模式识别与机器学习

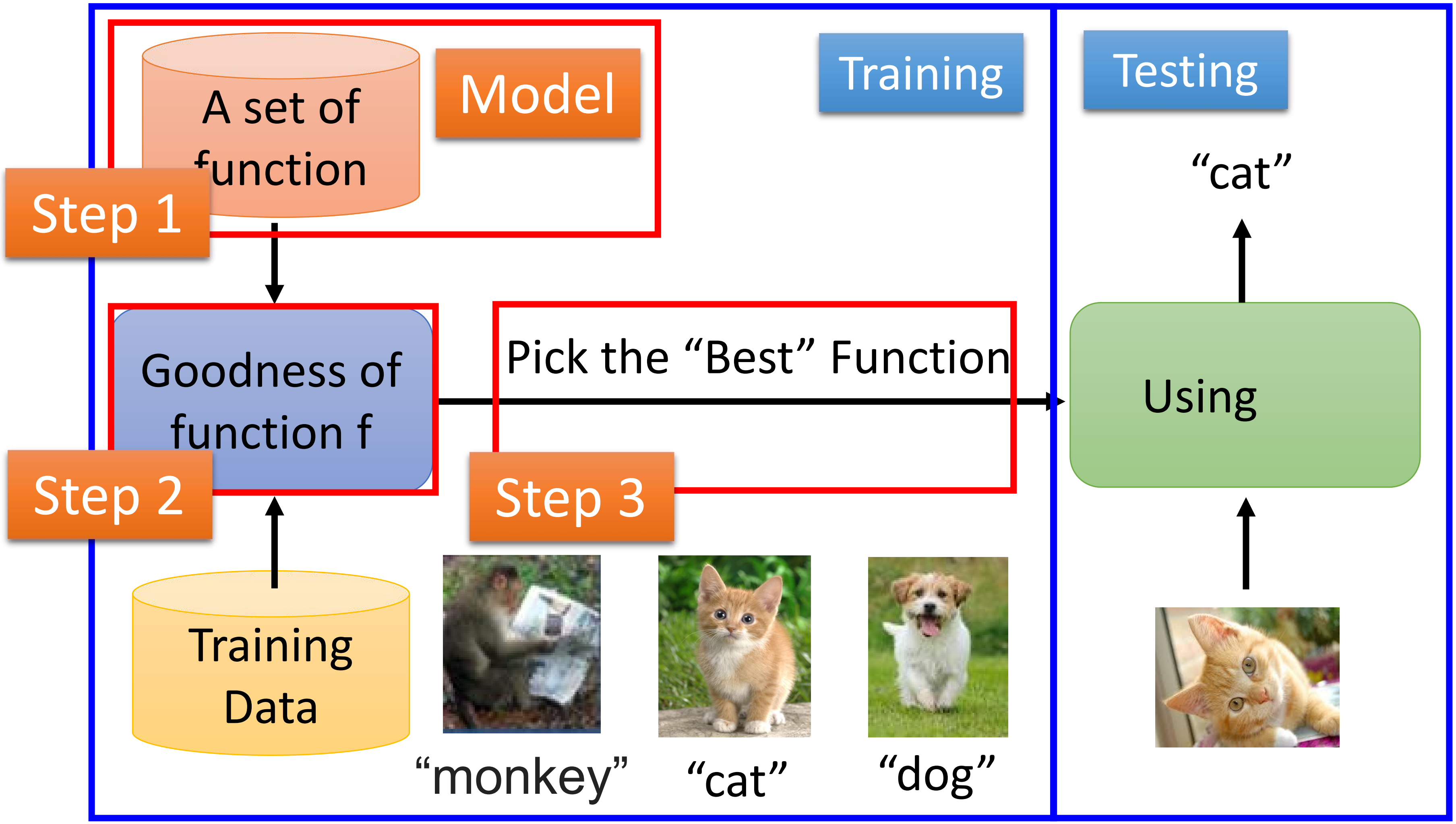
Pattern Recognition and Machine Learning

课程内容

- **模式识别与机器学习概述**
- **模式识别与机器学习的基本方法**
  - 回归分析、线性判别函数、线性神经网络、核方法和支持向量机、决策树分类
  - 贝叶斯统计决策理论、概率密度函数估计
  - 无监督学习和聚类
  - 特征选择与提取

监督式学习--三段式解法

- 有监督学习：　找一個函数的能力



参考自李宏毅老师课件

监督式学习--三段式解法

- **有监督学习： 找一個函数的能力**

Step 0: What kind of function do you want to find?

Regression, Classification, ……

Step 1: define a set of function

Linear Classifier
Probabilistic
SVM
Deep Learning
Decision Tree

Step 2: goodness of function

Regression(MSE)
Classification (Cross Entropy)
……

Step 3: pick the best function

Gradient Descent
……

回归分析任务简述

- 监督学习从给定的训练数据集中学习出数据及其标记的映射函数，当新的数据到来时，可以根据这个函数预测新数据的结果。训练集中的数据标记是由人标注的。
- 常见的监督学习算法包括回归分析和分类。
  - 回归：函数的输出是一个连续的值
  - 分类：函数的输出是一个离散的值（二分类、多分类）

监督式学习--三段式解法

- **Step 1: Model （模型）**

$$y = b + w \cdot x$$

w and b are parameters (can be any value)

A set of function

Model

$f_1, f_2 \cdots$

f_1: $y = 10.0 + 9.0 \cdot x$

f_2: $y = 9.8 + 9.2 \cdot x$

f_3: $y = -0.8 - 1.2 \cdot x$

…… infinite

$$f(x) = y$$

Linear model: $y = b + \sum w_i x_i$

x_i: $x_{cp}, x_{hp}, x_w, x_h \ldots$

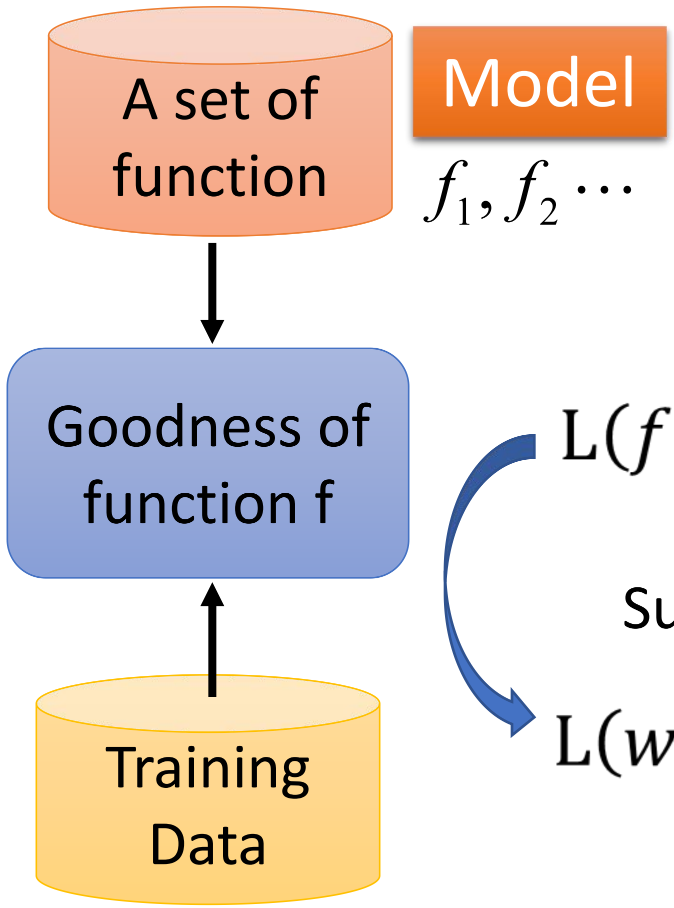
feature

w_i: weight, b: bias

监督式学习--三段式解法

- **Step 2: Goodness of Function （策略)**

$y = b + w \cdot x$



Loss function L:

Input: a function, output: how bad it is

Estimation error

$$L(f) = \underbrace{\sum_{n=1}^{10}}_{\text{Sum over examples}} \boxed{\left(\hat{y}^n - \underline{f(x^n)}\right)^2}$$

