

平面点列的曲线拟合

张景浩 id: 037 PB20010399

2023 年 4 月 29 日

摘要

在日常处理问题时，我们常常需要通过一些拟合的方法去处理数据之间的关系，目的是为了对数据的走向和分布有一个准确的了解。本次实验我们以二维平面上点列的曲线拟合为例，探讨不同的参数化形式和拟合基函数对于拟合曲线的影响，验证当参数化足够精确的近似于弧长时，拟合的曲线就可以达到完全近似阶^[1]，并初步对神经网络进行了解与实践。

关键词：参数化、曲线拟合、基函数、神经网络

一、前言

1.1 点列参数化

在几何建模中的一个常见的需求是使用参数多项式或者样条曲线插值取样的一系列点或导数，要做到这一点，我们就必须选择每一个取样点应得参数值，这种选择将影响拟合的准确性，详细的讨论和证明可以参考文章 [Parameterization for Curve Interpolation](#) ^[1] 和 [Choosing nodes in parameteric curve interpolation](#) ^[2]。

1.2 拟合基函数

当我们确定了点列的参数化方法后，就可以去尝试使用一些函数去分别对分量 x 和 y 去拟合这些参数，从而得到平面上的参数曲线。一般而言，我们会选择一些相对完备的函数空间中的一组基函数对参数点列进行拟合，在本次实验中重点尝试了多项式基函数和 Gauss 基函数，并在此之上尝试了多项式基函数的最小二乘拟合以及基于 Gauss 基函数的神经网络拟合。

二、问题分析

给定了平面中的一组点列，我们的目标是将它们处理成条尽可能光滑、误差小的曲线。考虑到点列在曲面上的任意性，所以目标曲线可能不是一个 $x \rightarrow y$ 的映射，所以一般的函数拟合是不做考虑的。自然地，我们会想到将曲线做参数化处理，对分量 x 和 y 分别进行函数拟合，最后作为参数曲线进行输出。要实现这个方法，我们就要考虑如何对点列进行参数化以及用什么类型的函数去对分量进行拟合才能达到预期的效果，在本次实验中的参数化方法选择了均匀参数化、弦长参数化、中心参数化以及 Foley-Nielsen 参数化 ^[2]，拟合基函数选择了多项式基函数和 Gauss 基函数，并尝试了不同的拟合思想。

三、数学模型建立

3.3 参数化方法

对于给定 R^2 中一组有序点 $\{P_i = (x_i, y_i)\}_{i=0}^n$ ，选择一组参数 $\{t_i\}_{i=0}^n$ ，使得存在一一映射 $f: t_i \rightarrow P_i$ 。

3.3.1 均匀参数化

均匀参数化是最简单的一种参数化方法，将取样点按照顺序将 t_i 均匀分布在 $[0,1]$ 中：

$$t_0 = 0, t_i - t_{i-1} = \frac{1}{n}, i = 1, \dots, n$$

3.3.2 弦长参数化

因为参数化越近似于弧长，拟合的曲线就越精确，所以有如下弦长参数化：

$$t_0 = 0, t_i - t_{i-1} = \frac{\|P_i - P_{i-1}\|_2}{\sum \|P_i - P_{i-1}\|_2}, i = 1, \dots, n$$

3.3.3 中心参数化

在弦长参数化的基础上，进一步演化出了中心参数化：

$$t_0 = 0, t_i - t_{i-1} = \frac{\sqrt{\|P_i - P_{i-1}\|_2}}{\sum \sqrt{\|P_i - P_{i-1}\|_2}}, i = 1, \dots, n$$

3.3.4 Foley-Nielsen 参数化

Foley-Nielsen 参数化使用 Nielsen 尺度来测量相邻采样点之间的距离，从而对曲线的旋转、平移和放缩具有仿射不变性，公式如下：

$$t_0 = 0, t_i - t_{i-1} = \frac{N(P_i, P_{i-1})}{\sum N(P_i, P_{i-1})}, i = 1, \dots, n$$

$$N(P_i, P_{i-1}) = \begin{cases} d_1(1 + \frac{3}{2} \frac{\hat{\alpha}_1 d_1}{d_1 + d_2}) \\ d_i(1 + \frac{3}{2} \frac{\hat{\alpha}_{i-1} d_{i-1}}{d_{i-1} + d_i} + \frac{3}{2} \frac{\hat{\alpha}_i d_i}{d_i + d_{i+1}}) & i = 2, \dots, n-1 \\ d_n(1 + \frac{3}{2} \frac{\hat{\alpha}_{n-1} d_{n-1}}{d_{n-1} + d_n}) \end{cases}$$

