① SVM Loss.

$L=\sum_{i=1}^{n} L_i$. $\quad L_i=\sum_{j\neq y_i}\max(0,(Wx_i)_j-(Wx_i)_{y_i}+\Delta) \triangleq \sum_{j\neq y_i}\max(0,gap_i(j,y_i))$.

$\forall i\in\{1,\dots,n\}$, $\cancel{\frac{\partial L_i}{\partial W_{kl}}=\sum_{j\neq y_i}}$ $\frac{\partial L_i}{\partial W_{kl}}=\sum_{j\neq y_i} a_i^{(kl)}(j,y_i)$.

其中 $a_i^{(kl)}(j,y_i)=\begin{cases}1\{j=k\}x_{il}-1\{y_i=k\}x_{il}, & gap_i(j,y_i)>0\\ 0, & gap_i(j,y_i)<0\end{cases}$

$\Downarrow$

$\forall i\in\{1,\dots,n\}$, $\begin{cases} k=y_i\text{ 时}, & \frac{\partial L_i}{\partial W_{kl}}=\sum_{j\neq y_i}-x_{il}1\{gap_i(j,y_i)>0\} = \underbrace{-x_{il}\sum_{j\neq y_i}1\{gap_i(j,y_i)>0\}}\\ k\neq y_i\text{ 时}, & \frac{\partial L_i}{\partial W_{kl}}=\sum_{j\neq y_i}x_{il}1\{gap_i(j,y_i)>0\}1\{j=k\} \\ & =\underbrace{x_i 1\{gap_i(j,y_i)>0\}}.\end{cases}$ (*)

记 $1\{gap_i(j,y_i)>0\}=m(i,j)$ 则 (*) 可表为 $\begin{cases}k=y_i, & \frac{\partial L_i}{\partial W_{kl}}=-x_{il}\sum_{j\neq y_i}m(i,j)\\ k\neq y_i, & \frac{\partial L_i}{\partial W_{kl}}=x_{il}m(i,k).\end{cases}$

向量化，抛掉 l

$\begin{cases}k=y_i, & \frac{\partial L_i}{\partial W_k}=-x_i\sum_{j\neq y_i}m(i,j)\\ k\neq y_i, & \frac{\partial L_i}{\partial W_k}=x_i m(i,k)\end{cases}$

$\Downarrow$ 向量化，$\underbrace{\text{构造矩阵 } C(i,k)}=\begin{cases}-\sum_{j\neq y_i}m(i,j), & k=y_i\\ m(i,k), & k\neq y_i\end{cases}$

$\cancel{\frac{\partial L}{\partial W_k}=(C(1,k),\dots,C(n,k))}$ $\frac{\partial L_i}{\partial W_k}=x_iC(i,k).=C(i,k)x_i$

数

向量

$$\left(\frac{\partial L_i}{\partial W_k} = C(i,k)\, x_i \right)$$

$\Downarrow$ 向量化，抛掉 i

$$\frac{\partial L}{\partial W_k} = (C(1,k), \dots, C(n,k))\, X$$

$\Downarrow$ 抛掉 k.

$$\frac{\partial L}{\partial W} = C^T X.$$

---

② Softmax ~~Cross~~ (Cross-entropy loss)

$$L = \frac{1}{N} \sum_i L_i + \lambda \sum_k \sum_l W_{kl}^2 , \quad L_i = -\log\left(\frac{e^{s_{y_i}}}{\sum_j e^{s_j}} \right)$$

$$= -s_{y_i} + \log\left(\sum_j e^{s_j}\right). \quad (*)$$

同理，先分析 $\frac{\partial L_i}{\partial W_{kl}}$. 据(*)式可得 ((*)上的等式不要求，改用(*)式)

$$\frac{\partial L_i}{\partial W_{kl}} = -x_{il} 1\{k=y_i\} + \frac{1}{\sum_j e^{s_j}} \cdot \sum_j e^{s_j} 1\{k=j\}\, x_{il}.$$

