Logistic Regression

xzn

二元逻辑回归（回顾）

- 目标函数定义如下：
- $f(x) = P(label|x) \in [0,1]$
- 即在给定特征向量 x 的情况下，属于 $label$ 类的可能性多大

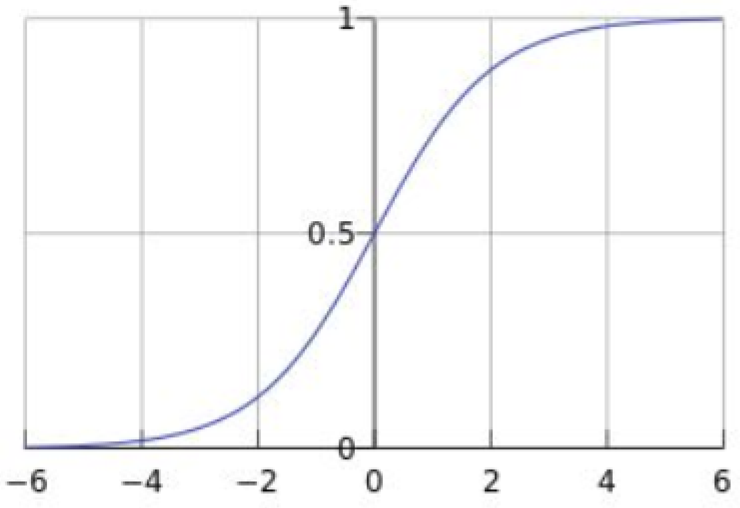
- 特征向量的每一个维度，都会对结果产生影响，所以我们希望可以给每一个特征给予一个带权重的分数：
- $s = \sum_{t=0}^{d} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$ （假设x是d维的）
- 表达式中的 w_i 表示第 i 维特征的权重，$w_i > 0$ 表示该特征对**正类别**有正面影响，且值越大，正面影响越大，反之亦然
- 得到了s之后，因为我们希望得到一个概率值，在（0,1）之间，所以我们经过一个sigmoid函数

首先我们要先介绍一下Sigmoid函数，也称为逻辑函数（Logistic function）：

- $g(z) = \dfrac{1}{1+e^{-z}}$

其函数曲线如下：

- 也即z>0我们预测其为正样本的概率大于0.5



- z<0我们预测其为正样本的概率小于0.5

从上图可以看到sigmoid函数是一个s形的曲线，它的取值在[0, 1]之间，在远离0的地方函数的值会很快接近0或者1。它的这个特性对于解决二分类问题十分重要

$$p(Y=1|X)=\frac{1}{1+e^{-z}}=\frac{e^{z}}{1+e^{z}}$$

$$p(Y=0|x)=1-\frac{1}{1+e^{-z}}=\frac{1}{1+e^{z}}$$

- 我们可以对公式做一点点小的变形，我们现在将e^{z}看成是计算出来的对正样本的贡献，其中z=$w^{t}x$。
- 可以这样理解成，我们计算出来正样本的贡献是e^{z}，我们规定负样本的贡献是1，那么正样本的概率就是正样本的贡献除以总的贡献也即$\frac{e^{z}}{1+e^{z}}$，同理负样本的概率是$\frac{1}{1+e^{z}}$

三元逻辑回归

- 思考这么一个问题
- 二元回归的时候，我们假设特征向量的每一个维度，都会对结果产生影响，所以我们希望可以给每一个特征给予一个带权重的分数。
- 不妨假设，我们这个权重代表了特征对样本是否是正样本所做的贡献，通过计算这个样本中的特征对正样本做的贡献的和，我们计算出样本属于正样本的概率，然后用1-正样本的概率得到结果。
- 如果考虑三元逻辑回归，假设每个样本有三个属性，正样本，中性样本，负样本，我们是否可以这样思考，我们考虑两组权重，一组计算正样本的贡献，一组计算负样本的贡献，中性样本贡献我们规定为1

- $\mathrm{Z_{pos}} = {W_{pos}}^T x$
- $\mathrm{Z_{neg}} = {W_{neg}}^T x$
- 正样本贡献记为$e^{Z_{pos}}$,负样本贡献记为$e^{Z_{neg}}$
- 那么有
- $\mathrm{p(Y = pos|X)} = \frac{e^{Z_{pos}}}{1+e^{Z_{pos}}+e^{Z_{neg}}}$
- $\mathrm{p(Y =} \ neg\mathrm{|X)} = \frac{e^{Z_{neg}}}{1+e^{Z_{pos}}+e^{Z_{neg}}}$
- $\mathrm{p(Y =} \ neutral\mathrm{|X)} = \frac{1}{1+e^{Z_{pos}}+e^{Z_{neg}}}$

