机器学习

第6章 贝叶斯学习

概述

- 贝叶斯推理提供了一种概率手段，基于如下的假定：待考察的量遵循某概率分布，且可根据这些概率及已观察到的数据进行推理，以作出最优的决策。
- 贝叶斯推理为衡量多个假设的置信度提供了定量的方法
- 贝叶斯推理为直接操作概率的学习算法提供了基础，也为其他算法的分析提供了理论框架

简介

- 贝叶斯学习算法与机器学习相关的两个原因：
  - 贝叶斯学习算法能够计算显示的假设概率，比如朴素贝叶斯分类
  - 贝叶斯方法为理解多数学习算法提供了一种有效的手段，而这些算法不一定直接操纵概率数据，比如
    - Find-S
    - 候选消除算法
    - 神经网络学习：选择使误差平方和最小化的神经网络
    - 推导出另一种误差函数：交叉熵
    - 分析了决策树的归纳偏置
    - 考察了最小描述长度原则

贝叶斯学习方法的特性

- 观察到的每个训练样例可以增量地降低或升高某假设的估计概率。而其他算法会在某个假设与任一样例不一致时完全去掉该假设
- 先验知识可以与观察数据一起决定假设的最终概率，先验知识的形式是：1）每个候选假设的先验概率；2）每个可能假设在可观察数据上的概率分布
- 贝叶斯方法可允许假设做出不确定性的预测
- 新的实例分类可由多个假设一起做出预测，用它们的概率来加权
- 即使在贝叶斯方法计算复杂度较高时，它们仍可作为一个最优的决策标准衡量其他方法

贝叶斯方法的难度

- 难度之一：需要概率的初始知识，当概率预先未知时，可以基于背景知识、预先准备好的数据以及基准分布的假定来估计这些概率
- 难度之二：一般情况下，确定贝叶斯最优假设的计算代价比较大（在某些特定情形下，这种计算代价可以大大降低）。

内容安排

- 介绍贝叶斯理论
- 定义极大似然假设和极大后验概率假设
- 将此概率框架应用于分析前面章节的相关问题和学习算法
- 介绍几种直接操作概率的学习算法
  - 贝叶斯最优分类器
  - Gibbs算法
  - 朴素贝叶斯分类器
- 讨论贝叶斯信念网，这是存在未知变量时被广泛使用的学习算法

贝叶斯法则

- 机器学习的任务：在给定训练数据D时，确定假设空间H中的最佳假设。
- 最佳假设：一种方法是把它定义为在给定数据D以及H中不同假设的先验概率的有关知识下的最可能假设
- 贝叶斯理论提供了一种计算假设概率的方法，基于假设的先验概率、给定假设下观察到不同数据的概率以及观察到的数据本身

先验概率和后验概率

- 用P(h)表示在没有训练数据前假设h拥有的初始概率。P(h)被称为h的先验概率。
- 先验概率反映了关于h是一正确假设的机会的背景知识
- 如果没有这一先验知识，可以简单地将每一候选假设赋予相同的先验概率
- 类似地，P(D)表示训练数据D的先验概率，P(D|h)表示假设h成立时D的概率
- 机器学习中，我们关心的是P(h|D)，即给定D时h的成立的概率，称为h的后验概率

贝叶斯公式

- 贝叶斯公式提供了从先验概率P(h)、P(D)和P(D|h)计算后验概率P(h|D)的方法

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$$

- P(h|D)随着P(h)和P(D|h)的增长而增长，随着P(D)的增长而减少，即如果D独立于h时被观察到的可能性越大，那么D对h的支持度越小

极大后验假设

- 学习器在候选假设集合H中寻找给定数据D时可能性最大的假设h，h被称为极大后验假设（MAP）
- 确定MAP的方法是用贝叶斯公式计算每个候选假设的后验概率，计算式如下

$$h_{MAP} = \arg\max_{h \in H} P(h \mid D) = \arg\max_{h \in H} \frac{P(D \mid h)P(h)}{P(D)} = \arg\max_{h \in H} P(D \mid h)P(h)$$

最后一步，去掉了P(D)，因为它是不依赖于h的常量

极大似然假设

- 在某些情况下，可假定H中每个假设有相同的先验概率，这样式子6.2可以进一步简化，只需考虑P(D|h)来寻找极大可能假设。
- P(D|h)常被称为给定h时数据D的似然度，而使P(D|h)最大的假设被称为极大似然假设

$$h_{ML} = \arg\max_{h \in H} P(D \mid h)$$

- 假设空间H可扩展为任意的互斥命题集合，只要这些命题的概率之和为1

举例：一个医疗诊断问题

- 有两个可选的假设：病人有癌症、病人无癌症
- 可用数据来自化验结果：正+和负-
- 有先验知识：在所有人口中，患病率是0.008
- 对确实有病的患者的化验准确率为98%，对确实无病的患者的化验准确率为97%
- 总结如下