其中， $d_i = \|P_i - P_{i-1}\|_2$, $\alpha_i = \text{angle}(P_{i-1}, P_i, P_{i+1})$, $\hat{\alpha}_i = \min(\pi - \alpha_i, \frac{\pi}{2})$

3.4 曲线拟合

对于给定 R^2 中一组有序点 $\{P_i = (x_i, y_i)\}_{i=0}^n$ ，有一组参数 $\{t_i\}_{i=0}^n$ ，存在一一映射 $t_i \rightarrow P_i$ ，找到一个向量值函数 $P = f(t) : R \rightarrow R^2$ ，使得 $f(t)$ 的轨迹尽可能地靠近这组点。

3.4.1 多项式插值法

$$\begin{pmatrix} x \\ y \end{pmatrix} = f(t) = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} t + \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} t^2 + \dots + \begin{pmatrix} a_n \\ b_n \end{pmatrix} t^n$$

将参数点列带入求解线性方程组：

$$\begin{pmatrix} 1 & t_0 & t_0^2 & \dots & t_0^n \\ 1 & t_1 & t_1^2 & \dots & t_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & t_n^2 & \dots & t_n^n \end{pmatrix} \begin{pmatrix} a_0 & b_0 \\ a_1 & b_1 \\ \vdots & \vdots \\ a_n & b_n \end{pmatrix} = \begin{pmatrix} x_0 & y_0 \\ x_1 & y_1 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix}$$

就可以得到拟合多项式的系数，但在实际应用中，由于 Vandermonde 矩阵较为病态，求解线性方程组误差会很大，所以往往采用 Lagrange 插值法和 Newton 插值法来计算拟合多项式。

3.4.2 Gauss 基函数插值法

$$\begin{pmatrix} x \\ y \end{pmatrix} = f(t) = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} g_1(t) + \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} g_2(t) + \cdots + \begin{pmatrix} a_n \\ b_n \end{pmatrix} g_n(t)$$

$$g_i(t) = e^{-\frac{(t-t_i)^2}{2\sigma^2}}, \quad i = 0, 1, \dots, n$$

将参数点列带入求解线性方程组：

$$\begin{pmatrix} 1 & g_0(t_0) & g_1(t_0) & \cdots & g_n(t_0) \\ 1 & g_0(t_1) & g_1(t_1) & \cdots & g_n(t_1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & g_0(t_n) & g_1(t_n) & \cdots & g_n(t_n) \end{pmatrix} \begin{pmatrix} a_0 & b_0 \\ a_1 & b_1 \\ \vdots & \vdots \\ a_n & b_n \end{pmatrix} = \begin{pmatrix} x_0 & y_0 \\ x_1 & y_1 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix}$$

由于线性方程组的系数矩阵是 $(n+1) \times (n+2)$ 的，所以方程解不唯一，我们可以添加一个归一化约束： $\sum a_i = \sum b_i = 1$ ，来得到方程的一个唯一解。

3.4.3 RBF 神经网络

将一般的 Gauss 函数表达为标准的 Gauss 函数的形式：

$$g_{\mu,\sigma}(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(t-\mu)^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t}{\sigma} - \frac{\mu}{\sigma})^2} = g_{0,1}(at + b)$$

我们得到：

$$\begin{pmatrix} x \\ y \end{pmatrix} = f(t) = \begin{pmatrix} \lambda_0 + \sum_{i=0}^n \lambda_i g_{0,1}(a_i t + b_i) \\ \mu_0 + \sum_{i=0}^n \mu_i g_{0,1}(c_i t + d_i) \end{pmatrix}$$

