中国科学: 信息科学 2023 年 第 53 卷 第 6 期: 1078–1095
SCIENTIA SINICA Informationis
《中国科学》杂志社
SCIENCE CHINA PRESS
论文

一种基于偏差 – 方差权衡的贝叶斯分类学习框架

张文钧[1], 蒋良孝[1,2*], 张欢[1], 胡成玉[1]

1. 中国地质大学计算机学院, 武汉 430074
2. 教育部人工智能重点实验室, 上海 200240
* 通信作者. E-mail: ljiang@cug.edu.cn

收稿日期: 2022–01–15; 修回日期: 2022–04–21; 接受日期: 2022–09–30; 网络出版日期: 2023–06–07

国家自然科学基金面上项目 (批准号: 62276241, 62073300)、湖北省揭榜制重大科技项目 (批准号: 2021BEC007) 和教育部人工智能重点实验室开放基金项目 (批准号: AI2020002) 资助

摘要 朴素贝叶斯由于其简单、高效和有效性成为十大数据挖掘算法之一. 然而它要求的属性条件独立假设在实际应用中很难成立. 为了削弱其属性条件独立假设, 学者们提出了结构扩展、属性选择、属性加权、实例选择、实例加权 5 类改进方法. 现有改进方法虽然在一定程度上降低了模型的偏差, 但同时也提高了模型的方差, 因而限制了模型的泛化性能. 偏差 – 方差权衡是机器学习的核心原则之一, 该原则要求模型具有较低偏差的同时, 方差也要尽量低. 如何在贝叶斯分类学习中引入偏差 – 方差权衡, 同时获得较低的偏差和方差, 从而进一步提升模型的泛化性能, 是本文关注的重点. 为此, 本文首先理论分析了在贝叶斯分类学习中做偏差 – 方差权衡的可行性, 探讨了保证可行性的关键因素; 然后通过构建回归任务来学习贝叶斯分类模型的后验概率损失, 调控关键因素的变化; 最后提出了一种基于偏差 – 方差权衡的贝叶斯分类学习框架, 并在提出的学习框架下重新实现了朴素贝叶斯及其各类改进模型. 在大量经典的 UCI 标准数据集上的实验结果表明, 现有的各类先进的贝叶斯分类模型在本文所提学习框架下的分类性能显著优于其原始性能.

关键词 朴素贝叶斯, 属性条件独立假设, 偏差 – 方差权衡, 后验概率损失, 学习框架

1 引言

分类是数据挖掘和模式识别的基本任务 [1]. 朴素贝叶斯 (naive Bayes, NB) 由于其简单、高效和有效性而成为十大数据挖掘算法之一 [2]. 假设 $A_1, A_2, \ldots, A_m$ 是 m 个属性变量, 给定一个测试实例 $\boldsymbol{x}$ 可由属性值向量 $\langle a_1, a_2, \ldots, a_m\rangle$ 表示, 其中 a_m 是 A_m 的属性值. 用 C 表示实例类别的集合, c 表示集合 C 中的类别元素, NB 使用式 (1) 来分类测试实例 $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \arg\max_{c\in C} P(c)\prod_{j=1}^{m} P(a_j|c), \tag{1}$$

引用格式: 张文钧, 蒋良孝, 张欢, 等. 一种基于偏差 – 方差权衡的贝叶斯分类学习框架. 中国科学: 信息科学, 2023, 53: 1078–1095, doi: 10.1360/SSI-2022-0025
Zhang W J, Jiang L X, Zhang H, et al. Bayesian classification learning framework based on bias–variance trade-off (in Chinese). Sci Sin Inform, 2023, 53: 1078–1095, doi: 10.1360/SSI-2022-0025

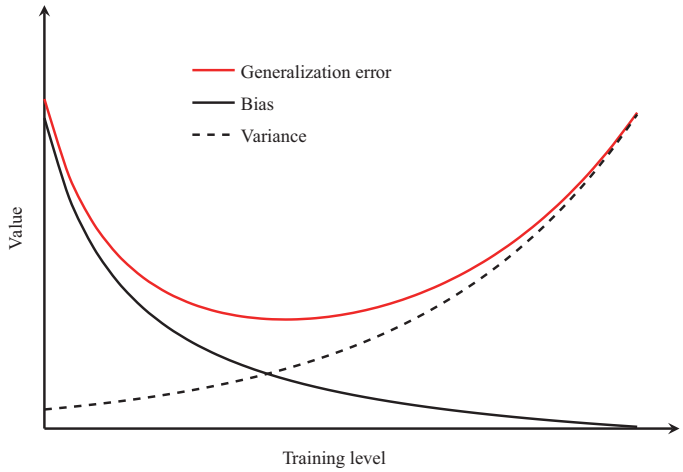

图 1 (网络版彩图) 泛化误差与偏差、方差的关系示意图

Figure 1 (Color online) The diagram of the relationship between generalization error, bias and variance

其中 $c(\boldsymbol{x})$ 为 NB 预测 $\boldsymbol{x}$ 的类别, $P(c)$ 是类别 c 的先验概率, $P(a_j|c)$ 是给定类别 c 时属性变量 A_j 取值为 a_j 的概率, 分别用式 (2) 和 (3) 来估计:

$$P(c) = \frac{\sum_{i=1}^{n} \delta(c_i, c) + 1}{n + n_c}, \tag{2}$$

$$P(a_j|c) = \frac{\sum_{i=1}^{n} \delta(a_{ij}, a_j)\, \delta(c_i, c) + 1}{\sum_{i=1}^{n} \delta(c_i, c) + n_j}, \tag{3}$$

其中 n 是训练实例个数, n_c 是类别个数, c_i 是第 i 个训练实例的类别, a_{ij} 是第 i 个训练实例的第 j 个属性值, n_j 是属性 A_j 的属性值个数, $\delta(\cdot)$ 是二元函数, 两个参数相同时为 1, 否则为 0.

给定训练数据集后, $P(c)$ 和 $P(a_j|c)$ 很容易被估计, 所以 NB 是一个非常简单、高效的模型. 有研究表明 [3], NB 的分类性能可以与其他先进的分类模型, 比如 C4.5, 相媲美. 然而, NB 要求的属性条件独立假设, 即给定类别时所有属性是完全相互独立的, 在实际应用中很难成立. 为了削弱其属性条件独立假设, 学者们提出了许多改进方法, 这些方法大致可以归纳为 5 类: 结构扩展 [3~7]、属性选择 [8~12]、属性加权 [13~19]、实例选择 [20~24]、实例加权 [25~27]. 所有这些改进方法虽然在一定程度上降低了模型的偏差, 但同时也提高了模型的方差, 因而限制了模型的泛化性能.

图 1 为泛化误差与偏差、方差关系示意图 [28]. 由图可知, 给定一个学习任务, 模型太过简单时, 对数据集拟合不足, 偏差较高而方差较低; 当模型拟合能力逐渐加强, 模型对数据扰动更加敏感, 导致偏差降低但方差提高. 偏差 – 方差权衡 [29,30] (bias–variance trade-off, BVT) 原则希望模型同时获得较低的偏差和方差, 从而获得较好的泛化性能.

若将 BVT 原则引入现有贝叶斯分类模型中, 同时降低模型的偏差和方差, 即可进一步提升模型的泛化性能. 以此为切入点, 本文提出了一种基于 BVT 的贝叶斯分类学习框架. 主要贡献如下:

(1) 总结了改进 NB 的 5 类方法, 并对每一类方法中的经典改进工作进行了系统的综述;

(2) 理论分析了在贝叶斯分类学习中做 BVT 的可行性, 探讨了保证可行性的关键因素;

(3) 构建了回归任务来学习贝叶斯分类模型的后验概率损失, 从而调控关键因素的变化;

(4) 提出了一种基于 BVT 的贝叶斯分类学习框架, 提高了 NB 及其改进模型的泛化性能.

2 相关工作

为了削弱 NB 要求的属性条件独立假设, 学者们提出了结构扩展、属性选择、属性加权、实例选择、实例加权 5 类改进方法, 这一节中对这 5 类方法进行全面的介绍.

2.1 结构扩展

结构扩展方法的基本思想是, 通过在属性之间添加有向边来显式表达属性之间的依赖关系, 使得属性之间不再完全独立, 从而扩展了 NB 的结构. 结构扩展方法使用式 (4) 来分类测试实例 $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \arg\max_{c\in C} P(c)\prod_{j=1}^{m} P\left(a_j|\Pi_{a_j}, c\right), \tag{4}$$

其中 Π_{a_j} 表示属性 A_j 的父亲属性集的属性值集. $P(c)$ 表示先验概率, 用式 (2) 来估计, $P\left(a_j|\Pi_{a_j}, c\right)$ 表示条件概率, 用式 (5) 来估计:

$$P\left(a_j|\Pi_{a_j}, c\right) = \frac{\sum_{i=1}^{n}\delta\left(a_{ij}, a_j\right)\delta\left(\Pi_{a_{ij}}, \Pi_{a_j}\right)\delta\left(c_i, c\right)+1}{\sum_{i=1}^{n}\delta\left(\Pi_{a_{ij}}, \Pi_{a_j}\right)\delta\left(c_i, c\right)+n_j}, \tag{5}$$

其中 $\Pi_{a_{ij}}$ 表示第 i 个训练实例在属性 A_j 的父亲属性集上的属性值集.

