第7章　贝叶斯分类器

刘家锋

哈尔滨工业大学

第7章　贝叶斯分类器

7.1 贝叶斯决策论

分类与概率

- **概率的角度看分类问题**
  - 将样例$\mathbf{x}$视作随机向量，类别标记y视作有N种取值的离散随机变量：
    $$y \in \mathcal{Y} = \{c_1, \cdots, c_N\}$$
  - 分类可以看作是在已知样例$\mathbf{x}$的条件下，对类别y的决策
  - y是随机的，因此任何的决策都有可能发生错误，分类问题自然希望发生决策错误的概率越小越好
- **类别的先验概率**
  - 如果我们不知道样例的属性$\mathbf{x}$，那么只能依据类别的先验概率$P(y)$来决策
  - 哪个类别的先验概率大，就判别样例属于哪个类别：
    $$y^* = \arg\max_{c \in \mathcal{Y}} P(y = c)$$

最小错误率

- **类别的后验概率**
  - 如果我们知道样例的属性$\mathbf{x}$，就可以依据类别的后验概率$P(y|\mathbf{x})$来决策
  - 哪个类别的后验概率大，就判别样例属于哪个类别：

$$y^* = \arg\max_{c \in \mathcal{Y}} P(y = c|\mathbf{x})$$

- **最小错误率判别**
  - 依据后验概率的判别，可以取得最小的错误率
  - 如果决策$y = c_i$，则当真实类别为$c_j, j \neq i$时发生错误，因此决策的错误率为：

$$P_i(\text{error}|\mathbf{x}) = \sum_{j \neq i} P(y = c_j|\mathbf{x}) = 1 - P(y = c_i|\mathbf{x})$$

最小化风险

- **条件风险**
  - 最小错误率认为所有的判别错误都是相同的
  - 如果将一个真实标记为c_j的样本误分类为c_i的损失为λ_{ij}，那么将$\mathbf{x}$判别为c_i类的条件风险为：
  $$R(c_i|\mathbf{x}) = \sum_{j=1}^{N} \lambda_{ij} P(y = c_j|\mathbf{x})$$
  - 依据最小化条件风险的准则判别为：
  $$y^* = \arg\min_{c \in \mathcal{Y}} R(c|\mathbf{x})$$
  - 最小错误率判别等价于：
  $$\lambda_{ij} = \begin{cases} 0, & \text{if} \quad i = j \\ 1, & \text{otherwise} \end{cases}$$

生成与判别模型

- **判别模型(discriminative models)**
  - 模型化后验概率$P(y|\mathbf{x})$来判别的方法，称为判别模型
  - 线性判别，SVM，神经网络和决策树都属于判别模型
- **生成模型(generative models)**
  - 模型化联合概率$P(\mathbf{x},y)$或类条件概率$p(\mathbf{x}|y)$来判别的方法，称为生成模型
  - 条件概率公式：
$$P(\mathbf{x},y)=P(y|\mathbf{x})p(\mathbf{x})=p(\mathbf{x}|y)P(y)$$
  - 贝叶斯公式：
$$P(y|\mathbf{x})=\frac{P(y)p(\mathbf{x}|y)}{p(\mathbf{x})}$$

贝叶斯判别

- **联合概率的判别**
  - $p(\mathbf{x})$是一个与类别无关的归一化因子，称为“证据”
  - 依据联合概率的判别等价于后验概率的判别：

$$\arg\max_{c\in\mathcal{Y}} P(\mathbf{x}, y=c) \quad \Leftrightarrow \quad \arg\max_{c\in\mathcal{Y}} P(y=c|\mathbf{x})$$

- **贝叶斯判别**
  - 依据贝叶斯公式可以得到：

$$\arg\max_{c\in\mathcal{Y}} p(\mathbf{x}|y=c)P(y=c) \quad \Leftrightarrow \quad \arg\max_{c\in\mathcal{Y}} P(y=c|\mathbf{x})$$

  - 先验概率$P(y)$可以利用先验知识，或者训练集中各个类别样本所占的比例来估计
  - 贝叶斯判别的学习，主要是估计类条件概率$p(\mathbf{x}|y)$

