问题描述

对于平面n个点$p_j(x_j, y_j), j = 1, 2, \ldots, n$，根据拟合算法求解其逼近函数，并用鼠标交互。

1. 用多项式函数$B_i(x) = x^i, i = 0, 1, \ldots, n-1$来实现。
2. 用径向及函数$f(x) = b_0 + \sum_{i=1}^{n} b_i g_i(x)$来实现，其中$g(x) = \frac{1}{|x-p_i|^2+d}$
   ==> 思考：变量比方程多，如何加约束条件。
   ==> 思考：常数b_0也可以改成一个低次的多项式，也要加约束条件。

算法实现

我们只需要对计算逼近函数构造一个相同的框架：对于逼近$\sum_{i=1}^{n} b_i f_i(x) = g(x)$
的逼近，考虑逼近条件$\sum_{i=1}^{n} b_i f_i(x_j) = g(x_j)$，得到方程

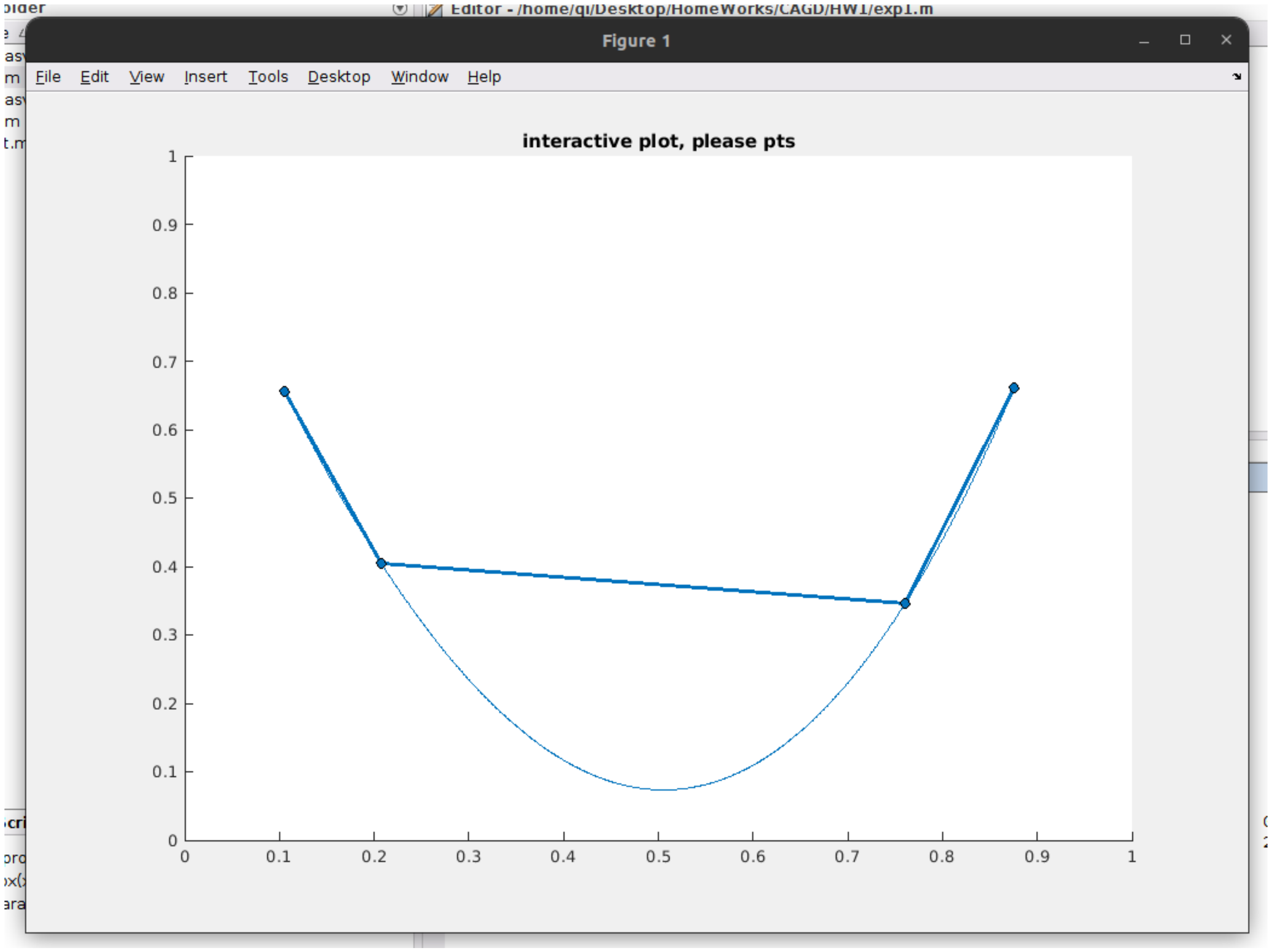
$$\begin{pmatrix} f_1(x_1) & \cdots & f_n(x_1) \\ f_1(x_2) & \cdots & f_n(x_2) \\ \vdots & \ddots & \vdots \\ f_1(x_n) & \cdots & f_n(x_n) \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} g(x_1) \\ g(x_2) \\ \vdots \\ g(x_n) \end{pmatrix}$$