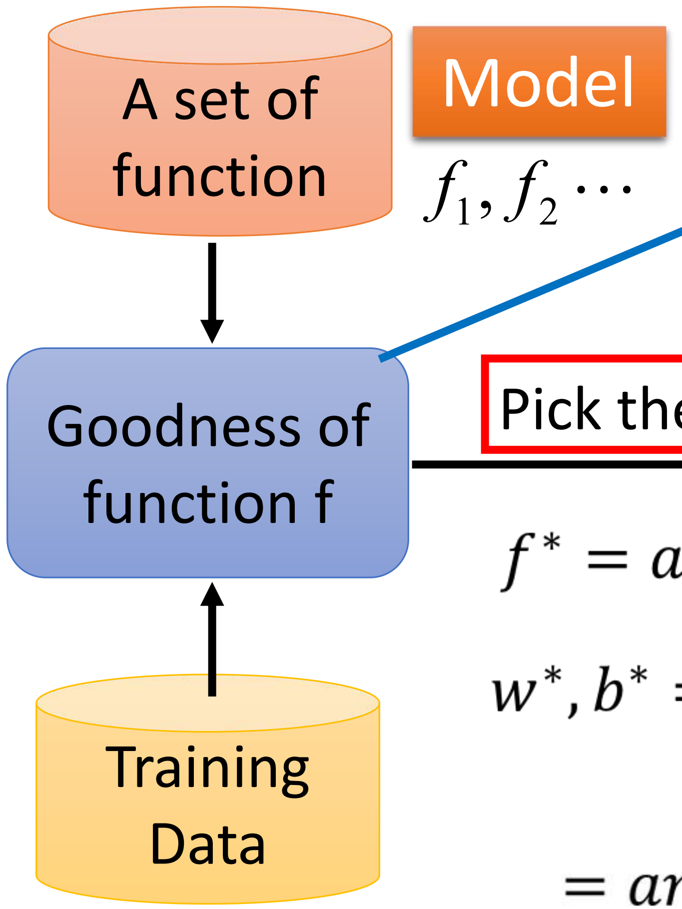
Sum over examples

Estimated y based on input function

$$L(w, b) = \sum_{n=1}^{10} \left(\hat{y}^n - (b + w \cdot x^n)\right)^2$$

监督式学习--三段式解法

- **Step 3: Best Function （优化方法）**



$$L(w,b) = \sum_{n=1}^{10} \left(\hat{y}^n - (b + w \cdot x^n)\right)^2$$

Pick the "Best" Function

$$f^* = arg \min_f L(f)$$

$$w^*, b^* = arg \min_{w,b} L(w,b)$$

$$= arg \min_{w,b} \sum_{n=1}^{10} \left(\hat{y}^n - (b + w \cdot x^n)\right)^2$$

参考自李宏毅老师课件

课程内容

- **回归分析**
  - 回归分析任务简述
  - 多元线性回归
  - 多项式回归
  - 其他回归方法

多元线性回归

■ 线性模型一般形式

$$f(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d + b$$

是由属性 $\boldsymbol{x} = (x_1; x_2; \ldots; x_d)$ 的示例，其中x_i是$\boldsymbol{x}$ 在第i 个属性上的取值

■ 向量形式

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$$

其中 $\boldsymbol{w} = (w_1; w_2; \ldots; w_d)$

多元线性回归

- 形式简单、易于建模
- 可解释性
- 非线性模型的基础
  - 引入层级结构或高维映射

- 一个例子
  - 综合考虑色泽、根蒂和敲声来判断西瓜好不好
  - 其中根蒂的系数最大，表明根蒂最要紧；而敲声的系数比色泽大，说明敲声比色泽更重要

$$f_{\text{好瓜}}(\boldsymbol{x}) = 0.2 \cdot x_{\text{色泽}} + 0.5 \cdot x_{\text{根蒂}} + 0.3 \cdot x_{\text{敲声}} + 1$$

多元线性回归

- 给定数据集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$

其中 $\boldsymbol{x}_i = (x_{i1}; x_{i2}; \ldots; x_{id})$ $y_i \in \mathbb{R}$

- 线性回归（linear regression）目的

学得一个线性模型以尽可能准确地预测实值输出标记

- 模型（Model of Step 1）确定后，如何确定具体的线性模型

Step 1: define a set of function → Step 2: goodness of function → Step 3: pick the best function

√ ? ?

多元线性回归

- 单一属性的线性回归目标

$$f(x) = wx_i + b \quad \text{使得} \quad f(x_i) \simeq y_i$$

- 参数/模型估计：最小二乘法（least square method）

$$\begin{aligned}(w^*, b^*) &= \arg\min_{(w,b)} \sum_{i=1}^{m} \left(f(x_i) - y_i\right)^2 \\ &= \arg\min_{(w,b)} \sum_{i=1}^{m} \left(y_i - wx_i - b\right)^2\end{aligned}$$

多元线性回归

- 最小化均方误差

$$E_{(w,b)} = \sum_{i=1}^{m} (y_i - wx_i - b)^2$$

- 分别对w 和b 求导，可得

$$\frac{\partial E_{(w,b)}}{\partial w} = 2\left(w\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m} (y_i - b)\, x_i\right)$$

$$\frac{\partial E_{(w,b)}}{\partial b} = 2\left(mb - \sum_{i=1}^{m} (y_i - wx_i)\right)$$

多元线性回归

- 得到闭式（closed-form）解

$$w = \frac{\sum_{i=1}^{m} y_i \left(x_i - \bar{x}\right)}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m} x_i\right)^2}$$

$$b = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - w x_i\right)$$

- 其中

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i$$

多元线性回归

- 给定数据集

$$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$$

$$\boldsymbol{x}_i = (x_{i1}; x_{i2}; \ldots; x_{id}) \quad y_i \in \mathbb{R}$$

- 多元线性回归目标

$$f(\boldsymbol{x}_i) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_i + b \text{ 使得 } f(\boldsymbol{x}_i) \simeq y_i$$

多元线性回归

- 把$\boldsymbol{w}$和 b 吸收入向量形式 $\hat{\boldsymbol{w}} = (\boldsymbol{w}; b)$ 数据集表示为

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} & 1 \\ \boldsymbol{x}_2^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathrm{T}} & 1 \end{pmatrix}$$

$$\boldsymbol{y} = (y_1; y_2; \ldots; y_m)$$

多元线性回归

■ 最小二乘法（least square method）

$$\hat{\boldsymbol{w}}^* = \underset{\hat{\boldsymbol{w}}}{\arg\min} \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}^{\mathrm{T}}\right) \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)$$

令 $E_{\hat{\boldsymbol{w}}} = \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)^{\mathrm{T}} \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)$ ，对$\hat{\boldsymbol{w}}$求导得到

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\mathbf{X}^{\mathrm{T}} \left(\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}\right)$$

令上式为零可得$\hat{\boldsymbol{w}}$最优解的闭式解

多元线性回归

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 是满秩矩阵或正定矩阵，则

$$\hat{\boldsymbol{w}}^{*}=\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

其中$\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}$是 $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 的逆矩阵，线性回归模型为

$$f\left(\hat{\boldsymbol{x}}_i\right)=\hat{\boldsymbol{x}}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 不是满秩矩阵
  - 引入正则化

多元线性回归--解析法回顾

- 对一元线性回归，最小化均方误差

$$E_{(w,b)}=\sum_{i=1}^{m}(y_i-wx_i-b)^2$$

- 分别对w 和 b 求导，可得

$$\frac{\partial E_{(w,b)}}{\partial w}=2\left(w\sum_{i=1}^{m}x_i^2-\sum_{i=1}^{m}(y_i-b)\,x_i\right)$$

$$\frac{\partial E_{(w,b)}}{\partial b}=2\left(mb-\sum_{i=1}^{m}(y_i-wx_i)\right)$$

多元线性回归--解析法回顾

- 得到闭式（closed-form）解

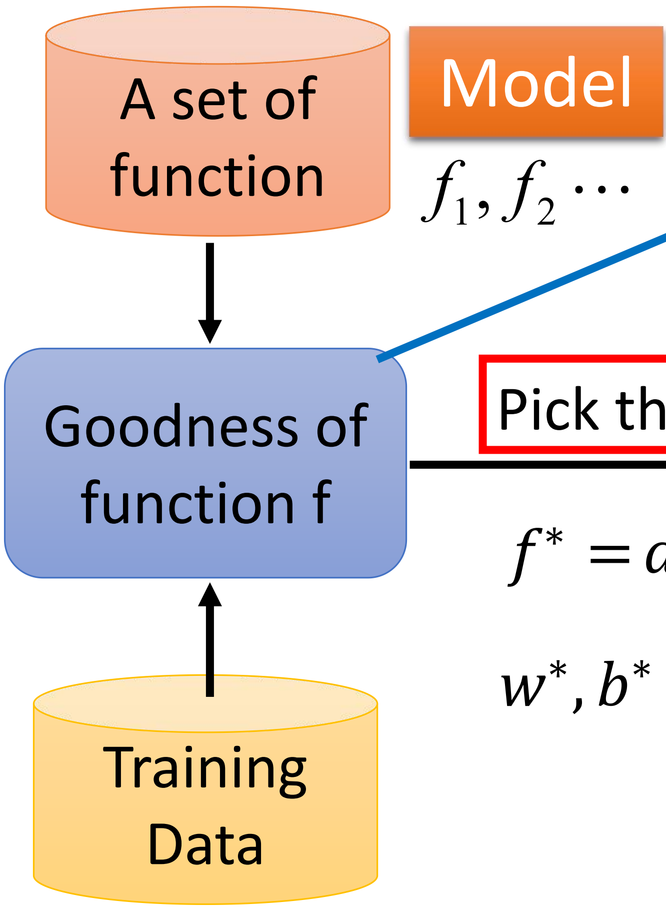
$$w=\frac{\sum_{i=1}^{m} y_i\left(x_i-\bar{x}\right)}{\sum_{i=1}^{m} x_i^2-\frac{1}{m}\left(\sum_{i=1}^{m} x_i\right)^2}$$

$$b=\frac{1}{m} \sum_{i=1}^{m}\left(y_i-w x_i\right)$$

- 其中

$$\bar{x}=\frac{1}{m} \sum_{i=1}^{m} x_i$$

多元线性回归--梯度下降法



$$L(w,b) = \sum_{n=1}^{10}\left(\hat{y}^n - \left(b + w \cdot x_{cp}^n\right)\right)^2$$

