

## 《神经网络与深度学习》



### 机器学习概述

<https://nndl.github.io/>

# 教学内容

---

- ▶ 机器学习
  - ▶ 概念
  - ▶ 原理
  - ▶ 线性回归
    - ▶ 定义
    - ▶ 经验风险最小化
      - ▶ 最小均方误差
    - ▶ 结构风险最小化
    - ▶ 最大似然估计
    - ▶ 最大后验估计
  - ▶ 机器学习的损失从哪里来
-

# 机器学习 ≈ 构建一个映射函数

---

► 语音识别

$$f(\text{[声波图]}) = \text{“你好”}$$

► 图像识别

$$f(\text{[猫的图片]}) = \text{“猫”}$$

► 围棋

$$f(\text{[围棋棋盘]}) = \text{“6-5”} \quad (\text{落子位置})$$

► 对话系统

$$f(\text{“你好”}) = \text{“今天天气真不错”}$$

用户输入

机器

# 为什么要“机器学习”？

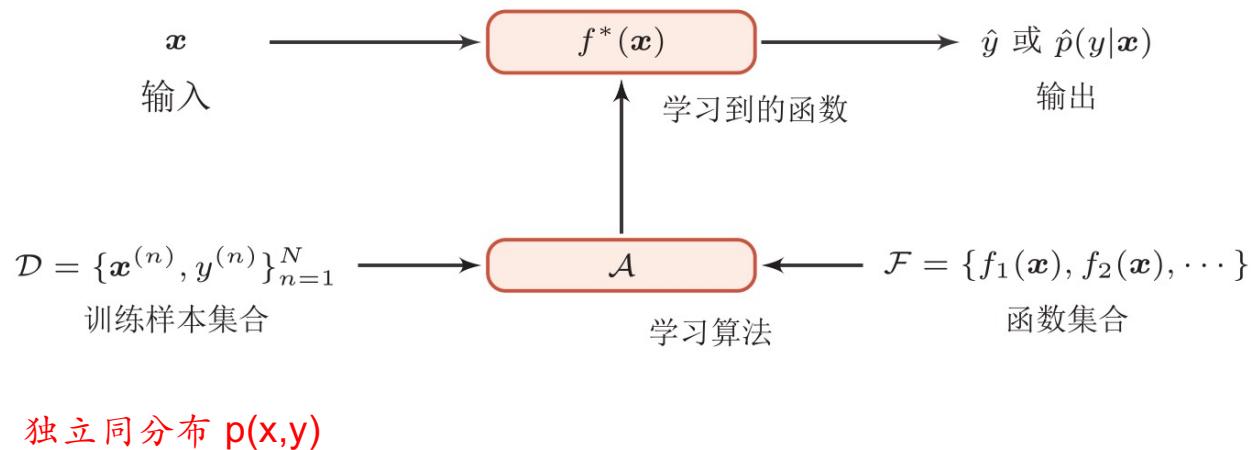
- ▶ 现实世界的问题都比较复杂
- ▶ 很难通过规则来手工实现



2	6	8	9	3	4	7	5	6
3	4	7	9	5	5	6	7	2
5	8	7	0	9	4	3	5	4
5	2	3	4	9	5	6	7	8

# 什么是机器学习？

- ▶ 机器学习：通过算法使得机器能从大量数据中学习规律从而对新的样本做决策。
- ▶ 规律：决策（预测）函数



# 机器学习的三要素

---

## ►模型

►线性方法:  $f(\mathbf{x}, \theta) = \mathbf{w}^T \mathbf{x} + b$

►非线性方法:  $f(\mathbf{x}, \theta) = \mathbf{w}^T \phi(\mathbf{x}) + b$

►如果  $\phi(\mathbf{x})$  为可学习的非线性基函数,  $f(\mathbf{x}, \theta)$  就等价于神经网络。

## ►学习准则

### ►期望风险

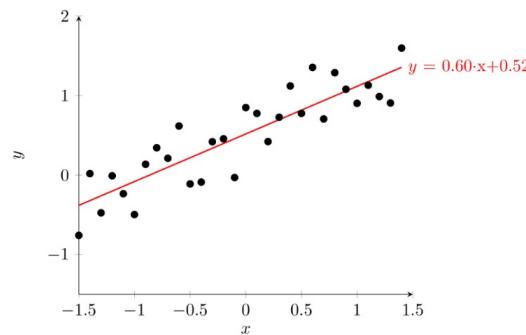
$$\mathcal{R}(f) = \mathbb{E}_{(\mathbf{x}, y) \sim p(\mathbf{x}, y)} [\mathcal{L}(f(\mathbf{x}), y)],$$

## ►优化

### ►梯度下降

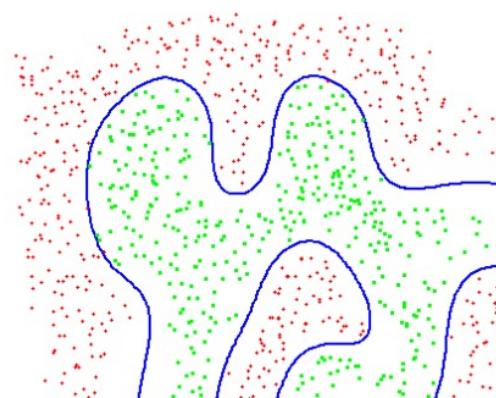
# 常见的机器学习问题

---

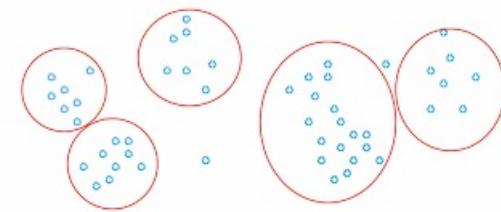


回归

---



分类

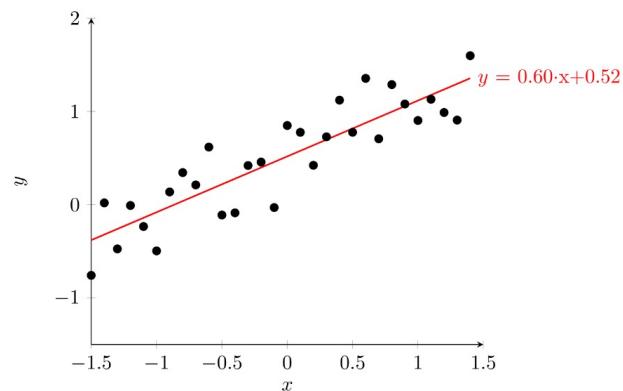


聚类

# 模型

- ▶ 以线性回归 (Linear Regression) 为例
- ▶ 模型：

$$f(\mathbf{x}; \mathbf{w}, b) = \mathbf{w}^T \mathbf{x} + b$$



# 学习准则

---

## ► 损失函数

### ► 0-1损失函数

$$\mathcal{L}(y, f(x, \theta)) = \begin{cases} 0 & \text{if } y = f(x, \theta) \\ 1 & \text{if } y \neq f(x, \theta) \end{cases}$$

### ► 平方损失函数

$$\mathcal{L}(y, \hat{y}) = (y - f(x, \theta))^2$$

## 学习准则

---

► 期望风险未知，通过经验风险近似

► 训练数据： $\mathcal{D} = \{x^{(n)}, y^{(n)}\}, i \in [1, N]$

$$\mathcal{R}_{\mathcal{D}}^{emp}(\theta) = \frac{1}{N} \sum_{n=1}^N \mathcal{L}(y^{(n)}, f(x^{(n)}, \theta))$$

► 经验风险最小化

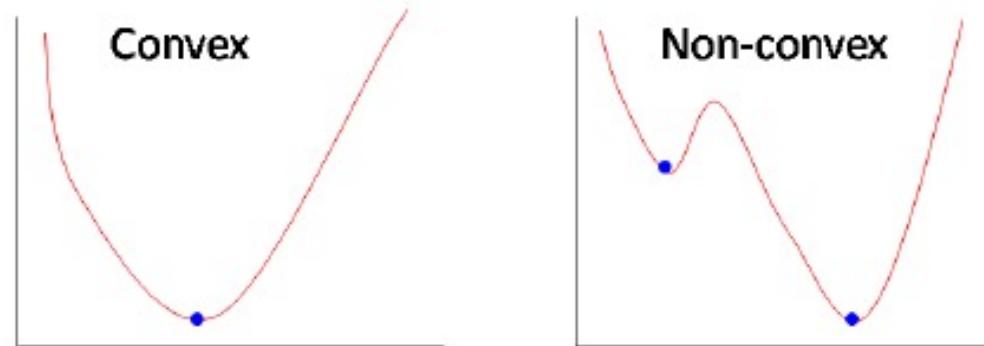
► 在选择合适的风险函数后，我们寻找一个参数 $\theta^*$ ，使得经验风险函数最小化。

