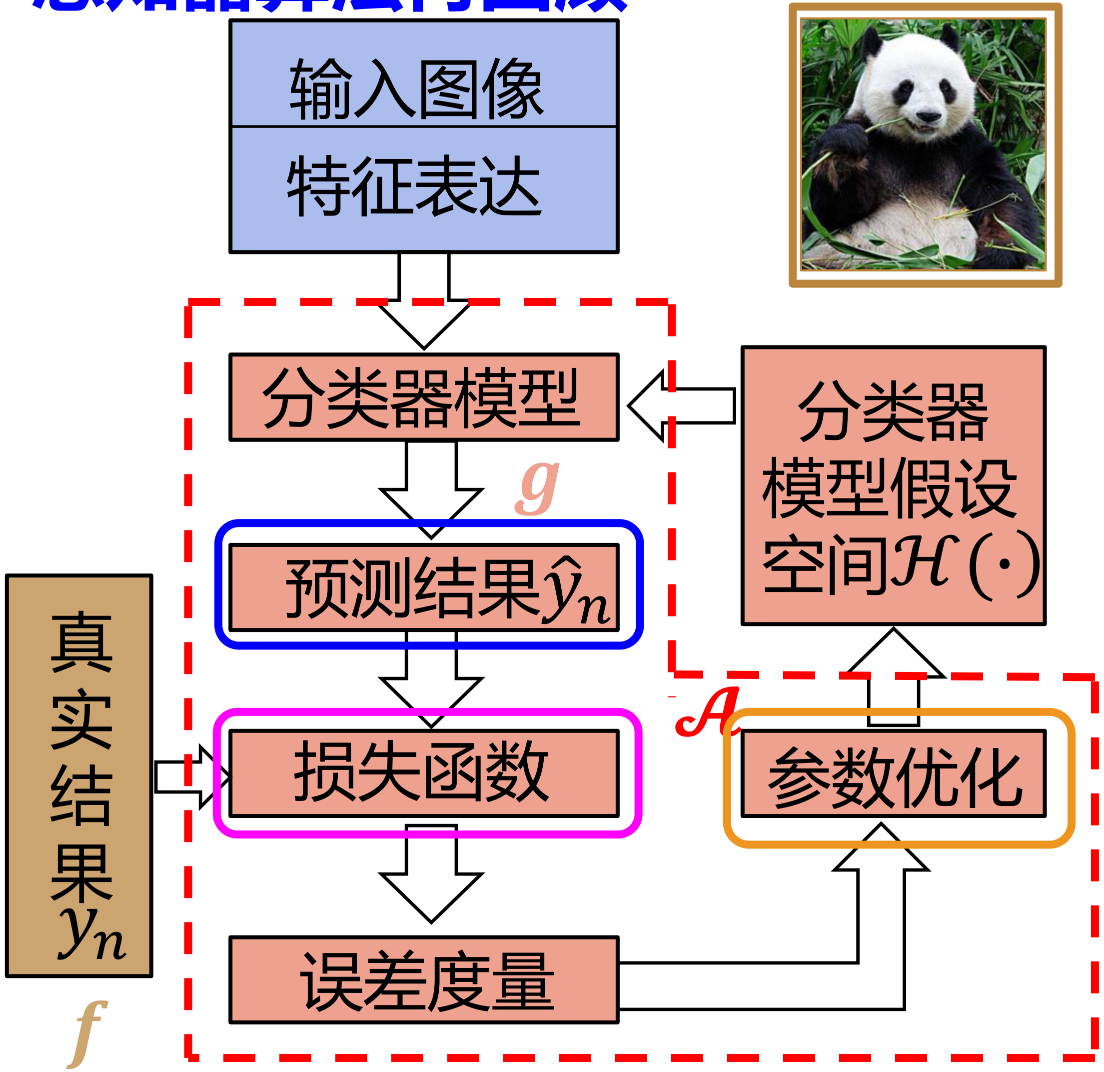
第五讲　逻辑斯蒂回归（*Logistic Regression*）

5.1 逻辑斯蒂回归问题

感知器算法再回顾



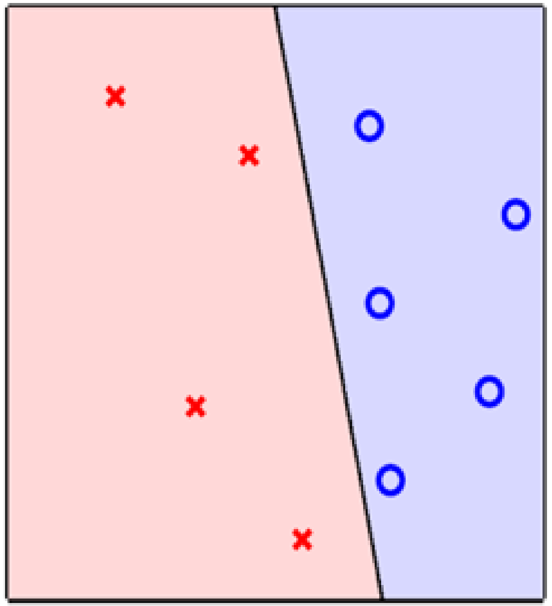
$$\hat{y}_{n(t)} = \text{sign}(\boldsymbol{w}_t^T \mathbf{x}_{n(t)})$$

算法收敛:

$$L_{in} = \sum_{n=1}^{N} [\![y_n \neq \hat{y}_n]\!] = 0$$

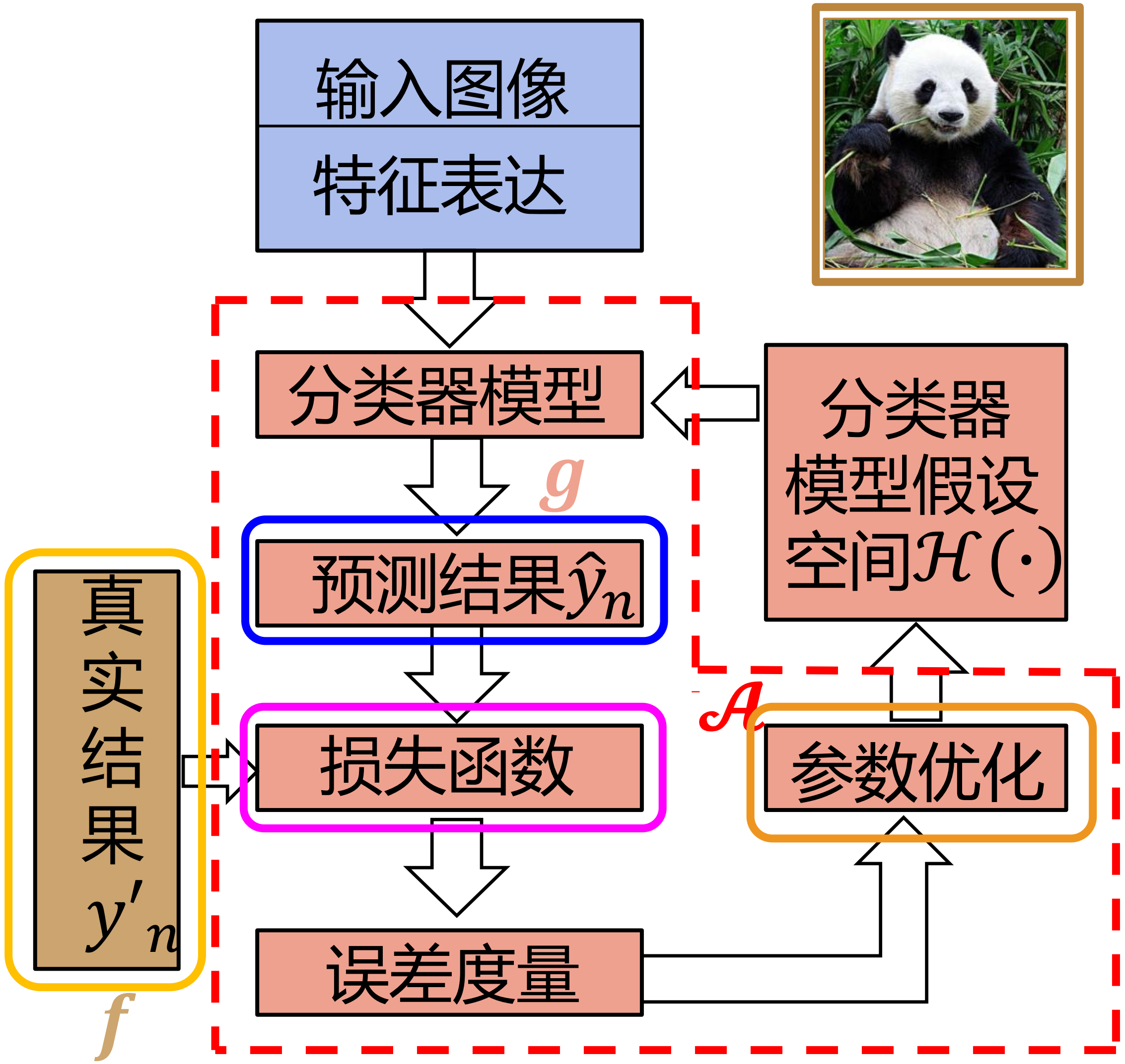
$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + y_n \mathbf{x}_{n(t)}$$

