第三章 线性回归

2020年3月10日　　17:35

0、回归跟分类的区别主要在于预测的目标函数是连续指/离散值

1、线性回归

给定由m个属性描述的样本 x=(x1;x2;...xm),其中xi 是 x 在第i个属性上的取值。线性回试图学得一个通过属性值的线性组合来预测的函数:

$$f(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + \ldots + w_m x_m + b$$

改用向量形式写为:

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$$

其中:

$$\boldsymbol{w} = (w_1; w_2; \ldots; w_m)。$$

属性多个时称”多元线性回归“。

模型主要做的是确定w和b的值，使用的是最小二乘法：基于真实值和预测值的均方差最小原则确定w和b

$$\begin{aligned}(w^*, b^*) &= \underset{(w,b)}{\arg\min} \sum_{i=1}^{n} (f(x_i) - y_i)^2 \\ &= \underset{(w,b)}{\arg\min} \sum_{i=1}^{n} (y_i - w x_i - b)^2\end{aligned}$$

其中二者的闭式(closed-form)解为:

$$w = \frac{\sum_{i=1}^{n} y_i (x_i - \bar{x})}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2}$$

$$b = \frac{1}{n} \sum_{i=1}^{n} (y_i - w x_i)$$

其中:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

2、广义线性回归

现实问题大多不满足线性回归所假设的输入空间到输出空间的线性映射关系。因此可以将线性回归的预测值做一个非线性的变换去逼近真实值。这叫做广义线性回归，例如“对数线性回归”试图让输出标记的对数作为线性模型逼近的目标。

$$\ln y = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b\ .$$

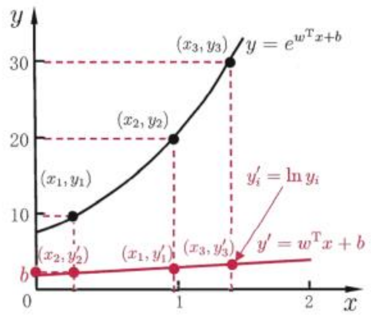


图 3.1 对数线性回归示意图

广义线性回归具体形式：

$$y = g(w^T x + b)$$

函数g称为联系函数 link function，理论上可以是任意函数，例如上图g为指数函数。

3、对数几率回归/逻辑斯蒂回归

线性模型不光能处理回归学习，也可以处理分类任务。对于二分类任务输出标记为 $y \in [0,1]$ ，但线性模型产生的预测值z是实值。考虑g为单位阶跃函数：

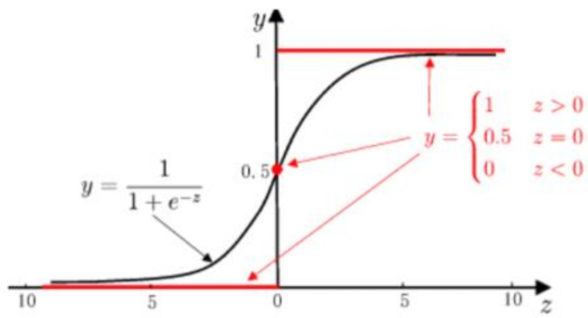
$$y = \begin{cases} 0, & z < 0; \\ 0.5, & z = 0; \\ 1, & z > 0, \end{cases}$$

