中国科学技术大学
University of Science and Technology of China

GAMES 102在线课程

# 几何建模与处理基础

刘利刚

中国科学技术大学

中国科学技术大学
University of Science and Technology of China

GAMES 102在线课程：几何建模与处理基础

# 数据拟合(2)

# 回顾

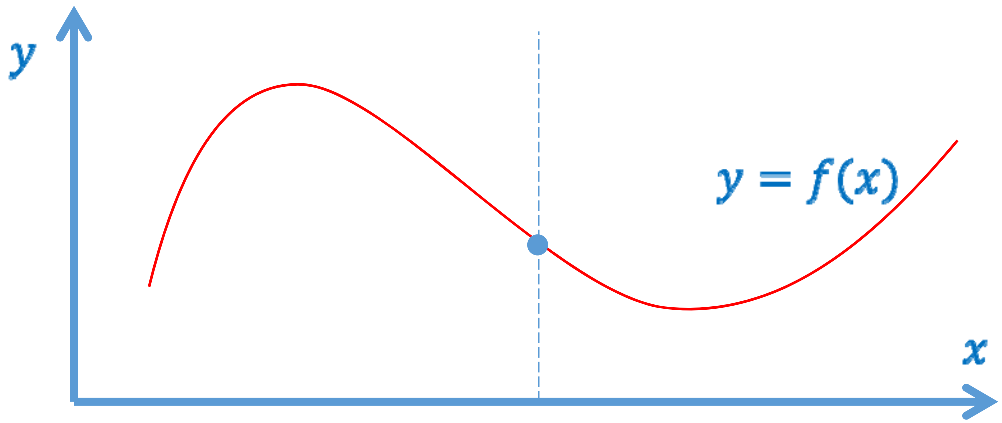
- 计算机图形学基本内容
- 函数、映射、变换…
- 数据拟合
  - 找哪个?
  - 到哪找?
  - 怎么找?
- 作业1：尝试、思考、困惑、讨论、理解…
- 补充：“GAMES 102学习材料(1)”

