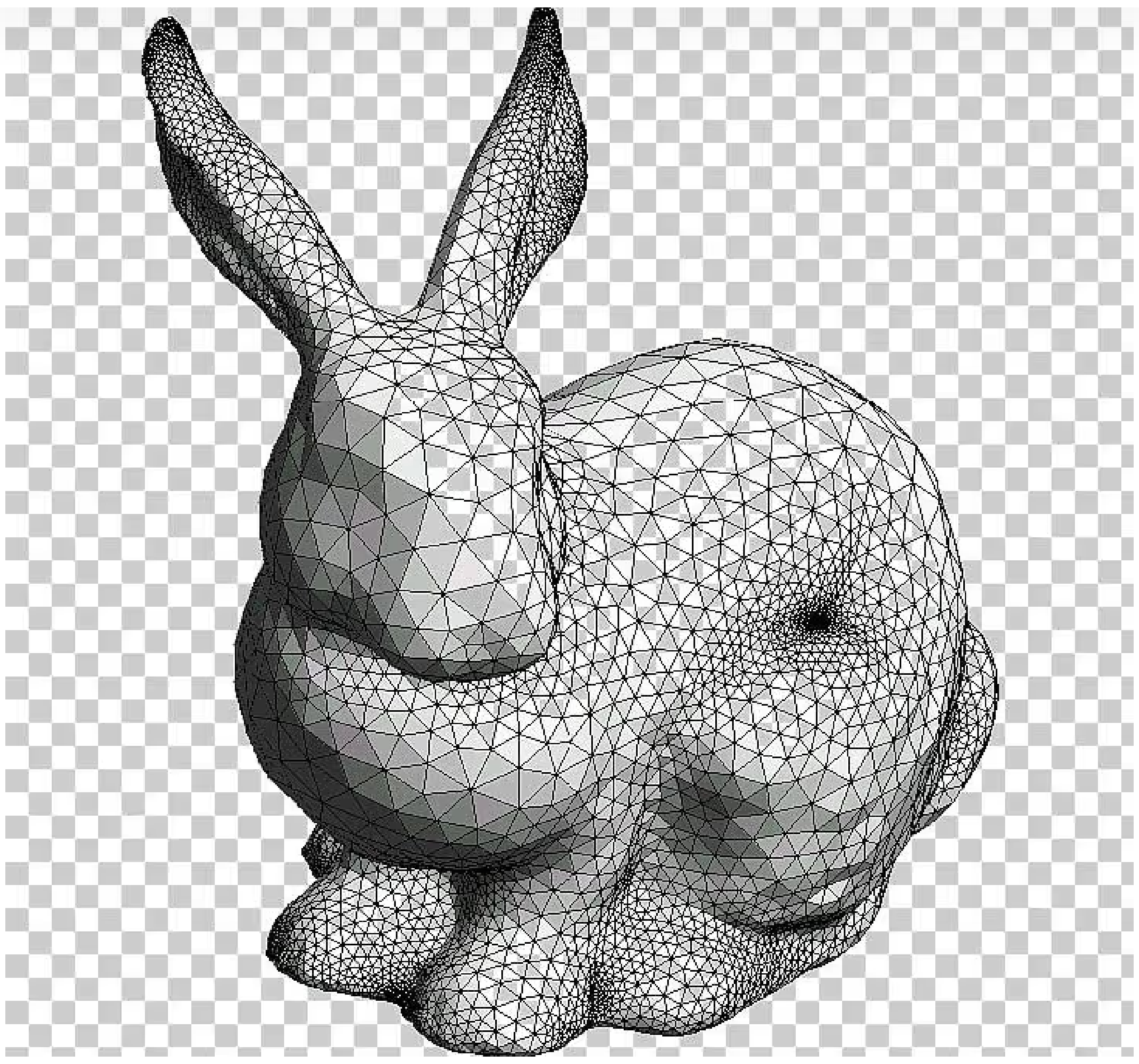


从零开始几何处理：函数拟合

启思
图形学，视觉，机器学习

前言

之前学完了 GAMES102，现在打算以 101 类似的形式，以作业为单位整理一下涉及的知识。

不过明显感觉到 102 涉及的知识更杂，包括函数拟合、离散微分几何，以及后面的三维重建等知识，后面有机会的话可以单独写个总结（挖个大坑）捋一遍。

同样尽我所能写的简单易懂点，如有错误欢迎指出……

引入

函数拟合这个词并不陌生，高中时我们就学过最小二乘法线性拟合二维点，这就是一个函数拟合的例子。

函数拟合，顾名思义，就是有一对数据点，我们要根据这些数据点找到一个函数，来描述数据集。

在实际生活中，许多应用也满足这一条件，比如三维重建就是用激光雷达等扫描仪采集数据后，再找到一个表面（函数）来拟合这些数据点。

函数拟合通常分为两大类：插值问题、逼近问题。插值问题限定函数必须经过那些数据点，逼近问题则不一定。

下面来分别介绍一下。

插值问题

首先来定义一下插值问题：已知离散的数据点集，构建（估计、寻找）一个新的数据点集。

换句话说，给出 m 个点 $(x_0, y_0), \dots, (x_{m-1}, y_{m-1})$，求一个函数 $f(x)$ 作为点集的估计，使得满足 $f(x_i) = y_i, 0 \le i < n$。我们可以通过函数 f 来获取新的点集。

函数空间

函数 f 是什么？我们并不知道。它有可能是多项式函数，有可能是三角函数的组合，有可能根本不存在解析式。我们的目标是找到一个比较好的估计。

一般来说，为了更方便地寻找函数 f，我们需要规定一个**函数空间**，在这个空间内寻找 f。和线性代数里的基向量类似，函数空间的若干基底称之为**基函数**，该函数空间内的任意函数都是基函数的线性组合。

一个栗子：多项式

一个最常用的函数空间就是多项式函数空间，基函数为 $1, x, x^2, \dots$

假定函数 f 是一个 n-1 次多项式，那么 f 可以写为

$$f(x) = a_0 + a_1 * x + \dots + a_{n-1} * x^{n-1}$$

其中

