第 5 章

贝叶斯分类

本章重点

- 最大后验概率的意义。
- 生成方法与判别方法的定义。
- 高斯判别分析的原理。
- 高斯判别分析与线性判别分析的关系。
- 朴素贝叶斯的原理及应用。

微课视频

感知机是从训练数据中学习一个用于分类的超平面，对于新的数据(样本)，可用得到的超平面判断其类别。而 logistic 回归则是将分类当成一个条件概率 $p(y|\boldsymbol{x})$ 问题，通过学习得到相应的概率分布。logistic 回归直接给定了条件概率的分布，这是一种比较简单的情形，更一般的概率分类方法是通过贝叶斯理论得到，即在给定样本 $\boldsymbol{x}$ 时，使得概率 $p(y|\boldsymbol{x})$ 最大，这也称为贝叶斯分类方法(或贝叶斯分类器)，这种方法与贝叶斯决策理论有关。贝叶斯分类器可以给出分类结果的概率。在一些机器学习应用中，给出结果出现的概率会更有意义，如在诊断疾病时，给出病人可能患某种疾病的概率比直接给出他是否患病更有意义。在介绍贝叶斯分类器前，先介绍贝叶斯决策理论。

贝叶斯决策理论(Bayes decision theory)是关于信息不完全的情况下应该如何进行决策的理论，是统计学习中的一个基本方法。它与著名数学家 Thomas Bayes 提出的贝叶斯理论相关。贝叶斯理论是为了解决逆概率问题而提出的，即如何通过发生的事件反推造成该事件发生的原因。该理论在 Thomas Bayes 生前并没有受重视，而是在他去世以后，他的好友在重新翻阅他生前的论文时发现了他提出的这一理论。在贝叶斯决策理论中，决策者会根据历史的数据学习其中的规律，掌握其变化的可能状况及各种状况的分布情况，对部分未知信息进行概率或者期望估计，并根据估计的概率或期望做出最优决策。对于分类问题，贝叶斯决策理论要考虑如何基于已有的概率和误分类损失得到最优的类标记。

贝叶斯分类是贝叶斯学习的一种具体应用。贝叶斯学习是指利用贝叶斯决策理论对未知知识进行学习的过程，这是一种基于概率的学习方法，该方法采用概率表示所有形式的不确定性，通过概率规则实现学习和推理。

不同的贝叶斯分类方法的差异通常在于概率估计方法和风险估计方法的差异。例如，朴素贝叶斯分类和贝叶斯网的主要差别在于对条件概率的估计方法不同。基于贝叶斯决策理论的贝叶斯分类方法是机器学习和模式识别中的一个基本方法，尤其是朴素贝叶斯分类方法，能够有效地对海量数据进行分析建模，构造相应的分类器，能对未知数据进行分类识别。

下面举例说明贝叶斯决策理论在分类中的应用。

假设由 n 个样本构成训练数据集为 $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n]$，样本的类标记向量为 $\boldsymbol{y}=[y_1,y_2,\cdots,y_n]$，另外假设这些样本有 k 个类，即 $\boldsymbol{c}=[c_1,c_2,\cdots,c_k]$。若用 0-1 损失函数(loss function)来度量分类决策函数 $f(\boldsymbol{X})$的分类误差，则有

$$L(y_i,f(\boldsymbol{x}_i))=\begin{cases}1, & y_i\neq f(\boldsymbol{x}_i)\\ 0, & y_i=f(\boldsymbol{x}_i)\end{cases}$$

定义分类决策函数 $f(\boldsymbol{X})$的期望风险函数：

$$R(f)=E[L(y,f(\boldsymbol{X}))]=\sum_{k=1}^{K}[L(c_k,f(\boldsymbol{X}))p(c_k\mid\boldsymbol{X})]$$

为了让期望风险最小化，只需要对每个样本的期望风险最小化，也就是说，对于任意一个样本 $\boldsymbol{x}$，则要最小化下面的目标函数：

$$\min\sum_{k=1}^{K}[L(c_k,y_i)p(c_k\mid\boldsymbol{x})] \tag{5.1}$$

由式(5.1)可以得到下面的结果

$$\max_{c_k}p(c_k\mid\boldsymbol{x}) \tag{5.2}$$

由式(5.2)可以看出：要让决策函数的期望风险最小，将 $p(c_k|\boldsymbol{x})$中最大的 c_k 作为类别。在某些情况下，$p(c_k|\boldsymbol{x})$也称为后验概率(posterior probability)。后验概率是统计学上的概念，它与先验分布有关。后验概率是指在给出相关证据或数据后所得到的条件概率，后验概率分布通常是未知的，需要通过数据才能推导出来。

