МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Информатика»

**Отчет по лабораторной работе №2.1**

**по дисциплине**

**«Базовые средства математических пакетов»**

Выполнил: студент гр. БЭИ2202

Кулешов А. С.

Вариант 14.

Проверил: доцент Шакин В. Н.

Москва, 2024г.

1. **Цель работы:** Изучение технологий интерполяции с использованием функций Scilab, получение интерполяционных многочленов в явном виде и построение их графиков.
2. **Индивидуальное задание**
3. ***Изучить материал учебника*** [1](*п. 2.2).*
4. ***Выбрать индивидуальное задание****: номера узлов и номер аппроксимируемой функции из* ***табл. 2.2-1****; узлы интерполяции и значения функции в узлах из* ***табл. 2.2-2****.*
5. ***Создать сценарий*** *для выполнения практического задания.*
6. ***Задать*** *в виде векторов значения узлов и соответствующие им значения функции.*
7. ***Вычислить коэффициенты*** *функций для линейной, квадратичной и кубической интерполяции с использованием* **datafit** *и* ***получить*** *соответствующие функции в явном виде.*
8. ***Вычислить*** *с использованием полученных аппроксимирующих функций значения в произвольной точке, принадлежащей отрезку, но не совпадающей с узлами аппроксимации, и* ***сравнить полученные*** *результаты.*
9. ***Построить графики*** *табличной и трех функций в одном графическом окне, снабдив их легендой* [1]*.*
10. ***Провести*** *линейную и кубическую интерполяцию функции, заданной таблично, с использованием* **interpln** *и* **interp***.* ***Получить*** *значения интерполирующих функций в точке, не совпадающей с узлами интерполяции, и* ***проанализировать*** *полученные результаты.*
11. ***Построить*** *графики табличной и двух интерполирующих функций в одном графическом окне, снабдив их легендой* [1].
12. ***Спроектировать и реализовать приложение****:* «*Интерполяции функций» для ввода данных и отображения результатов (по требованию преподавателя).*
13. ***Предоставить*** *результаты работы преподавателю и* ***ответить*** *на поставленные вопросы.*
14. ***Оформить отчет*** *по выполненной работе.*



1. Выполнение задания

import plotly.graph\_objects as go  
import pandas as pd  
import numpy as np  
from scipy.interpolate import interp1d  
  
def polynm(x):  
 return x\*\*4 - 6 \* x\*\*3 + 13 \* x \*\* 2 - 12 \* x + 4  
  
x = np.array([-5, -3.5, -2.5, -1.5, -0.5, 0, 0.5])  
y = np.array([polynm(v) for v in x])  
  
# Создаем DataFrame для хранения результатов интерполяции  
df\_interp = pd.DataFrame({'x': x, 'y': y})  
  
fig = go.Figure()  
  
# Добавляем исходные данные на график  
fig.add\_trace(go.Scatter(x=x,  
 y=y,  
 mode='markers',  
 marker=dict(color='LightSkyBlue', size=15, line=dict(color='MediumPurple', width=3)),  
 name="Исходные данные"))  
  
  
# Интерполируем данные  
interp\_func = interp1d(x[::2], y[::2], kind='cubic')  
  
# Вычисляем среднеквадратичное отклонение  
y\_pred = interp\_func(x)  
df\_interp['interp'] = y\_pred  
sko = np.sqrt(np.mean((y - y\_pred) \*\* 2))  
print("СКО при кубической интерполяции (4 корня): ", sko)  
print(df\_interp)  
  
# Генерируем значения для построения графика интерполяции  
x\_interp1 = np.linspace(x[::2].min(), x[::2].max(), 100)  
y\_interp1 = interp\_func(x\_interp1)  
  
# Строим график интерполяции  
fig.add\_trace(go.Scatter(x=x\_interp1,  
 y=y\_interp1,  
 name="Интерполяция P1, кубическая, используя 4 корня"))  
  
# Интерполируем данные  
interp\_func = interp1d(x, y, kind='cubic')  
  
# Генерируем значения для построения графика интерполяции  
x\_interp = np.linspace(x.min(), x.max(), 100)  
y\_interp = interp\_func(x\_interp)  
  
