朴素贝叶斯分类器

维基百科，自由的百科全书

在机器学习中，**單純貝氏分类器**是一系列以假设特征之间强（朴素）独立下运用贝叶斯定理为基础的简单概率分类器。

單純貝氏自20世纪50年代已广泛研究。在20世纪60年代初就以另外一个名称引入到文本信息检索界中，[1]:488 并仍然是文本分类的一种热门（基准）方法，文本分类是以词频为特征判断文件所属类别或其他（如垃圾邮件、合法性、体育或政治等等）的问题。通过适当的预处理，它可以与这个领域更先进的方法（包括支持向量机）相竞争。[2] 它在自动医疗诊断中也有应用。[3]

單純貝氏分类器是高度可扩展的，因此需要数量与学习问题中的变量（特征/预测器）成线性关系的参数。最大似然训练可以通过评估一个封闭形式的表达式来完成，[1]:718 只需花费线性时间，而不需要其他很多类型的分类器所使用的费时的迭代逼近。

在统计学和计算机科学文献中，單純貝氏模型有各种名称，包括**简单贝叶斯**和**独立贝叶斯**。[4] 所有这些名称都参考了贝叶斯定理在该分类器的决策规则中的使用，但單純貝氏不（一定）用到贝叶斯方法；[4] 《Russell和Norvig》提到“『單純貝氏』有时被称为**贝叶斯分类器**，这个马虎的使用促使真正的贝叶斯论者称之为**傻瓜贝叶斯**模型。”[1]:482

目录

简介

單純貝氏是一种构建分类器的简单方法。该分类器模型会给问题实例分配用特征值表示的类标签，类标签取自有限集合。它不是训练这种分类器的单一算法，而是一系列基于相同原理的算法：所有單純貝氏分类器都假定样本每个特征与其他特征都不相关。举个例子，如果一种水果其具有红，圆，直径大概3英寸等特征，该水果可以被判定为是苹果。尽管这些特征相互依赖或者有些特征由其他特征决定，然而單純貝氏分类器认为这些属性在判定该水果是否为苹果的概率分布上独立的。

对于某些类型的概率模型，在监督式学习的样本集中能获取得非常好的分类效果。在许多实际应用中，單純貝氏模型参数估计使用最大似然估计方法；换而言之，在不用到贝叶斯概率或者任何贝叶斯模型的情况下，單純貝氏模型也能奏效。

尽管是带着这些朴素思想和过于简单化的假设，但單純貝氏分类器在很多复杂的现实情形中仍能够取得相当好的效果。2004年，一篇分析贝叶斯分类器问题的文章揭示了單純貝氏分类器取得看上去不可思议的分类效果的若干理论上的原因。[5] 尽管如此，2006年有一篇文章详细比较了各种分类方法，发现更新的方法（如决策树和随机森林）的性能超过了贝叶斯分类器。[6]

單純貝氏分类器的一个优势在于只需要根据少量的训练数据估计出必要的参数（变量的均值和方差）。由于变量独立假设，只需要估计各个变量的方法，而不需要确定整个协方差矩阵。

單純貝氏概率模型

理论上，概率模型分类器是一个条件概率模型。

$$p(C|F_1,\dots,F_n)$$

独立的类别变量C有若干类别，条件依赖于若干特征变量F_1,F_2,...,F_n。但问题在于如果特征数量n较大或者每个特征能取大量值时，基于概率模型列出概率表变得不现实。所以我们修改这个模型使之变得可行。贝叶斯定理有以下式子：

$$p(C|F_1,\dots,F_n) = \frac{p(C)\,p(F_1,\dots,F_n|C)}{p(F_1,\dots,F_n)}.$$

用朴素的语言可以表达为：

$$\mathrm{posterior} = \frac{\mathrm{prior} \times \mathrm{likelihood}}{\mathrm{evidence}}.$$