$$\theta^* = \arg \min_{\theta} \mathcal{R}_{\mathcal{D}}^{emp}(\theta)$$

► 机器学习问题转化成为一个最优化问题

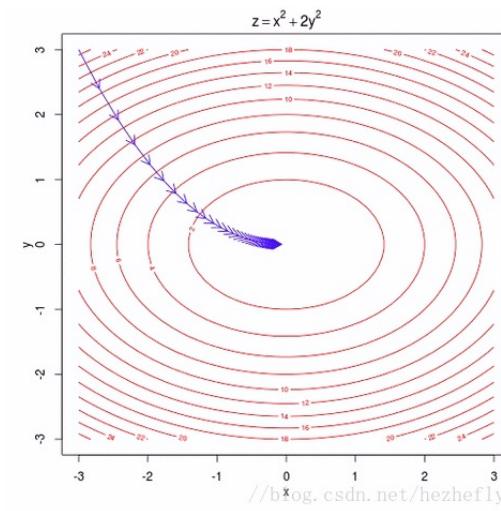
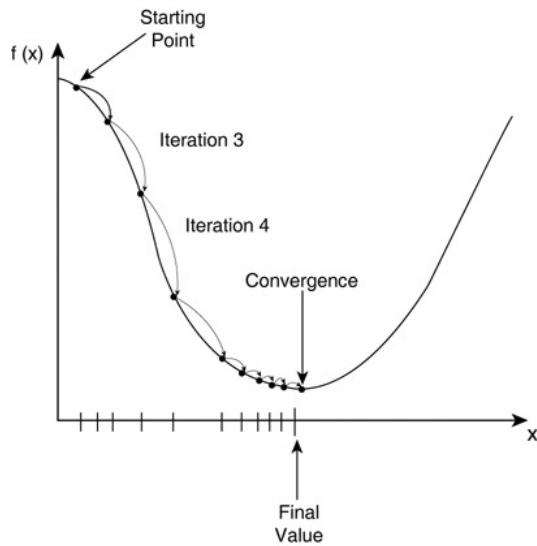
# 最优化问题

► 机器学习问题转化成为一个最优化问题



$$\min_{\mathbf{x}} f(\mathbf{x})$$

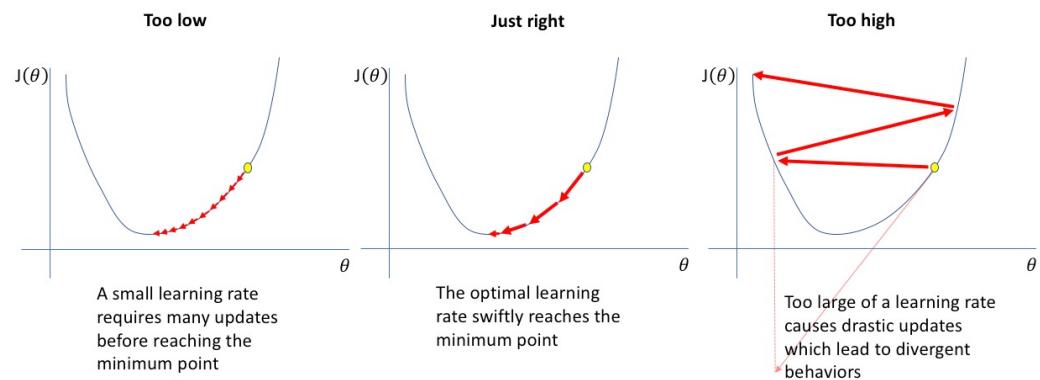
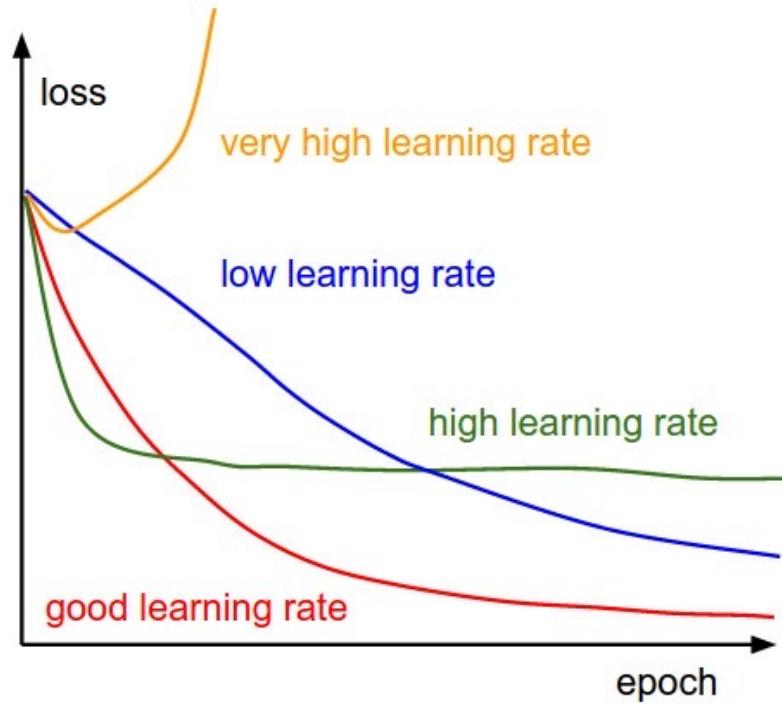
# 梯度下降法 ( Gradient Descent )



$$\begin{aligned}\theta_{t+1} &= \theta_t - \alpha \frac{\partial \mathcal{R}(\theta)}{\partial \theta} \\ &= \theta_t - \alpha \frac{1}{N} \sum_{n=1}^N \frac{\partial \mathcal{L}(y^{(n)}, f(x^{(n)}; \theta))}{\partial \theta}\end{aligned}$$

搜索步长 $\alpha$ 中也叫作学习率 (Learning Rate)

# 学习率是十分重要的超参数！



## 随机梯度下降法

- ▶ 随机梯度下降法 (Stochastic Gradient Descent, SGD) 也叫增量梯度下降，每个样本都进行更新

$$\theta_{t+1} = \theta_t - \alpha \frac{\partial \mathcal{L}(\theta_t; x^{(t)}, y^{(t)})}{\partial \theta},$$

- ▶ 小批量 (Mini-Batch) 随机梯度下降法

$$\theta_{t+1} = \theta_t - \alpha \frac{1}{K} \sum_{(x,y) \in S_t} \frac{\partial L(y, f(x; \theta))}{\partial \theta}$$

# 随机梯度下降法

---

## 算法 2.1: 随机梯度下降法

---

输入: 训练集  $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$ , 验证集  $\mathcal{V}$ , 学习率  $\alpha$

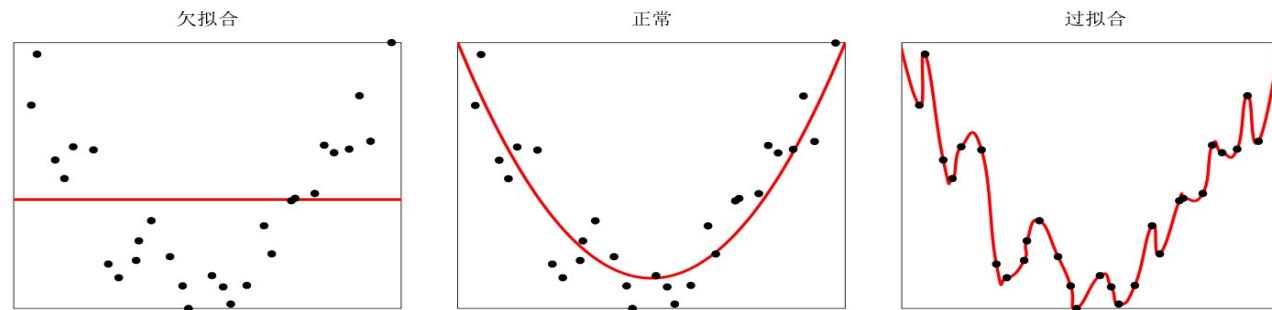
```
1 随机初始化  $\theta$ ;  
2 repeat  
3   对训练集  $\mathcal{D}$  中的样本随机重排序;  
4   for  $n = 1 \dots N$  do  
5     从训练集  $\mathcal{D}$  中选取样本  $(\mathbf{x}^{(n)}, y^{(n)})$ ;  
6     // 更新参数  
7      $\theta \leftarrow \theta - \alpha \frac{\partial \mathcal{L}(\theta; \mathbf{x}^{(n)}, y^{(n)})}{\partial \theta}$ ;  
8   end  
9 until 模型  $f(\mathbf{x}; \theta)$  在验证集  $\mathcal{V}$  上的错误率不再下降;  
输出:  $\theta$ 
```

---



# 机器学习 = 优化？

机器学习 = 优化? NO!



