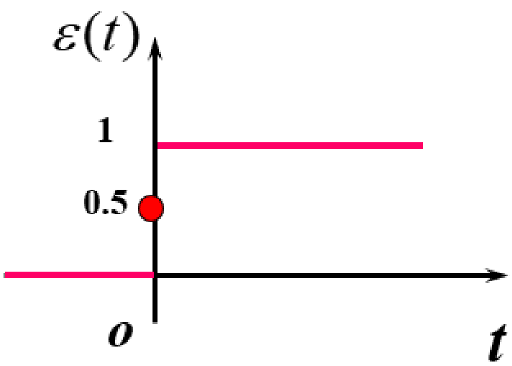
Logistics Regression

数学基础

1.什么是单位阶跃函数？

$$\epsilon(t) = \begin{cases} 0 & t < 0 \\ 0.5 & t = 0 \\ 1 & t > 0 \end{cases}$$



2.什么是单调可微函数？

在微积分学中，可微函数是指那些在定义域中所有点都存在导数的函数。可微函数的图像在定义域内的每一点上必存在非垂直切线。因此，可微函数的图像是相对光滑的，没有间断点、尖点或任何有垂直切线的点。

3.什么是指示函数？

$$I\{expression\} = \begin{cases} 1, & expression = True \\ 0, & expression = False \end{cases}$$

4.概率和统计的区别

一句话总结：**概率是已知模型和参数，推数据。统计是已知数据，推模型和参数。**
4.1.概率：已知一个模型和参数，怎么去预测这个模型产生的结果的特性（例如均值，方差，协方差等等）。 举个例子，我想研究怎么养猪（模型是猪），我选好了想养的品种、喂养方式、猪棚的设计等等（选择参数），我想知道我养出来的猪大概能有多肥，肉质怎么样（预测结果）。
4.2.统计：有一堆数据，要利用这堆数据去预测模型和参数。仍以猪为例。现在我买到了一堆肉，通过观察和判断，我确定这是猪肉（这就确定了模型。在实际研究中，也是通过观察数据推测模型是像高斯分布的、指数分布的、拉普拉斯分布的等等），然后，可以进一步研究，判定这猪的品种、这是圈养猪还是跑山猪还是网易猪，等等（推测模型参数）。

5.什么是先验、后验概率？

5.1.先验概率
事情发生前的预判概率。可以是基于历史数据的统计，可以由背景常识得出。也可以是人的主观观点给出。一般都是单独事件概率。
5.2.后验概率
事件发生后求的反向条件概率；或者说，基于先验概率求得的反向条件概率。概率形式与条件概率相同。
5.3.条件概率
一个时间发生后另一个事件发生的概率。一般的形式为$P(x|y)$表示y发生的条件下x发生的概率。
贝叶斯公式：

