Logistic Regression

① step 1 Function set

Function set: Including all different w and b

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

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

$$z = wx + b = \sum_i w_i x_i + b$$

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

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 $f_{w,b}(x) = P_{w,b}(C_1|x)$

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

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

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

$$= \arg\min_{w,b} -\ln L(w,b)$$

$\hat{y}$: 1 class 1 ; 0 class 2

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

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

$$= \sum -[\hat{y}^n \ln f_{w,b}(x^n) + (1-\hat{y}^n)(1 - f_{w,b}(x^n))]$$

cross entropy between two Bernoulli distribution

Distribution p:		Distribution q:
$p(x=1) = \hat{y}^n$	cross entropy	$q(x=1) = f(x^n)$
$p(x=0) = 1 - \hat{y}^n$		$q(x=0) = 1 - f(x^n)$

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

minimize

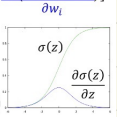
Step 3 Find the best function

$$f_{w,b}(x) = \sigma(z) = \frac{1}{1+e^{-z}} \qquad z = wx + b = \sum_i w_i x_i + b$$

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

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

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



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

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

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

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

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

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

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

Larger difference, larger update

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

$f_{w,b}(x^n)$ 距离目标越远，参数 w 更新的越大

thinking：为什么不用 square Error 计算 Loss

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:

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

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

更新 w

$\hat{y}^n = 1$

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) ⇒ $\partial L/\partial w_i = 0$

$\hat{y}^n = 0$

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) ⇒ $\partial L/\partial w_i = 0$

logistic Regression and Linear Regression

	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 $L(f) = \frac{1}{2}\sum_n (f(x^n) - \hat{y}^n)^2$
Step 3:	Logistic regression: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right) x_i^n$ Linear regression: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right) x_i^n$	

② Discriminative vs Generative

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

Generative → directly find w and b

Discriminative → find μ^1 μ^2 Σ^{-1}

Will we obtain the same set of w and b?

No

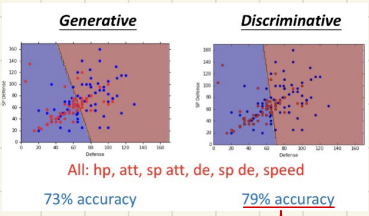
$w^T = (\mu^1 - \mu^2)^T \Sigma^{-1}$

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

Generative：有假设，找服从什么分布 eg：朴素贝叶斯

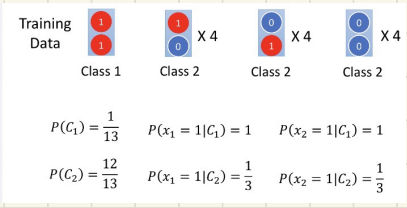
Discriminative：计算 w, b 没有任何假设 eg：KNN, SVM, 决策树

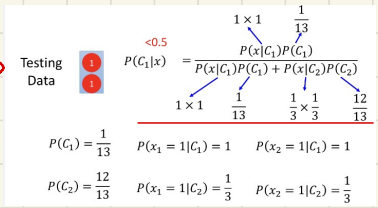
Generative vs Discriminative



better why?

example:



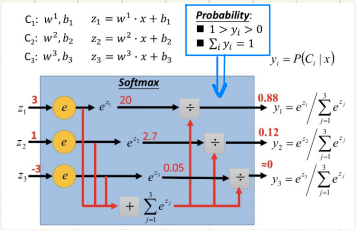


朴素贝叶

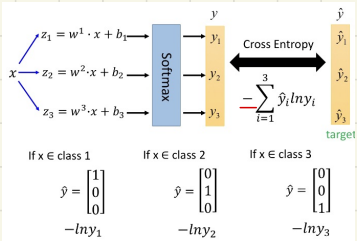
- Usually people belive discriminative model is better
- Benefit of generative model

- With the assumption of probability distribution
  - less training data is needed
  - more robust to the noise
- Priors and class-dependent probabilities can be estimated from different sources.

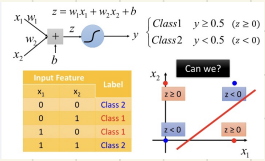
③ Multi-class classification



softmax function：归一化指数函数，二分类函数 sigmoid 在多分类上的推广，目的是将多分类结果以概率形式展示。

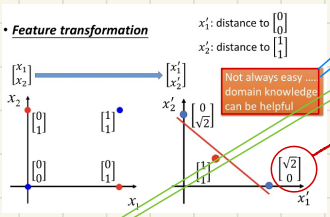


④ Limitation of logistic Regression



无法进行分类!!

feature transformation

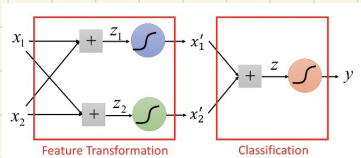


我们希望机器可以自己学习

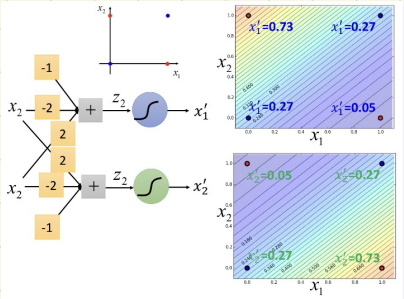
$\sqrt{2} = \sqrt{(1-0)^2+(1-0)^2}$

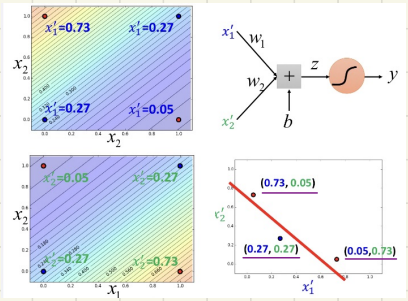
$0 = \sqrt{(1-1)^2+(1-1)^2}$

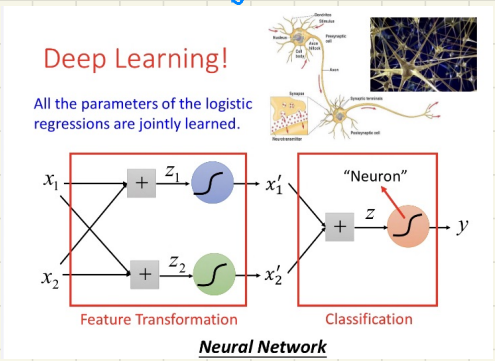
cascading logistic regression models



eg:







3个logistic Regression