7.2 极大似然估计

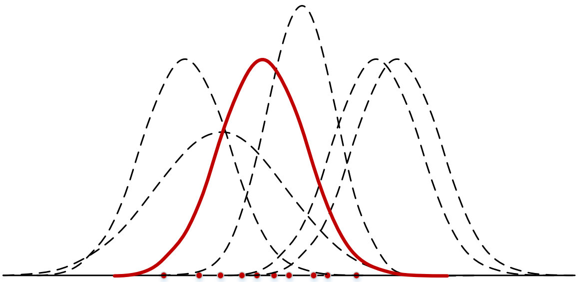
极大似然估计

- **概率分布的参数估计**
  - 假定类条件概率$p(\mathbf{x}|y=c)$具有确定的分布形式，并且被参数$\boldsymbol{\theta}_c$唯一确定
  - 令D_c表示训练集D中第c类样本组成的集合，并且是独立同分布的样本
  - 贝叶斯分类器的学习就是利用数据集D_c来估计参数$\boldsymbol{\theta}_c$，其中$c \in \mathcal{Y} = \{c_1, \cdots, c_N\}$
- **似然函数**
  - 定义给定参数$\boldsymbol{\theta}_c$条件下，样本集D_c中样本发生的联合概率为似然函数
  - 似然函数为参数$\boldsymbol{\theta}_c$的函数，根据独立同分布假设有：

$$p(D_c|\boldsymbol{\theta}_c) = \prod_{\mathbf{x} \in D_c} p(\mathbf{x}|\boldsymbol{\theta}_c)$$

极大似然估计

- **极大似然估计**
  - 极大似然估计的思路是在给定的分布形式中，找到一个最有可能产生出训练集D_c的分布
  - 给定形式的分布由参数$\boldsymbol{\theta}_c$唯一确定，因此以最大化似然函数的参数作为估计结果



极大似然估计

- **对数似然函数**
  - 概率密度函数的值往往比较小，连乘容易造成计算下溢
  - 对数函数是单调上升的，一般以对数似然函数代替似然函数作为最大似然估计的优化目标：

$$LL(\boldsymbol{\theta}_c) = \ln p(D_c|\boldsymbol{\theta}_c) = \sum_{\mathbf{x} \in D_c} \ln p(\mathbf{x}|\boldsymbol{\theta}_c)$$

- **极大似然估计**
  - 极大似然估计需要求解如下优化问题：

$$\hat{\boldsymbol{\theta}}_c = \arg\max_{\boldsymbol{\theta}_c} LL(\boldsymbol{\theta}_c)$$

例7.1 高斯分布参数估计

类别c的训练集$D_c = \{x_1, \cdots, x_{m_c}\}$，服从参数$\boldsymbol{\theta}_c = (\mu_c, \sigma_c^2)^t$的1维高斯分布：

$$p(x|\mu_c, \sigma_c^2) = \frac{1}{\sqrt{2\pi}\sigma_c} \exp\left[-\frac{(x-\mu_c)^2}{2\sigma_c^2}\right]$$

对数似然函数：

$$LL(\mu_c, \sigma_c^2) = \sum_{i=1}^{m_c} \ln p(x_i|\mu_c, \sigma_c^2) = \sum_{i=1}^{m_c} -\frac{1}{2}\left[\ln 2\pi + \ln \sigma_c^2 + \frac{(x_i-\mu_c)^2}{\sigma_c^2}\right]$$

计算偏导数，求极值：

$$\frac{\partial LL(\mu_c, \sigma_c^2)}{\partial \mu_c} = \sum_{i=1}^{m_c} \frac{1}{\sigma_c^2}(x_i - \mu_c) = 0$$

$$\frac{\partial LL(\mu_c, \sigma_c^2)}{\partial \sigma_c^2} = \sum_{i=1}^{m_c} \left[-\frac{1}{2\sigma_c^2} + \frac{(x_i-\mu_c)^2}{2\sigma_c^4}\right] = 0$$

例7.1 高斯分布参数估计

求解方程，得到参数的极大似然估计：

$$\hat{\mu}_c = \frac{1}{m_c}\sum_{i=1}^{m_c} x_i, \qquad \hat{\sigma}_c^2 = \frac{1}{m_c}\sum_{i=1}^{m_c}(x_i - \hat{\mu}_c)^2$$

样本集D_c服从d维高斯分布：

$$p(\mathbf{x}|\boldsymbol{\mu}_c, \Sigma_c) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_c|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^t \Sigma_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c)\right]$$

