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、逻辑斯特回归
  - 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

逻辑斯特回归

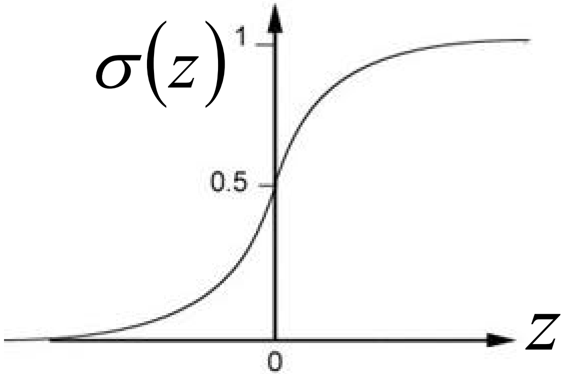
Posterior Probability(后验概率)

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

$$= \frac{1}{1 + \frac{P(x|C_2)P(C_2)}{P(x|C_1)P(C_1)}} = \frac{1}{1 + exp(-z)} = \sigma(z)$$

Sigmoid function

$$z = ln\frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)}$$



逻辑斯特回归

Posterior Probability(后验概率)

$$P(C_1|x) = \sigma(z)$$ sigmoid $$z = ln\frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)}$$

$$z = ln\frac{P(x|C_1)}{P(x|C_2)} + ln\boxed{\frac{P(C_1)}{P(C_2)}} \Rightarrow \frac{\frac{N_1}{N_1+N_2}}{\frac{N_2}{N_1+N_2}} = \frac{N_1}{N_2}$$

$$P(x|C_1) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma^1|^{1/2}}exp\left\{-\frac{1}{2}(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1)\right\}$$

$$P(x|C_2) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma^2|^{1/2}}exp\left\{-\frac{1}{2}(x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)\right\}$$

参考自李宏毅老师课件

逻辑斯特回归

Posterior Probability(后验概率)

$$P(x|C_1) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^1|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1)\right\}$$

$$P(x|C_2) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)\right\}$$

$$ln\frac{\cancel{\frac{1}{(2\pi)^{D/2}}} \frac{1}{|\Sigma^1|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1)\right\}}{\cancel{\frac{1}{(2\pi)^{D/2}}} \frac{1}{|\Sigma^2|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)\right\}}$$

$$= \boxed{ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}}} \boxed{exp\left\{-\frac{1}{2}[(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1) - (x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)]\right\}}$$

$$= ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2}[(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1) - (x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)]$$

参考自李宏毅老师课件

逻辑斯特回归

Posterior Probability(后验概率)

$$= ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2}[(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1) - (x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)]$$

$$(x-\mu^1)^T(\Sigma^1)^{-1}(x-\mu^1)$$

$$= x^T(\Sigma^1)^{-1}x - x^T(\Sigma^1)^{-1}\mu^1 - (\mu^1)^T(\Sigma^1)^{-1}x + (\mu^1)^T(\Sigma^1)^{-1}\mu^1$$

$$= x^T(\Sigma^1)^{-1}x - 2(\mu^1)^T(\Sigma^1)^{-1}x + (\mu^1)^T(\Sigma^1)^{-1}\mu^1$$

$$(x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)$$

$$= x^T(\Sigma^2)^{-1}x - 2(\mu^2)^T(\Sigma^2)^{-1}x + (\mu^2)^T(\Sigma^2)^{-1}\mu^2$$

$$\boxed{\begin{aligned} z = ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} & - \frac{1}{2}x^T(\Sigma^1)^{-1}x + (\mu^1)^T(\Sigma^1)^{-1}x - \frac{1}{2}(\mu^1)^T(\Sigma^1)^{-1}\mu^1 \\ & + \frac{1}{2}x^T(\Sigma^2)^{-1}x - (\mu^2)^T(\Sigma^2)^{-1}x + \frac{1}{2}(\mu^2)^T(\Sigma^2)^{-1}\mu^2 + ln\frac{N_1}{N_2} \end{aligned}}$$

逻辑斯特回归

Posterior Probability(后验概率) $P(C_1|x) = \sigma(z)$

$$\boxed{\begin{aligned} z = {} & \cancel{\ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}}} \cancel{- \frac{1}{2} x^T (\Sigma^1)^{-1} x} + (\mu^1)^T (\Sigma^1)^{-1} x - \frac{1}{2} (\mu^1)^T (\Sigma^1)^{-1} \mu^1 \\ & \cancel{+ \frac{1}{2} x^T (\Sigma^2)^{-1} x} - (\mu^2)^T (\Sigma^2)^{-1} x + \frac{1}{2} (\mu^2)^T (\Sigma^2)^{-1} \mu^2 + \ln \frac{N_1}{N_2} \end{aligned}}$$

$$\Sigma_1 = \Sigma_2 = \Sigma$$

$$z = \underbrace{(\mu^1 - \mu^2)^T \Sigma^{-1}}_{\boldsymbol{w}^T} x \underbrace{- \frac{1}{2} (\mu^1)^T \Sigma^{-1} \mu^1 + \frac{1}{2} (\mu^2)^T \Sigma^{-1} \mu^2 + \ln \frac{N_1}{N_2}}_{\mathrm{b}}$$

$$P(C_1|x) = \sigma(w \cdot x + b)$$

How about directly find ***w*** and b?

In generative model, we estimate $N_1, N_2, \mu^1, \mu^2, \Sigma$

Then we have ***w*** and b

参考自李宏毅老师课件

逻辑斯特回归

Step 1: Function Set

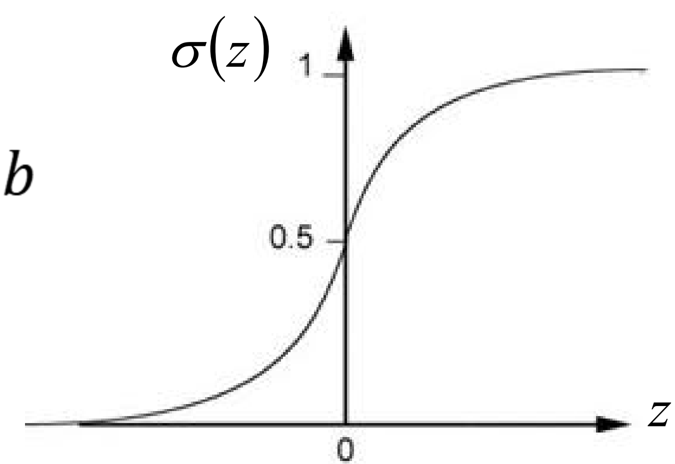
Function set: Including all different w and b

$$\begin{cases} \boxed{z} \geq \boxed{0} & \text{class 1} \\ \boxed{z} < \boxed{0} & \text{class 2} \end{cases}$$