从式(5.2)可知，要让分类决策函数的期望风险最小，首先要得到 $p(c_k|\boldsymbol{x})$的分布。通常有两种方法可以用来获得这种概率分布。

(1) 判别方法(discriminative approach)：其对应的模型为判别模型。

(2) 生成方法(generative approach)：其对应的模型为生成模型。

判别方法会直接用决策函数(如感知机等)或 $p(c_k|\boldsymbol{x})$的分布函数(如 logistic 回归等，也称为概率判别模型)来得到预测模型；而生成方法会通过贝叶斯定理来学习 $p(c_k|\boldsymbol{x})$，即

$$p(c_k\mid\boldsymbol{x})=\frac{p(\boldsymbol{x}\mid c_k)p(c_k)}{p(\boldsymbol{x})}$$

式中，$p(c_k)$称为先验概率(prior probability)，当训练集中包括足够多的独立同分布样本时，它的分布通常由训练样本中各个类别所占比得到；而 $p(\boldsymbol{x}|c_k)$称为后验概率；$p(\boldsymbol{x})$是归一化证据因子。在样本给定的情况下，证据因子与类标记无关。典型的生成模型有高斯判别分析、朴素贝叶斯和隐马尔可夫模型等。

判别方法和生成方法都是用来获取后验概率的，它们之间的区别：判别方法直接假定 $p(c_k|\boldsymbol{x})$的形式，然后通过一种学习策略(如极大似然估计)来学习该函数的参数；生成方法则不是这样，它通过计算 $p(\boldsymbol{x}|c_k)$，然后通过贝叶斯定理得到 $p(c_k|\boldsymbol{x})$。$p(\boldsymbol{x}|c_k)$表示样本 $\boldsymbol{x}$ 是由某个分布函数产成(生成)的，这就是该方法被称为生成模型的原因。生成方法是按不同的类别学习模型，例如，狗和猫是不同的类，它们的特征会有所不同，因此可针对狗和猫分别学习相应的模型，然后在测试时，看测试样本属于哪类的概率大，就将其归到哪类中。简单地说，判别方法和生成方法的区别：判别方法假定了 $p(c_k|\boldsymbol{x})$的分布；而生成方法则会通过计算 $p(\boldsymbol{x}|c_k)$和 $p(c_k)$，然后通过贝叶斯定理计算 $p(c_k|\boldsymbol{x})$。

本章将介绍两种经典的生成模型：高斯判别分析和朴素贝叶斯。

5.1 高斯判别分析

高斯判别分析(Gaussian discriminant analysis,GDA)是一种生成模型。对于一个二分类问题,假设训练数据集为 $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n]$,相应的类标记为 $\boldsymbol{y}=[y_1,y_2,\cdots,y_n]$,$y_i\in\{0,1\}$。假设多元高斯分布为

$$N(\boldsymbol{\mu},\boldsymbol{\Sigma})=\frac{1}{(2\pi)^{d/2}\mid\boldsymbol{\Sigma}\mid^{1/2}}\mathrm{e}^{\left\{\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)\right.}$$

式中,$\boldsymbol{\Sigma}$ 为协方差矩阵;$|\boldsymbol{\Sigma}|$为协方差矩阵的行列式;$\boldsymbol{\mu}$ 为均值向量。定义概率分布为

$$\begin{cases}p(\boldsymbol{x}\mid y=0)=N(\boldsymbol{\mu}_0,\boldsymbol{\Sigma})\\p(\boldsymbol{x}\mid y=1)=N(\boldsymbol{\mu}_1,\boldsymbol{\Sigma})\end{cases}\tag{5.3}$$

从式(5.3)可以看出,这两类数据是由两个多元高斯分布生成的,这两个多元高斯分布的协方差是一样,但均值向量不同。图 5.1 为具有相同协方差矩阵、不同均值的两个二元高斯分布的等高线示意图。