Pick the "Best" Function

$$f^* = arg\min_f L(f)$$

$$w^*, b^* = arg\min_{w,b} L(w,b)$$

$$= arg\min_{w,b} \sum_{n=1}^{10}\left(\hat{y}^n - \left(b + w \cdot x_{cp}^n\right)\right)^2$$

Gradient Descent

参考自李宏毅老师课件

多元线性回归--梯度下降法

$$\nabla L = \begin{bmatrix} \frac{\partial L}{\partial w} \\ \frac{\partial L}{\partial b} \end{bmatrix}$$ gradient

■ Step 3: Best Function

- How about two parameters? $w^*, b^* = arg \min_{w,b} L(w, b)$

➢ (Randomly) Pick an initial value w^0, b^0

➢ Compute $\frac{\partial L}{\partial w}|_{w=w^0,b=b^0}, \frac{\partial L}{\partial b}|_{w=w^0,b=b^0}$

$$w^1 \leftarrow w^0 - \eta \frac{\partial L}{\partial w}|_{w=w^0,b=b^0} \qquad b^1 \leftarrow b^0 - \eta \frac{\partial L}{\partial b}|_{w=w^0,b=b^0}$$

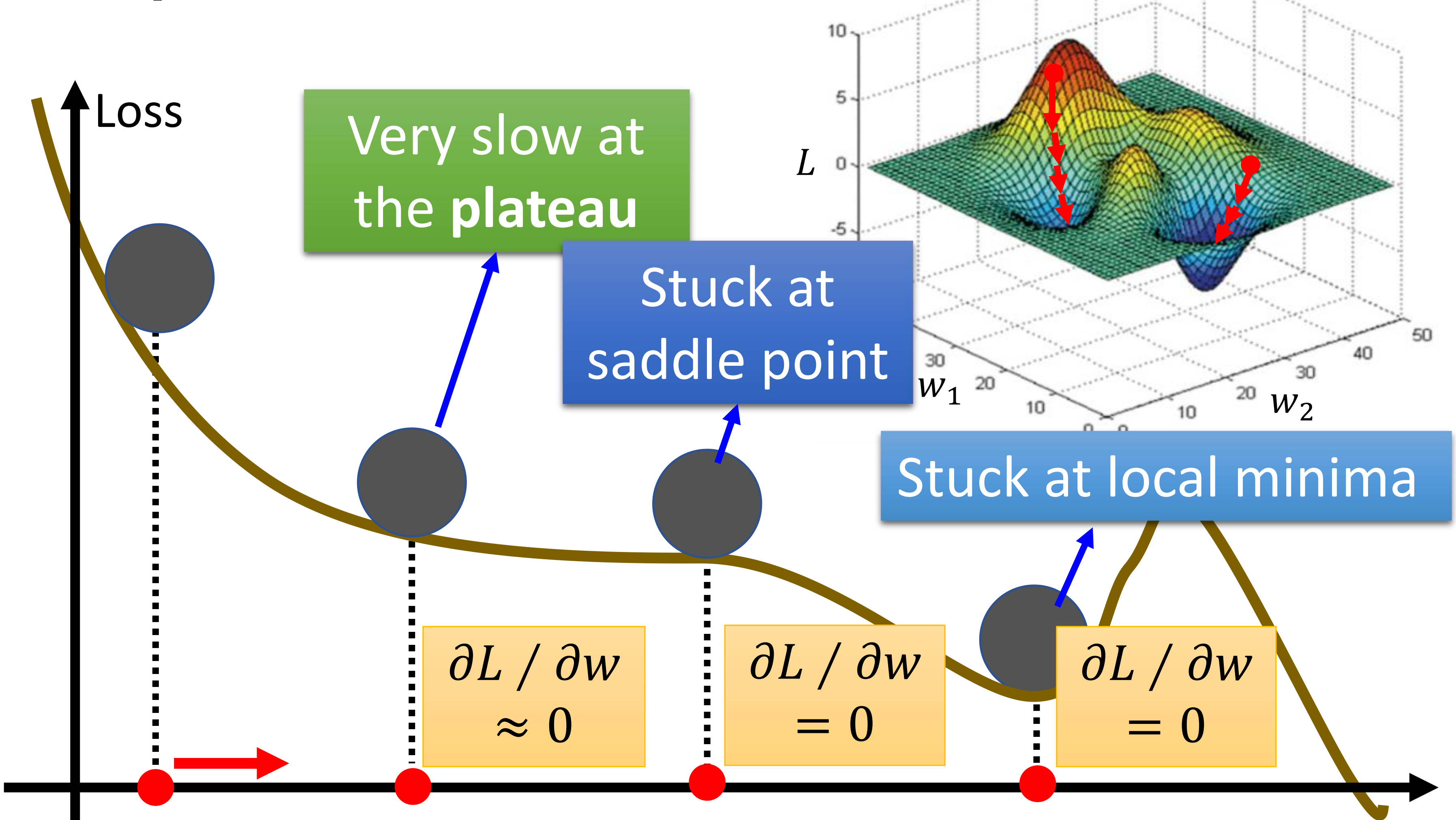
➢ Compute $\frac{\partial L}{\partial w}|_{w=w^1,b=b^1}, \frac{\partial L}{\partial b}|_{w=w^1,b=b^1}$

$$w^2 \leftarrow w^1 - \eta \frac{\partial L}{\partial w}|_{w=w^1,b=b^1} \qquad b^2 \leftarrow b^1 - \eta \frac{\partial L}{\partial b}|_{w=w^1,b=b^1}$$