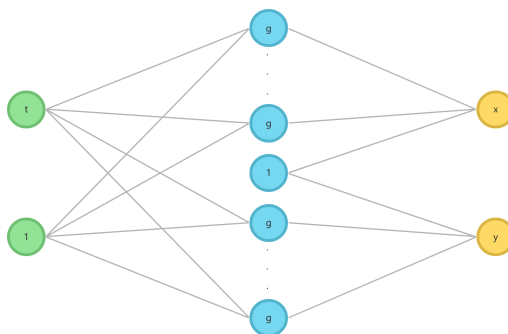


图 1: RBF 神经网络图示

3.4.4 最小二乘法

在多项式拟合的基础上，我们为了使拟合曲线尽可能地靠近目标点列，使用最高次数不超过 $m(m < n)$ 的多项式作为插值函数：

$$\begin{pmatrix} x \\ y \end{pmatrix} = f(t) = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} t + \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} t^2 + \cdots + \begin{pmatrix} a_m \\ b_m \end{pmatrix} t^m$$

将参数点列带入求解最小二乘问题:

$$\begin{pmatrix} 1 & t_0 & t_0^2 & \cdots & t_0^m \\ 1 & t_1 & t_1^2 & \cdots & t_1^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & t_n^2 & \cdots & t_n^m \end{pmatrix} \begin{pmatrix} a_0 & b_0 \\ a_1 & b_1 \\ \vdots & \vdots \\ a_m & b_m \end{pmatrix} = \begin{pmatrix} x_0 & y_0 \\ x_1 & y_1 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix}$$

就可以得到拟合多项式的系数。

3.4.5 三次自然样条函数

我们在每一个子区间 $[t_i, t_{i+1}]$ 上用一个三次多项式 $S_i(x)$ 去插值, $i = 0, \dots, n-1$, 满足在参数结点处 $S_{i-1}^{(m)}(t_i) = S_i^{(m)}(t_i)$, $m = 0, 1, 2$ 。

首先由于多项式 $S_{i-1}(t_i) = y_i = S_i(t_i)$ ($1 \leq i \leq n-1$) 有相同的插值, 所以 S 是连续的。我们定义一组数 $z_i = S''(t_i)$, 因为 S'' 在每个内结点上连续, 所以显然, 对于 $0 \leq i \leq n$, 存在 z_i 且满足:

$$\lim_{x \rightarrow t_i^-} S''(x) = z_i = \lim_{x \rightarrow t_i^+} S''(x) \quad (1 \leq i \leq n-1) \quad (1)$$

因为 S_i 是 $[t_i, t_{i+1}]$ 上的三次多项式, 所以 S_i'' 也是 $[t_i, t_{i+1}]$ 上的一次函数:

$$S_i''(x) = \frac{z_i}{h_i}(t_{i+1} - x) + \frac{z_{i+1}}{h_i}(x - t_i) \quad (2)$$

其中, $h_i = t_{i+1} - t_i$ 。将 (2) 式积分两次并结合插值条件 $S_i(t_i) = y_i$ 和 $S_i(t_{i+1}) = y_{i+1}$, 我们得到:

$$S_i(x) = \frac{z_i}{6h_i}(t_{i+1} - x)^3 + \frac{z_{i+1}}{6h_i}(x - t_i)^3 + \left(\frac{y_{i-1}}{h_i} - \frac{z_{i+1}h_i}{6}\right)(x - t_i) + \left(\frac{y_i}{h_i} - \frac{z_ih_i}{6}\right)(t_{i+1} - x) \quad (3)$$

接下来通过 S' 的连续性来确定 z_1, z_2, \dots, z_{n-1} , 得到等式:

$$h_{i-1}z_{i-1} + 2(h_i + h_{i-1})z_i + h_iz_{i+1} = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}) \quad (4)$$

令 $\mu_i = \frac{h_{i-1}}{h_i + h_{i-1}}, \lambda_i = \frac{h_i}{h_i + h_{i-1}} = 1 - \mu_i, g_i = \frac{6}{h_{i-1} + h_i} \left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} \right)$, 所以 (4) 式简化为:

$$\mu_iz_{i-1} + 2z_i + \lambda_iz_{i+1} = g_i$$

带入边界条件 $S'(0) = S'(1) = 0$ 后有:

$$g_0 = 2z_0 + z_1 = \frac{6}{h_0} \frac{y_1 - y_0}{h_0}$$

$$g_n = z_{n-1} + 2z_n = -\frac{6}{h_{n-1}} \frac{y_n - y_{n-1}}{h_{n-1}}$$

解方程组:

$$\begin{bmatrix} 2 & 1 & & & \\ \mu_1 & 2 & \lambda_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 2 & \lambda_{n-1} \\ & & & 1 & 2 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{n-1} \\ g_n \end{bmatrix}$$