  $P(cancer)=0.008, P(\neg cancer)=0.992$

  $P(+|cancer)=0.98, P(-|cancer)=0.02$

  $P(+|\neg cancer)=0.03, P(-|\neg cancer)=0.97$

举例：一个医疗诊断问题（2）

- 问题：假定有一个新病人，化验结果为正，是否应将病人断定为有癌症？求后验概率P(cancer|+)和P(¬cancer|+)
- 利用式子6.2找到极大后验假设
  - P(+|cancer)P(cancer)=0.0078
  - P(+|¬cancer)P(¬cancer)=0.0298
  - h_{MAP}=¬cancer
- 确切的后验概率可将上面的结果归一化以使它们的和为1
  - P(canner|+)=0.0078/(0.0078+0.0298)=0.21
  - P(¬cancer|-)=0.79
- 贝叶斯推理的结果很大程度上依赖于先验概率，另外不是完全接受或拒绝假设，只是在观察到较多的数据后增大或减小了假设的可能性

基本概率公式表

- 乘法规则：
  $P(A \wedge B)=P(A|B)P(B)=P(B|A)P(A)$
- 加法规则：$P(A \vee B)=P(A)+P(B)-P(A \wedge B)$
- 贝叶斯法则：$P(h|D)=P(D|h)P(h)/P(D)$
- 全概率法则：如果事件$A_1 \ldots A_n$互斥，且满足$\sum_{i=1}^{n} P(A_i)=1$，则$P(B)=\sum_{i=1}^{n} P(B \mid A_i)P(A_i)$

贝叶斯法则和概念学习

- 贝叶斯法则为计算给定训练数据下任一假设的后验概率提供了原则性方法，因此可以直接将其作为一个基本的学习方法：计算每个假设的概率，再输出其中概率最大的。这个方法称为Brute-Force贝叶斯概念学习算法。
- 将上面方法与第2章介绍的概念学习算法比较，可以看到：在特定条件下，它们学习得到相同的假设，不同的是第2章的方法不明确计算概率，而且效率更高。

Brute-Force贝叶斯概念学习

- 概念学习问题：有限假设空间H定义在实例空间X上，任务是学习某个目标概念c。
- Brute-Force MAP学习算法
  - 对于H中每个假设h，计算后验概率 $P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$
  - 输出有最高后验概率的假设 $h_{MAP} = \arg\max_{h \in H} P(h \mid D)$
- 上面算法需要较大计算量，因为它要计算每个假设的后验概率，对于大的假设空间显得不切实际，但是它提供了一个标准以判断其他概念学习算法的性能

特定情况下的MAP假设

- 假定
  - 训练数据D是无噪声的，即$d_i=c(x_i)$
  - 目标概念c包含在假设空间H中
  - 每个假设的概率相同
- 求得
  - 由于所有假设的概率之和是1，因此 $P(h)=\frac{1}{|H|}$
  - 由于训练数据无噪声，那么给定假设h时，与h一致的D的概率为1，不一致的概率为0，因此

$$P(D|h)=\begin{cases}1 & \forall d_i, d_i=h(x_i)\\ 0 & otherwise\end{cases}$$

特定情况下的MAP假设（2）

- 考虑Brute-Force MAP算法的第一步
  - h与D不一致，$P(h|D)=\frac{0\cdot P(h)}{P(D)}=0$
  - h与D一致，$P(h|D)=\frac{1\cdot\frac{1}{|H|}}{P(D)}=\frac{\frac{1}{|H|}}{\frac{|VS_{H,D}|}{|H|}}=\frac{1}{|VS_{H,D}|}$，

    $VS_{H,D}$是关于D的变型空间（见第2章，即与D一致的假设集）

特定情况下的MAP假设（3）

- P(D)的推导

$$P(D) = \sum_{h_i \in H} P(D|h_i)P(h_i)$$

$$= \sum_{h_i \in VS_{H,D}} 1 \times \frac{1}{|H|} + \sum_{h_i \in VS_{H,D}} 0 \times \frac{1}{|H|}$$

$$= \sum_{h_i \in VS_{H,D}} 1 \times \frac{1}{|H|}$$

$$= \frac{|VS_{H,D}|}{|H|}$$

- 假设的概率演化情况如图6-1所示，初始时所有假设具有相同的概率，当训练数据逐步出现后，不一致假设的概率变为0，而整个概率的和为1，它们均匀分布到剩余的一致假设中
- 每个一致的假设都是MAP假设

MAP假设和一致学习器

- 一致学习器：如果某个学习器输出的假设在训练样例上为0错误率，则称为一致学习器
- 如果H上有均匀的先验概率，且训练数据是确定性和无噪声的，任意一致学习器将输出一个MAP假设
- Find-S算法按照特殊到一般的顺序搜索架设空间H，并输出一个极大特殊的一致假设，因此可知在上面定义的P(h)和P(D|h)概率分布下，它输出MAP假设
- 更一般地，对于先验概率偏袒于更特殊假设的任何概率分布，Find-S输出的假设都是MAP假设