($s = Wx_i$ ，一般 $1\{k=y_i\}$、$1\{k=j\}$ 的出现是因为取了 Wx_i 的 y_i 分量或 j 分量)

$\Downarrow$ 向量化，抛掉 l.

$$\frac{\partial L_i}{\partial W_k} = -x_i 1\{k=y_i\} + \frac{1}{\sum_j e^{s_j}} e^{s_k} \cdot x_i.$$

$$\left(\frac{\partial L_i}{\partial W_k} = -x_i 1\{k=y_i\} + \frac{1}{\sum_j e^{s_j}} e^{s_k} \cdot x_i \right)$$

$$\Downarrow$$

$$\frac{\partial L_i}{\partial W_k} = \begin{cases} k=y_i \text{ 时}, = \left[-1 + \frac{e^{s_k}}{\sum e^{s_j}}\right] x_i \\ k \neq y_i \text{ 时}, = \left[\frac{e^{s_k}}{\sum e^{s_j}}\right] \cdot x_i \end{cases}$$

$\Downarrow$ 向量化，构造 $A(i,k) = \begin{cases} \frac{e^{s_k}}{\sum e^{s_j}} - 1 & , k=y_i \\ \frac{e^{s_k}}{\sum e^{s_j}} & , k \neq y_i \end{cases}$

$$\frac{\partial L_i}{\partial W_k} = A(i,k)\, x_i$$

（A的计算中涉及exp()，为防止大数溢出，可用上下同乘常数的技巧

$$\frac{C e^{s_k}}{C \sum e^{s_j}} = \frac{e^{s_k - c'}}{\sum e^{s_j - c'}}$$，令 $c' = \max_j \{s_j\}$，则指数部分 $s_j - c'$

被中心化了）

$\Downarrow$ 向量化，抛掉 i

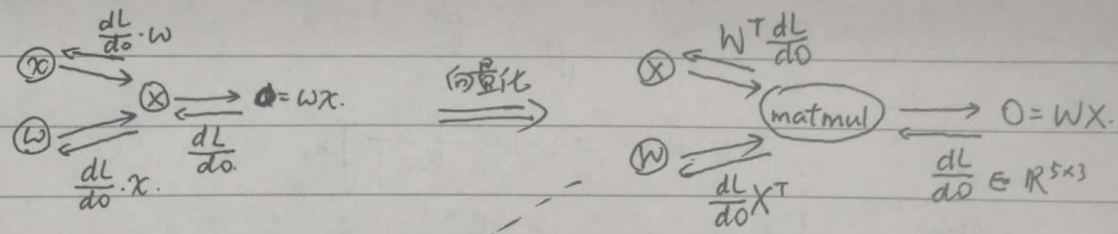
$$\frac{\partial L}{\partial W_k} = (A(1,k), \ldots, A(n,k))\, X.$$

$\Downarrow$ 抛掉 k.

$$\frac{\partial L}{\partial W} = A^T X.$$

③ Two Layers NN

重要1:



$(X \in \mathbb{R}^{10\times3},\ W \in \mathbb{R}^{5\times10},\ O \in \mathbb{R}^{5\times3})$.

(按照维度看是否转置).

(维度相同时套用此规律)

$$\left(\frac{\partial L}{\partial X} = \frac{\partial f}{\partial O}\cdot\frac{\partial O}{\partial X}, \quad \frac{\partial L}{\partial X_{kl}} = \sum_{i,j} \frac{\partial L}{\partial O_{ij}}\cdot\frac{\partial O_{ij}}{\partial X_{kl}} \right.$$