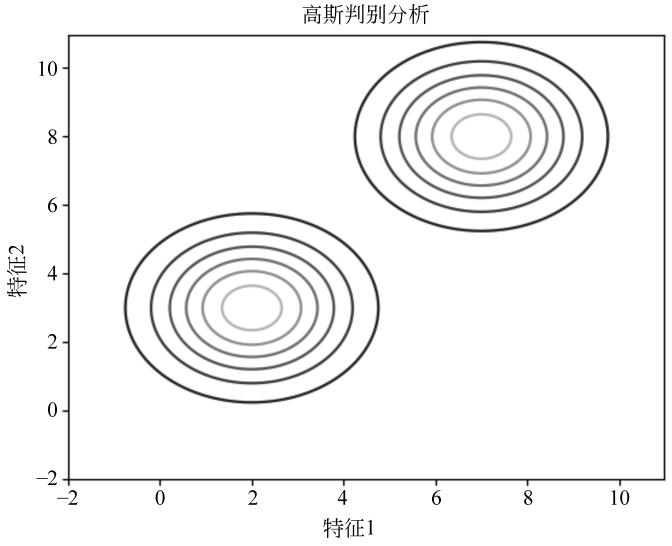

图 5.1 具有相同协方差矩阵、不同均值的两个二元高斯分布的等高线示意图

若假定类标记服从伯努利分布,即 $p(y)=\theta^y(1-\theta)^{1-y}$,其中 θ 为伯努利分布的参数。由贝叶斯定理可知,$p(y\mid\boldsymbol{x})=\dfrac{p(\boldsymbol{x},y)}{p(\boldsymbol{x})}$,而 $p(\boldsymbol{x})$通常为固定值,为了得到 $p(y\mid\boldsymbol{x})$,可通过最大后验概率估计得到高斯判别模型的参数,具体的目标函数为

$$\begin{aligned}\max L(\boldsymbol{\mu}_0,\boldsymbol{\mu}_1,\boldsymbol{\Sigma},\theta)&=\ln\prod_{i=1}^{m}p(\boldsymbol{x}_i,y_i,\theta;\boldsymbol{\mu}_0,\boldsymbol{\mu}_1,\boldsymbol{\Sigma})\\&=\ln\prod_{i=1}^{m}p(\boldsymbol{x}_i\mid y_i;\boldsymbol{\mu}_0,\boldsymbol{\mu}_1,\boldsymbol{\Sigma})p(y_i;\theta)\end{aligned}\tag{5.4}$$

对式(5.4)中的各个参数求偏导,并令其为 0,就可得到所估计的参数,即

$$\begin{cases}\hat{\theta}=\dfrac{1}{n}\sum\limits_{i=1}^{n}I(y_i=1)\\ \hat{\boldsymbol{\mu}}_0=\dfrac{\sum\limits_{i=1}^{n}I(y_i=1)\ \boldsymbol{x}_i}{\sum\limits_{i=1}^{n}I(y_i=0)}\\ \hat{\boldsymbol{\mu}}_1=\dfrac{\sum\limits_{i=1}^{n}I(y_i=1)\ \boldsymbol{x}_i}{\sum\limits_{i=1}^{n}I(y_i=1)}\\ \hat{\Sigma}=\dfrac{1}{n}\sum\limits_{i=1}^{n}(\boldsymbol{x}_i-\boldsymbol{\mu}_{y_i})\ (\boldsymbol{x}_i-\boldsymbol{\mu}_{y_i})^{\mathrm{T}}\end{cases}\tag{5.5}$$

式中，$I(y_i=1)$为当 $y_i=1$ 时，返回为 1，否则返回为 0。$\boldsymbol{\mu}_{y_i}$ 表示与类标记 y_i 一样的 $\boldsymbol{\mu}$，如当 $y_i=1$，则 $\boldsymbol{\mu}_{y_i}=\boldsymbol{\mu}_1$。

在得到高斯判别模型的所有参数后，可以通过下面的公式预测新样本 $\boldsymbol{x}$ 的类别，即

$$\begin{aligned}p(y_k\mid\boldsymbol{x})&=p(\boldsymbol{x}\mid y_k)p(y_k)=N(\boldsymbol{\mu}_k,\boldsymbol{\Sigma})\hat{\theta}\\&=\left(\frac{1}{(2\pi)^{d/2}\mid\boldsymbol{\Sigma}\mid^{1/2}}\mathrm{e}^{\left(-\frac{1}{2}(x-\mu)^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(x-\mu)\right)}\right)\hat{\theta}\end{aligned}\tag{5.6}$$