$$P(y|x) = \frac{P(x|y) * P(y)}{P(x)}$$

其中：
$P(y|x)$就是后验概率
$P(x|y)$就是后验概率
$P(x)$就是先验概率
$P(y)$就是先验概率

6.似然函数

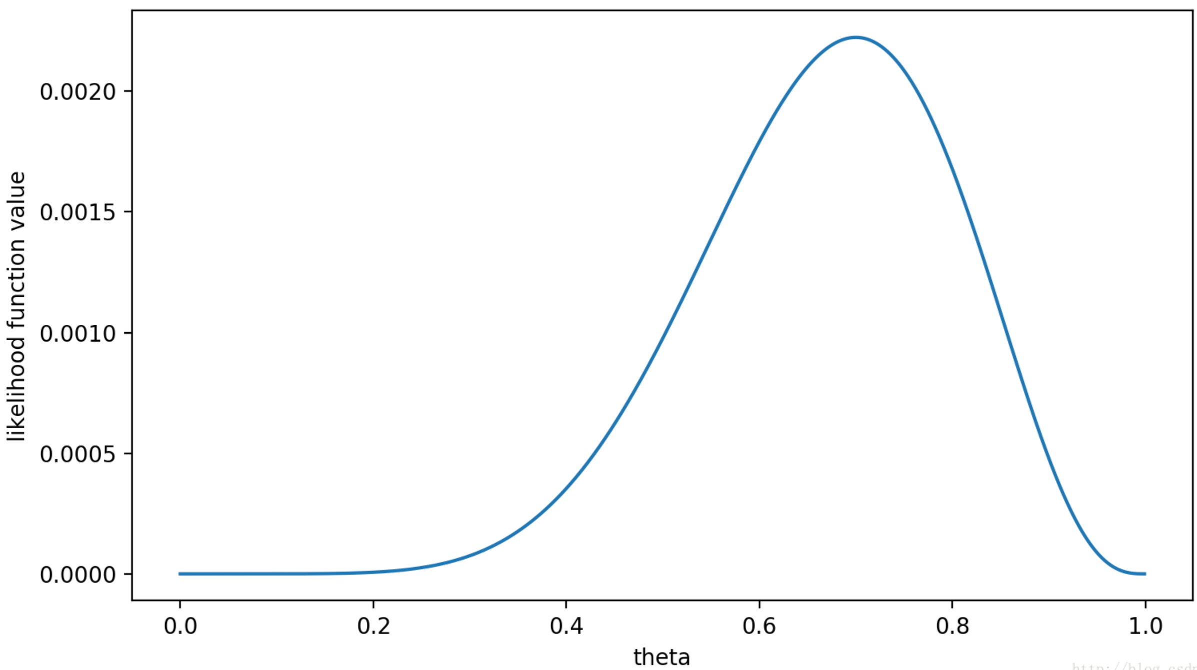
似然（likelihood）这个词其实和概率（probability）是差不多的意思，Colins字典这么解释：The likelihood of something happening is how likely it is to happen. 你把likelihood换成probability，这解释也读得通。但是在统计里面，似然函数和概率函数却是两个不同的概念（其实也很相近就是了）。
对于这个函数：$P(x|\theta)$输入有两个：x表示某一个具体的数据；θ表示模型的参数。
如果θ是已知确定的，x是变量，这个函数叫做概率函数(probability function)，它描述对于不同的样本点x，其出现概率是多少。
如果x是已知确定的，θ是变量，这个函数叫做似然函数(likelihood function), 它描述对于不同的模型参数，出现x这个样本点的概率是多少。
这有点像"一菜两吃"的意思。其实这样的形式我们以前也不是没遇到过。例如，$f(x,y)=x^y$, 即x的y次方。如果x是已知确定的(例如x=2)，这就是$f(y)=2^y$, 这是指数函数。
如果y是已知确定的(例如y=2)，这就是$f(x)=x^2$，这是二次函数。同一个数学形式，从不同的变量角度观察，可以有不同的名字。

7.什么是极大似然估计？

假设有一个造币厂生产某种硬币，现在我们拿到了一枚这种硬币，想试试这硬币是不是均匀的。即想知道抛这枚硬币，正反面出现的概率（记为θ）各是多少？
这是一个统计问题，回想一下，解决统计问题需要什么？ **数据！**
于是我们拿这枚硬币抛了10次，得到的数据x_0是：反正正正正反正正正反。我们想求的正面概率θ是模型参数，而抛硬币模型我们可以假设是 二项分布。
那么，出现实验结果x_0（即反正正正正反正正正反）的似然函数是多少呢？

$$f(x_0,\theta)=(1-\theta)\times\theta\times\theta\times\theta\times\theta\times(1-\theta)\times\theta\times\theta\times\theta\times(1-\theta)=\theta^7(1-\theta)^3=f(\theta)$$

注意，这是个只关于θ的函数。而最大似然估计，顾名思义，就是要最大化这个函数。我们可以画出f(θ)的图像：

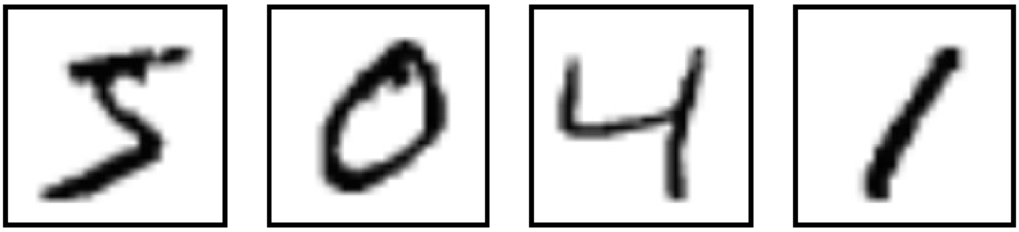


可以看出，在θ=0.7时，似然函数取得最大值。

Why?

