数值分析大作业实验报告

选题

二、函数逼近

（1） 请写一个程序，可以分别实现离散数据的 3 次和 4 次多项式拟合。

（2） 对于已知的如下 6 对离散数据，请给出 3 次和 4 次拟合结果，并估计最小均方误差。

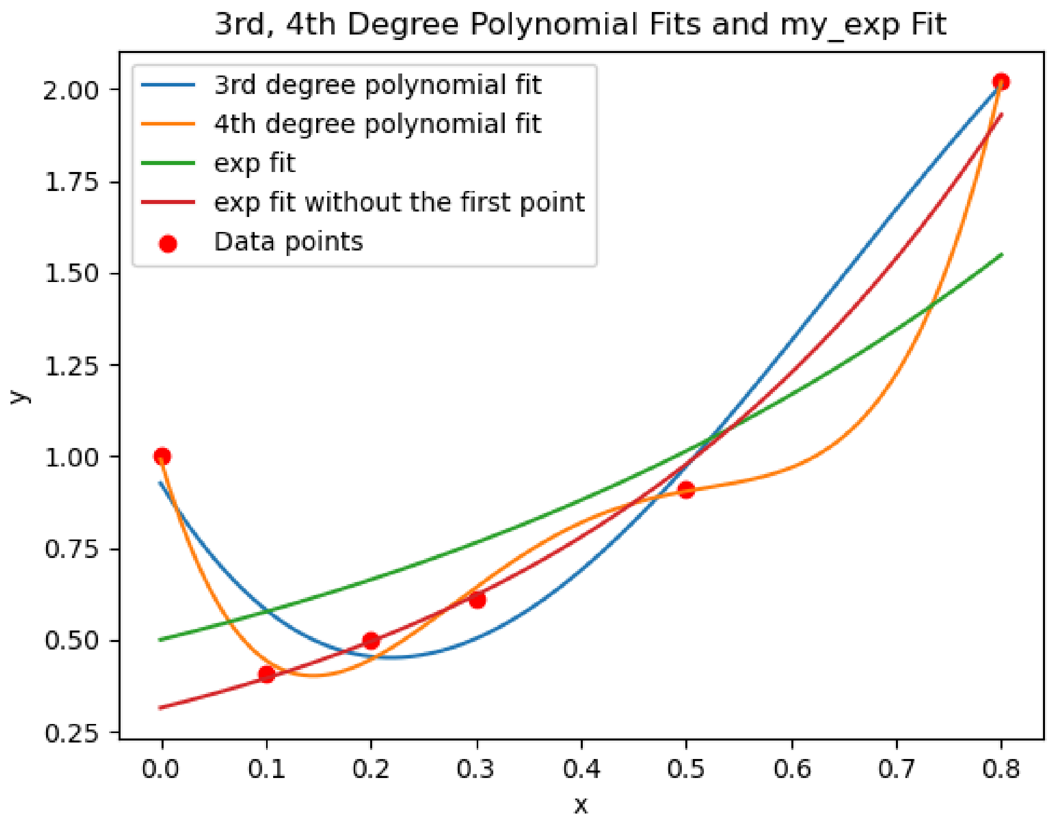
$$\{0.00, 1.00\},\{0.10, 0.41\},\{0.20, 0.50\},\{0.30, 0.61\},\{0.50, 0.91\},\{0.80, 2.02\}$$

请根据数据曲线形状，求出更合适的拟合曲线，画图比较三种拟合曲线。

- 问题描述：实现离散点对的多项式拟合
- 数值求解方法：
  - 对于任务中的多项式逼近，采用最小二乘逼近，其中在解矩阵方程中使用了高斯消元法。
  - 在自己给出的拟合模型中，我最开始考虑建模$ax + \frac{b}{x+c} + d$，即用双曲模型进行最小二乘逼近，但是发现解方程的过程过于繁琐。用指数函数模型$\alpha e^{\beta x}$进行直接最小二乘模拟会导致均方误差较大。于是我也考虑了将第一个点视作有较大统计误差的点，对后五个点做指数模型拟合。

结果分析

- 代码分析：
  - main.py：包括函数逼近功能的主要函数。其中`poly_fit`函数用于做多项式近似，`exp_fit`函数用于做指数拟合，`delta`函数用于计算均方误差。`plot_combined_fit`函数用于将所有的曲线绘制在一张图片上。
  - matrix.py：包含`Gause_solve`函数，使用高斯消元法解方程组的函数。
- 代码执行结果如下：




```
请输入离散数据点的数量： 6
请输入第1对离散数据点（格式： x,y） : 0.00,1.00
请输入第2对离散数据点（格式： x,y） : 0.10,0.41
请输入第3对离散数据点（格式： x,y） : 0.20,0.50
请输入第4对离散数据点（格式： x,y） : 0.30,0.61
请输入第5对离散数据点（格式： x,y） : 0.50,0.91
请输入第6对离散数据点（格式： x,y） : 0.80,2.02
3次多项式拟合误差： 0.052549
4次多项式拟合误差： 0.005210
指数函数拟合误差： 0.560775
去掉首点的指数函数拟合误差： 0.013059
```

可以看到，用四次多项式拟合的效果比三次多项式明显更强，拟合误差下降了一个数量级。但是四次多项式的图像较为奇怪，存在过拟合的可能。

但是在我们简化模型为指数模拟时，直接对六个点做模拟的效果较差，均方误差比较大。因此我考虑了去掉第一个点，将其视为统计中出现了较大误差的点，此时的指数模拟模型对后五个点的拟合效果有了很大提升。