将式(5.5)的结果代入式(5.6)，然后取对数，并去掉常量，则有

$$\ln p(y_k\mid\boldsymbol{x})=\boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}^{-1}\hat{\boldsymbol{\mu}}_k-\frac{1}{2}\hat{\boldsymbol{\mu}}_k\hat{\boldsymbol{\Sigma}}^{-1}\hat{\boldsymbol{\mu}}_k-\frac{1}{2}\boldsymbol{x}\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}+\ln\hat{\theta}\tag{5.7}$$

可用式(5.7)计算样本 $\boldsymbol{x}$ 属于 y_k 的概率，将 $\boldsymbol{x}$ 归入概率最大的那一类。在式(5.7)中，$\hat{\boldsymbol{\Sigma}}^{-1}\hat{\boldsymbol{\mu}}_k$是一个向量，可以令其等于 $\boldsymbol{w}_k$；$-\frac{1}{2}\hat{\boldsymbol{\mu}}_k\hat{\boldsymbol{\Sigma}}^{-1}\hat{\boldsymbol{\mu}}_k-\frac{1}{2}\boldsymbol{x}\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}+\ln\hat{\theta}$为常量，也可以令其等于 b_k，则式(5.7)可以改写为

$$\ln p(y_k\mid\boldsymbol{x})=\boldsymbol{w}_k\boldsymbol{x}^{\mathrm{T}}+b_k\tag{5.8}$$

从式(5.8)可以看出，对样本 $\boldsymbol{x}$ 的分类问题，最终转换成通过一个超平面进行判断，因此该方法称为线性判别分析(linear discriminant analysis，LDA)。

这里讨论的是二分类问题，因此只需要一个超平面。如果有 K 个类，则需要 $K-1$ 个超平面。图 5.2 是用线性判别分析对数据分类，白色的斜线为分类超平面，这两类数据是由相同协方差、不同均值的高斯分布生成的，给出了一个用线性判别分析进行分类的示意图。

讨论上面的问题，假定两个类别的数据分别由两个多元高斯函数生成，这两个高斯函数的协方差矩阵一样，只是均值向量不一样。若它们的协方差矩阵不一样，均值向量也不一样，所得到的方法称为二次判别法(quadratic discriminant analysis，QDA)。

在 sklearn 的 discriminant_analysis 包中，提供了一个可以执行线性判别分析的类：LinearDiscriminantAnalysis。可用下面的方法来实例化该类：

```
lda=LinearDiscriminantAnalysis(solver="svd")
```

在实例化这个类后，要以调用 fit()方法训练模型，并用 predict()方法预测样本的类别。具体的调用例子如下：

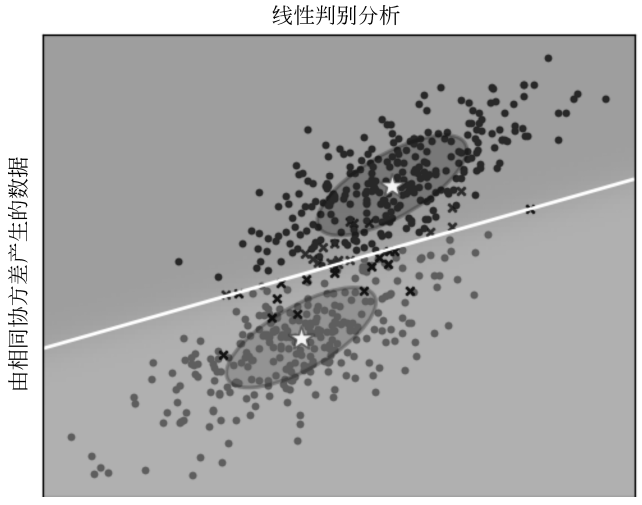

图 5.2　用线性判别分析对数据分类

```
y_pred=lda.fit(X,y).predict(X)
```

在 sklearn 的 discriminant_analysis 包中也提供用于执行二次判别分析的类：QuadraticDiscriminantAnalysis。其具体使用方法与 LinearDiscriminantAnalysis 类一样。