同样方法，可以得到多元高斯分布参数的极大似然估计：

$$\hat{\boldsymbol{\mu}}_c = \frac{1}{|D_c|}\sum_{\mathbf{x}\in D_c} \mathbf{x}$$

$$\hat{\Sigma}_c = \frac{1}{|D_c|}\sum_{\mathbf{x}\in D_c} (\mathbf{x}-\hat{\boldsymbol{\mu}}_c)(\mathbf{x}-\hat{\boldsymbol{\mu}}_c)^t$$

7.3 朴素贝叶斯分类器

Naïve Bayes Classifier

- **朴素贝叶斯分类器**
  - 有限训练样本估计高维联合概率(密度)$p(\mathbf{x}|y=c)$存在困难
  - 朴素贝叶斯对模型进行了简化，假设$\mathbf{x}$的属性之间是相互独立的，即：

$$p(\mathbf{x}|y=c)=\prod_{i=1}^{d}p(x_i|y=c)$$

  - 相应的贝叶斯判别：

$$y^*=\arg\max_{c\in\mathcal{Y}}P(y=c)\prod_{i=1}^{d}p(x_i|y=c)$$

  - 学习时，可以由D_c单独估计每个属性的分布$p(x_i|y=c)$

例7.2 朴素贝叶斯分类(7-2 NaiveBayesian.ipynb)

1. 连续属性分类

西瓜数据集

编号	密度	含糖率	好瓜	编号	密度	含糖率	好瓜
1	0.697	0.460	是	9	0.666	0.091	否
2	0.774	0.376	是	10	0.243	0.267	否
3	0.634	0.264	是	11	0.245	0.057	否
4	0.608	0.318	是	12	0.343	0.099	否
5	0.556	0.215	是	13	0.639	0.161	否
6	0.403	0.237	是	14	0.657	0.198	否
7	0.481	0.149	是	15	0.360	0.370	否
8	0.437	0.211	是	16	0.593	0.042	否
				17	0.719	0.103	否

假设数据集中的正例和反例分别服从2维高斯分布，采用最小错误率贝叶斯准则判别下列测试数据：

编号	密度	含糖率	好瓜
测试1	0.697	0.460	?

例7.2-1 连续属性朴素贝叶斯分类

估计正例和反例类别的先验概率：

$$P(\text{好瓜=是}) = \frac{8}{17} \approx 0.471, \qquad P(\text{好瓜=否}) = \frac{9}{17} \approx 0.529$$

估计正例和反例类别两个属性的均值：

$$\hat{\mu}_{\text{密度|是}} = \frac{1}{8}\sum_{i=1}^{8}\mathbf{x}_i^{\text{密度}} \approx 0.574, \quad \hat{\mu}_{\text{含糖|是}} = \frac{1}{8}\sum_{i=1}^{8}\mathbf{x}_i^{\text{含糖}} \approx 0.279$$

$$\hat{\mu}_{\text{密度|否}} = \frac{1}{9}\sum_{i=9}^{17}\mathbf{x}_i^{\text{密度}} \approx 0.496, \quad \hat{\mu}_{\text{含糖|否}} = \frac{1}{9}\sum_{i=9}^{17}\mathbf{x}_i^{\text{含糖}} \approx 0.154$$

估计正例和反例类别两个属性的方差：

$$\hat{\sigma}^2_{\text{密度|是}} = \frac{1}{8}\sum_{i=1}^{8}\left(x_i^{\text{密度}} - \hat{\mu}_{\text{是}}^{\text{密度}}\right)^2 \approx 0.0146, \quad \hat{\sigma}^2_{\text{含糖|是}} = \frac{1}{8}\sum_{i=1}^{8}\left(x_i^{\text{含糖}} - \hat{\mu}_{\text{是}}^{\text{含糖}}\right)^2 \approx 0.0089$$

$$\hat{\sigma}^2_{\text{密度|否}} = \frac{1}{9}\sum_{i=9}^{17}\left(x_i^{\text{密度}} - \hat{\mu}_{\text{否}}^{\text{密度}}\right)^2 \approx 0.0337, \quad \hat{\sigma}^2_{\text{含糖|否}} = \frac{1}{9}\sum_{i=9}^{17}\left(x_i^{\text{含糖}} - \hat{\mu}_{\text{否}}^{\text{含糖}}\right)^2 \approx 0.0103$$