$$\mathbf{Y}=\begin{bmatrix}y_0\\y_1\\\vdots\\y_{m-1}\end{bmatrix},\mathbf{G}=\begin{bmatrix}1&x_0&x_0^2&\cdots&x_0^{n-1}\\1&x_1&x_1^2&\cdots&x_1^{n-1}\\\vdots&\vdots&\vdots&\ddots&\vdots\\1&x_{m-1}&x_{m-1}^2&\cdots&x_{m-1}^{n-1}\end{bmatrix},\mathbf{A}=\begin{bmatrix}a_0\\a_1\\\vdots\\a_{n-1}\end{bmatrix}$$

注意到 G 是一个 $m\times n$ 的矩阵。假设方程之间线性无关，由线性代数知识可知，求解线性方程组需要比较未知数个数 n 和方程个数 m 的大小：

1. 若 m > n，即插值问题中给定点数比未知数要多，则无解。
2. 若 m = n，则有唯一解。
3. 若 m < n，则有无穷解。

值得一提的是，对于多项式函数的插值，当 m = n 时，存在一种快速求解 f 的方式，称为**拉格朗日插值法**，如下

$$f(x)=\sum_{0\le i<n}\frac{\prod_{j\ne i}(x_j-x)}{\prod_{j\ne i}(x_j-x_i)}y_i$$

即构造了 $f(x)=\sum_i y_il_i(x)$，使得 $l_i(x_i)=1,l_i(x_j)=0,j\ne i$。所以多项式插值可以在 $O(n^2)$ 下完成。

任意函数空间

事实上，对于任意函数空间，假设基函数为 $g_i(x)$，$f(x)$ 可以写作

$$f(x)=a_0*g_0(x)+a_1*g_1(x)+\ldots+a_{n-1}*g_{n-1}(x)$$

类似于上述，插值问题即求解 $\mathbf{Y}=\mathbf{GA}$ 中的 $\mathbf{G}$，其中 $\mathbf{Y},\mathbf{A}$ 与上述类似，而

$$\mathbf{G}=\begin{bmatrix}g_0(x_0)&g_1(x_0)&\cdots&g_{n-1}(x_0)\\g_0(x_1)&g_1(x_1)&\cdots&g_{n-1}(x_1)\\\vdots&\vdots&\ddots&\vdots\\g_0(x_{m-1})&g_1(x_{m-1})&\cdots&g_{n-1}(x_{m-1})\end{bmatrix}$$

解的情况与上述讨论类似，不同的是，非多项式函数空间可能并不存在像拉格朗日插值一样方便的求解方法。

另外，在实际操作中，如果出现 m<n 的情况，即方程个数（给定数据点个数）小于未知量个数时，可以使用某些方法添加约束（添加方程/数据点）使得满足 m=n。例如取平均等。