线性可分



- 设置初始分类面(权重)$\boldsymbol{w}_0$
- 如果有样本分错，就修正权重

5.1 逻辑斯蒂回归问题

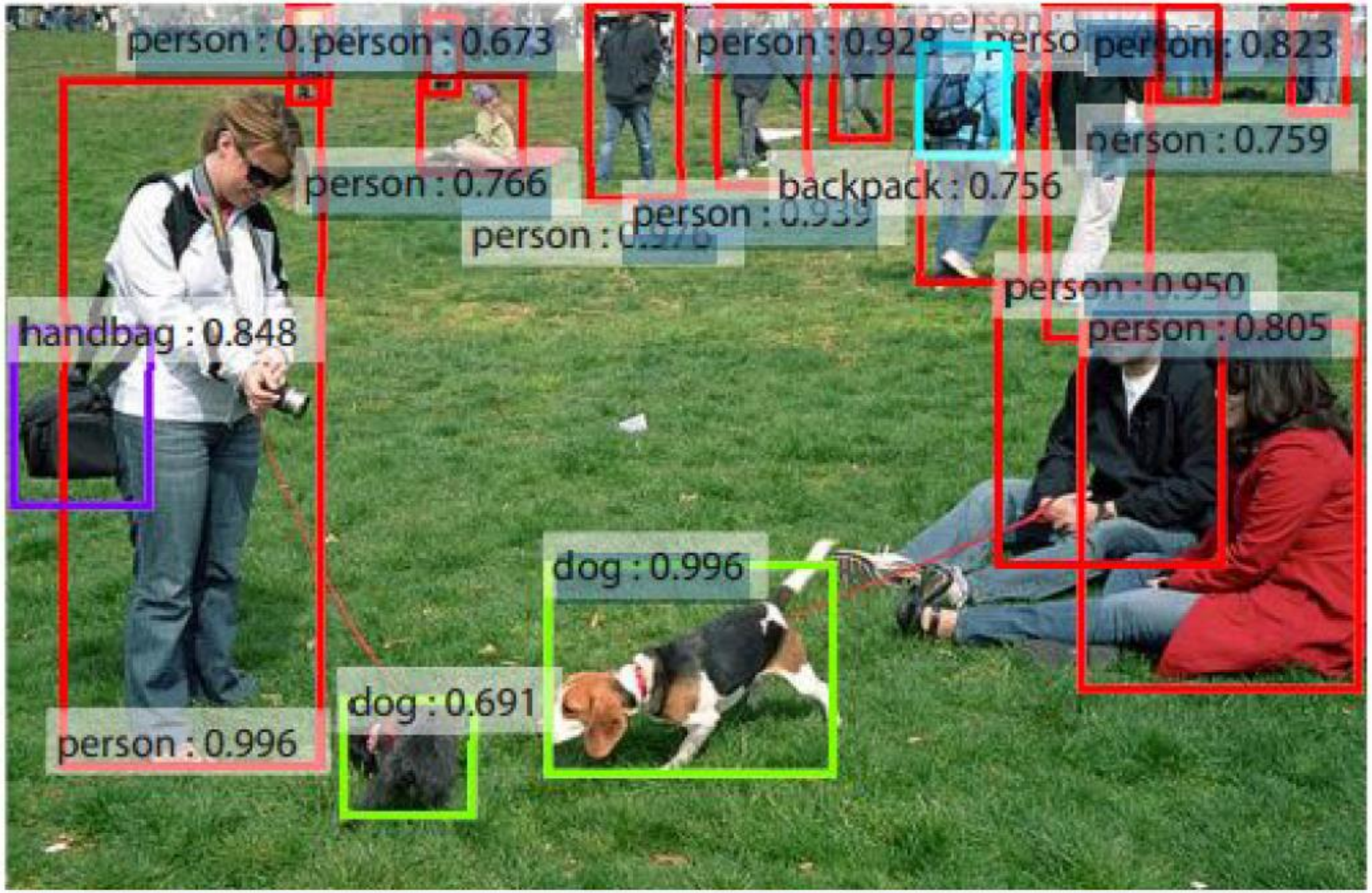


$$f(\mathbf{x}_n) = y'_n = P(+1|\mathbf{x}_n) \in [0,1]$$

$$\hat{y}_n = sign(P(+1|\mathbf{x}_n) - 0.5) \in \{1, -1\}$$

5.1 逻辑斯蒂回归问题

逻辑斯蒂回归应用示例



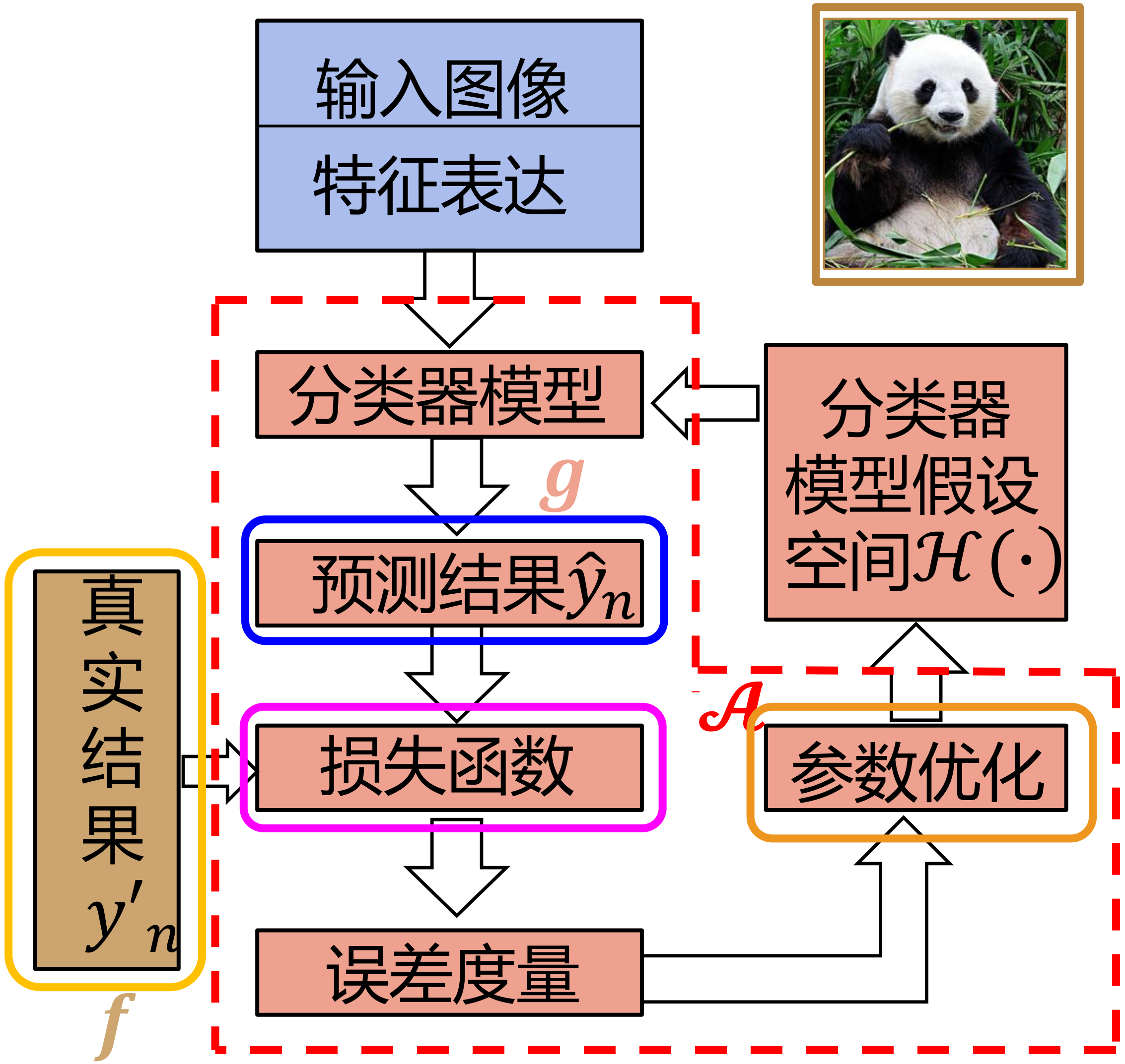
感知器算法：硬分类

逻辑斯蒂回归：软分类

（*“Soft” binary classification*）

Source: Faster RCNN, Ren

5.1 逻辑斯蒂回归问题



逻辑斯蒂回归：软分类

$$f(\mathbf{x}_n) = y'_n = P(+1|\mathbf{x}_n) \in [0,1]$$

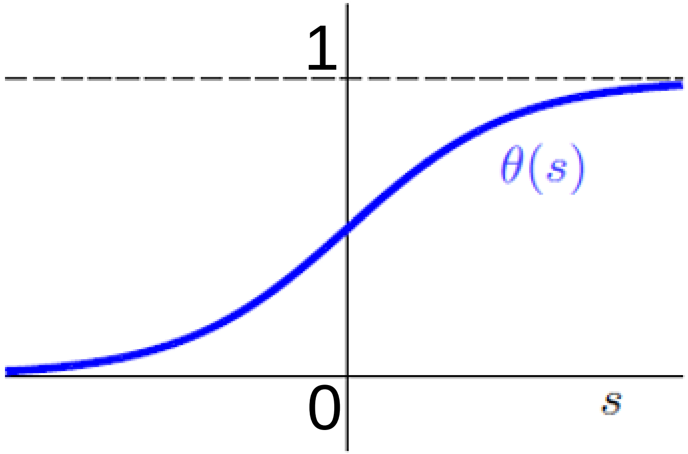
$$\hat{y}_n = sign(P(+1|\mathbf{x}_n) - 0.5) \in \{1, -1\}$$

$$\mathbf{x} = (x_1, x_2, \ldots x_d)^T$$

$$s = \sum_{i=0}^{d} w_i x_i$$

logistic hypothesis:

$$h(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x})$$



逻辑斯蒂函数

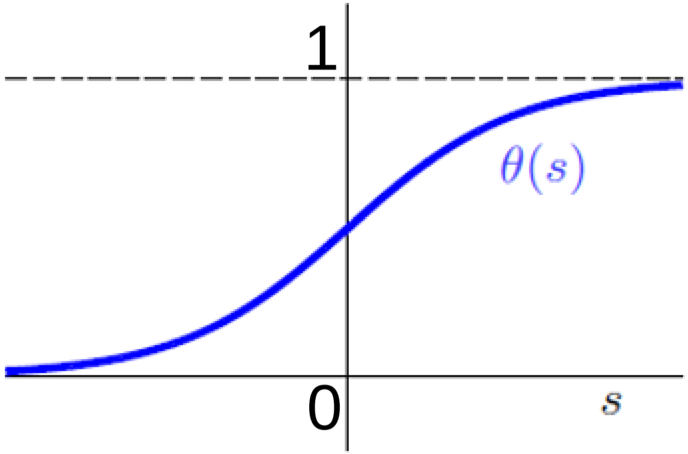
$\theta(-\infty) = 0;$ $\theta(0) = \frac{1}{2};$ $\theta(\infty) = 1$

$$\theta(s) = \frac{e^s}{1+e^s} = \frac{1}{1+e^{-s}}$$

---- Sigmoid 函数：平滑 (*Smooth*)、单调 (*Monotonic*)