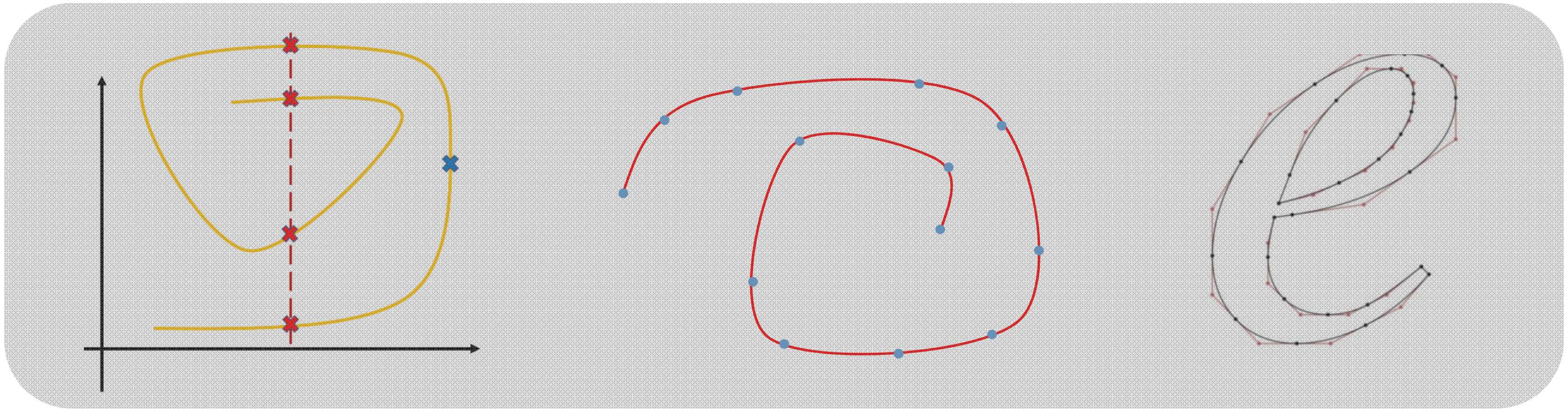
# 假定： 函数形式

- 假定：仅函数形式，一般曲线（非函数形式）后面再学习

$f: R^1 \to R^1$

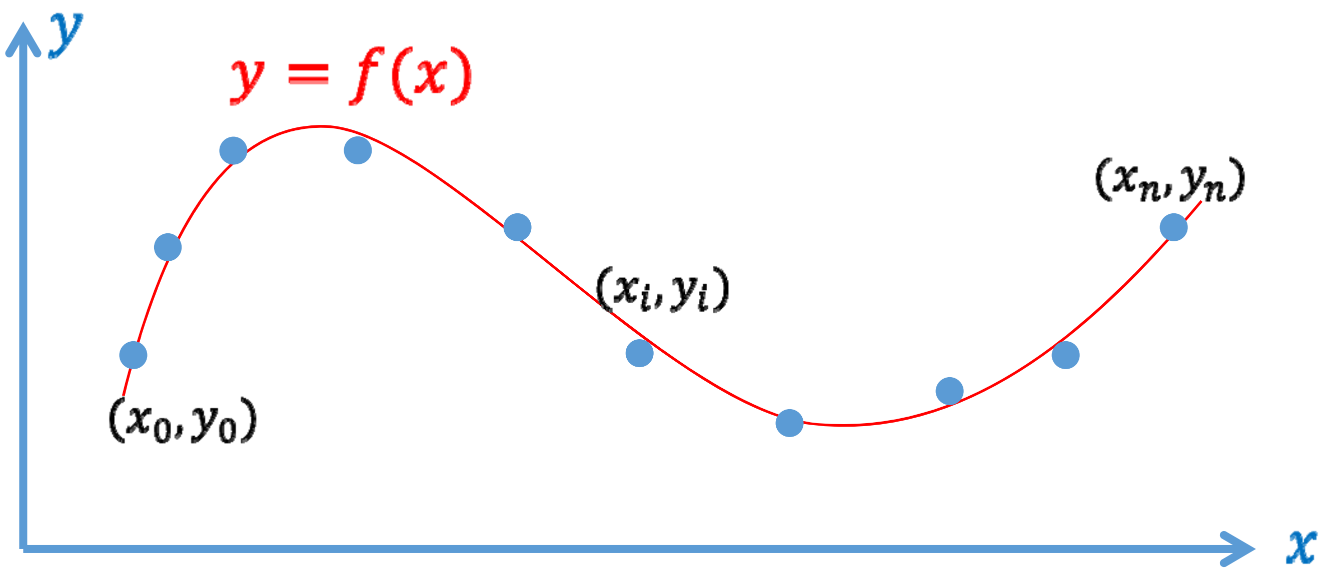
$y = f(x)$



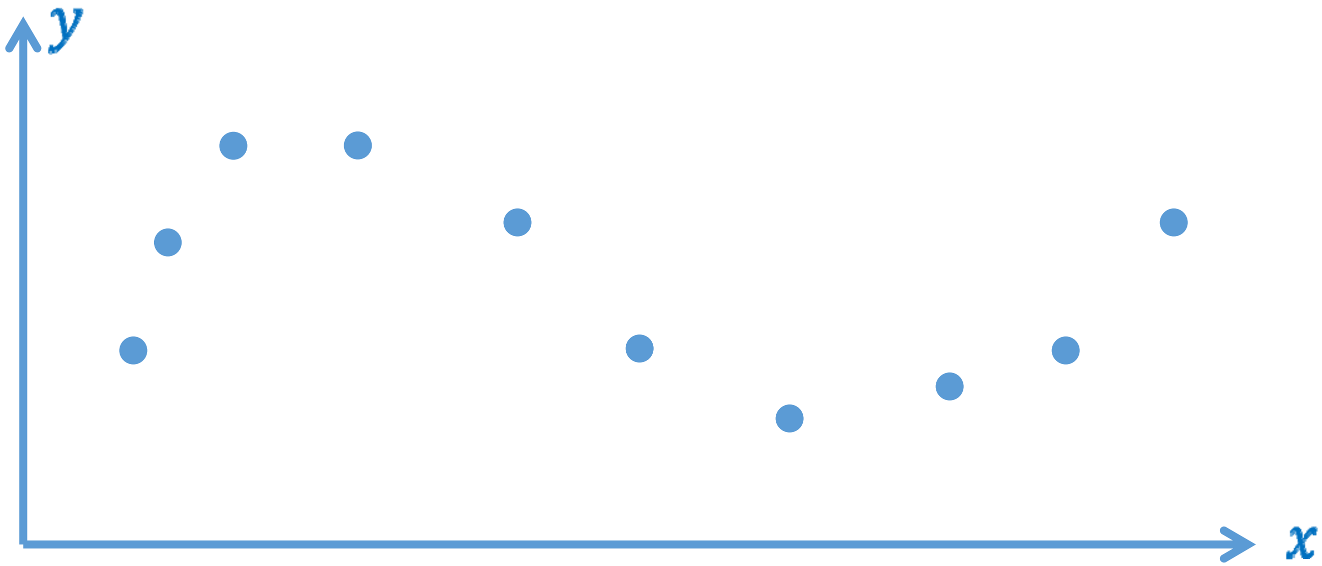
# 函数拟合问题

- 输入：一些观察（采样）的数据点$\{x_i, y_i\}_{i=0}^{n}$
- 输出：拟合数据点的函数$y = f(x)$，并用于**预测**



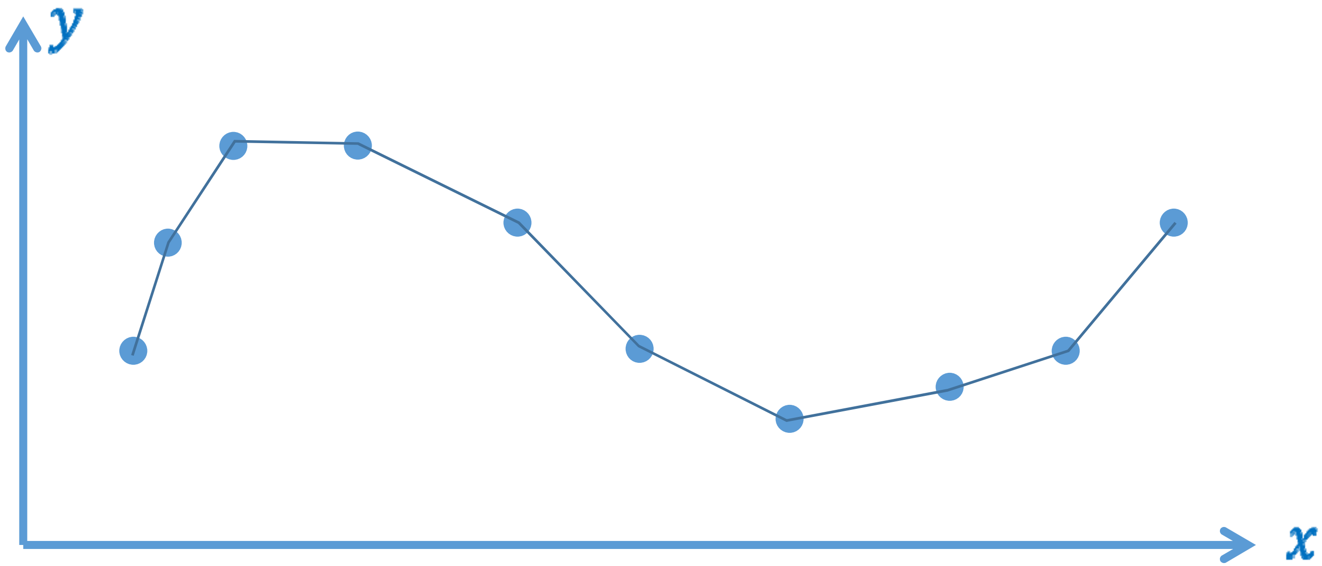

这种拟合函数有多少个？

# 拟合函数的“好坏”



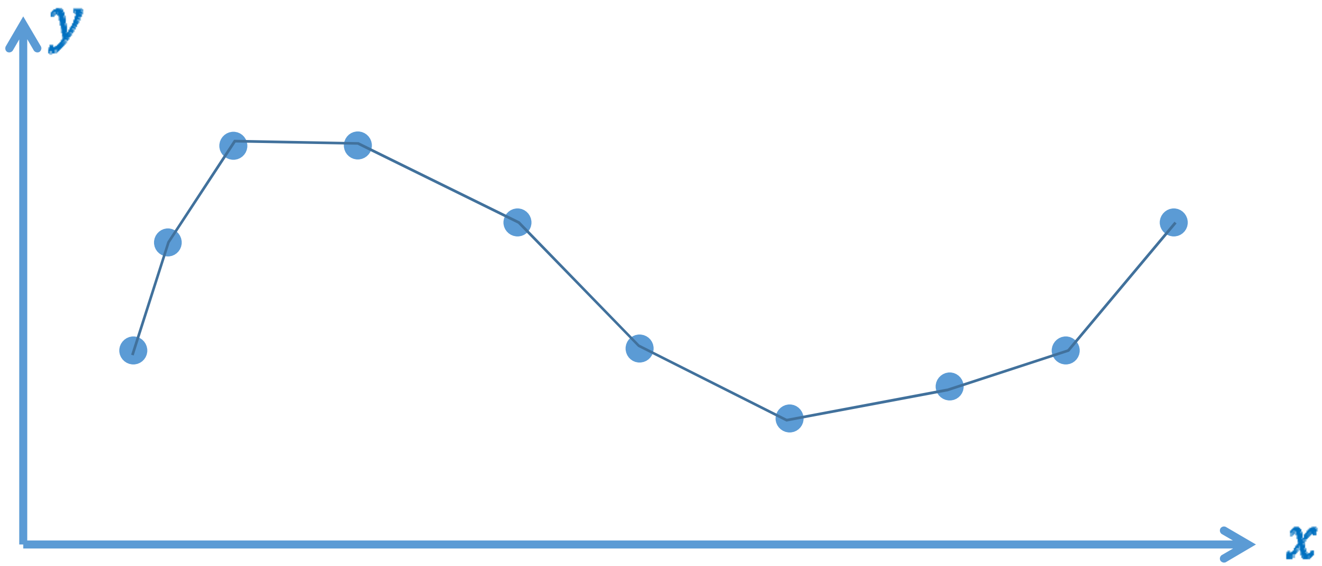
# 拟合函数的“好坏”

- **分段线性插值函数$y = f_1(x)$**
  - **数据误差为0**



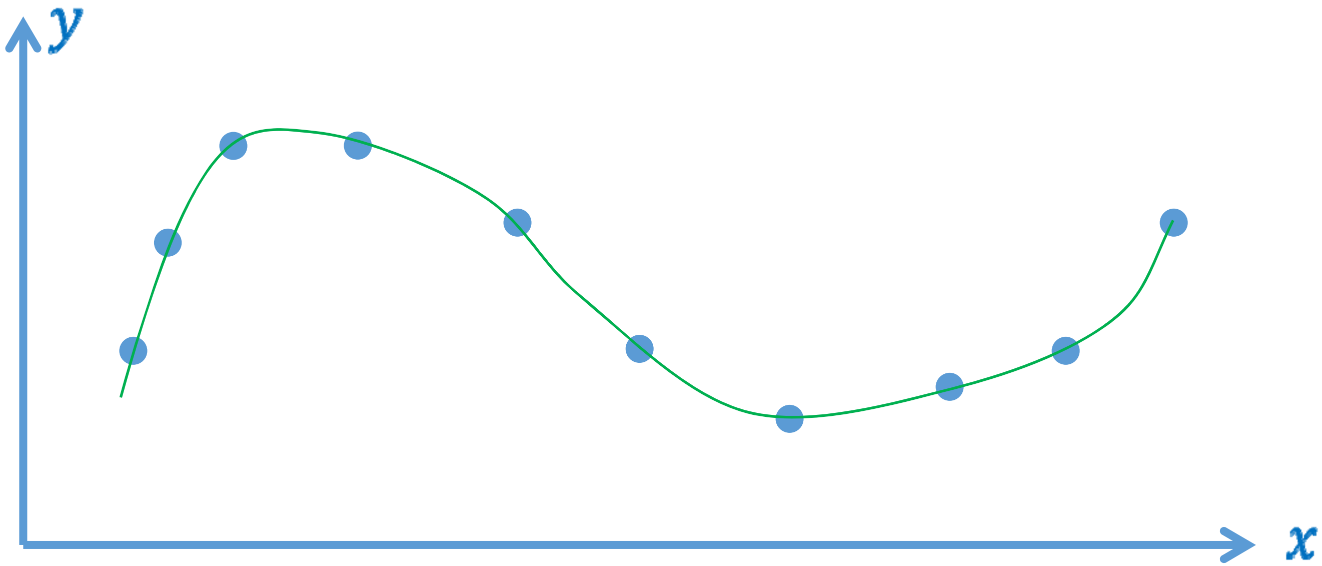
# 拟合函数的“好坏”

- 分段线性插值函数$y = f_1(x)$
  - 数据误差为0
  - 函数性质不够好：只有$C^0$连续，不光滑（数值计算）



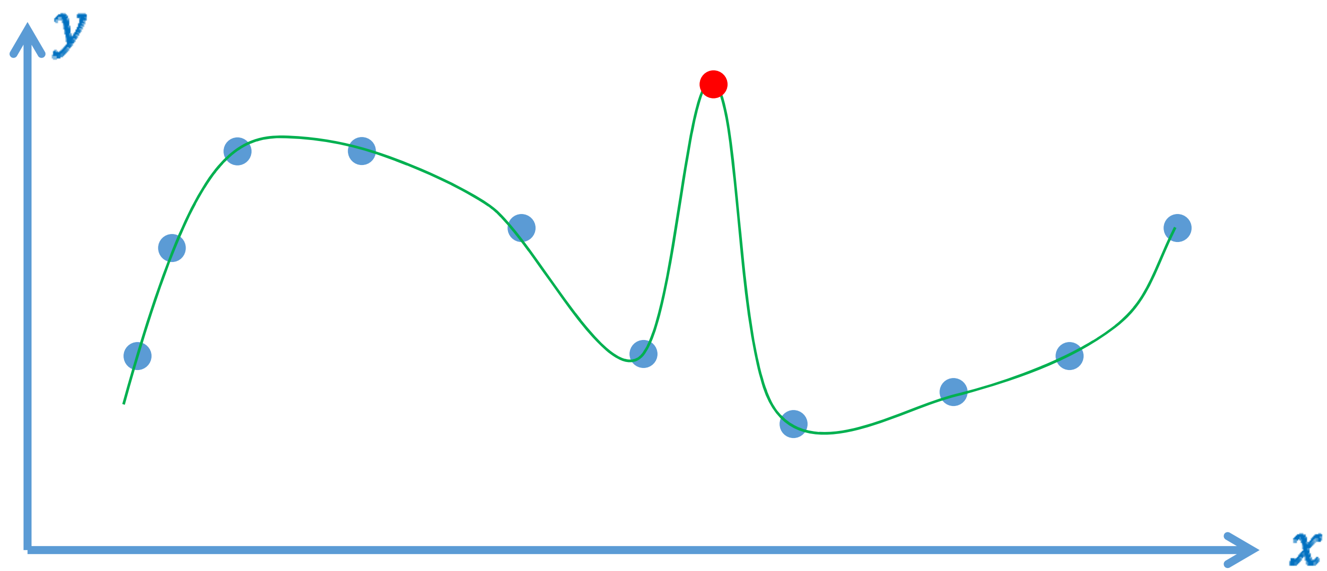
# 拟合函数的“好坏”

- 光滑插值函数$y = f_2(x)$
  - 数据误差为0



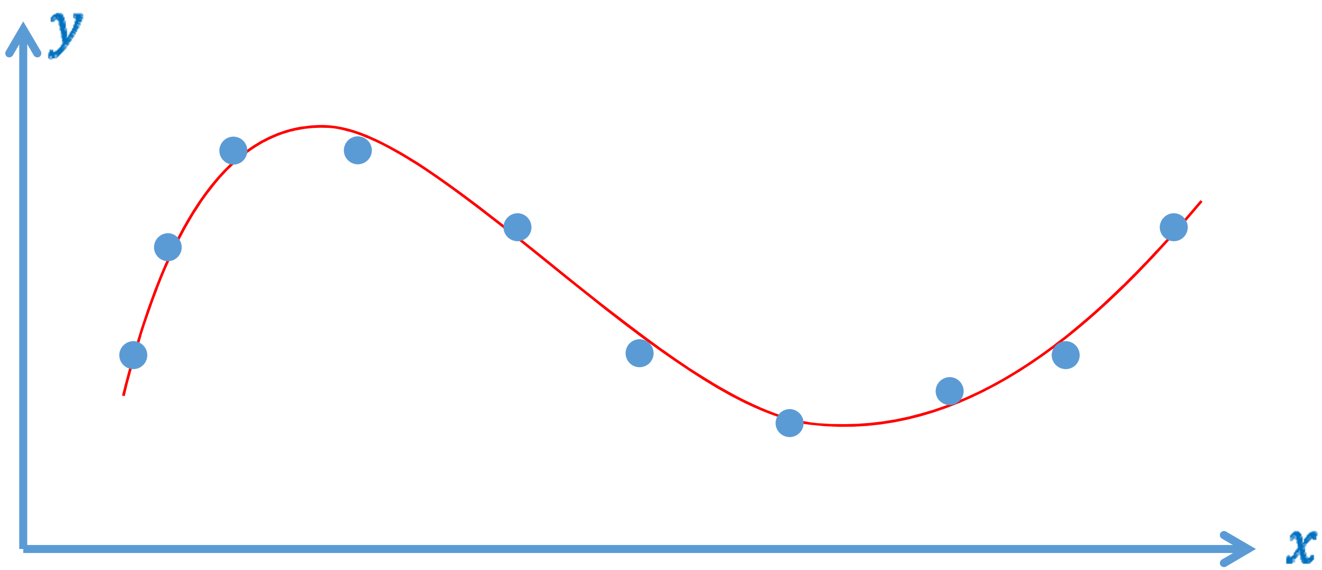
# 拟合函数的“好坏”

- **光滑插值函数$y = f_2(x)$**
  - **数据误差为0**
  - **可能被“差数据”（噪声、outliers）带歪，导致函数性质不好、预测不可靠**