例7.2-1 连续属性朴素贝叶斯分类

计算正例的条件概率密度：

$$p\left(密度=0.697|是\right)=\frac{1}{\sqrt{2\pi}\sigma_{密度|是}}\exp\left[-\frac{\left(0.697-\hat{\mu}_{密度|是}\right)^2}{2\hat{\sigma}^2_{密度|是}}\right]\approx 1.967$$

$$p\left(含糖=0.460|是\right)=\frac{1}{\sqrt{2\pi}\sigma_{含糖|是}}\exp\left[-\frac{\left(0.460-\hat{\mu}_{含糖|是}\right)^2}{2\hat{\sigma}^2_{含糖|是}}\right]\approx 0.671$$

计算反例的条件概率密度：

$$p\left(密度=0.697|否\right)=\frac{1}{\sqrt{2\pi}\sigma_{密度|否}}\exp\left[-\frac{\left(0.697-\hat{\mu}_{密度|否}\right)^2}{2\hat{\sigma}^2_{密度|否}}\right]\approx 1.193$$

$$p\left(含糖=0.460|否\right)=\frac{1}{\sqrt{2\pi}\sigma_{含糖|否}}\exp\left[-\frac{\left(0.460-\hat{\mu}_{含糖|否}\right)^2}{2\hat{\sigma}^2_{含糖|否}}\right]\approx 0.042$$

例7.2-1 连续属性朴素贝叶斯分类

贝叶斯判别：

$$P(\text{好瓜=是})p\left(\text{密度}=0.697|\text{是}\right)p\left(\text{含糖}=0.460|\text{是}\right)\approx 0.471\times 1.967\times 0.671=0.622$$
$$P(\text{好瓜=否})p\left(\text{密度}=0.697|\text{否}\right)p\left(\text{含糖}=0.460|\text{否}\right)\approx 0.529\times 1.193\times 0.042=0.026$$

由于：

$$p(\mathbf{x}|\text{是})P(\text{是})>p(\mathbf{x}|\text{否})P(\text{否})$$

判别测试数据**x**“是”好瓜。

学习连续属性朴素贝叶斯分类器

```
import numpy as np
from sklearn.naive_bayes import GaussianNB

X_cont = np.array([ [0.697,0.460], [0.774,0.376], [0.634,0.264], [0.608,0.318],
                    [0.556,0.215], [0.403,0.237], [0.481,0.149], [0.437,0.211],
                    [0.666,0.091], [0.243,0.267], [0.245,0.057], [0.343,0.099],
                    [0.639,0.161], [0.657,0.198], [0.360,0.370], [0.593,0.042],
                    [0.719,0.103] ] )
y = ['是', '是', '是', '是', '是', '是', '是', '是',
     '否', '否', '否', '否', '否', '否', '否', '否', '否']

nb_cont = GaussianNB().fit(X_cont,y)
print("Parameters:\n\t Prior of classes:", nb_cont.class_prior_)
print("\t",chr(956),":",nb_cont.theta_[0],nb_cont.theta_[1])
print("\t",chr(963),":",nb_cont.var_[0],nb_cont.var_[1])
```

```
Parameters:
        Prior of classes: [0.52941176 0.47058824]
        μ : [0.49611111 0.15422222] [0.57375 0.27875]
        σ : [0.03370254 0.01032862] [0.01460844 0.00891244]
```

显示分类结果

```python
print("\nPredict probabilities:")
print(nb_cont.predict_proba(X_cont))
print("\nPredict labels:")
print(nb_cont.predict(X_cont))
print("Score of train set:", nb_cont.score(X_cont,y))
```

```
Predict probabilities:
[[0.04164757 0.95835243]
 [0.11946261 0.88053739]
 [0.24918736 0.75081264]
 [0.15038577 0.84961423]
 [0.40922077 0.59077923]
 [0.56498589 0.43501411]
 [0.70271521 0.29728479]
 [0.57828441 0.42171559]
 [0.78129434 0.21870566]
 [0.85959214 0.14040786]
 [0.99090406 0.00909594]
 [0.94079438 0.05920562]
 [0.56082358 0.43917642]
 [0.43838558 0.56161442]
 [0.29487969 0.70512031]
 [0.88435478 0.11564522]
 [0.77153256 0.22846744]]
```

```
Predict labels:
['是' '是' '是' '是' '是' '否' '否' '否' '否' '否' '否' '否' '否' '是' '是' '否' '否']
Score of train set: 0.7058823529411765
```

例7.2-2 离散属性朴素贝叶斯分类

2. 离散属性分类

西瓜数据集

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

例7.2-2 离散属性朴素贝叶斯分类

采用最小错误率贝叶斯准则判别下列测试数据：

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
测试2	青绿	蜷缩	浊响	清晰	稍凹	硬滑	?