参考自李宏毅老师课件

多元线性回归--梯度下降法

■ Step 3: Best Function



多项式回归

■ **线性函数?**

$y = b + w \cdot x$

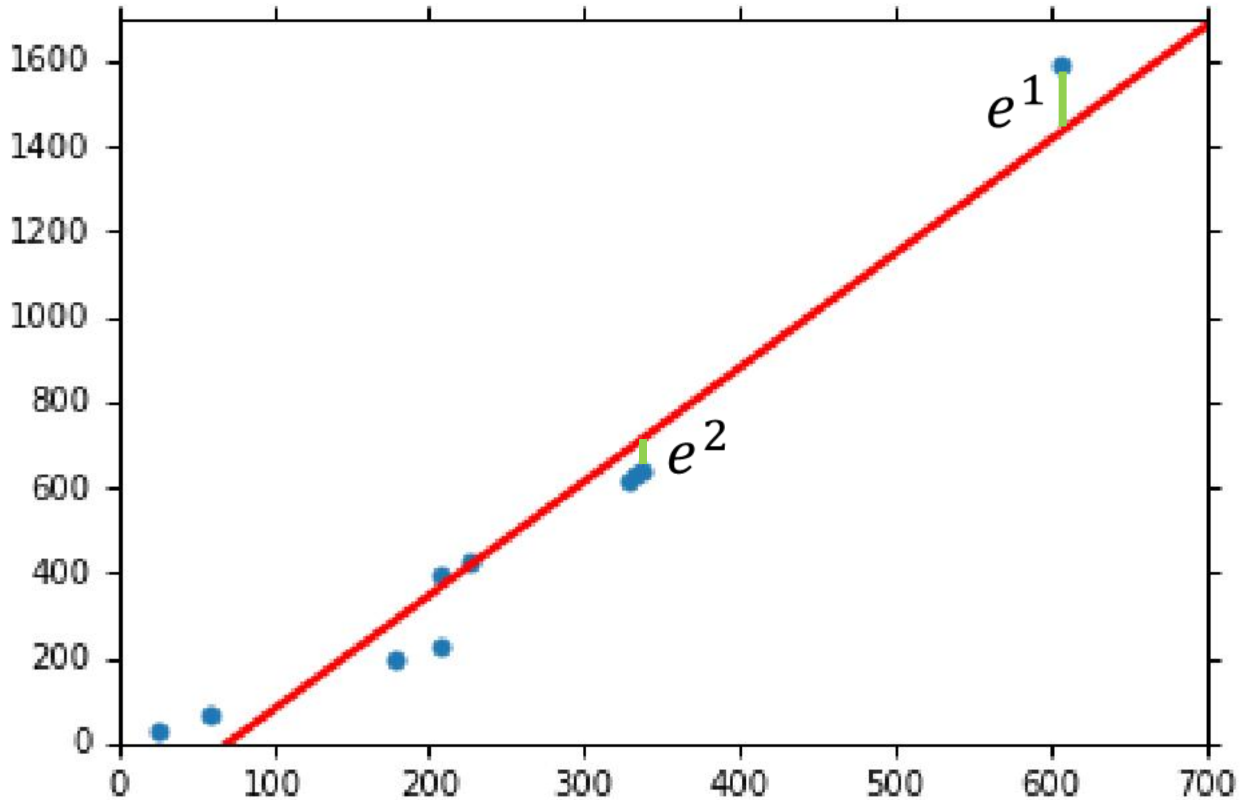
b = -188.4

w = 2.7

Average Error on Training Data

$$= \frac{1}{10}\sum_{n=1}^{10} e^n = 31.9$$

Training Data



多项式回归

■ **线性函数?**

What we really care about is the error on new data (testing data)

$y = b + w \cdot x$

$b = -188.4$

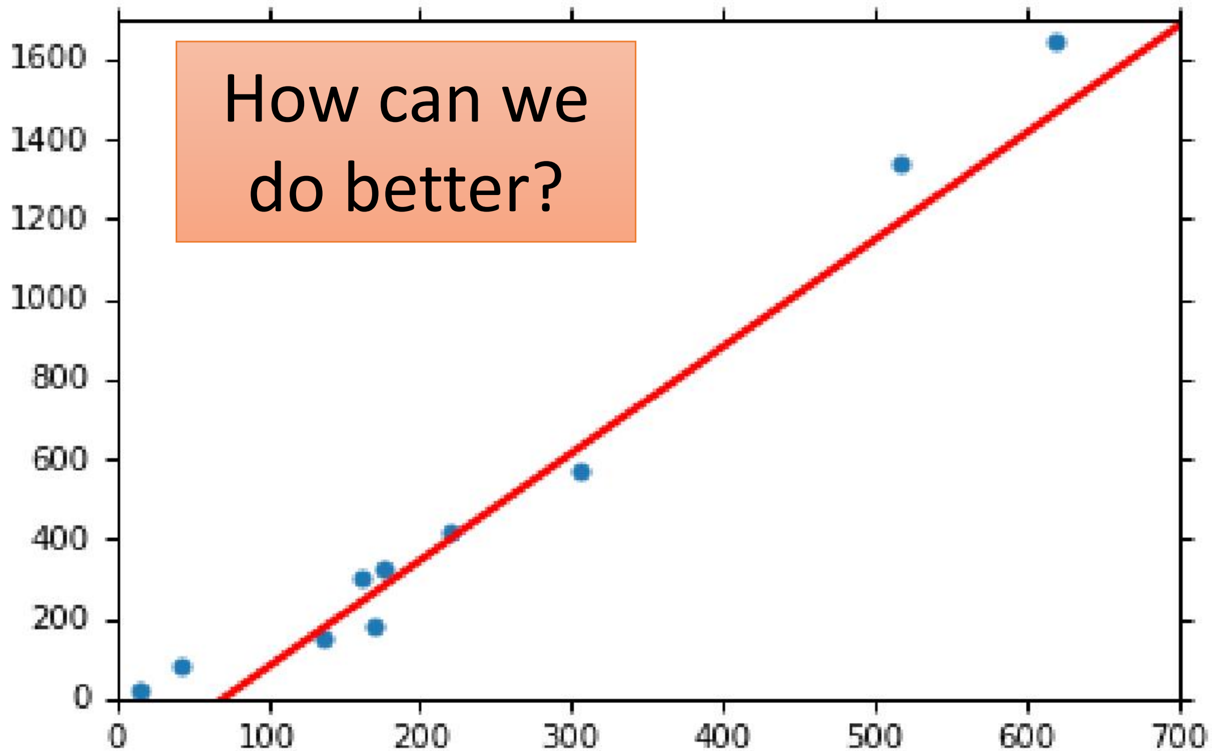
$w = 2.7$

Average Error on Testing Data

$$= \frac{1}{10} \sum_{n=1}^{10} e^n = 35.0$$

> Average Error on Training Data (31.9)

Another 10 data as testing data



多项式回归

■ **线性函数？二次多项式函数**

$$y = b + w_1 \cdot x + w_2 \cdot (x)^2$$

Best Function

$b = -10.3$

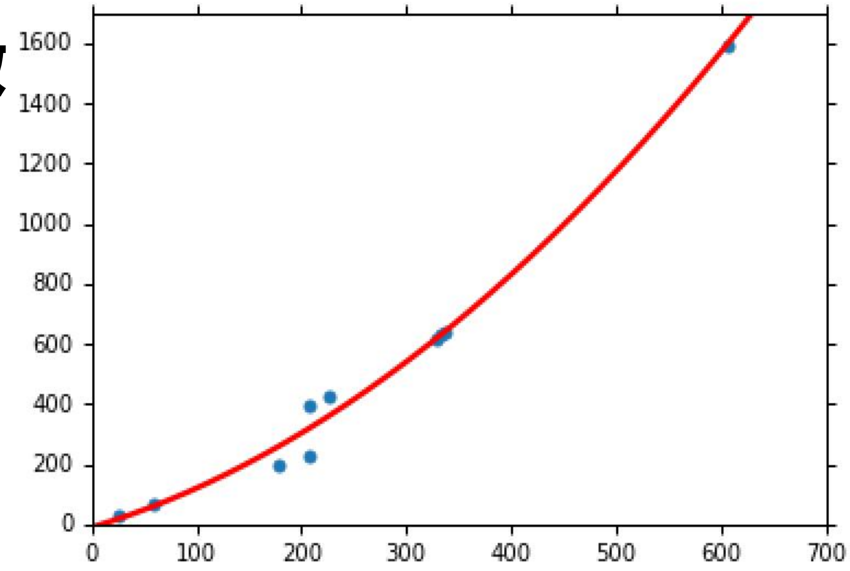
$w_1 = 1.0,\ w_2 = 2.7 \times 10^{-3}$

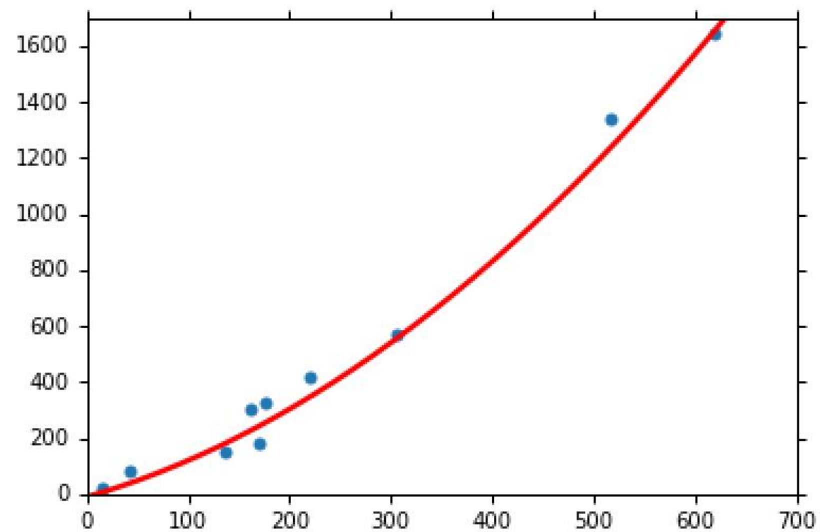
Average Error = 15.4

Testing:

Average Error = 18.4

Better! Could it be even better?