从上面介绍的内容可以看出：可由高斯判别分析推导出线性判别分析，但无法从线性判别分析推导出高斯判别分析。高斯判别分析是一种生成方法，而线性判别分析是一种判别方法，这说明生成方法和判别方法在某些情况下是有联系的。在第 4 章介绍 logistic 回归时，通过高斯判别模型和贝叶斯公式推导出了 logistic 回归。这也说明在某些情况下生成方法和判别方法会有联系。若假设 $p(\boldsymbol{x}|y=1)$ 和 $p(\boldsymbol{x}|y=0)$为两个不同的泊松分布，则通过类似的方式也可以得到 logistic 回归。这就说明 logistic 回归可用来对由高斯分布和泊松分布所产生的数据进行分类。而高斯判别分析只能对基于高斯分布的数据进行有效分类，不能对基于泊松分布的数据进行有效分类。但需要注意的是通过 logistic 回归却得不到高斯判别模型。

5.2　朴素贝叶斯

通过贝叶斯定理学习 $p(c_k|\boldsymbol{x})$时，最重要的一步是计算后验概率 $p(\boldsymbol{x}|c_k)$，5.1 节在介绍高斯判别模型时，假定 $p(\boldsymbol{x}|c_k)$的分布为多元高斯分布，但在很多情况下并不知道 $p(\boldsymbol{x}|c_k)$的真实分布，这时只有直接求 $p(\boldsymbol{x}|c_k)$。假设样本 $\boldsymbol{x}$ 包含 m 个特征，即 $\boldsymbol{x}=[f_1,f_2,\cdots,f_m]$，则有

$$p(\boldsymbol{x}\mid c_k)=p(f_1,f_2,\cdots,f_m\mid c_k) \tag{5.9}$$

式(5.9)是一个联合概率分布，并不知道这些特征之间的关系，因此要得到相应的概率分布是一件非常困难的事情。假设这些特征都是相互独立(也称为属性条件独立性假设(attribute conditional independence assumption))，则有

$$p(\boldsymbol{x}\mid c_k)=\prod_{i=1}^{m}p(f_i\mid c_k)$$

在这种假设条件下计算 $p(c_k|\boldsymbol{x})$ 的方法称为朴素贝叶斯(naive Bayes)方法。所得到的分类器称为朴素贝叶斯分类器(naïve Bayes classifier)。在这种假设条件下,式(5.2)为

$$\max_{c_k} p(c_k \mid \boldsymbol{x}_i) = \max_{c_k} p(c_k) \prod_{i=1}^{m} p(f_i \mid c_k) \tag{5.10}$$

从式(5.10)可以看出,若要得到朴素贝叶斯分类器,则必须要从训练数据集中估计先验概率 $p(c_k)$,同时还需要针对每个特征计算相应的条件概率 $p(f_i|c_k)$。下面介绍计算这两种概率的方法。

假设训练数据集 $\boldsymbol{X}=[(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_n,y_n)]$,其中第 i 个样本 $\boldsymbol{x}_i$ 有 m 个特征,即

$$\boldsymbol{x}=[f_1,f_2,\cdots,f_m]$$

第 i 个特征可能的取值为 $f_i \in \{v_{i1},v_{i2},\cdots,v_{iS_i}\}$,训练数据集中的样本总共有 K 个类,因此第 i 个类标记的可能值为 $y_i \in \{c_1,c_2,\cdots,c_k\}$。

(1) 计算先验概率的公式为

$$p(c_k)=\frac{\sum_{i=1}^{n} I(y_i=c_k)}{n},\quad k=1,2,\cdots,K \tag{5.11}$$

计算第 i 个特征的条件概率为

$$p(f_i=v_{il} \mid c_k)=\frac{\sum_{i=1}^{n} I(f_i=v_{il},y_i=c_k)}{\sum_{i=1}^{n} I(y_i=c_k)} \tag{5.12}$$

式中,$k=1,2,\cdots,K$;$i=1,2,\cdots,n$;$l=1,2,\cdots,S_i$。

(2) 计算给定样本 $\boldsymbol{x}$ 属于第 c_k 类的概率

$$p(\boldsymbol{x} \mid c_k)=p(c_k)\prod_{i=1}^{m} p(f_i \mid c_k)$$

式中,$k=1,2,\cdots,K$。

(3) 根据最大后验概率确定样本 $\boldsymbol{x}$ 的类别,即

$$y=\arg\max_{c_k} p(c_k \mid \boldsymbol{x})=\max_{c_k} p(c_k)\prod_{i=1}^{m} p(f_i \mid c_k)$$