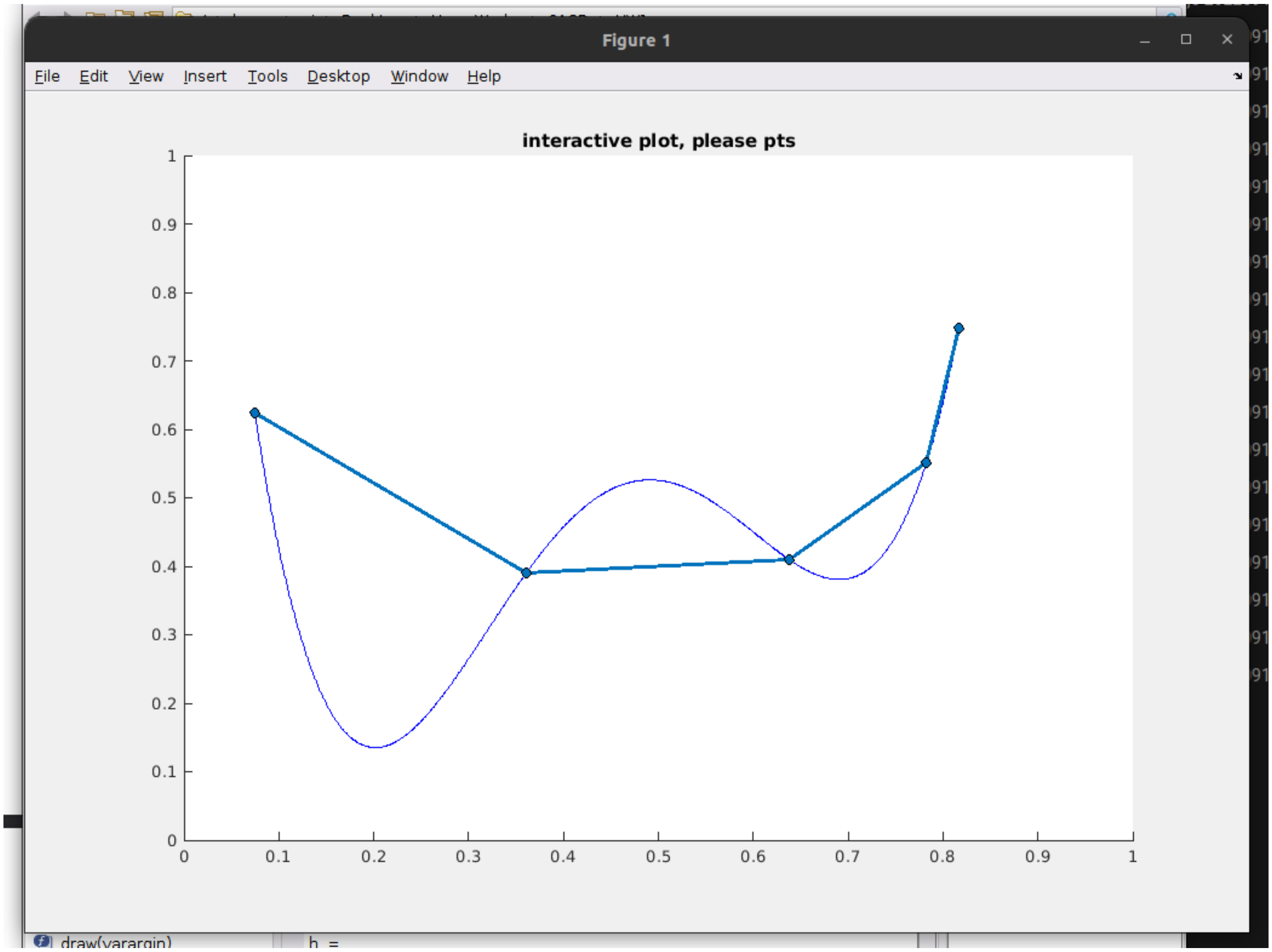
考虑其解，然后交互地画出即可。

对于未知变量比参数多的，我们考虑用最小二乘解，具有十分好的鲁棒性。
也能消掉一个变量(这等同于任何固定b_0的方法，因为考虑逼近$g(x) - b_0$即可)