多项式回归

- **线性函数？三次多项式函数**

$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3$$

Best Function

$b = 6.4$, $w_1 = 0.66$

$w_2 = 4.3 \times 10^{-3}$

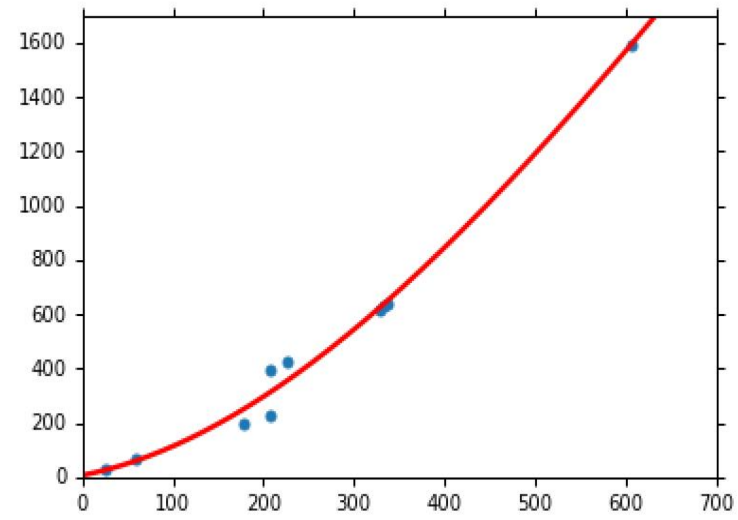
$w_3 = -1.8 \times 10^{-6}$

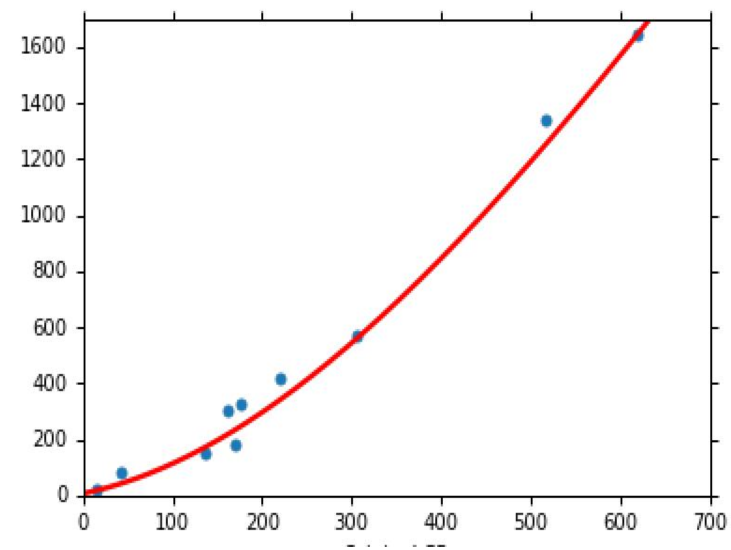
Average Error = 15.3

Testing:

Average Error = 18.1

Slightly better.
How about more complex model?





多项式回归

■ 线性函数？四次多项式函数

$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3 + w_4 \cdot (x)^4$$

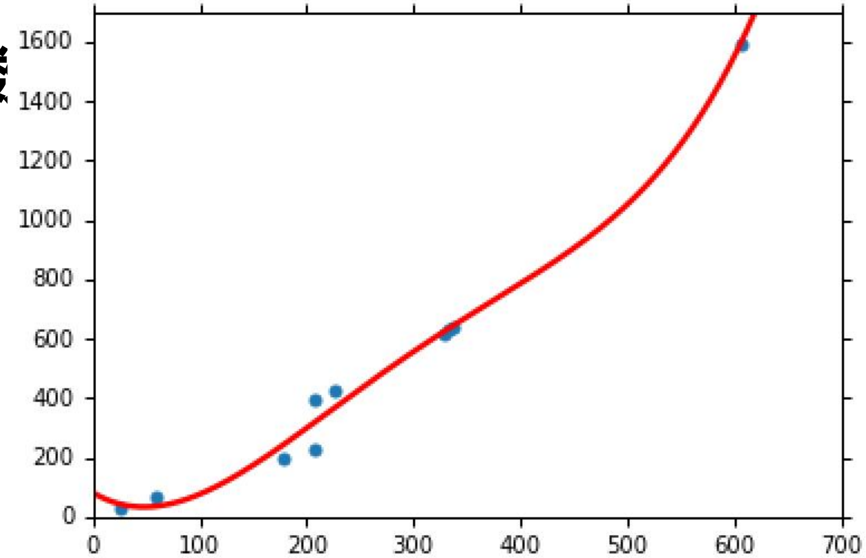
Best Function

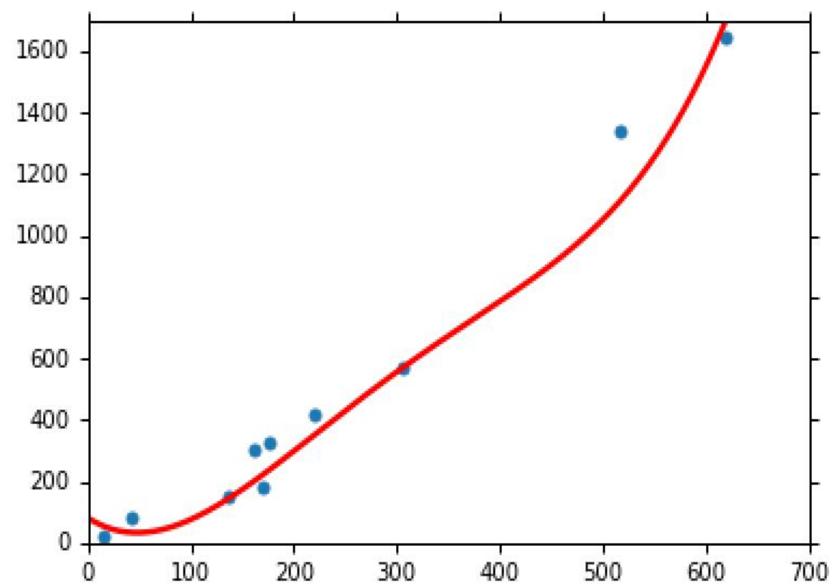
Average Error = 14.9

Testing:

Average Error = 28.8

The results become worse ...

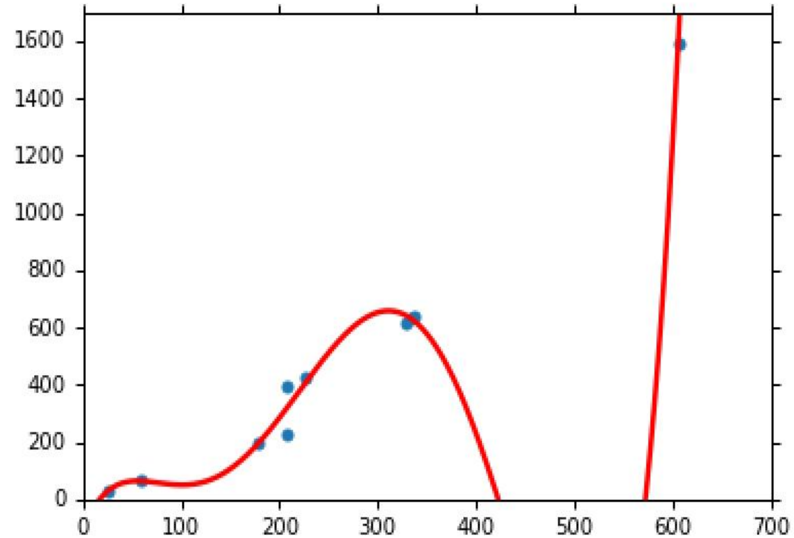




多项式回归

- **线性函数？五次多项式函数**

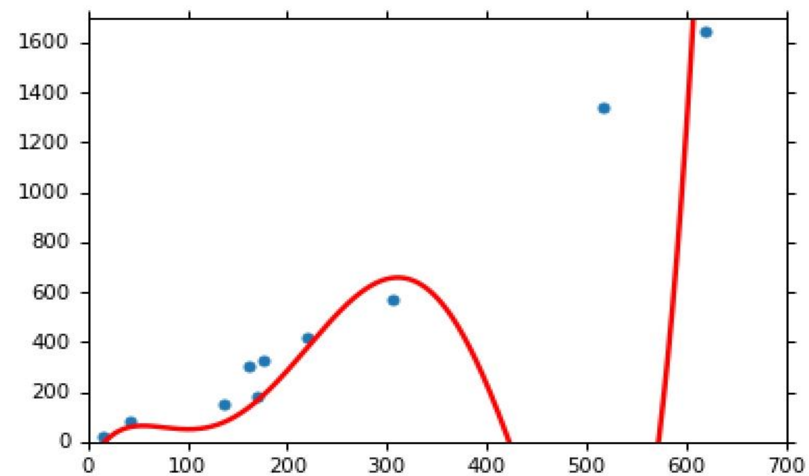
$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3 + w_4 \cdot (x)^4 + w_5 \cdot (x)^5$$



Best Function

Average Error = 12.8

Testing:



Average Error = 232.1

The results are so bad.