计算正例和反例中测试2所有属性发生的联合概率：

$$P(\mathbf{x}|\text{是}) = P(\text{青绿,蜷缩,浊响,清晰,稍凹,硬滑}|\text{是}) = 0$$
$$P(\mathbf{x}|\text{否}) = P(\text{青绿,蜷缩,浊响,清晰,稍凹,硬滑}|\text{否}) = 0$$

无法判别测试数据$\mathbf{x}$。

例7.2-2 离散属性朴素贝叶斯分类

采用朴素贝叶斯模型分类：

$$
\begin{aligned}
P(\mathbf{x}|\text{是}) &= P(\text{青绿,蜷缩,浊响,清晰,稍凹,硬滑}|\text{是}) \\
&= P_{\text{青绿}|\text{是}} \times P_{\text{蜷缩}|\text{是}} \times P_{\text{浊响}|\text{是}} \times P_{\text{清晰}|\text{是}} \times P_{\text{稍凹}|\text{是}} \times P_{\text{硬滑}|\text{是}} \\
&= \frac{3}{8} \times \frac{5}{8} \times \frac{6}{8} \times \frac{7}{8} \times \frac{3}{8} \times \frac{6}{8} = 0.0433 \\
P(\mathbf{x}|\text{否}) &= P(\text{青绿,蜷缩,浊响,清晰,稍凹,硬滑}|\text{否}) \\
&= P_{\text{青绿}|\text{否}} \times P_{\text{蜷缩}|\text{否}} \times P_{\text{浊响}|\text{否}} \times P_{\text{清晰}|\text{否}} \times P_{\text{稍凹}|\text{否}} \times P_{\text{硬滑}|\text{否}} \\
&= \frac{3}{9} \times \frac{3}{9} \times \frac{4}{9} \times \frac{2}{9} \times \frac{3}{9} \times \frac{6}{9} = 0.0024
\end{aligned}
$$

贝叶斯判别：

$$
\begin{aligned}
P(\mathbf{x}|\text{是})P(\text{是}) &= 0.0433 \times 0.471 = 0.0204 \\
P(\mathbf{x}|\text{否})P(\text{否}) &= 0.0024 \times 0.529 = 0.0013
\end{aligned}
$$

$P(\mathbf{x}|\text{是})P(\text{是}) > P(\mathbf{x}|\text{否})P(\text{否})$，判别测试数据$\mathbf{x}$“是”好瓜。

数据准备

```
X_discret = np.array([
    ['青绿', '蜷缩', '浊响', '清晰', '凹陷', '硬滑'],
    ['乌黑', '蜷缩', '沉闷', '清晰', '凹陷', '硬滑'],
    ['乌黑', '蜷缩', '浊响', '清晰', '凹陷', '硬滑'],
    ['青绿', '蜷缩', '沉闷', '清晰', '凹陷', '硬滑'],
    ['浅白', '蜷缩', '浊响', '清晰', '凹陷', '硬滑'],
    ['青绿', '稍蜷', '浊响', '清晰', '稍凹', '软粘'],
    ['乌黑', '稍蜷', '浊响', '稍糊', '稍凹', '软粘'],
    ['乌黑', '稍蜷', '浊响', '清晰', '稍凹', '硬滑'],
    ['乌黑', '稍蜷', '沉闷', '稍糊', '稍凹', '硬滑'],
    ['青绿', '硬挺', '清脆', '清晰', '平坦', '软粘'],
    ['浅白', '硬挺', '清脆', '模糊', '平坦', '硬滑'],
    ['浅白', '蜷缩', '浊响', '模糊', '平坦', '软粘'],
    ['青绿', '稍蜷', '浊响', '稍糊', '凹陷', '硬滑'],
    ['浅白', '稍蜷', '沉闷', '稍糊', '凹陷', '硬滑'],
    ['乌黑', '稍蜷', '浊响', '清晰', '稍凹', '软粘'],
    ['浅白', '蜷缩', '浊响', '模糊', '平坦', '硬滑'],
    ['青绿', '蜷缩', '沉闷', '稍糊', '稍凹', '硬滑']
])
```