MAP假设和一致学习器（2）

- 贝叶斯框架提出了一种刻画学习算法行为的方法，即便该学习算法不进行概率操作，通过确定算法输出最优假设时使用的概率分布P(h)和P(D|h)，可以刻画出算法具有最优行为时的隐含假定
- 使用贝叶斯方法刻画学习算法，与揭示学习器中的归纳偏置在思想上是类似的
- 在第2章，将学习算法的归纳偏置定义为断言集合B，通过它可充分地演绎推断出学习器所执行的归纳推理结果，即学习器的输出是由其输入和隐含的归纳偏置所演绎得出的

MAP假设和一致学习器（3）

- 贝叶斯解释对于描述学习算法中的隐含假定提供了另一种方法，用基于贝叶斯理论的一个等效的概率推理系统来建模
- 贝叶斯解释隐含的假定形式为：H上的先验概率由P(h)分布给出，数据拒绝或接受假设的强度由P(D|h)给出
- 在已知这些假定的概率分布后，一个基于贝叶斯理论的概率推理系统将产生等效于Find-S、候选消除等算法的输入-输出行为

极大似然和最小误差平方假设

- 前面分析表明：某些学习算法即使没有显示地使用贝叶斯规则，或以某种形式计算概率，但它们输出的结果符合贝叶斯原理，是一个MAP假设
- 通过简单的贝叶斯分析，可以表明在特定前提下，任一学习算法如果使输出的假设预测和训练数据之间的误差平方和最小化，它将输出一极大似然假设
- 上面结论的意义是，对于许多神经网络和曲线拟合的方法，如果它们试图在训练数据上使误差平方和最小化，此结论提供了基于贝叶斯的理论依据

极大似然和最小误差平方假设（2）

- 问题框架：
  - 学习器L工作在实例空间X和假设空间H上，H中的假设为X上定义的某种实数值函数。
  - L面临的问题是学习一个从H中抽取出的未知目标函数f，给定m个训练样例的集合，每个样例的目标值被某随机噪声干扰，此随机噪声服从正态分布
  - 更精确地讲，每个训练样例是序偶$<x_i,d_i>$，$d_i=f(x_i)+e_i$，e_i是代表噪声的随机变量，假定e_i的值是独立抽取的，并且它们的分布服从0均值的正态分布
  - 学习器的任务是在所有假设有相等的先验概率前提下，输出极大似然假设（即MAP假设）

极大似然和最小误差平方假设（3）

- 用一个简单情况，即线性函数来说明问题。如图6-2所示，实线表示线性目标函数f，实点表示有噪声的训练样例集，虚线对应有最小平方训练误差的假设h_{ML}，即极大似然假设。
- 对于e这样的连续变量上的概率，使用概率密度表示概率分布，它在所有值上的积分为1，用小写的p表示。有限概率P有时又称为概率质量
- 概率密度函数：$p(x_0) = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} P(x_0 \le x < x_0 + \varepsilon)$

极大似然和最小误差平方假设（4）

- 假定有一固定的训练实例集合，因此只考虑相应的目标值序列$D=<d_1...d_m>$，这里$d_i=f(x_i)+e_i$。
- 假定训练样例是相互独立的，给定h时，可将P(D|h)写成各$p(d_i|h)$的积

$$h_{ML} = \arg\max_{h \in H} \prod_{i=1}^{m} p(d_i \mid h)$$

- 如果误差e_i服从0均值和未知方差σ^2的正态分布，那么每个d_i服从均值为$f(x_i)$，方差不变的正态分布。因此，$p(d_i|h)$可写为方差σ^2、均值$f(x_i)$的正态分布
- 使用表5-4中的正态分布公式并将相应的参数代入，由于概率d_i的表达式是在h为目标函数f的正确描述条件下的，所以替换$\mu=f(x_i)=h(x_i)$

极大似然和最小误差平方假设（5）

- h_{ML}

$$
\begin{aligned}
&= \arg\max_{h \in H} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i-\mu)^2} \\
&= \arg\max_{h \in H} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i-h(x_i))^2} \\
&= \arg\max_{h \in H} \sum_{i=1}^{m} \ln \frac{1}{\sqrt{2\pi\sigma^2}} - - \frac{1}{2\sigma^2}(d_i - h(x_i))^2 \\
&= \arg\max_{h \in H} \sum_{i=1}^{m} -\frac{1}{2\sigma^2}(d_i - h(x_i))^2 \\
&= \arg\min_{h \in H} \sum_{i=1}^{m} (d_i - h(x_i))^2
\end{aligned}
$$

- 上式说明了极大似然假设等价于使训练值和假设预测值之间的误差的平方和最小的那个假设
- 这个结论的前提是：训练值等于真实目标值加上随机噪声，其中随机噪声从一个均值为0的正态分布中独立抽取

采用正态分布的合理性

- 数学计算的简洁性
- 对许多物理系统的噪声都有良好的近似
- 第5章中心极限定律显示，足够多的独立同分布随机变量的和服从正态分布
- 由许多独立同分布的因素的和所生成的噪声将成为正态分布（当然，现实中不同的分量对噪声的贡献也许不是同分布的）
- 使误差平方最小化的方法经常被用于神经网络、曲线拟合及其他许多实函数逼近的算法中
- 上面的分析只考虑了训练样例的目标值中的噪声，而没有考虑实例属性值的噪声