多项式函数空间是万能的吗？

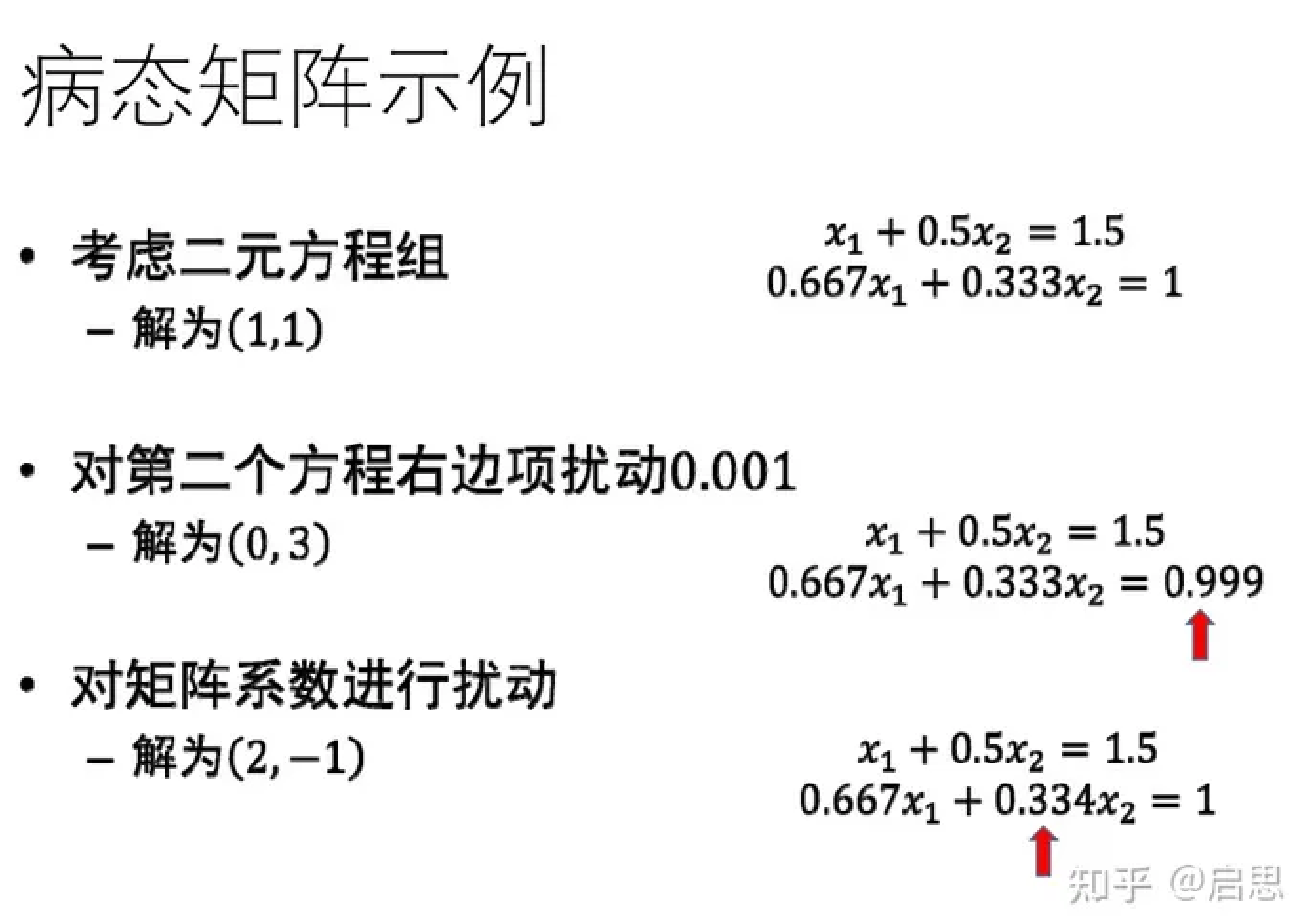
首先看一个问题：既然我们有了拉格朗日插值法，可以快速插出任意的多项式，那么其他函数空间还有存在的必要吗？