上面介绍朴素贝叶斯分类器的具体实现过程时,假定特征的取值是离散型。但在实际应用中,特征经常取连续值,如将人的身高作为特征时,其取值的类型为连续型。对于这种取连续值的特征,在计算其条件概率时,可以假设相应的分布,例如,若第 i 个特征 f_i 取连续值,则可假设其条件概率密度为

$$p(f_i \mid c_k)=\frac{1}{\sqrt{2\pi\sigma_k^2}}\mathrm{e}^{\frac{-(f_i-\mu_k)^2}{2\sigma_k^2}}$$

式中,μ_k 为类别为 c_k 时所对应的特征 f_i 的样本均值,而 σ_k 则为相应的样本方差。下面举例说明 μ_k 和 σ_k 的计算方法。

假设有一个训练数据集,样本的类别为“男性”和“女性”,构成每个样本的特征分别为身高和体重,具体信息如表 5.1 所示。

表 5.1 身高和体重数据

性别	身高/cm	体重/kg
男性	178	90
男性	183	85
男性	170	75
男性	165	63
女性	160	55
女性	165	60
女性	162	55
女性	158	50
女性	163	58

计算每个类别对应的不同特征的均值和方差,其结果如表 5.2 所示。

表 5.2 计算得到的均值与方差

性别	身高的均值/cm	身高的方差/cm	体重的均值/kg	体重的方差/kg
男性	174	64.66	78.25	142.25
女性	161.6	7.3	55.6	14.3

在计算 p(身高|男性)的条件概率时,其均值和方差分别为 174cm 和 64.66cm。

上面介绍的这种朴素贝叶斯称为高斯朴素贝叶斯(Gaussian naive Bayes),在 sklearn 的 naive_bayes 包中有一个 GaussianNB 类,实例化这个类后就可以训练高斯朴素贝叶斯模型。朴素贝叶斯还有其他一些变种,如假定第 i 个特征 f_i 只取两种值:0 和 1。取 0 的概率为 p,则会得到伯努利朴素贝叶斯。这时计算条件概率的公式为

$$p(f_i=0 \mid c_k)=p(c_k)p$$

$$p(f_i=1 \mid c_k)=p(c_k)(1-p)$$

同样在 sklearn 的 naive_bayes 包中有一个 BernoulliNB 类,实例化这个类后就可以训练伯努利朴素贝叶斯模型。

下面的示例介绍如何用朴素贝叶斯根据天气情况预测用户是否外出爬山。样本主要由两个特征组成:天气和气温。具体数据如表 5.3 所示。

表 5.3 天气与人们是否爬山的数据

天气	气温	爬山	天气	气温	爬山
天晴	高	否	多云	中	是
多云	中	是	天晴	中	是
下雨	中	否	多云	低	是
下雨	低	否	下雨	高	否
多云	高	否			

这两个特征涉及的内容都是文字，需要将其转换成数值才能传递给朴素贝叶斯算法处理。在 sklearn 中，有一个名为 preprocessing 的包，这里面有一个 LabelEncoder 类，可对这个类进行实例化，然后调用 fit_transform()方法将每个特征中的不同文字转换成数字。例如：

```
weather=['高','中','中','低']
LabEncoder=preprocessing.LabelEncoder()
W=LabEncoder.fit_transform(weather)
```

变量 W 为列表(list)类型，相应的值为

```
[2 0 0 1]
```

按上面的方式可将表 5.3 中两个特征转换成数字，然后用 Python 的 zip()函数将它们合并成数字表示的样本。zip()函数能将两个列表合并成元组(tuple)。例如：

```
A=[1,2,3]
B=[4,5,6]
Zipped=zip(A,B)
```

可通过 list(Zipped)查看变量 Zipped 的值，其运行的结果如下：

```
[(1, 4), (2, 5), (3, 6)]
```

再将类别也转换为数字，这样就可以调用 naive_bayes 包中的相关算法进行学习。由于这个例子中的数据很简单，因此也可以手工来实现朴素贝叶斯。其计算步骤如下。