实际中，我们只关心分式中的分子部分，因为分母不依赖于C而且特征F_i的值是给定的，于是分母可以认为是一个常数。这样分子就等价于联合分布模型。

$$p(C, F_1, \dots, F_n)$$

重复使用链式法则，可将该式写成条件概率的形式，如下所示：

$$\begin{aligned}
&p(C, F_1, \dots, F_n) \\
&\quad \propto p(C)\, p(F_1, \dots, F_n|C) \\
&\quad \propto p(C)\, p(F_1|C)\, p(F_2, \dots, F_n|C, F_1) \\
&\quad \propto p(C)\, p(F_1|C)\, p(F_2|C, F_1)\, p(F_3, \dots, F_n|C, F_1, F_2) \\
&\quad \propto p(C)\, p(F_1|C)\, p(F_2|C, F_1)\, p(F_3|C, F_1, F_2)\, p(F_4, \dots, F_n|C, F_1, F_2, F_3) \\
&\quad \propto p(C)\, p(F_1|C)\, p(F_2|C, F_1)\, p(F_3|C, F_1, F_2)\, \dots p(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}).
\end{aligned}$$

现在“朴素”的条件独立假设开始发挥作用假设每个特征F_i对于其他特征$F_j, j \neq i$是条件独立的。这就意味着

$$p(F_i|C, F_j) = p(F_i|C)$$

对于$i \neq j$，所以联合分布模型可以表达为

$$\begin{aligned}
p(C|F_1, \dots, F_n) &\propto p(C, F_1, \dots, F_n) \\
&\propto p(C)\, p(F_1|C)\, p(F_2|C)\, p(F_3|C)\, \cdots \\
&\propto p(C) \prod_{i=1}^{n} p(F_i|C).
\end{aligned}$$

这意味着上述假设下，类变量C的条件分布可以表达为：

$$p(C|F_1, \dots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^{n} p(F_i|C)$$

其中Z(证据因子)是一个只依赖与$F_1, \dots, F_n$等的缩放因子，当特征变量的值已知时是一个常数。 由于分解成所谓的类先验概率$p(C)$和独立概率分布$p(F_i|C)$，上述概率模型的可掌控性得到很大的提高。如果这是一个k分类问题，且每个$p(F_i|C = c)$可以表达为r个参数，于是相应的單純貝氏模型有$(k-1) + n\,r\,k$个参数。实际应用中，通常取$k = 2$（二分类问题），$r = 1$ (伯努利分布作为特征），因此模型的参数个数为$2n + 1$，其中n是二值分类特征的个数。

从概率模型中构造分类器

讨论至此为止我们导出了独立分布特征模型，也就是單純貝氏概率模型。單純貝氏分类器包括了这种模型和相应的决策规则。一个普通的规则就是选出最有可能的那个：这就是大家熟知的最大后验概率（MAP）决策准则。相应的分类器便是如下定义的$\mathbf{classify}$公式：

$$\mathbf{classify}(f_1, \dots, f_n) = \operatorname*{\mathbf{argmax}}_c\, p(C = c) \prod_{i=1}^{n} p(F_i = f_i|C = c).$$

参数估计

所有的模型参数都可以通过训练集的相关频率来估计。常用方法是概率的最大似然估计。类的先验概率可以通过假设各类等概率来计算（先验概率 1 / (类的数量)），或者通过训练集的各类样本出现的次数来估计（A类先验概率=（A类样本的数量）/(样本总数)）。为了估计特征的分布参数，我们要先假设训练集数据满足某种分布或者非参数模型。[7]