$$P_{w,b}(C_1|x) = \sigma(z)$$

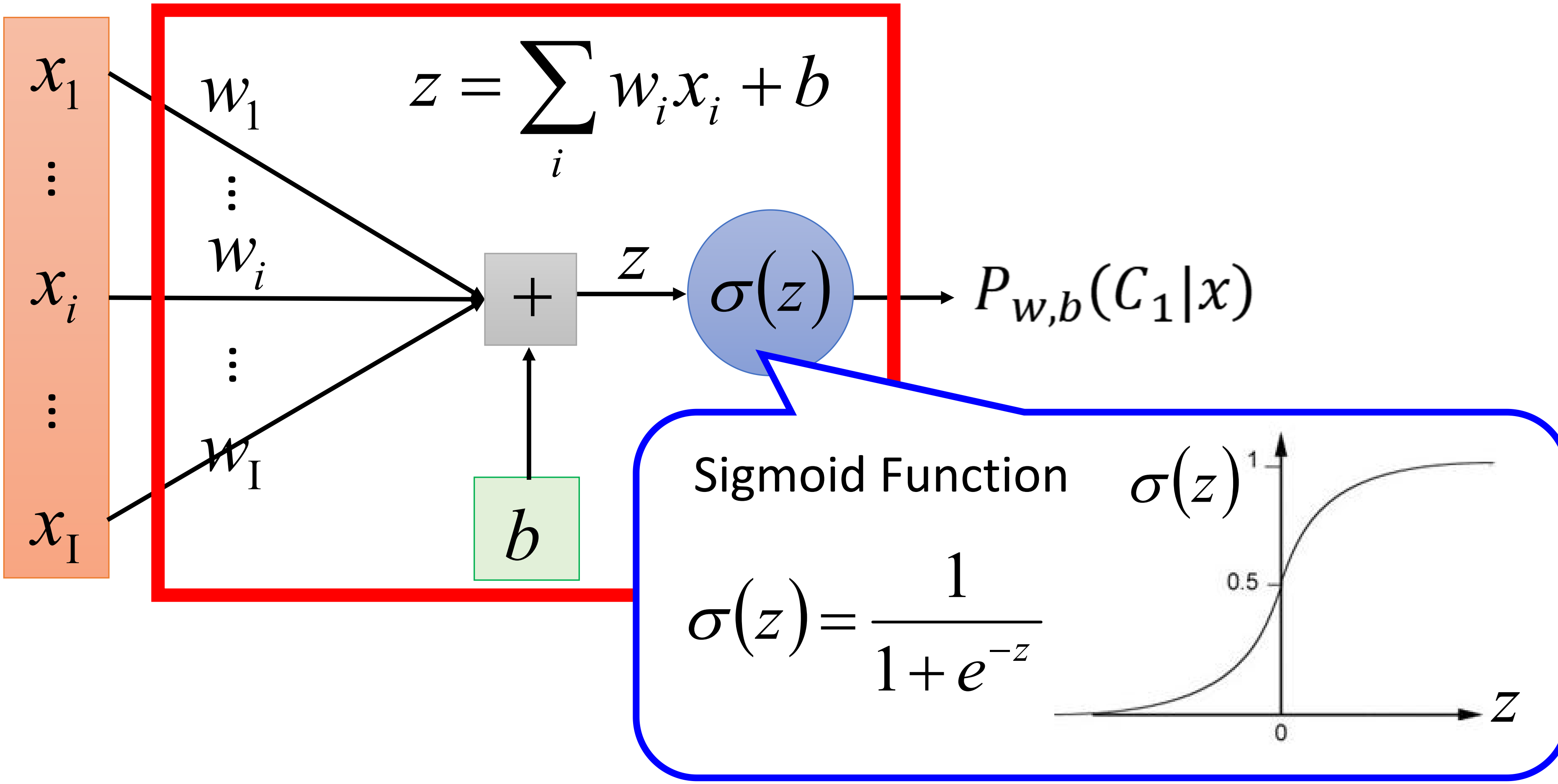
$$z = w \cdot x + b = \sum_i w_i x_i + b$$

$$\sigma(z) = \frac{1}{1 + exp(-z)}$$



逻辑斯特回归

Step 1: Function Set



Step 2: Goodness of a Function

Training Data	x^1	x^2	x^3	......	x^N
	C_1	C_1	C_2		C_1

Assume the data is generated based on $f_{w,b}(x) = P_{w,b}(C_1|x)$

Given a set of w and b, what is its probability of generating the data?

$$L(w,b) = f_{w,b}(x^1) f_{w,b}(x^2) \left(1 - f_{w,b}(x^3)\right) \cdots f_{w,b}(x^N)$$

The most likely w* and b* is the one with the largest $L(w,b)$.

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

逻辑斯特回归

x^1	x^2	x^3	……
C_1	C_1	C_2	

$\Rightarrow$

x^1	x^2	x^3	……
$\hat{y}^1 = 1$	$\hat{y}^2 = 1$	$\hat{y}^3 = 0$	

$\hat{y}^n$: 1 for class 1, 0 for class 2

$$L(w,b) = f_{w,b}(x^1) f_{w,b}(x^2)\left(1 - f_{w,b}(x^3)\right)\cdots$$

$$\boxed{w^*, b^* = \arg\max_{w,b} L(w,b)} = \boxed{w^*, b^* = \arg\min_{w,b} -\ln L(w,b)}$$

$$-\ln L(w,b)$$

$$= -\ln f_{w,b}(x^1) \Rightarrow -\left[1\ \ln f(x^1) + \cancel{0\ \ln(1 - f(x^1))}\right]$$

$$-\ln f_{w,b}(x^2) \Rightarrow -\left[1\ \ln f(x^2) + \cancel{0\ \ln(1 - f(x^2))}\right]$$

$$-\ln\left(1 - f_{w,b}(x^3)\right) \Rightarrow -\left[\cancel{0\ \ln f(x^3)} + 1\ \ln(1 - f(x^3))\right]$$

$\vdots$

逻辑斯特回归

Step 2: Goodness of a Function

$$L(w,b) = f_{w,b}(x^1) f_{w,b}(x^2)\left(1 - f_{w,b}(x^3)\right) \cdots f_{w,b}(x^N)$$

$$-lnL(w,b) = -\left(lnf_{w,b}(x^1) + lnf_{w,b}(x^2) + ln\left(1 - f_{w,b}(x^3)\right)\right) \cdots$$

$\hat{y}^n$: 1 for class 1, 0 for class 2

$$= \sum_n \underline{-\left[\hat{y}^n lnf_{w,b}(x^n) + (1-\hat{y}^n) ln\left(1 - f_{w,b}(x^n)\right)\right]}$$

Cross entropy between two Bernoulli distribution

Distribution p:

$\mathrm{p}(x=1) = \hat{y}^n$

$\mathrm{p}(x=0) = 1 - \hat{y}^n$

cross entropy

Distribution q:

$\mathrm{q}(x=1) = f(x^n)$

$\mathrm{q}(x=0) = 1 - f(x^n)$

$$H(p,q) = -\sum_x p(x) ln\left(q(x)\right)$$

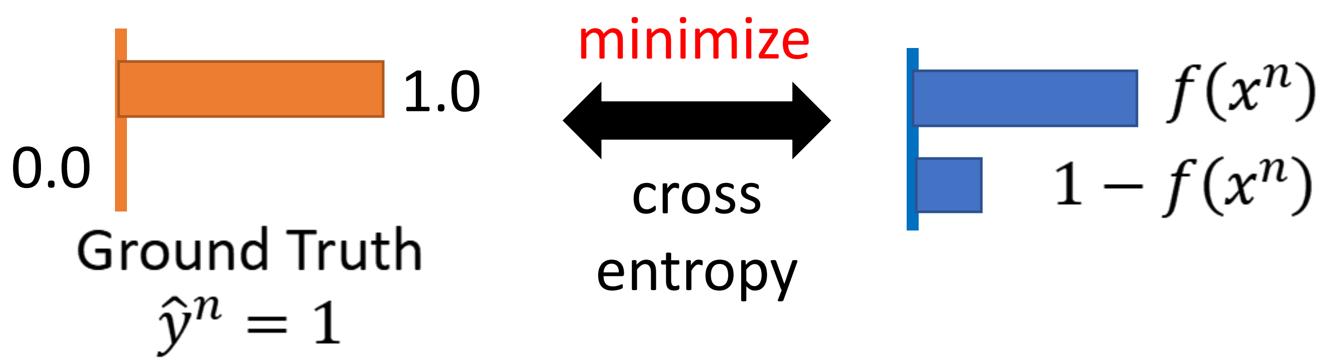
Step 2: Goodness of a Function

$$L(w,b) = f_{w,b}(x^1) f_{w,b}(x^2) \left(1 - f_{w,b}(x^3)\right) \cdots f_{w,b}(x^N)$$

$$-lnL(w,b) = -\left(lnf_{w,b}(x^1) + lnf_{w,b}(x^2) + ln\left(1 - f_{w,b}(x^3)\right)\right) \cdots$$

$\hat{y}^n$: 1 for class 1, 0 for class 2

$$= \sum_n \underbrace{- \left[\hat{y}^n lnf_{w,b}(x^n) + (1-\hat{y}^n) ln\left(1 - f_{w,b}(x^n)\right)\right]}_{\text{Cross entropy between two Bernoulli distribution}}$$