多项式回归

■ 模型选择

1. $y = b + w \cdot x_{cp}$

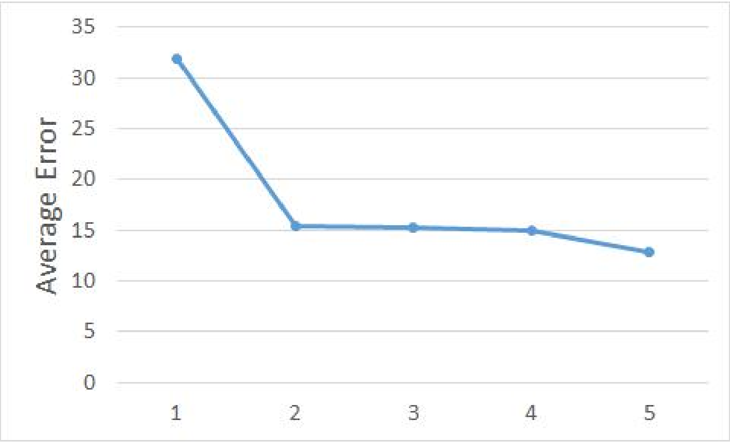
2. $y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2$

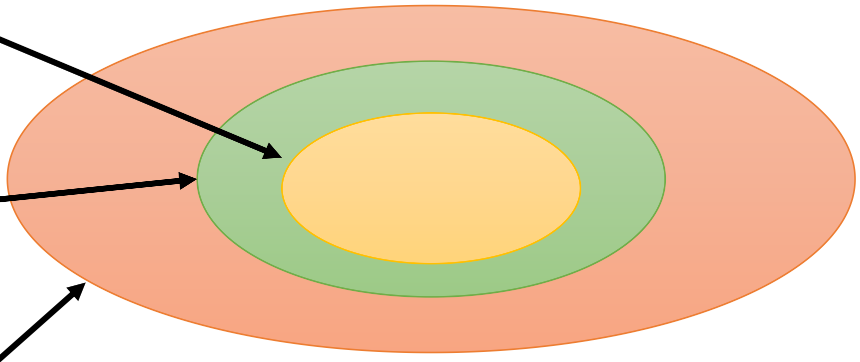
3. $y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3$

4. $y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3 + w_4 \cdot (x_{cp})^4$

5. $y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3 + w_4 \cdot (x_{cp})^4 + w_5 \cdot (x_{cp})^5$

Training Data





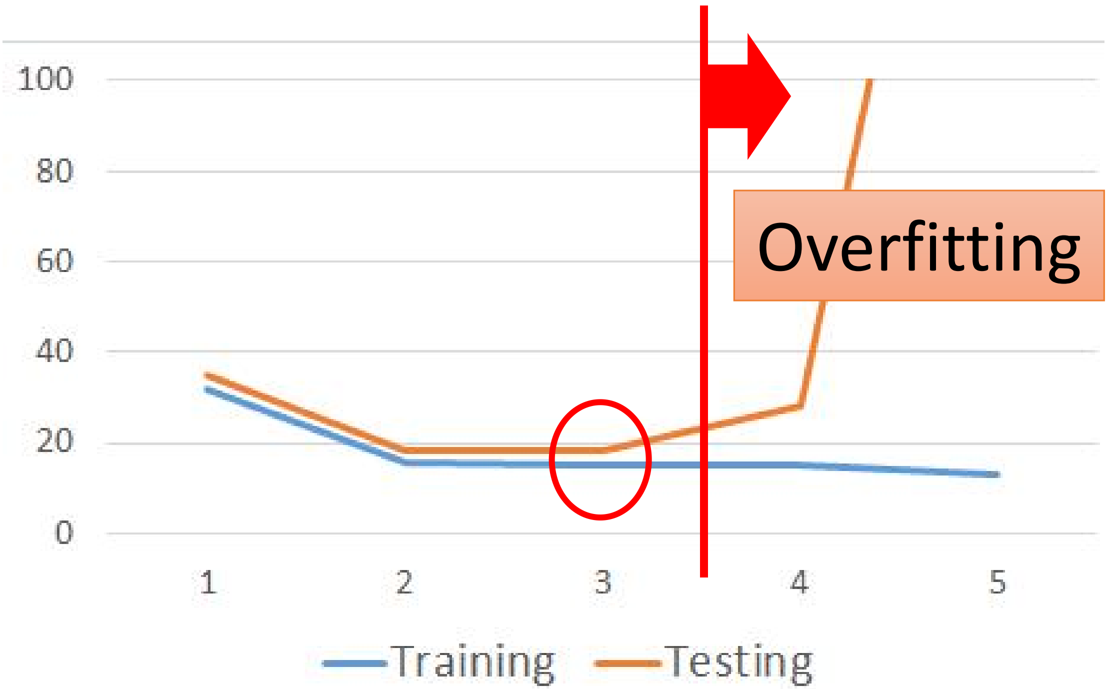
A more complex model yields lower error on training data.

If we can truly find the best function

参考自李宏毅老师课件

多项式回归

■ 模型选择



	Training	Testing
1	31.9	35.0
2	15.4	18.4
3	15.3	18.1
4	14.9	28.2
5	12.8	232.1

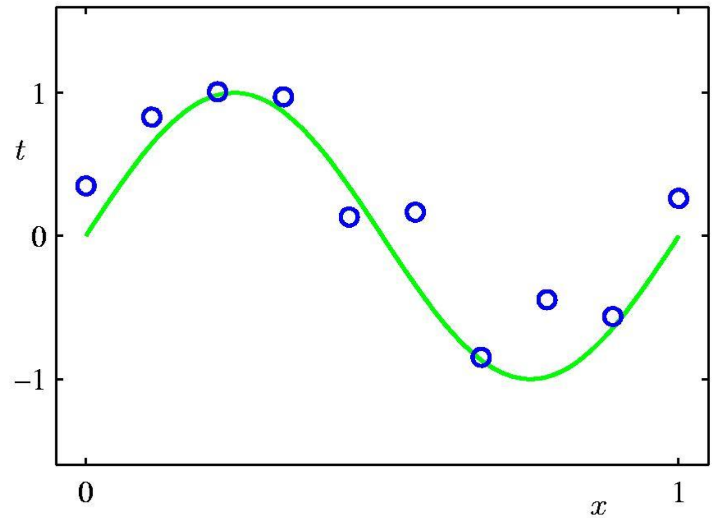
A more complex model does not always lead to better performance on ***testing data***.