学习最优贝叶斯网络结构是一个 NP-hard 问题, 因此在实际的结构学习中常常添加一些约束条件. 为了学习属性 A_j 的父亲属性集, Friedman 等 [3] 假设属性之间的依赖关系可用树形结构表示, 提出了树扩展的朴素贝叶斯 (tree augmented naive Bayes, TAN). Webb 等 [4] 提出了平均的一阶依赖估计 (averaged one-dependence estimators, AODE), AODE 依次将每个属性作为其他所有属性的父亲属性, 由此学习到一个一阶依赖估计的集合, 最终通过直接平均所有符合要求的一阶依赖估计的预测给出最终的分类结果. 作为 AODE 的改进, Li 等 [5] 提出了一种一阶依赖估计的半朴素使用方法 (semi-naive exploitation of ODEs, SNODE). SNODE 认为 AODE 中对概率项求平均的方法不足以很好地近似属性之间的真实依赖关系, 进而构建了一个广义加性模型取代直接求平均的方法, 并通过最大化条件似然对数对广义加性模型的参数进行优化. Jiang 等 [6] 提出了隐朴素贝叶斯 (hidden naive Bayes, HNB), HNB 通过为每一个属性节点分别创建一个隐父亲属性节点, 综合了其他所有属性节点对该属性节点的影响. Wang 等 [7] 提出了一种判别结构学习方法 (unrestricted k-dependence Bayesian classifier, UKDB), UKDB 从 Markov 毯的角度依据训练数据学习一个全局的贝叶斯网络, 同时依据测试实例学习一个局部的贝叶斯网络, 最终根据两个贝叶斯网络预测概率的均值给出最终分类结果.

2.2 属性选择

属性选择方法的基本思想是, 除去原始属性空间中的无关属性和冗余属性, 得到一个最佳属性子集, 然后在最佳属性子集上学习 NB. 属性选择方法使用式 (6) 来分类测试实例 $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \arg\max_{c\in C} P(c)\prod_{j=1}^{s} P\left(a_j|c\right), \tag{6}$$

其中 s 表示最佳属性子集的大小, $P(c)$ 和 $P(a_j|c)$ 仍然使用式 (2) 和 (3) 来估计.

为了学习最佳属性子集, Langley 和 Sage [8] 提出了一种有选择的贝叶斯模型 (selective Bayes, SB), 该方法采用贪婪搜索的方式, 每一次从未被选择的属性集中选择一个最能提高当前模型分类精度的属性, 直至从未被选择的属性集中选择任意属性均不可改善模型分类精度为止. Jiang 等 [9] 提出了进化

的朴素贝叶斯 (evolutional naive Bayes, ENB), 通过进化算法选择最佳属性子集, 降低了模型陷入局部最优的概率. Ratanamahatana 和 Gunopulos [10] 提出了一种基于决策树属性选择的贝叶斯分类模型 (selective Bayesian classifier, SBC), SBC 每次在打乱后的部分训练集上学习一个 C4.5 模型, 并选择在其前 3 层出现过的属性, 将这个过程重复 5 次, 最终采用多次选择的属性的并集作为最佳属性子集, 在此基础上构建贝叶斯分类模型. Chen 等 [11] 提出了一种基于抽样的属性选择技术 (sample-based attribute selective technique, SAS), SAS 首先通过互信息对属性进行排序, 并依据排序后的有序属性序列来选择父亲属性集和子属性集构建模型. 在最近的工作中, Jiang 等 [12] 提出了测试代价敏感的属性选择通用框架 (test-cost-sensitive feature selection, TCSFS), TCSFS 在属性选择中同时优化了分类精度和测试代价, 允许使用者以最低的测试代价选择属性子集, 同时保持了较高的分类精度.

2.3 属性加权

属性加权方法的基本思想是, 首先为每个属性学习一个权值, 然后在属性加权的训练集上构建 NB. 相比于属性选择方法从原始属性集中完全除去一个属性, 属性加权方法拥有更高的普适性. 属性加权方法使用式 (7) 来分类测试实例 $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \arg\max_{c\in C} P(c)\prod_{j=1}^{m} P\left(a_j|c\right)^{w_j}, \tag{7}$$

其中 w_j 为属性 A_j 的权值, $P(c)$ 和 $P(a_j|c)$ 仍然使用式 (2) 和 (3) 来估计.

为了学习属性权值 w_j, Zhang 和 Sheng [13] 提出了基于增益率属性加权的朴素贝叶斯 (gain ratio-based attribute weighted naive Bayes, GRAWNB), 认为增益率较高的属性具有较高的权值, 因此设置权值 w_j 与属性 A_j 的增益率成正比. Hall [14] 提出了基于决策树属性加权的朴素贝叶斯 (decision tree-based attribute weighted naive Bayes, DTAWNB), 依据属性在决策树中的最小深度来为属性加权. Zaidi 等 [15] 提出了一种属性加权模型 (weighting attributes to alleviate naive Bayes' independence assumption, WANBIA), 通过最大化条件似然对数或最小化均方差来优化属性权值. Jiang 等 [16] 考虑到不同类别下, 不同的属性应该具有不同的权值, 进而提出了一种类依赖属性加权的朴素贝叶斯 (class-specific attribute weighted naive Bayes, CAWNB). Lee [17] 提出了一种属性值加权的朴素贝叶斯 (value weighted naive Bayes, VWNB), VWNB 认为不同的属性值应当具有不同的权值, 并通过 KL (Kullback-Leibler) 散度来学习属性值的权值. 受启发于 CAWNB 和 VWNB, Zhang 等 [18] 提出了一种类依赖的属性值加权的朴素贝叶斯 (class-specific attribute value weighted naive Bayes, CAVWNB), 既考虑到在不同类别下, 不同的属性应该具有不同的权值; 也考虑到不同属性值应当具有不同的权值. 最近, Jiang 等 [19] 提出了一种基于相关性属性加权的朴素贝叶斯 (correlation-based feature weighted naive Bayes, CFWNB), CFWNB 不但考虑了属性和类别之间的相关性, 还考虑了属性之间的冗余性.

2.4 实例选择

实例选择方法的基本思想是, 从全部训练集中选取部分训练实例作为测试实例的邻域, 然后在测试实例的邻域构建一个 NB. 实例选择方法仍然使用式 (1) 来分类测试实例 $\boldsymbol{x}$, 但公式中的 $P(c)$ 和 $P(a_j|c)$ 分别使用式 (8) 和 (9) 来估计:

$$P(c) = \frac{\sum_{i=1}^{q}\delta\left(c_i, c\right)+1}{q+n_c}, \tag{8}$$

$$P\left(a_j|c\right) = \frac{\sum_{i=1}^{q}\delta\left(a_{ij}, a_j\right)\delta\left(c_i, c\right)+1}{\sum_{i=1}^{q}\delta\left(c_i, c\right)+n_j}, \tag{9}$$

其中 q 表示测试实例 $\boldsymbol{x}$ 的邻域中训练实例的个数.

为了快速学习实例 $\boldsymbol{x}$ 的邻域, Kohavi [20] 结合决策树和 NB 的优势提出了朴素贝叶斯树 (naive Bayes tree, NBTree), NBTree 的生长过程类似于 C4.5, 但是选择最优划分属性时的度量标准采用 NBTree 的分类精度而不是增益率. NBTree 在所有的叶子节点上都构建了 NB, 用叶子节点对应的 NB 为测试实例分类. Frank 等 [21] 提出了局部加权的朴素贝叶斯 (locally weighted naive Bayes, LWNB), 该方法首先利用 KNN 算法找到测试实例的邻域, 然后依据邻域内的实例到测试实例的距离进行加权, 最后在加权后的邻域实例上构建 NB. Jiang 等 [22] 提出了组合邻域的朴素贝叶斯 (combined neighbourhood naive Bayes, CNNB), CNNB 基于不同的邻域大小构建一组 NB, 然后通过平均它们的类概率估计来分类测试实例. Zheng 和 Webb [23] 提出了一种消极的贝叶斯规则学习模型 (lazy Bayesian rule, LBR), LBR 首先为每个测试实例产生一组规则, 然后选择满足规则的训练实例来构成测试实例的邻域, 接着在邻域上构建 NB, 最后用构建的 NB 来分类测试实例. 最近, Hindi 等 [24] 提出了一种消极的微调朴素贝叶斯 (lazy fine-tuning naive Bayes, LFTNB), LFTNB 首先需要找到测试实例的邻域, 然后依据邻域内的实例调整 NB 使用到的概率项, 最终可以使得 NB 的概率估计更加准确.