答案当然是有必要的。事实上，多项式函数空间存在一些问题。

注：本小节中假设 n=m，即插值点数（方程数）等于基函数个数

我们已经知道，插值问题等价于一个线性方程组求解问题，而线性方程组的求解问题又等价于矩阵求逆问题。

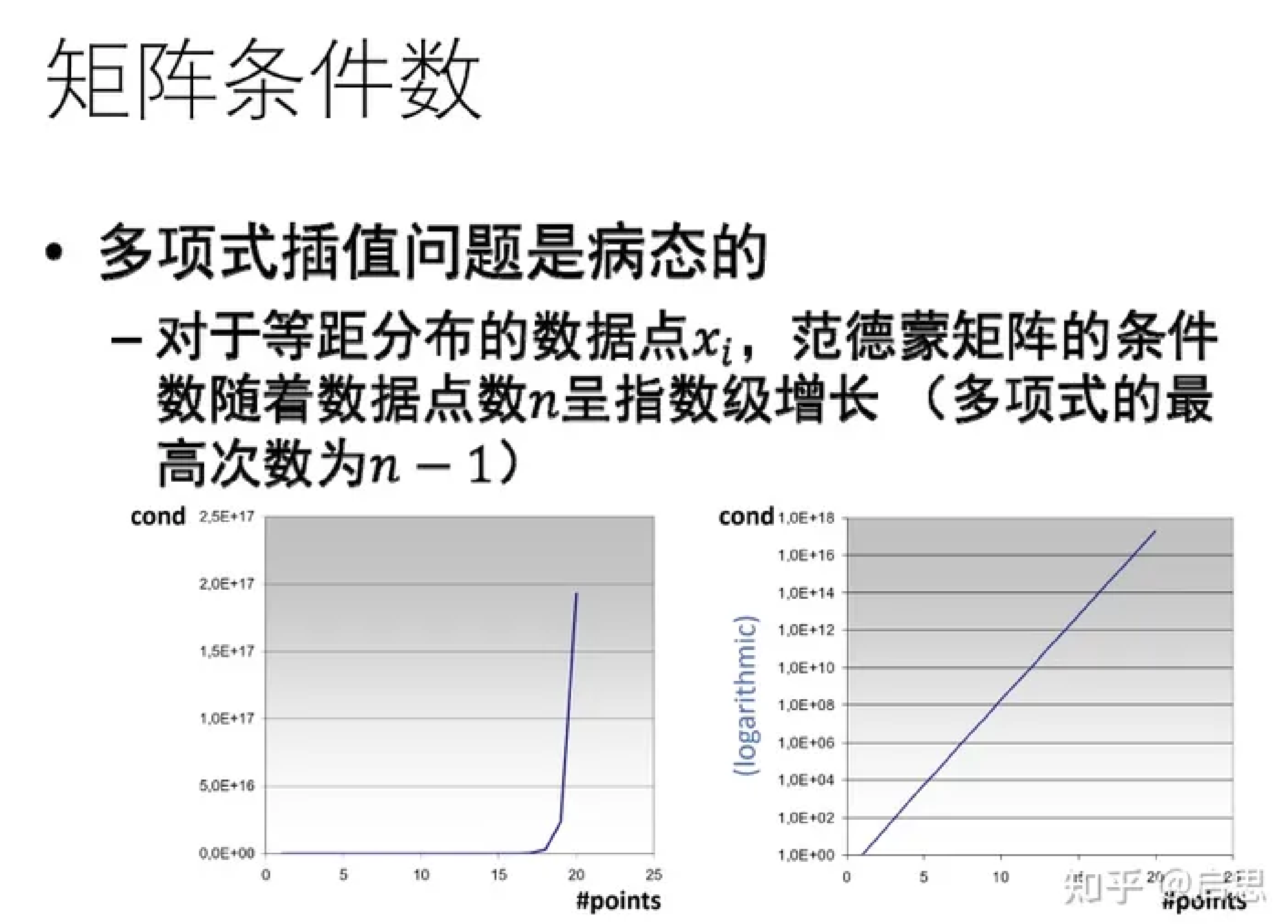
然而在实际操作中，逆矩阵可能遭遇数值精度等问题，导致**难以求解**。例如下图所示的病态矩阵，对其做很小的扰动便导致了解的剧变