结果比较

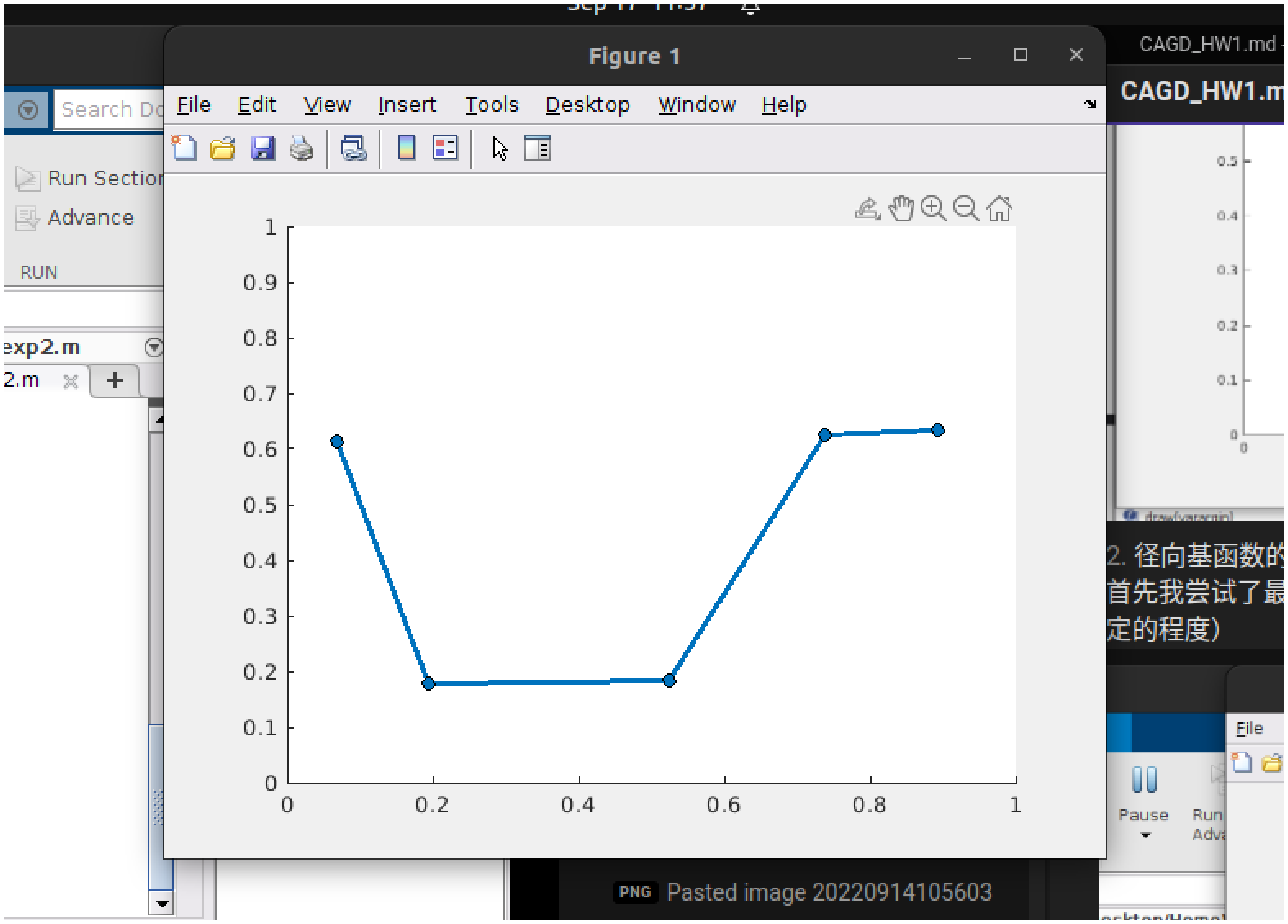
1. 幂级数函数的拟合

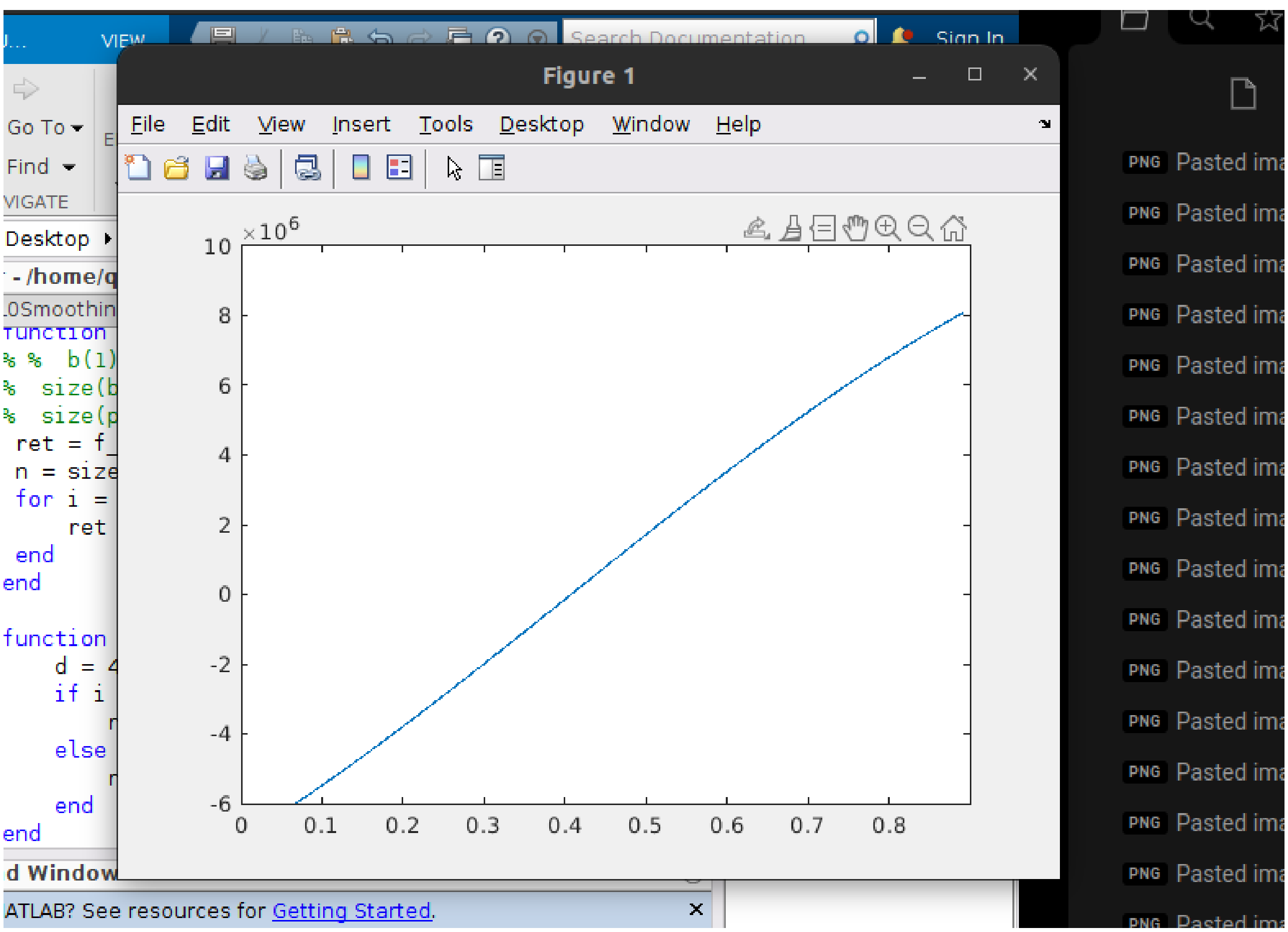

Editor - /home/qi/Desktop/HomeWorks/CAGD/HW1/exp1.m
Figure 1
File Edit View Insert Tools Desktop Window Help
interactive plot, please pts
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1