更一般的对于n元logistic regression，我们有

$$p(Y=k|x)=\frac{e^{(W_k{}^T x)}}{1+\sum_{k=1}^{n-1} e^{(W_k{}^T x)}}, k=1,2,\cdots n-1$$

$$p(Y=k|x)=\frac{e^{(W_k{}^T x)}}{1+\sum_{k=1}^{n-1} e^{(W_k{}^T x)}}, k=1,2,\cdots n-1$$

$$p(Y=0|x)=\frac{1}{1+\sum_{k=1}^{n-1} e^{(W_k{}^T x)}}, k=1,2,\cdots n-1$$

为何规定Y=0贡献为1，可不可以也用加权和？
当然可以，上式变为，这就是传说中的softmax函数

$$p(Y=k|x)=\frac{e^{(W_k{}^T x)}}{\sum_{k=1}^{n} e^{(W_k{}^T x)}}, k=1,2,\cdots n$$

Softmax

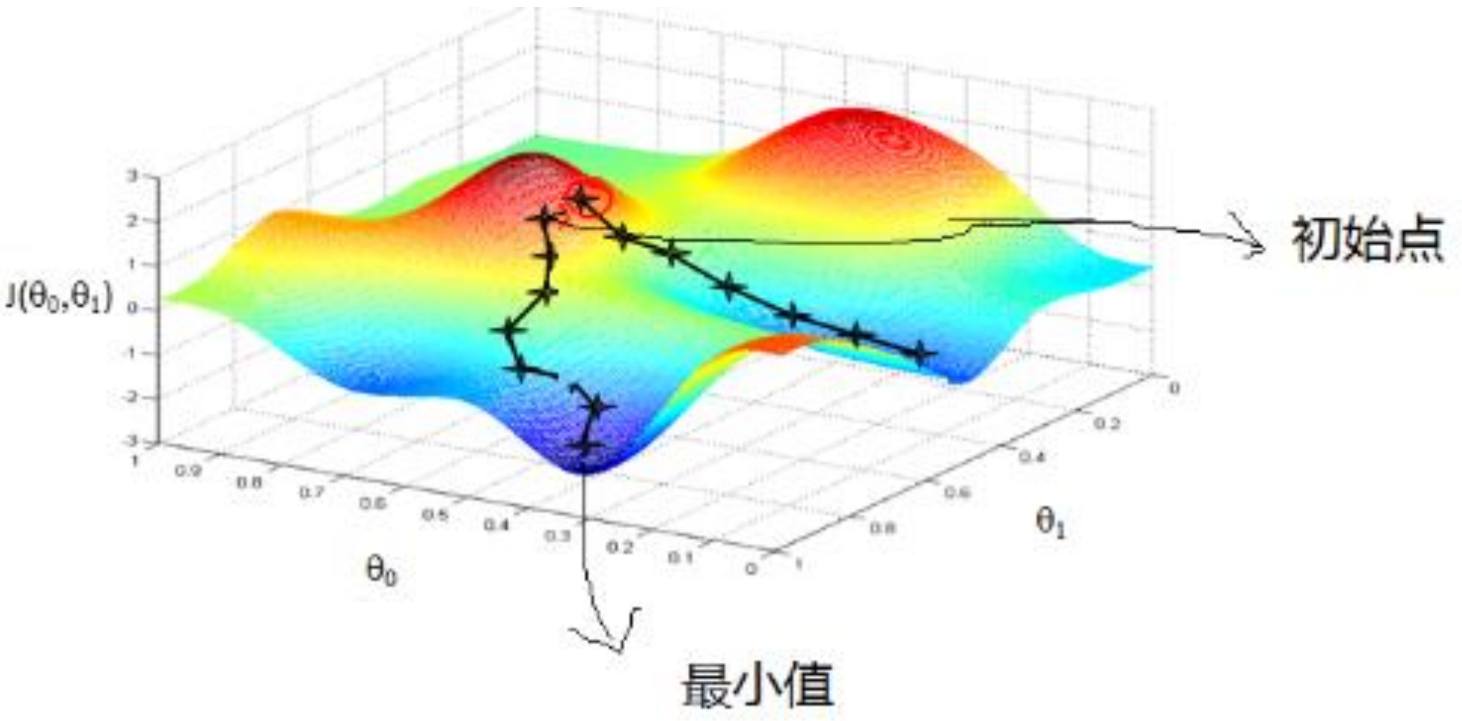
Softmax希望将数据转换成对应的概率值
但是这个概率不是均匀的。
举个栗子，假设有一个均匀的六面骰子，我们假设1~3都是小明获胜，4~5是小红获胜，6是小绿获胜，因为小红获胜的情况多所以小红获胜的概率自然就比较大，但是这里的概率是均匀的，你多一种可能就多一份胜出的概率。
但是softmax不是这样的思路，它觉得越大的数越应该更大比例的占有更大的概率，并且不是均匀的，通俗的说我比你多一份可能，可能比你要多两份的生出概率。