即可得到系数 z_0, z_1, \dots, z_n , 带入 (3) 式得三次样条函数。

四、结果与对比

4.5 结果展示

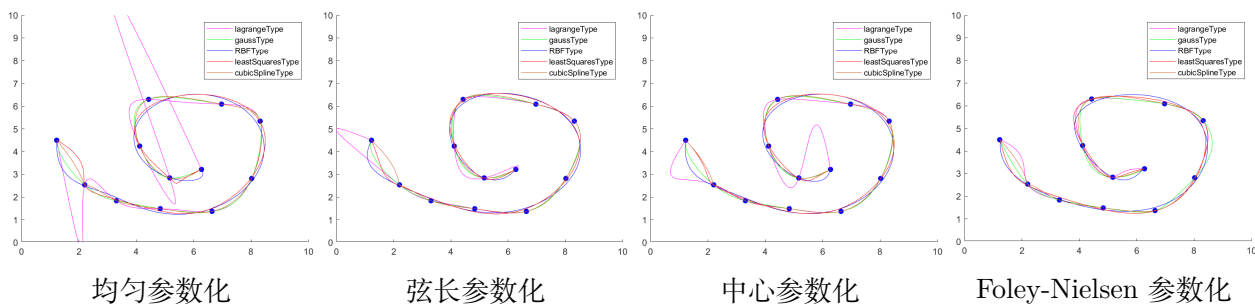


图 2: 不同的拟合曲线比较

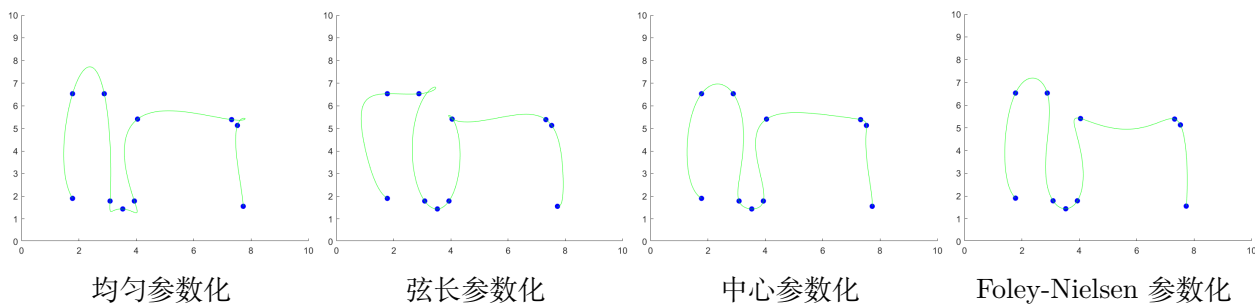


图 3: Gauss 基函数在不同参数化下的拟合效果

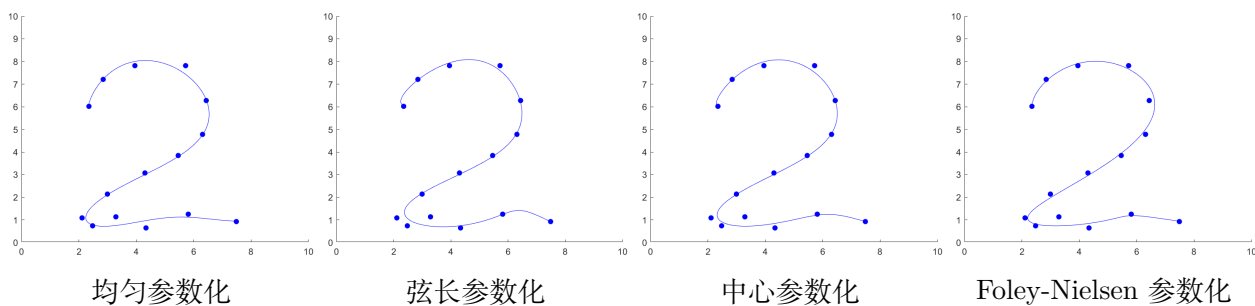


图 4: RBF 神经网络在不同参数化下的拟合效果

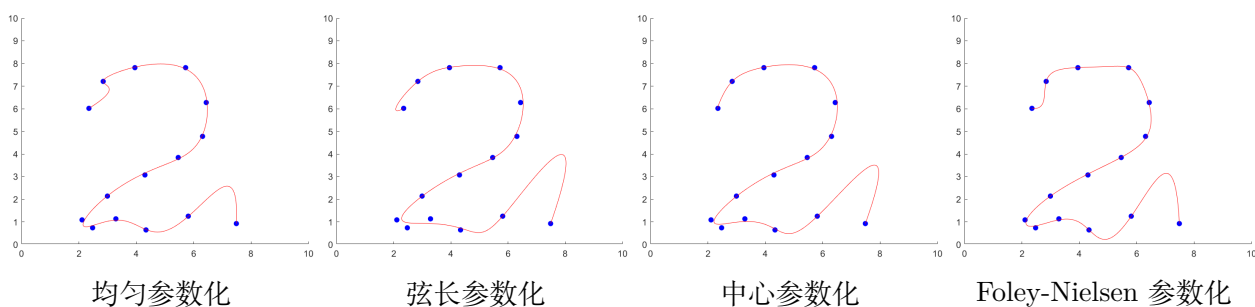