2.5 实例加权

实例加权方法的基本思想是, 先为训练集中不同的训练实例学习不同的权值, 然后使用加权后的训练集构建 NB. 实例加权方法仍然使用式 (1) 来分类测试实例 $\boldsymbol{x}$, 但公式中的 $P(c)$ 和 $P(a_j|c)$ 分别使用式 (10) 和 (11) 来估计:

$$P(c) = \frac{\sum_{i=1}^{n} w_i \delta(c_i, c) + 1}{\sum_{i=1}^{n} w_i + n_c}, \tag{10}$$

$$P(a_j|c) = \frac{\sum_{i=1}^{n} w_i \delta(a_{ij}, a_j)\delta(c_i, c) + 1}{\sum_{i=1}^{n} w_i \delta(c_i, c) + n_j}, \tag{11}$$

其中 w_i 表示训练集中第 i 个训练实例的权值.

为了学习实例权值 w_i, Elkan [25] 结合 boosting 技术提出了提升的朴素贝叶斯 (boosted naive Bayes, BNB), BNB 为当前迭代中误分类的实例赋予更高的权值, 更新权值后的训练集被用于学习下一轮的 NB. BNB 的最终输出是多次迭代产生的不同 NB 的输出加权和, 其中每个 NB 的权值为其在训练集上的分类精度. 为了降低集成学习带来的计算难度, Jiang 等 [26] 提出了判别加权的朴素贝叶斯 (discriminatively weighted naive Bayes, DWNB). DWNB 用实例真实后验概率与模型估计后验概率的差值作为下一次迭代中实例的权值增量, 最终返回最后一次构建的 NB. 最近, Xu 等 [27] 提出了属性值频率加权的朴素贝叶斯 (attribute value frequency weighted naive Bayes, AVFWNB), AVFWNB 用实例的属性值频率向量和属性值个数向量的内积作为实例权值, 无需多次迭代, 具有简单高效的特点.

3 基于 BVT 的贝叶斯分类学习框架

3.1 BVT 的定义与可行性分析

偏差 – 方差分解 (bias–variance decomposition) 是理解机器学习算法的一个有用的工具. Geman 等 [29] 最初针对基于均方误差的回归任务提出了偏差 – 方差分解, 揭示了学习任务内在的误差决定因素. 与回归任务相比, 分类任务具有跳变性, 原始的分解方法不再适用. 针对分类任务, 目前已有多种

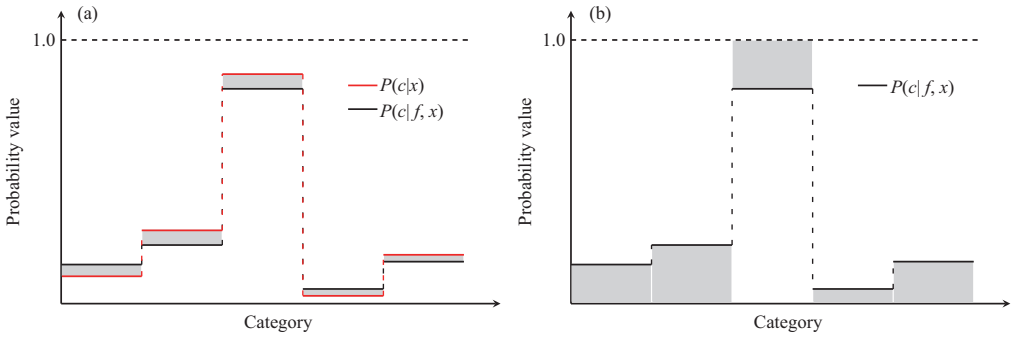

图 2 (网络版彩图) 偏差及方差示意图

Figure 2 (Color online) The diagram of (a) bias and (b) variance

偏差 – 方差分解的定义, 本文采用 Kohavi 和 Wolpert [30] 给出的定义:

$$E(f)=\sum_{\boldsymbol{x}}P(\boldsymbol{x})\left(\mathrm{bias}_{\boldsymbol{x}}^2+\mathrm{var}_{\boldsymbol{x}}+\sigma_{\boldsymbol{x}}^2\right), \tag{12}$$

其中 f 表示训练好的模型, $E(f)$ 表示模型 f 的泛化误差, $\boldsymbol{x}$ 仍表示测试实例, $P(\boldsymbol{x})$ 表示从测试集选中 $\boldsymbol{x}$ 的概率, $\mathrm{bias}_{\boldsymbol{x}}^2$, $\mathrm{var}_{\boldsymbol{x}}$, $\sigma_{\boldsymbol{x}}^2$ 分别表示偏差项、方差项和噪声项, 具体计算公式如下:

$$\mathrm{bias}_{\boldsymbol{x}}^2=\frac{1}{2}\sum_{c\in C}\left[P\left(c|\boldsymbol{x}\right)-P\big(c|f,\boldsymbol{x}\big)\right]^2, \tag{13}$$

$$\mathrm{var}_{\boldsymbol{x}}=\frac{1}{2}\left[1-\sum_{c\in C}P\big(c|f,\boldsymbol{x}\big)^2\right], \tag{14}$$

$$\sigma_{\boldsymbol{x}}^2=\frac{1}{2}\left[1-\sum_{c\in C}P\left(c|\boldsymbol{x}\right)^2\right], \tag{15}$$

其中 $P\left(c|\boldsymbol{x}\right)$ 表示实例 $\boldsymbol{x}$ 属于类别 c 的真实概率分布, 取值与模型无关. 无噪声情况下当 c 为 $\boldsymbol{x}$ 的真实类别时 $P\left(c|\boldsymbol{x}\right)$ 取值为 1, 否则取值为 0. $P\big(c|f,\boldsymbol{x}\big)$ 表示模型 f 在交叉验证产生的多组分类结果中将 $\boldsymbol{x}$ 判为类别 c 的观测概率分布. 由于 $\sigma_{\boldsymbol{x}}^2$ 的值较小, 通常假设 $\sigma_{\boldsymbol{x}}^2$ 的期望为 0, 因此在分析泛化误差时, 往往只分析 $\mathrm{bias}_{\boldsymbol{x}}^2$ 和 $\mathrm{var}_{\boldsymbol{x}}$ 的影响.

基于上述定义, 本文分别使用图 2(a) 和 (b) 来表示偏差和方差的示意图, 图中阴影区域的大小象征了偏差和方差的大小. 依据式 (13) 和图 2(a), 偏差度量了模型 f 的观测概率分布 $P\big(c|f,\boldsymbol{x}\big)$ 与训练数据的真实概率分布 $P\left(c|\boldsymbol{x}\right)$ 之间的差距, 偏差越小, 模型的观测概率分布和训练数据的真实概率分布越一致; 依据式 (14) 和图 2(b), 方差度量了模型 f 的观测概率分布 $P\big(c|f,\boldsymbol{x}\big)$ 的置信度, 方差越小, $P\big(c|f,\boldsymbol{x}\big)$ 的值越趋近于 0 或 1. 在无噪声情况下 $P\left(c|\boldsymbol{x}\right)$ 的取值为 0 或 1, 此时偏差的降低同样可以使得方差降低, 因此无噪声情况下降低偏差可以使得模型的泛化误差降低. 但由于在实际分类问题中, 训练数据通常包含一些噪声, 如图 2(a) 所示, $P\left(c|\boldsymbol{x}\right)$ 的取值可能并不是 0 或 1, 而是 0∼1 之间的值. 此时只考虑降低偏差, 会使得 $P\big(c|f,\boldsymbol{x}\big)$ 的值偏离 0 或 1, 这将导致方差增大, 最终导致泛化误差无法降低. 降低方差可以减弱噪声对模型的影响, 使得模型不易陷入过拟合麻烦, 但由于方差的计算不受训练数据的真实概率分布影响, 若只考虑方差的降低, 模型的预测结果虽然更加绝对, 但也可能是绝对的 “错”. 因此为了降低模型的泛化误差, 通常需要同时考虑偏差和方差的影响.