高斯單純貝氏

如果要处理的是连续数据一种通常的假设是这些连续数值为高斯分布。 例如，假设训练集中有一个连续属性，x。我们首先对数据根据类别分类，然后计算每个类别中x的均值和方差。令μ_c 表示为x在c类上的均值，令σ_c^2为 x在c类上的方差。在给定类中某个值的概率，$P(x = v|c)$，可以通过将v表示为均值为μ_c方差为σ_c^2正态分布计算出来。如下，$P(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}}\, e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}$ 处理连续数值问题的另一种常用的技术是通过离散化连续数值的方法。通常，当训练样本数量较少或者是精确的分布已知时，通过概率分布的方法是一种更好的选择。在大量样本的情形下离散化的方法表现更优，因为大量的样本可以学习到数据的分布。由于單純貝氏是一种典型的用到大量样本的方法（越大计算量的模型可以产生越高的分类精确度），所以單純貝氏方法都用到离散化方法，而不是概率分布估计的方法。

样本修正

如果一个给定的类和特征值在训练集中没有一起出现过，那么基于频率的估计下该概率将为0。这将是一个问题。因为与其他概率相乘时将会把其他概率的信息统统去除。所以常常要求要对每个小类样本的概率估计进行修正，以保证不会出现有为0的概率出现。

讨论

尽管实际上独立假设常常是不准确的，但單純貝氏分类器的若干特性让其在实践中能够取得令人惊奇的效果。特别地，各类条件特征之间的解耦意味着每个特征的分布都可以独立地被当做一维分布来估计。这样减轻了由于维数灾带来的阻碍,当样本的特征个数增加时就不需要使样本规模呈指数增长。然而單純貝氏在大多数情况下不能对类概率做出非常准确的估计，但在许多应用中这一点并不要求。例如，單純貝氏分类器中，依据最大后验概率决策规则只要正确类的后验概率比其他类要高就可以得到正确的分类。所以不管概率估计轻度的甚至是严重的不精确都不影响正确的分类结果。在这种方式下，分类器可以有足够的鲁棒性去忽略單純貝氏概率模型上存在的缺陷。

实例

性别分类

问题描述:通过一些测量的特征，包括身高、体重、脚的尺寸，判定一个人是男性还是女性。

训练

训练数据如下：

性别	身高(英尺)	体重(磅)	脚的尺寸(英寸)
男	6	180	12
男	5.92 (5'11")	190	11
男	5.58 (5'7")	170	12
男	5.92 (5'11")	165	10
女	5	100	6
女	5.5 (5'6")	150	8
女	5.42 (5'5")	130	7
女	5.75 (5'9")	150	9

假设训练集样本的特征满足高斯分布，得到下表：

性别	均值(身高)	方差(身高)	均值(体重)	方差(体重)	均值(脚的尺寸)	方差(脚的尺寸)
男性	5.855	3.5033e-02	176.25	1.2292e+02	11.25	9.1667e-01
女性	5.4175	9.7225e-02	132.5	5.5833e+02	7.5	1.6667e+00

我们认为两种类别是等概率的，也就是P(male)= P(female) = 0.5。在没有做辨识的情况下就做这样的假设并不是一个好的点子。但我们通过数据集中两类样本出现的频率来确定P(C)，我们得到的结果也是一样的。

测试

以下给出一个待分类是男性还是女性的样本。

性别	身高(英尺)	体重(磅)	脚的尺寸(英尺)
sample	6	130	8

我们希望得到的是男性还是女性哪类的后验概率大。男性的后验概率通过下面式子来求取

$$posterior(male) = \frac{P(male)\,p(height|male)\,p(weight|male)\,p(footsize|male)}{evidence}$$

女性的后验概率通过下面式子来求取

$$posterior(female) = \frac{P(female)\,p(height|female)\,p(weight|female)\,p(footsize|female)}{evidence}$$

证据因子（通常是常数）用来对各类的后验概率之和进行归一化

$$evidence = P(male)\,p(height|male)\,p(weight|male)\,p(footsize|male) + P(female)\,p(height|female)\,p(weight|female)\,p(footsize|female)$$