$$V = \begin{bmatrix} -3 \\ 2 \\ -1 \\ 0 \end{bmatrix}$$

$$S = \begin{bmatrix} 0.0057 \\ 0.8390 \\ 0.0418 \\ 0.1135 \end{bmatrix}$$

$$S_i = \frac{e^{v_i}}{\sum_{k=i}^{j} e^{v_k}}$$

梯度下降



二元逻辑回归

- **理论推导**
  - 利用 logistic 函数，我们可以构成一个新的假说模型：
  - $h(x) = \frac{1}{1+e^{-w^T x}}$
  - 要求解的是 w

  - 根据上面的假说模型，***h(x)* 算得的是属于正类的概率，属于负类别的概率即为 *1 - h(x)***
  - 当 h(x) 大于 0.5 的时候，说明该数据更大可能属于正类别；

逻辑回归

- **理论推导**
  - 那么我们可以把最开始提及的目标函数 *f(x)* 与 *h(x)* 联合起来：
  - $f(x) = P(label|x) = h(x)^{y}\left(1 - h(x)\right)^{1-y}$
  - y 表示 x 对应的分类标签
  - 当 $y = 1$，$f(x) = P(label|x) = h(x)$
  - 当 $y = 0$，$f(x) = P(label|x) = 1 - h(x)$

  - 用贝叶斯派的观点来看待这个问题
  - 不同的参数设置代表着不同的模型，**在某种模型下利用给定数据 *x* 得到给定标签 *y* 的概率，是这个问题中的似然（*likelihood*）**

逻辑回归

- **理论推导**
  - 考虑整个数据集，似然函数如下：
  - $likelihood = \prod_{i=1}^{M} P(label|x_i) = \prod_{i=1}^{M} h(x_i)^{y_i}\left(1 - h(x_i)\right)^{1-y_i}$
  - 根据最大似然估计算法，要找到一组模型参数，使得上式最大
  - 对 *likelihood* 取对数，再取负数之后，即可得到以下的函数：
  - $-log(likelihood) = -log \prod_{i=1}^{M} P(label|x_i)$
  - $= -\sum_{i=1}^{M} y_i log(h(x_i)) + (1 - y_i)log(1 - h(x_i))$
  - **对以上的函数取最小**，即达到最大似然的目的

逻辑回归

- **理论推导**
  - $-\sum_{i=1}^{M} y_i log(h(x_i)) + (1-y_i)log(1-h(x_i))$对 w 求导
  - 利用**梯度下降法**，通过不断地迭代使 w 逼近最优解直至收敛

$$\text{Repeat:}\ \ \tilde{\mathbf{W}}_{new}^{(j)} = \tilde{\mathbf{W}}^{(j)} - \eta \frac{\partial C(\tilde{\mathbf{W}})}{\partial \tilde{\mathbf{W}}^{(j)}}$$

$$= \tilde{\mathbf{W}}^{(j)} - \eta \sum_{i=1}^{n} \left[\left(\frac{e^{\tilde{\mathbf{W}}^{\mathrm{T}} \tilde{\mathbf{x}}_i}}{1 + e^{\tilde{\mathbf{W}}^{\mathrm{T}} \tilde{\mathbf{x}}_i}} - y_i \right) \tilde{\mathbf{X}}_i^{(j)} \right]$$