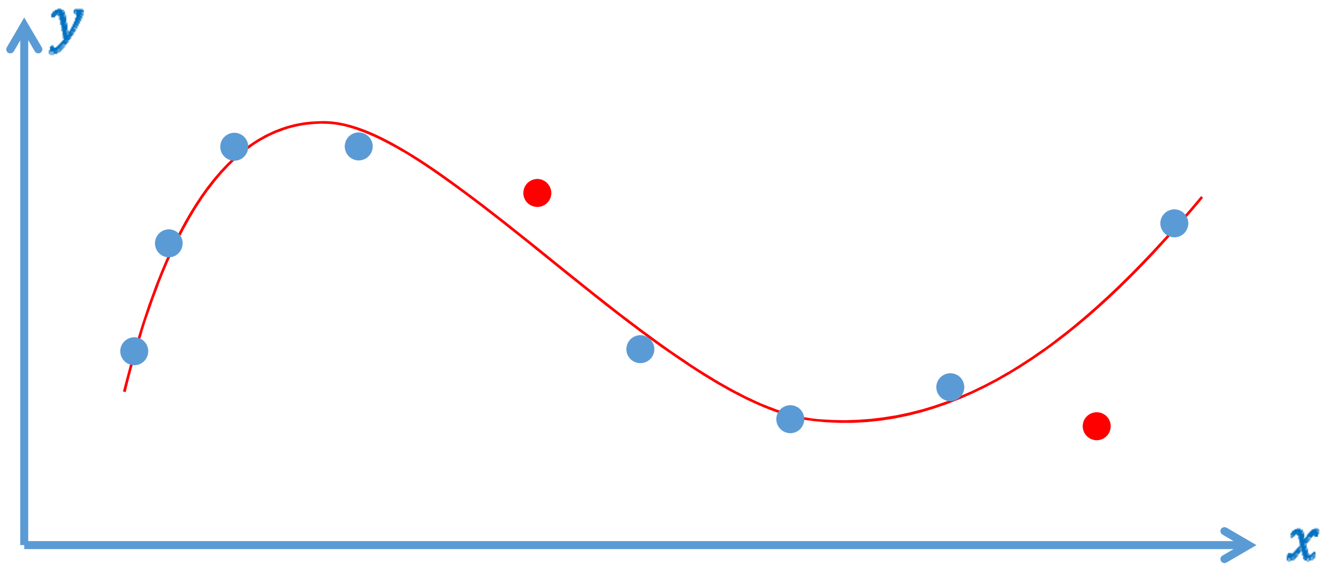
# 拟合函数的“好坏”

- 逼近拟合函数$y = f_3(x)$
  - 数据误差不为0，但足够小



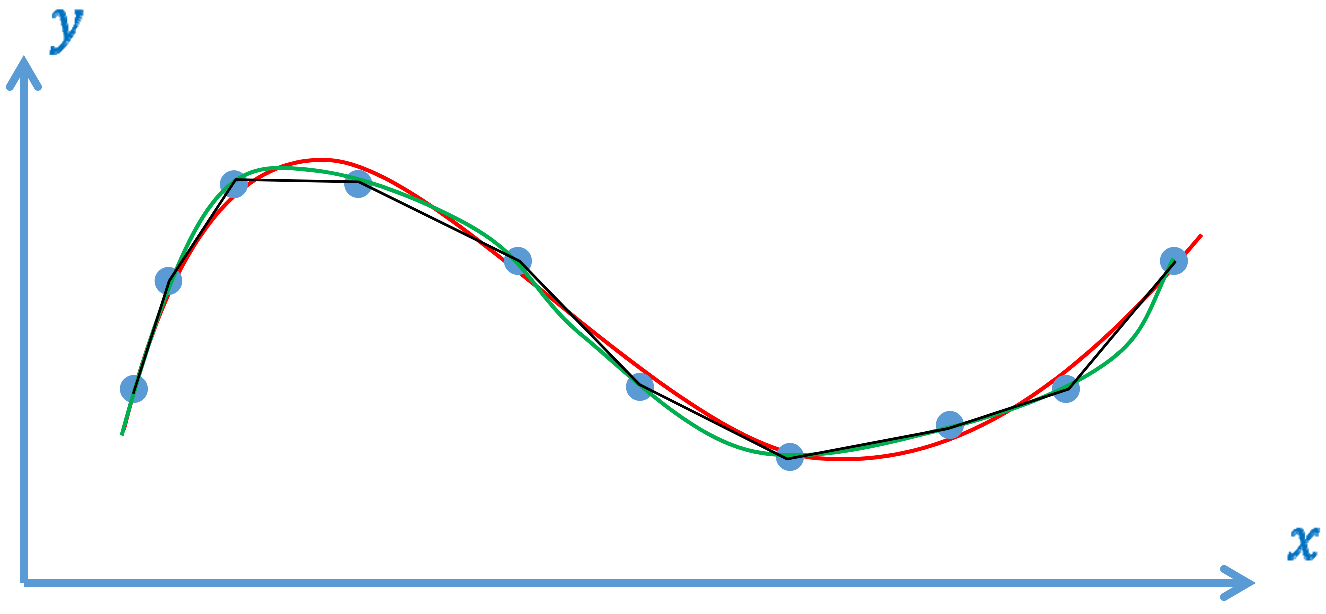
# 拟合函数的“好坏”

- **逼近拟合函数$y = f_3(x)$**
  - **数据误差不为0，但足够小**



# 拟合函数的“好坏”

- 分段线性函数$f = f_1(x)$
- 光滑插值函数$f = f_2(x)$
- 逼近拟合函数$f = f_3(x)$



# 求拟合函数：应用驱动

- 大部分的实际应用问题
  - 可建模为：找一个映射/变换/函数
  - 输入不一样、变量不一样、维数不一样
- 三步曲方法论：
  - 到哪找？
    - 确定某个函数集合/空间
  - 找哪个？
    - 度量哪个函数是好的/“最好”的
  - 怎么找？
    - 求解或优化：不同的优化方法与技巧，既要快、又要好…

# 数据拟合的方法论

- 到哪找?
  - 确定函数的表达形式（函数集、空间）$L = span\{b_0(x), \ldots, b_n(x)\}$
  - 待定基函数的组合系数（求解变量）$f_\lambda(x) = \sum_{k=0}^{n} \lambda_i b_i(x)$
- 找哪个?
  - 优化模型（最小化问题）
    - 能量项 = 误差项 + 正则项
  - 统计模型、规划模型…
- 怎么找?
  - 求解误差函数的驻点（导数为0之处）
  - 转化为系数的方程组
    - 如果是欠定的（有无穷多解），则修正模型
      - 改进/增加各种正则项：Lasso、岭回归、稀疏正则项…
      - 返回修改模型

$f$由待定系数$\lambda = \begin{pmatrix} \lambda_1 \\ \ldots \\ \lambda_n \end{pmatrix}$确定

# 1. 多项式插值

# 多项式插值定理

**定理**：若 $x_i$ 两两不同，则对任意给定的 $y_i$，存在唯一的次数至多是 $n$ 次的多项式 $p_n$，使得 $p_n(x_i) = y_i\ i = 0, \cdots, n$。

证明：在幂基 $\{1, x, \cdots, x^n\}$ 下待定多项式 $p$ 的形式为：

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

由插值条件 $p(x_i) = y_i, i = 0, \cdots, n$，得到如下方程组：

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_o^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ 1 & x_2 & x_2^2 & \dots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

如果基函数选取不一样，方程组的系数矩阵不同

系数矩阵为 **Vandermonde** 矩阵，其行列式非零，因此方程组有唯一解。

# 技巧1：构造插值问题的通用解

- **构造插值问题的通用解**
  - 给定$n+1$个点$\{(x_0,y_0),\ldots,(x_n,y_n)\}$，寻找一组次数为$n$的多项式基函数$l_i$使得

$$l_i(x_j)=\begin{cases}1, 若 i=j\\ 0, 若 i\neq j\end{cases}$$

  - 插值问题的解为：

$$P(x)=y_0l_0(x)+y_1l_1(x)+\cdots+y_nl_n(x)=\sum_{i=0}^{n}y_il_i(x)$$

# 一般形式

- 怎么计算多项式$l_i(x)$?
  - $n$阶多项式，且有以下$n$个根
$$x_0, x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n$$
  - 故可表示为

$$l_i(x) \\ = C_i(x - x_0)(x - x_1) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n) \\ = C_i \prod_{j \neq i} (x - x_j)$$

  - 由$l_i(x_i) = 1$可得
$$1 = C_i \prod_{j \neq i} (x_i - x_j) \Rightarrow C_i = \frac{1}{\prod_{j \neq i} (x_i - x_j)}$$

# 技巧1：构造插值问题的通用解

- **最终多项式基函数为**

$$l_i(x) = \frac{\prod_{j\neq i}(x - x_j)}{\prod_{j\neq i}(x_i - x_j)}$$

- **多项式$l_i(x)$被称为拉格朗日多项式**

# 技巧2：更方便的求解表达

- Newton插值：具有相同“导数”（差商）的多项式构造（*n*阶Taylor展开）

**定义**：

一阶差商：

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

$k$ 阶差商：

设 $\{x_0, x_1, \cdots, x_k\}$ 互不相同，$f(x)$ 关于 $\{x_0, x_1, \cdots, x_k\}$ 的 $k$ 阶差商为：

$$f[x_0, x_1, \cdots, x_k] = \frac{f[x_1, \cdots, x_k] - f[x_0, x_1, \cdots, x_{k-1}]}{x_k - x_0}$$

所以 **Newton 插值多项式**表示为：

$$N_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \cdots, x_n](x - x_0)\cdots(x - x_{n-1})$$

# 多项式插值存在的问题

- 系统矩阵稠密
- 依赖于基函数选取，矩阵可能病态，导致难于求解（求逆）

*Thanks to Renjie Chen*

# 病态矩阵示例

- 考虑二元方程组
  - 解为(1,1)

$$\begin{aligned} x_1 + 0.5x_2 &= 1.5 \\ 0.667x_1 + 0.333x_2 &= 1 \end{aligned}$$

- 对第二个方程右边项扰动0.001
  - 解为(0, 3)

$$\begin{aligned} x_1 + 0.5x_2 &= 1.5 \\ 0.667x_1 + 0.333x_2 &= 0.999 \end{aligned}$$

- 对矩阵系数进行扰动
  - 解为(2, −1)

$$\begin{aligned} x_1 + 0.5x_2 &= 1.5 \\ 0.667x_1 + 0.334x_2 &= 1 \end{aligned}$$

# 病态问题

- 输入数据的细微变化导致输出(解)的剧烈变化
- 将线性方程看成直线（超平面）
  - 当系统病态时，直线变为近似平行
  - 求解（即直线求交）变得困难、不精确

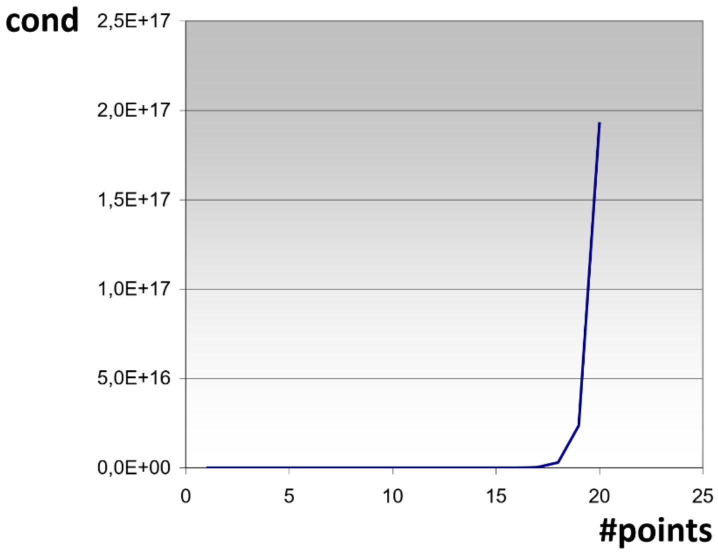
# 矩阵条件数

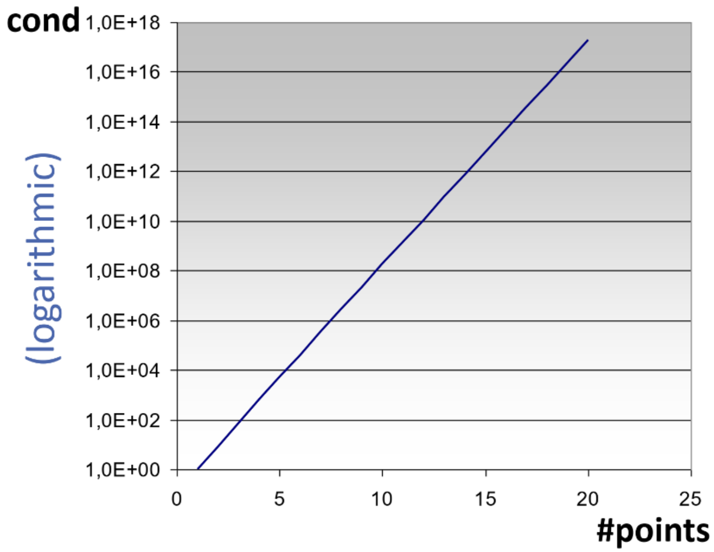
$$\kappa_2(A) = \frac{\max\limits_{x \neq 0} \frac{\|Ax\|}{\|x\|}}{\min\limits_{x \neq 0} \frac{\|Ax\|}{\|x\|}}$$