(因 $O = WX$,故 $j \neq l$ 时 $\frac{\partial O_{ij}}{\partial X_{kl}} = 0$)

$$= \sum_i \frac{\partial L}{\partial O_{il}}\cdot\frac{\partial O_{il}}{\partial X_{kl}}$$

$$= \sum_i \frac{\partial L}{\partial O_{il}}\cdot w_{ik}$$

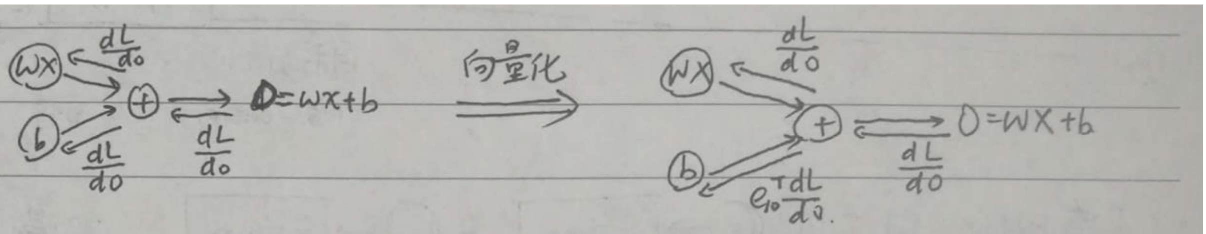
← 此处 l 是自由的!该式意义为 $\frac{dL}{dO}$ 的第 l 列与 W 的第 k 列内积作为 $\frac{dL}{dX}$ 的第 k 行第 l 列元素.

$$\Downarrow$$

$$\frac{\partial L}{\partial X} = W^T\frac{\partial L}{\partial O}.$$

关于 $\frac{\partial L}{\partial W}$ 可类似推导.)

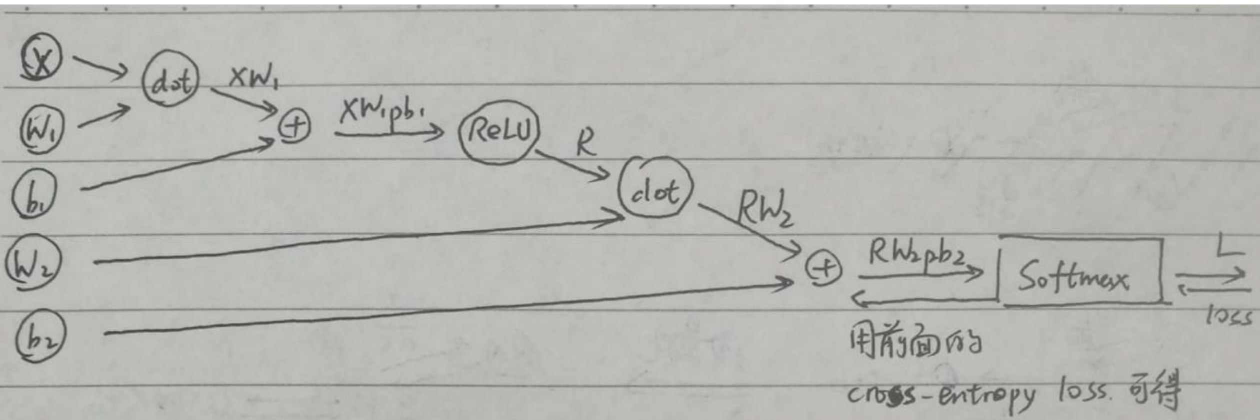
重要2:



$(WX \in \mathbb{R}^{10\times 3}, \ b \in \mathbb{R}^{1\times 3}, \ O \in \mathbb{R}^{10\times 3})$

(对于 broadcast 变量，因最终的损失L是在行坐标上求和
故梯度也应在行坐标上累加)

由上述两个重要性质，结合下述 computation graph，则可对 Two Layer NN 进行 BP.



(若有正则化，则应有 $W_1 \rightarrow \boxed{\text{Softmax}}$ 与 $W_2 \rightarrow \boxed{\text{Softmax}}$，勿漏!)