图 5: 最小二乘法在不同参数化下的拟合效果

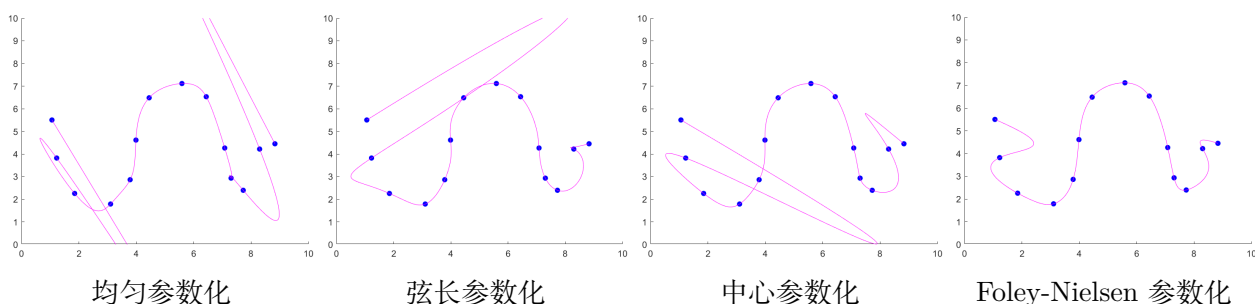


图 6: 多项式基函数在不同参数化下的拟合效果

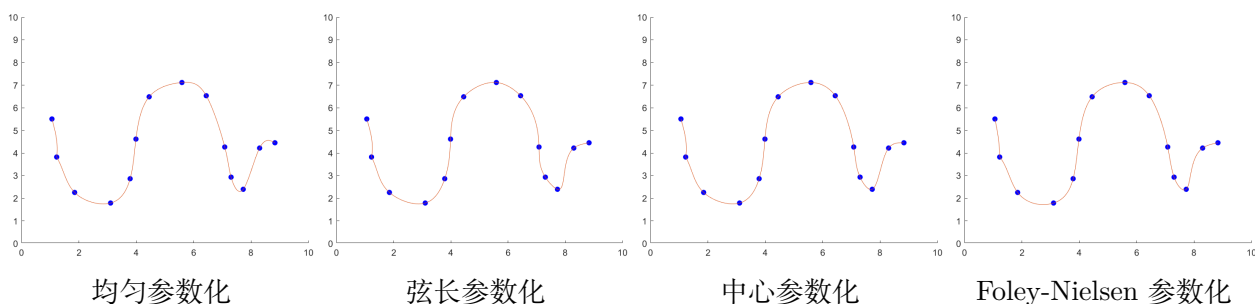


图 7: 三次自然样条函数在不同参数化下的拟合效果

4.6 结果对比

综合上面的实验结果可以看出参数化方法中 Foley-Nielsen 参数化得到的曲线拟合效果最好，均匀参数化效果很一般。而四种曲线拟合的方法中，RBF 神经网络和三次样条函数拟合出的曲线是最接近预期目标的，Gauss 基函数和最小二乘法次之，多项式插值法得到的曲线在边界处的结果不尽如人意。