逻辑斯特回归

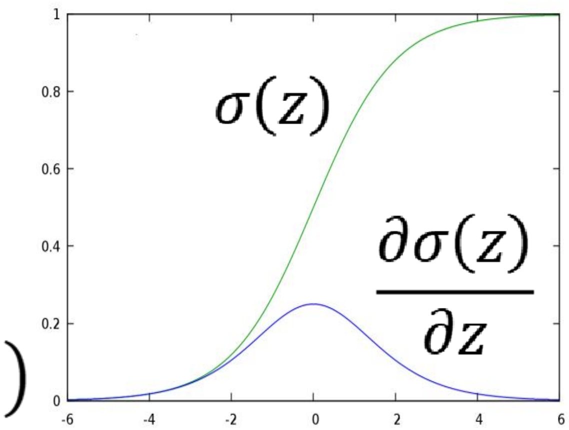
Step 3: Find the best function

$$(1-f_{w,b}(x^n))x_i^n$$

$$\frac{-lnL(w,b)}{\partial w_i} = \sum_n -\left[\hat{y}^n \boxed{\frac{lnf_{w,b}(x^n)}{\partial w_i}} + (1-\hat{y}^n)\frac{ln(1-f_{w,b}(x^n))}{\partial w_i}\right]$$

$$\frac{\partial lnf_{w,b}(x)}{\partial w_i} = \frac{\partial lnf_{w,b}(x)}{\partial z}\frac{\partial z}{\partial w_i} \qquad \frac{\partial z}{\partial w_i} = x_i$$

$$\frac{\partial ln\sigma(z)}{\partial z} = \frac{1}{\sigma(z)}\frac{\partial \sigma(z)}{\partial z} = \frac{1}{\cancel{\sigma(z)}}\cancel{\sigma(z)}(1-\sigma(z))$$



$$f_{w,b}(x) = \sigma(z)$$
$$= 1/1+exp(-z)$$

$$z = w \cdot x + b = \sum_i w_i x_i + b$$

参考自李宏毅老师课件

Step 3: Find the best function

$$\frac{-lnL(w,b)}{\partial w_i} = \sum_n - \left[\hat{y}^n \overbrace{\boxed{\frac{lnf_{w,b}(x^n)}{\partial w_i}}}^{(1-f_{w,b}(x^n))x_i^n} + (1-\hat{y}^n) \overbrace{\boxed{\frac{ln(1-f_{w,b}(x^n))}{\partial w_i}}}^{-f_{w,b}(x^n)x_i^n}\right]$$

$$\frac{\partial ln(1-f_{w,b}(x))}{\partial w_i} = \frac{\partial ln(1-f_{w,b}(x))}{\partial z}\frac{\partial z}{\partial w_i} \qquad \frac{\partial z}{\partial w_i} = x_i$$

$$\frac{\partial ln(1-\sigma(z))}{\partial z} = -\frac{1}{1-\sigma(z)}\frac{\partial \sigma(z)}{\partial z} = -\frac{1}{\cancel{1-\sigma(z)}}\sigma(z)\cancel{(1-\sigma(z))}$$

$$f_{w,b}(x) = \sigma(z)$$
$$= 1/1+exp(-z)$$

$$z = w \cdot x + b = \sum_i w_i x_i + b$$

参考自李宏毅老师课件

逻辑斯特回归

Step 3: Find the best function

$$\frac{-lnL(w,b)}{\partial w_i} = \sum_n -\left[\hat{y}^n \boxed{\frac{lnf_{w,b}(x^n)}{\partial w_i}} + (1-\hat{y}^n)\boxed{\frac{ln\left(1-f_{w,b}(x^n)\right)}{\partial w_i}}\right]$$

$$\left(1-f_{w,b}(x^n)\right)x_i^n \qquad -f_{w,b}(x^n)x_i^n$$

$$= \sum_n -\left[\hat{y}^n \left(1-f_{w,b}(x^n)\right)x_i^n - (1-\hat{y}^n)f_{w,b}(x^n)x_i^n\right]$$

$$= \sum_n -\left[\hat{y}^n \cancel{-\hat{y}^n f_{w,b}(x^n)} - f_{w,b}(x^n) \cancel{+\hat{y}^n f_{w,b}(x^n)}\right]x_i^n$$

$$= \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right)x_i^n$$

Larger difference, larger update

$$w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right)x_i^n$$

逻辑斯特回归

	Logistic Regression	***Linear Regression***
Step 1:	$f_{w,b}(x) = \sigma\left(\sum_i w_i x_i + b\right)$ Output: between 0 and 1	$f_{w,b}(x) = \sum_i w_i x_i + b$ Output: any value
Step 2:	Training data: $(x^n, \hat{y}^n)$ $\hat{y}^n$: 1 for class 1, 0 for class 2 $L(f) = \sum_n l(f(x^n), \hat{y}^n)$	Training data: $(x^n, \hat{y}^n)$ $\hat{y}^n$: a real number $L(f) = \frac{1}{2}\sum_n (f(x^n) - \hat{y}^n)^2$

Cross entropy:

$$l(f(x^n), \hat{y}^n) = -\left[\hat{y}^n ln f(x^n) + (1 - \hat{y}^n) ln\left(1 - f(x^n)\right)\right]$$

参考自李宏毅老师课件

逻辑斯特回归

	Logistic Regression	***Linear Regression***
Step 1:	$f_{w,b}(x) = \sigma\left(\sum_i w_i x_i + b\right)$ Output: between 0 and 1	$f_{w,b}(x) = \sum_i w_i x_i + b$ Output: any value
Step 2:	Training data: $(x^n, \hat{y}^n)$ $\boxed{\hat{y}^n\text{: 1 for class 1, 0 for class 2}}$ $L(f) = \sum_n l(f(x^n), \hat{y}^n)$	Training data: $(x^n, \hat{y}^n)$ $\hat{y}^n$: a real number $L(f) = \frac{1}{2}\sum_n (f(x^n) - \hat{y}^n)^2$

Step 3:

Logistic regression: $w_i \leftarrow w_i - \eta \sum_n -\underline{\left(\hat{y}^n - f_{w,b}(x^n)\right)} x_i^n$

Linear regression: $w_i \leftarrow w_i - \eta \sum_n -\underline{\left(\hat{y}^n - f_{w,b}(x^n)\right)} x_i^n$

逻辑斯特回归

Logistic Regression + Square Error

Step 1: $f_{w,b}(x) = \sigma\left(\sum_i w_i x_i + b\right)$

Step 2: Training data: $(x^n, \hat{y}^n)$, $\hat{y}^n$: 1 for class 1, 0 for class 2

$$L(f) = \frac{1}{2}\sum_n \left(f_{w,b}(x^n) - \hat{y}^n\right)^2$$

Step 3:

$$\frac{\partial\,(f_{w,b}(x)-\hat{y})^2}{\partial w_i} = 2\left(f_{w,b}(x) - \hat{y}\right)\frac{\partial f_{w,b}(x)}{\partial z}\frac{\partial z}{\partial w_i}$$

$$= 2\left(f_{w,b}(x) - \hat{y}\right) f_{w,b}(x)\left(1 - f_{w,b}(x)\right) x_i$$

$\hat{y}^n = 1$

If $f_{w,b}(x^n) = 1$ (close to target) ➡ $\partial L / \partial w_i = 0$

If $f_{w,b}(x^n) = 0$ (far from target) ➡ $\partial L / \partial w_i = 0$

参考自李宏毅老师课件

逻辑斯特回归

Logistic Regression + Square Error

Step 1: $$f_{w,b}(x) = \sigma\left(\sum_i w_i x_i + b\right)$$

Step 2: Training data: $(x^n, \hat{y}^n)$, $\hat{y}^n$: 1 for class 1, 0 for class 2

$$L(f) = \frac{1}{2}\sum_n \left(f_{w,b}(x^n) - \hat{y}^n\right)^2$$

Step 3:

$$\frac{\partial\,(f_{w,b}(x)-\hat{y})^2}{\partial w_i} = 2\left(f_{w,b}(x) - \hat{y}\right)\frac{\partial f_{w,b}(x)}{\partial z}\frac{\partial z}{\partial w_i}$$

