神经网络&深度学习 TENSORFLOW Google

11 分类问题

西安科技大学 牟琦
muqi@xust.edu.cn

神经网络&深度学习 TENSORFLOW Google

11.1 逻辑回归

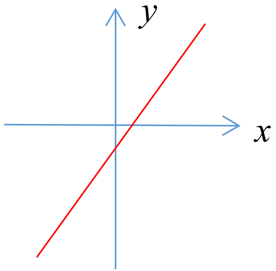
中国大学MOOC

线性回归

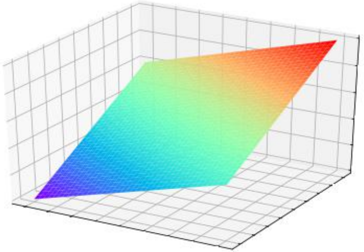
将自变量和因变量之间的关系，用**线性模型**来表示

根据已知的样本数据，对**未来**的、或者**未知**的数据进行**估计**

$$y = wx + b$$



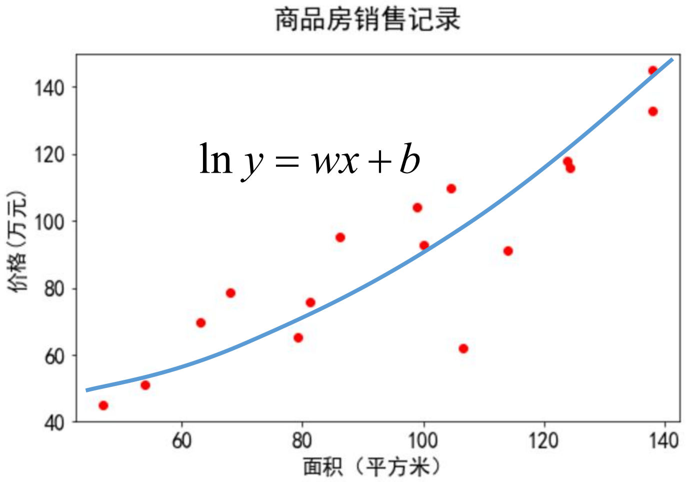
$$y = w_1x_1 + w_2x_2 + b$$



$$y = w_1x_1 + \ldots + w_mx_m + b$$

超平面 (Hyperplane)

■ 广义线性回归



对数线性回归 (log-linear regression)

$$\ln y = wx + b \qquad y = e^{wx+b}$$

$$Y = wx + b$$

$$g(y) = wx + b \qquad y = h(wx + b)$$

$$y = g^{-1}(wx + b)$$

广义线性模型 (generalized linear model)

$g(\cdot)$: 联系函数 (link function)

任何单调可微函数

高维模型 $Y = g^{-1}(W^T X)$

$$W = (w_0, w_1, \ldots, w_m)^T$$

$$X = (x^0, x^1, \ldots, x^m)^T$$

$$x^0 = 1$$

分类问题：垃圾邮件识别、图片分类、疾病判断

分类器：能够自动对输入的数据进行分类

输入：**特征**，输出：**离散**值

花萼宽度
花萼长度
花瓣宽度
花瓣长度

鸢尾花分类

0：山鸢尾
1：杂色鸢尾
2：维吉尼亚鸢尾

手写数字图片
28×28=784

手写数字识别

0
1
2
…
9

实现分类器

准备训练样本
训练分类器
对新样本分类

面 积
房间数
……
→ 商品房分类 →
0：普通住宅
1：高档住宅

$$z = wx + b$$

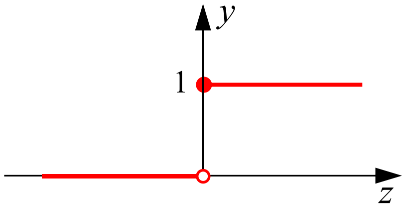
$$y = step(z) = \begin{cases} 0, & z - 1000000 < 0 \\ 1, & z - 1000000 \geq 0 \end{cases}$$