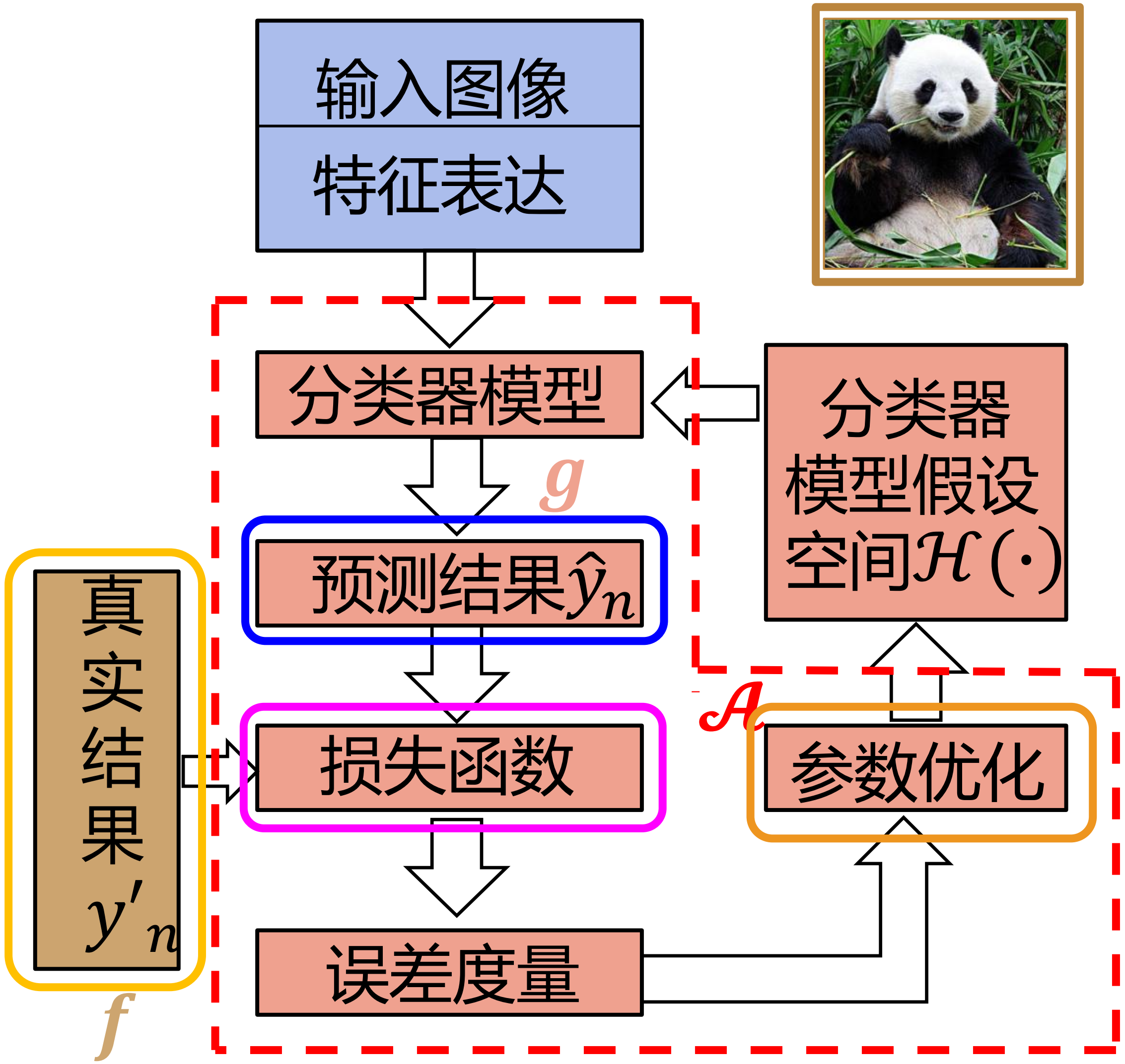
逻辑斯蒂回归用如下模型来估计$f(\mathbf{x}_n)$

$$h(\mathbf{x}_n) = \theta\left(\mathbf{w}^T\mathbf{x}_n\right) = \frac{1}{1+exp\left(-\mathbf{w}^T\mathbf{x}_n\right)}$$



Ref.: NTU-LIN

5.1 逻辑斯蒂回归问题



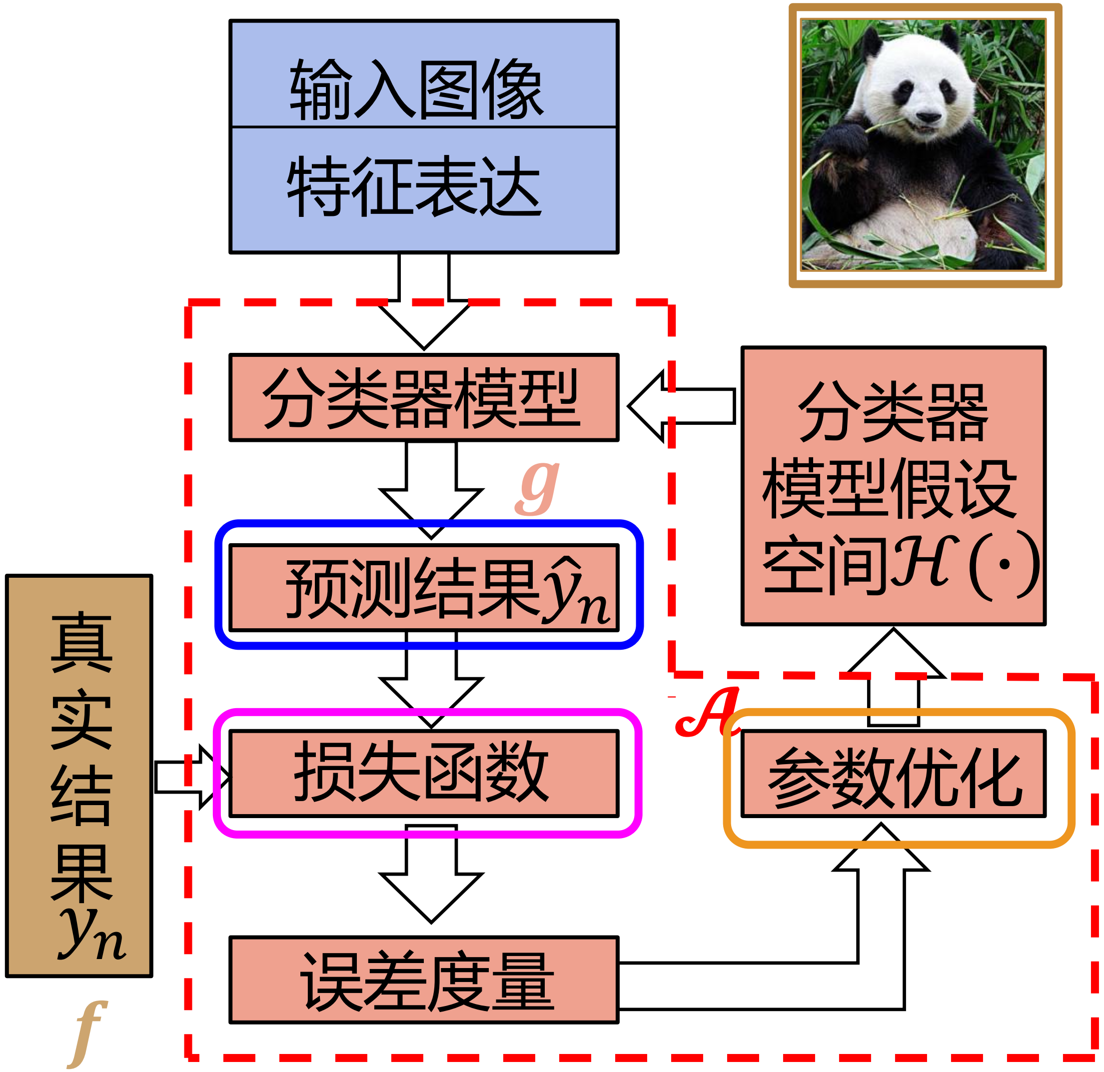
逻辑斯蒂回归：软分类

$$f(\mathbf{x}_n) = y'_n = P(+1|\mathbf{x}_n) \in [0,1]$$

$$\hat{y}_n = h(\mathbf{x}_n) = \theta(\mathbf{w}^T\mathbf{x}_n) = \frac{1}{1+exp(-\mathbf{w}^T\mathbf{x}_n)}$$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失



感知器(线性分类)

$$\hat{y}_{n(t)} = sign(\mathbf{w}_t^T \mathbf{x}_{n(t)})$$

$$L_{in} = \sum_{n=1}^{N} [\![y_n \neq \hat{y}_n]\!]$$

线性回归

$$\hat{y}_n = \mathbf{w}^T \mathbf{x}_n$$

$$L_{in}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{w}^T\mathbf{x}_n - y_n)^2$$

逻辑斯蒂回归

$$\hat{y}_n = h(\mathbf{x}_n) = \theta(\mathbf{w}^T\mathbf{x}_n) = \frac{1}{1 + exp(-\mathbf{w}^T\mathbf{x}_n)}$$

$$L_{in} = ?$$

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归

$$f(\mathbf{x}_n) = y'_n = P(+1|\mathbf{x}_n) \in [0,1]$$

(理想)训练样本：

$(\mathbf{x}_1, \; y'_1 = 0.9 = P(+1|\mathbf{x}_1))$

$(\mathbf{x}_2, \; y'_2 = 0.2 = P(+1|\mathbf{x}_2))$

$\vdots$

$(\mathbf{x}_N, \; y'_N = 0.6 = P(+1|\mathbf{x}_N))$

实际训练样本(含噪标签)：

$(\mathbf{x}_1, \; y_1 = \mathrm{o} = 1 \sim P(+1|\mathbf{x}_1))$

$(\mathbf{x}_2, \; y_2 = \times = -1 \sim P(+1|\mathbf{x}_2))$

$\vdots$

$(\mathbf{x}_N, \; y_N = \times = -1 \sim P(+1|\mathbf{x}_N))$

$$L_{in} = ?$$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归可以使用平方误差作为损失函数吗?

$$L_{in}(\mathbf{w}) = (\mathbf{w}^T\mathbf{x}_n - y)^2$$

5.2 逻辑斯蒂回归损失

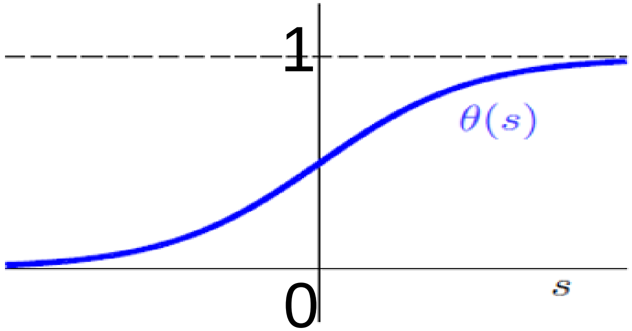
逻辑斯蒂回归可以使用平方误差作为损失函数吗?

$$L_{in}(\mathbf{w}) = (\theta(\mathbf{w}^T\mathbf{x}_n) - y')^2$$

逻辑斯蒂回归可以使用平方误差作为损失函数吗?