3.1.1 BVT 的定义

事实上, 偏差和方差之间的变化往往存在冲突. 当模型 f 的拟合能力发生变化时, 观测概率分布 $P(c|f,\boldsymbol{x})$ 随之变化, 设 $P(c|f,\boldsymbol{x})$ 的变化量为 $\Delta_{\boldsymbol{x},c}$, 将 $\Delta_{\boldsymbol{x},c}$ 代入式 (13) 和 (14) 中, 则有

$$\begin{aligned}\tilde{\text{bias}}_{\boldsymbol{x}}^2 &= \frac{1}{2}\sum_{c\in C}\left[P\left(c|\boldsymbol{x}\right)-\left(P(c|f,\boldsymbol{x})+\Delta_{\boldsymbol{x},c}\right)\right]^2\\ &= \text{bias}_{\boldsymbol{x}}^2+\sum_{c\in C}\Delta_{\boldsymbol{x},c}P(c|f,\boldsymbol{x})+\frac{1}{2}\sum_{c\in C}\Delta_{\boldsymbol{x},c}^2-\sum_{c\in C}\Delta_{\boldsymbol{x},c}P\left(c|\boldsymbol{x}\right),\end{aligned}\tag{16}$$

$$\begin{aligned}\tilde{\text{var}}_{\boldsymbol{x}} &= \frac{1}{2}\left[1-\sum_{c\in C}\left(P(c|f,\boldsymbol{x})+\Delta_{\boldsymbol{x},c}\right)^2\right]\\ &= \text{var}_{\boldsymbol{x}}-\sum_{c\in C}\Delta_{\boldsymbol{x},c}P(c|f,\boldsymbol{x})-\frac{1}{2}\sum_{c\in C}\Delta_{\boldsymbol{x},c}^2,\end{aligned}\tag{17}$$

其中用 $\tilde{\cdot}$ 表示 $\cdot$ 的更新值.

比较式 (16) 和 (17) 不难发现, 公共变化项 $\frac{1}{2}\sum_{c\in C}\Delta_{\boldsymbol{x},c}^2$ 与 $\sum_{c\in C}\Delta_{\boldsymbol{x},c}P(c|f,\boldsymbol{x})$ 在 $\tilde{\text{bias}}_{\boldsymbol{x}}^2$ 和 $\tilde{\text{var}}_{\boldsymbol{x}}$ 中的符号刚好相反, 导致偏差和方差的变化趋势总是相反的, 此现象称为偏差 – 方差窘境 (bias–variance dilemma). 偏差 – 方差窘境导致模型的泛化误差与偏差、方差之间的变化趋势如图 1 所示. 要求同时得到较低的偏差和方差, 使得模型泛化误差较低的原则被称为偏差 – 方差权衡 (BVT).

3.1.2 可行性分析

依据式 (12) 中泛化误差的定义, 当模型 f 的拟合能力发生变化时, 将随之变化的偏差 $\tilde{\text{bias}}_{\boldsymbol{x}}^2$ 和方差 $\tilde{\text{var}}_{\boldsymbol{x}}$ 代入式 (12), 即可得到变化后的泛化误差 $\tilde{E}(f)$, 最终化简得到

$$\tilde{E}(f)=E(f)-\sum_{\boldsymbol{x}}P(\boldsymbol{x})\sum_{c\in C}\Delta_{\boldsymbol{x},c}P\left(c|\boldsymbol{x}\right),\tag{18}$$

一般情况下, 当 c 为 $\boldsymbol{x}$ 的真实类别时, 真实的后验概率 $P\left(c|\boldsymbol{x}\right)$ 趋近于 1, 否则趋近于 0. 因此若用 $c_{\boldsymbol{x}}$ 表示实例 $\boldsymbol{x}$ 的真实类别: 当 $c\neq c_{\boldsymbol{x}}$ 时, 无论 $\Delta_{\boldsymbol{x},c}$ 是否大于 0, $\Delta_{\boldsymbol{x},c}P\left(c|\boldsymbol{x}\right)$ 均趋近于 0, 不会对式 (18) 产生较大影响; 当 $c=c_{\boldsymbol{x}}$ 时, 由于式 (18) 中概率项 $P(\boldsymbol{x})$ 和 $P\left(c|\boldsymbol{x}\right)$ 非负, 因此只要 $\Delta_{\boldsymbol{x},c}$ 大于 0, 即可保证 $\tilde{E}(f)$ 小于 $E(f)$. 综上可知, 实现 BVT 的关键因素为观测概率 $P\left(c_{\boldsymbol{x}}|f,\boldsymbol{x}\right)$ 的变化量 $\Delta_{\boldsymbol{x},c_{\boldsymbol{x}}}$. 当模型 f 的拟合能力发生变化时, 为使 $\Delta_{\boldsymbol{x},c_{\boldsymbol{x}}}$ 大于 0, f 需满足在给定实例 $\boldsymbol{x}$ 时, 可以较变化之前更准确地将 $\boldsymbol{x}$ 划分到其真实类别 $c_{\boldsymbol{x}}$ 这一条件. 为满足这一条件, 可以构建以下优化任务:

$$\begin{aligned}&\underset{\boldsymbol{x}}{\text{maximize}}\quad P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})\\ &\text{s.t.}\quad P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})-\max_{c\in C,c\neq c_{\boldsymbol{x}}}P(c)\prod_{j=1}^{m}P(a_j|c)\geqslant 0.\end{aligned}\tag{19}$$

参考式 (1) 可知, 最大化 $P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})$ 有助于模型 f 在分类时更容易将 $\boldsymbol{x}$ 的类别预测为 $c_{\boldsymbol{x}}$, 同时式 (19) 中的约束条件也进一步保证了在多分类情况下, 最大化 $P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})$ 的同时, 其余类别中对应的最大值 $\max_{c\in C,c\neq c_{\boldsymbol{x}}}P(c)\prod_{j=1}^{m}P(a_j|c)$ 不会同步增大且超过 $P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})$ 进而导致分类错误. 依据拉格朗日乘数法 (Lagrange multiplier) [31], 可以构造如下拉格朗日函数:

$$L(\boldsymbol{x})=P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})+\lambda\left[P(c_{\boldsymbol{x}})\prod_{j=1}^{m}P(a_j|c_{\boldsymbol{x}})-\max_{c\in C,c\neq c_{\boldsymbol{x}}}P(c)\prod_{j=1}^{m}P(a_j|c)\right],\tag{20}$$

使得式 (19) 表示的带约束的优化任务转化为最大化 $L(\boldsymbol{x})$, 其中 $\lambda \geqslant 0$. 为便于分析, 可进一步构造 $L'(\boldsymbol{x})$, $L'(\boldsymbol{x})$ 满足

$$\begin{aligned} L'(\boldsymbol{x}) &= L(\boldsymbol{x}) - \max_{c\in C, c\neq c_{\boldsymbol{x}}} P(c)\prod_{j=1}^{m} P(a_j|c) \\ &= (1+\lambda)\left[P(c_{\boldsymbol{x}})\prod_{j=1}^{m} P(a_j|c_{\boldsymbol{x}}) - \max_{c\in C, c\neq c_{\boldsymbol{x}}} P(c)\prod_{j=1}^{m} P(a_j|c)\right]. \end{aligned} \tag{21}$$

由于概率值 $P(c)$ 和 $P(a_j|c)$ 均大于或等于 0, $L(\boldsymbol{x}) \geqslant L'(\boldsymbol{x})$ 恒成立, 因此最终可通过优化 $L'(\boldsymbol{x})$ 使得 $L(\boldsymbol{x})$ 取得较优结果. 同时由于 $\lambda \geqslant 0$, 因此 $L'(\boldsymbol{x})$ 的值与如下差式正相关:

$$P(c_{\boldsymbol{x}})\prod_{j=1}^{m} P(a_j|c_{\boldsymbol{x}}) - \max_{c\in C, c\neq c_{\boldsymbol{x}}} P(c)\prod_{j=1}^{m} P(a_j|c). \tag{22}$$

综上可知, 提高差式 (22) 中差值的大小, 有助于 $\Delta_{\boldsymbol{x},c_{\boldsymbol{x}}} > 0$ 的实现. 受文献 [26] 启发, 模型 f 估计的后验概率 $\hat{P}(c_{\boldsymbol{x}}|f,\boldsymbol{x})$ 损失 (与 $P(c_{\boldsymbol{x}}|\boldsymbol{x})$ 的差值) 对分类有指示意义, 且可被学习. 若构建回归任务, 使得 f 对测试实例 $\boldsymbol{x}$ 估计的后验概率损失可被拟合, 且将其补全到差式 (22) 中的被减数 $P(c_{\boldsymbol{x}})\prod_{j=1}^{m} P(a_j|c_{\boldsymbol{x}})$, 则有望提高差值, 进而保证 $\Delta_{\boldsymbol{x},c_{\boldsymbol{x}}} > 0$, 最终保证了通过 BVT 提升 f 的泛化性能的可行性.