$$= 2\left(f_{w,b}(x) - \hat{y}\right)f_{w,b}(x)\left(1 - f_{w,b}(x)\right)x_i$$

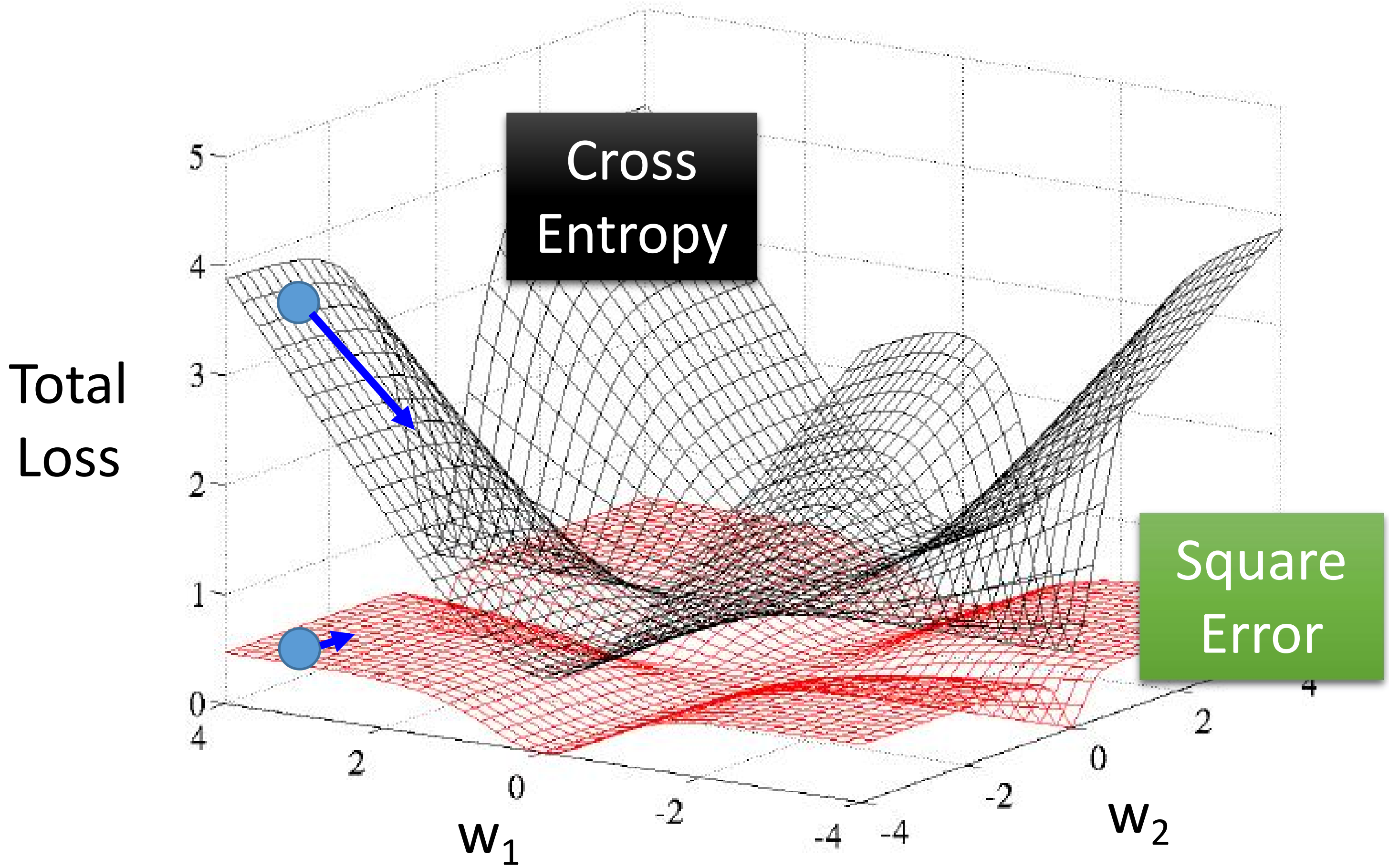
$\hat{y}^n = 0$

If $f_{w,b}(x^n) = 1$ (far from target) ➡ $\partial L/\partial w_i = 0$

If $f_{w,b}(x^n) = 0$ (close to target) ➡ $\partial L/\partial w_i = 0$

逻辑斯特回归

Cross Entropy v.s. Square Error



参考自李宏毅老师课件

Discriminative v.s. Generative

$$P(C_1|x) = \sigma(w \cdot x + b)$$

directly find ***w*** and b

Find $\mu^1, \mu^2, \Sigma^{-1}$

$$w^T = (\mu^1 - \mu^2)^T \Sigma^{-1}$$

$$b = -\frac{1}{2}(\mu^1)^T(\Sigma^1)^{-1}\mu^1 + \frac{1}{2}(\mu^2)^T(\Sigma^2)^{-1}\mu^2 + ln\frac{N_1}{N_2}$$

Will we obtain the same set of w and b?

The same model (function set), but different function may be selected by the same training data.

参考自李宏毅老师课件

逻辑斯特回归

Three Steps

x^1 x^2 x^3 ...... x^n

$\hat{y}^1$ $\hat{y}^2$ $\hat{y}^3$ $\hat{y}^n$

- Step 1. Function Set (Model)

$\hat{y}^n = class\ 1, class\ 2$

x feature → If $P(C_1|x) > 0.5$, output: y = class 1

Otherwise, output: y = class 2 → y class

$$P(C_1|x) = \sigma(w \cdot x + b)$$

w and b are related to $N_1, N_2, \mu^1, \mu^2, \Sigma$

- Step 2. Goodness of a function

$$L(f) = \sum_n \delta(f(x^n) \neq \hat{y}^n) \Rightarrow L(f) = \sum_n l(f(x^n) \neq \hat{y}^n)$$

- Step 3. Find the best function: gradient descent

逻辑斯特回归

Multi-class Classification (3 classes as example)

C_1: w^1, b_1　　$z_1 = w^1 \cdot x + b_1$

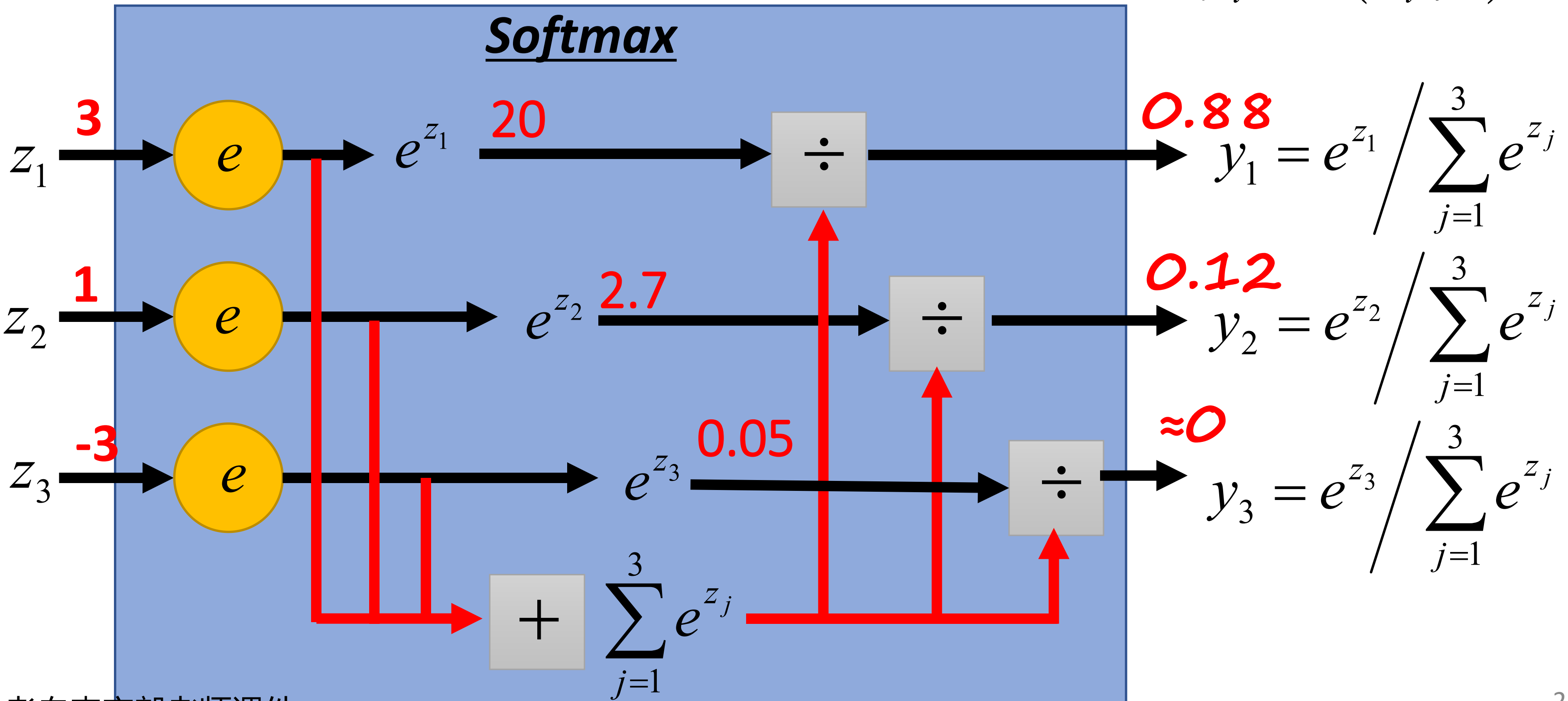
C_2: w^2, b_2　　$z_2 = w^2 \cdot x + b_2$