Question 1

下面这几个数字分别是几？如何让机器判断？



Question 2

这是哈士奇吗？如何让机器判断？

Question 3

这个明星是谁？

现实中，我们经常遇到的问题有两类，一类是回归预测问题，第二类是分类问题。比如，上述的问题，或者是上电子邮箱时，需要机器判断邮件是否是垃圾邮件，从而获得更好的用户体验；需要通过机器检测某人是否患了某种病，等等。此时，假设我们用y表示预测结果，则y只有两个值，1或者0（是或者否）。相比于回归预测**y是连续的**来说，**y是离散的。**因此，线性回归就没法很好的解决这类分类问题了。我们需要找到一种映射，能够把线性回归的输出映射成只有0或者1的两种表示。即：如何让

$$z = \omega^T + b$$

的输出通过一个映射

$$g(.) -> (0, 1)$$

.

What?

二分类问题

1. 解决思路
明确目标：需要找到一个能够把线性输出的z映射成0或1的数学模型。利用这个模型，我们就可以预测一个物体属于某一类或者不输入某一类。
第一个想法：分段函数，线性输出的结果给个阈值，我们假使线性输出归一化到$(0, 1)$之间，那么我们是不是可以用0.5划界，小于0.5的为0，大于0.5的为1.但是这个想法存在缺陷。为什么？
第二个想法：我们找一个既能符合这种映射关系的连续函数，是不是就可以解决求导或者求偏微分的问题了。

- Logistic function

$$f(x) = \frac{1}{1+e^{-x}}$$

- hyperbolic tangent (shifted and scaled version of Logistic, above)

$$f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

- arctangent function

$$f(x) = \arctan x$$

- Gudermannian function

$$f(x) = \operatorname{gd}(x) = \int_0^x \frac{1}{\cosh t}\,dt$$

- Error function

$$f(x) = \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\,dt$$

- Generalised logistic function

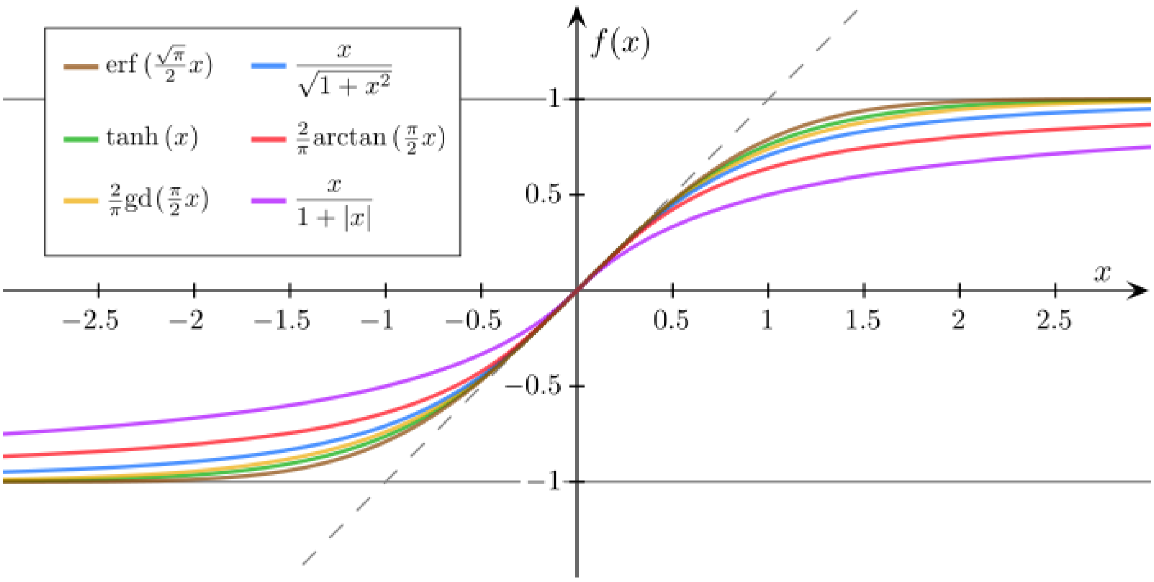
$$f(x) = (1+e^{-x})^{-\alpha}, \quad \alpha > 0$$