$$L_{in}(\mathbf{w}) = (\theta(y_n\mathbf{w}^T\mathbf{x}_n) - 1)^2$$

$$\frac{\partial L_{in}(\mathbf{w},\mathbf{x},y)}{\partial \mathbf{w}} = 2(\theta(y_n\mathbf{w}^T\mathbf{x}_n) - 1)\underbrace{\theta(y_n\mathbf{w}^T\mathbf{x}_n)(1\text{-}\theta(y_n\mathbf{w}^T\mathbf{x}_n))}_{\frac{\partial\theta(z)}{\partial z}}y_n\mathbf{x}_n^T$$



$$if\ \ (y\mathbf{w}^T\mathbf{x}) > 0 \quad \nabla L_{in}(\mathbf{w},\mathbf{x},y) = 0$$

$$\boxed{if\ \ (y\mathbf{w}^T\mathbf{x}) < 0 \quad \nabla L_{in}(\mathbf{w},\mathbf{x},y) = 0}$$

Ref.: NTU-LEE

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失

(*Cross-Entropy Loss*)

$$\boxed{L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))}$$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失

交叉熵介绍：

分布 p(x)
(Distribution p(x))

cross
entropy

分布 q(x)
(Distribution q(x))

$$H(p,q) = -\sum_{x} p(x)\ln(q(x))$$

Ref.: NTU-LEE

5.2 逻辑斯蒂回归损失

交叉熵介绍：

Distribution p(x)

$p(x=1)=f(x_n)$

$p(x=0)=1-f(x_n)$

cross entropy

Distribution q(x)

$q(x=1)=\theta(x_n)$

$q(x=0)=1-\theta(x_n)$

$$H(p,q)=-\sum_{x}p(x)\ln(q(x))$$

$$H(f(x_n),\theta(x_n)))=\sum_{n}-[f(x_n)ln\theta(x_n)+(1-f(x_n))ln(1-ln\theta(x_n))]$$

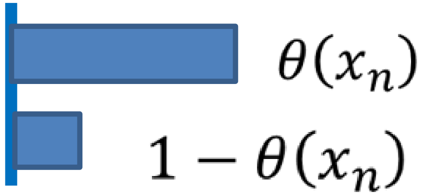
$$H(f(x_n),\theta(x_n)))=\sum_{n}-[ln\theta(x_n)]$$

Target function

$f(x_n)=1$

minimize

cross entropy



Ref.: NTU-LEE

5.2 逻辑斯蒂回归损失

交叉熵介绍：

Distribution p(x)

$p(x=1)=f(x_n)$

$p(x=0)=1-f(x_n)$

cross entropy

Distribution q(x)

$q(x=1)=\theta(x_n)$

$q(x=0)=1-\theta(x_n)$

$$H(p,q)=-\sum_{x}p(x)\ln(q(x))$$

$$s=w^T x$$

$$\theta(s)=\frac{1}{1+\exp(-s)}$$

$$H(f(x_n),\theta(x_n)))=\sum_{n}-[f(x_n)ln\theta(x_n)+(1-f(x_n))ln(1-ln\theta(x_n))]$$

$$H(f(x_n),\theta(x_n)))=\sum_{n}-[ln\theta(x_n)]$$

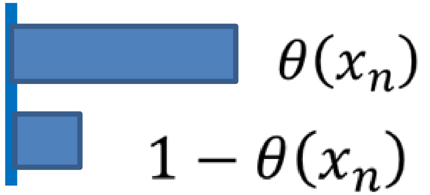
$$H(f(x_n),\theta(x_n)))=\sum_{n}[\ln(1+\exp(-ys)]$$

Target function

$f(x_n)=1$

minimize

cross entropy



Ref.: NTU-LEE

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失

(*Cross-Entropy Loss*)

$$\boxed{L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))}$$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失

(*Cross-Entropy Loss*)

$$\boxed{L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))}$$

Ref.: NTU-LIN

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失

(*Cross-Entropy Loss*)

$$L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

令：$s = \mathbf{w}^T \mathbf{x}$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失
(*Cross-Entropy Loss*)

$$L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-ys))$$

令： $s = \mathbf{w}^T \mathbf{x}$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + \exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失

(*Cross-Entropy Loss*)

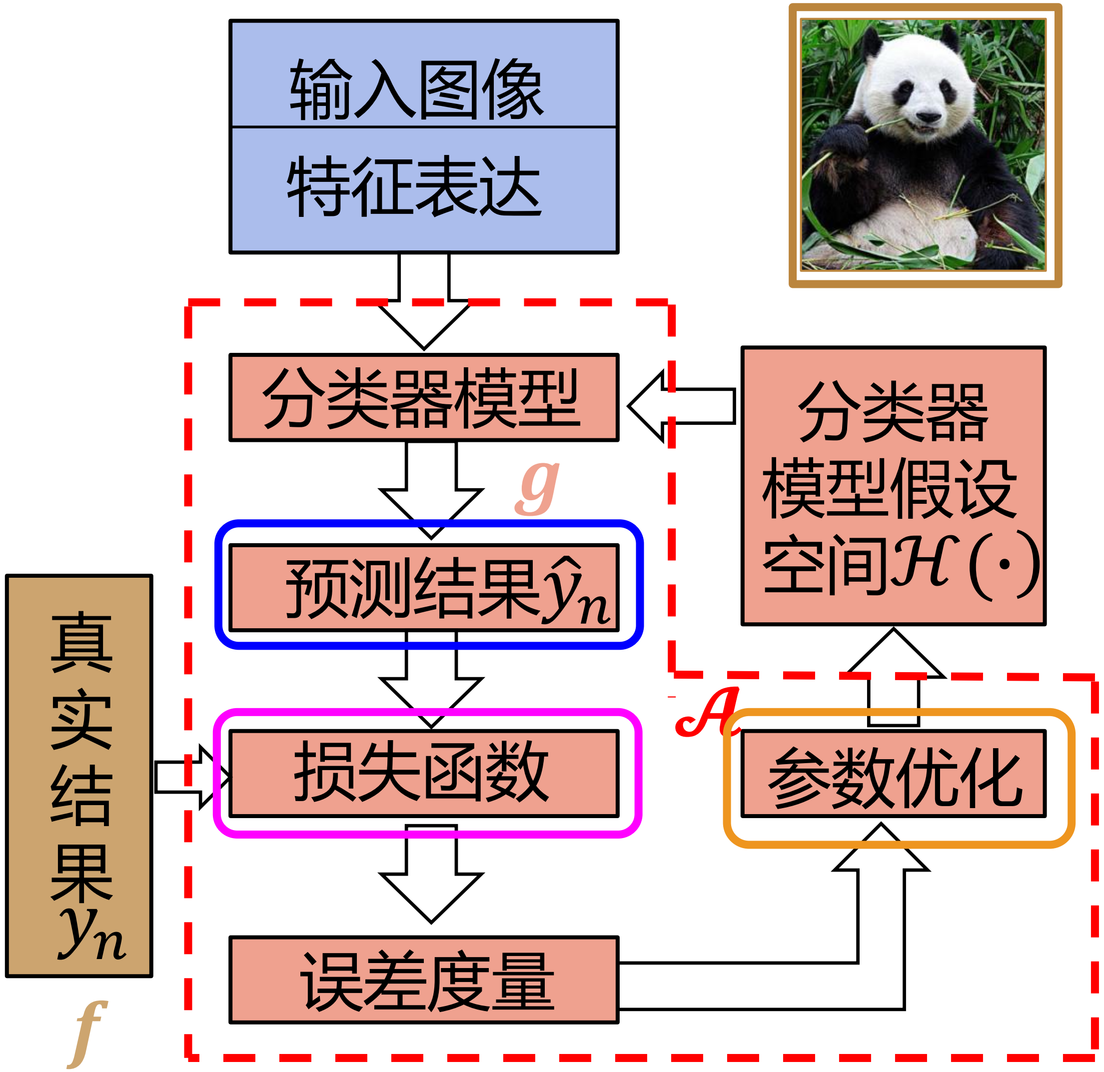
$$\boxed{L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-ys))}$$

令： $s = \mathbf{w}^T \mathbf{x}$

$$H(f(x_n), \theta(x_n))) = \sum_{n} [\ln(1 + \exp(-ys)]$$

Ref.: NTU-LIN

5.2 逻辑斯蒂回归损失



感知器(线性分类)

$$\hat{y}_{n(t)} = \text{sign}(\mathbf{w}_t^T \mathbf{x}_{n(t)})$$

$$L_{in} = \sum_{n=1}^{N} [\![y_n \neq \hat{y}_n]\!]$$

线性回归

$$\hat{y}_n = \mathbf{w}^T \mathbf{x}_n$$

$$L_{in}(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} (\mathbf{w}^T \mathbf{x}_n - y_n)^2$$

逻辑斯蒂回归

$$\hat{y}_n = h(\mathbf{x}_n) = \theta(\mathbf{w}^T \mathbf{x}_n) = \frac{1}{1 + exp(-\mathbf{w}^T \mathbf{x}_n)}$$

$$L_{in} = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

5.2 逻辑斯蒂回归损失

逻辑斯蒂回归的最佳解：

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N}\sum_{n=1}^{N} -ln\theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

$$\theta(y_n \mathbf{w}^T \mathbf{x}_n) = \frac{1}{1+\exp(-y_n \mathbf{w}^T \mathbf{x}_n)}$$

$$g = \underset{\mathbf{w}}{argmin} \frac{1}{N}\sum_{n=1}^{N} \ln(1+exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

交叉熵损失
(*Cross-Entropy Loss*)

$$L_{in} = \frac{1}{N}\sum_{n=1}^{N} \ln(1+exp(-y_n \mathbf{w}^T \mathbf{x}_n))$$

令：$s = \mathbf{w}^T \mathbf{x}$

$$L_{in} = \ln(1+exp(-ys))$$

交叉熵损失梯度：

$$L_{in} = \ln(\underbrace{1 + exp(\overbrace{-y_n \mathbf{w}^T \mathbf{x}_n}^{\bullet})}_{\blacksquare})$$

$$\frac{\partial L_{in}(\mathbf{w}, \mathbf{x}, y)}{\partial \mathbf{w}} = \frac{\partial \ln(\blacksquare)}{\partial \blacksquare} \frac{\partial (1 + \exp(\bullet))}{\partial \bullet} \frac{\partial (-y_n \mathbf{w}^T \mathbf{x}_n)}{\partial \mathbf{w}}$$

$$= \frac{1}{\blacksquare} \exp(\bullet) \left(-y_n \mathbf{x}_n^T\right) = \frac{\exp(\bullet)}{1 + \exp(\bullet)} (-y_n \mathbf{x}_n^T)$$

$$\nabla L_{in}(\mathbf{w}, \mathbf{x}, y) = \theta(-y_n \mathbf{w}^T \mathbf{x}_n)(-y_n \mathbf{x}_n^T)$$