用于预测概率的极大似然假设

- 问题框架：
  - 学习一个不确定性函数f: X→{0,1}，它有两个离散的值输出
  - 这种不可预测性来源于未能观察到的因素，导致目标函数的输出是输入的概率函数
- 学习得到的神经网络（或其他实函数学习器）的输出是f(x)=1的概率，表示为f': X→[0,1]，即f'=P(f(x)=1)

用于预测概率的极大似然假设（2）

- Brute-Force法
  - 首先收集对x的每个可能值观察到的1和0的频率，然后训练神经网络，对每个x输出目标频率
- 可以直接从f的训练样例中训练神经网络，然后推导出f'的极大似然假设
  - $D=\{<x_1,d_1>...<x_m,d_m>\}$
  - $P(D|h)=\prod_{i=1}^{m}P(x_i,d_i|h)=\prod_{i=1}^{m}P(d_i|h,x_i)P(x_i)$

用于预测概率的极大似然假设（3）

— $P(d_i \mid h, x_i) = \begin{cases} h(x_i) & d_i = 1 \\ 1 - h(x_i) & d_i = 0 \end{cases} = h(x_i)^{d_i}(1 - h(x_i))^{1-d_i}$

— $P(D \mid h) = \prod_{i=1}^{m} h(x_i)^{d_i}(1 - h(x_i))^{1-d_i} P(x_i)$

– $h_{ML} = \arg\max_{h \in H} \prod_{i=1}^{m} h(x_i)^{d_i}(1 - h(x_i))^{1-d_i} p(x_i)$

$$= \arg\max_{h \in H} \prod_{i=1}^{m} h(x_i)^{d_i}(1 - h(x_i))^{1-d_i}$$

$$= \arg\max_{h \in H} \sum_{i=1}^{m} d_i \ln h(x_i) + (1 - d_i) \ln(1 - h(x_i))$$

– 式子6.13与熵函数的一般式相似，因此它的负值常称为交叉熵

在神经网络中梯度搜索以达到似然最大化

- 前面讨论了利用式子6.13求极大似然假设，现用G(h,D)表示，为神经网络学习推导一个权值训练法则，使用梯度上升法使G(h,D)最大化

$$\begin{aligned}\frac{\partial G(h,D)}{\partial w_{jk}} &= \sum_{i=1}^{m} \frac{\partial G(h,D)}{\partial h(x_i)} \frac{\partial h(x_i)}{\partial w_{jk}} \\ &= \sum_{i=1}^{m} \frac{\partial (d_i \ln h(x_i) + (1-d_i)\ln(1-h(x_i)))}{\partial h(x_i)} \frac{\partial h(x_i)}{\partial w_{jk}} \\ &= \sum_{i=1}^{m} \frac{d_i - h(x_i)}{h(x_i)(1-h(x_i))} \frac{\partial h(x_i)}{\partial w_{jk}}\end{aligned}$$

- 考虑简单的情况，假定神经网络从一个单层的sigmoid单元建立，则

$$\frac{\partial h(x_i)}{\partial w_{jk}} = \sigma'(x_i) x_{ijk} = h(x_i)(1-h(x_i)) x_{ijk}$$

在神经网络中梯度搜索以达到似然最大化（2）

$$\frac{\partial G(h,D)}{\partial w_{jk}} = \sum_{i=1}^{m} (d_i - h(x_i)) x_{ijk}$$

- 因为要使P(Dlh)最大化而不是最小化，因此执行梯度上升搜索，而不是梯度下降搜索。

$$w_{jk} \leftarrow w_{jk} + \Delta w_{jk} \qquad \Delta w_{jk} = \eta \sum_{i=1}^{m} (d_i - h(x_i)) x_{ijk}$$

- 与反向传播更新法则对比
  - 使误差平方最小化的法则寻找到极大似然假设的前提是：训练数据可以由目标函数值加上正态分布噪声来模拟
  - 使交叉熵最小化的法则寻找极大似然假设基于的前提是：观察到的布尔值为输入实例的概率函数

最小描述长度准则

- 奥坎姆剃刀可以概括为：为观察到的数据选择最短的解释
- 此处给出一个贝叶斯分析，提出最小描述长度准则，根据信息论中的基本概念来解释h_{MAP}的定义

$$\begin{aligned} h_{MAP} &= \arg\max_{h \in H} P(D|h)P(h) \\ &= \arg\max_{h \in H} \log_2 P(D|h) + \log_2 P(h) \\ &= \arg\min_{h \in H} -\log_2 P(D|h) - \log_2 P(h) \end{aligned}$$

- 上式可以解释为在特定的假设编码表示方案上“优先选择短的假设”

最小描述长度准则（2）

- 信息论中的编码理论
  - 设想要为随机传送的消息设计一个编码，其中遇到消息i的概率是p_i
  - 感兴趣的是，使得传输随机信息所需的最小期望传送位数的编码
  - 直观上，为使期望的编码长度最小，可能性大的消息应该赋予较短的编码
  - Shannon & Weaver证明了最优编码对消息i的编码长度为$-\log_2 p_i$
  - 使用代码C来编码消息i所需的位数被称为消息i关于C的描述长度，记为$L_C(i)$