二分类问题： **1 / 0——正例**和**反例**

线性回归 $z=wx+b$ $\xrightarrow{step(z)}$ 分类器 0 / 1

单位阶跃函数 (unit-step function)

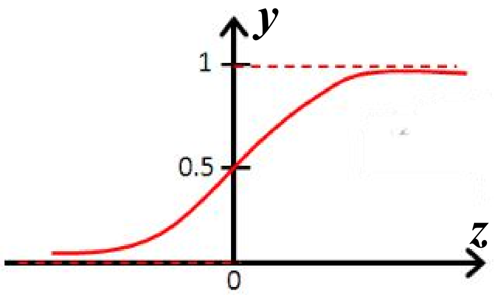
$$y = \begin{cases} 0, & z < 0\,; \\ 1, & z \geq 0\,; \end{cases}$$



- 不光滑
- 不连续

对数几率函数（logistic function）

$$y=\frac{1}{1+e^{-z}} \Rightarrow \ln\frac{y}{1-y}=z$$



- 单调上升，连续，光滑
- 任意阶可导

对数几率回归/逻辑回归（logistic regression）

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

Sigmoid函数

$$y=g^{-1}(z)=\underline{\sigma(z)}=\sigma(wx+b)$$

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

多元模型

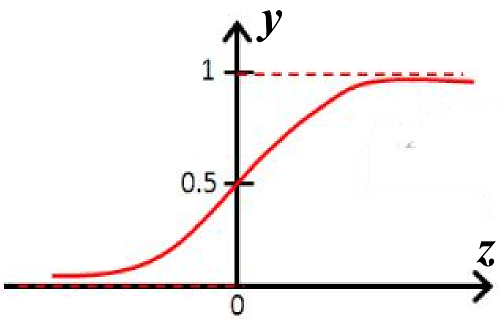
$$y=\frac{1}{1+e^{-(W^TX)}}$$

$$W=(w_0,w_1,\ldots,w_m)^T$$
$$X=(x^0,x^1,\ldots,x^m)^T$$
$$x^0=1$$

逻辑回归

$$y = g^{-1}(z) = \sigma(z) = \sigma(wx+b)$$

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



平方损失函数

$$Loss = \frac{1}{2}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 = \frac{1}{2}\sum_{i=1}^{n}(y_i-\sigma(wx_i+b))^2 = \underline{\sum_{i=1}^{n}\left(y_i-\frac{1}{1+e^{-(wx_i+b)}}\right)^2}$$

非凸函数

$$\boxed{\begin{aligned} w^{(k+1)} &= w^{(k)} - \eta\boxed{\frac{\partial Loss}{\partial w}} \\ b^{(k+1)} &= b^{(k)} - \eta\boxed{\frac{\partial Loss}{\partial b}} \end{aligned}}$$

$$\boxed{\frac{\partial Loss}{\partial w}} = \sum_{i=1}^{n}(y_i-\sigma(wx_i+b))(\overset{\to 0}{\boxed{-\sigma'(wx_i+b)}})x_i$$

$$\boxed{\frac{\partial Loss}{\partial b}} = \sum_{i=1}^{n}(y_i-\sigma(wx_i+b))(\overset{\to 0}{\boxed{-\sigma'(wx_i+b)}})$$

交叉熵损失函数

$$Loss = -\sum_{i=1}^{n}[y_i \ln \hat{y}_i + (1-y_i)\ln(1-\hat{y}_i)]$$

y_i	第i个样本的标记
$\hat{y}_i$	$\hat{y}_i = \sigma(wx_i + b)$

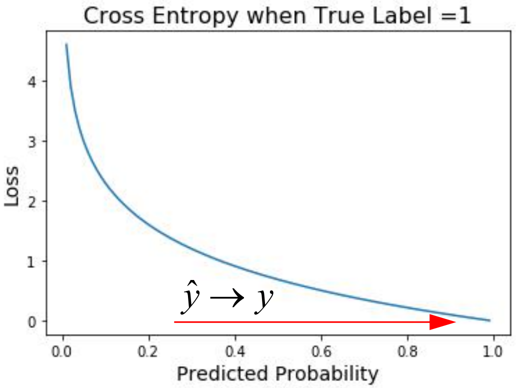
$$\times \; Loss = -\sum_{i=1}^{n}[\hat{y}_i \ln \underline{y_i} + (1-\hat{y}_i)\ln(\underline{1-y_i})]$$