证据因子是一个常数（在正态分布中通常是正数），所以可以忽略。接下来我们来判定这样样本的性别。

$$P(male) = 0.5$$

$p(\mathrm{height}|\mathrm{male}) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{-(6-\mu)^2}{2\sigma^2}\right) \approx 1.5789$,其中$\mu = 5.855$，$\sigma^2 = 3.5033e^{-02}$是训练集样本的正态分布参数. 注意，这里的值大于1也是允许的 – 这里是概率密度而不是概率，因为身高是一个连续的变量

$$p(weight|male) = 5.9881e^{-06}$$

$$p(footsize|male) = 1.3112e^{-3}$$

$$posteriornumerator(male) = 6.1984e^{-09}$$

$$P(female) = 0.5$$

$$p(height|female) = 2.2346e^{-1}$$

$$p(weight|female) = 1.6789e^{-2}$$

$$p(footsize|female) = 2.8669e^{-1}$$

$$posteriornumerator(female) = 5.3778e^{-04}$$

由于女性后验概率的分子比较大，所以我们预计这个样本是女性。

文本分类

这是一个用單純貝氏分类做的一个文本分类问题的例子。考虑一个基于内容的文本分类问题，例如判断邮件是否为垃圾邮件。想像文本可以分成若干的类别，首先文本可以被一些单词集标注，而这个单词集是独立分布的，在给定的C类文本中第i个单词出现的概率可以表示为：

$$p(w_i|C)$$

（通过这种处理，我们进一步简化了工作，假设每个单词是在文中是随机分布的，也就是单词不依赖于文本的长度，与其他词出现在文中的位置，或者其他文本内容。）

所以，对于一个给定类别C，文本D包含所有单词w_i的概率是:

$$p(D|C) = \prod_i p(w_i|C)$$

我们要回答的问题是「文档D属于类C的概率是多少？」换而言之$p(C|D)$ 是多少？ 现在定义

$$p(D|C) = \frac{p(D \cap C)}{p(C)}$$

$$p(C|D) = \frac{p(D \cap C)}{p(D)}$$

通过贝叶斯定理将上述概率处理成似然度的形式

$$p(C|D) = \frac{p(C)}{p(D)} p(D|C)$$

假设现在只有两个相互独立的类别，S和$\neg S$（垃圾邮件和非垃圾邮件），这里每个元素（邮件）要么是垃圾邮件，要么就不是。

$$p(D|S) = \prod_i p(w_i|S)$$

$$p(D|\neg S) = \prod_i p(w_i|\neg S)$$

用上述贝叶斯的结果，可以写成

$$p(S|D) = \frac{p(S)}{p(D)} \prod_i p(w_i|S)$$

$$p(\neg S|D) = \frac{p(\neg S)}{p(D)} \prod_i p(w_i|\neg S)$$

两者相除:

$$\frac{p(S|D)}{p(\neg S|D)} = \frac{p(S) \prod_i p(w_i|S)}{p(\neg S) \prod_i p(w_i|\neg S)}$$

整理得:

$$\frac{p(S|D)}{p(\neg S|D)} = \frac{p(S)}{p(\neg S)} \prod_i \frac{p(w_i|S)}{p(w_i|\neg S)}$$

这样概率比p(S | D) / p(¬S | D)可以表达为似然比。实际的概率p(S | D)可以很容易通过log (p(S | D) / p(¬S | D))计算出来，基于p(S | D) + p(¬S | D) = 1。

结合上面所讨论的概率比，可以得到：

$$\ln \frac{p(S|D)}{p(\neg S|D)} = \ln \frac{p(S)}{p(\neg S)} + \sum_i \ln \frac{p(w_i|S)}{p(w_i|\neg S)}$$

(这种对数似然比的技术在统计中是一种常用的技术。在这种两个独立的分类情况下（如这个垃圾邮件的例子），把对数似然比转化S曲线的形式)。

最后文本可以分类，当$p(S|D) > p(\neg S|D)$或者$\ln \frac{p(S|D)}{p(\neg S|D)} > 0$时判定为垃圾邮件，否则为正常邮件。