- 等于最大特征值和最小特征值之间比例
- 条件数大意味着基元之间有太多相关性

# 矩阵条件数

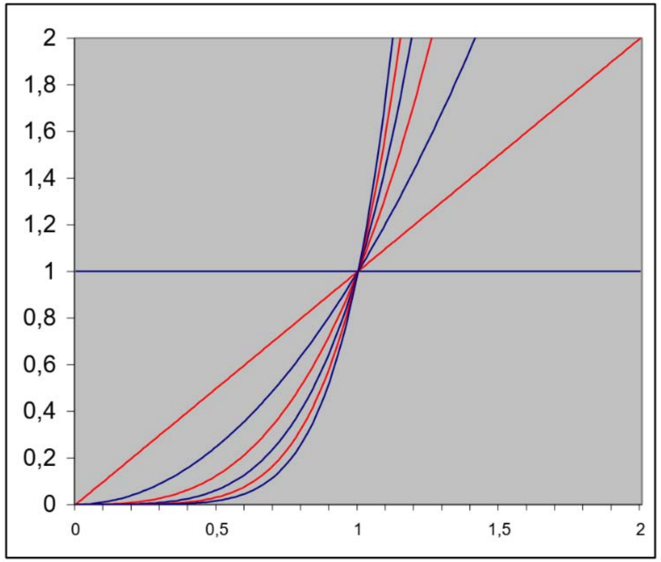
- 多项式插值问题是病态的
  - 对于等距分布的数据点$x_i$，范德蒙矩阵的条件数随着数据点数$n$呈指数级增长 （多项式的最高次数为$n-1$）



# 为什么？

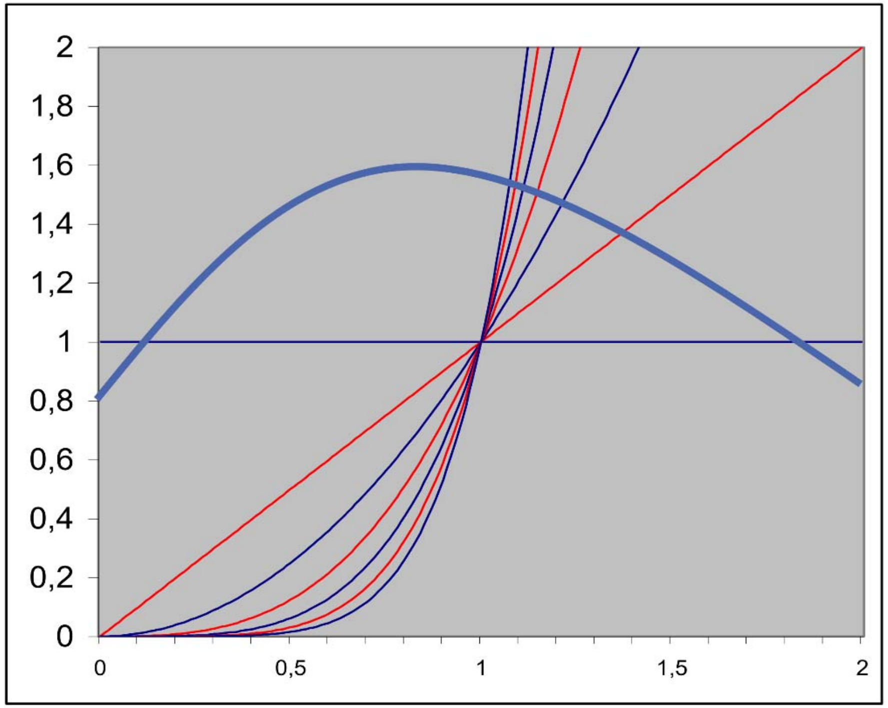
- 幂（单项式）函数基
  - 幂函数之间差别随着次数增加而减小
  - 不同幂函数之间唯一差别为增长速度($x^i$比$x^{i-1}$增长快)




幂(单项式) 函数

# 函数互相抵消

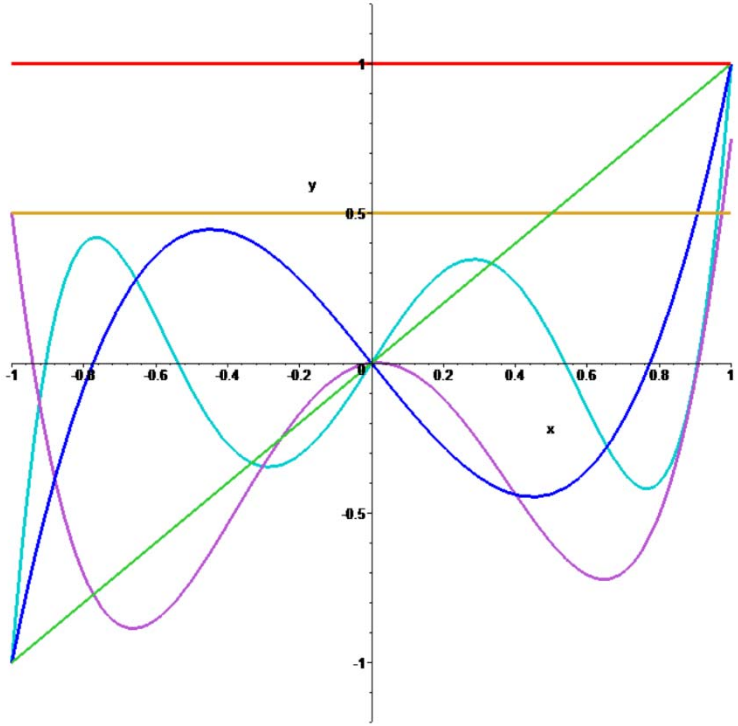
- **单项式：**
  - 从左往右
  - 首先常函数1主宰
  - 接着$x$增长最快
  - 接着$x^2$增长最快
  - 接着$x^3$增长最快
  - …
- **趋势**
  - 好的基函数一般需要系数交替
  - 互相抵消问题



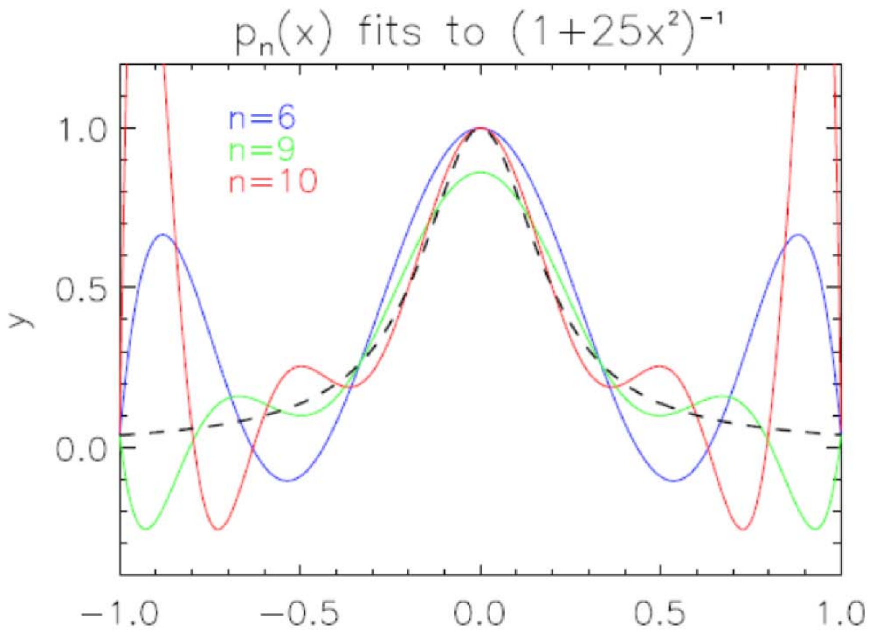
# 解决方法

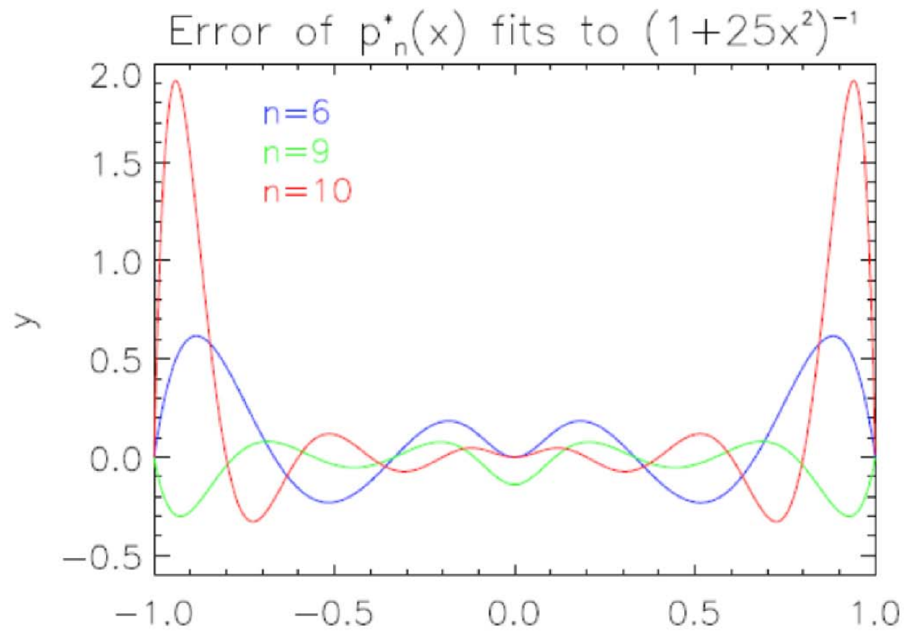
- 使用正交多项式基

- 如何获得?
  - Gram-Schmidt正交化



# 多项式插值结果好吗？




**振荡(龙格Runge)现象和对插值点数的高度敏感性**

**观察$n = 9$(10个数据点)和$n = 10$(11个数据点)的差别**

# 结论

- 多项式插值不稳定
  - 控制点的微小变化可导致完全不同的结果

- 振荡(Runge)现象
  - 多项式随着插值点数(可以是细微)增加而摆动

➡ 需要更好的基函数来做插值
  - Bernstein基函数?
  - 分片多项式?