3.2 回归任务的构建

模型 f 仍以 NB 为例, c_k 表示第 k 个类别, 给定测试实例 $\boldsymbol{x}$, 后验概率 $\hat{P}(c_k|f,\boldsymbol{x})$ 使用式 (23) 来估计:

$$\hat{P}(c_k|f,\boldsymbol{x}) = \frac{P(c_k)\prod_{j=1}^{m} P(a_j|c_k)}{\sum_{c\in C} P(c)\prod_{j=1}^{m} P(a_j|c)}. \tag{23}$$

若用 l_{ik} 表示 D 中第 i 个训练实例 $\boldsymbol{x}_i$ 回代到模型 f 中得到在第 k 个类别 c_k 上的后验概率损失, 则可使用式 (24) 构建一个损失矩阵 $\boldsymbol{L}$:

$$\boldsymbol{L} = \begin{pmatrix} l_{11} & l_{12} & \cdots & l_{1n_c} \\ l_{21} & l_{22} & \cdots & l_{2n_c} \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn_c} \end{pmatrix} = \left(\boldsymbol{l}_1;\boldsymbol{l}_2;\ldots;\boldsymbol{l}_{n_c}\right), \tag{24}$$

其中 $\boldsymbol{l}_k$ 表示 D 中 n 个训练实例在 c_k 上的后验概率损失, 是一个列向量, 满足 $\boldsymbol{l}_k^{\mathrm{T}} = \langle l_{1k}, l_{2k}, \ldots, l_{nk}\rangle$, 其中 l_{ik} 使用式 (25) 来计算:

$$l_{ik} = \begin{cases} P(c_k|\boldsymbol{x}_i) - \hat{P}(c_k|f,\boldsymbol{x}_i), & c_k = c_{\boldsymbol{x}_i}, \\ 0, & c_k \neq c_{\boldsymbol{x}_i}, \end{cases} \tag{25}$$

其中 $c_{\boldsymbol{x}_i}$ 为 $\boldsymbol{x}_i$ 的真实类别. 根据前面的讨论, 当 $c_k = c_{\boldsymbol{x}_i}$ 时 $P(c_k|\boldsymbol{x}_i)$ 的值为 1, 因此损失向量 $\boldsymbol{l}_k$ 中只包含真实类别为 c_k 的实例的后验概率损失, 而其余真实类别不为 c_k 的实例的后验概率损失均被设置

为 0, 这使得回归任务更容易被学习. 为了进一步提高差式 (22) 中的差值, 为 c_k 添加一个固定的补充向量 $\boldsymbol{\beta}_k^{\mathrm{T}} = \langle \beta_{1k}, \beta_{2k}, \ldots, \beta_{nk} \rangle$, β_{ik} 使用式 (26) 来计算:

$$\beta_{ik} = \begin{cases} \gamma, & c_k = c_{\boldsymbol{x}_i}, \\ 0, & c_k \neq c_{\boldsymbol{x}_i}, \end{cases} \tag{26}$$

其中, γ 是一个经验参数, 若无特殊说明, 本文默认设置为 1.

根据类别 c_k 的损失向量和补充向量, 可使用式 (27) 构建一个回归任务 T_k:

$$T_k = (\hat{\boldsymbol{X}}; \boldsymbol{l}_k + \boldsymbol{\beta}_k), \tag{27}$$

其中矩阵 $\hat{\boldsymbol{X}}$ 使用式 (28) 构建

$$\hat{\boldsymbol{X}} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} & 1 \\ a_{21} & a_{22} & \cdots & a_{2m} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} & 1 \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{x}}_1 \\ \hat{\boldsymbol{x}}_2 \\ \vdots \\ \hat{\boldsymbol{x}}_n \end{pmatrix}, \tag{28}$$

其中 $\hat{\boldsymbol{x}}_i$ 表示任务 T_k 中的第 i 个训练实例, 可由属性值向量 $\langle a_{i1}, a_{i2}, \ldots, a_{im}, 1 \rangle$ 来表示.

本文所提学习框架通过一组回归模型来学习上述回归任务. 虽然任意回归模型均可适用于本文所提学习框架, 不过为了在证明学习框架有效性的同时保证框架的简洁性, 本文选择了时间复杂度较低的线性回归模型. 本文针对 T_k 的线性回归模型 h_k 应当满足

$$h_k(\hat{\boldsymbol{x}}_i) = \hat{\boldsymbol{x}}_i \boldsymbol{w}_k^{\mathrm{T}}, \text{ 使得 } h_k(\hat{\boldsymbol{x}}_i) \simeq l_{ik} + \beta_{ik}, \tag{29}$$

$\boldsymbol{w}_k$ 是参数向量, 为了确定最优参数向量 $\boldsymbol{w}_k^*$, 使得 h 在 T_k 上的均方误差最低, $\boldsymbol{w}_k^*$ 需满足

$$\boldsymbol{w}_k^* = \arg\min_{\boldsymbol{w}_k} (\boldsymbol{l}_k + \boldsymbol{\beta}_k - \hat{\boldsymbol{X}} \boldsymbol{w}_k^{\mathrm{T}})^{\mathrm{T}} (\boldsymbol{l}_k + \boldsymbol{\beta}_k - \hat{\boldsymbol{X}} \boldsymbol{w}_k^{\mathrm{T}}). \tag{30}$$

上式对 $\boldsymbol{w}_k^{\mathrm{T}}$ 求导并令导数为 0, 可解得 $\boldsymbol{w}_k^*$ 的闭式解如下:

$$\boldsymbol{w}_k^* = \left((\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})^{-1} \hat{\boldsymbol{X}}^{\mathrm{T}} (\boldsymbol{l}_k + \boldsymbol{\beta}_k) \right)^{\mathrm{T}}, \tag{31}$$

其中 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})^{-1}$ 表示矩阵 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})$ 的逆矩阵或伪逆矩阵: 当 $|\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}}|$ 不为 0 时为逆矩阵; 当 $|\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}}|$ 为 0 时为伪逆矩阵, 此时先通过式 (32) 更新 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})$:

$$\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}} = \begin{pmatrix} x_{11} + \Lambda & x_{12} & \cdots & x_{1(m+1)} \\ x_{21} & x_{22} + \Lambda & \cdots & x_{2(m+1)} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(m+1)1} & x_{(m+1)2} & \cdots & x_{(m+1)(m+1)} + \Lambda \end{pmatrix}, \tag{32}$$

其中 x_{ij} 表示矩阵 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})$ 中未更新的元素, Λ 的初始值设为 0.1. 若更新后 $|\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}}|$ 仍为 0, 使 $\Lambda = 10\Lambda$, 继续执行式 (32) 直至 $|\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}}|$ 不为 0 为止, 最后用更新后的 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})$ 求伪逆矩阵 $(\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})^{-1}$.

给定测试实例 $\hat{\boldsymbol{x}}$, 线性回归模型 h_k 使用式 (33) 来计算 $\hat{\boldsymbol{x}}$ 在类别 c_k 下的损失:

$$h_k(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{x}} (\hat{\boldsymbol{X}}^{\mathrm{T}} \hat{\boldsymbol{X}})^{-1} \hat{\boldsymbol{X}}^{\mathrm{T}} (\boldsymbol{l}_k + \boldsymbol{\beta}_k). \tag{33}$$

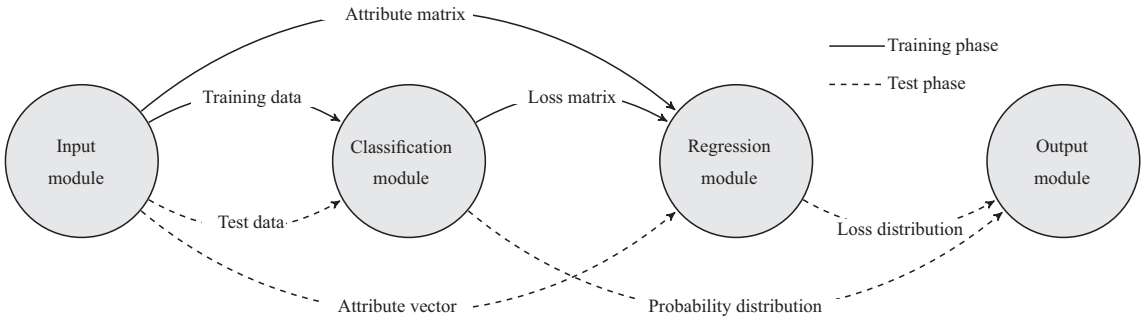

图 3 基于 BVT 的贝叶斯分类学习框架

Figure 3 Bayesian classification learning framework based on bias–variance trade-off

3.3 学习框架的详细实现

基于前文的分析推导, 这一节对本文所提贝叶斯分类学习框架的具体结构及实现做进一步描述, 具体如图 3 所示. 其中, 实线表示训练阶段, 虚线表示测试阶段, 节点表示不同模块, 连接节点的边表示模块间的信息传递方向以及内容. 下文结合训练阶段和测试阶段对框架中的细节做进一步描述.