Ref.: NTU-LIN

5. 2 逻辑斯蒂回归损失

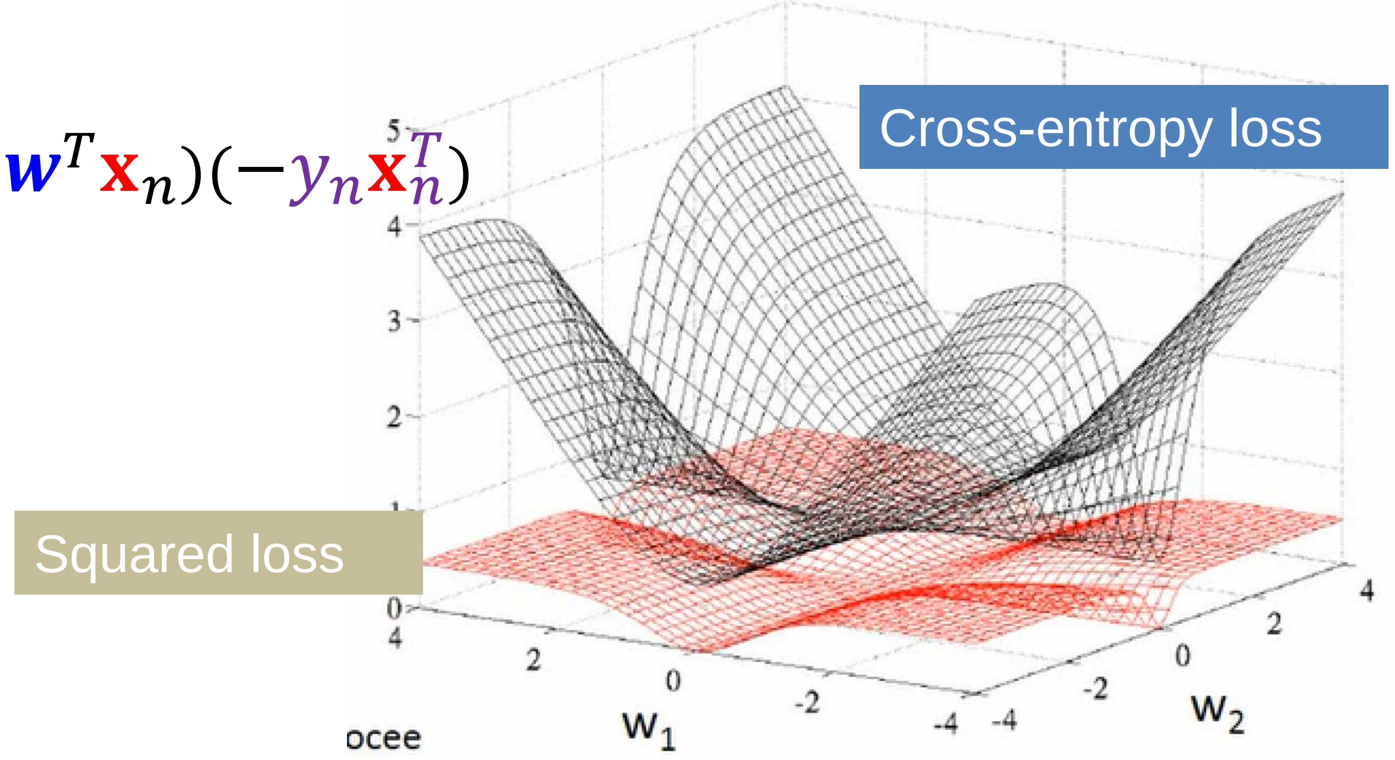
交叉熵损失与平方损失的梯度对比：

平方损失的梯度：

$$\nabla L_{in}(\mathbf{w},\mathbf{x},y) = 2(\theta(y_n\mathbf{w}^T\mathbf{x}_n) - 1)\theta(y_n\mathbf{w}^T\mathbf{x}_n)(1-\theta(y_n\mathbf{w}^T\mathbf{x}_n))y_n\mathbf{x}_n^T$$

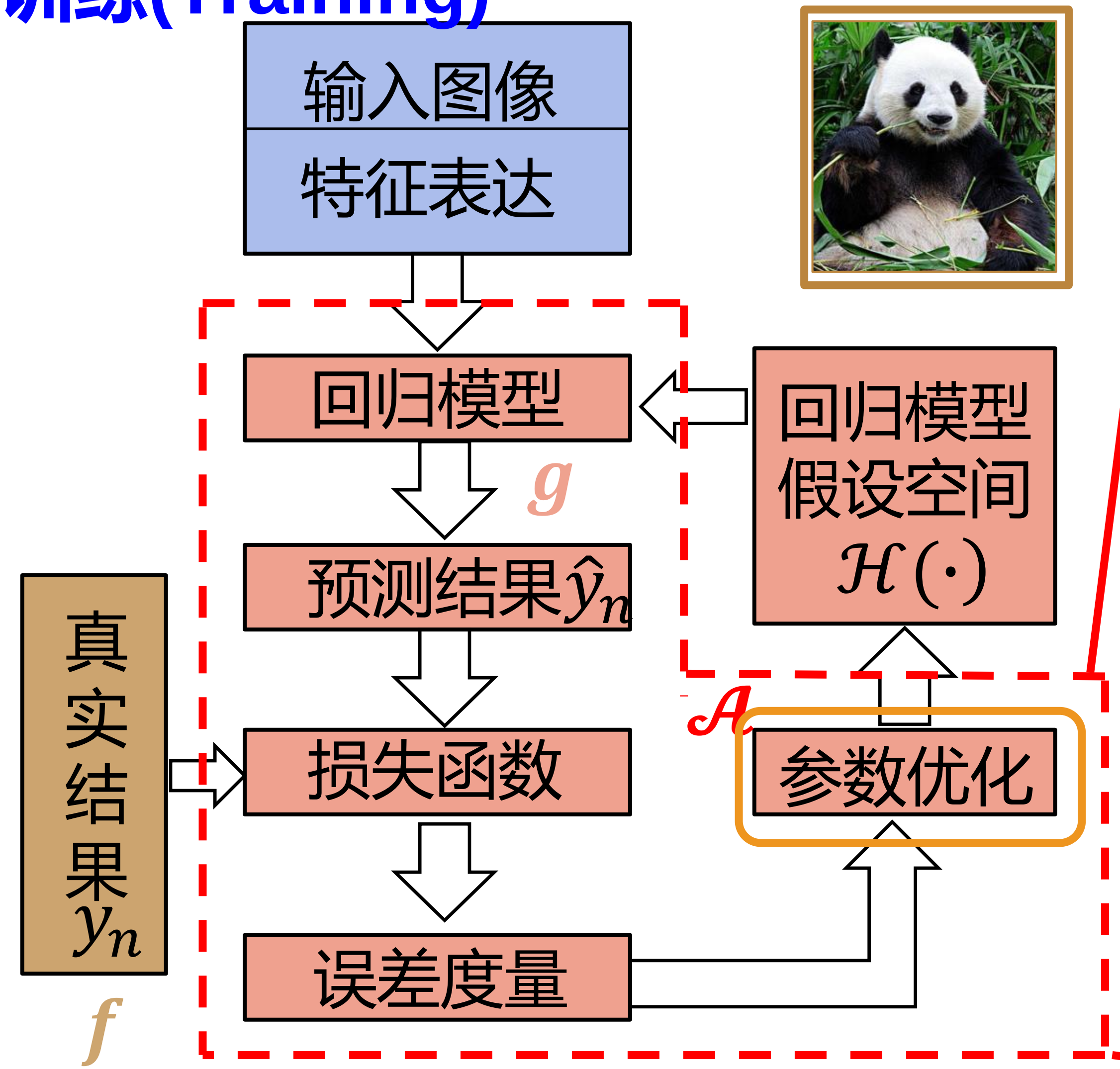
交叉熵损失的梯度：

$$\nabla L_{in}(\mathbf{w},\mathbf{x},y) = \theta(-y_n\mathbf{w}^T\mathbf{x}_n)(-y_n\mathbf{x}_n^T)$$



Ref.: NTU-LEE

训练(Training)



随机梯度下降法(SGD):

$$\nabla L_{in}(\mathbf{w}) = \sum_{n=1}^{B} (\mathbf{w}^T \mathbf{x}_n - y_n)\mathbf{x}_n$$

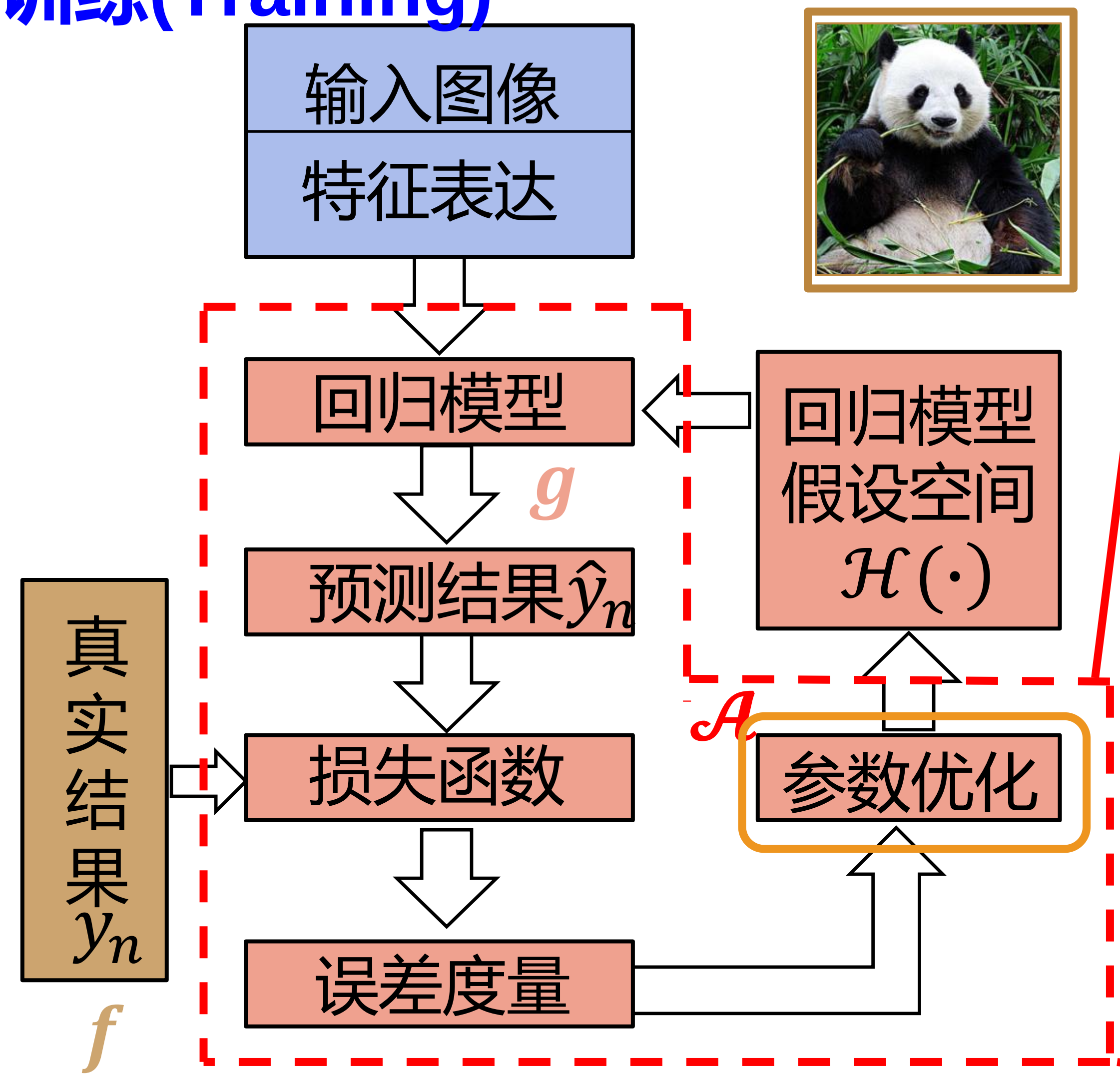
$\boldsymbol{m}_{i,t+1} = \lambda \boldsymbol{m}_{i,t}$ ...