This is ***Overfitting***. ➡ Select suitable model

参考自李宏毅老师课件

多项式回归

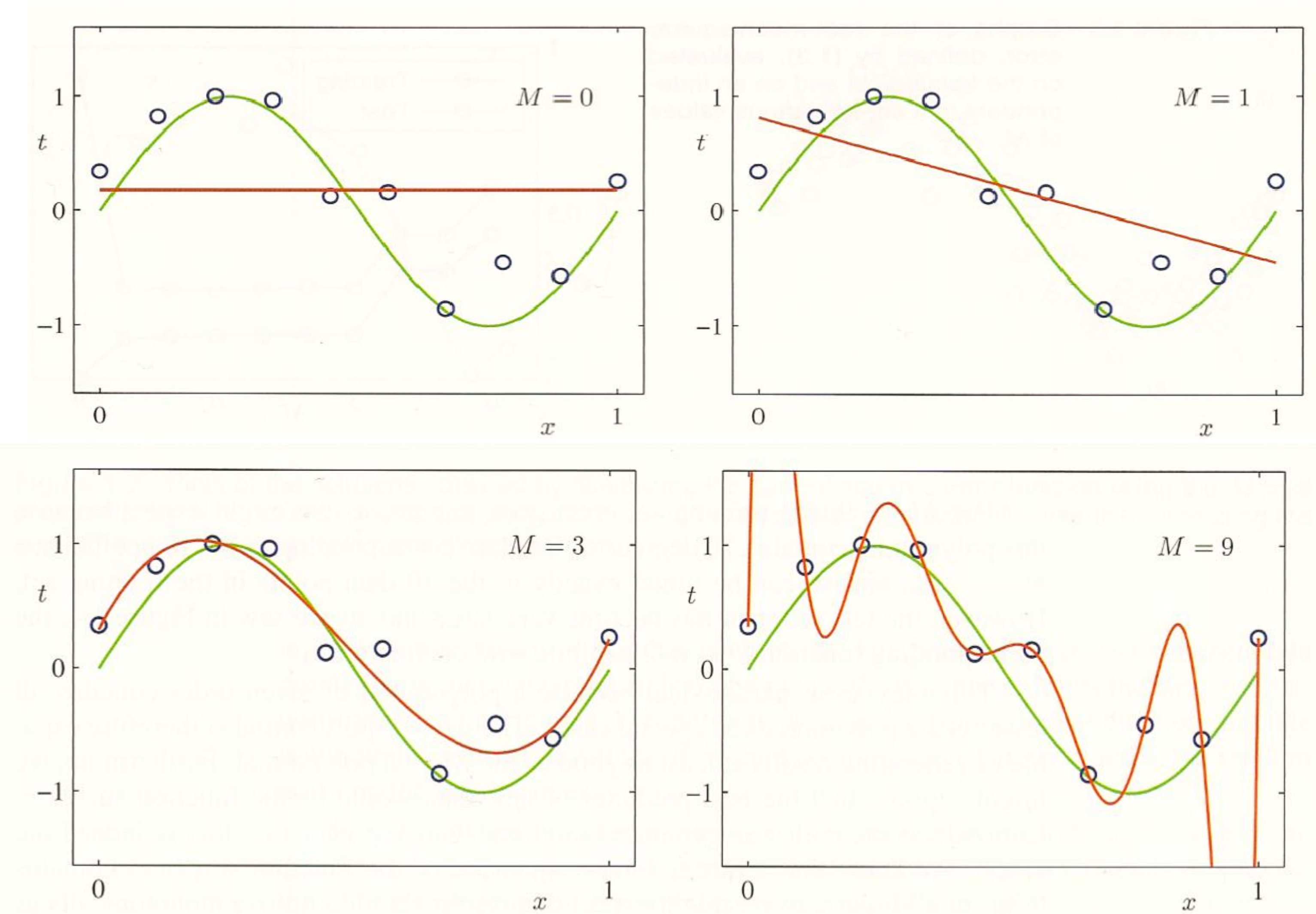
■ **实验练习（鼓励）**



$$f(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

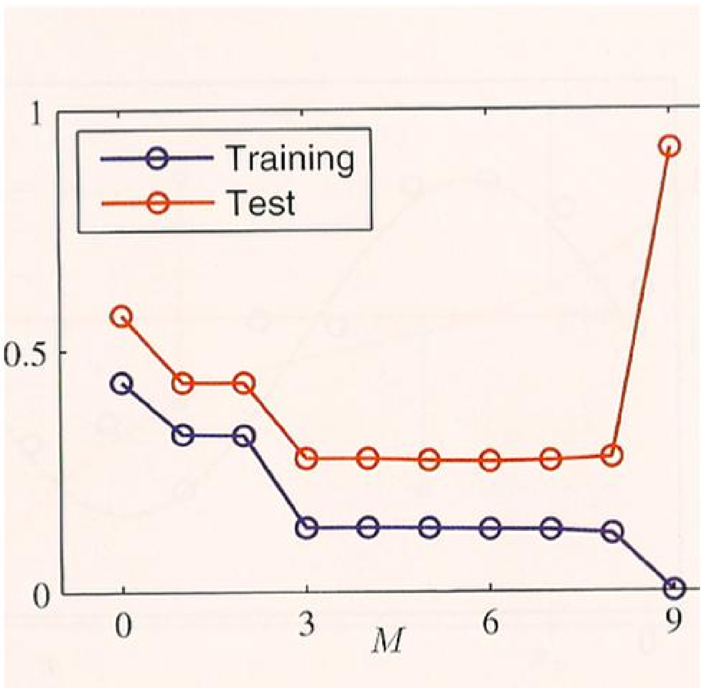
多项式回归

■ **实验练习（鼓励）**



多项式回归

■ 实验练习（鼓励）



	M = 0	M = 1	M = 3	M = 9
w_0	0.19	0.82	0.31	0.35
w_1		−1.27	7.99	232.37
w_2			−25.43	−5321.83
w_3			17.37	48568.31
w_4				−231639.30
w_5				640042.26
w_6				−1061800.52
w_7				1042400.18
w_8				−557682.99
w_9				125201.43

如何确定适当的M（多项式阶数）？如何防止过拟合?

多项式回归

■ 多项式回归--正则化

损失函数 $$E(w) = \frac{1}{2}\sum_{n=0}^{N-1}\{y_n - \boldsymbol{f}(x_n, w)\}^2$$

具有更小的w更好

带有正则化的回归 $$E(w) = \frac{1}{2}\sum_{n=0}^{N-1}(y_n - \boldsymbol{f}(x_n, w))^2 + \boxed{\frac{\lambda}{2}\|w\|^2}$$

- 更小的 w意味着... smoother

$$y = b + \sum w_i x_i$$

$$y + \sum w_i \Delta x_i = b + \sum w_i (x_i + \Delta x_i)$$

- 我们认为越平滑的函数效果越好。

多项式回归

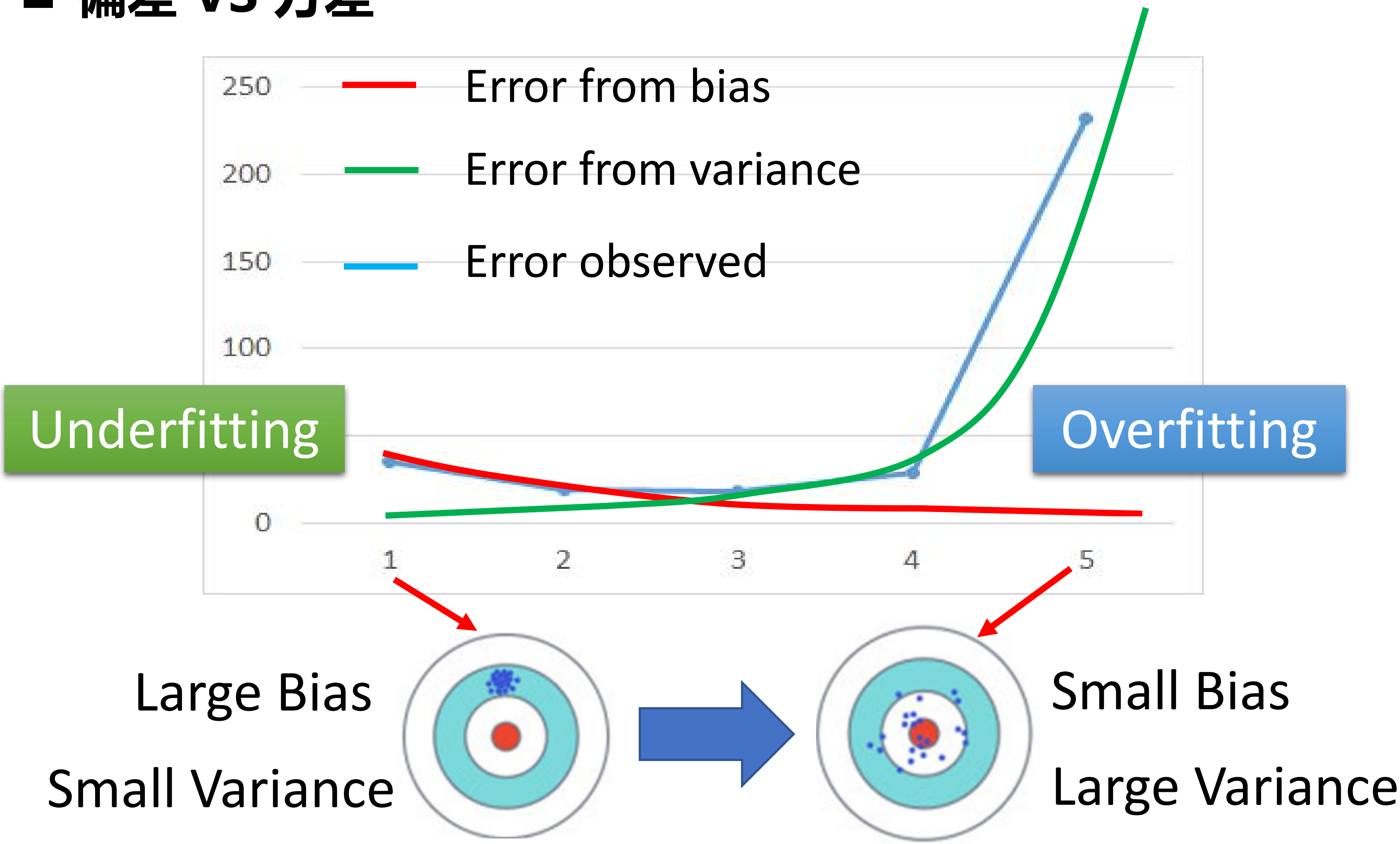
■ 多项式回归--正则化

$$E(w) = \frac{1}{2}\sum_{n=0}^{N-1}(y_n - \mathbf{f}(x_n, w))^2 + \frac{\lambda}{2}\|w\|^2$$