# 2. 多项式逼近

# 为什么逼近?

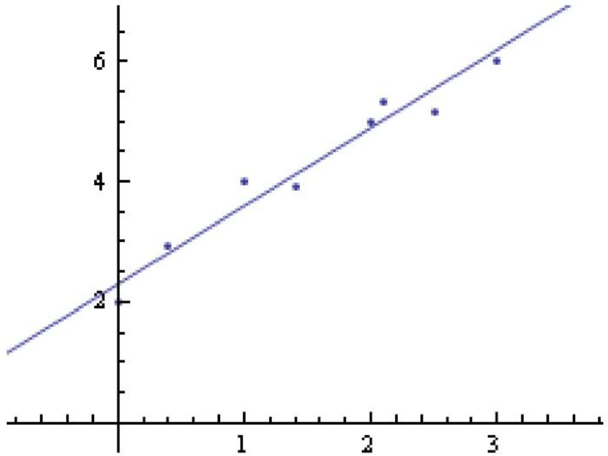
- 数据点含噪声、outliers等

- 更紧凑的表达

- 计算简单、更稳定

# 最小二乘逼近

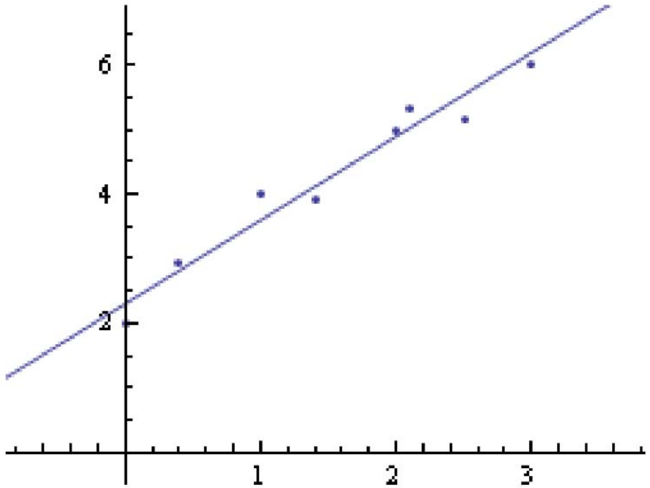
- 逼近问题
  - 给定一组线性无关的连续函数集合$B = \{b_1, \ldots b_n\}$和一组结点$\{(x_1, y_1), \ldots, (x_m, y_m)\}$，其中$m > n$
  - 在$B$张成空间中哪个函数$f \in span(B)$对结点逼近最好？
  - 示例：给定一组点，找到最佳逼近的线性函数
  - 怎么定义“最佳逼近”?



# 最佳逼近的定义

- **最小二乘逼近**

$$\underset{f \in span(B)}{\operatorname{argmin}} \sum_{j=1}^{m} \left(f(x_j) - y_j\right)^2$$




$$M = \begin{pmatrix} b_1(x_1) & \dots & b_n(x_1) \\ \dots & \dots & \dots \\ b_1(x_m) & \dots & b_n(x_m) \end{pmatrix}$$

$$\sum_{j=1}^{m} \left(f(x_j) - y_j\right)^2 = \sum_{j=1}^{m} \left(\sum_{i=1}^{n} \lambda_i b_i(x_j) - y_j\right)^2$$

$$= (M\lambda - y)^T (M\lambda - y)$$

$$= \lambda^T M^T M\lambda - y^T M\lambda - \lambda^T M^T y + y^T y$$

$$= \lambda^T M^T M\lambda - 2y^T M\lambda + y^T y$$

# 求解

- 关于$\lambda$的二次多项式

$$\lambda^T M^T M\lambda - 2y^T M\lambda + y^T y$$

- 法方程
  - 最小解满足

$$M^T M\lambda = M^T y$$

- 提示
  - 最小化二次目标函数 $x^T Ax + b^T x + c$
  - 充分必要条件: $2Ax = -b$

# 3. 函数空间及基函数

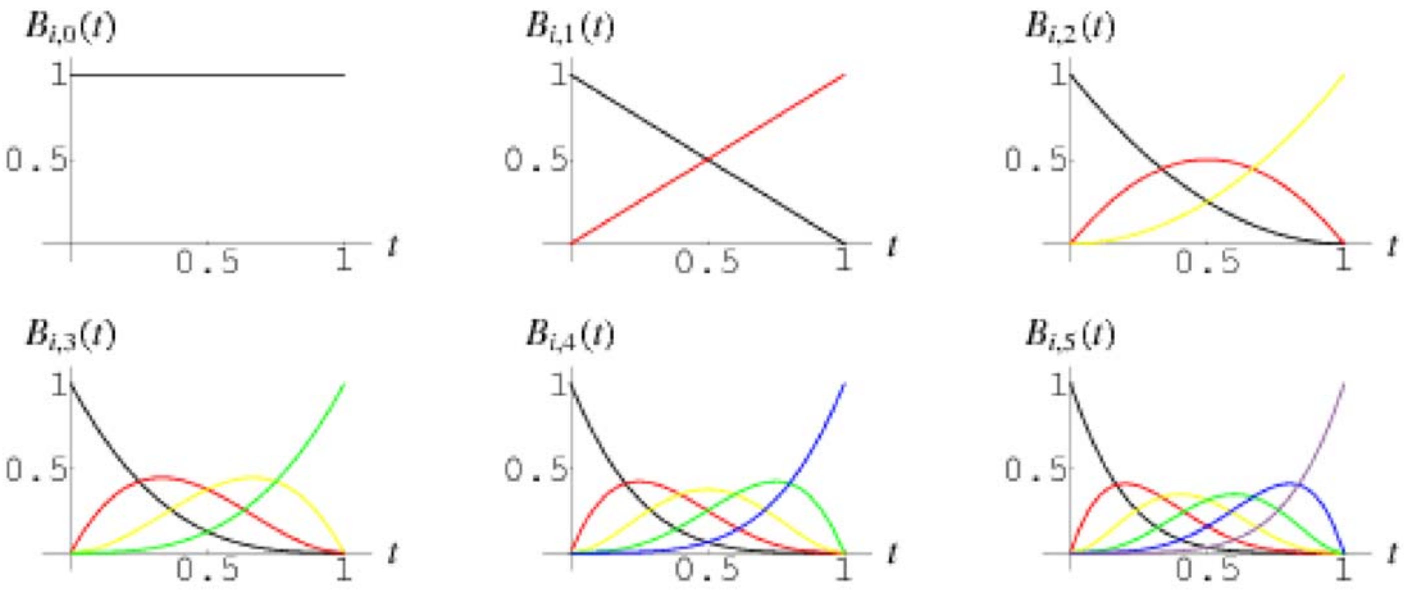
# 为什么用多项式?

- 易于计算，表现良好，光滑，...
- 稠密性与完备性：表达能力足够！
  - 魏尔斯特拉斯Weierstrass定理：令$f$为闭区间$[a,b]$上任意连续函数，则对任意给定$\varepsilon$，存在$n$和多项式$P_n$使得
  $$|f(x)-P_n(x)|<\varepsilon, \forall x \in [a,b]$$
  - Weierstrass只证明了存在性，未给出多项式

# 用Bernstein多项式做逼近

- 伯恩斯坦Bernstein给出了构造性证明（强大!）
  - 对$[\mathbf{0},\mathbf{1}]$区间上任意连续函数$f(x)$和任意正整数$n$，以下不等式对所有$x \in [0,1]$成立
  $$|f(x) - B_n(f,x)| < \frac{9}{4} m_{f,n}$$
  - $m_{f,n} = \underset{y_1,y_2 \in [0,1] \text{且} |y_1 - y_2| < \frac{1}{\sqrt{n}}}{\text{lower upper bound}} |f(y_1) - f(y_2)|$
  - $B_n(f,x) = \sum_{j=0}^{n} f(x_j) b_{n,j}(x)$，其中$x_j$为[0,1]上等距采样点
  - $b_{n,j} = \binom{n}{j} x^j (1-x)^{n-j}$为Bernstein多项式

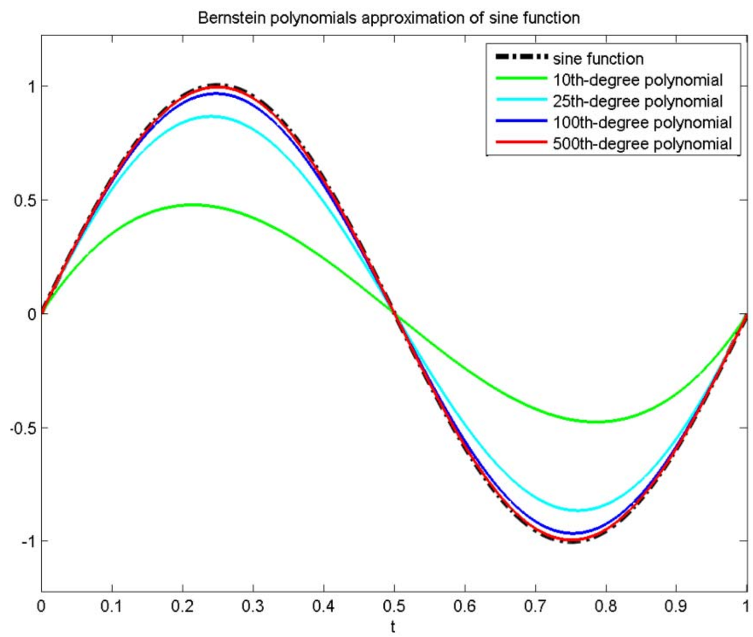
# Bernstein多项式




- $b_{0,0}(x) = 1$
- $b_{0,1}(x) = 1 - x, \quad b_{1,1} = x$
- $b_{0,2}(x) = (1-x)^2, \quad b_{1,2} = 2x(1-x), \quad b_{2,2} = x^2$
- $b_{0,3}(x) = (1-x)^3, \quad b_{1,3} = 3x(1-x)^2, \quad b_{2,3} = 3x^2(1-x), \quad b_{3,3} = x^3$
- $b_{0,4}(x) = (1-x)^4, \quad b_{1,4} = 4x(1-x)^3, \quad b_{2,4} = 6x^2(1-x)^2, \quad b_{3,4} = 4x^3(1-x), \quad b_{4,4} = x^4$

# 用Bernstein多项式做逼近

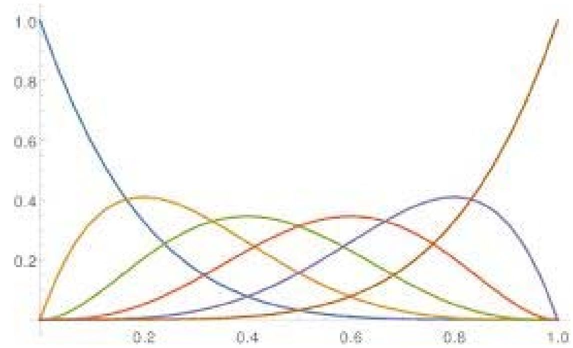
- Bernstein基函数的良好性质：非常好的几何意义！
  - 正性、权性（和为1）→凸包性
  - 变差缩减性
  - 递归线性求解方法
  - 细分性
  - …
- Bernstein多项式逼近示例
  - 逼近结果优秀
  - 需要高阶