可以把每个方程看做一个超平面，求解方程组变成若干超平面的求交问题，如果超平面接近于平

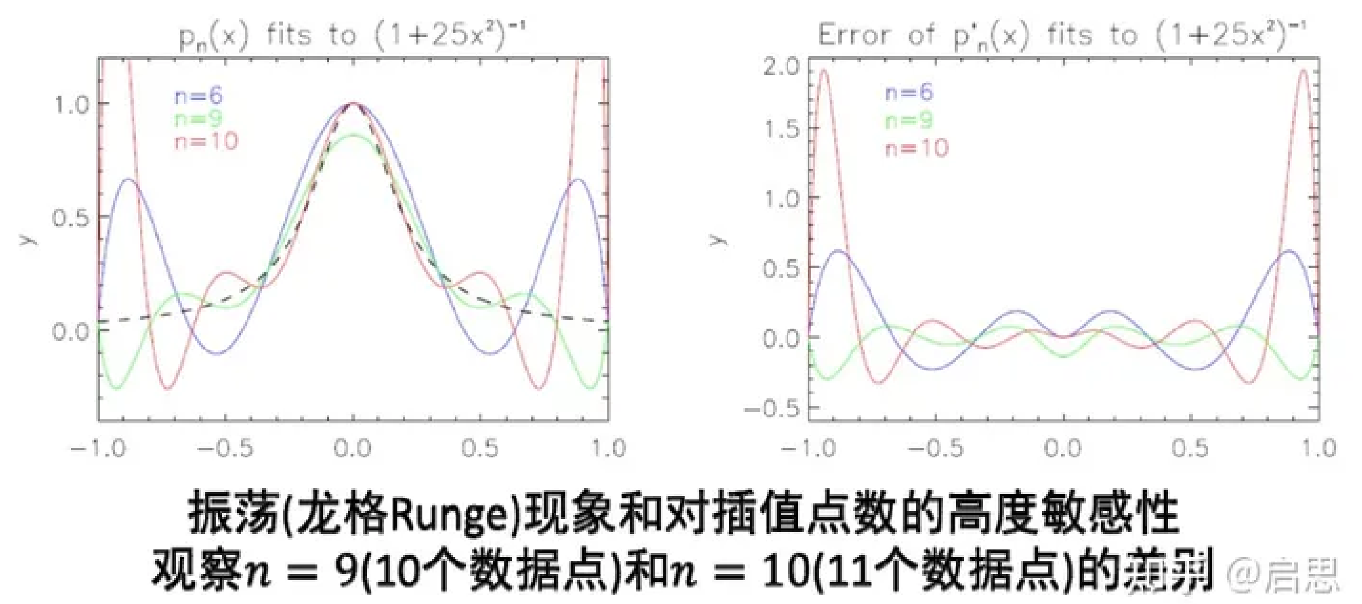
在插值问题中，系数矩阵只与**基函数**和**给定数据点**有关。如果基函数选的不好，那么可能会导致病态问题。

多项式插值的系数矩阵是病态的。因为多项式插值的系数矩阵，即范德蒙德矩阵，的条件数呈指数增加



另外，我们知道任意函数都可以由无穷多的多项式函数任意逼近，但是在实际应用时，我们不可能使用无穷多个基函数。我们希望，在使用有限基函数时，随着基函数（插值点数）的增加，插值函数逐渐逼近于目标函数。

然而，**多项式插值具有强烈的振荡现象**，逼近的过程并不稳定。这使得我们对插值函数的可信度有所质疑



综上所述，多项式函数空间存在其本身的缺点：病态问题、振荡问题。所以我们需要更好的函数空间。

总结

插值问题，即给定若干离散的数据点，构建出新的数据点的问题。

插值问题可以认为是一种函数拟合问题，我们需要在特定的函数空间中寻找，以找到一个函数来描述数据。

在特定函数空间中寻找函数，本质上是一个线性方程组求解问题。

在实际应用中，一个好的函数空间可以让问题更容易解决。

逼近问题

在实验学科中，为了得到离散的数据点，常常使用采样等方式。通常情况下，这不可避免地带来噪声、异常值等问题，也就是说不是所有的数据点都 100% 可信。另一方面，如果必须完美拟合所有数据点，这带来的代价可能是巨大的（例如过高的参数量、计算量）。

舍弃完美拟合所有的数据点，而是寻找一个逼近（近似）函数，就是逼近问题。

更准确的，逼近问题是，给出 m 个点 $(x_0, y_0), \dots, (x_{m-1}, y_{m-1})$，求一个函数 $f(x)$ 使其为目标函数的一种最佳逼近（近似）。

类似于插值问题，我们仍然需要一个函数空间。

为了求解此问题，我们需要先**定义“什么是最佳逼近（近似）”**。事实上这并不唯一。

一个栗子：最小二乘逼近

一个广为人知的逼近算法是最小二乘法，它定义最佳逼近为：