$$\text{Until convergence}$$

  - η 表示学习率，j 表示第几维, i 表示第几个样本

$$sigmoid(z) = \frac{1}{1+e^{-z}}$$

$$sigmoid'(z) = \frac{e^{-z}}{(1+e^{-z})^2} = \frac{e^{-z}+1-1}{(1+e^{-z})^2} = \frac{1}{1+e^{-z}} - \frac{1}{(1+e^{-z})^2} = sigmoid(z)(1-sigmoid(z))$$

$$y(x) = wx \qquad \frac{\partial y}{\partial w} = x$$

$$\mathrm{g}(\mathrm{w}) = ylog(h(wx)) + (1-y)log(1-h(wx))$$

$$\begin{aligned} g'(w) &= y\frac{h(wx)\big(1-h(wx)\big)x}{h(wx)} + (1-y)\frac{-h(wx)\big(1-h(wx)\big)x}{1-h(wx)} \\ &= xy\big(1-h(wx)\big) + x(y-1)h(wx) \\ &= \big(y-h(wx)\big)x \end{aligned}$$

逻辑回归

- **理论推导**
  - 综上，逻辑回归算法流程如下：
  - 输入：特征向量集合 {x}，标签集合 {y}
  - 输出：最优解 w
  - 初始化 w_0
  - 利用梯度下降法更新 w
  - **直至梯度为 0 或者迭代足够多次**
  - 利用最优 w 来预测测试集特征向量所对应的标签，计算属于正/负类别的概率

Given the following two-dimensional points and their actual labels:

$$\mathbf{x}_A = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad , \quad y_A = 0$$

$$\mathbf{x}_B = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad , \quad y_B = 0$$

$$\mathbf{x}_C = \begin{pmatrix} 3 \\ 3 \end{pmatrix} \quad , \quad y_C = 1$$

$$\mathbf{x}_D = \begin{pmatrix} 4 \\ 3 \end{pmatrix} \quad , \quad y_D = 1$$

If we initial the vector of weights for each dimension (including w_0) as

$\tilde{\mathbf{w}} = \begin{pmatrix} -5 \\ 2 \\ 1 \end{pmatrix}$. What's the vector of weights using Logistic Regression Model after only

one iteration by gradient decent (the learning rate $\eta = 0.1$)?

$$
\begin{aligned}
w_0^{new} &= w_0^{old} - \eta \sum \left[\left(\frac{e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}}{1+e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}} - y \right) x_0 \right] \\
&= -5 - \eta \left[\left(\frac{e^{-4}}{1+e^{-4}} - 0 \right) \times 1 + \left(\frac{e^{-2}}{1+e^{-2}} - 0 \right) \times 1 + \left(\frac{e^{4}}{1+e^{4}} - 1 \right) \times 1 + \left(\frac{e^{6}}{1+e^{6}} - 1 \right) \times 1 \right] \\
&= -5 - \eta \times 0.1167 \\
&= -5.0117
\end{aligned}
$$

$$
\begin{aligned}
w_1^{new} &= w_1^{old} - \eta \sum \left[\left(\frac{e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}}{1+e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}} - y \right) x_1 \right] \\
&= 2 - \eta \left[\left(\frac{e^{-4}}{1+e^{-4}} - 0 \right) \times 0 + \left(\frac{e^{-2}}{1+e^{-2}} - 0 \right) \times 1 + \left(\frac{e^{4}}{1+e^{4}} - 1 \right) \times 3 + \left(\frac{e^{6}}{1+e^{6}} - 1 \right) \times 4 \right] \\
&= 2 - \eta \times 0.0554 \\
&= 1.9945
\end{aligned}
$$

$$
\begin{aligned}
w_2^{new} &= w_2^{old} - \eta \sum \left[\left(\frac{e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}}{1+e^{\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}}} - y \right) x_2 \right] \\
&= 1 - \eta \left[\left(\frac{e^{-4}}{1+e^{-4}} - 0 \right) \times 1 + \left(\frac{e^{-2}}{1+e^{-2}} - 0 \right) \times 1 + \left(\frac{e^{4}}{1+e^{4}} - 1 \right) \times 3 + \left(\frac{e^{6}}{1+e^{6}} - 1 \right) \times 3 \right] \\
&= 1 - \eta \times 0.0758 \\
&= 0.9924
\end{aligned}
$$

Thus, $\tilde{\mathbf{w}}^{new} = \begin{pmatrix} -5.0117 \\ 1.9945 \\ 0.9924 \end{pmatrix}$.