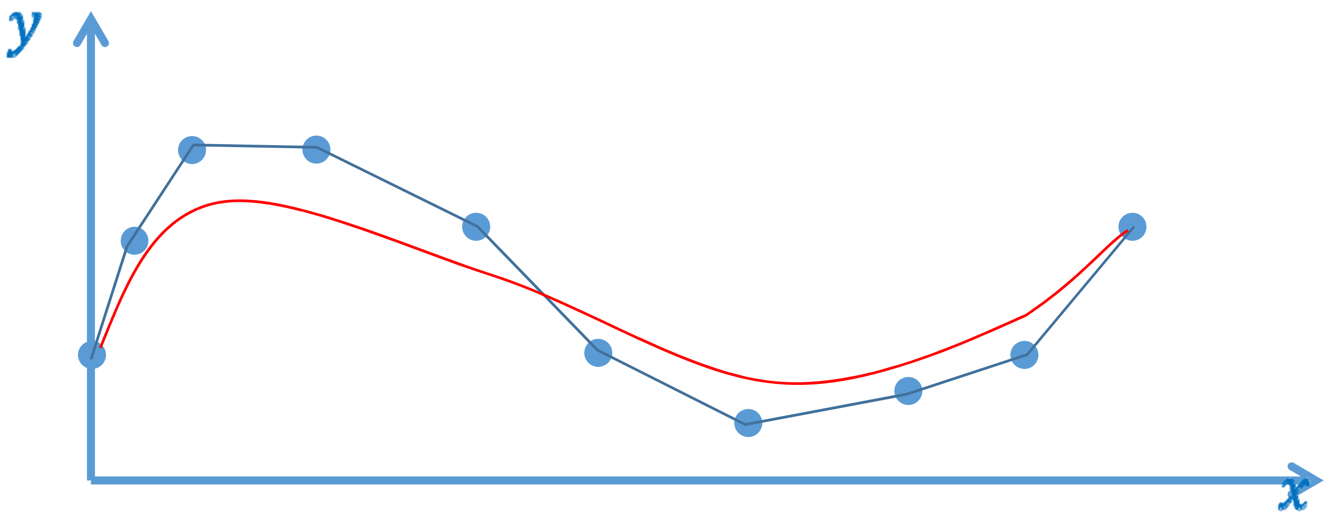



**丰富的理论：CAGD课程**

# 关于Bernstein函数...

- $B_n(f,x) = \sum_{j=0}^{n} f\left(\frac{i}{n}\right) b_{n,j}(x)$
- **两种观点：**
  - **几何观点、代数观点**




**丰富的理论：CAGD课程**

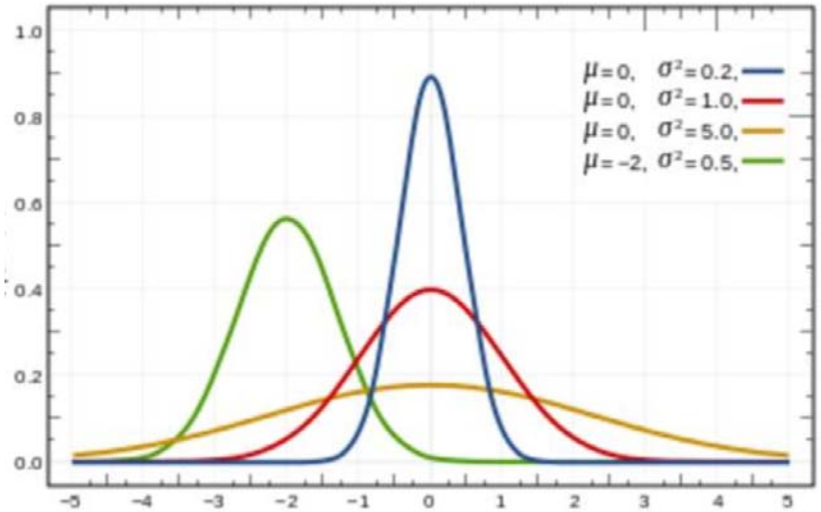
# 4. RBF函数插值/逼近

# Gauss函数

- 两个参数：均值$\mu$，方差$\sigma$

$$g_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- 几何意义：
  - 均值$\mu$：位置
  - 方差$\sigma$：支集宽度




- 不同均值和方差的Gauss函数都线性无关
  - 有什么启发？

# RBF函数拟合

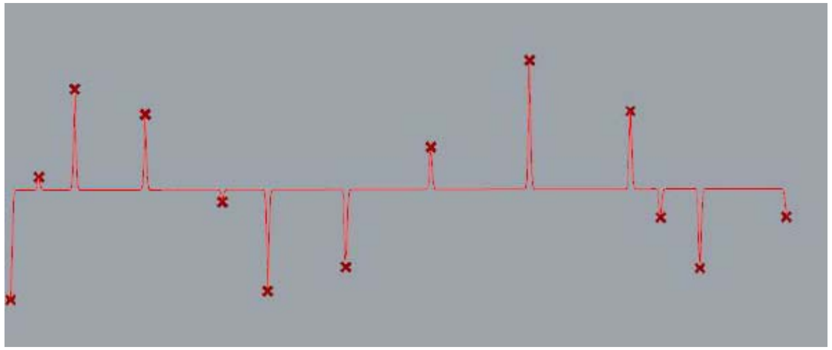
- RBF函数

$$f(x) = b_0 + \sum_{i=1}^{n} b_i g_i(x)$$

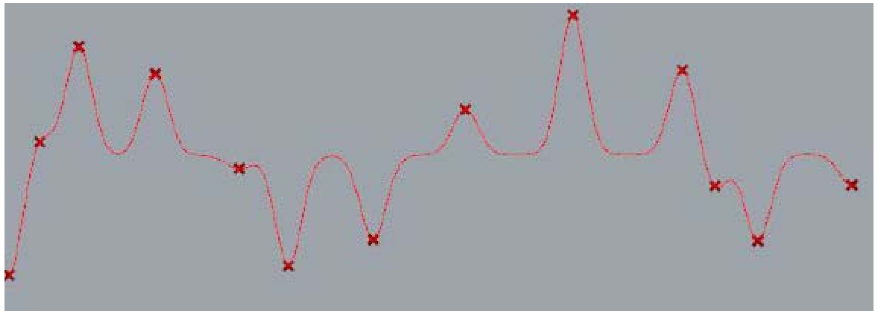
- 方法

- 原因

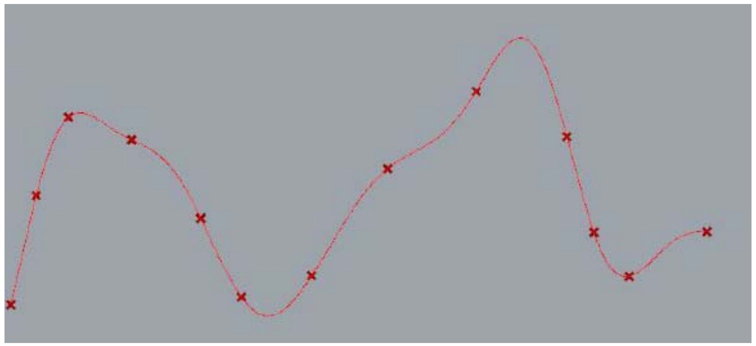
# 讨论：现象



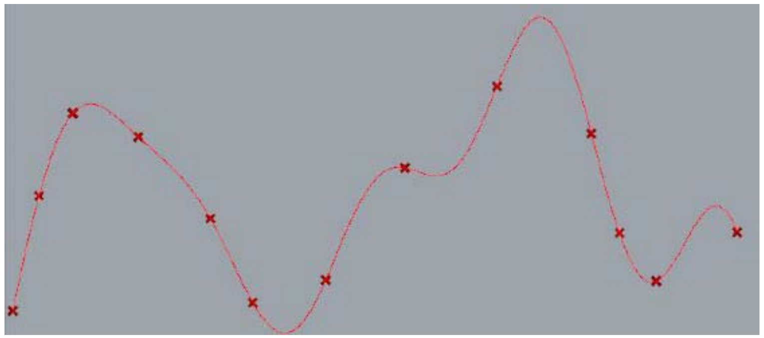
$\sigma = 0.1$



$\sigma = 1$

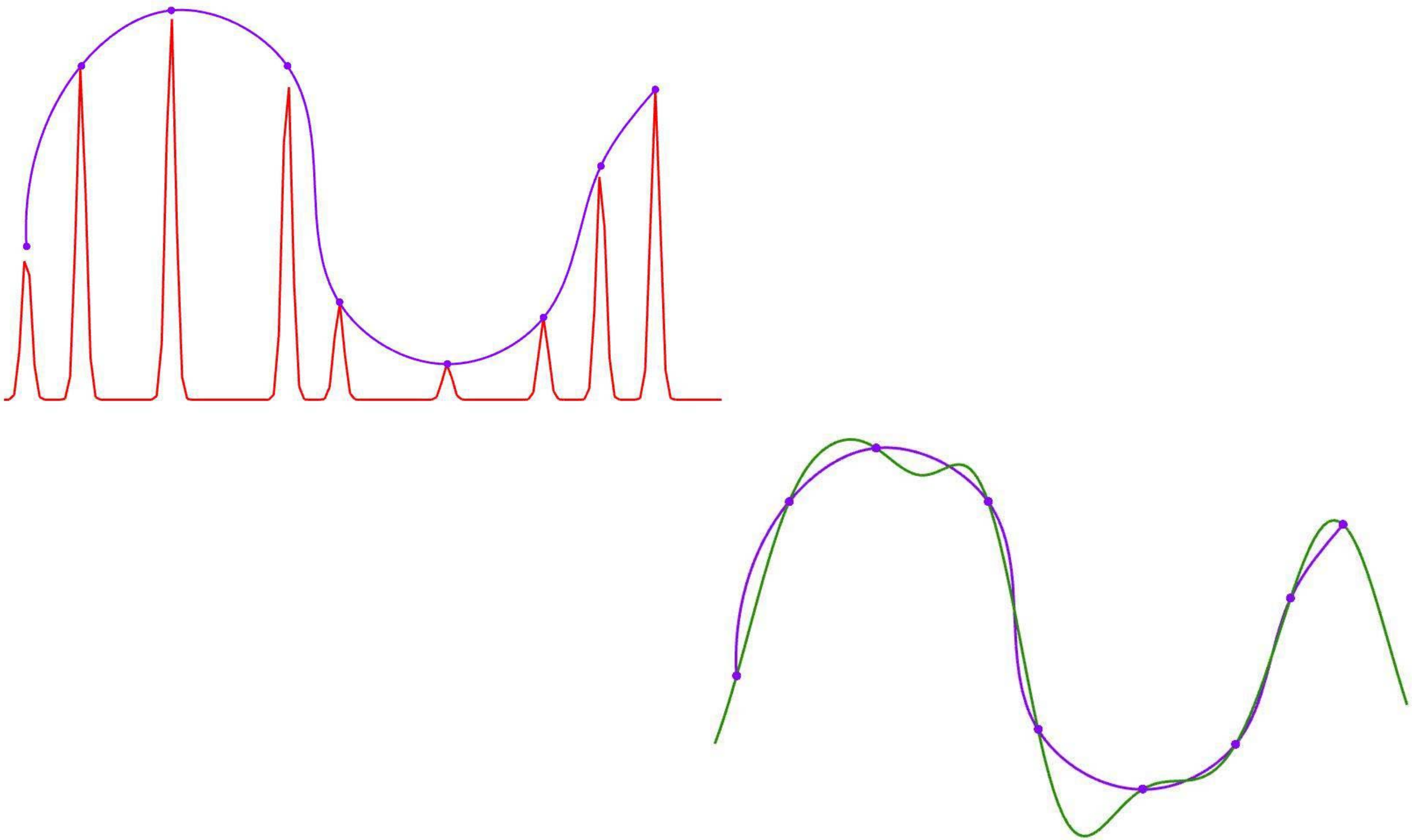


$\sigma = 5$



$\sigma = 10$

# 讨论：现象

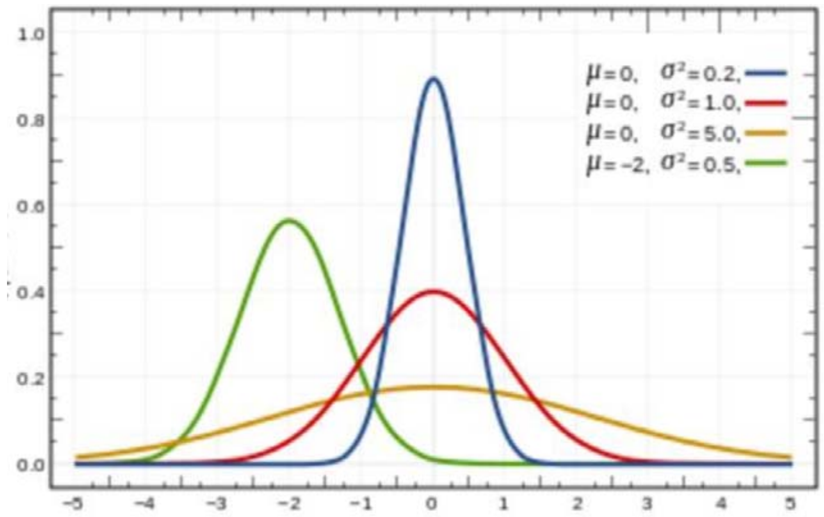


# 思考：

- 均值$\mu$和方差$\sigma$是否可以一起来优化？

$$g_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$f(x) = b_0 + \sum_{i=1}^{n} b_i g_i(x)$$