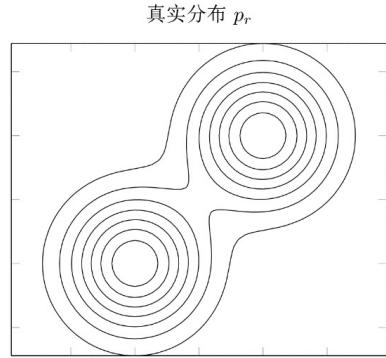
过拟合：经验风险最小化原则很容易导致模型在训练集上错误率很低，但是在未知数据上错误率很高。

过拟合问题往往是由于训练数据少和噪声等原因造成的。

# 泛化错误

期望风险

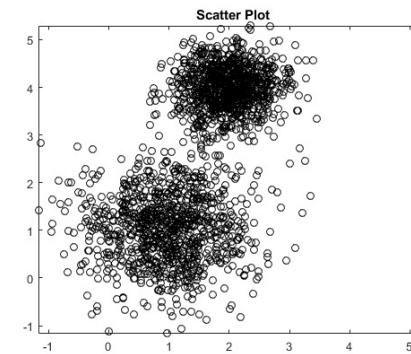
$$\mathcal{R}(f) = \mathbb{E}_{(\mathbf{x}, y) \sim p(\mathbf{x}, y)} [\mathcal{L}(f(\mathbf{x}), y)],$$



≠

经验风险

$$\mathcal{R}_{\mathcal{D}}^{emp}(\theta) = \frac{1}{N} \sum_{n=1}^N \mathcal{L}(y^{(n)}, f(x^{(n)}, \theta))$$



$$\mathcal{G}_{\mathcal{D}}(f) = \mathcal{R}(f) - \mathcal{R}_{\mathcal{D}}^{emp}(f)$$

泛化错误

# 如何减少泛化错误？

---

优化

经验风险最小

正则化

降低模型复杂度



# 正则化 ( regularization )

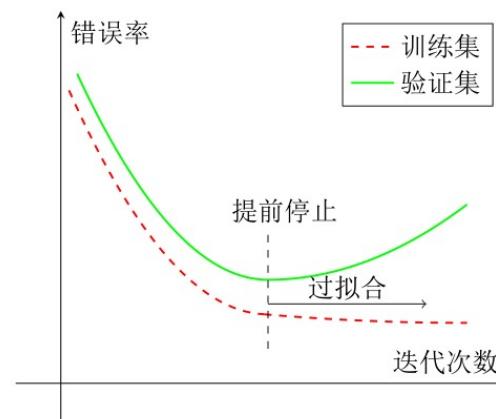
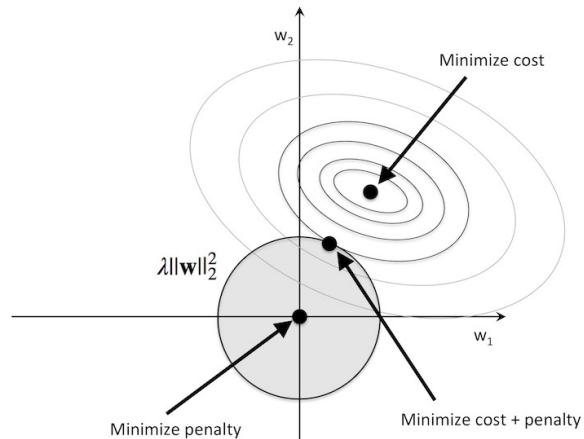
所有损害优化的方法都是正则化。

增加优化约束

L1/L2约束、数据增强

干扰优化过程

权重衰减、随机梯度下降、提前停止





线性回归

# 线性回归 ( Linear Regression )

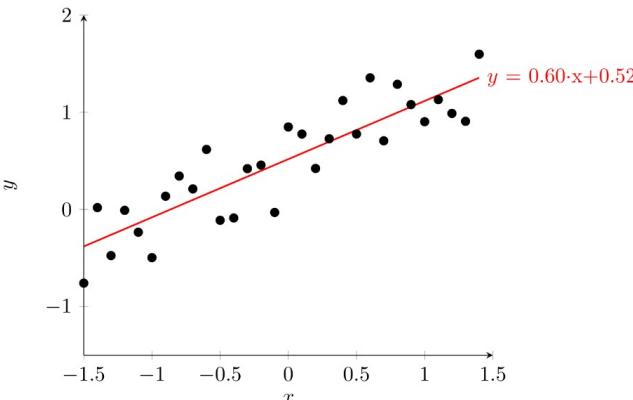
► 模型：

$$f(\mathbf{x}; \mathbf{w}, b) = \mathbf{w}^T \mathbf{x} + b$$

► 增广权重向量和增广特征向量

$$\hat{\mathbf{x}} = \mathbf{x} \oplus 1 \triangleq \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ 1 \end{bmatrix},$$

$$\hat{\mathbf{w}} = \mathbf{w} \oplus b \triangleq \begin{bmatrix} \mathbf{w} \\ b \end{bmatrix} = \begin{bmatrix} w_1 \\ \vdots \\ w_k \\ b \end{bmatrix},$$



$$f(\mathbf{x}; \hat{\mathbf{w}}) = \hat{\mathbf{w}}^T \hat{\mathbf{x}},$$

## 优化方法

---

- ▶ 经验风险最小化（最小二乘法）
- ▶ 结构风险最小化（岭回归）
- ▶ 最大似然估计
- ▶ 最大后验估计



# 矩阵微积分

---

## ► 标量关于向量的偏导数

$$\frac{\partial y}{\partial \mathbf{x}} = \left[ \frac{\partial y}{\partial x_1}, \dots, \frac{\partial y}{\partial x_M} \right]^\top$$

## ► 向量关于向量的偏导数

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \dots & \frac{\partial y_N}{\partial x_1} \\ \vdots & \vdots & \vdots \\ \frac{\partial y_1}{\partial x_M} & \dots & \frac{\partial y_N}{\partial x_M} \end{bmatrix}$$

## ► 向量函数及其导数

$$\frac{\partial \mathbf{x}}{\partial \mathbf{x}} = \mathbf{I},$$

$$\frac{\partial \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \mathbf{A}^\top,$$

$$\frac{\partial \mathbf{x}^\top \mathbf{A}}{\partial \mathbf{x}} = \mathbf{A}$$

# 经验风险最小化

---

## ►模型

$$f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \mathbf{x}$$

## ►学习准则

$$\begin{aligned}\mathcal{R}(\mathbf{w}) &= \sum_{n=1}^N \mathcal{L}(y^{(n)}, f(\mathbf{x}^{(n)}; \mathbf{w})) \\ &= \frac{1}{2} \sum_{n=1}^N \left( y^{(n)} - \mathbf{w}^T \mathbf{x}^{(n)} \right)^2 \\ &= \frac{1}{2} \|\mathbf{y} - \mathbf{X}^T \mathbf{w}\|^2,\end{aligned}$$

# 经验风险最小化

---

## ► 优化

$$\frac{\partial}{\partial \mathbf{w}} \mathcal{R}(\mathbf{w}) = 0$$

$$\begin{aligned}\frac{\partial \mathcal{R}(\mathbf{w})}{\partial \mathbf{w}} &= \frac{1}{2} \frac{\partial \| \mathbf{y} - \mathbf{X}^T \mathbf{w} \|^2}{\partial \mathbf{w}} \\ &= -\mathbf{X}(\mathbf{y} - \mathbf{X}^T \mathbf{w}),\end{aligned}$$

$$\begin{aligned}-\mathbf{X}(\mathbf{y} - \mathbf{X}^T \mathbf{w}) &= 0 \\ \mathbf{X} \mathbf{X}^T \mathbf{w} &= \mathbf{X} \mathbf{y} \\ \mathbf{w} &= (\mathbf{X} \mathbf{X}^T)^{-1} \mathbf{X} \mathbf{y}\end{aligned}$$

## 最小二乘估计

$$\begin{aligned} Q(\beta_0, \beta_1) &= \sum_{i=1}^n (y_i - E(y_i))^2 \\ &= \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2 \end{aligned}$$

- 要求  $Q(\beta_0, \beta_1)$  的最小值，令其一阶导数为零，即

$$\begin{cases} \frac{\partial Q}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) = 0 \\ \frac{\partial Q}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) x_i = 0 \end{cases}$$

- 经整理后，得到正规方程组

$$\begin{cases} n\beta_0 + n\bar{x}\beta_1 = n\bar{y} \\ n\bar{x}\beta_0 + \sum_{i=1}^n x_i^2 \beta_1 = \sum_{i=1}^n x_i y_i \end{cases}$$

- 于是， $\beta_0, \beta_1$  的最小二乘估计为

$$\begin{cases} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}, \\ \hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}. \end{cases}$$

- 其中，

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

分别为  $x_1, x_2, \dots, x_n$  和  $y_1, y_2, \dots, y_n$  的样本均值。

- 根据一阶导等于零，所求的  $\hat{\beta}_0, \hat{\beta}_1$  实际上是  $Q(\beta_0, \beta_1)$  的稳定点。
- 但是否为最小值点，仍需要根据其二阶导在  $(\hat{\beta}_0, \hat{\beta}_1)$  上的表现来判断是否为最小值点。
- 对  $Q(\beta_0, \beta_1)$  求二阶偏导，我们有

$$\begin{aligned} \left| \begin{pmatrix} 2n & 2n\bar{x} \\ 2n\bar{x} & 2\sum_{i=1}^n x_i^2 \end{pmatrix} \right| &= 4n \sum_{i=1}^n x_i^2 - 4n^2 (\bar{x})^2 \\ &= 4n \sum_{i=1}^n (x_i - \bar{x})^2 \\ &> 0. \end{aligned}$$

