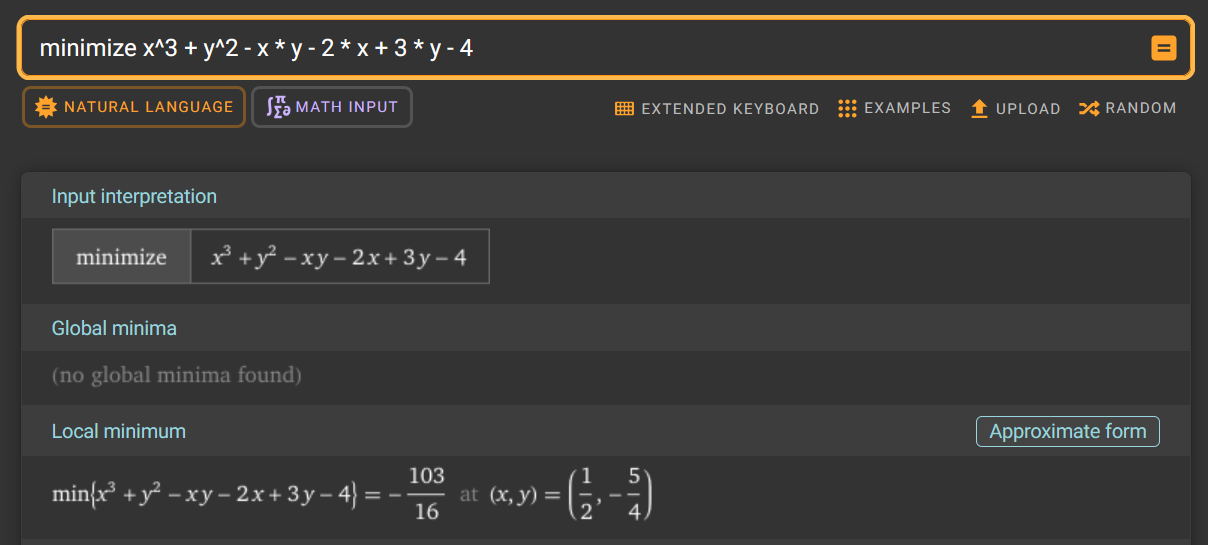


Рисунок 4 Координаты и значение функции в точке минимума

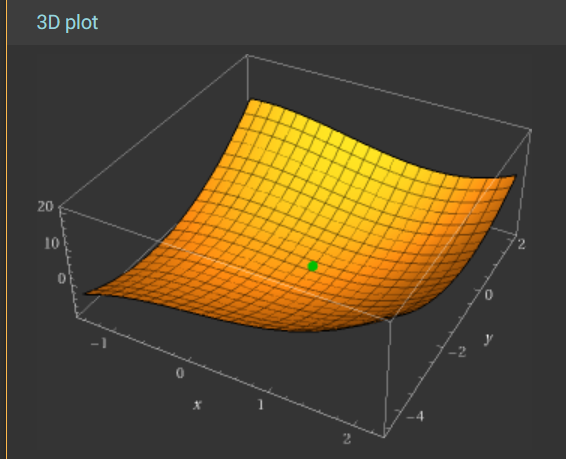


Рисунок 5 Построенный график

**Вывод:** в ходе лабораторной работы я построил полиномиальную модель для аппроксимации функции Рунге, подобрал её оптимальную степень и оценил качество на контрольной выборке. Выявлены признаки переобучения модели.