$\mathbf{w}_{i,t+1} \leftarrow \mathbf{w}$ ...

- 问题1：学习率...
- 问题2：梯度为...佳解?
- 问题3：训练样本批...量大小的影响?
- 问题4：损失函数的影响?

第五讲
再讨论

训练(Training)



随机梯度下降法(SGD)：

$$\nabla L_{in}(\mathbf{w}) = \sum_{n=1}^{B} (\mathbf{w}^T \mathbf{x}_n - y_n)\mathbf{x}_n$$

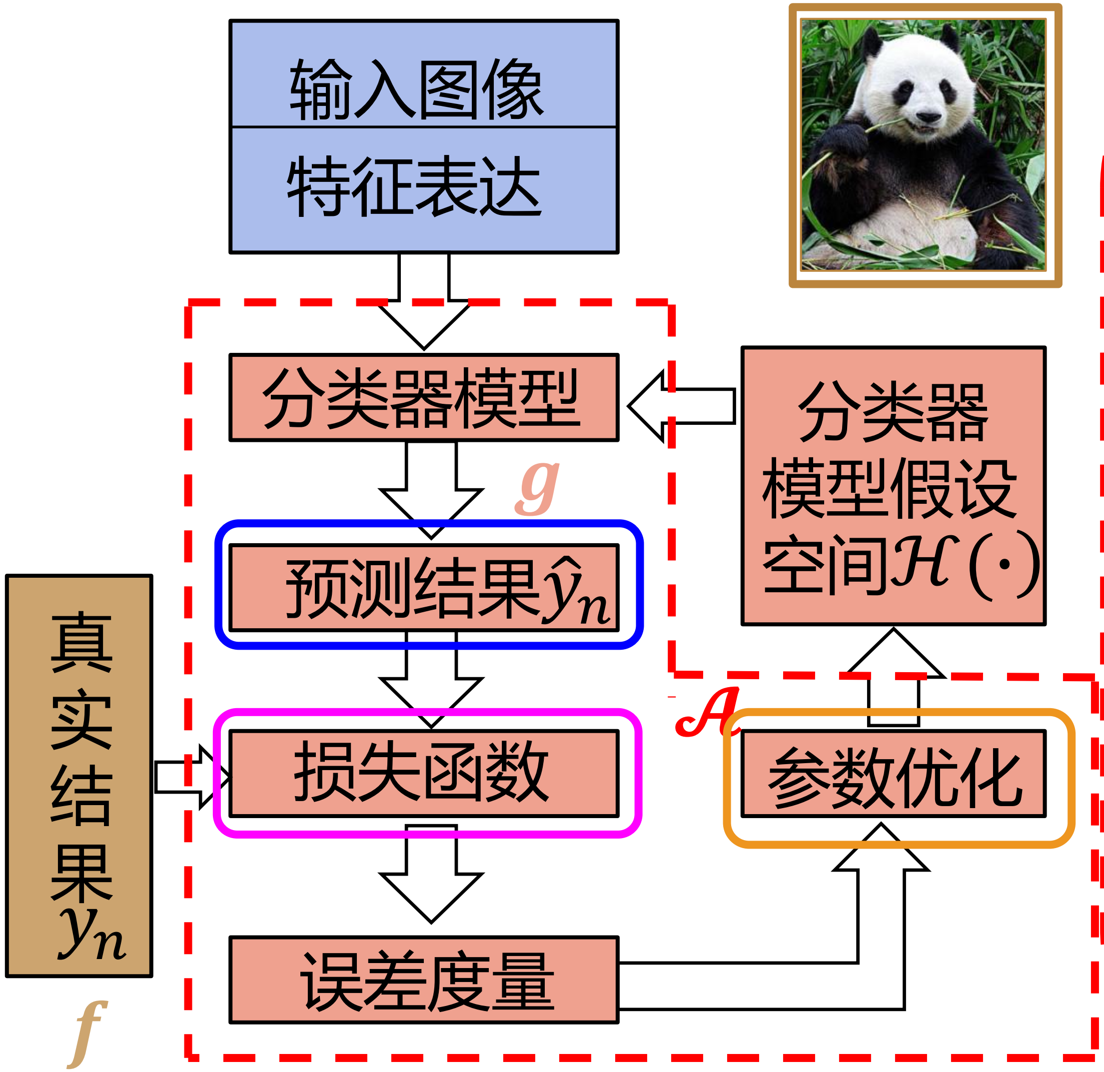
$$\mathbf{m}_{i,t+1} = \lambda \mathbf{m}_{i,t} \ldots$$

$$\mathbf{w}_{i,t+1} \leftarrow \mathbf{w} \ldots$$

不同损失函数在梯度下降时的速度不同

- 问题1：学习率…
- 问题2：梯度为…佳解？
- 问题3：训练样本…量大小的影响？
- 问题4：损失函数的影响？

5.2 逻辑斯蒂回归损失



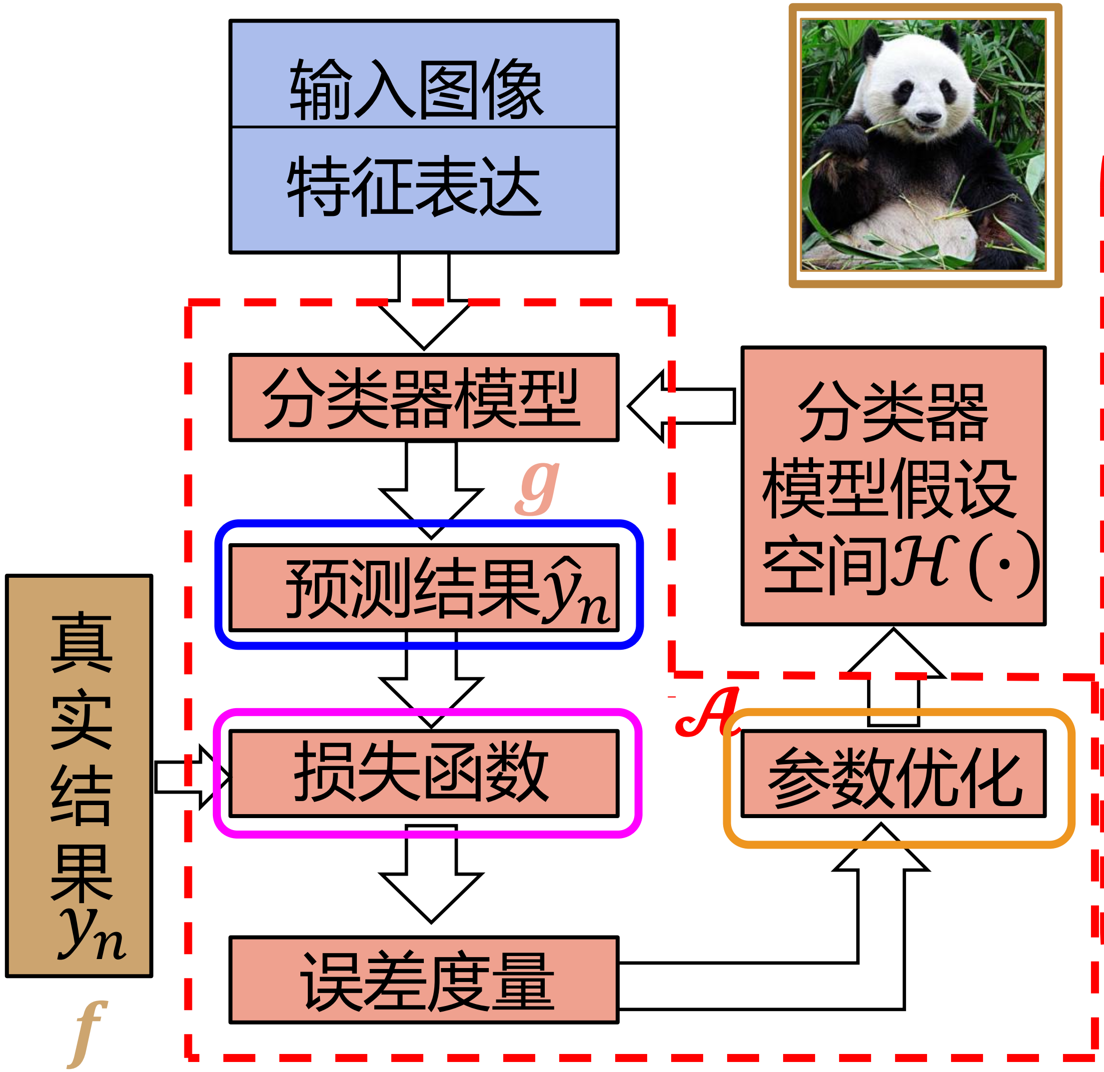
逻辑斯蒂回归

$$\hat{y}_n = h(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + exp(-\mathbf{w}^T\mathbf{x})}$$

$$L_{in} = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + exp(-y_n\mathbf{w}^T\mathbf{x}_n))$$

$$\nabla L_{in}(\mathbf{w}, \mathbf{x}, y) = \theta(-y\mathbf{w}^T\mathbf{x})(-y\mathbf{x}^T)$$