## 结构风险最小化

---

► 结构风险最小化准则

$$\mathcal{R}(\mathbf{w}) = \frac{1}{2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2 + \frac{1}{2} \lambda \|\mathbf{w}\|^2,$$

► 得到

$$\mathbf{w}^* = (\mathbf{X}\mathbf{X}^\top + \lambda I)^{-1} \mathbf{X}\mathbf{y},$$

► 岭回归 ( Ridge Regression )



# 关于概率的一些基本概念

---

## ► 概率 (Probability)

► 一个随机事件发生的可能性大小，为0到1之间的实数。

## ► 随机变量 (Random Variable)

► 比如随机掷一个骰子，得到的点数就可以看成一个随机变量X，其取值为 $\{1,2,3,4,5,6\}$ 。

## ► 概率分布 (Probability Distribution)

► 一个随机变量X取每种可能值的概率

$$P(X = x_i) = p(x_i), \quad \forall i \in \{1, \dots, n\}.$$

► 并满足

$$\sum_{i=1}^n p(x_i) = 1,$$

$$p(x_i) \geq 0, \quad \forall i \in \{1, \dots, n\}.$$

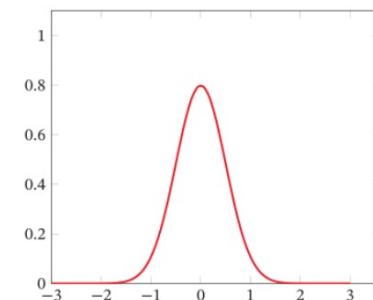
## 概率的一些基本概念

- ▶ 连续随机变量  $Y$  的概率分布一般用概率密度函数（Probability Density Function，PDF） $p(x)$  来描述。

$$\int_{-\infty}^{+\infty} p(x)dx = 1$$

- ▶ 高斯分布 (Gaussian Distribution)

$$X \sim \mathcal{N}(\mu, \sigma^2) \quad p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$



# 概率的一些基本概念

---

## ► 条件概率 (Conditional Probability)

► 对于离散随机向量(X,Y)，已知X = x的条件下，随机变量Y = y的条件概率为：

$$p(y|x) = P(Y = y|X = x) = \frac{p(x, y)}{p(x)}$$

## ► 贝叶斯公式

► 两个条件概率  $p(y|x)$  和  $p(x|y)$  之间的关系

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

## 似然 ( Likelihood )

► 似然函数是关于模型  $p(x;w)$  的参数  $w$  的函数

贝叶斯公式: 
$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}$$

$$p(w|X) \propto p(X|w)p(w)$$

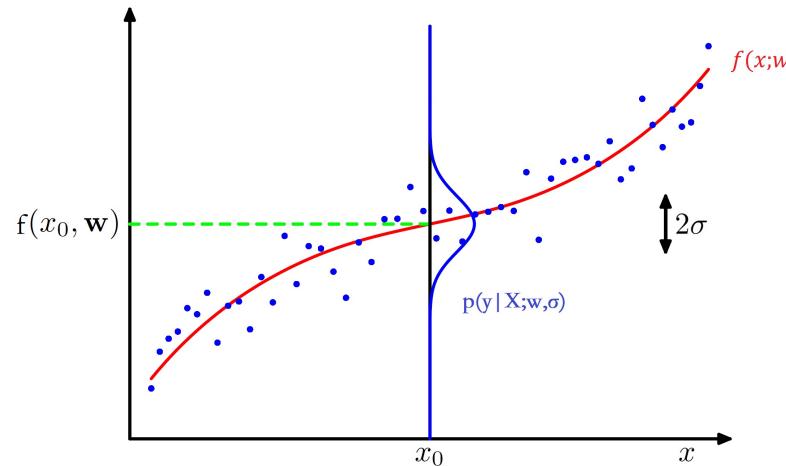
后验              似然              先验  
posterior        likelihood        prior

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

## 从概率角度来看线性回归

►假设标签 $y$ 为一个随机变量，其服从以均值为 $f(x; w) = w^T x$ ，方差为 $\sigma^2$ 的高斯分布。

$$\begin{aligned} p(y|\mathbf{x}; \mathbf{w}, \sigma) &= \mathcal{N}(y; \mathbf{w}^T \mathbf{x}, \sigma^2) \\ &= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y - \mathbf{w}^T \mathbf{x})^2}{2\sigma^2}\right). \end{aligned}$$



## 线性回归中的似然函数

---

►参数w在训练集D上的似然函数 (Likelihood) 为

$$\begin{aligned} p(\mathbf{y}|X; \mathbf{w}, \sigma) &= \prod_{n=1}^N p(y^{(n)}|\mathbf{x}^{(n)}; \mathbf{w}, \sigma) \\ &= \prod_{n=1}^N \mathcal{N}(y^{(n)}; \mathbf{w}^T \mathbf{x}^{(n)}, \sigma^2) \end{aligned}$$

## 最大似然估计

- ▶ 最大似然估计 (Maximum Likelihood Estimate, MLE)
- ▶ 是指找到一组参数  $w$  使得似然函数  $p(y|X; w, \sigma)$  最大

$$\text{令 } \frac{\partial \log p(\mathbf{y}|X; \mathbf{w}, \sigma)}{\partial \mathbf{w}} = 0$$



$$\mathbf{w}^{ML} = (X X^T)^{-1} X \mathbf{y}.$$

# 最大似然估计

## 分布假定

- 在一元线性回归模型中，最常见的假定为  $\varepsilon$  服从正态分布，即

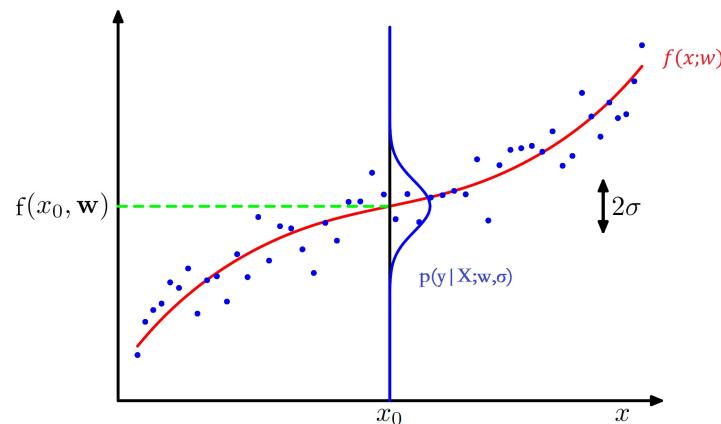
$$\varepsilon \sim N(0, \sigma^2)$$

- 从数据的角度来看，由于  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$  都是与  $\varepsilon$  独立同分布的随机变量，因而有

$$\varepsilon_i \stackrel{\text{i.i.d}}{\sim} N(0, \sigma^2), \quad i = 1, 2, \dots, n$$

- 在  $\varepsilon_i$  服从正态分布的假定下， $y_i$  也服从正态分布，即

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2), \quad i = 1, 2, \dots, n.$$



## 解法

- $y_i$  的密度函数为

$$f(y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}[y_i - (\beta_0 + \beta_1 x_i)]^2\right\}.$$

- 因为  $y_1, y_2, \dots, y_n$  的密度函数的形式是不尽相同的，我们用  $f_i(y_i)$  代替  $f(y_i)$  更为合适。
- 似然函数为

$$\begin{aligned} L(\beta_0, \beta_1, \sigma^2) &= \prod_{i=1}^n f_i(y_i) \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left\{-\frac{1}{2\sigma^2} \sum_{i=1}^n [y_i - (\beta_0 + \beta_1 x_i)]^2\right\} \end{aligned}$$

- 易知， $L$  的最大值点与  $\ln L$  的最大值点是相同的。
- 于是，对数似然函数为

$$\ln L(\beta_0, \beta_1, \sigma^2) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2$$



最大后验估计

# 最大后验估计

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}; \nu, \sigma) = \frac{p(\mathbf{w}, \mathbf{y}|\mathbf{X}; \nu, \sigma)}{\sum_{\mathbf{w}} p(\mathbf{w}, \mathbf{y}|\mathbf{X}; \nu, \sigma)}$$
$$\propto \underbrace{p(\mathbf{y}|\mathbf{X}, \mathbf{w}; \sigma)}_{\text{似然 likelihood}} p(\mathbf{w}; \nu),$$
$$p(\mathbf{w}; \nu) = \mathcal{N}(\mathbf{w}; \mathbf{0}, \nu^2 I)$$

后验 posterior      似然 likelihood      先验 prior

$$\log p(\mathbf{w}|\mathbf{X}, \mathbf{y}; \nu, \sigma) \propto \log p(\mathbf{y}|\mathbf{X}, \mathbf{w}; \sigma) + \log p(\mathbf{w}; \nu)$$

$$\propto -\frac{1}{2\sigma^2} \sum_{n=1}^N \left( y^{(n)} - \mathbf{w}^\top \mathbf{x}^{(n)} \right)^2 - \frac{1}{2\nu^2} \mathbf{w}^\top \mathbf{w},$$
$$= \underbrace{-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2}_{\text{正则化项 regularization term}} - \underbrace{\frac{1}{2\nu^2} \mathbf{w}^\top \mathbf{w}}_{\text{正则化系数 regularization coefficient}}.$$