- Smoothstep function

$$f(x) = \begin{cases} \left(\int_0^1 \left(1-u^2\right)^N du\right)^{-1} \int_0^x \left(1-u^2\right)^N du & |x| \le 1 \\ \operatorname{sgn}(x) & |x| \ge 1 \end{cases} \quad N \ge 1$$

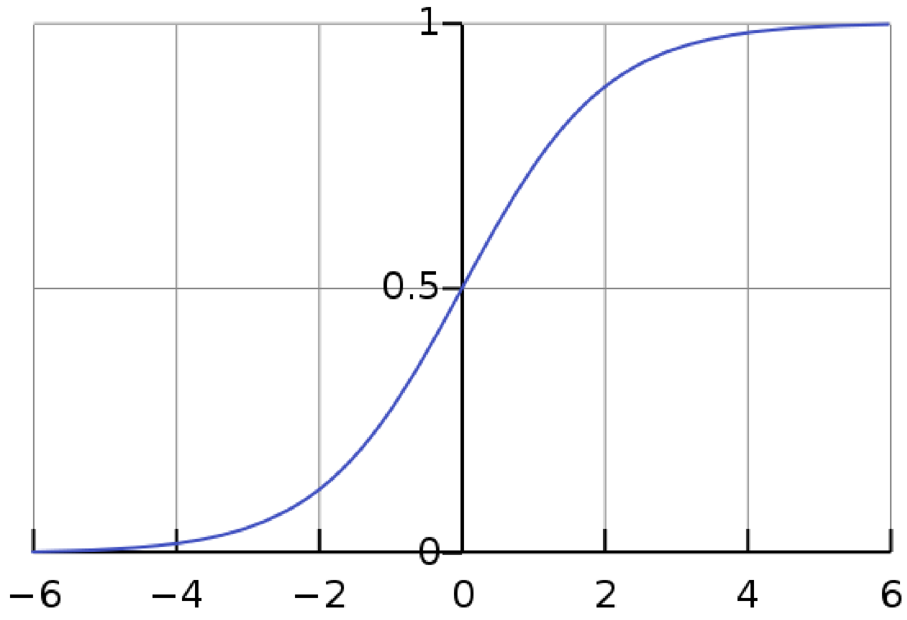
- Specific algebraic functions

$$f(x) = \frac{x}{\sqrt{1+x^2}}$$



最终我们选择了sigmod函数：

$$S(x) = \frac{1}{1+e^{-x}}$$



大家一定想知道为什么？这么多的函数，就选择这其中最不起眼的一个函数？（Talk）

性质：

1.函数连续，光滑，严格单调，以$(0, 0.5)$为对称中心，当x趋近负无穷时，y趋近于0；x趋近于正无穷时，y趋近于1；$x = 0$时，$y = 0.5$。Sigmoid函数的值域范围限制在$(0, 1)$之间，我们知道$[0, 1]$与概率值的范围是相对应的，这样sigmoid函数就能与一个概率分布联系起来了。