属性数据转换

```
from sklearn import preprocessing

print("Original features:\n", X_discret)

le = preprocessing.LabelEncoder()
for col in range(6):
    f = le.fit_transform(X_discret[:,col])
    X_discret[:,col] = f
print("\nTransformed features:\n", X_discret)

X_discret = X_discret.astype(np.uint8)
print("\nTransformed digital features:\n", X_discret)
```

属性数据转换

```
Original features:
 [['青绿' '蜷缩' '浊响' '清晰' '凹陷' '硬滑']
 ['乌黑' '蜷缩' '沉闷' '清晰' '凹陷' '硬滑']
 ['乌黑' '蜷缩' '浊响' '清晰' '凹陷' '硬滑']
 ['青绿' '蜷缩' '沉闷' '清晰' '凹陷' '硬滑']
 ['浅白' '蜷缩' '浊响' '清晰' '凹陷' '硬滑']
 ['青绿' '稍蜷' '浊响' '清晰' '稍凹' '软粘']
 ['乌黑' '稍蜷' '浊响' '稍糊' '稍凹' '软粘']
 ['乌黑' '稍蜷' '浊响' '清晰' '稍凹' '硬滑']
 ['乌黑' '稍蜷' '沉闷' '稍糊' '稍凹' '硬滑']
 ['青绿' '硬挺' '清脆' '清晰' '平坦' '软粘']
 ['浅白' '硬挺' '清脆' '模糊' '平坦' '硬滑']
 ['浅白' '蜷缩' '浊响' '模糊' '平坦' '软粘']
 ['青绿' '稍蜷' '浊响' '稍糊' '凹陷' '硬滑']
 ['浅白' '稍蜷' '沉闷' '稍糊' '凹陷' '硬滑']
 ['乌黑' '稍蜷' '浊响' '清晰' '稍凹' '软粘']
 ['浅白' '蜷缩' '浊响' '模糊' '平坦' '硬滑']
 ['青绿' '蜷缩' '沉闷' '稍糊' '稍凹' '硬滑']]
```

```
Transformed features:
 [['2' '2' '1' '1' '0' '0']
 ['0' '2' '0' '1' '0' '0']
 ['0' '2' '1' '1' '0' '0']
 ['2' '2' '0' '1' '0' '0']
 ['1' '2' '1' '1' '0' '0']
 ['2' '1' '1' '1' '2' '1']
 ['0' '1' '1' '2' '2' '1']
 ['0' '1' '1' '1' '2' '0']
 ['0' '1' '0' '2' '2' '0']
 ['2' '0' '2' '1' '1' '1']
 ['1' '0' '2' '0' '1' '0']
 ['1' '2' '1' '0' '1' '1']
 ['2' '1' '1' '2' '0' '0']
 ['1' '1' '0' '2' '0' '0']
 ['0' '1' '1' '1' '2' '1']
 ['1' '2' '1' '0' '1' '0']
 ['2' '2' '0' '2' '2' '0']]
```

```
Transformed digital features:
 [[2 2 1 1 0 0]
 [0 2 0 1 0 0]
 [0 2 1 1 0 0]
 [2 2 0 1 0 0]
 [1 2 1 1 0 0]
 [2 1 1 1 2 1]
 [0 1 1 2 2 1]
 [0 1 1 1 2 0]
 [0 1 0 2 2 0]
 [2 0 2 1 1 1]
 [1 0 2 0 1 0]
 [1 2 1 0 1 1]
 [2 1 1 2 0 0]
 [1 1 0 2 0 0]
 [0 1 1 1 2 1]
 [1 2 1 0 1 0]
 [2 2 0 2 2 0]]
```

