第 8 章 朴素贝叶斯

8.1 朴素贝叶斯

假设训练数据为$\left\{\mathbf{x}_i, y_i\right\}_{i=1}^{n}$，而$y_i$的可能取值为$y_i \in \left\{1, 2, \cdots, K\right\}$，共分为$K$类。

“贝叶斯最优决策”(Bayes optimal decision)通过最大化“后验概率”(posterior probability)来作预测：

$$\max_{k} \; P(y_i = k \mid \mathbf{x} = \mathbf{x}_i) \qquad (8.1)$$

进一步，使用贝叶斯公式计算后验概率：

$$\begin{aligned} P(y_i = k \mid \mathbf{x} = \mathbf{x}_i) &= \frac{P(\mathbf{x} = \mathbf{x}_i \mid y_i = k) P(y_i = k)}{P(\mathbf{x} = \mathbf{x}_i)} \\ &\equiv \frac{f_k(\mathbf{x}_i)\pi_k}{P(\mathbf{x} = \mathbf{x}_i)} \end{aligned} \tag{8.2}$$

其中， $\pi_k \equiv P(y_i = k)$ 为“先验概率”(prior probability)，而 $f_k(\mathbf{x}_i) \equiv P(\mathbf{x} = \mathbf{x}_i \mid y_i = k)$为给定类别“$y_i = k$”的条件概率密度 (class-conditional density of $\mathbf{x}_i$ in class k)。

但(联合)条件概率 $f_k(\mathbf{x}_i) = P(\mathbf{x} = \mathbf{x}_i \mid y_i = k)$可能很难估计。

特征向量$\mathbf{x}_i$可以是连续、离散或混合型随机向量(同时包括连续型与离散型变量)，故有时无法假设$\mathbf{x}_i$服从多维正态分布。

$\mathbf{x}_i$的维度也可能很高。例如，在使用“词频”(word frequency)判定“正常邮件”(email)与“垃圾邮件”(spam)时，涉及的关键词可能成千上万，而所得数据矩阵一般为高维的稀疏矩阵(sparse matrix)，故不易估计其协方差矩阵。

这也是“维度灾难”(curse of dimensionality)的一种表现形式。

为简化计算，**朴素贝叶斯分类器**(Naive Bayes Classifier)对高维的条件概率 $f_k(\mathbf{x}_i)$ 作了一个“天真”(naive)的假定。

假设在给定类别“$y_i = k$”的情况下，$\mathbf{x}_i$ 的“各分量属性之间条件独立”(attribute conditional independence assumption)。

在此假定下，条件概率 $f_k(\mathbf{x}_i)$ 可写为

$$
\begin{aligned}
f_k(\mathbf{x}_i) &\equiv P(\mathbf{x}=\mathbf{x}_i \mid y_i = k) \\
&= P(x_{i1} \mid y_i = k)\cdot P(x_{i2} \mid y_i = k)\cdots P(x_{ip} \mid y_i = k) \qquad (8.3) \\
&= \prod_{j=1}^{p} P(x_{ij} \mid y_i = k)
\end{aligned}
$$

其中，p为$\mathbf{x}_i = (x_{i1}\cdots x_{ip})'$的维度，即特征变量的个数。

根据朴素贝叶斯的假定，事实上将高维问题降为一维问题，因为只要分别估计p个单变量的条件概率$P(x_{ij} \mid y_i = k)$，其中$j = 1,\cdots,p$，然后连乘在一起即可。

尽管朴素贝叶斯的假定不切实际，但在不少情况下却能得到较好的预测效果。

可能由于在有些情况下，属性之间的部分相关性可能互相抵消。

而且，我们关心的是对于类别的预测，并非准确地估计条件概率。

8.2 拉普拉斯修正

在应用朴素贝叶斯分类器时，有时还需进行**拉普拉斯修正**(Laplacian correction)。

这是因为，朴素贝叶斯的“概率连乘”形式使其具有“一票否决”的特点。

比如，某个虚拟变量x_{ij}在训练数据的第k类中共有n_k次取值为 0，而取值为 1 的次数为 0。

此时，在给定“$y_i = k$”情况下，对于事件“$x_{ij} = 1$”后验概率之样本估计为

$$\hat{P}(x_{ij} = 1 \mid y_i = k) = \frac{0}{n_k} = 0 \qquad (8.4)$$

将上式代入方程(8.3)，会使得整个后验概率的估计值变为 0，即

$$\hat{P}(x_{i1}, \cdots, x_{ij} = 1, \cdots, x_{ip} \mid y_i = k) = 0 \qquad (8.5)$$

一旦在未来的观测数据中出现$x_{ij} = 1$(无论其他属性取值如何)，则会自动排除“$y_i = k$”的可能性(这显然不合理)，从而导致偏差。

拉普拉斯修正的解决方案为，将x_{ij}的不同取值在第k类数据中的出现次数均加上 1，从而共有(n_k+1)次取值为 0，而有 1 次取值为 1。

修正之后，在给定“$y_i=k$”情况下，事件“$x_{ij}=1$”的后验概率之估计值为

$$\hat{P}(x_{ij}=1\,|\,y_i=k)=\frac{1}{n_k+2} \tag{8.6}$$

而事件“$x_{ij}=0$”的后验概率之估计值为

$$\hat{P}(x_{ij}=0\,|\,y_i=k)=\frac{n_k+1}{n_k+2} \tag{8.7}$$

更一般地，在做拉普拉斯修正时，也可将x_{ij}的不同取值在第k类数据中的出现次数均加上一个很小的正数$c>0$，而不一定限制$c=1$。

由于拉普拉斯修正将等于 0 的后验概率修正为正数，起着平滑的作用，故也称为**拉普拉斯平滑**(Laplace smoothing)，而c为拉普拉斯平滑参数。

8.3 朴素贝叶斯的 Python 案例

使用 Hastie et al. (2009)的 `spam` 数据，演示使用朴素贝叶斯过滤垃圾邮件。

响应变量 `spam` 取值为 `spam` (垃圾邮件)或 `email`(正常邮件)。

特征变量`A.1`至`A.54`分别表示54个不同的词汇(word)或字符(character)在邮件中的出现频率(以百分数计，取值范围为$[0, 100]$)。

变量 A.55-A.57 分别表示连续大写字母序列的平均长度(average length of uninterrupted sequences of capital letters)、连续大写字母序列的最大长度(length of longest uninterrupted sequence of capital letters)与邮件中大写字母总数(total number of capital letters in the e-mail)。

* 详见教材，以及配套 Python 程序（现场演示）。