如预测值大于零就判为正例，
小于零就判为反例，
预测值为临界值零则可任意判别

但是单位阶跃函数不连续，不可直接作为联系函数g(·)。于是考虑逻辑斯蒂函数作为其代替函数。

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

$$y = \frac{1}{1 + e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}}$$

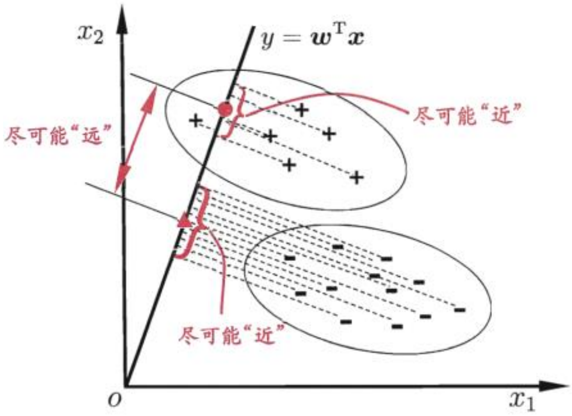


单位阶跃函数与对数几率函数的比较

逻辑斯蒂函数是任意阶可导的凸函数，可直接应用现有的数值优化算法求取最优解
逻辑斯蒂回归只能求解连续属性值问题，不能求解离散属性值问题。遇到离散属性值：①若属性值之间存在“序”关系，通过连续化将其转化为连续值②：若属性值之间不存在“序”关系：通常可将K个属性值转换为K维向量

4、线性判别分析

LDA是一种线性学习方法即：给定训练样例集，设法将样例投影到一条直线上，使得同类样例的投影点尽可能接近、异类样例的投影点尽可能远离；在对新样本分类时，将其投影到这条直线上，根据投影点位置确定其类别。三维图如下：



+ - 表示正例反例

欲使同类样例的投影点尽可能接近, 可以让同类样例投影点的协方差尽可能小, 即 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}$ 尽可能小; 而欲使异类样例的投影点尽可能远离, 可以让类中心之间的距离尽可能大, 即 $\|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1\|_2^2$ 尽可能大. 同时考虑二者, 则可得到欲最大化的目标

$$\begin{aligned} J &= \frac{\|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1\|_2^2}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}} \\ &= \frac{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1)\boldsymbol{w}} . \end{aligned} \tag{3.32}$$

————课本 P61

这就是LDA欲最大化的目标，常见的优化是：

$$\max_{\mathbf{W}} \frac{\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}\right)}{\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_w\mathbf{W}\right)} ,$$

其中W视为一个投影矩阵，则多分类LDA将样本投影到N-1维空间，减少样本点的维度。因此LDA也是一种经典的监督降维技术

5、多分类学习

多分类学习的基本思路是：拆解法，即将多分类任务拆分为若干个二分类任务求解。经典拆分策略有：一对一、一对多、多对多

拆分后为每个二分类任务训练一个分类器，在测试时对这些分类器的预测结果进行集成以获得最终的多分类结果。

OvO：N个类别两两配对，产生 N(N-1)/2 个二分类任务。

OvR：将每一个类的样例作为正例、所有其他类的样例作为反例来训练N个分类器。

MvM：每次将若干个类作为正类，若干个其他类作为反类。

常用MVM技术，纠错输出码

编码过程分为两步：

- 编码: 对 N 个类别做 M 次划分, 每次划分将一部分类别划为正类, 一部分划为反类, 从而形成一个二分类训练集; 这样一共产生 M 个训练集, 可训练出 M 个分类器.
- 解码: M 个分类器分别对测试样本进行预测, 这些预测标记组成一个编码. 将这个预测编码与每个类别各自的编码进行比较, 返回其中距离最小的类别作为最终预测结果.

对于同一个学习任务，纠错输出码越长纠错能力越大，所需分类器越多，计算和存储开

销都会更大。

6、类别不平衡问题 课本P66

类别不平衡（class-imbalance)指分类任务种不同类别的训练样例数目差别很大的情况。

需要用到基本策略---再缩放（rescaling）。原因是因为训练集：“真实样本的无偏采样”这个假设经常很难成立。所用到的基本方法：（一下均假定正类样例远少于反例）

①欠采样（undersampling)：去除一些反例来使得正反例数目接近然后再进行学习。

②过采样（oversampling):增加一些正例使得正反例数目接近

③阈值移动（threshold-moving)：基于原始训练集进行学习，但在用训练好的分类器进行预测时将下式嵌入到决策过程中。

$$\frac{y'}{1-y'} = \frac{y}{1-y} \times \frac{m^-}{m^+} . \tag{3.48}$$