(1) 分别计算爬山和不爬山的先验概率。

$$p(\text{爬山}=\text{是})=\frac{4}{9}$$

$$p(\text{爬山}=\text{否})=\frac{5}{9}$$

(2) 针对两个特征的不同取值，分别计算在爬山和不爬山的条件下相应的概率：

$p(\text{天晴}\mid\text{爬山}=\text{是})=\frac{1}{4}$　　$p(\text{天晴}\mid\text{爬山}=\text{否})=\frac{1}{5}$

$p(\text{多云}\mid\text{爬山}=\text{是})=\frac{3}{4}$　　$p(\text{多云}\mid\text{爬山}=\text{否})=\frac{1}{5}$

$p(\text{下雨}\mid\text{爬山}=\text{是})=0$　　$p(\text{下雨}\mid\text{爬山}=\text{否})=\frac{3}{5}$

$p(\text{气温}=\text{高}\mid\text{爬山}=\text{是})=0$　　$p(\text{气温}=\text{高}\mid\text{爬山}=\text{否})=\frac{3}{5}$

$p(\text{气温}=\text{中}\mid\text{爬山}=\text{是})=\frac{3}{4}$　　$p(\text{气温}=\text{中}\mid\text{爬山}=\text{否})=\frac{1}{5}$

$p(\text{气温}=\text{低}\mid\text{爬山}=\text{是})=\frac{1}{4}$　　$p(\text{气温}=\text{低}\mid\text{爬山}=\text{否})=\frac{1}{5}$

要预测给定样本(天晴，气温低)是否会爬山，则可通过下面的方式来计算：

$$p(\text{爬山}=\text{是})p(\text{天晴}\mid\text{爬山}=\text{是})p(\text{气温}=\text{低}\mid\text{爬山}=\text{是})=\frac{4}{9}\times\frac{1}{4}\times\frac{1}{4}=\frac{4}{144}=\frac{1}{36}$$

$$p(\text{爬山}=\text{否})p(\text{天晴}\mid\text{爬山}=\text{否})p(\text{气温}=\text{低}\mid\text{爬山}=\text{否})=\frac{5}{9}\times\frac{1}{5}\times\frac{1}{5}=\frac{5}{225}=\frac{1}{45}$$

由于 $p(\text{爬山}=\text{是})p(\text{天晴}\mid\text{爬山}=\text{是})p(\text{气温}=\text{低}\mid\text{爬山}=\text{是})$ 大,因此预测的结果为"要爬山"。

在计算上面的各种条件概率时,有可能出现概率值为 0 的情况,这会影响后面的分类结果。为了解决这个问题,可以将式(5.11)和式(5.12)分别修改为

$$p(c_k)=\frac{\sum_{i=1}^{n}I(y_i=c_k)+\lambda}{n+K},\quad k=1,2,\cdots,K$$

和

$$p(f_i=v_{il}\mid c_k)=\frac{\sum_{i=1}^{n}I(f_i=v_{il},y_i=c_k)+\lambda}{\sum_{i=1}^{n}I(y_i=c_k)+S_i\lambda}$$

当 $\lambda=1$ 时,称为拉普拉斯光滑(Laplace smoothing);当 $\lambda=0$ 时,则就是经典的朴素贝叶斯的计算方法。

这里给出了朴素贝叶斯的简单实现过程。目前,贝叶斯分类已被广泛地应用于垃圾邮件分类、拼写纠正和医疗诊断等,并获得了不错的效果。

5.3 改进的朴素贝叶斯

朴素贝叶斯假设各个特征之间是条件独立的,但在现实应用中,这种假设通常很难成立。假设特征之间存在各种关系就得到了各种改进的朴素贝叶斯方法。下面简单介绍这些改进的方法。

(1) 半朴素贝叶斯分类器(semi-naive Bayes classifiers)。它就是一种改进的朴素贝叶斯分类器,它的基本思想是考虑一部分特征之间的相互依赖关系。经典的半朴素贝叶斯分类器会假定每个特征最多依赖一个其他特征,这种假设称为独依赖估计(one-dependent estimator,ODE)。独依赖估计又分很多种情形,如所有特征都依赖同一个特征,这种方法称为超父独依赖估计(super parent ODE)。