- 问题？

# 5. 从另一个角度来看拟合函数

# Gauss拟合函数

- 一般Gauss函数表达为标准Gauss函数的形式

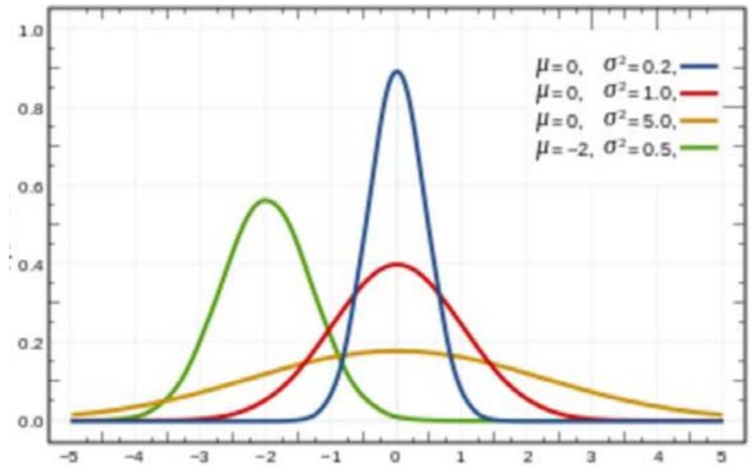
$$g_{\mu,\sigma}(x)=\frac{1}{\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}=\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x}{\sigma}-\frac{\mu}{\sigma}\right)^2}=g_{0,1}(ax+b)$$

$$a=\frac{x}{\sigma},b=-\frac{\mu}{\sigma}$$

$$f(x)=b_0+\sum_{i=1}^{n}b_i g_i(x)$$

$$f(x)=\omega_0+\sum_{i=1}^{n}\omega_i \underline{g_{0,1}(a_i x+b_i)}$$

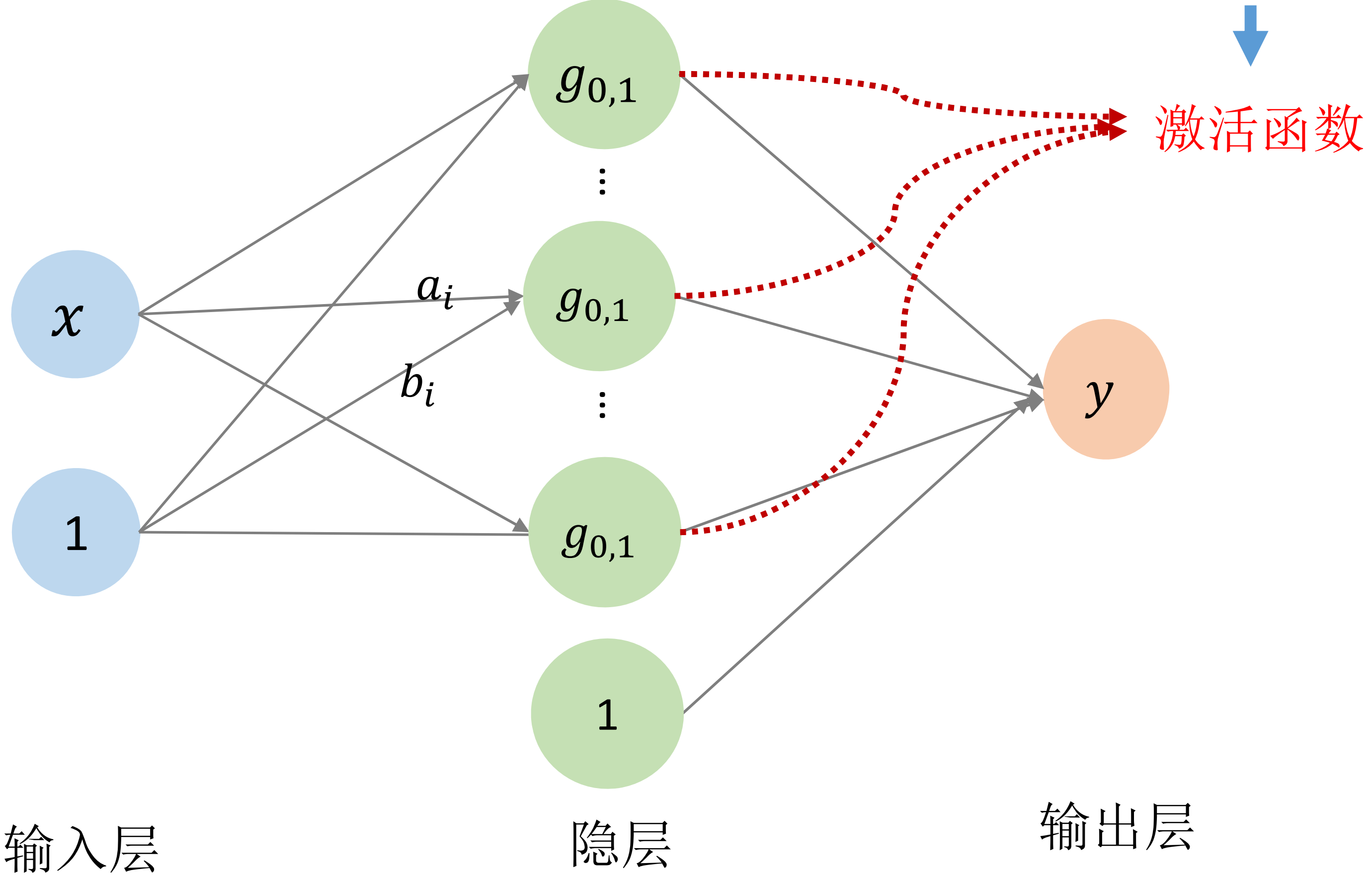
基函数是由一个基本函数通
过平移和伸缩变换而来的



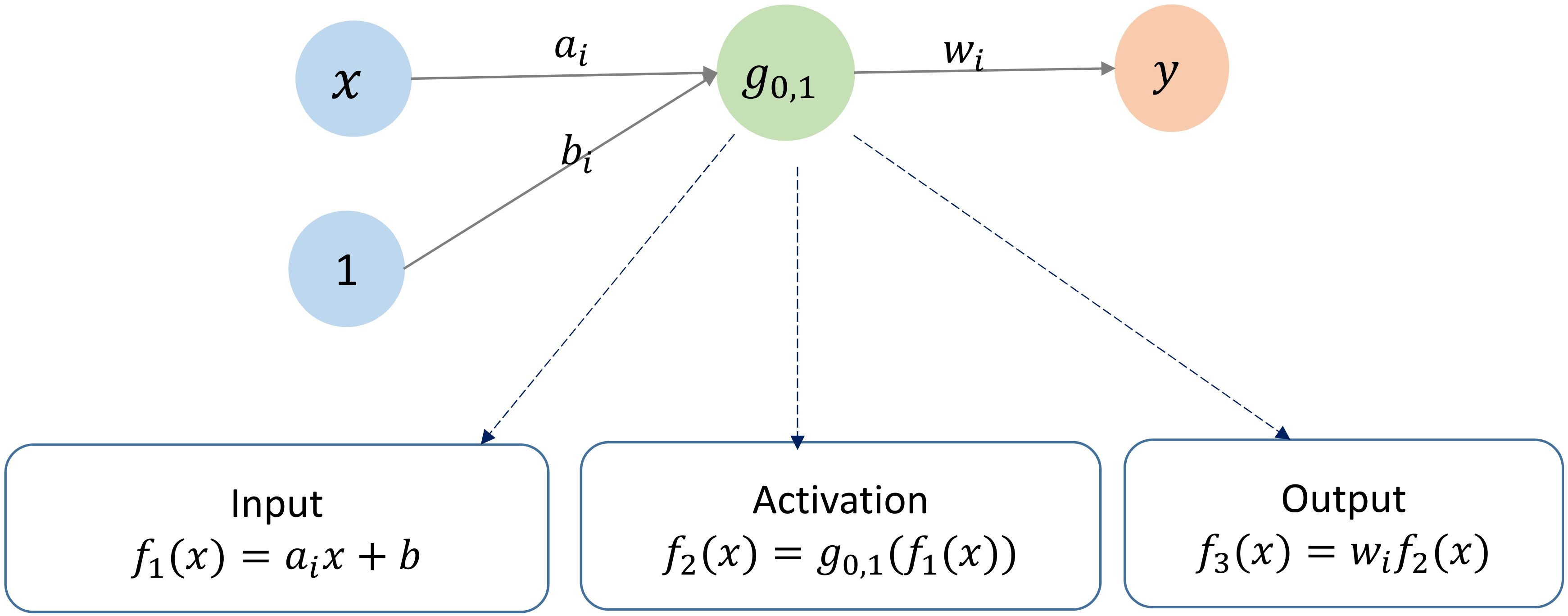
# 换个方式看函数：神经网络

- 将Gauss函数看成网络

$$f(x) = \omega_0 + \sum_{i=1}^{n} \underline{\omega_i g_{0,1}(a_i x + b_i)}$$



# 抽象：神经元



$x$
$1$
$a_i$
$b_i$
$g_{0,1}$
$w_i$
$y$
Input
$f_1(x) = a_i x + b$
Activation
$f_2(x) = g_{0,1}(f_1(x))$
Output
$f_3(x) = w_i f_2(x)$

# RBF 神经网络

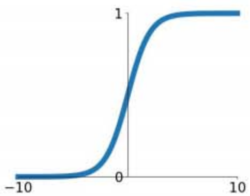
- 高维情形： RBF (Radial Basis Function)， 径向基函数
- 一种特殊的BP网络
  - 优化： BP算法
- 核函数思想
- Gauss函数的特性： 拟局部性