C_3: w^3, b_3　　$z_3 = w^3 \cdot x + b_3$

Probability:

- $1 > y_i > 0$
- $\sum y_i = 1$

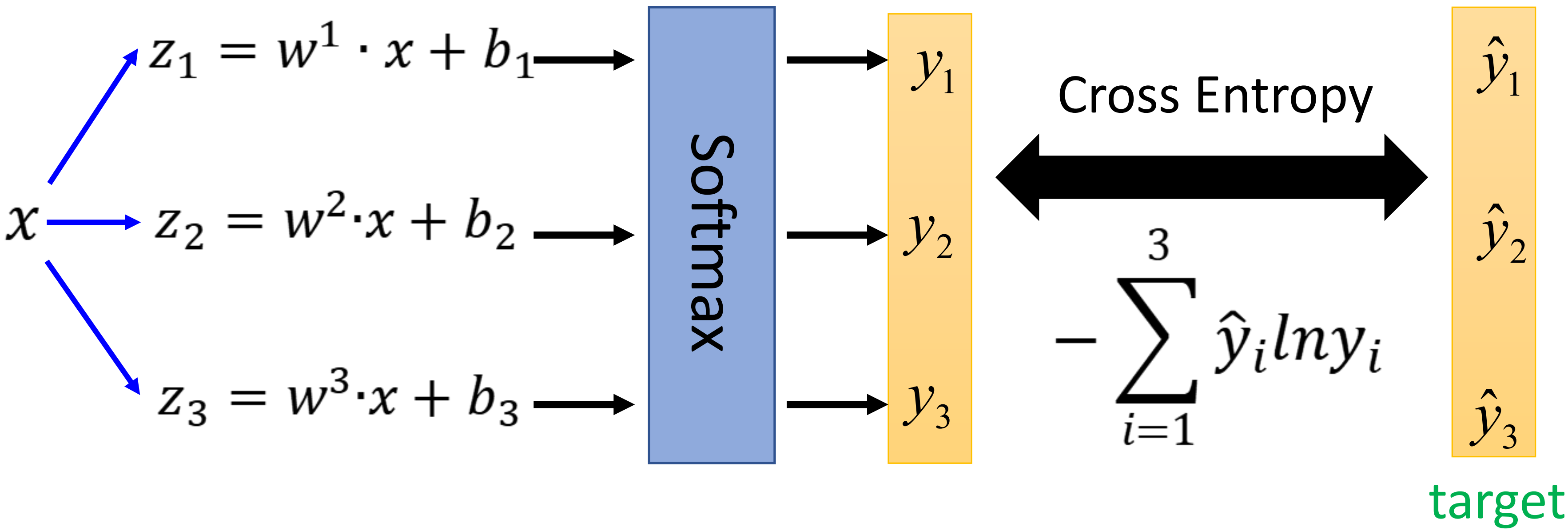
$$y_i = P(C_i \mid x)$$



参考自李宏毅老师课件

逻辑斯特回归

Multi-class Classification (3 classes as example)



If x ∈ class 1

$$\hat{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$-ln y_1$$

If x ∈ class 2

$$\hat{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$-ln y_2$$

If x ∈ class 3

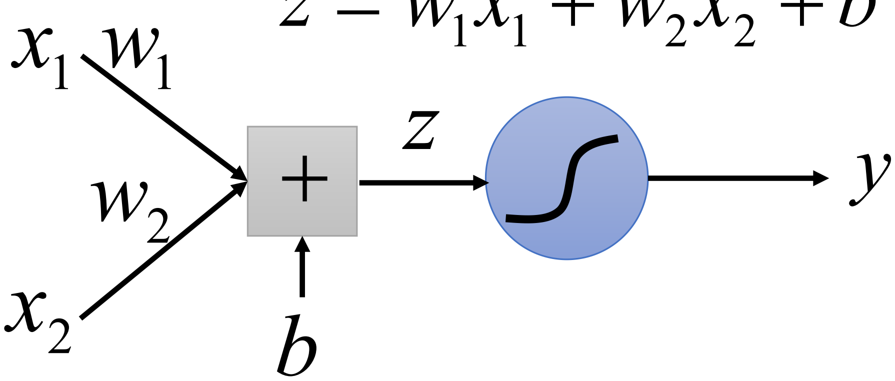
$$\hat{y} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$-ln y_3$$

参考自李宏毅老师课件

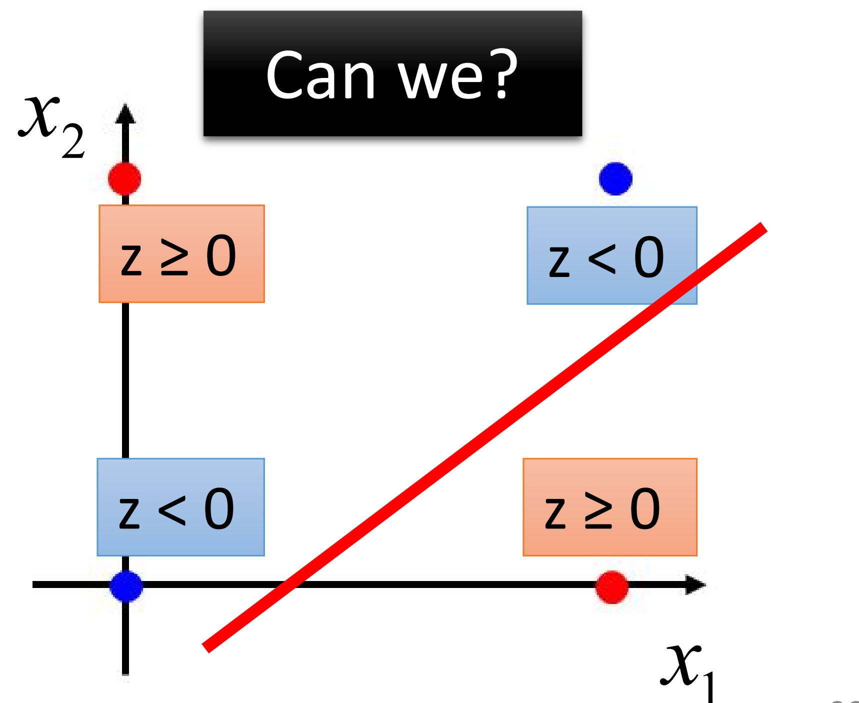
Limitation of Logistic Regression

$$z = w_1 x_1 + w_2 x_2 + b$$



$$\begin{cases} Class1 & y \geq 0.5 \quad (z \geq 0) \\ Class2 & y < 0.5 \quad (z < 0) \end{cases}$$