5.2 逻辑斯蒂回归损失



逻辑斯蒂回归

$$\hat{y}_n = h(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + exp(-\mathbf{w}^T\mathbf{x})}$$

$$L_{in} = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + exp(-y_n\mathbf{w}^T\mathbf{x}_n))$$

$$\nabla L_{in}(\mathbf{w}, \mathbf{x}, y) = \frac{1}{N}\sum_{n=1}^{N} \theta(-y_n\mathbf{w}^T\mathbf{x}_n)(-y_n\mathbf{x}_n^T)$$

梯度下降法实现逻辑斯蒂回归

- 初始化权向量$\boldsymbol{w}_0$
- $for\ t = 0,1,2,\ldots$ (t 代表迭代次数)

① 计算梯度：$\nabla L_{in}(\boldsymbol{w}_t) = \frac{1}{N}\sum_{n=1}^{N} \theta(-y_n \boldsymbol{w}_t^T \mathbf{x}_n)(-y_n \mathbf{x}_n)$

② 对权向量$\boldsymbol{w}_t$进行更新：$\boldsymbol{w}_{t+1} \leftarrow \boldsymbol{w}_t - \eta \nabla L_{in}(\boldsymbol{w}_t)$

…直到$\nabla L_{in}(\boldsymbol{w}) = \mathbf{0}$，或者迭代足够多次数

返回最终的$\boldsymbol{w}_{t+1}$作为学到的g

梯度下降法实现逻辑斯蒂回归

Stochastic Gradient Descent(SGD)

- 初始化权向量$\boldsymbol{w}_0$
- $\boldsymbol{for}\ t = 0,1,2,\dots$ (t 代表迭代次数)

① 计算梯度：$\nabla L_{in}(\boldsymbol{w}_t) = \frac{1}{B}\sum_{n=1}^{B}\theta(-y_n \boldsymbol{w}_t^T \mathbf{x}_n)(-y_n \mathbf{x}_n)$

② 对权向量$\boldsymbol{w}_t$进行更新：$\boldsymbol{w}_{t+1} \leftarrow \boldsymbol{w}_t - \eta\nabla L_{in}(\boldsymbol{w}_t)$

…直到$\nabla L_{in}(\boldsymbol{w}) = \boldsymbol{0}$，或者迭代足够多次数

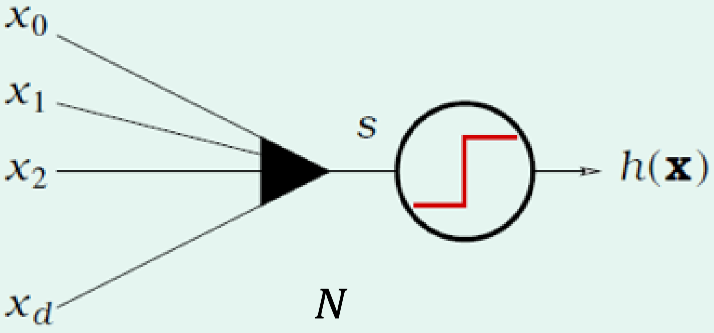
返回最终的$\boldsymbol{w}_{t+1}$作为学到的g

5.4 二元分类线性模型讨论

三种线性模型比较

线性分类(感知器):

$$h(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^T\mathbf{x})$$

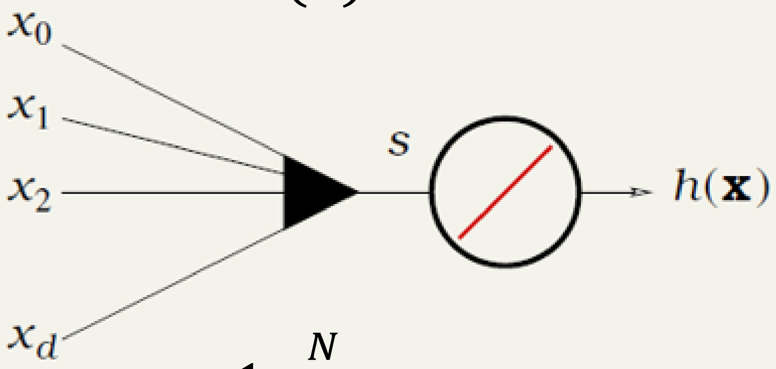


$$L_{in} = \sum_{n=1}^{N} [\![y_n \neq \hat{y}_n]\!]$$

NP难问题

线性回归:

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

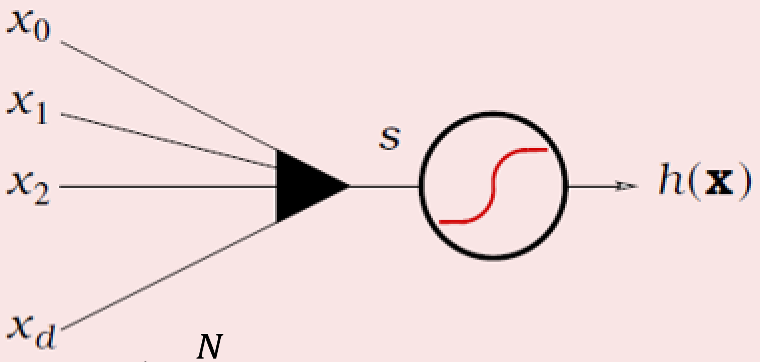


$$L_{in} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{w}^T\mathbf{x}_n - y_n\right)^2$$

凸函数、易优化、解析解

逻辑斯蒂回归:

$$h(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x})$$



$$L_{in} = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + exp(-y_n\mathbf{w}^T\mathbf{x}_n))$$

凸函数、平滑、梯度下降

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1,-1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s=\boldsymbol{w}^T\mathbf{x}$

线性分类(感知器)：

$$h(\mathbf{x})=\mathrm{sign}(s)$$

$$L_{in}=[\![h(\mathbf{x})\neq y]\!]$$

$$L_{0/1}(s,y)=[\![\mathrm{sign}(s)\neq y]\!]$$
$$=[\![\mathrm{sign}(ys)\neq 1]\!]$$

线性回归：

$$h(\mathbf{x})=s$$

$$L_{in}=(h(\mathbf{x})-y)^2$$

$$L_{sqr}(s,y)=(s-y)^2$$
$$=(ys-1)^2$$

逻辑斯蒂回归：

$$h(\mathbf{x})=\theta(s)$$

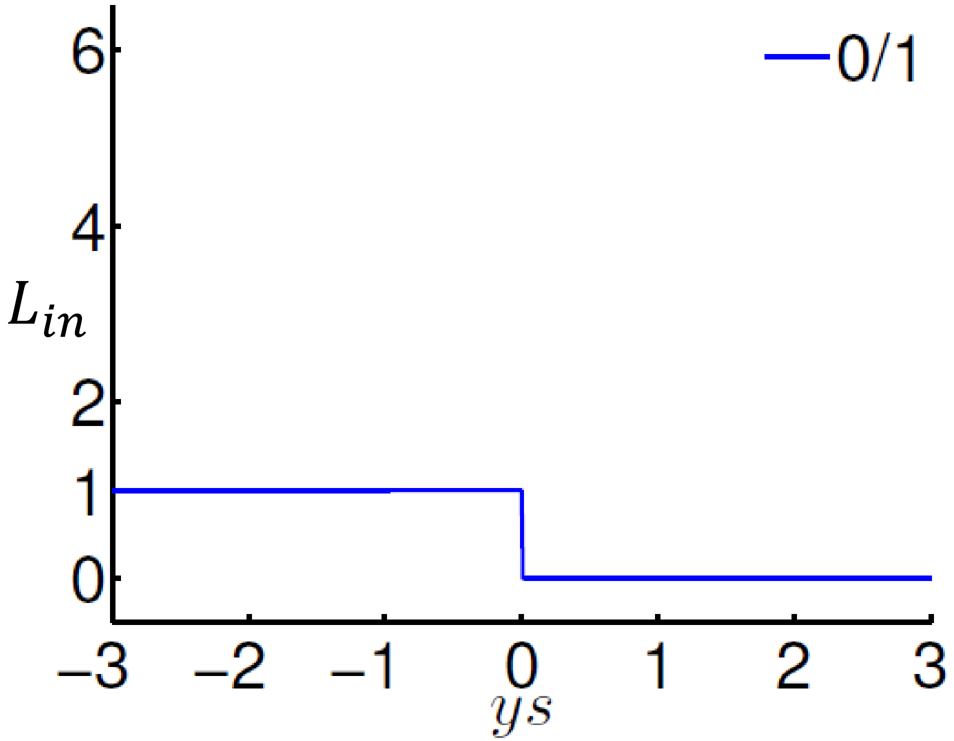
$$L_{in}=-\ln(h(y\mathbf{x}))$$

$$L_{ce}=\ln(1+exp(-ys))$$

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1, -1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s = \boldsymbol{w}^T\mathbf{x}$

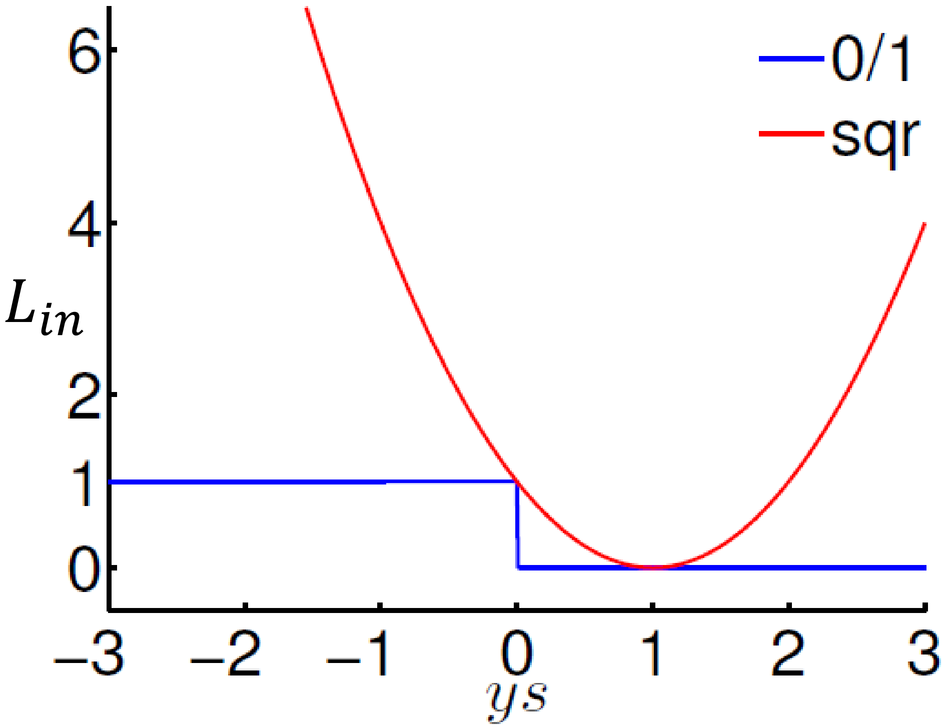


0/1 $L_{0/1}(s, y) = [\![sign(ys) \neq 1]\!]$

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1, -1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s = \boldsymbol{w}^T\mathbf{x}$



