

**模式识别大作业**

题 目 基于朴素贝叶斯的Wine数据集分类

学 院 信息科学与工程学院

专 业 控制科学与工程

组 员 张鑫婷

指导教师 赵海涛

**完成日期： 2018 年 10 月24日**

**基于朴素贝叶斯的Wine数据集分类**

组员：张鑫婷

本学期的模式识别原理与应用课程讲述了很多数据处理的方法，梯度下降法和最小二乘法可以用于优化问题，奇异值分解和主成分分析可以用于降维，贝叶斯决策和决策树可以用于数据分类等等，经过一系列的学习，我对分类器比较感兴趣，因此，我选择了其中一个问题进行了实验。

下面我用贝叶斯决策进行具体的实验，将Wine数据集作为实验对象，利用朴素贝叶斯方法训练Wine数据集的分类。

1. **Wine数据集**

Wine数据集是最流行的4个机器学习数据集中的一个，这是对在意大利同一地区生产的3种不同品种的酒，做大量分析所得出的数据。

这些数据包括了3种酒中13种不同成分的数量。13种成分分别为：Alcohol，Malicacid，Ash，Alcalinity of ash，Magnesium，Total phenols，Flavanoids，Nonflavanoid phenols，Proanthocyanins，Color intensity，Hue，OD280/OD315 of diluted wines，Proline。

Wine 数据集可以表示为13×178的矩阵，表示Wine数据集共有178个样本（列），每个样本都包含13种成分的数据（行），Wine数据集的178个样本可以分为3类，其中第1类有59个样本（第1-59列），第2类有71个样本（第60-130列），第3类有48个样本（第131-178列）。

1. **贝叶斯分类**

**3.1．贝叶斯决策**

贝叶斯决策 (Bayesian Decision Theory)就是在不完全情报下，对部分未知的状态用[主观概率](https://baike.so.com/doc/2264987-2396232.html)（先验概率）估计，然后用贝叶斯公式对发生概率进行修正，最后再利用期望值和修正概率（后验概率）做出最优决策。这里的贝叶斯公式即为：

(3.1)

其中，是条件概率密度函数（likelihood），为先验概率(priori probability)，为后验概率（posterior probability）, p(**x**)为证据因子（evidence）,保证各类的后验概率和为1。

(3.2)

贝叶斯决策属于[风险型决策](https://baike.so.com/doc/5662931-5875587.html)，决策者虽不能控制客观因素的变化，但却掌握其变化的可能状况及各状况的分布概率，并利用期望值即未来可能出现的平均状况作为决策准则。

[贝叶斯决策理论](https://baike.so.com/doc/5707680-5920401.html)方法是统计模型决策中的一个基本方法，其基本思想是:

1、已知条件[概率密度](https://baike.so.com/doc/298847-316371.html)参数表达式和先验概率。

2、利用贝叶斯公式转换成[后验概率](https://baike.so.com/doc/6550112-6763859.html)。

3、根据后验概率大小进行决策分类。

**3.2.朴素贝叶斯**

在使用贝叶斯决策的过程中，我们需要估计条件概率密度函数和先验概率。的估计相对简单，而当**x**的维度很高时，估计的数据量就会非常大。

朴素贝叶斯法是基于贝叶斯定理与特征条件独立假设的分类方法 。它可以很好地解决这个问题，其思路就是假设**x**的不同维度相互独立，因此条件概率密度函数可以写成：

(3.3)

这样的估计就转化为𝐷个相互独立的一维随机变量的密度估计问题，在得到和后，就可以按照贝叶斯决策规则进行分类。

1. **实验思路**

将一组数据进行分类，可以利用贝叶斯决策、k近邻分类器、决策树等方法，本实验利用朴素贝叶斯对Wine数据集进行分类。

Wine数据集中，每一类的数目均不同，本实验取118个样本用于训练，60个样本用于预测，其中，将预测数据数目保持一致，即从每一类中选取后20个样本作为预测数据，其余作为训练数据。为简化实验，将Wine数据集的每一类的每一维都假设为正态分布，并分别求其均值和方差。

在预测数据时，采用朴素贝叶斯方法来处理似然函数的估计问题，通过比较各数据的后验概率，对数据进行分类，即预测数据中，对每个样本都计算该样本属于第一、第二和第三类的后验概率，并将数据归到后验概率最大的那一类。