2.$f'(x) = f(x)(1 - f(x))$ 函数的求导是它本身函数的关系式。

推导过程：

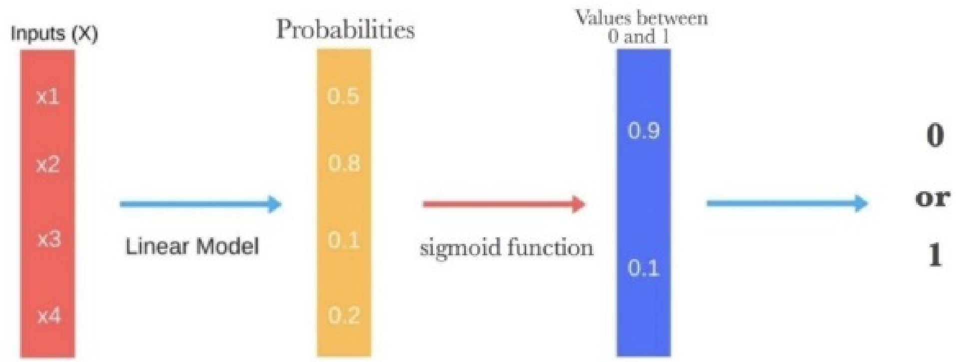
$$f(x) = \frac{e^x}{1 + e^x}$$

$$1 - f(x) = \frac{1}{1 + e^x}$$

$$f'(x) = \frac{e^x \times (1 + e^x) - e^x \times e^x}{(1 + e^x)^2}$$

$$f'(x) = \frac{e^x}{(1 + e^x)^2}$$

模型:



ok,我们得到一个比较好的数学模型，过程如上图所示：

1.首先将输入的数据x进行线性回归，即$z = wx + b$.

3.将z送入sigmod函数进行离散化成为0或者1.即 $y = \frac{1}{1+e^{-z}}$

这里，我们已经知道的参数是x和y,需要求的是什么？w和b.怎么求？这是一个什么问题？

没错！这是一个典型的根据数据求解模型最优参数的统计问题！

因此，我们需要利用似然估计来求解最优参数，但是我们得先建立一个**概率模型**，这样才能进行似然函数的计算。

1.引入概率知识：二分类问题对应的概率模型是二项分布，二项分布的概率模型是：

$$P(Y=1|x) = p$$
$$P(Y=0|x) = 1-p$$

2.我们令：

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

3.事件几率和对数几率：定义一个事件发生的几率表示为该事件发生和不发生的概率之比，即

$$\frac{p}{1-p}$$

,而该事件的对数几率定义为

$$log\,\frac{p}{1-p}$$

推导一下：

$$\frac{p}{1-p} = e^{(wx+b)}$$

$$log\,\frac{p}{1-p} = wx + b$$

这就可以换一个角度考虑：对输入的x进行分类的线性函数$wx+b$,其值域为实数域，通过我们的数学模型sigmodh函数可以将线性函数$wx+b$转换为概率： $P(Y=1|x) = p = \frac{1}{1+e^{-(wx+b)}}$

这里，我们就可以给我们之前使用的数学模型一个概率的含义，这样我们就可以用似然估计去估计我们的模型参数w和

b.

2. 似然估计求解最优w和b

已知条件：对于给定的数据集$T=\{(x_1,y_1),(x_2,y_2),...,(x_N,y_N)\}$其中$x_i\in R^n, y\in\{0,1\}$.**概率模型为：**

$$P(Y=1|x)=p=\frac{1}{1+e^{-(wx+b)}}$$

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

根据极大似然估计法，似然函数为：

$$g(w)=\prod_{i=1}^{N}p^{y_i}(1-p)^{1-y_i}$$

两边取对数：

$$L(w)=log(g(x))=\sum_{i=1}^{N}[y_i logp+(1-y_i)log(1-p)]$$

$$=\sum_{i=1}^{N}[y_i log\frac{p}{1-p}+log(1-p)]$$

$$=\sum_{i=1}^{N}[y_i(wx_i+b)-log(1+e^{wx_i+b})]$$

我们定义这个取负号就是我们的代价函数。

求$\hat{w}$:

$$\left.\frac{\partial L(w,b)}{\partial w}\right|_w=\frac{1}{N}\sum_{i=1}^{N}[(y_i-\hat{y}(x_i,w,b))x_i]$$

求$\hat{b}$:

$$\left.\frac{\partial L(w,b)}{\partial b}\right|_b=\frac{1}{N}\sum_{i=1}^{N}[(y_i-\hat{y}(x_i,w,b))]$$

3.根据求得的$\hat{w}$和$\hat{b}$进行预测

多分类问题

两种思路：

1. one vs all:

假如我们判断一个数字输入0-9其中的哪一个，按照这个策略，你可以训练 10 个二分类器，每个数字一个。这意味着训练一个分类器来检测 0，一个检测 1，一个检测 2，以此类推。当你想要对图像进行分类时，只需看看哪个分类器的预测分数最高。

2. softmax:

对于二分类来说，我们只有两个概率，要么为真的概率，要么为假的概率，对于多分类来说，我们的y有k个分类。因

此，我们需要求的概率如下：

$$h_\theta\left(x^{(i)}\right)=\begin{pmatrix}P\left(y^{(i)}=1\mid x^{(i)};\theta\right)\\ P\left(y^{(i)}=2\mid x^{(i)};\theta\right)\\ \cdots\\ P\left(y^{(i)}=k\mid x^{(i)};\theta\right)\end{pmatrix}=\frac{1}{\sum_{j=1}^{k}e^{\theta_j^T x^{(i)}}}\begin{bmatrix}e^{\theta_1^T x^{(i)}}\\ e^{\theta_2^T x^{(i)}}\\ \cdots\\ e^{\theta_k^T x^{(i)}}\end{bmatrix}$$

我们依然按照上面的思路来推导如何求解最优化的参数：

其中，$p(y^{(i)}=j|x^{(i)};\theta)=\dfrac{exp(\theta_i^T x)}{\sum_{k=1}^{K}exp(\theta_k^T x)}$则，

$$J(\theta)=-\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{k}1\{y^{(i)}=j\}\cdot(\theta_j^T x^{(i)}-log(\sum_{l=1}^{k}e^{\theta_l^T\cdot x^{(i)}}))\right]$$

一般使用梯度下降优化算法来最小化代价函数，而其中会涉及到偏导数，即$\theta_j:=\theta_j-\alpha\delta_{\theta_j}J(\theta)$，则$J(\theta)$对$\theta_j$求偏导，得到，

$$\frac{\nabla J(\theta)}{\nabla\theta_j}=-\frac{1}{m}\sum_{i=1}^{m}\left[\frac{\nabla\sum_{j=1}^{k}1\{y^{(i)}=j\}\theta_j^T x^{(i)}}{\nabla\theta_j}-\frac{\nabla\sum_{j=1}^{k}1\{y^{(i)}=j\}log(\sum_{l=1}^{k}e^{\theta_l^T\cdot x^{(i)}}))}{\nabla\theta_j}\right]$$

$$=-\frac{1}{m}\sum_{i=1}^{m}\left[1\{y^{(i)}=j\}x^{(i)}-\frac{\nabla\sum_{j=1}^{k}1\{y^{(i)}=j\}\sum_{l=1}^{k}e^{\theta_l^T\cdot x^{(i)}}}{\sum_{l=1}^{k}e^{\theta_l^T\cdot x^{(i)}}\nabla\theta_j}\right]$$

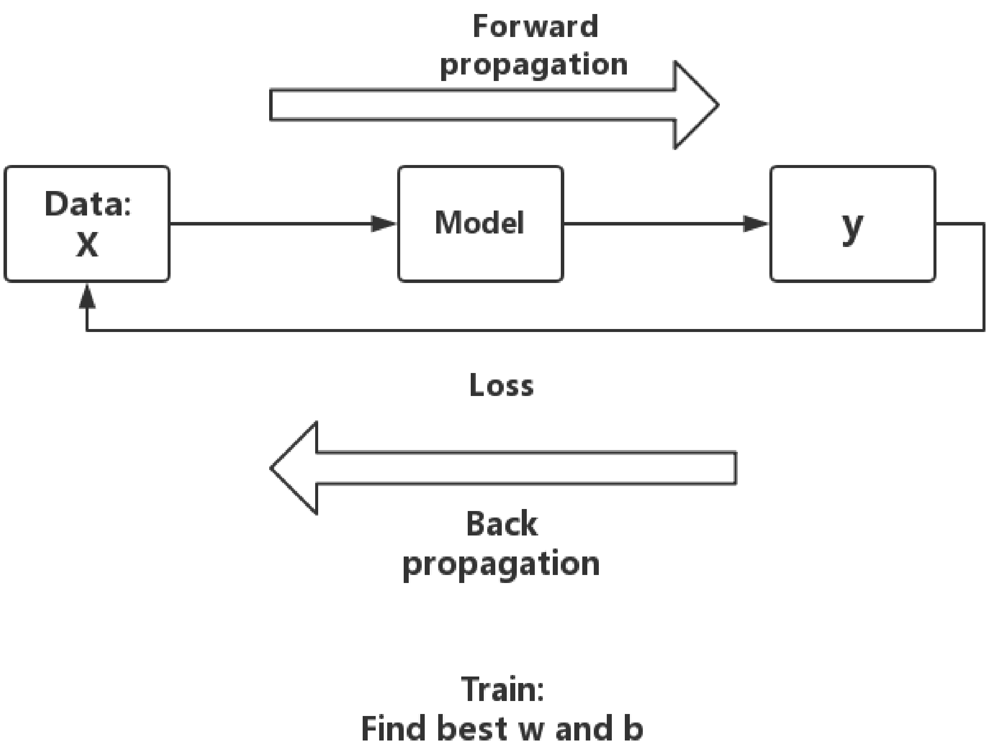
$$=-\frac{1}{m}\sum_{i=1}^{m}\left[1\{y^{(i)}=j\}x^{(i)}-\frac{x^{(i)}e^{\theta_j^T\cdot x^{(i)}}}{\sum_{l=1}^{k}e^{\theta_l^T\cdot x^{(i)}}}\right]$$