(2) 贝叶斯网(Bayes network),也称为信念网(belief network),由 Judea Pearl 在 1985 年首先提出,它通过有向无环图来描述特征之间的关系。贝叶斯网是一种概率图模型(probability graph model,PGM),它通常由两部分组成:贝叶斯网的结构和贝叶斯网的参数。若特征之间有依赖关系,则用一条边连接起来,参数用来描述这种依赖关系。贝叶斯网的困难在于无法完全知道网络结构,因此,贝叶斯网的学习首先要找出与训练数据集样本结构一致的网络结构。贝叶斯网结构学习算法主要有 3 种:①基于依赖统计分析的方法。该方法通常利用统计或信息论的方法分析特征之间的依赖关系,从而获得最优的网络结构。而节点之间的依赖关系通常由两点的互信息或者条件互信息决定。②基于评分搜索的方法。该方法由评分函数和搜索算子两部分构成,评分函数评价网络结构与训练样本结构的相似程度,搜索算子决定对网络结构空间的搜索过程。③混合方法。这种方法将上述两种

方法结合在一起，通过统计分析缩小网络结构空间，再对缩小后的网络结构空间进行评分搜索，从而得到最优的网络结构。

人们通常称有固定结构的贝叶斯网为静态贝叶斯网，还有一类贝叶斯网的结构会不断变化，人们称其为动态贝叶斯网(dynamic Bayesian network)。隐马尔可夫模型(hidden Markov model，HMM)是结构最简单的动态贝叶斯网，这是一种著名的有向图模型，主要用于时序数据建模，在自然语言处理、语音识别等领域有着广泛的应用。

5.4　总结

本章首先以贝叶斯决策理论为基础，得出分类问题的期望风险最小其实就是最大化后验概率，然后介绍了计算后验概率的两种方法：判别方法和生成方法。本章介绍了两种典型的生成方法：高斯判别分析和朴素贝叶斯。高斯判别分析方法假设似然函数服从多元正态分布，这些正态分布有相同的协方差矩阵，不同的均值向量，然后通过最大似然估计得到正态分布的参数。由高斯判别分析可以得到线性判别分析，这表明生成方法和判别方法有时会有一定联系。朴素贝叶斯方法假设特征之间是条件独立的，在这个假设下计算后验概率。这是一个很强的假设条件，后来研究人员适当放宽了这个假设条件，得到了多种改进的朴素贝叶斯方法，如半朴素贝叶斯方法等。朴素贝叶斯方法在邮件分类、拼写纠正等领域有着广泛的应用。

5.5　习题

(1) 如何得到式(5.2)？

(2) Fisher 线性判别分析(FLDA)是一个著名的监督学习方法。这种方法的基本思想：将样本点投影到一条直线或超平面上，使得投影后同类样本尽量靠在一起，而不同类的样本之间尽量分开。对于一个分类问题，FLDA 方法会先将样本投影到向量 $\boldsymbol{w}$ 上，然后分别计算类间(between-class)散度矩阵 $\boldsymbol{S}_{\mathrm{B}}$ 和类内(within-class)散度矩阵 $\boldsymbol{S}_{\mathrm{W}}$，然后求解下面这个目标函数来得到 $\boldsymbol{w}$，即

$$\max_{\boldsymbol{w}} f(\boldsymbol{w}) = \frac{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{B}}\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{W}}\boldsymbol{w}} \tag{5.13}$$

① 写出 $\boldsymbol{S}_{\mathrm{B}}$ 和 $\boldsymbol{S}_{\mathrm{W}}$ 的具体形式。

② 给出式(5.13)的求解方法

③ 假设有一个训练数据集，它的样本总共分为两类，每个样本的特征数为 2，第一类总共有 5 个样本，具体数据为

$$C_1 = \{(1,2),(1,2),(2,3),(3,3),(4,5)\}$$

第二类总共有 6 个样本，具体数据为

$$C_2 = \{(1,0),(2,1),(3,1),(3,2),(5,3),(6,5)\}$$

绘制这些训练样本的散点(scatter)图，不同类别的样本要用不同颜色标示出来。在这个训练数据集上用 Fisher 线性判别分析找到分类超平面。

(3) 通过表 5.4 中的数据学习一个朴素贝叶斯分类器，并确定样本 $\boldsymbol{x} = (2, D)$ 的类别。