最后计算分类正确的概率，并作每个样本在每一维度下分好类的散点图，观察数据分布。

对Wine数据集的分类流程可以如图3-1所示。



图3-1 算法流程

1. **朴素贝叶斯算法实现**

**4.1.程序说明**

本实验使用MATLAB进行编程，程序中各变量的名称和维度说明如表4-1所示：

表4-1 实验变量

|  |  |  |
| --- | --- | --- |
| 变量名 | 名称说明 | 数据说明（维度） |
| winedata | 原始数据集 | 13×178维矩阵 |
| traindata | 训练数据集 | 13×118维矩阵 |
| testdata | 测试数据集 | 13×60维矩阵 |
| mu | 每一类每一维度数据的均值 | 13×3维矩阵 |
| sigma | 每一类每一维度数据的方差 | 13×3维矩阵 |
| priori | 先验概率 | 3×1维矩阵 |
| likelihood | 条件概率密度函数 | 3×1维矩阵 |
| posterior | 后验概率 | 3×60维矩阵 |
| category | 测试数据分类结果 | 1×60维矩阵 |

本实验的输出变量为true\_probabolity,用于计算数据分类的准确率，同时分别统计每类分错类别的数据数。

**4.2．程序实现**

整个程序可以分为六部分，分别为获取训练数据和测试数据、计算训练数据的参数、分类训练、分类、计算准确率和作图。

计算训练数据的参数是训练和分类数据的基础，本实验中，将每一类每一维度的数据都假定为各自服从正态分布，因此，此处的参数即为均值（mu）和方差(sigma)，其具体程序如下：

mu=zeros(13,3);

sigma=zeros(13,3);

for i=1:13

[mu(i,1),sigma(i,1)]=normfit(traindata1(i,:));

[mu(i,2),sigma(i,2)]=normfit(traindata2(i,:));

[mu(i,3),sigma(i,3)]=normfit(traindata3(i,:));

end

分类训练用到了朴素贝叶斯方法，计算出每组数据的后验概率后，将其分到后验概率最大的那一类，分类训练和分类的具体程序为：

posterior=zeros(3,60);

priori=zeros(3,1);

category=zeros(1,30);

for i=1:60

for j=1:3

if j==1

priori(j,1)=59/178;

elseif j==2

priori(j,1)=71/178;

else

priori(j,1)=48/178;

end

likelihood=ones(3,1);

for d=1:13

likelihood(j,1)=likelihood(j,1)\*normpdf(testdata(d,i),mu(d,j),sigma(d,j));

end

posterior(j,i)=likelihood(j,1)\*priori(j,1);

end

A=posterior(:,i)';

[c,s]=max(A);

category(1,i)=s;

end

分完类后，分别统计第一类、第二类和第三类中分类正确和分类错误的数目，以下为计算第一类中分类正确和错误的数目的程序（省略初始化过程）：

for k=1:20

if category(k)==1

category\_true1=category\_true1+1;

else

category\_false1=category\_false1+1;

end

end

同时计算分类的正确率，并作图展现各个维度下的数据分类结果。

**4.3.实验结果**

运行程序，实验结果如图4-1所示：

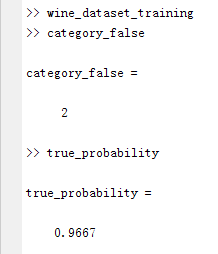


图4-1 程序运行结果

图4-2为Wine数据集的第一维度下的分类情况，从图中可以发现，该维度的60个数据中，第一类和第二类中各有一个数据分类错误。说明用于测试的13×60维的数据中，有两组数据分类错误。

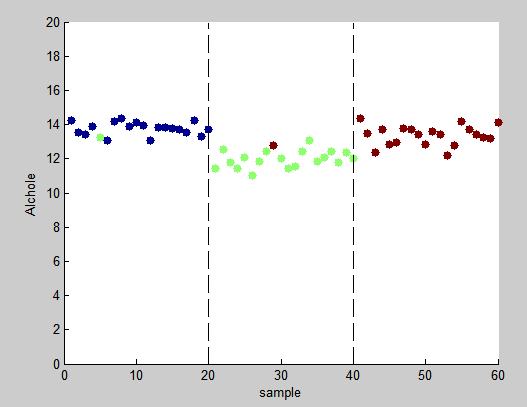


图4-2 第一维数据分类情况

1. **小组分工**

程序设计及编写：张鑫婷

程序调试：张鑫婷

实验报告撰写：张鑫婷

1. **小结**

从以上实验中可以发现，贝叶斯决策在数据分类中有较高的精度（本实验的准确率达到了96.67%），且朴素贝叶斯方法能够方便快捷地估计条件概率密度，简化贝叶斯决策过程。

Wine 数据集的数据较少，在整个程序中耗时不长（可以说是立刻跑完程序），朴素贝叶斯的优越性在本实验中没有得到很好的体现，若处理的数据非常多时，可以更好地展现朴素贝叶斯在数据处理中的优越性，本实验还可以针对更大的数据集和更实际的问题进行拓展，此处不再继续。

附：文件说明

本次附件一共包含：

1. 大作业报告
2. 最终的MATLAB实现程序源代码：wine\_dataset\_training.m