最小描述长度准则（3）

- 使用编码理论的结论来解释等式6.16
  - $-\log_2 P(h)$是在假设空间H的最优编码下h的描述长度。换言之，这是假设h使用其最优表示时的大小
  - $L_{C_H}(h) = -\log_2 P(h)$，$C_H$为假设空间H的最优编码
  - $-\log_2 P(D|h)$是在给定假设h时，训练数据D的描述长度，$L_{C_{D|h}}(D|h) = -\log_2 P(D|h)$，$C_{D|h}$是假定发送者和接送者都知道假设h时描述数据D的最优编码
  - 因此式子6.16显示，h_{MAP}是使假设描述长度和给定假设下数据描述长度之和最小化的假设
- 最小描述长度准则：$h_{MDL} = \arg\min_{h \in H} L_{C_1}(h) + L_{C_2}(D|h)$

最小描述长度准则（4）

- 如果选择C_1为假设的最优编码C_H，C_2为最优编码$C_{D|h}$，那么$h_{MDL}=h_{MAP}$
- 可将MDL准则想象为选择最短的方法来重新编码训练数据，其中不仅计算假设的大小，并且计算给定假设时编码数据的附加开销
- 将MDL准则应用于决策树，如何选择假设和数据的表示C_1和C_2?
  - 对于C_1，很自然地选择某种明确的决策树编码方法，其中描述长度随着树中节点和边的增长而增加
  - 对于C_2，如果训练分类$f(x_i)$与假设的预计相同，那么就不需要传输有关这些样例的任何信息；如果不同，则要传输更正消息

最小描述长度准则（5）

- MDL准则提供了一种方法在假设的复杂性和假设产生错误的数量之间进行折中，它有可能选择一个较短的产生少量错误的假设，而不是完美地分类训练数据的较长的假设
- 上面讨论自然给出了一种处理数据过度拟合的方法
- Quinlan & Rivest描述了应用MDL准则选择决策树大小的几个实验，报告指出，基于MDL的方法产生的决策树的精度相当于第3章中讨论的标准树修剪方法
- 第125页，6.6节最后一段的含义？

贝叶斯最优分类器

- 前面我们讨论的问题是：给定训练数据，最可能的假设是什么？
- 另一个相关的更有意义的问题是：给定训练数据，对新实例的最可能的分类是什么？
- 显然，第二个问题的解决可以将第一个问题的结果（MAP）应用到新实例上得到，还存在更好的算法

贝叶斯最优分类器（2）

- 例子
  - 考虑一个包含三个假设h_1, h_2, h_3的假设空间。
  - 假定已知训练数据时三个假设的后验概率分别是0.4, 0.3, 0.3，因此h_1为MAP假设。
  - 若一新实例x被h_1分类为正，被h_2和h_3分类为反
  - 计算所有假设，x为正例的概率为0.4，为反例的概率为0.6
  - 因此，这时最可能的分类与MAP假设生成的分类不同

举例：学习分类文本（2）

- 应用朴素贝叶斯分类器的两个主要设计问题：
  - 怎样将任意文档表示为属性值的形式
  - 如何估计朴素贝叶斯分类器所需的概率
- 表示文档的方法
  - 给定一个文本文档，对每个单词的位置定义一个属性，该属性的值为在此位置上找到的英文单词
- 假定我们共有1000个训练文档，其中700个分类为dislike，300个分类为like，现在要对下面的新文档进行分类：
  - This is an example document for the naive Bayes classifier. This document contains only one paragraph, or two sentences.

举例：学习分类文本（3）

- 计算式 $v_{NB} = \arg\max_{v_j \in \{like, dislike\}} P(v_j) \prod_{i=1}^{19} P(a_i \mid v_j)$

$$= \arg\max_{v_j \in \{like, dislike\}} P(v_j) P(a_1 = "this" \mid v_j) ... P(a_{19} = "sentences" \mid v_j)$$

- 注意此处贝叶斯分类器隐含的独立性假设并不成立。通常，某个位置上出现某个单词的概率与前后位置上出现的单词是相关的
- 虽然此处独立性假设不精确，但别无选择，否则要计算的概率项极为庞大。
- 另外实践中，朴素贝叶斯学习器在许多文本分类问题中性能非常好

举例：学习分类文本（4）

- 需要估计概率项$P(v_i)$和$P(a_i=w_k|v_i)$。前一项可基于每一类在训练数据中的比例很容易得到，后一项含三个参数，出现数据稀疏问题
- 再引入一个假定以减少需要估计的概率项的数量：假定单词w_k出现的概率独立于单词所在的位置，即$P(a_i=w_k|v_i)=P(w_k|v_j)$
- 作此假定的一个主要优点在于：使可用于估计每个所需概率的样例数增加了，因此增加了估计的可靠程度
- 采纳m-估计方法，即有统一的先验概率并且m等于词汇表的大小，因此

$$P(w_k|v_j)=\frac{n_k+1}{n+|Vocabulary|}$$

表6-2 用于学习和分类文本的朴素贝叶斯算法

- Learn_Naive_Bayes_Text(Examples, V)