正则化系数  $\lambda = \sigma^2/\nu^2$

# 总结

---

	无先验	引入先验
平方误差	经验风险 最小化	结构风险 最小化
概率	最大似然估计	最大后验估计

$$\mathbf{w}^{ML} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{y}$$

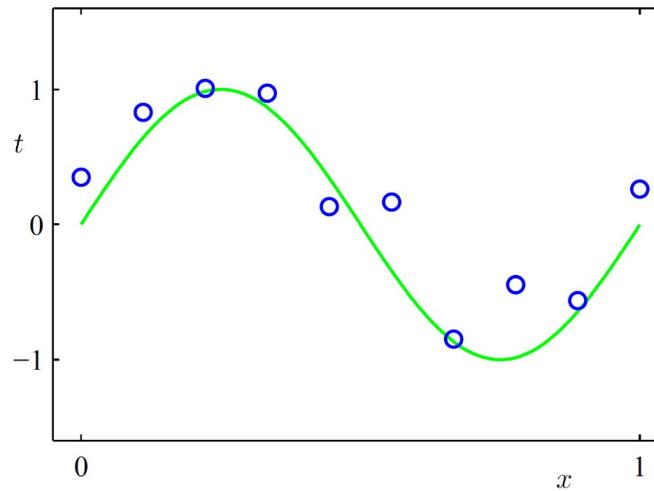
$$\mathbf{w}^* = (\mathbf{X}\mathbf{X}^\top + \lambda I)^{-1}\mathbf{X}\mathbf{y}$$



多项式回归

# 一个例子：Polynomial Curve Fitting

From chapter 1 of Bishop's PRML.



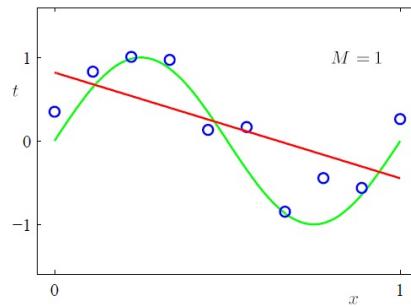
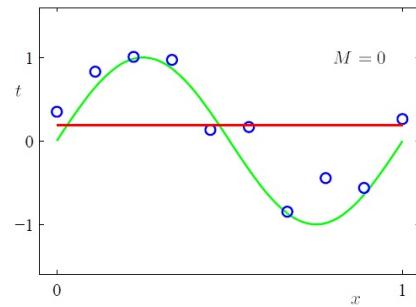
模型

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \dots + w_M x^M$$

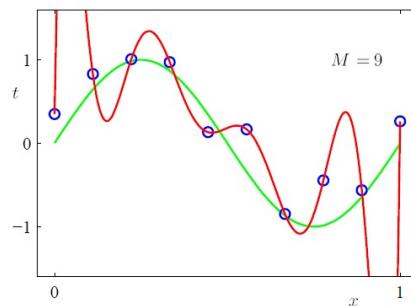
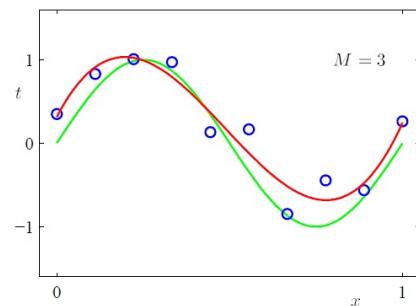
损失函数

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

# Which Degree of Polynomial?

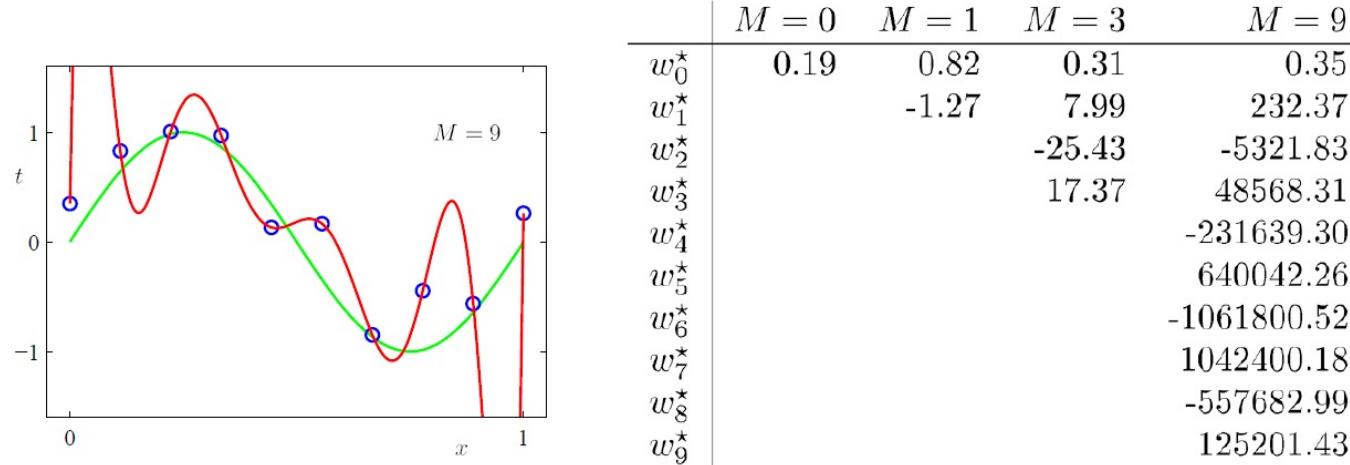


A **model selection** problem



$M = 9 \rightarrow E(w) = 0$ : This is **overfitting**

# Controlling Overfitting: Regularization



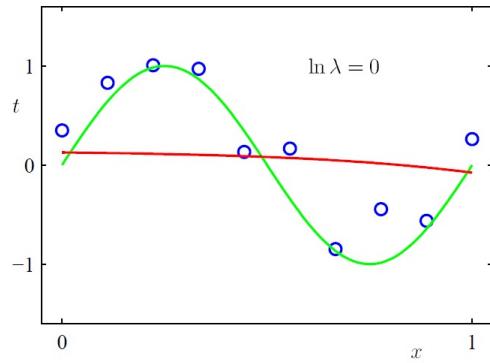
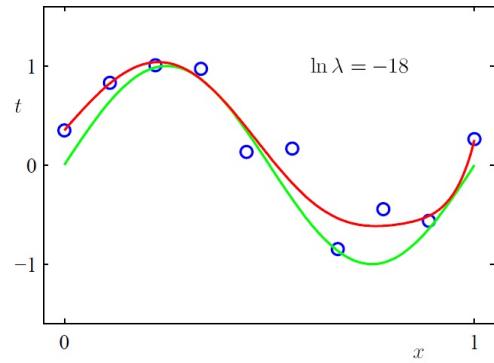
As order of polynomial M increases, so do coefficient magnitudes!

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

对大的系数进行惩罚

# Controlling Overfitting: Regularization

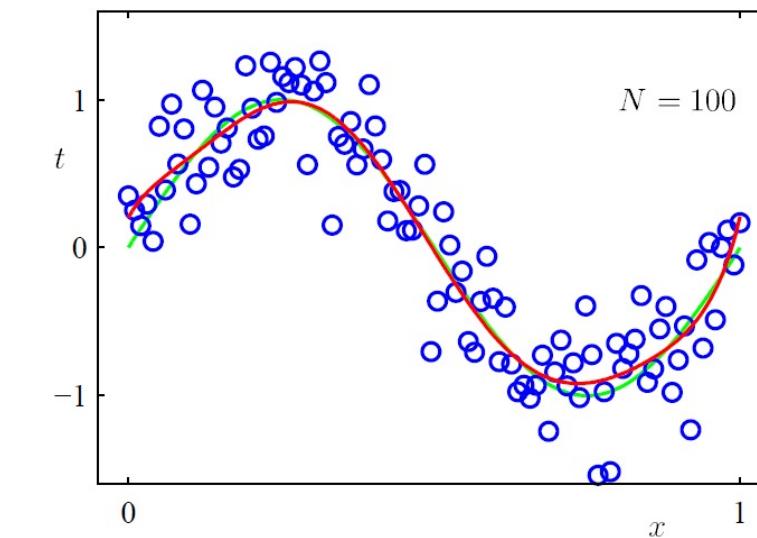
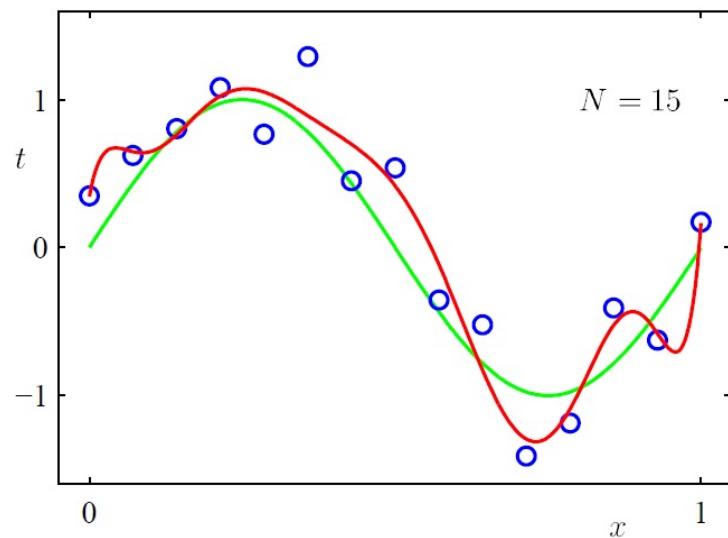
$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$