# 思考：激活函数的选择？

- 启发：由一个简单的函数通过（仿射）变换构造出一组基函数，张成一个函数空间
- 表达能力是否足够强：是否完备/稠密的？

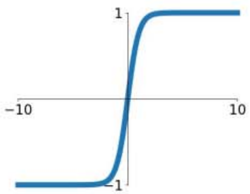
**Sigmoid**

$\sigma(x) = \frac{1}{1+e^{-x}}$



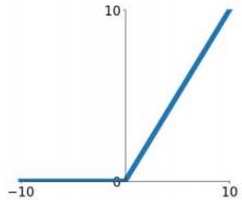

**tanh**

$\tanh(x)$



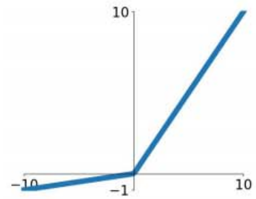

**ReLU**

$\max(0, x)$




**Leaky ReLU**

$\max(0.1x, x)$

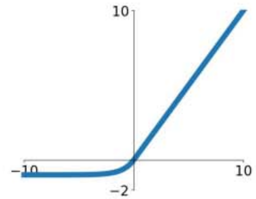



**Maxout**

$\max(w_1^T x + b_1, w_2^T x + b_2)$

**ELU**

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



# 高维情形：多元函数
（后面的课程再展开解释）

- 变量的多个分量的线性组合

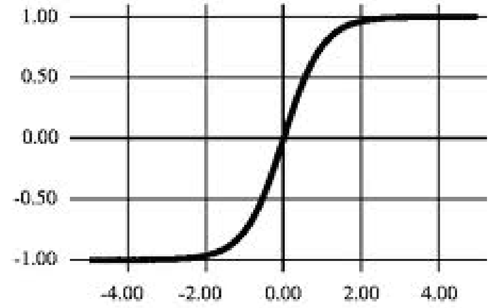
$$(x_1, x_2, \dots, x_n) \longrightarrow g_{0,1}(a_1^i x_1 + a_2^i x_2 + \dots + a_n^i x_n + b_i)$$

- 单隐层神经网络函数：

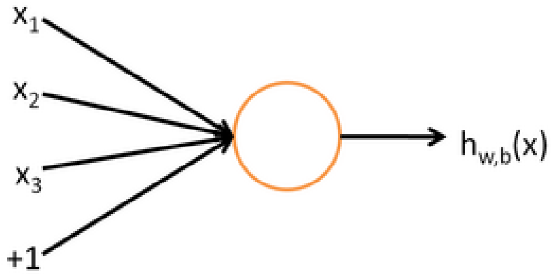
$$f(x_1, x_2, \dots, x_n) = \omega_0 + \sum_{i=1}^{n} \omega_i g_{0,1}\left(a_1^i x_1 + a_2^i x_2 + \dots + a_n^i x_n + b_i\right)$$

# 多层神经网络：多重复合的函数

- 线性函数和非线性函数的多重复合

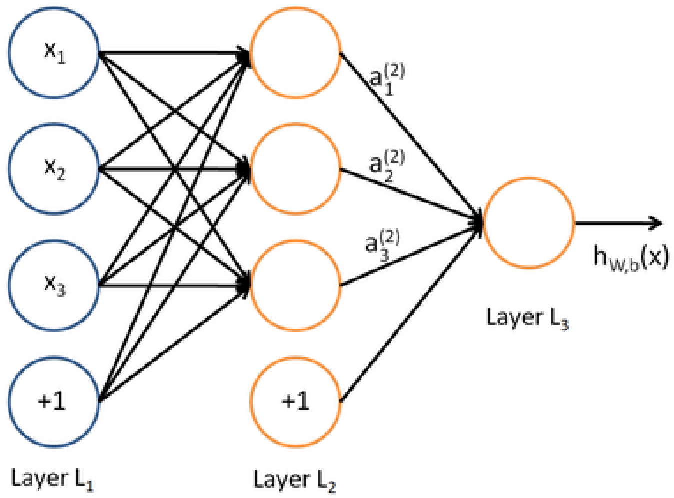



$f(net)$




Neuron

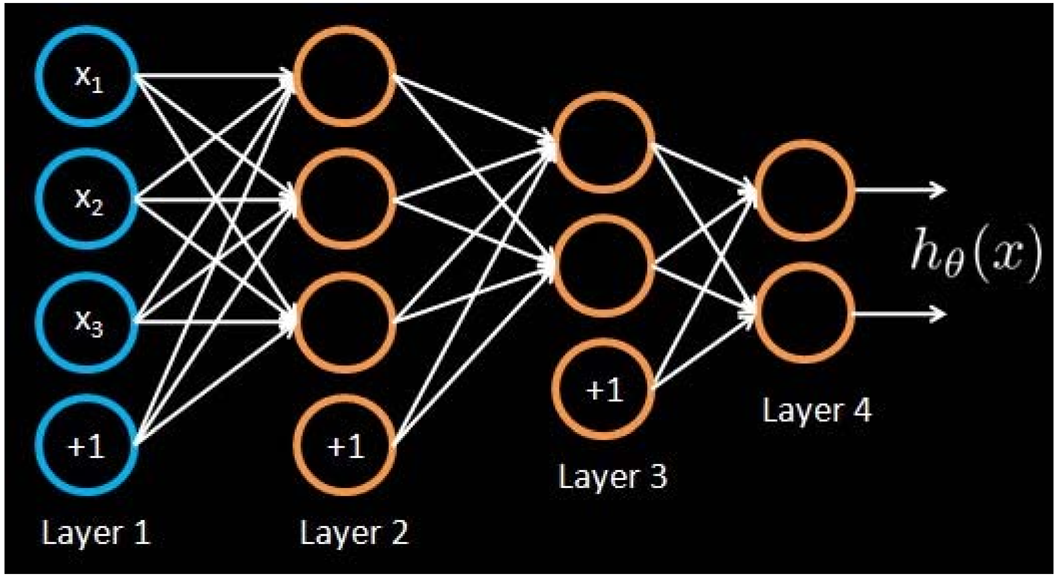
$$h_{W,b}(x) = f(W^T x) = f(\textstyle\sum_{i=1}^{3} W_i x_i + b)$$




Neural network

$$\begin{aligned} a_1^{(2)} &= f(W_{11}^{(1)} x_1 + W_{12}^{(1)} x_2 + W_{13}^{(1)} x_3 + b_1^{(1)}) \\ a_2^{(2)} &= f(W_{21}^{(1)} x_1 + W_{22}^{(1)} x_2 + W_{23}^{(1)} x_3 + b_2^{(1)}) \\ a_3^{(2)} &= f(W_{31}^{(1)} x_1 + W_{32}^{(1)} x_2 + W_{33}^{(1)} x_3 + b_3^{(1)}) \\ h_{W,b}(x) = a_1^{(3)} &= f(W_{11}^{(2)} a_1^{(2)} + W_{12}^{(2)} a_2^{(2)} + W_{13}^{(2)} a_3^{(2)} + b_1^{(2)}) \end{aligned}$$

# 用神经网络函数来拟合数据




Regression problem:

Input: Given training set $(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots$

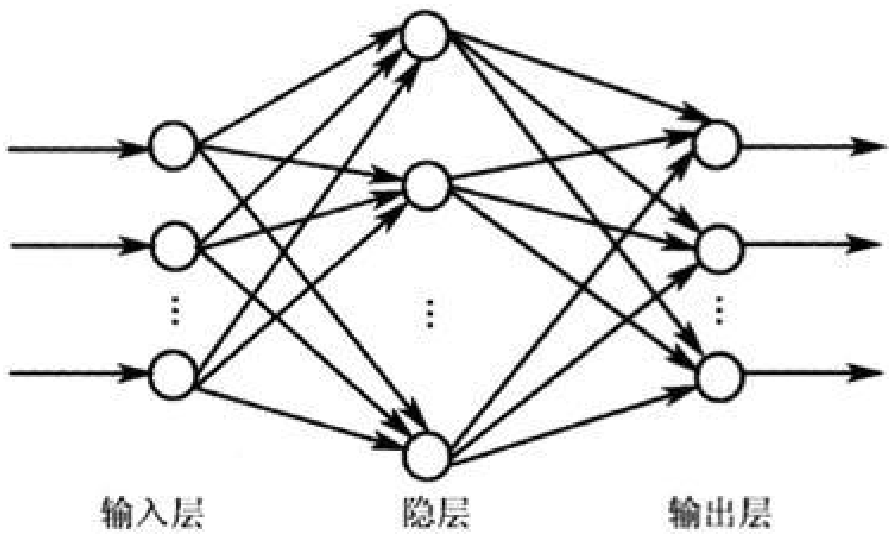
Output: Adjust parameters $\theta$ (for every node) to make:

$$h(x_i) \approx y_i$$

# Why it works?

- 万能逼近定理：自由度足够多！

$$F(\mathrm{x})=\sum_{i=1}^{N} v_i \varphi\left(\mathrm{w}_i^{\mathrm{T}}\mathrm{x}+b_i\right)$$




与传统拟合一样存在同样的问题：
函数个数如何选？！

调参！

# Deep Learning Frameworks

TensorFlow

DL4J Deep Learning for Java

neon

PaddlePaddle

K Keras

theano

torch

Chainer

dmlc
mxnet

Caffe

Microsoft
CNTK

cuDNN

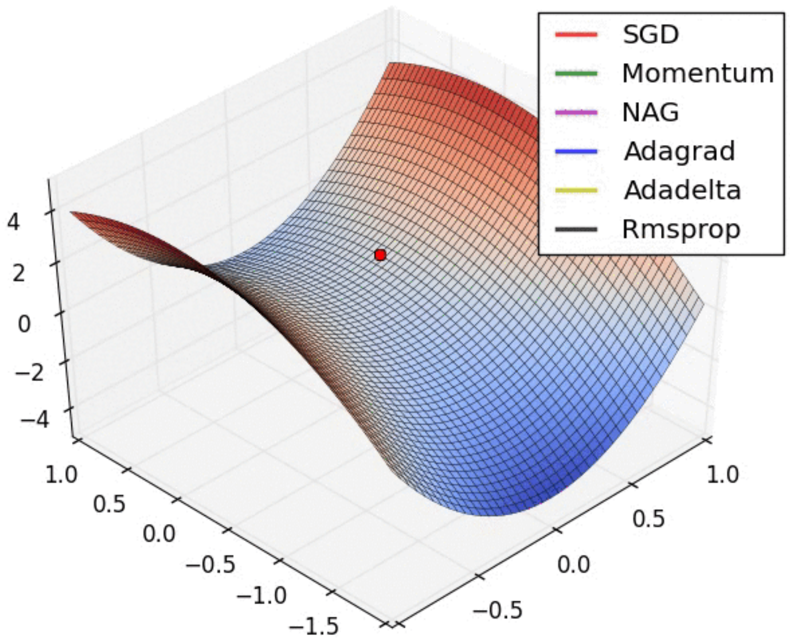
Lasagne

NVIDIA DIGITS

# 使用深度学习的方法

- 问题建模
  - 理解问题、问题分解（多个映射级联）…
- 找哪个？
  - 损失函数、各种Penalty、正则项…
- 到哪找？
  - 神经网络函数、网络简化…
- 怎么找？
  - 优化方法（BP方法）
  - 初始值、参数…

**调参：有耐心、有直觉…**



# 未来课程内容

- 全局函数→局部函数
  - 样条函数
- 多元函数→一般曲线
  - 参数曲线、参数域为本征维数
- 隐函数
- 曲线设计
  - 计算机辅助几何设计
- 曲面设计
  - 张量积的参数曲面

# 作业1情况

- 作业1情况
  - 演示优秀demo
  - 优秀代码和优秀报告
- 其他学员可以继续完成提交
  - 可参照优秀作业尽快完成，赶上大部队

# 作业2

- 任务
  - 使用RBF神经网络函数来拟合数据
    - 仍限制在函数情形
  - 与作业1的方法进行比较
- 目的
  - 理解神经网络优化
  - 学习使用TensorFlow来优化
- 要求
  - 推荐使用无境框架：后面的网格处理较方便
    - Windows, VS2019, DirectX, 显卡要求
  - 可以使用其他语言(Matlab, Python等)或其他框架
- Deadline：2020 年10 月24 日晚

# 谢 谢 ！