  Examples为一组文本文档以及它们的目标值。V为所有可能目标值的集合。此函数作用是学习概率项$P(w_k|v_j)$和$P(v_j)$。
  - 收集Examples中所有的单词、标点符号以及其他记号
    - Vocabulary←在Examples中任意文本文档中出现的所有单词及记号的集合
  - 计算所需要的概率项$P(v_j)$和$P(w_k|v_j)$
    - 对V中每个目标值v_j
      - $docs_j$←Examples中目标值为v_j的文档子集
      - $P(v_j)$←$|docs_j| / |Examples|$
      - $Text_j$←将$docs_j$中所有成员连接起来建立的单个文档
      - n←在$Text_j$中不同单词位置的总数
      - 对Vocabulary中每个单词w_k
        - » n_k←单词w_k出现在$Text_j$中的次数
        - » $P(w_k|v_j)$←$(n_k+1) / (n+|Vocabulary|)$

表6-2 用于学习和分类文本的朴素贝叶斯算法（2）

- Classify_Naive_Bayes_Text(Doc)

  对文档Doc返回其估计的目标值，a_i代表在Doc中的第i个位置上出现的单词

  – positions←在Doc中的所有单词位置，它包含能在Vocabulary中找到的记号

  – 返回v_{NB}，$v_{NB} = \arg\max_{v_j \in V} P(v_j) \prod_{i \in positions} P(a_i \mid v_j)$

实验结果

- Joachims将此算法用于新闻组文章的分类
  - 每一篇文章的分类是该文章所属的新闻组名称
  - 20个新闻组，每个新闻组有1000篇文章，共2万个文档
  - 2/3作为训练样例，1/3进行性能测量
  - 词汇表不包含最常用词（比如the、of）和罕见词（数据集中出现次数少于3）
- Lang用此算法学习目标概念“我感兴趣的新闻组文章”
  - NewsWeeder系统，让用户阅读新闻组文章并为其评分，然后使用这些评分的文章作为训练样例，来预测后续文章哪些是用户感兴趣的
  - 每天向用户展示前10%的自动评分文章，它建立的文章序列中包含的用户感兴趣的文章比通常高3~4倍

贝叶斯信念网

- 朴素贝叶斯分类器假定各个属性取值在给定目标值v下是条件独立的，从而化简了最优贝叶斯分类的计算复杂度。但在多数情况下，这一条件独立假定过于严厉了。
- 贝叶斯信念网描述的是一组变量所遵从的概率分布，它通过一组条件概率来指定一组条件独立性假设
- 贝叶斯信念网中可表述变量的一个子集上的条件独立性假定，因此，贝叶斯信念网提供了一种中间的方法，它比朴素贝叶斯分类器的限制更少，又比在所有变量中计算条件依赖更可行

贝叶斯信念网（2）

- 贝叶斯信念网描述了一组变量上的概率分布
- 考虑一任意的随机变量集合$Y_1 ... Y_n$，其中每个Y_i可取的值集合为$V(Y_i)$
- 变量集合Y的联合空间为叉乘$V(Y_1) \times ... \times V(Y_n)$
- 在此联合空间上的概率分布称为联合概率分布，联合概率分布指定了元组的每个可能的变量约束的概率
- 贝叶斯信念网则对一组变量描述了联合概率分布

条件独立性

- 精确定义条件独立性
  - 令X, Y和Z为3个离散值随机变量，当给定Z值时X服从的概率分布独立于Y的值，称X在给定Z时条件独立于Y，即

$$(\forall x_i, y_j, z_k) P(X = x_i \mid Y = y_j, Z = z_k) = P(X = x_i \mid Z = z_k)$$

  - 上式通常简写成P(X|Y,Z)=P(X|Z)
- 扩展到变量集合
  - 下面等式成立时，称变量集合$X_1...X_l$在给定变量集合$Z_1...Z_n$时条件独立于变量集合$Y_1...Y_m$

$$P(X_1...X_l \mid Y_1...Y_m, Z_1...Z_n) = P(X_1...X_l \mid Z_1...Z_n)$$

- 条件独立性与朴素贝叶斯分类器的之间的关系

$$\begin{aligned} P(A_1, A_2 \mid V) &= P(A_1 \mid A_2, V) P(A_2 \mid V) \\ &= P(A_1 \mid V) P(A_2 \mid V) \end{aligned}$$

贝叶斯信念网的表示

- 贝叶斯信念网（简称贝叶斯网）表示一组变量的联合概率分布
- 一般地说，贝叶斯网表示联合概率分布的方法是指定一组条件独立性假定（有向无环图）以及一组局部条件概率集合
- 图6-3，联合空间中每个变量在贝叶斯网中表示为一个节点，每个变量需要两种类型的信息
  - 网络弧表示断言“此变量在给定其直接前驱时条件独立于其非后继”
  - 每个变量有一个条件概率表，描述了该变量在给定其立即前驱时的概率分布

