线性模型

基本形式

实际上就是找到一种对于各个属性的线性组合，使得其最接近真实的 lable。

整体的形式是$f(x) = w^T x + b$. w,b 两者学习到之后，模型就被确定。通过 w，我们可以看出每个元素的重要性，从而有一定的可解释性。

线性回归

对于各种属性，怎么将其转换为数字呢？

1. 数字元素 直接使用就可以
2. 有序的属性 如高、中、低，可以分别用 1,0.5,0 来表示
3. 无序的属性 如红、黄、蓝，进行 one-hot 编码

如何确定两个参数呢？这取决于如何定义最接近。最常见的做法是用最小二乘法，即使得预测结果和目标的欧式距离最短。即$(w^*, b^*) = \arg\min_{\{a,b\}} \left(w^T x + b - y\right)^2$

对于更加一般的数据集 D,我们要试图学到$f(x_i) = w^T x_i + b$，使得均方误差最小。为了便于处理，我们让$\hat{w} = (w; b)$,将数据集表示为一个$(m+1) \times b$的矩阵X. 对于每一个数据x_i,我们让X的第 i 行为$(x_i^T 1)$,于是有$\hat{w} = \arg\min_{\{w\}} (y - Xw)^{T(y-Xw)}$,对$w$求偏导，得$\frac{\partial(E)}{\partial(w)} = 2X^{T(Xw-y)}$

如果$2X^T X$是正定的/满秩的，那么便有一个 close form 的解。

实际上大概率是非满秩的。所以我们常引入一个正则项(regularization)。

如果要拟合的函数不是一个线性的怎么办呢？假设是f性质的函数，那么令$f^-1(y) = w^t x + b$，先处理 y，然后拟合就可以了。

对数几率回归

对于一个分类问题，我们更需要知道是概率如何，应当是一个$[0,1]$的数。为了将线性回归出来的数映射到$[0,1]$上，我们有很多种方法。

最为显然的一个方法是，建立分段函数

$$f(x) = \begin{cases} 1 x > 0 \\ 0.5 x = 0 \\ 0 x < 0 \end{cases}$$

显然可以描述。

但显然这个玩意不可逆，于是改变方法，用 Sigmoid 函数($\frac{1}{1+e^-x}$)来映射。

这玩意 loss 推导那块很无聊，反正最后得出来

$$l(\beta) = \sum_{i=1}^{m} (-y_i \beta^T x_i + \log\left(1 + e^{\beta^T x_i}\right)$$

是个凸函数，可以用凸优化的常规方法求出最小值。

线性判别分析

整体的思路就是将一堆点映射到某个直线上，让类内点距离最近的同时类间点距离最远。