参见

- AODE
- 貝葉斯垃圾郵件過濾
- 贝叶斯网络
- 随机森林
- 线性分类器
- 提升方法
- 模糊逻辑
- 邏輯迴歸
- Class membership probabilities
- 神经网络
- 预测分析
- 感知机

- 支持向量机
- 贝叶斯定理
- 有监督学习
- 分类器
- 最大似然估计
- 贝叶斯概率
- boosted trees
- 随机森林

参考文献

1. Russell, Stuart; Norvig, Peter. Artificial Intelligence: A Modern Approach (2nd. Prentice Hall. 2003 [1995]. ISBN 978-0137903955.
2. Rennie, J.; Shih, L.; Teevan, J.; Karger, D. Tackling the poor assumptions of Naive Bayes classifiers (PDF). ICML. 2003.
3. Rish, Irina. An empirical study of the naive Bayes classifier (PDF). IJCAI Workshop on Empirical Methods in AI. 2001.
4. Hand, D. J.; Yu, K. Idiot's Bayes — not so stupid after all?. International Statistical Review. 2001, **69** (3): 385–399. ISSN 0306-7734. doi:10.2307/1403452.
5. Harry Zhang "The Optimality of Naive Bayes". FLAIRS2004 conference. *(available online: PDF (http://www.cs.unb.ca/profs/hzhang/publications/FLAIRS04ZhangH.pdf))*
6. Caruana, R. and Niculescu-Mizil, A.: "An empirical comparison of supervised learning algorithms". Proceedings of the 23rd international conference on Machine learning, 2006. *(available online [1] (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.122.5901&rep=rep1&type=pdf PDF))*
7. George H. John and Pat Langley (1995). Estimating Continuous Distributions in Bayesian Classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338-345. Morgan Kaufmann, San Mateo.

延伸阅读

- Domingos, Pedro; Pazzani, Michael. On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning. 1997, **29**: 103–137.
- Webb, G. I.; Boughton, J.; Wang, Z. Not So Naive Bayes: Aggregating One-Dependence Estimators. Machine Learning (Springer). 2005, **58** (1): 5–24. doi:10.1007/s10994-005-4258-6.
- Mozina, M.; Demsar, J.; Kattan, M.; Zupan, B. Nomograms for Visualization of Naive Bayesian Classifier (PDF). Proc. PKDD-2004: 337–348. 2004.
- Maron, M. E. Automatic Indexing: An Experimental Inquiry. JACM. 1961, **8** (3): 404–417. doi:10.1145/321075.321084.
- Minsky, M. Steps toward Artificial Intelligence. Proc. IRE: 8–30. 1961.

外部链接

- Book Chapter: Naive Bayes text classification, Introduction to Information Retrieval
- Naive Bayes for Text Classification with Unbalanced Classes
- Benchmark results of Naive Bayes implementations
- Hierarchical Naive Bayes Classifiers for uncertain data (an extension of the Naive Bayes classifier).

软件

- Naive Bayes classifiers are available in many general-purpose machine learning and NLP packages, including Apache Mahout, Mallet, NLTK, Orange, scikit-learn and Weka.
- IMSL Numerical Libraries Collections of math and statistical algorithms available in C/C++, Fortran, Java and C#/.NET. Data mining routines in the IMSL Libraries include a Naive Bayes classifier.
- Winnow content recommendation An open source Naive Bayes text classifier works with very small training and unbalanced training sets. High performance, C, any Unix.
- An interactive Microsoft Excel spreadsheet Naive Bayes implementation using VBA (requires enabled macros) with viewable source code.
- jBNC - Bayesian Network Classifier Toolbox.
- Statistical Pattern Recognition Toolbox for Matlab.
- ifile - the first freely available (Naive) Bayesian mail/spam filter
- NClassifier - NClassifier is a .NET library that supports text classification and text summarization. It is a port of Classifier4J.
- Classifier4J - Classifier4J is a Java library designed to do text classification. It comes with an implementation of a Bayesian classifier