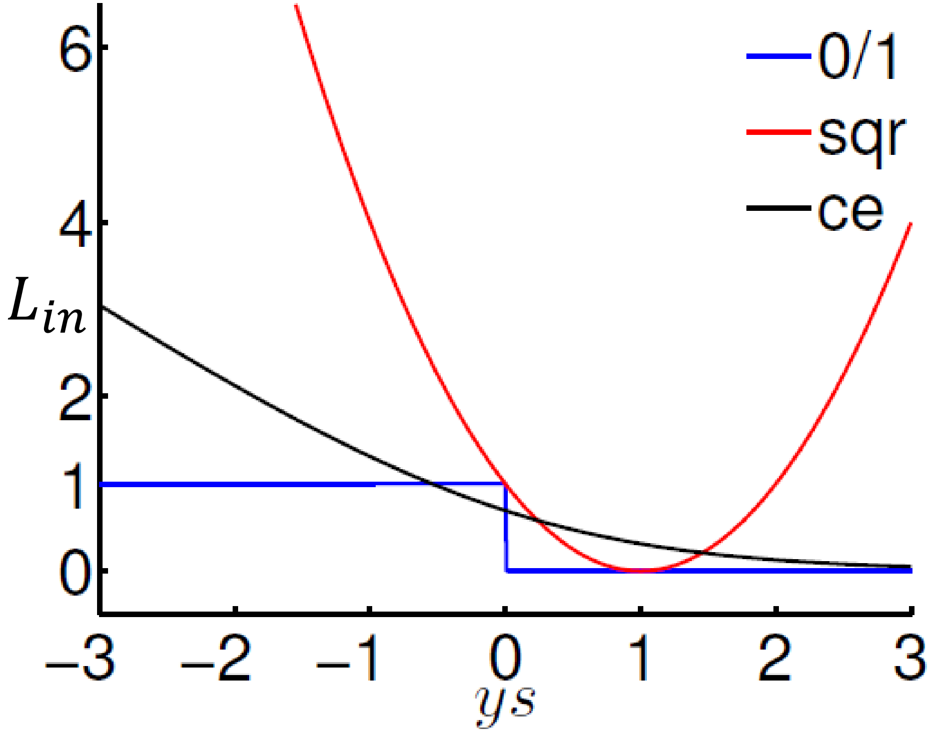
0/1 $L_{0/1}(s, y) = [\![sign(ys) \neq 1]\!]$

sqr $L_{sqr}(s, y) = (ys - 1)^2$

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1,-1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s = \boldsymbol{w}^T\mathbf{x}$



0/1 $L_{0/1}(s,y) = [\![sign(ys) \neq 1]\!]$

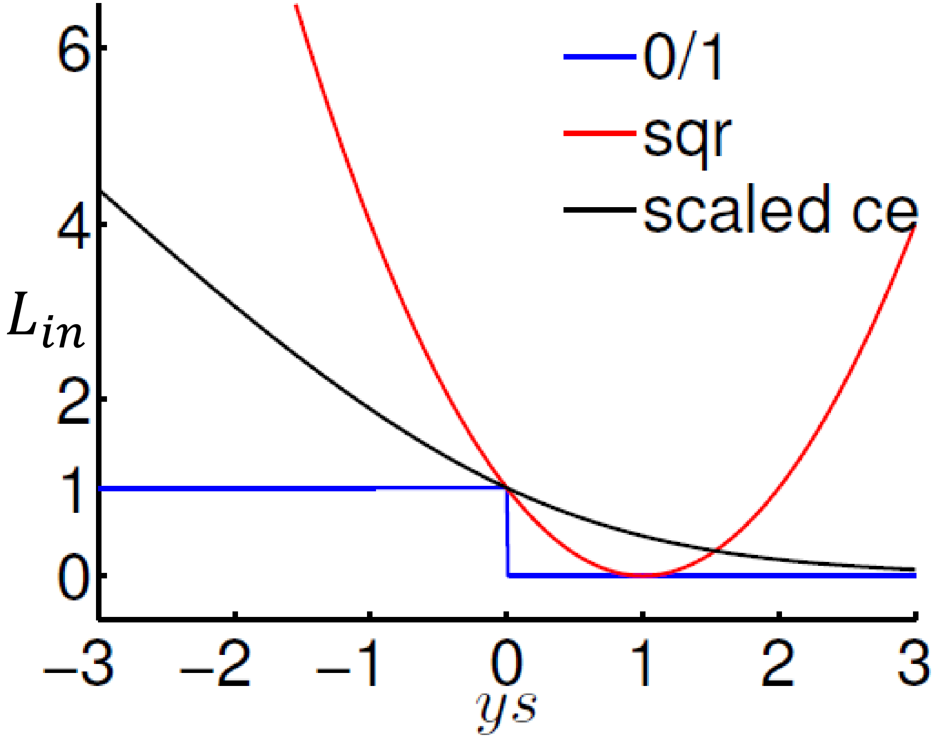
sqr $L_{sqr}(s,y) = (ys-1)^2$

ce $L_{ce}\ (s,y) = ln(1+\exp(-ys))$

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1, -1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s = \boldsymbol{w}^T\mathbf{x}$



0/1 $L_{0/1}(s,y) = [\![sign(ys) \neq 1]\!]$

sqr $L_{sqr}(s,y) = (ys-1)^2$

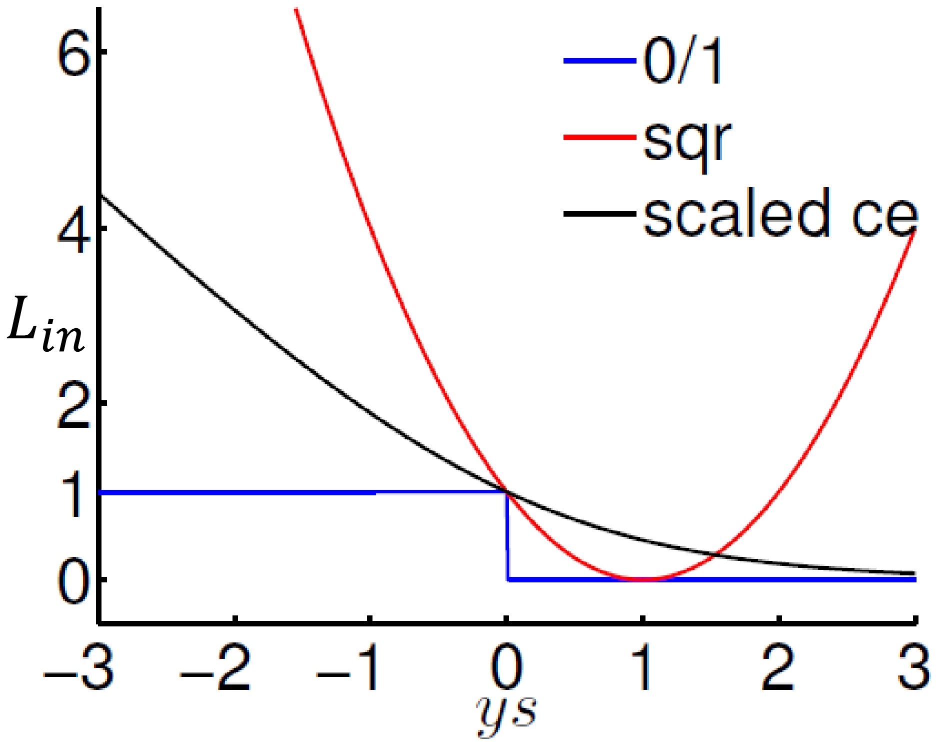
ce $L_{ce}(s,y) = ln(1+\exp(-ys))$

Scaled ce $L_{sce}(s,y) = log_2(1+\exp(-ys))$

Ref.: NTU-LIN

三种线性模型用于二元分类时(即： $y \in \{+1, -1\}$)损失函数比较

样本特征向量 $\mathbf{x}$ 与模型的权向量 $\boldsymbol{w}$ 的内积用 s 表示： $s = \boldsymbol{w}^T\mathbf{x}$



$$L_{0/1}(s,y) \le L_{sqr}(s,y)$$

$$L_{0/1}(s,y) \le L_{sce}(s,y)$$

$$L_{0/1}(s,y) \le L_{ce}(s,y)$$

训练或测试时，只要做到 $L_{sqr}(s,y)$ 或者 $L_{ce}(s,y)$ 很小， $L_{0/1}(s,y)$ 也会很小

线性回归与逻辑斯蒂回归可用于线性分类

Ref.: NTU-LIN

5.4 二元分类线性模型讨论

① 在标签为{+1，-1}的训练样本集 $\mathcal{D}$ 上运行线性回归/逻辑斯蒂回归算法，得到$\boldsymbol{w}^*$

② 返回分类结果：$g(\mathbf{x}) = sign({\boldsymbol{w}^*}^T \mathbf{x})$

线性分类(感知器)：

优点：样本线性可分时，算法收敛有理论保障

不足：样本非线性可分时NP难问题，可用Pocket算法实现

线性回归：

优点：凸函数，最容易优化，有解析解

不足：当$|ys|$很大时，$L_{0/1}(s,y)$的上界过于宽松

逻辑斯蒂回归：

优点：凸函数，易于优化

不足：当$ys \ll 0$时，$L_{0/1}(s,y)$的上界过于宽松

Ref.: NTU-LIN

第五讲　逻辑斯蒂回归（*Logistic Regression*）

5.1 逻辑斯蒂回归问题

模型的理论输出为概率值，分类面假设空间模型用Sigmoid函数

5.2 逻辑斯蒂回归损失

用交叉熵(cross-entropy)作为损失函数

5.3 逻辑斯蒂回归算法

用梯度下降法迭代实现参数更新

5.4 二元分类线性模型讨论

三个线性模型的特点及用途