3.3.1 训练阶段

本文所提学习框架的训练阶段主要分为两步, 首先根据输入模块提供的训练数据完成贝叶斯分类模型 f 的训练, 然后根据训练数据的属性矩阵和回代模型 f 生成的损失矩阵构建回归任务. 图 3 中的属性矩阵 $\hat{\boldsymbol{X}}$ 使用式 (28) 构建, 损失矩阵 $\boldsymbol{L}$ 使用式 (24) 构建, 最后针对任意类别 c_k 使用式 (31) 学习一个线性回归模型 h_k. 综上, 本文所提学习框架在训练阶段输出分类模型 f 和回归模型集合 $\boldsymbol{H} = \{h_1, h_2, \ldots, h_{n_c}\}$. 分类模型 f 以 NB 为例, 训练阶段可描述为算法 1.

Algorithm 1 Bayesian classification learning framework based on bias–variance trade-off (training phase)

Input: Training dataset D.

Output: Bayesian classification model f, regression model set $\boldsymbol{H}$.

Estimate $P(c)$ and $P(a_j|c)$ of f by Eqs. (2) and (3);
for k from 1 to n_c **do**
  for i from 1 to n **do**
    Calculate the posterior probability $\hat{P}(c_k|f, \boldsymbol{x}_i)$ of $\boldsymbol{x}_i \in D$ by Eq. (23);
    Calculate the probability loss l_{ik} by Eq. (25);
  end for
  Construct the loss matrix $\boldsymbol{L} = (\boldsymbol{l}_1; \boldsymbol{l}_2; \ldots; \boldsymbol{l}_{n_c})$ by Eq. (24);
end for
Construct the attribute matrix $\hat{\boldsymbol{X}}$ by Eq. (28);
for $k = 1, 2, \ldots, n_c$ **do**
  Construct the complementary vector $\boldsymbol{\beta}_k$ by Eq. (26);
  Construct the regression task T_k by Eq. (27);
  Learn the regression model h_k by Eq. (31);
end for
return Bayesian classification model f, $\boldsymbol{H} = \{h_1, h_2, \ldots, h_{n_c}\}$.

3.3.2 测试阶段

本文所提学习框架的测试阶段同样分为两步, 首先根据输入模块提供的测试数据分别从分类模块和回归模块获得模型估计的后验概率分布和损失分布, 然后由输出模块给出最终分类结果.

给定测试实例 $\boldsymbol{x}$, 用 $\hat{\boldsymbol{P}}_{\boldsymbol{x}}$ 表示 f 估计的后验概率分布, 由 $\langle \hat{P}(c_1|f,\boldsymbol{x}), \hat{P}(c_2|f,\boldsymbol{x}), \ldots, \hat{P}(c_{n_c}|f,\boldsymbol{x})\rangle$ 表示, 其中 $\hat{P}(c_k|f,\boldsymbol{x})$ 使用式 (23) 估计; 构造实例 $\hat{\boldsymbol{x}} = (\boldsymbol{x};1)$ 作为回归模型的输入, 用 $\boldsymbol{L}_{\hat{\boldsymbol{x}}}$ 表示 $\boldsymbol{H}$ 估计的损失分布, 由 $\langle h_1(\hat{\boldsymbol{x}}), h_2(\hat{\boldsymbol{x}}), \ldots, h_{n_c}(\hat{\boldsymbol{x}})\rangle$ 表示, 其中 $h_k(\hat{\boldsymbol{x}})$ 使用式 (33) 估计. 输出模块汇总分类模块提供的 $\hat{\boldsymbol{P}}_{\boldsymbol{x}}$ 和回归模块提供的 $\boldsymbol{L}_{\hat{\boldsymbol{x}}}$ 后, 使用式 (34) 分类测试实例 $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \arg\max_{c_k \in C} (\hat{P}(c_k|f,\boldsymbol{x}) + h_k(\hat{\boldsymbol{x}})). \tag{34}$$

综上, 本文所提学习框架在测试阶段输出测试实例 $\boldsymbol{x}$ 的最终预测类别 $c(\boldsymbol{x})$. 分类模型 f 以 NB 为例, 整个测试阶段可描述为算法 2.

Algorithm 2 Bayesian classification learning framework based on bias–variance trade-off (test phase)

Input: Test instance $\boldsymbol{x}$, Bayesian classification model f, regression model set $\boldsymbol{H}$.
Output: The predicted class label of $\boldsymbol{x}$.

```
Construct the instance x̂ = (x;1) as the input of the regression models;
for k from 1 to n_c do
    Construct P̂(c_k|x) of Bayesian classification model f by Eq. (23);
    Construct h_k(x̂) of regression model h_k by Eq. (33);
    Store P̂(c_k|x) and h_k(x̂) into P̂_x and L_x̂, respectively;
end for
Classify x by Eq. (34);
return c(x).
```

4 实验与结果

为了评估所提学习框架是否有效提高贝叶斯分类模型的泛化性能, 以及是否可以使模型同时得到较低的偏差与方差, 本文设计了一组实验来比较贝叶斯分类模型的原始分类性能与在本文所提学习框架下的分类性能. 在第 4.1 小节中, 首先介绍了详细的实验设置和实验数据; 在第 4.2 小节中, 通过比较使用学习框架改进前后模型的分类精度和偏差、方差变化, 来验证学习框架的有效性; 在第 4.3 小节中, 比较了学习框架对分类精度、迭代次数和训练时间的影响, 探讨了学习框架的潜在优势; 在第 4.4 小节中, 分析了经验参数 γ 取不同值时, 对学习框架下模型的分类精度的影响.

4.1 实验设置和实验数据

作为用于实验比较的基础模型, 除 NB 之外, 本文从现有的 5 类改进方法中分别挑选一种经典模型, 包括: TAN, SB, CFWNB, NBTree, DWNB. 这些改进模型已经被证明具有很好的改进效果, 下面是所有这些比较对象的全称和缩写:

- NB: 朴素贝叶斯;
- TAN: 树扩展的朴素贝叶斯 [3];
- SB: 分类精度属性选择的朴素贝叶斯 [8];
- CFWNB: 相关性属性加权的朴素贝叶斯 [19];

表 1 实验使用数据集的标号和名称

Table 1 The indexs and names of datasets used in the experiments

ID	Dataset	ID	Dataset	ID	Dataset	ID	Dataset
1	anneal.ORIG	16	credit-g	31	kr-vs-kp	46	robot-24
2	anneal	17	cylinder-bands	32	labor	47	segment
3	artificial	18	diabetes	33	letter	48	sick
4	audiology	19	ecoli	34	libras	49	sonar
5	autos	20	energy-y1	35	lymph	50	soybean
6	balance-scale	21	energy-y2	36	mfeat-f	51	spectrometer
7	breast-cancer	22	glass	37	monks	52	splice
8	breast-w	23	hayes-roth	38	mushroom	53	steel
9	car	24	heart-c	39	newthyroid	54	texture
10	cardiotocography	25	heart-h	40	optdigits	55	thyroid-disease
11	climate	26	heart-statlog	41	page-blocks	56	vehicle
12	colic.ORIG	27	hepatitis	42	parkinsons	57	vote
13	colic	28	hypothyroid	43	pendigits	58	vowel
14	connectionist	29	ionosphere	44	primary-tumor	59	waveform-5000
15	credit-a	30	iris	45	qar	60	zoo

- NBTree: 朴素贝叶斯树 [20];
- DWNB: 判别加权的朴素贝叶斯 [26].

我们在国际机器学习与数据挖掘实验平台 (Waikato environment for knowledge analysis, WEKA) [32] 上实现了基于 BVT 的贝叶斯分类学习框架, 并在学习框架下重新学习了上述所有模型.

本文的实验使用了 60 个经典的 UCI 标准数据集, 这些数据集代表了广泛的应用领域和数据特征, 表 1 列出了这些数据集在本文中的使用标号和对应名称. 由于实验选择的模型都适用于处理不存在缺失值的名词性属性, 导致部分数据集不可直接被使用, 因此在实验前本文对全部实验数据进行了以下预处理工作, 这些数据预处理工作在相关研究中也得到了应用 [18]: 首先, 使用名词性属性的众数 (mode) 替换了名词性属性的缺失值, 使用数值性属性的均值 (mean) 替换了数值性属性的缺失值; 然后, 按照等宽度的 ten bins 方法将数值性属性离散化为名词性属性. 同时, 当一个属性的属性值个数等于数据集的实例个数时, 这个属性就是实例标号 (ID), 对分类建模没有任何作用. 因此, 本文提前手动删除了 4 个这样的属性: "colic.ORIG" 数据集中的 "Hospital Number" 属性、"parkinsons" 数据集中的 "name" 属性、"splice" 数据集中的 "instance name" 属性、"zoo" 数据集中的 "animal" 属性. 此外, 由于名词性属性值无法直接参与回归模型的计算, 因此在输入回归模型前需要对名词性属性做数值化编码. 由于属性值的下标均为自然数, 且一个属性值下标可以唯一对应一个属性值, 因此本文在实验中首先采用属性值下标对名词性属性做数值化编码, 然后再将其作为回归模型的输入.