学习离散属性朴素贝叶斯分类器

```python
from sklearn.naive_bayes import CategoricalNB

nb_discret = CategoricalNB().fit(X_discret,y)

print("\nPredict probabilities:\n", nb_discret.predict_proba(X_discret))
print("\nPredict labels:", nb_discret.predict(X_discret))
print("Score of train set:", nb_discret.score(X_discret,y))
```

```
Predict probabilities:
 [[0.05580349 0.94419651]
 [0.06208417 0.93791583]
 [0.03424653 0.96575347]
 [0.09936112 0.90063888]
 [0.1287331  0.8712669 ]
 [0.22810193 0.77189807]
 [0.54171145 0.45828855]
 [0.11737074 0.88262926]
 [0.62333002 0.37666998]
 [0.93708157 0.06291843]
 [0.99665408 0.00334592]
 [0.95457414 0.04542586]
 [0.42488068 0.57511932]
 [0.77515921 0.22484079]
 [0.15060222 0.84939778]
 [0.94033562 0.05966438]
 [0.59530102 0.40469898]]
```

```
Predict labels: ['是' '是' '是' '是' '是' '是' '否' '是' '否' '否' '否' '否' '是' '否' '是' '否' '否']
Score of train set: 0.8235294117647058
```

拉普拉斯修正

- **未出现的属性值**
  - 某个属性值在训练集没有与某个类别同时出现，则该属性的条件概率为0
  - 如果在测试数据中该属性值出现，直接将其判别为不属于此类，是不合理的

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
测试2	青绿	蜷缩	清脆	清晰	凹陷	硬滑	0.697	0.460	?

$$p(\mathbf{x}|\text{是})P(\text{是}) = 0 \quad < \quad p(\mathbf{x}|\text{否})P(\text{否}) \approx 2.56 \times 10^{-5}$$

判别“测试2”样本“不是”好瓜

拉普拉斯修正

- **概率估计的平滑**
  - 拉普拉斯修正可以对概率估计值进行平滑
  - 令N表示类别数，N_i表示第i个属性的取值数，D_{c,x_i}表示类别c的训练集中属性i取值x_i的样本集合
  - 类别c的先验概率和条件概率估计的修正为：

$$\hat{P}(y=c)=\frac{|D_c|+1}{|D|+N}, \qquad \hat{P}(x_i|y=c)=\frac{|D_{c,x_i}|+1}{|D_c|+N_i}$$

  - 修正后的概率估计：

$$\hat{P}(\text{好瓜}=\text{是})=\frac{8+1}{17+2}\approx 0.474, \quad \hat{P}(\text{好瓜}=\text{否})=\frac{9+1}{17+2}\approx 0.526$$

$$\hat{P}_{\text{清脆}|\text{是}}=\hat{P}(\text{敲声}=\text{清脆}|\text{好瓜}=\text{是})=\frac{0+1}{8+3}\approx 0.091$$

$$\hat{P}_{\text{清脆}|\text{否}}=\hat{P}(\text{敲声}=\text{清脆}|\text{好瓜}=\text{否})=\frac{2+1}{9+3}=0.25$$

例7.2-3 混合属性朴素贝叶斯分类

西瓜数据集

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	0.697	0.460	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	0.774	0.376	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	0.634	0.264	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	0.608	0.318	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	0.556	0.215	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	0.403	0.237	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	0.481	0.149	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	0.437	0.211	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	0.666	0.091	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	0.243	0.267	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	0.245	0.057	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	0.343	0.099	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	0.639	0.161	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	0.657	0.198	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	0.360	0.370	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	0.593	0.042	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	0.719	0.103	否

混合属性朴素贝叶斯分类器

```python
proba = nb_cont.predict_proba(X_cont) * nb_discret.predict_proba(X_discret)
                                / nb_cont.class_prior_
print("Predict probabilies:\n", proba)

predict_label = proba.argmax(axis=1)
print("\nPredict labels:", predict_label)
```

```
Predict probabilies:
 [[4.38992896e-03 1.92285515e+00]
 [1.40093923e-02 1.75497366e+00]
 [1.61194031e-02 1.54083732e+00]
 [2.82247196e-02 1.62604067e+00]
 [9.95071565e-02 1.09379357e+00]
 [2.43429373e-01 7.13546422e-01]
 [7.19041208e-01 2.89514707e-01]
 [1.28205821e-01 7.90964353e-01]
 [9.19896863e-01 1.75057191e-01]
 [1.52151503e+00 1.87727632e-02]
 [1.86544510e+00 6.46727379e-05]
 [1.69633175e+00 5.71511598e-03]
 [4.50090307e-01 5.36730044e-01]
 [6.41879611e-01 2.68331892e-01]
 [8.38846808e-02 1.27272120e+00]
 [1.57078167e+00 1.46622887e-02]
 [8.67555558e-01 1.96478649e-01]]
```

```
Predict labels: [1 1 1 1 1 1 0 1 0 0 0 0 1 0 1 0 0]
```