	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^*$	0.35	0.35	0.13
$w_1^*$	232.37	4.74	-0.05
$w_2^*$	-5321.83	-0.77	-0.06
$w_3^*$	48568.31	-31.97	-0.05
$w_4^*$	-231639.30	-3.89	-0.03
$w_5^*$	640042.26	55.28	-0.02
$w_6^*$	-1061800.52	41.32	-0.01
$w_7^*$	1042400.18	-45.95	-0.00
$w_8^*$	-557682.99	-91.53	0.00
$w_9^*$	125201.43	72.68	0.01

## Controlling Overfitting: Dataset size

---





机器学习的损失从哪里来？

# 如何选择一个合适的模型？

## ► 模型选择

- 拟合能力强的模型一般复杂度会比较高，容易过拟合。
- 如果限制模型复杂度，降低拟合能力，可能会欠拟合。

## ► 偏差与方差分解

- 期望错误可以分解为

$$\mathcal{R}(f) = (\text{bias})^2 + \text{variance} + \varepsilon.$$

$$\begin{aligned} & \mathbb{E}_{\mathbf{x}} \left[ \left( \mathbb{E}_{\mathcal{D}} [f_{\mathcal{D}}(\mathbf{x})] - f^*(\mathbf{x}) \right)^2 \right] \\ & \qquad \qquad \qquad \diagdown \\ & \mathbb{E}_{\mathbf{x}} \left[ \mathbb{E}_{\mathcal{D}} \left[ (f_{\mathcal{D}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}} [f_{\mathcal{D}}(\mathbf{x})])^2 \right] \right] \\ & \qquad \qquad \qquad \diagup \\ & \mathbb{E}_{(\mathbf{x}, y) \sim p_r(\mathbf{x}, y)} \left[ (y - f^*(\mathbf{x}))^2 \right] \end{aligned}$$

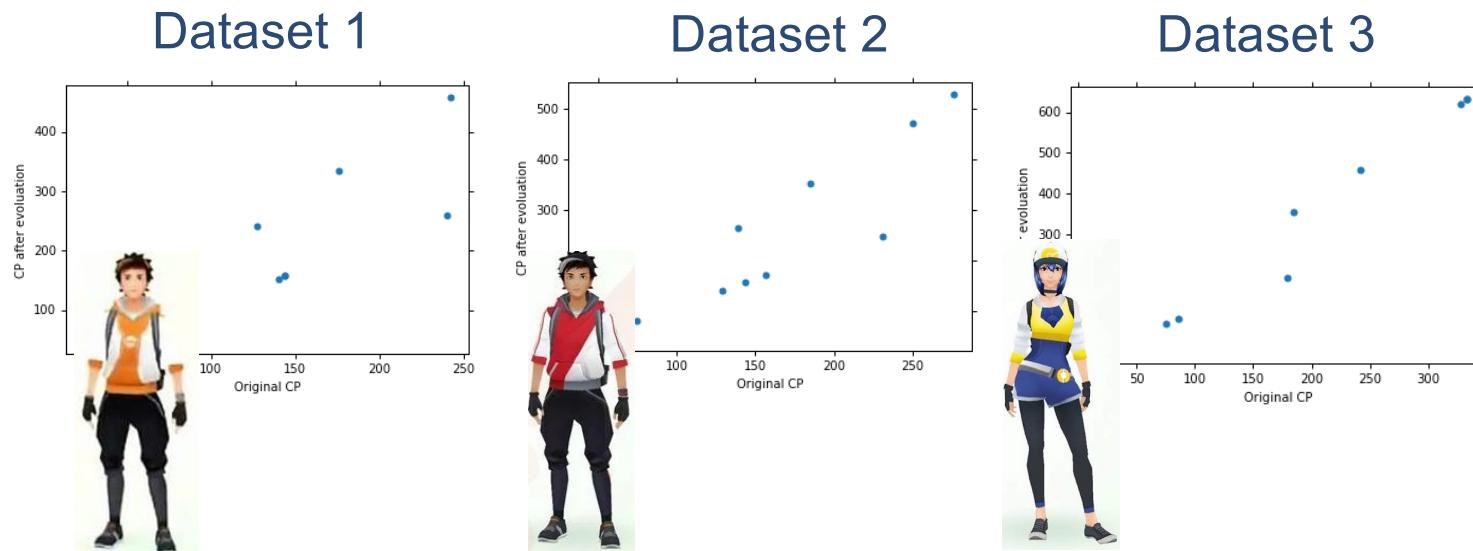
## 偏差-方差分解推导

$$\begin{aligned}\triangleright R(f) &= E_{(x,y) \sim P(x,y)} [(y - f(x))^2] \\&= E_{(x,y) \sim P(x,y)} [(y - f^*(x) + f^*(x) - f(x))^2] \\&= E_{x \sim P(x)} [(f(x) - f^*(x))^2] + \underbrace{E_{(x,y) \sim P(x,y)} [(y - f^*(x))^2]}_{E_D [(f_D(x) - f^*(x))^2]} \\&= E_D [(f_D(x) - E_D(f_D(x)) + E_D(f_D(x)) - f^*(x))^2] \\&= E_D [(f_D(x) - E_D(f_D(x)))^2] + E_D [(E_D(f_D(x)) - f^*(x))^2] \\&\quad + \underbrace{2E_D[(f_D(x) - E_D(f_D(x)))(E_D(f_D(x)) - f^*(x))]}_{=0} = 0\end{aligned}$$