4.2 实验结果与分析

本文采用分类精度评估模型的泛化性能. 表 2 给出了分类精度的详细比较结果. 表中所有的分类精度都是通过十次十折交叉验证获得的平均值. 表中 ORI 列表示模型的原始结果, BVT 列表示模型在本文所提学习框架下的结果. 根据显著性水平为 $p = 0.05$ 的成对双尾 t 检验, 表中 • 表示在本文所提学习框架下, 相应模型在相应数据集上的分类精度显著优于原始分类精度, ◦ 则相反. 作为各模型相对

表 2　本文所提框架使用前后各模型的分类精度比较结果 (%)

Table 2　Classification accuracy comparisons of the models before and after using the proposed framework (%)

ID	NB		TAN		SB		CFWNB		NBTree		DWNB	
	ORI	BVT	ORI	BVT	ORI	BVT	ORI	BVT	ORI	BVT	ORI	BVT
1	88.16	89.33	90.49	89.44	89.68	89.90	90.04	85.09 ○	91.27	92.08	89.50	90.48
2	94.32	95.91 ●	96.73	96.24	96.94	96.52	96.08	96.06	98.40	98.49	98.50	98.36
3	36.49	37.86 ●	58.01	57.54	36.45	37.80 ●	35.41	37.10 ●	64.13	64.26	37.07	38.61 ●
4	71.40	78.79 ●	65.35	74.87 ●	74.06	81.74 ●	70.10	78.87 ●	76.66	81.67 ●	81.48	84.17
5	63.97	65.83	72.54	74.34	68.69	68.52	65.25	66.32	74.75	74.91	75.10	75.70
6	91.44	90.02 ○	86.14	88.93 ●	91.44	90.02 ○	90.34	88.79	91.44	90.02 ○	91.76	89.80 ○
7	72.94	72.81	69.53	70.34	72.53	72.18	72.88	71.70	71.66	71.59	70.22	70.62
8	97.30	97.28	95.45	96.42	96.58	96.67	97.24	97.41	97.23	97.23	96.45	96.38
9	85.46	82.44 ○	93.93	89.14 ○	70.02	74.01 ●	76.36	79.57 ●	94.43	92.82 ○	90.90	84.48 ○
10	84.15	86.56 ●	89.40	89.76	88.70	88.88	86.83	88.31 ●	91.40	91.59	90.94	90.90
11	87.50	89.98 ●	85.00	89.74 ●	91.48	91.48	89.59	91.31 ●	86.67	89.67 ●	87.93	90.04 ●
12	74.21	76.22	67.71	71.70 ●	74.83	78.37 ●	74.50	77.72	74.83	76.76	76.57	77.69
13	78.86	82.75 ●	80.11	83.20 ●	83.37	84.10	82.44	84.32	82.50	83.72	82.26	84.92
14	74.17	75.36	81.29	82.44	74.09	75.13	74.01	74.41	79.60	80.40	83.97	84.28
15	84.74	86.10	84.10	86.14	85.36	85.36	85.96	85.86	84.86	85.71	85.33	86.07
16	75.93	76.53	74.88	76.16	74.76	76.27	76.16	76.11	75.54	76.16	75.58	76.38
17	75.52	77.06	66.41	77.06 ●	70.00	78.20 ●	77.46	79.00	73.81	75.89 ●	80.93	82.26
18	75.68	76.51	76.31	77.37	76.00	76.45	76.72	76.77	75.28	76.11	76.03	76.29
19	81.61	84.64 ●	78.07	82.88 ●	81.96	85.05 ●	77.48	84.46 ●	81.85	84.43 ●	83.71	85.23
20	45.63	46.19	61.06	60.01	48.86	48.21	45.49	44.35	67.38	67.46	51.04	49.50
21	46.70	47.19	49.93	50.12	50.74	50.05	48.37	46.63 ○	55.62	55.61	49.55	49.62
22	57.69	60.62	58.69	59.67	56.33	57.96	56.84	60.44	58.00	59.12	60.89	63.08
23	82.63	78.19	71.19	73.13	84.13	79.44	74.56	64.06 ○	82.50	78.13	84.44	82.19
24	83.44	83.57	79.70	82.41 ●	81.12	82.55	84.59	84.02	81.10	82.21	81.92	82.41
25	83.64	83.99	81.27	83.01	80.19	82.43	83.82	84.23	82.46	83.00	83.40	84.43
26	83.78	85.00	79.48	82.33	80.85	83.89	84.78	84.93	82.26	83.52	82.70	84.48
27	84.06	84.64	83.00	84.86	82.51	84.65	85.03	85.10	82.90	83.62	84.15	85.57
28	92.79	93.60 ●	93.36	93.61	93.46	93.49	93.49	93.66	93.05	93.60 ●	93.33	93.49
29	90.86	91.23	91.34	90.92	91.25	91.49	91.48	92.06	89.18	90.24	91.54	92.40
30	94.33	94.60	94.27	93.87	96.67	95.20	95.53	94.33	95.27	94.67	95.00	95.33
31	87.79	94.26 ●	92.88	95.39 ●	94.34	94.55	93.59	94.62 ●	97.81	98.01	94.71	94.83
32	96.70	96.87	89.00	90.83	82.63	88.70	91.97	94.27	95.60	96.13	95.83	96.33
33	70.09	70.81 ●	82.69	82.93 ●	70.71	71.49 ●	70.87	72.08 ●	83.49	83.79 ●	75.40	76.18 ●
34	64.53	65.44	68.17	69.17	60.50	65.72 ●	64.22	65.47	60.69	62.92 ●	68.50	69.11
35	85.97	86.82	83.69	85.41	80.24	83.17	84.76	85.34	82.21	84.50	83.30	85.10
36	76.64	77.49 ●	77.97	78.68 ●	77.93	79.49 ●	78.26	79.06 ●	76.56	77.40 ●	77.14	78.43 ●
37	74.64	72.41	100.0	99.95	74.64	73.81	74.64	73.45	100.0	99.93	74.64	70.01 ○
38	95.52	95.24	99.99	99.38 ○	99.70	98.10 ○	98.93	98.14 ○	100.0	99.41 ○	99.85	98.78 ○
39	91.72	91.68	89.67	90.00	91.58	91.16	90.97	91.08	91.58	91.63	95.07	93.89
40	92.25	92.63 ●	94.78	95.07 ●	92.07	92.97 ●	92.68	93.43 ●	92.25	92.63 ●	94.26	94.58 ●
41	92.31	92.48	93.14	93.17	92.47	92.63	92.36	92.65	93.52	93.37	92.96	92.43
42	76.13	79.56	89.41	89.83	86.94	86.64	81.99	83.73	86.70	88.54	91.53	89.84
43	87.07	87.64 ●	95.67	95.79	87.70	88.29 ●	87.49	88.37 ●	95.78	95.98 ●	92.29	92.88 ●
44	47.20	49.61	44.77	46.87	44.49	48.38 ●	45.99	47.62	45.84	48.53	46.58	49.41
45	79.87	82.88 ●	83.82	85.44 ●	81.34	84.69 ●	80.08	84.11 ●	83.36	84.50 ●	85.19	86.08
46	80.33	81.78 ●	87.97	88.67 ●	84.81	85.03	83.80	85.25 ●	92.68	92.60	84.16	86.40 ●
47	89.03	90.47 ●	93.91	94.07	90.65	91.53 ●	90.27	91.68 ●	92.64	93.23 ●	90.86	91.90 ●
48	96.78	97.15	97.70	97.42	97.51	97.06 ○	97.44	97.21	97.86	97.65	97.30	96.10 ○
49	76.35	80.75	75.34	78.76	69.78	76.69	75.27	79.92	71.40	76.00 ●	77.63	80.85
50	92.20	92.84	94.98	94.70	91.99	93.50 ●	92.68	93.60 ●	92.30	92.78	93.94	93.94
51	47.44	47.18	41.17	41.13	44.52	45.25	47.89	47.74	48.29	48.48	48.17	48.68
52	95.42	95.83 ●	94.95	95.70 ●	94.95	95.14	96.20	96.01	95.42	95.81	95.76	95.91
53	94.89	100.0 ●	87.99	99.98 ●	97.84	100.0 ●	97.23	100.0 ●	98.28	100.0 ●	99.31	100.0 ●
54	79.43	81.98 ●	94.40	94.93 ●	82.74	87.19 ●	79.86	84.02 ●	93.34	93.54 ●	82.00	87.01 ●
55	93.68	93.95 ●	93.48	93.56	93.76	93.67	93.78	93.74	93.68	93.95 ●	93.69	93.66
56	61.03	67.98 ●	73.35	74.46	60.98	70.34 ●	61.52	70.21 ●	68.91	70.19	65.34	70.72 ●
57	90.21	94.64 ●	94.43	95.31	95.59	95.61	91.98	94.90 ●	94.78	95.65	94.66	95.45
58	66.09	67.76 ●	91.89	92.30	68.59	70.03	68.31	70.87 ●	88.01	88.85 ●	79.23	79.67
59	79.97	81.86 ●	80.44	81.96 ●	81.17	82.79 ●	81.19	83.57 ●	81.62	82.57 ●	81.02	82.92 ●
60	94.37	95.75	96.63	97.23	94.04	96.62	94.57	96.15	94.55	95.44	96.05	96.35
Average	79.92	81.28	82.32	83.76	80.15	81.60	80.33	81.35	83.75	84.48	82.64	83.21
W/T/L	–	24/34/2	–	18/40/2	–	19/38/3	–	21/35/4	–	18/39/3	–	12/43/5

性能的总体指标, 在 60 个数据集上的平均分类精度被汇总在表的底部. 表中任意一个模型的 $W/T/L$ 表明: 与它的原始分类精度相比, 模型在本文所提学习框架下的分类精度在 W 个数据集上显著提升, 在 T 个数据集上基本持平, 在 L 个数据集上显著降低.