- 当w 非常小(趋近于0)时，输入变化对结果的影响趋近于0。
- 我们可以通过调整λ参数来调整正则化的强度。
- λ越大，模型曲线越平滑。
- 我们不需要对b进行正则化，因为b只影响模型曲线的上下移动(b是常数)，所以我们不需要对其进行调整。

多项式回归

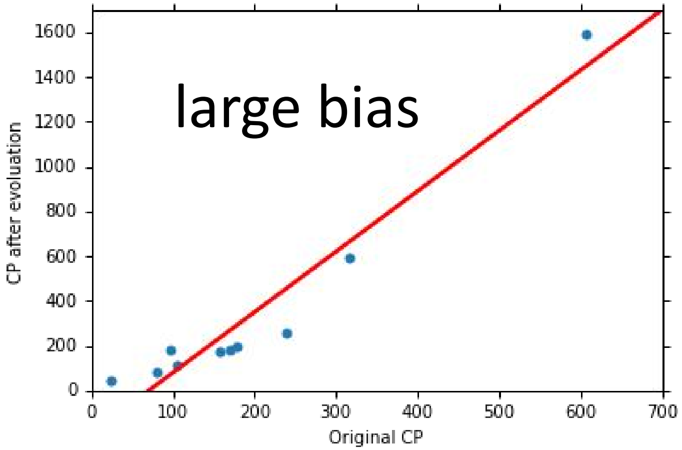
■ **偏差 VS 方差**



参考自李宏毅老师课件

多项式回归

- **怎样处理大的偏差（欠拟合）**
  - Diagnosis:
    - If your model cannot even fit the training examples, then you have large bias Underfitting
    - If you can fit the training data, but large error on testing data, then you probably have large variance Overfitting
  - For bias, redesign your model:
    - Add more features as input
    - A more complex model

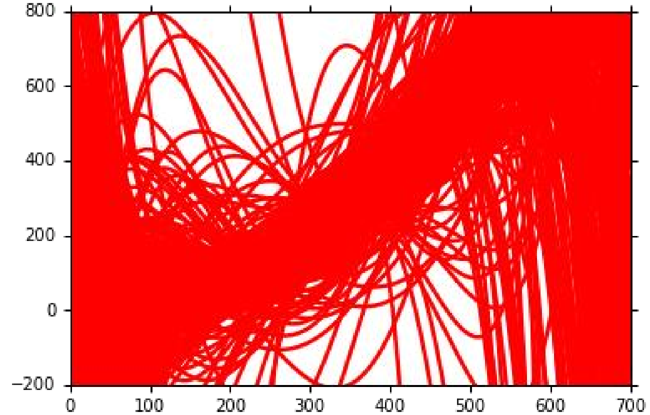


多项式回归

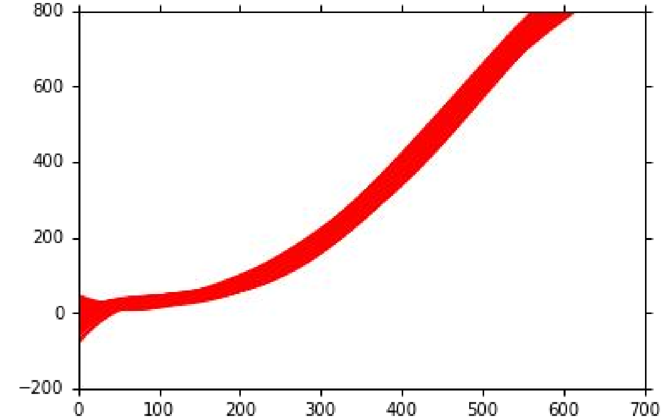
■ 怎样处理大的方差（过拟合）

- More data

Very effective, but not always practical

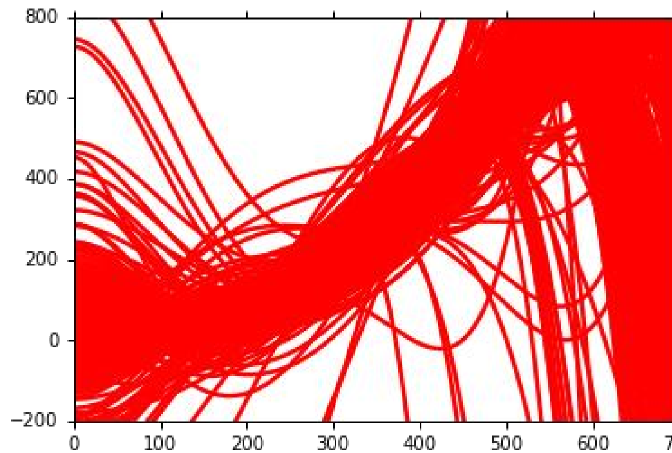


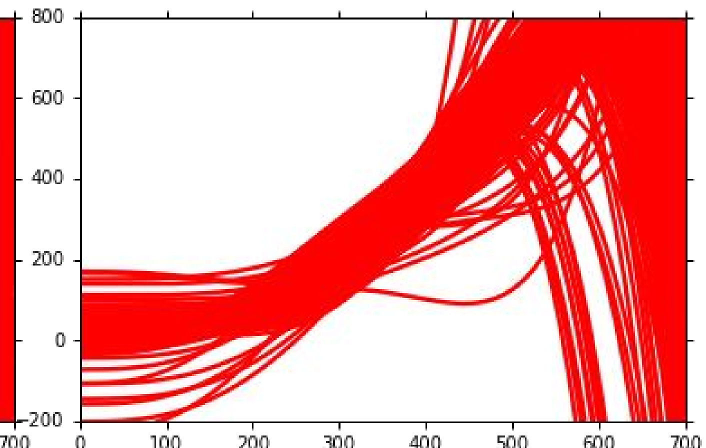
10 examples



100 examples

- Regularization → May increase bias

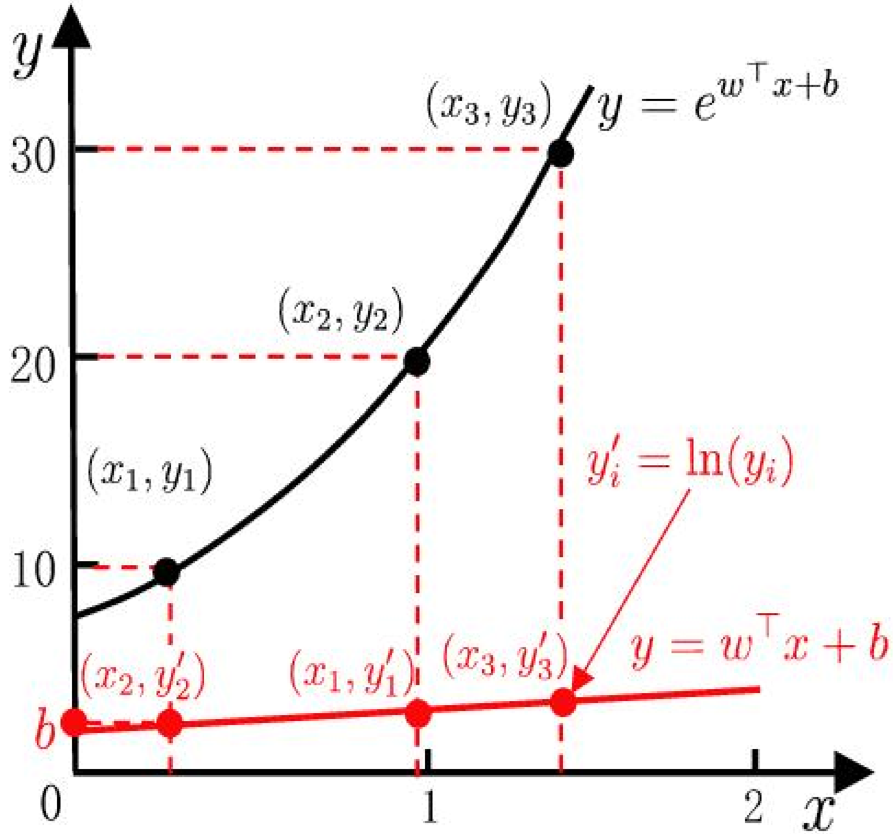




其他回归方法

■ 对数线性模型

输出标记的对数为线性模型逼近的目标



$$\ln y = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$$

$\Updownarrow$

$$y = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$$

其他回归方法

- **广义线性模型**
  - 一般形式

$$y = g^{-1}\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b\right)$$

  - $g\left(\cdot\right)$ 称为联系函数（link function）
    - 单调可微函数

  - 对数线性回归是 $g\left(\cdot\right) = \ln\left(\cdot\right)$ 时广义线性模型的特例

Thanks