# 多个训练集

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

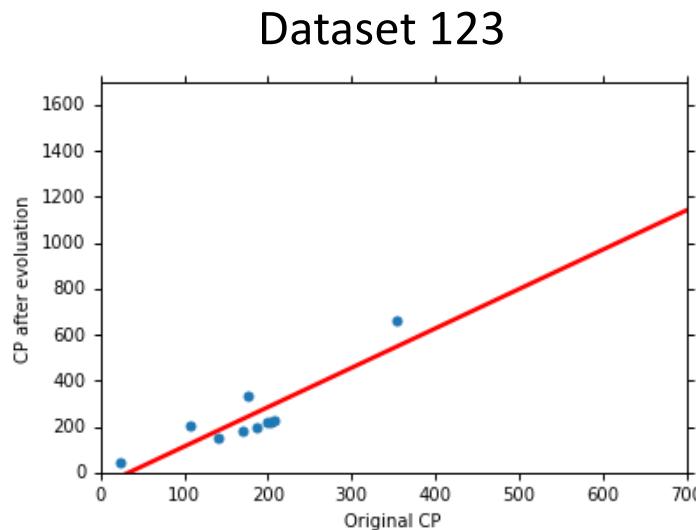
- ▶ 我们尝试从超市选10个芒果作为训练数据来找到最佳模型  
 $f^*$



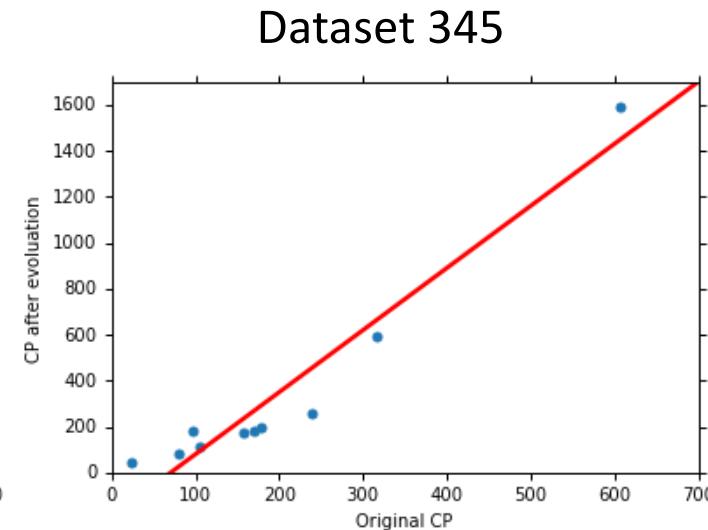
## 训练模型的平均

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

- 在收集到的不同训练集的基础上，每个人拿选到的芒果训练一个模型 $f_D(x)$



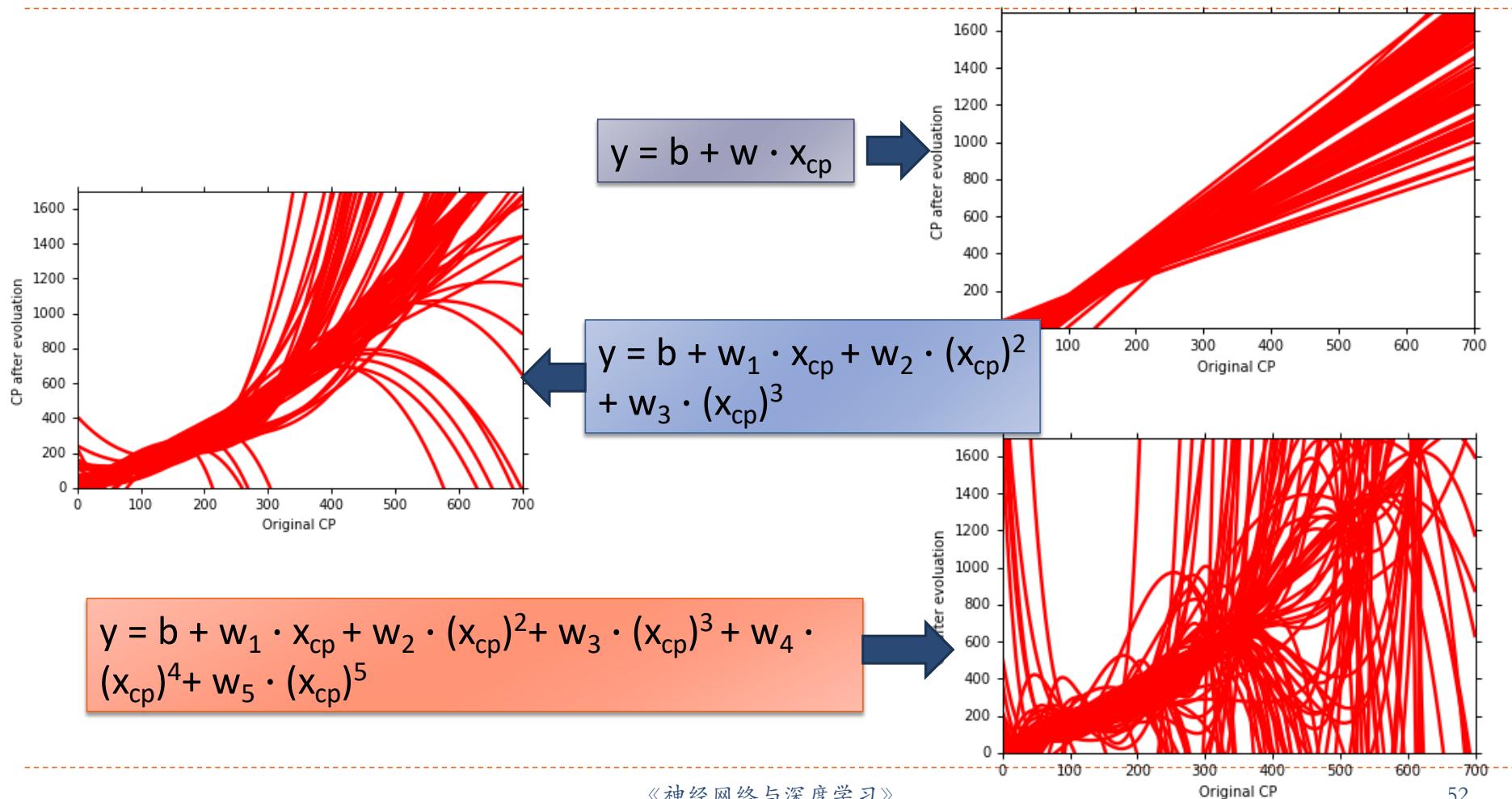
$$y = b + w \cdot x_{cp}$$



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

# 100个人训练的模型 $f_D$

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

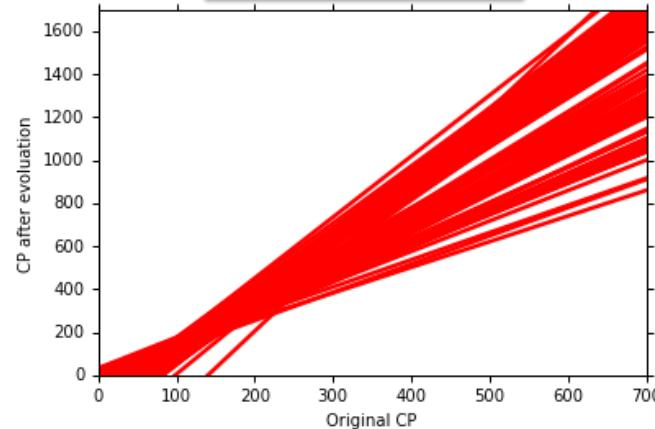


# 方差(variance)

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

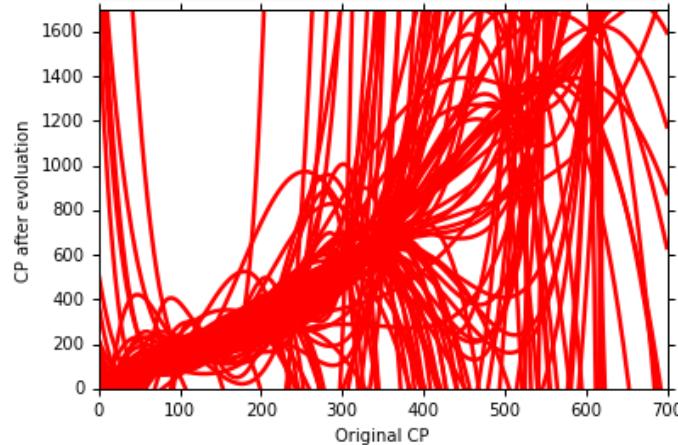
$$\mathbb{E}_{\mathbf{x}} \left[ \mathbb{E}_{\mathcal{D}} \left[ (f_{\mathcal{D}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}}[f_{\mathcal{D}}(\mathbf{x})])^2 \right] \right]$$

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



Small  
Variance

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



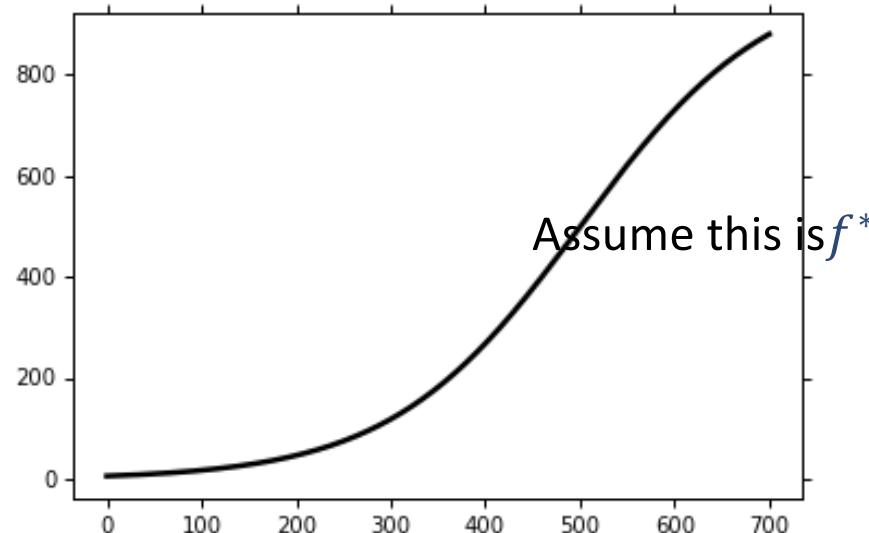
Large  
Variance

## 偏差 ( bias)

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

$$\mathbb{E}_{\mathbf{x}} \left[ \left( \mathbb{E}_{\mathcal{D}} [f_{\mathcal{D}}(\mathbf{x})] - f^*(\mathbf{x}) \right)^2 \right] \quad E_D[f_D] = \bar{f}$$