$$=-\frac{1}{m}\sum_{i=1}^{m}x^{(i)}\left[1\{y^{(i)}=j\}-p(y^{(i)}=j|x^{(i)};\theta)\right]$$

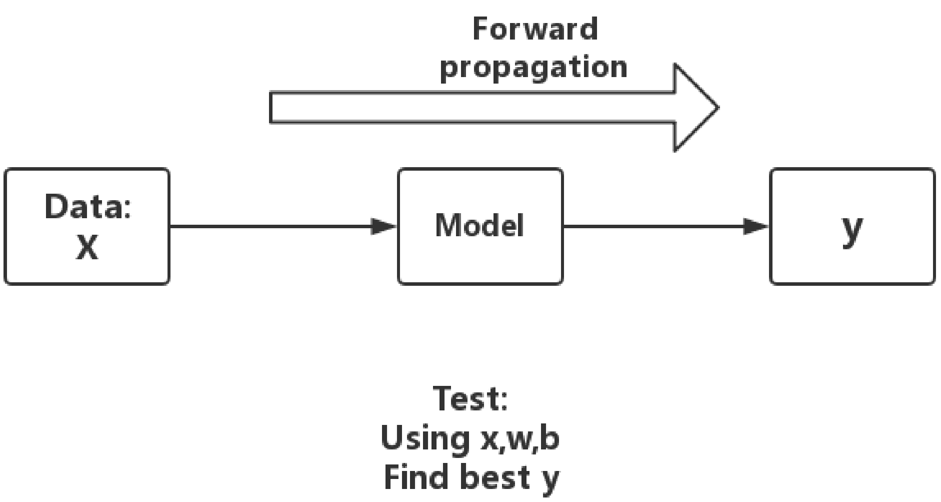
得到代价函数对参数权重的梯度就可以优化了。

How?

Train process



Predict process



Result

1. 二分类结果

```
In [9]: y_test_pred = classifier.predict(X_test_feats)
        y_test_pred=np.around(y_test_pred)
        print "The accuracy socre is ", np.mean(y_test == y_test_pred)

The accuracy socre is  0.976928571429
```

2. one vs all结果

```
In [14]: # you may change your code in function `predict`
         print X_test_feats.shape
         y_test_pred = classifier.pre2(classifier,X_test_feats)
         print y_test_pred.shape
         y_test_pred=np.around(y_test_pred)
         print y_test_pred.shape
         print "The accruacy socre is ", np.mean(y_test == y_test_pred)

(14000L, 785L)
[3 1 0 ..., 7 6 0]
(14000L,)
(14000L,)
The accruacy socre is  0.703857142857
```

Talk

1.那么多平滑的0-1之间的函数，就只选择了sigmod?
2.在多分类的时候，什么时候选择one vs all，什么时候选择softmax？
3.二分类中，最后的求得的两个概率用0.5的阈值区别是最好的吗？

Reference

1.周志华 机器学习
2.李航 统计学习方法
3.http://ufldl.stanford.edu/wiki/index.php/Softmax%E5%9B%9E%E5%BD%92
4.https://en.wikipedia.org/wiki/Sigmoid_function