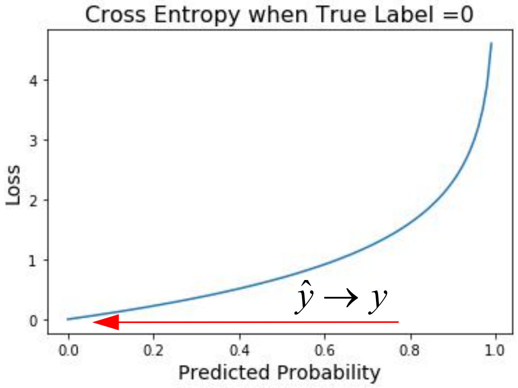
平均交叉熵损失函数

$$Loss = -\frac{1}{\boxed{n}}\sum_{i=1}^{n}[y_i \ln \hat{y}_i + (1-y_i)\ln(1-\hat{y}_i)]$$

交叉熵损失函数 :概率分布之间的误差

$$Loss = -\sum_{i=1}^{n}[y_i \ln \hat{y}_i + (1-y_i)\ln(1-\hat{y}_i)]$$





- 无需对σ函数求导

$$\frac{\partial Loss}{\partial w} = \frac{1}{n}\sum_{i=1}^{n} x_i(\hat{y}_i - y_i)$$

$$\frac{\partial Loss}{\partial b} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y)$$

- 凸函数

样本	标记	预测值	结果判断
样本1	0	0.1	正确
样本2	0	0.2	正确
样本3	1	0.8	正确
样本4	1	0.99	正确

准确率 (accuracy): $\frac{\text{正确分类的样本数}}{\text{总样本数}}$ =100%

交叉熵损失：$Loss = -\frac{1}{n}\sum_{i=1}^{n}[y_i \ln \hat{y}_i + (1-y_i)\ln(1-\hat{y}_i)]$

样本1：-(0×ln0.1 + 1×ln0.9)= -ln0.9=0.1053...

样本2：-(0×ln0.2 + 1×ln0.8)= -ln0.8=0.2231...

样本3：-(1×ln0.8 + 0×ln0.2)= -ln0.8=0.2231...

样本4：-(1×ln0.99 + 0×ln0.2)= -ln0.99=0.0100...

交叉熵损失：0.5616...　　平均损失：0.1404...

模型A

样本	标记	预测值	结果判断
样本1	0	0.1	正确
样本2	0	0.2	正确
样本3	1	0.8	正确
样本4	1	0.49	错误

准确率：75%

交叉熵损失：

-(0×ln0.1+1×ln0.9)=-ln0.9=0.1053...

-(0×ln0.2+1×ln0.8)=-ln0.8=0.2231...

-(1×ln0.8+0×ln0.2)=-ln0.8=0.2231...

-(1×ln0.49+0×ln0.51)=-ln0.49=0.7133...

平均损失：**0.3162...**

模型B

样本	标记	预测值	结果判断
样本1	0	0.49	正确
样本2	0	0.45	正确
样本3	1	0.51	正确
样本4	1	0.1	错误

准确率：75%

交叉熵损失：

-(0×ln0.49+1×ln0.51)=-ln0.51=0.6733...

-(0×ln0.45+1×ln0.55)=-ln0.55=0.5978...

-(1×ln0.51+0×ln0.49)=-ln0.51=0.6733...

-(1×ln0.1+0×ln0.9)=-ln0.1=2.3025...

平均损失：**1.0617...**