► 偏差：如果我们平均5000个模型，能否更接近 $f^*$ ？



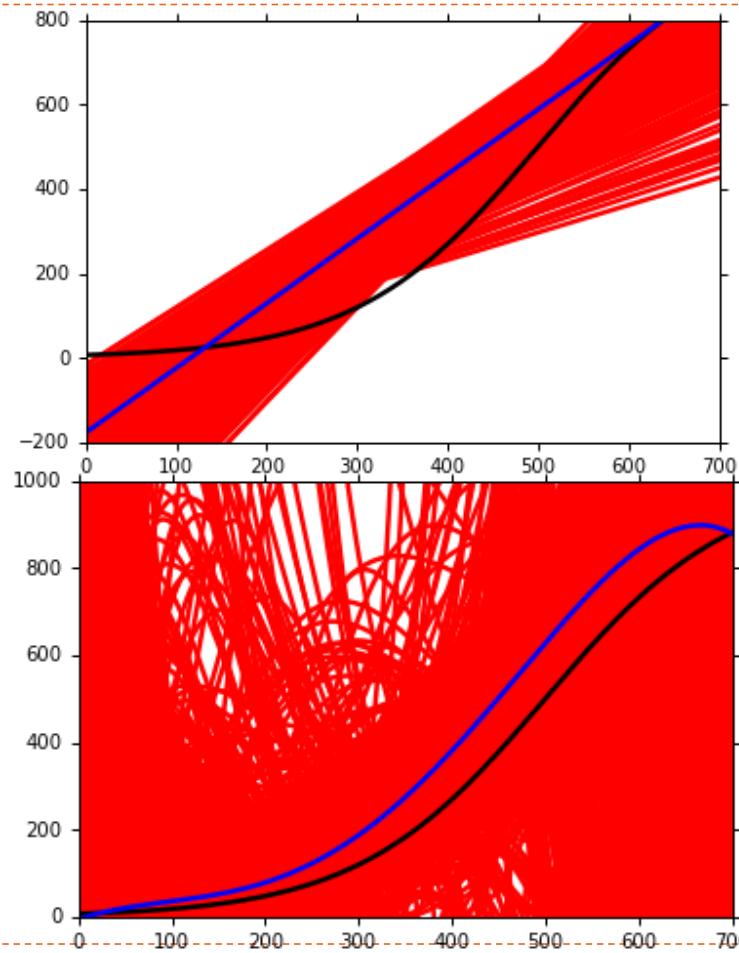
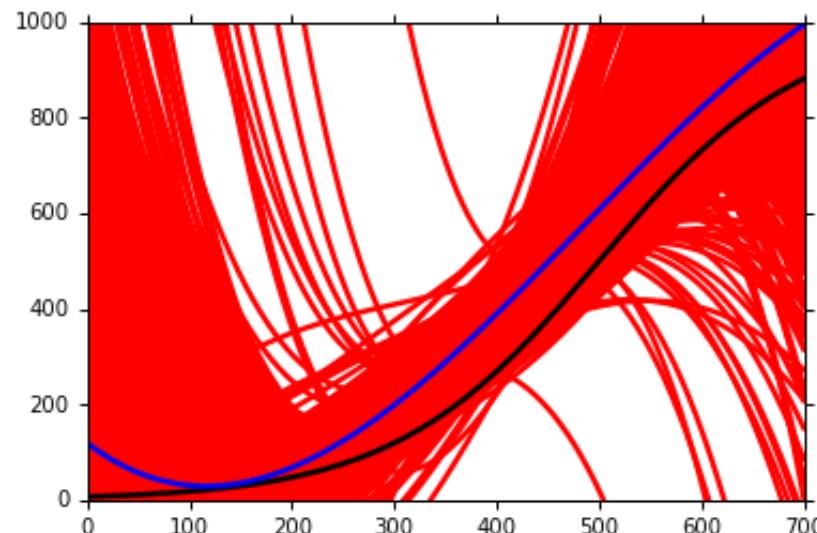
## 偏差 ( bias)

[http://speech.ee.ntu.edu.tw/~tlkagk/courses\\_ML17.html](http://speech.ee.ntu.edu.tw/~tlkagk/courses_ML17.html)

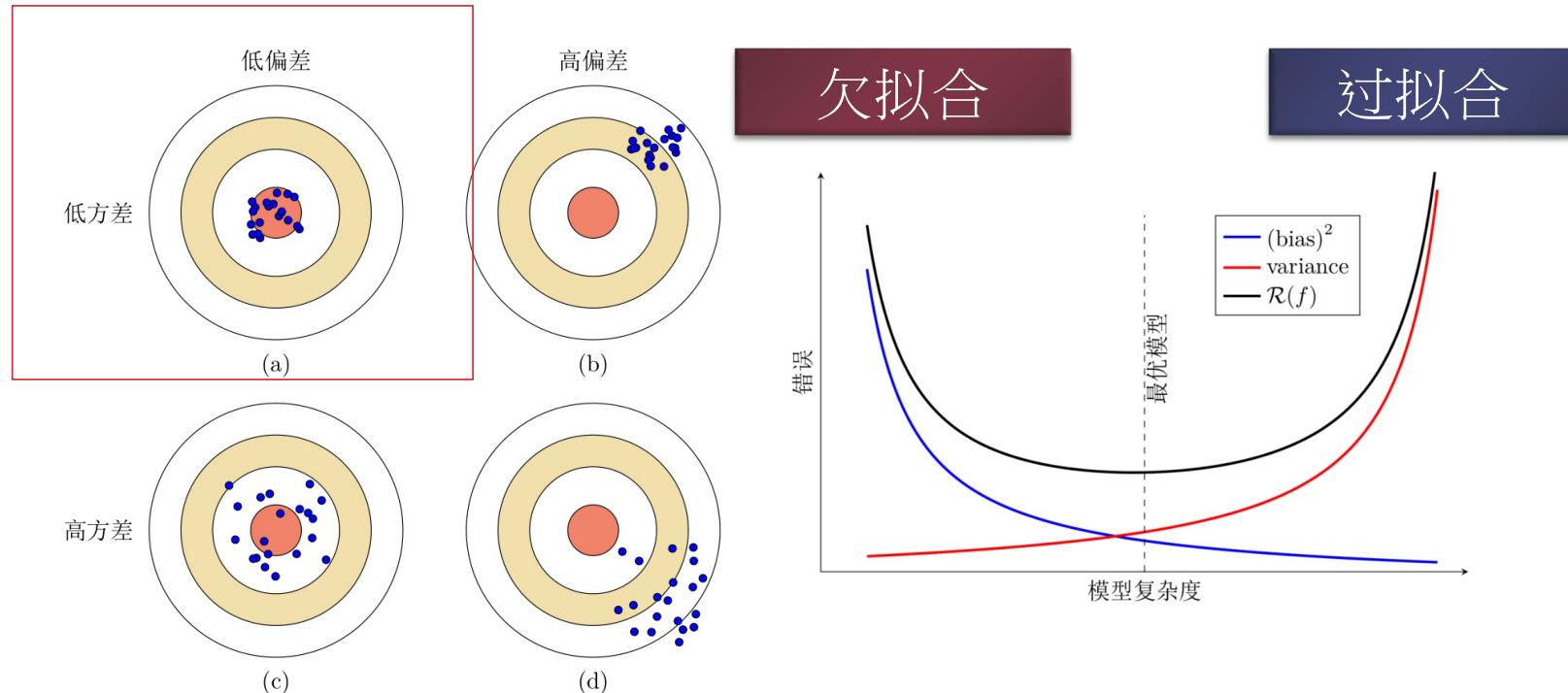
Black curve: 最佳模型  $f^*$

Red curves: 5000模型  $f_D$

Blue curve: 5000模型的平均  $\bar{f}$



# 模型选择：偏差与方差



## 模型选择

---

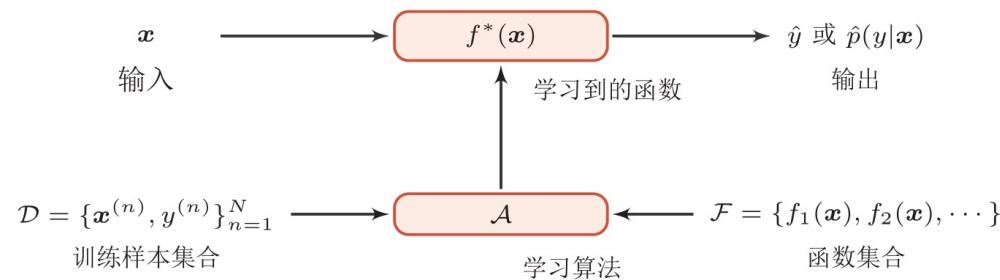
- ▶ 模型在训练集上错误率高，拟合能力不够，偏差较大
  - ▶ 增加数据特征，提高模型复杂度，减小正则化系数...
- ▶ 模型在训练集上错误率低，验证集上错误率高，过拟合，方差较大
  - ▶ 降低模型复杂度，加大正则化系数，集成学习...

# 常见的机器学习类型

	监督学习	无监督学习	强化学习
训练样本	训练集 $\{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$	训练集 $\{\mathbf{x}^n\}_{n=1}^N$	智能体和环境交互的轨迹 $\tau$ 和累积奖励 $G_\tau$
优化目标	$y = f(\mathbf{x})$ 或 $p(y \mathbf{x})$	$p(\mathbf{x})$ 或带隐变量 $\mathbf{z}$ 的 $p(\mathbf{x} \mathbf{z})$	期望总回报 $\mathbb{E}_\tau[G_\tau]$
学习准则	期望风险最小化 最大似然估计	最大似然估计 最小重构错误	策略评估 策略改进

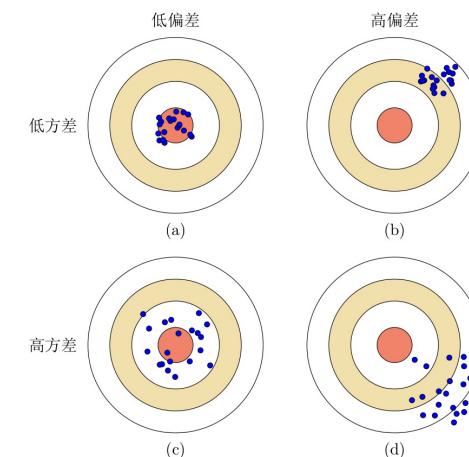
# 教学内容

## ► 机器学习



## ► 线性回归

	无先验	引入先验
平方误差	经验风险 最小化	结构风险 最小化
概率	最大似然估计	最大后验估计



## ► 机器学习的损失从哪里来