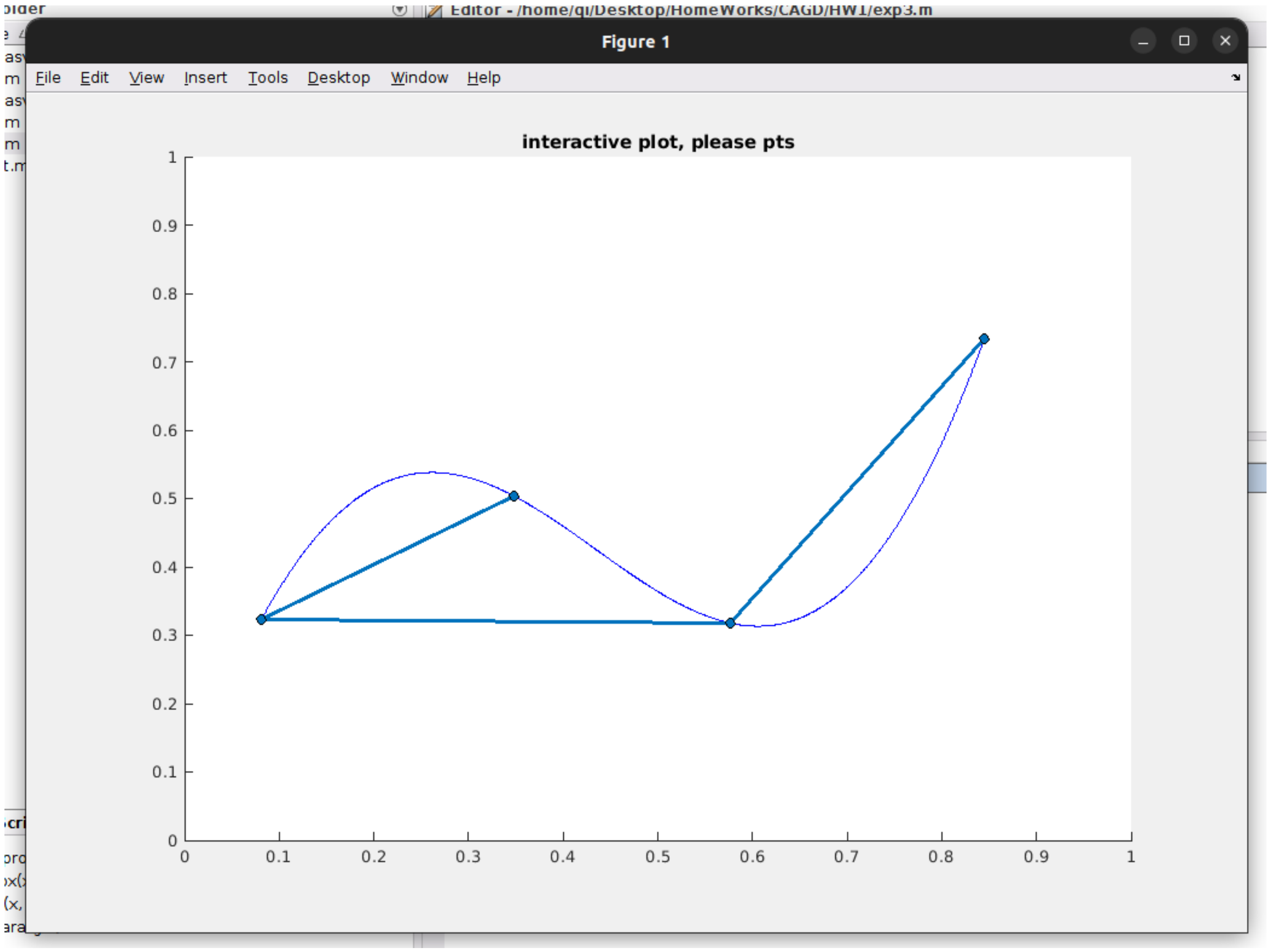
2. 径向基函数的拟合

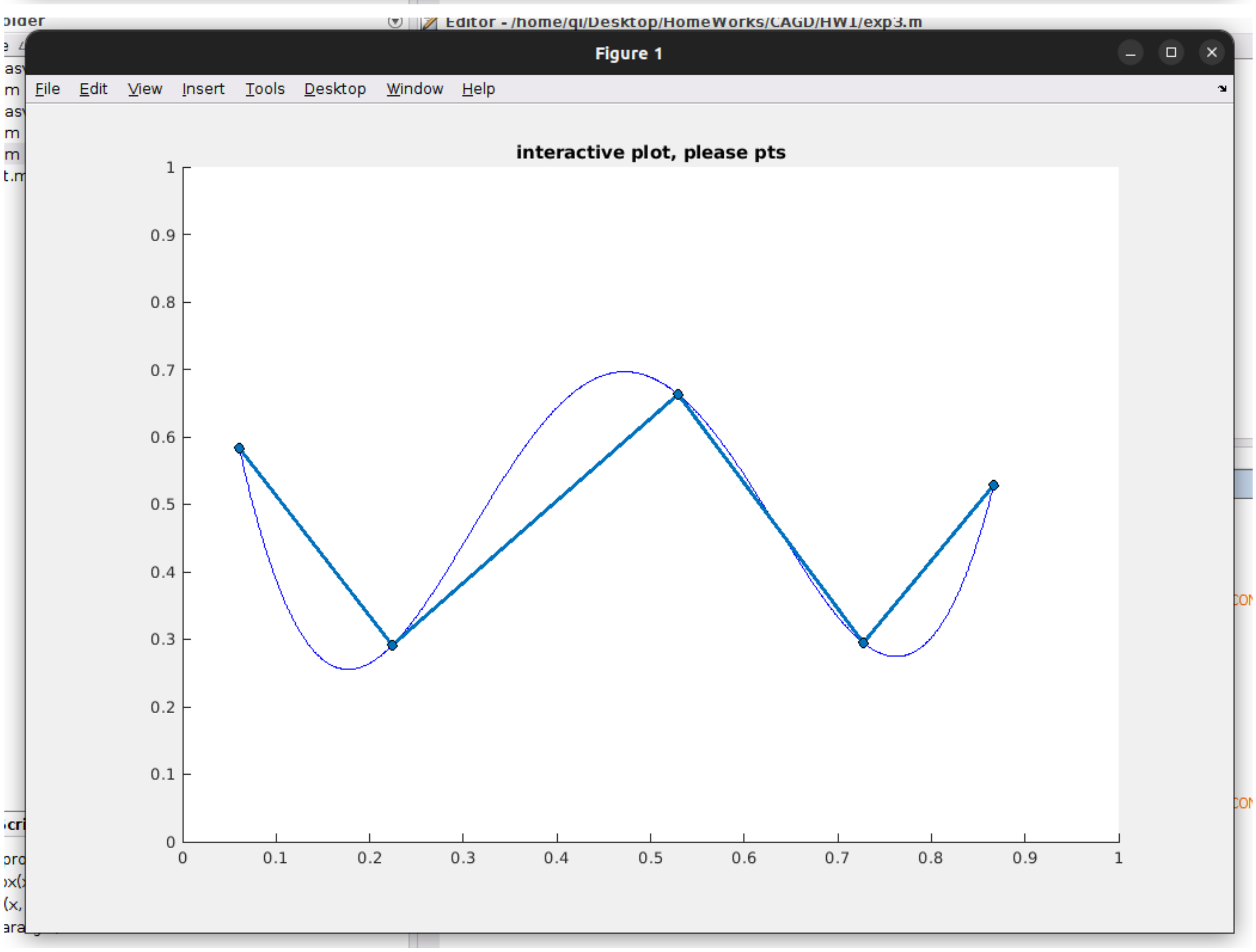
   首先我尝试了最小二乘解，得到的效果并不理想（可能因为最后导致b处的值堆积到一定的程度）





然后我采用了设$b_0=0$,得到的结果比较理想


Editor - /home/qi/Desktop/HomeWorks/CAGD/HW1/exp3.m
Figure 1
File Edit View Insert Tools Desktop Window Help
interactive plot, please pts
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Editor - /home/qi/Desktop/HomeWorks/CAGD/HW1/exp3.m
Figure 1
File Edit View Insert Tools Desktop Window Help
interactive plot, please pts
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

结论

1. 径向基函数的逼近性质更加好，效果理想，但是要注意用纯径向函数
2. 最小二乘解要注意实际情况，有可能出现极大无比的值（有可能是bug）.