贝叶斯信念网的表示（2）

- 对网络变量的元组<$Y_1 \ldots Y_n$>赋以所希望的值($y_1 \ldots y_n$)的联合概率计算公式如下：

$$P(y_1,\ldots,y_n)=\prod_{i=1}^{n}P(y_i \mid Parents(Y_i))$$

$$P(Campfire=True \mid Storm=True, BusTourGroup=True)=0.4$$

- 所有变量的局部条件概率表以及由网络所描述的一组条件独立假定，描述了该网络的整个联合概率分布

贝叶斯信念网的推理

- 可以用贝叶斯网在给定其他变量的观察值时推理出某些目标变量的值
- 由于所处理的是随机变量，所以一般不会赋予目标变量一个确切的值
- 真正需要推理的是目标变量的概率分布，它指定了在给予其他变量的观察值条件下，目标变量取每一个可能值的概率
- 在网络中所有其他变量都确切知道的情况下，这一推理步骤很简单
- 一般来说，贝叶斯网络可用于在知道某些变量的值或分布时计算网络中另一部分变量的概率分布

贝叶斯信念网的推理（2）

- 对任意贝叶斯网络的概率的确切推理已经知道是一个NP难题
- Monte Carlo方法提供了一种近似的结果，通过对未观察到的变量进行随机采样
- 理论上，即使是贝叶斯网络中的近似推理也可能是NP难题
- 实践中许多情况下近似的方法被证明是有效的

学习贝叶斯信念网

- 从训练数据中学到贝叶斯信念网，有多种讨论的框架：
  - 网络结构可以预先给出，或由训练数据中得到
  - 所有的网络变量可以直接从每个训练样例中观察到，或某些变量不能观察到
- 如果网络结构已知且变量可以从训练样例中完全获得，那么得到条件概率表就比较简单
- 如果网络结构已知，但只有一部分变量值能在数据中观察到，学习问题就困难多了。这类似于在人工神经网络中学习隐藏单元的权值
- Russtll（1995）提出了一个简单的梯度上升过程以学习条件概率表中的项，相当于对表项搜索极大似然假设

贝叶斯网的梯度上升训练

- 令w_{ijk}代表条件概率表的一个表项，即在给定父节点U_i取值u_{ik}时，网络变量Y_i值为y_{ij}的概率
- 例如图6-3，w_{ijk}为最右上方的表项，那么Y_i为变量Campfire，U_i是其父节点的元组<Storm, BusTourGroup>，y_{ij}=True，且u_{ik}=<False, False>

贝叶斯网的梯度上升训练（2）

- lnP(D|h)的梯度由对每个w_{ijk}求导数得到

$$\frac{\partial \ln P(D \mid h)}{\partial w_{ijk}} = \sum_{d \in D} \frac{P(Y_i = y_{ij}, U_i = u_{ik} \mid d)}{w_{ijk}}$$

- 例如，为计算图6-3中表左上方的表项的lnP(D|h)的导数，需要对D中每个训练样例d计算P(Campfire=True, Storm=False, BusTourGroup=False|d)
- 当训练样例中无法观察到这些变量时，这些概率可用标准的贝叶斯网从d中观察到的变量中推理得到
- 这些量能够很容易地从贝叶斯网推理过程中得到，几乎不需要附加的开销

贝叶斯网的梯度上升训练（3）

- 式子6.25的推导
  - 用$P_h(D)$来表示$P(D|h)$
  - 假定在数据集D中的各样例d都是独立抽取的

$$
\begin{aligned}
\frac{\partial \ln P_h(D)}{\partial w_{ijk}} &= \frac{\partial}{\partial w_{ijk}} \ln \prod_{d \in D} P_h(d) \\
&= \sum_{d \in D} \frac{\partial \ln P_h(d)}{\partial w_{ijk}} \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{\partial P_h(d)}{\partial w_{ijk}} \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{\partial}{\partial w_{ijk}} \sum_{j',k'} P_h(d \mid y_{ij'}, u_{ik'}) P_h(y_{ij'}, u_{ik'}) \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{\partial}{\partial w_{ijk}} \sum_{j',k'} P_h(d \mid y_{ij'}, u_{ik'}) P_h(y_{ij'} \mid u_{ik'}) P_h(u_{ik'}) \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{\partial}{\partial w_{ijk}} P_h(d \mid y_{ij}, u_{ik}) P_h(y_{ij} \mid u_{ik}) P_h(u_{ik}) \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{\partial}{\partial w_{ijk}} P_h(d \mid y_{ij}, u_{ik}) w_{ijk} P_h(u_{ik}) \\
&= \sum_{d \in D} \frac{1}{P_h(d)} P_h(d \mid y_{ij}, u_{ik}) P_h(u_{ik}) \\
&= \sum_{d \in D} \frac{1}{P_h(d)} \frac{P_h(y_{ij}, u_{ik} \mid d) P_h(d) P_h(u_{ik})}{P_h(y_{ij}, u_{ik})} \\
&= \sum_{d \in D} \frac{P_h(y_{ij}, u_{ik} \mid d) P_h(u_{ik})}{P_h(y_{ij}, u_{ik})} \\
&= \sum_{d \in D} \frac{P_h(y_{ij}, u_{ik} \mid d)}{P_h(y_{ij} \mid u_{ik})} \\
&= \sum_{d \in D} \frac{P_h(y_{ij}, u_{ik} \mid d)}{w_{ijk}}
\end{aligned}
$$