Input Feature		Label
x_1	x_2	
0	0	Class 2
0	1	Class 1
1	0	Class 1
1	1	Class 2

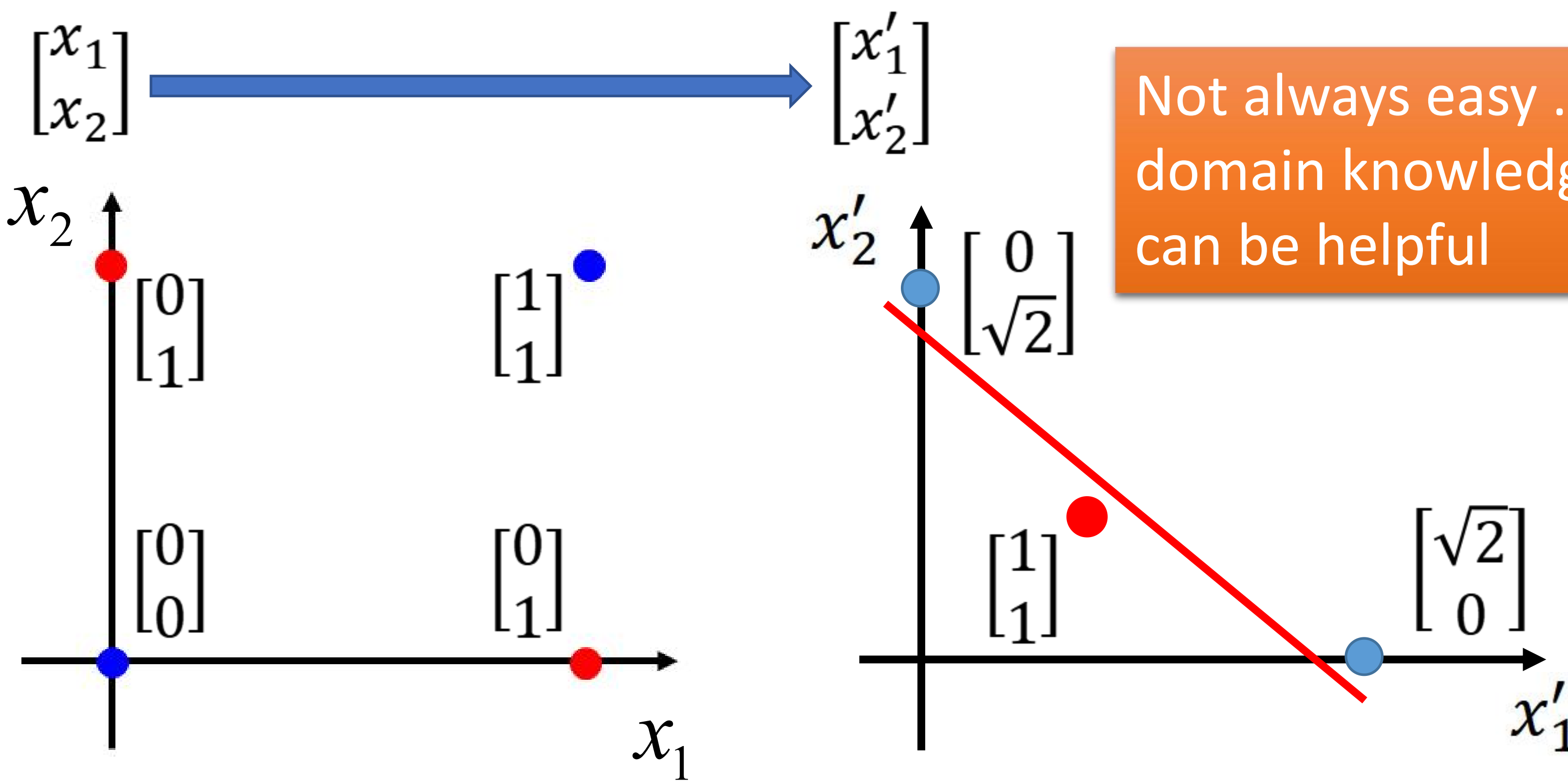


Limitation of Logistic Regression

- ***Feature transformation***

x_1': distance to $\begin{bmatrix}0\\0\end{bmatrix}$

x_2': distance to $\begin{bmatrix}1\\1\end{bmatrix}$

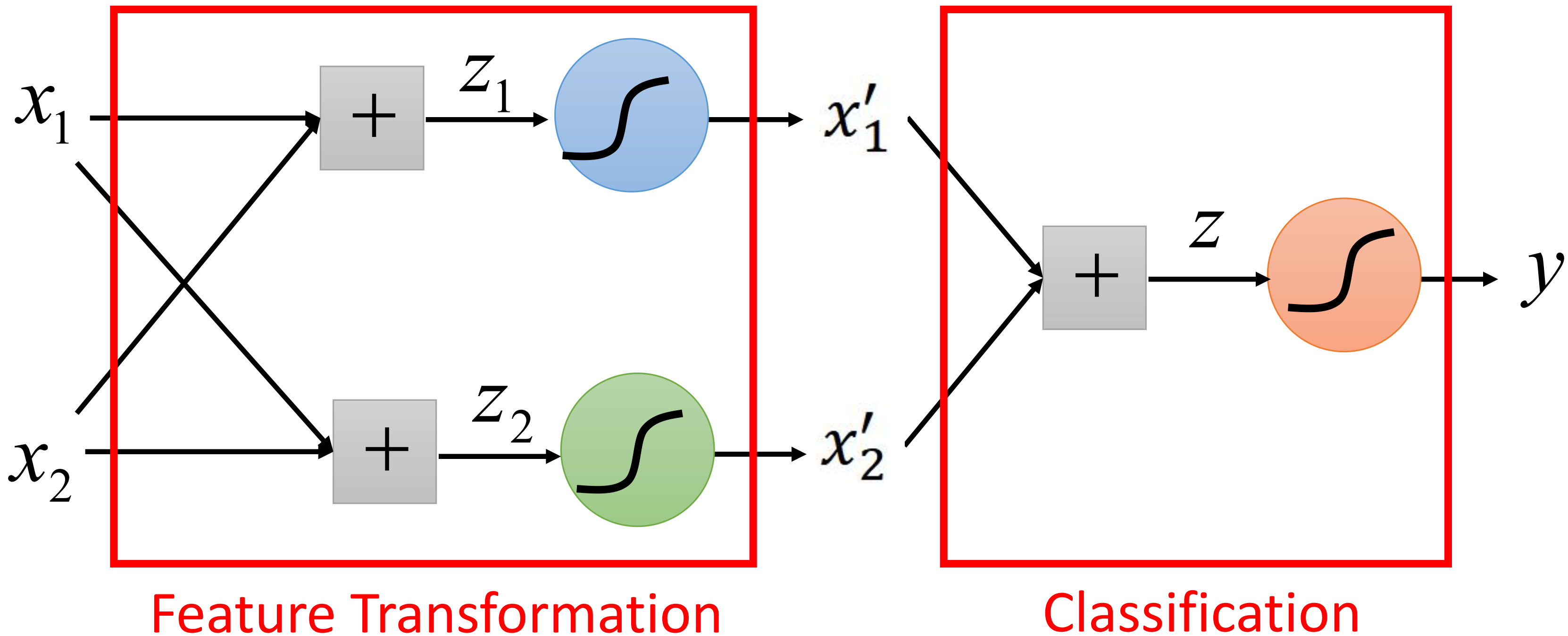


Not always easy ..... domain knowledge can be helpful

