1 分类的线性方法

1.1 线性拟合与感知器

既然分类的输出是离散值, 它当然也属于连续值. 于是, 我们可以简单地沿用线性回归方法, 即

$$y = g(\boldsymbol{x};\boldsymbol{w}) = w_0x_0 + \cdots + w_Mx_M = \boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}$$

回归模型输出的是连续值, 因此需要离散化. 例如, 当输出值为 $\{0,1\}$ 时, 可以使用

$$g(\boldsymbol{x}) = \begin{cases} 1, & f(\boldsymbol{x};\boldsymbol{w}) \geq \dfrac{1}{2} \\ 0, & f(\boldsymbol{x};\boldsymbol{w}) < \dfrac{1}{2} \end{cases}$$

将结果离散化. 然而, 如果出现远离数据聚集处的点, 那么线性回归可能会因为迎合这些点而产生较大偏差. 因此, 我们可以直接用阶跃函数进行拟合, 即

$$y = g(\boldsymbol{x};\boldsymbol{w}) = f\left(\boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}\right) = H\left(\boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}\right), \quad \text{where } H(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

其中 $f(z)$ 为激活函数, 对输入的 z 进行非线性变换得到结果. 这里采用的激活函数为阶跃函数 $H(z)$. 直接用上述定义的 $g(\boldsymbol{x};\boldsymbol{w})$ 进行拟合 (例如进行最大似然法估计等), 这就是**感知器 (perceptron)** 模型.

定义 1.1 感知器 使用阶跃函数 $H(z)$ 作为激活函数的线性分类模型, 即

$$y = g(\boldsymbol{x};\boldsymbol{w}) = H\left(\boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}\right)$$

称为**感知器 (perceptron)** 模型.

1.2 逻辑回归

1.2.1 逻辑回归的模型

感知器模型的激活函数是阶跃函数, 它在 $x=0$ 处不可导, 因此无法使用梯度下降法等涉及导数的方法进行优化. 为了替换成光滑的函数以便优化, 我们可以用 sigmoid 函数 $\sigma(z)$ 替换前面的阶跃函数:

$$y = g(\boldsymbol{x};\boldsymbol{w}) = \sigma\left(\boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}\right), \quad \text{where } \sigma(z) = \frac{1}{1+e^{-z}}$$

这就是**逻辑回归 (logistic regression)** 模型.

定义 1.2 逻辑回归 使用 sigmoid 函数 $\sigma(z)$ 作为激活函数的线性分类模型, 即

$$y = g(\boldsymbol{x};\boldsymbol{w}) = \sigma\left(\boldsymbol{w}^{\mathrm{t}}\boldsymbol{x}\right), \quad \text{where } \sigma(z) = \frac{1}{1+e^{-z}}$$

称为**逻辑回归 (logistic regression)** 模型.

上述函数返回一个 $(0, 1)$ 间的连续值. 因此, 我们需要将结果解读为样本属于某一类别的概率.

当 $\sigma\left(\boldsymbol{w}^{\mathrm{t}} \boldsymbol{x}\right) > 0.5$ 时, 我们预测样本属于类别 1 的概率更大; 否则预测样本属于类别 0 的概率更大. 所预测的两种类别的边界称作**决策面 (Decision Boundary)**, 由下式描述:

$$\boldsymbol{w}^{\mathrm{t}} \boldsymbol{x} = 0$$

这是 $\boldsymbol{x}$ 的空间中的一个线性平面.

1.2.2 逻辑回归的求解

与前面一样, 逻辑回归也可以归结为一个概率优化问题, 使用极大似然法即可. 仍然假定数据集为 $\mathcal{D} = \{\boldsymbol{x}_n, t_n\}_{n=1}^{N}$, 我们的目标是最大化概率

$$\max_{\boldsymbol{w}} P\left(\right)$$