在图 2 中我们可以最直观地感受到四种拟合方法在不同的参数化下得到的拟合曲线的效果。由于多项式插值法处理均匀结点时在边界部分会出现 Runge 现象，可以看到多项式曲线在周围处波动幅度巨大，显然不是理想的拟合曲线，这种情况随着参数化越来越近似于弦长而得到减弱，但相较于其余三种拟合曲线，多项式曲线仍然不能很好地达到处理预期效果。而剩下三种拟合曲线地结果只有不明显的一些波动，在 Foley-Nielsen 参数化下的结果基本上就是目标曲线。

在图 3 中我们可以清楚地看到四种不同地参数化方法对于曲线弯折处地影响^[2]，均匀参数化得到的曲线在弯曲处比较“锐利”，并且在凹陷处有向中线靠拢的倾向；弦长参数化得到的曲线在第一个弯曲处明显不符合预期，并且整条曲线有着以凹陷处为中线，向两侧凸出的趋势；中心参数化和 Foley-Nielsen

参数化对于曲线的弯曲处理地比较平缓，没有出现明显地“缠绕”现象，只是在凹陷处对于曲线的走向有着些许差异。

图 4 和图 5 都是一种最优化的拟合思想，所以将它们的结果放在一起比对。可以看出 RBF 神经网络拟合出来的曲线要更加的平滑，也更加符合点列的走向；而最小二乘法拟合出的曲线在起止位置波动还是有些明显，推测原因和拟合中具体的一些参数有关。

图 6 和图 7 是多项式基函数在四种参数化下对于有大幅波动的点列的拟合效果，可以看出对于一般的多项式插值，Runge 现象还是很严重的，在曲线的两端出现了明显的过拟合情况，不过随着参数化逐渐近似弦长这种情况也得到了缓解，但综合来看仍然不如其他参数化方法的拟合效果好。而三次自然样条函数得到的结果是十分优秀的，在四种不同的参数化方法下并没有明显的不同，拟合出的曲线非常光滑且没有多余的波动，是一种十分稳定和精确的拟合方法。

五、结论

从参数化方面来说，实验结果确实证明了参数化越接近与弦长，拟合的曲线就越精确这一结论。综合实验结果来看，Foley-Nielsen 参数化在处理有着大幅度弯曲的点列时有着优于其它参数化方法的效果，这是因为 Nielsen 度量本身所具有的仿射不变性，它会最大程度的保留点列在弯曲处的信息，并将其还原到拟合曲线上。

从曲线拟合方法来讲，通过最优化过程来选择拟合曲线所得结果一般而言要优于函数插值，这是因为最优化的方法会将整体误差尽可能地平均在整条曲线上，而插值法则会要求在结点处误差严格为零，这就可能导致在其他位置的误差会增加。结合图 4 的结果来说明，在实验中我们要求最小二乘拟合的多项式次数 m 要小于插值多项式次数 n ，但是 m 的选择将会影响最终得到的拟合曲线效果： m 过小则损失的信息太多，拟合出的曲线没有意义； m 过大则会向插值多项式一样出现 Runge 现象。实际上本次实验中 m 的取值为 $\lceil \frac{n}{3} \rceil + 1$ ，所以在边界处就出现了一些明显的波动。而 RBF 神经网络因为选取了充分多的神经元个数，所以拟合曲线基本上没有明显的波动。样条函数由于其独特的性质，不能和一般的多项式插值相提并论，其对于散点列的拟合效果应该是这几种方法中最好的，因此也被广泛应用与图像处理中。

六、问题

1. 在多项式基函数中，我们也可以使用 Bernstein 多项式进行拟合，得到的结果就是 Bezier 曲线，这也是一种性质非常优秀的拟合曲线，被广泛运用于处理实际问题。但是 Bezier 曲线似乎只能进行均匀的参数化，不能满足本次实验的对照方法，所以留待以后实现。

2. 在最小二乘拟合中，为了达到理想的拟合效果，我们除了可以调整拟合多项式的最高次数来达到一个折中，也可以为每个点增加一个权重来让我们需要的某些点达到更好的拟合效果。也就是说，对于整个的最小二乘增加一个正则化约束，这就是岭回归最小二乘法。

参考文献

- [1] Floater, Michael & Surazhsky, Tatiana. (2006). Parameterization for Curve Interpolation. Studies in Computational Mathematics. 12. 39-54.
- [2] E.T.Y. Lee, Choosing nodes in parametric curve interpolation, Computer-Aided Design, Volume 21, Issue 6, 1989, Pages 363-370.