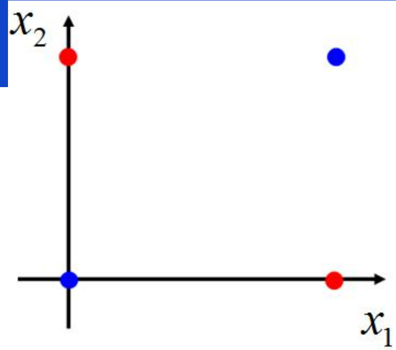
参考自李宏毅老师课件

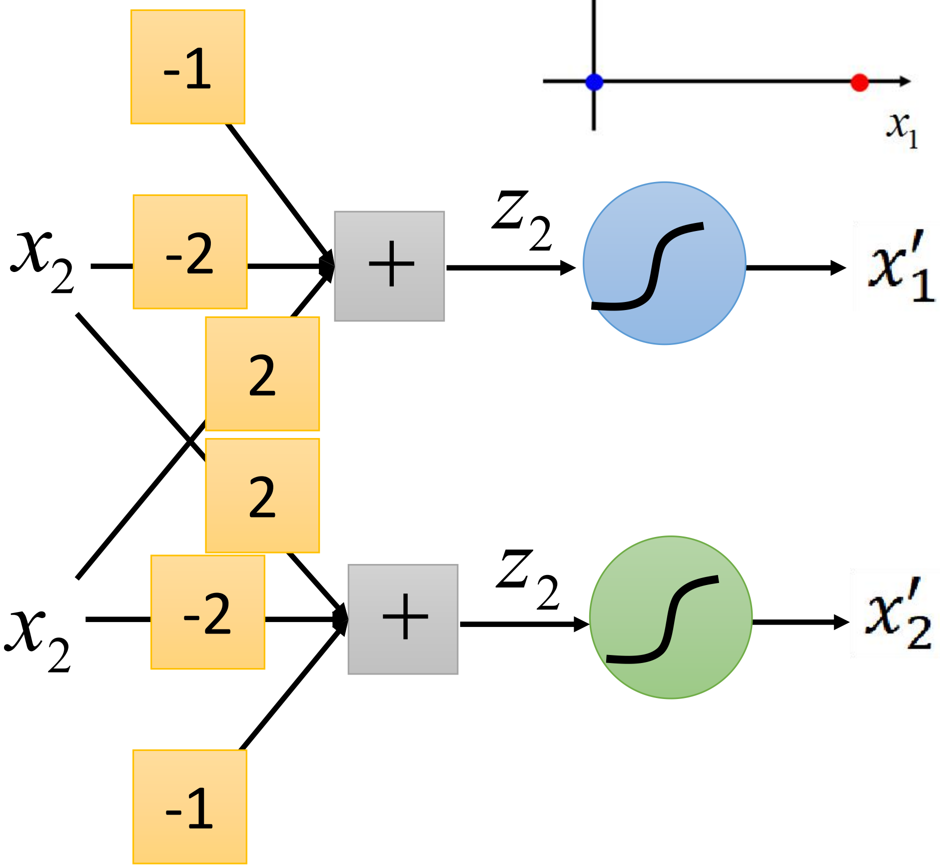
Limitation of Logistic Regression

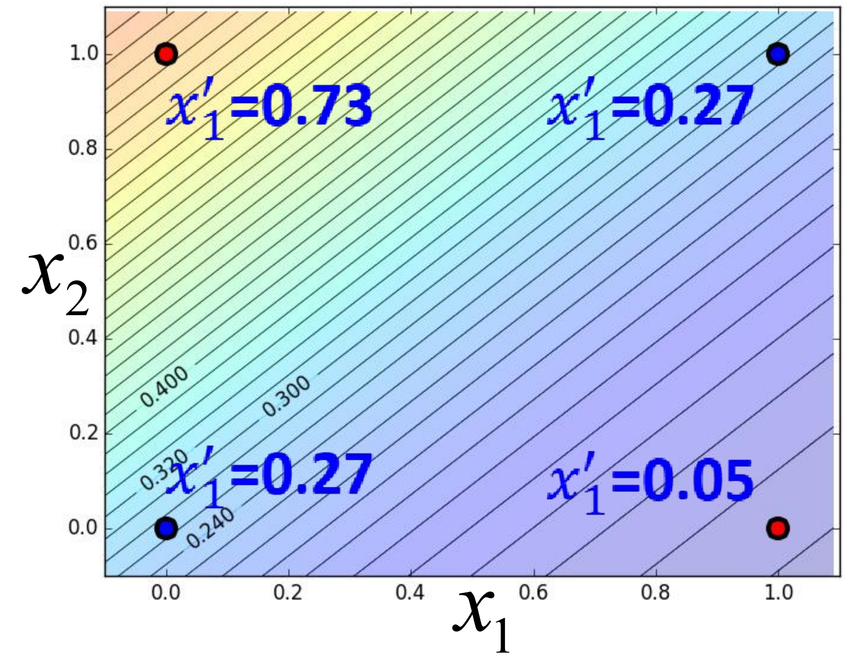
- Cascading logistic regression models

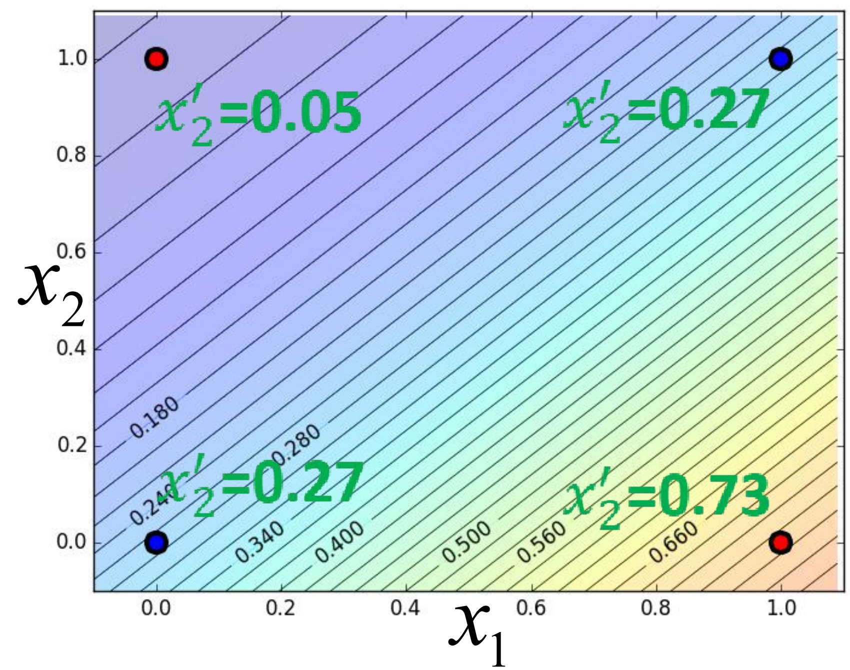


逻辑斯特回归

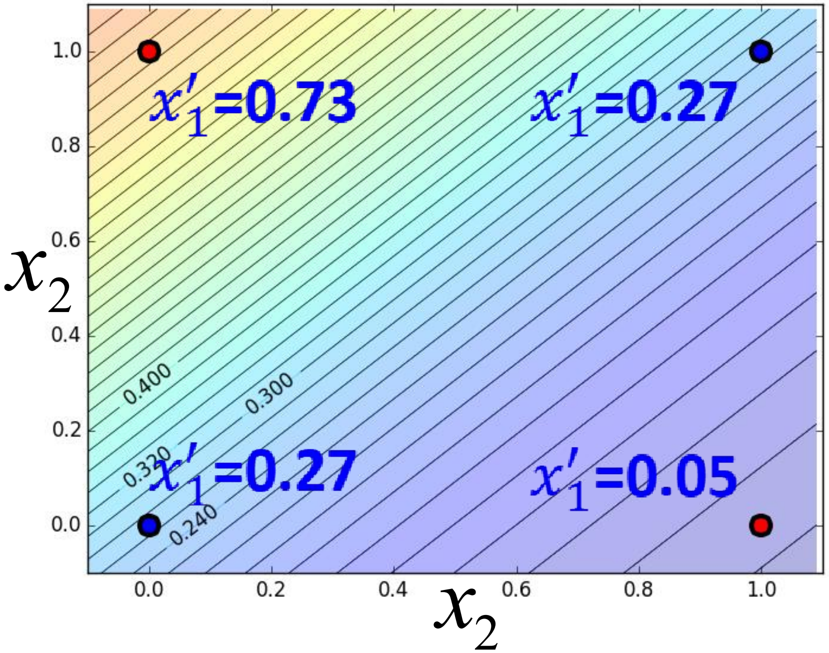


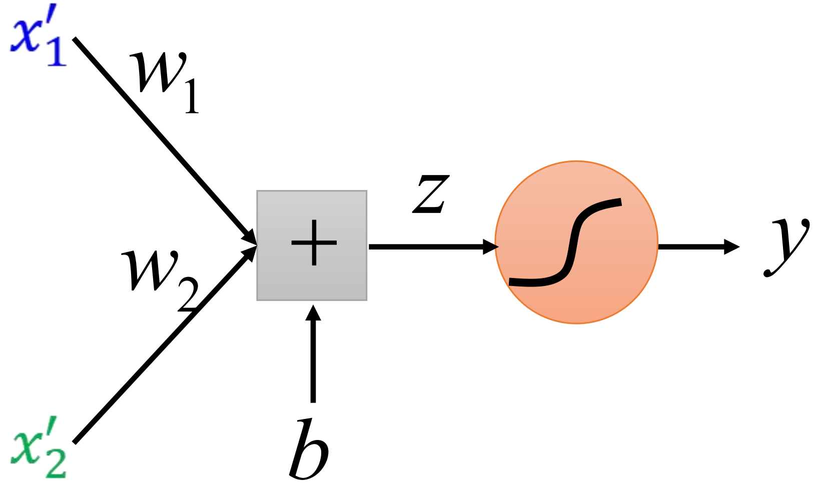


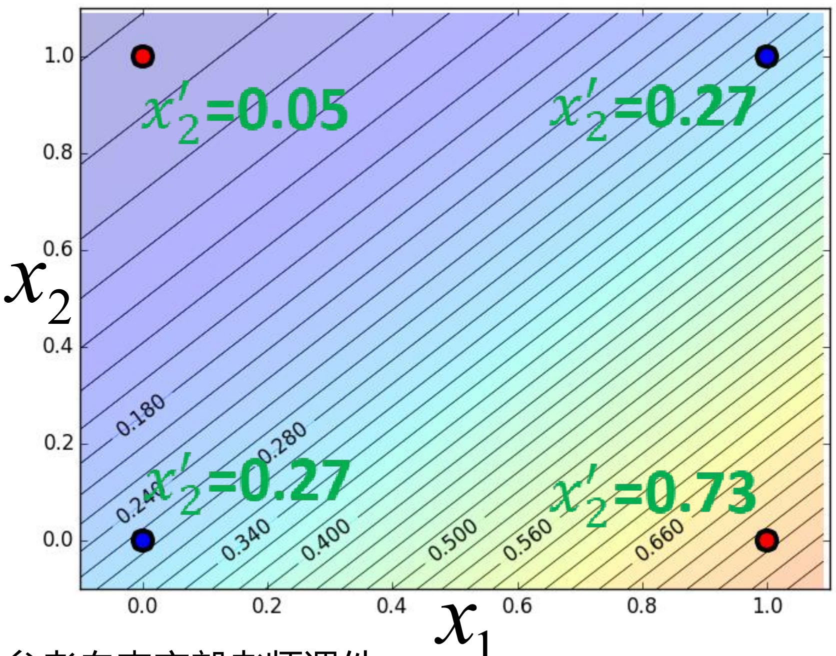


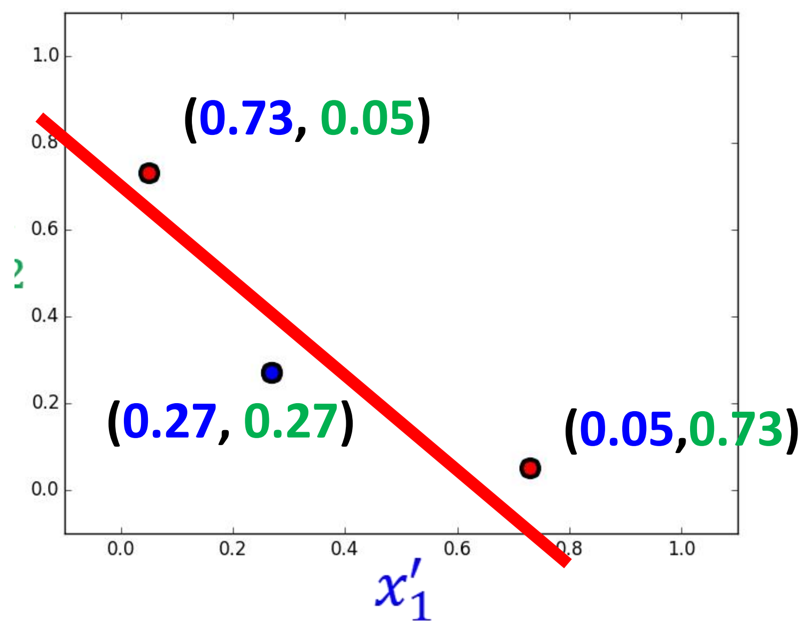