# Строим график интерполяции  
fig.add\_trace(go.Scatter(x=x\_interp,  
 y=y\_interp,  
 name="Интерполяция P2, кубическая, используя 7 корней"))  
  
# Интерполируем данные  
interp\_func = interp1d(x, y, kind='linear')  
  
  
# Генерируем значения для построения графика интерполяции  
x\_interp = np.linspace(x.min(), x.max(), 100)  
y\_interp = interp\_func(x\_interp)  
  
# Строим график интерполяции  
fig.add\_trace(go.Scatter(x=x\_interp,  
 y=y\_interp,  
 name="Интерполяция P3, линейная, используя 7 корней"))  
  
x\_interp1 = np.linspace(x.min(), x.max(), 100)  
y\_interp1 = polynm(x\_interp1)  
# Строим график интерполяции  
fig.add\_trace(go.Scatter(x=x\_interp1,  
 y=y\_interp1,  
 name="Изначальная функция многочлена"))  
  
  
  
  
fig.update\_layout(title="Построение графика интерполяции",  
 xaxis\_title="Значение аргумента",  
 yaxis\_title="Значение интерполяции")  
  
fig.show()

1. Полученный результат, рисунок 1

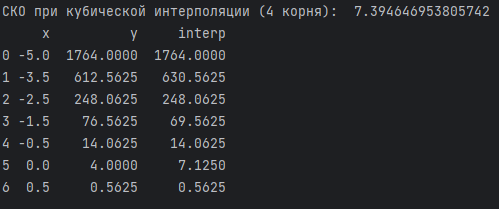


Рисунок 1 – Результат работы программы

1. Полученный график, рисунок 2

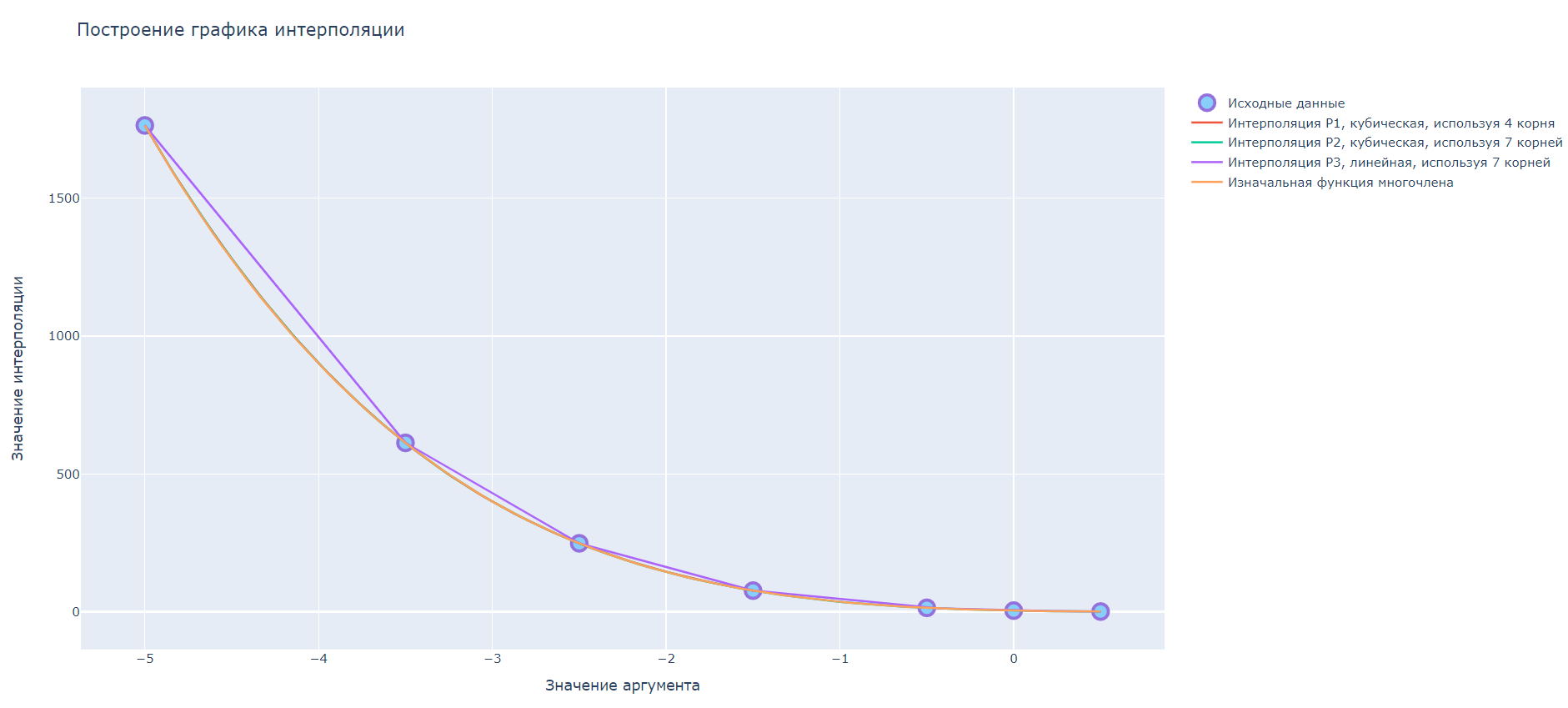


Рисунок 2 – Полученный график интерполяции

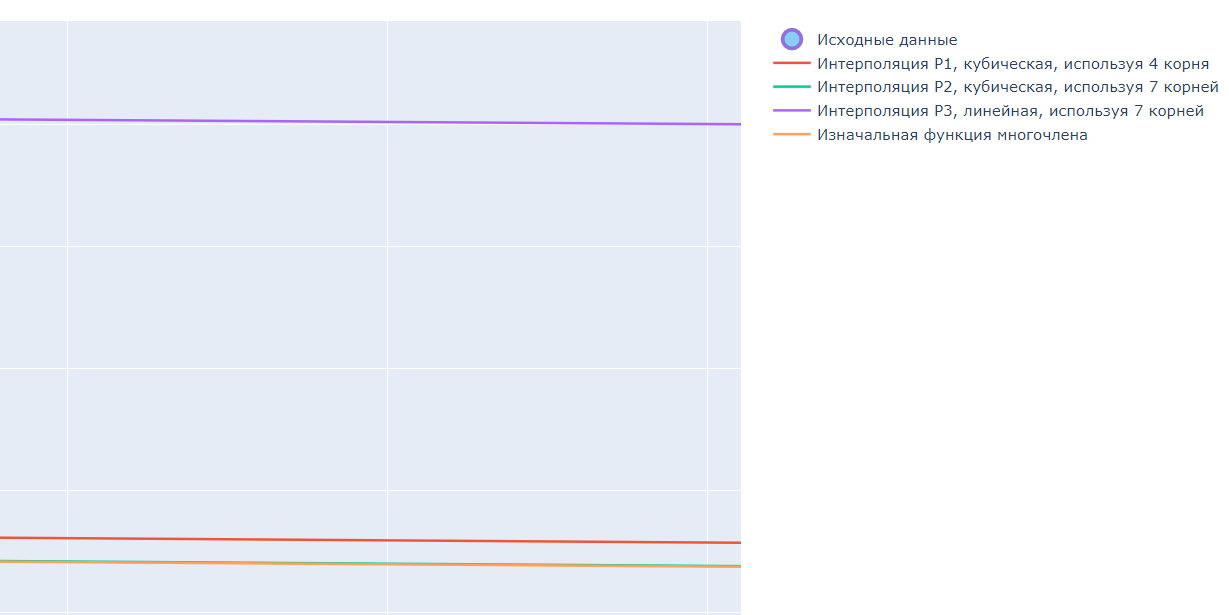


Рисунок 3 – Демонстрация разницы в точности у разных методов



Рисунок 4 – Демонстрация разницы в точности у кубического метода с разным числом корней