贝叶斯网的梯度上升训练（4）

- 更新权值

$$w_{ijk} \leftarrow w_{ijk} + \eta \sum_{d \in D} \frac{P_h(y_{ij}, u_{ik} \mid d)}{w_{ijk}}$$

- 归一化处理，保持在区间[0,1]之间，且$\sum_j w_{ijk}$对所有i,k保持为1

$$w_{ijk} \leftarrow \frac{w_{ijk}}{\sum_{i,k} w_{ijk}}$$

- 这个算法只保证找到局部最优解，替代梯度上升的一个算法是EM算法

学习贝叶斯网的结构

- 如果贝叶斯网的结构未知，那么需要学习贝叶斯网的结构
- Cooper & Herskovits提出了一个贝叶斯评分尺度，以便从不同网络中进行选择
- Cooper & Herskovits提出了算法K2，启发式算法，用于在数据完全可观察时学习网络结构
- 基于约束的学习贝叶斯网络结构：从数据中推导出独立和相关的关系，然后用这些关系来构造贝叶斯网

K均值算法的推导（2）

– 所有实例的概率的对数

$$\begin{aligned}\ln P(Y \mid h') &= \ln \prod_{i=1}^{m} p(y_i \mid h') \\ &= \sum_{i=1}^{m} \ln p(y_i \mid h') \\ &= \sum_{i=1}^{m} \left(\ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2} \sum_{j=1}^{k} z_{ij} (x_i - \mu_j')^2 \right)\end{aligned}$$

– 计算期望值

$$\begin{aligned}E[\ln P(Y \mid h')] &= E\left[\sum_{i=1}^{m} \left(\ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2} \sum_{j=1}^{k} z_{ij} (x_i - \mu_j')^2 \right) \right] \\ &= \sum_{i=1}^{m} \left(\ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2} \sum_{j=1}^{k} E[z_{ij}] (x_i - \mu_j')^2 \right)\end{aligned}$$

K均值算法的推导（3）

– 求使Q函数最大的假设

$$\arg\max_{h'} Q(h'|h) = \arg\max_{h'} \sum_{i=1}^{m}\left(\ln\frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}\sum_{j=1}^{k} E[z_{ij}](x_i-\mu_j')^2\right)$$

$$= \arg\max_{h'} \sum_{i=1}^{m}\sum_{j=1}^{k} E[z_{ij}](x_i-\mu_j')^2$$

– 解上式得到

$$\mu_j \leftarrow \frac{\sum_{i=1}^{m} E[z_{ij}]x_i}{\sum_{i=1}^{m} E[z_{ij}]}$$

– 另外

$$E[zij] \leftarrow \frac{e^{-\frac{1}{2\sigma^2}(x_i-\mu_j)2}}{\sum_{n=1}^{k} e^{-\frac{1}{2\sigma^2}(x_i-\mu_j)^2}}$$

小结

- 概率学习方法利用关于不同假设的先验概率，以及在给定假设时观察到不同数据的概率的知识
- 贝叶斯方法提供了概率学习方法的基础，基于这些先验和数据观察假定，赋予每个假设一个后验概率
- 贝叶斯方法确定的极大后验概率假设是最可能成为最优假设的假设
- 贝叶斯最优分类器将所有假设的预测结合起来，并用后验概率加权，以计算对新实例的最可能分类
- 朴素贝叶斯分类器增加了简化假定：属性值在给定实例的分类时条件独立
- 贝叶斯信念网能够表示属性的子集上的一组条件独立性假定

小结（2）

- 贝叶斯推理框架可对其他不直接应用贝叶斯公式的学习方法的分析提供理论基础
- 最小描述长度准则建议选取这样的假设，它使假设的描述长度和给定假设下数据的描述长度的和最小化。贝叶斯公式和信息论中的基本结论提供了此准则的根据
- EM算法提供了一个通用的算法，在存在隐藏变量时进行学习。算法开始于一个任意的初始假设，然后迭代地计算隐藏变量的期望值，再重新计算极大似然假设，这个过程收敛到一个局部极大似然假设和隐藏变量的估计值

补充读物

- Casella & Berger 1990在概率和统计方面的介绍性文章
- Maisel 1971, Speigel 1991的快速参考书籍
- Duda & Hart 1973对贝叶斯分类器和最小平方误差分类器的介绍
- Domigos & Pazzani 1996分析了朴素贝叶斯分类器输出最优分类的条件
- Cestnik 1990讨论了m-估计
- Michie et al. 1994将不同贝叶斯方法与决策树等其他算法进行比较
- Chauvin & Rumelhart1995提供了基于反向传播算法的神经网络的贝叶斯分析
- Rissanen1 1983, 1989讨论了最小描述长度准则
- Quinlan & Rivest 1989描述了利用最小描述长度准则避免决策树过度拟合的方法