参考自李宏毅老师课件

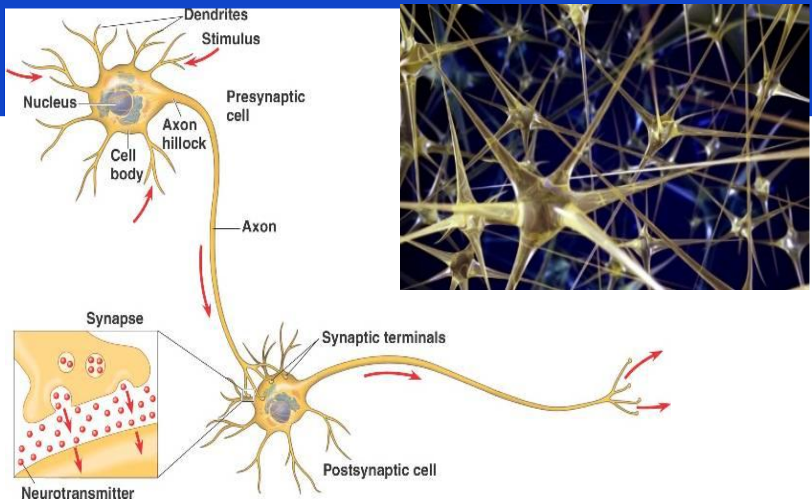

x'_1=0.73
x'_1=0.27
x'_1=0.27
x'_1=0.05
x_2
x_2


x'_1
x'_2
w_1
w_2
z
b
y

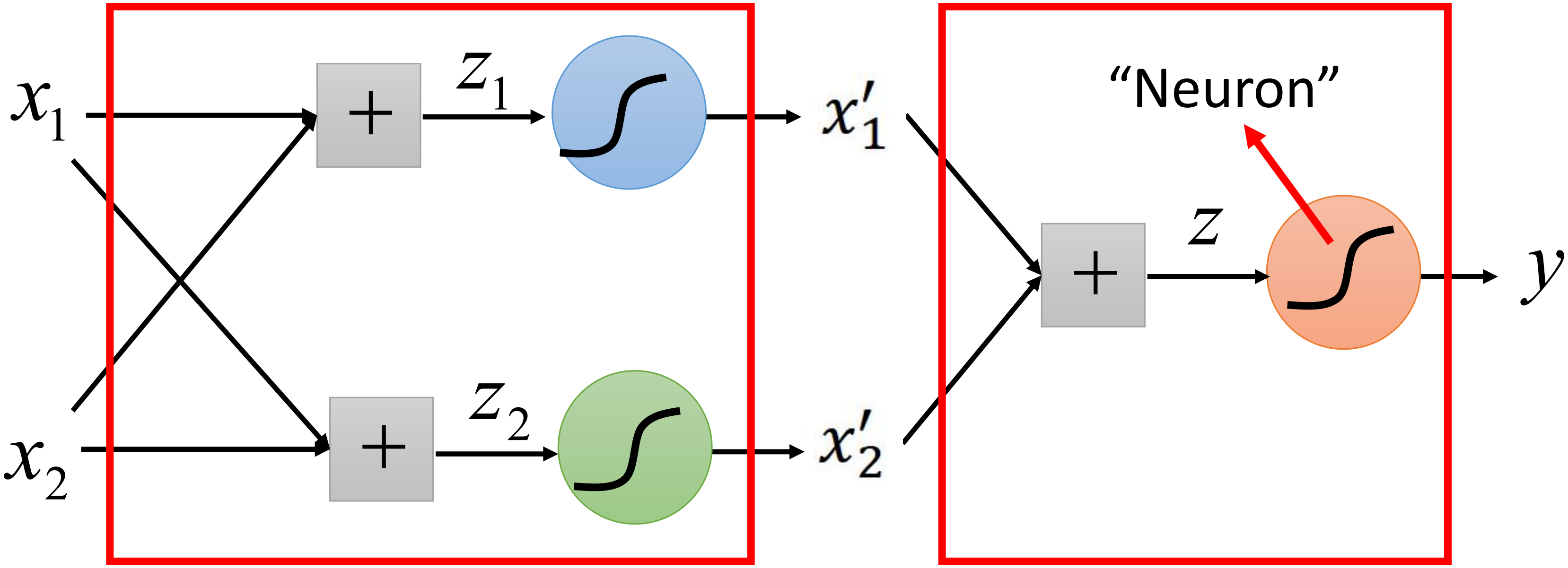

x'_2=0.05
x'_2=0.27
x'_2=0.27
x'_2=0.73
x_2
x_1


(0.73, 0.05)
(0.27, 0.27)
(0.05,0.73)
x'_1

Deep Learning



All the parameters of the logistic regressions are jointly learned.



Neural Network

参考自李宏毅老师课件

Thanks