Softmax分类器

> 设线性模型为$x \cdot w$，其中w为(D,C)维矩阵，而x为(N,D)维矩阵。则他们的内积可以看做得分$score$，结果即为(N,C)维矩阵，每一行代表一个样本，对应的每一列代表该样本在这个分类时的得分s_{ij}

svm分类器在上式中用$Hinge\ Loss$作为损失函数，即$L = \sum_i \sum_{j \neq y_i} max(0, s_{ij} - s_{y_i} + 1)$，对于svm，当$s_{ij}$和$s_{y_i}$超过一定的安全距离时损失即为0，并且得分无法直观地给出每一分类的百分比。softmax将$score$压缩到0-1使其拥有概率意义，并且损失函数永远不会为0，使得模型可以一直被优化。

> softmax分类器：将上述得分$score$中的元素取以e为底的自然指数，可以发现，每类得分之间的差距变大了，这可以保证得分高的类所对应的概率更大；然后每一行的元素除以每一行元素的和，这样得分就有了概率意义。
>
> 现在我们希望分类正确的概率值越大越好，即 $max\ \sum_{i=1}^{N} p_{iy_i}$ 其中p_{iy_i}表示$score$第i行y_i列所对应的元素，也就是希望正确分类的概率越大。
>
> 所以可以将损失函数设置为 $L = -\sum_{i=1}^{N} logp_{iy_i}$ 希望p_{iy_i}越大，也就是$-logp_{iy_i}$最小，所以就得到了损失函数L
>
> 具体地，这里的损失函数其实是交叉熵损失的简单情况

梯度下降极小化损失函数

> $p_{iy_i} = \frac{e^{z_{y_i}}}{\sum_{c=1}^{C} e^{z_c}}$
>
> 其中z_c表示第c类原得分，具体的 $z_c = x_i \cdot w_c$ 将损失函数拆分 $L_i = -logp_{iy_i}$ 对z_c求导得：$\frac{\partial L_i}{\partial z_c} = \frac{\partial L_i}{\partial p_{iy_i}} \frac{\partial p_{iy_i}}{\partial z_c} \qquad = -\frac{1}{p_{iy_i}} \frac{\partial p_{iy_i}}{\partial z_c}$ 其中第二项求导需要分情况讨论：
>
> - 当$z_c = z_{y_i}$时：$\frac{\partial p_{iy_i}}{\partial z_{y_i}} = p_{iy_i}(1 - p_{iy_i})$
> - 当$z_c \neq z_{y_i}$时：$\frac{\partial p_{iy_i}}{\partial z_c} = -p_{iy_i} p_{ic}$
>
> 所以
>
> $$\begin{aligned} \frac{\partial L_i}{\partial z_c} &= \frac{\partial L_i}{\partial p_{iy_i}} \frac{\partial p_{iy_i}}{\partial z_c} \\ &= -\frac{1}{p_{iy_i}} \frac{\partial p_{iy_i}}{\partial z_c} \\ &= \begin{cases} p_{iy_i} - 1 & , z_c = z_{y_i} \\ p_{ic} & , z_c \neq z_{y_i} \end{cases} \end{aligned}$$
>
> 而z_c对w_c求导可得 $\frac{\partial z_c}{\partial w_c} = x_i$ 所以综上可得
>
> $$\frac{\partial L_i}{\partial w_c} = \begin{cases} x_i(p_{iy_i} - 1) & , z_c = z_{y_i} \\ x_i p_{ic} & , z_c \neq z_{y_i} \end{cases}$$