从表 2 所示的比较结果可以看出, 在本文所提学习框架下的贝叶斯分类模型在大多数情况下能显著提升原始模型的泛化性能、且很少降低原始模型的泛化性能, 这充分验证了本文所提学习框架的有效性. 将这些发现总结如下:

(1) 在全部的 60 个数据集上, NB, TAN, SB, CFWNB, NBtree, DWNB 6 个贝叶斯分类模型的原始平均分类精度分别为: 79.92%, 82.32%, 80.15%, 80.33%, 83.75%, 82.64%, 在本文所提学习框架下的平均分类精度分别为: 81.28%, 83.76%, 81.60%, 81.35%, 84.48%, 83.21%. 改进后的平均分类精度均高于原始平均分类精度, 这表明学习框架可以有效提升各类基础模型的分类精度;

(2) 根据显著性水平为 $p = 0.05$ 的成对双尾 t 检验的显著性统计测试比较结果, 6 个贝叶斯分类模型在本文所提学习框架下的分类精度相比于它们的原始分类精度: NB 为 24 胜 2 负, TAN 为 18 胜 2 负, SB 为 19 胜 3 负, CFWNB 为 21 胜 4 负, NBTree 为 18 胜 3 负, DWNB 为 12 胜 5 负. 这表明在学习框架下模型的分类精度在绝大多数分类场景可以显著优于或基本持平于相应的基础模型, 极少显著差于基础模型.

为了进一步观察本文所提学习框架的 BVT 效果, 统计了学习框架使用先后各模型的偏差与方差变化. 本文中偏差与方差的估计与文献 [33] 保持一致. 在 60 个数据集上, 各贝叶斯分类模型原始的和相应的在学习框架下的平均偏差与平均方差分别汇总于图 4 和 5, 改进前后的平均值分别用 ORI 和 BVT 标识. 从图中所示的统计结果可以看出, 本文所提学习框架可以同时有效降低各类贝叶斯分类模型的偏差和方差, 学习框架不仅具有有效性, 也同样遵循了偏差 – 方差权衡原则. 将这些发现分别总结如下:

(1) 在全部的 60 个数据集上, NB, TAN, SB, CFWNB, NBtree, DWNB 6 个贝叶斯分类模型的原始平均偏差分别为: 0.1715, 0.1351, 0.1486, 0.1700, 0.1106, 0.1408, 经本文所提学习框架改进后的平均偏差分别为: 0.1590, 0.1281, 0.1454, 0.1592, 0.1095, 0.1383. 改进后的平均偏差均低于原始平均偏差, 这表明学习框架可以有效降低各类贝叶斯分类模型的偏差;

(2) 在全部的 60 个数据集上, NB, TAN, SB, CFWNB, NBtree, DWNB 6 个贝叶斯分类模型的原始平均方差分别为: 0.0401, 0.0578, 0.0621, 0.0377, 0.0658, 0.0450, 经本文所提学习框架改进后的平均方差分别为: 0.0393, 0.0490, 0.0500, 0.0373, 0.0586, 0.0418. 改进后的平均方差均低于原始平均方差, 这表明学习框架可以有效降低各类贝叶斯分类模型的方差.

4.3 讨论与拓展

上述实验已经通过比较分类精度的变化验证了本文所提学习框架的有效性, 同时通过统计偏差和方差的变化证明了学习框架同样遵循了偏差 – 方差权衡原则. 本小节通过一个补充实验, 讨论学习框架对分类精度、迭代次数和训练时间的影响, 分析学习框架的潜在优势, 进一步拓展学习框架的应用场景. 为此, 本小节实验中的基础模型选择具有迭代过程的 DWNB, 同时为了便于观察模型的收敛过程, 实验数据采用 60 个数据集中最大的 “letter” 数据集. 实验中 DWNB 的迭代次数从 5 轮开始, 到 40 轮结束, 间隔 5 轮观测一次, 其余实验设置与 4.2 小节的实验保持一致.

图 6(a) 所示为分类精度随迭代次数的变化曲线, 用于观测模型的分类精度随迭代次数的变化. 从图 6(a) 可以看出, 迭代次数相同时使用学习框架可以获得更高的分类精度, 再次验证了学习框架的有效性. 同时, 经学习框架改进后 DWNB 的分类精度走势和原始 DWNB 的分类精度走势基本相同, 这

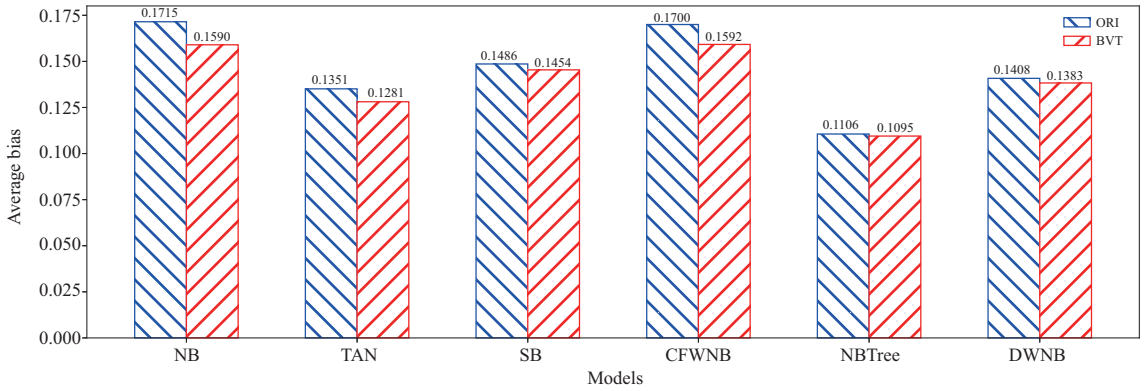

图 4　(网络版彩图) 本文所提框架使用前后各模型的平均偏差比较图

Figure 4　(Color online) The average bias comparison chart of models before and after using the proposed framework

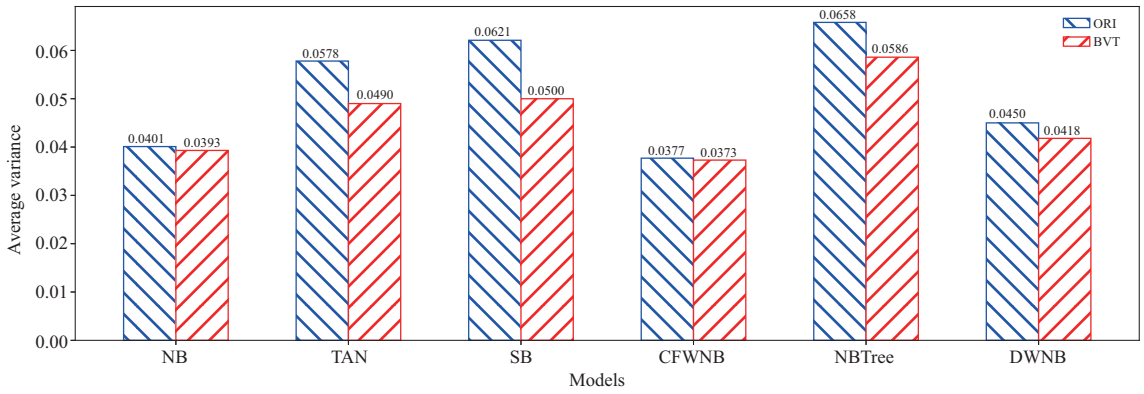

图 5　(网络版彩图) 本文所提框架使用前后各模型的平均方差比较图

Figure 5　(Color online) The average variance comparison chart of models before and after using the proposed framework

表明 DWNB 在学习框架下的分类性能依赖于其原始分类性能; 图 6(b) 所示为训练时间随迭代次数的变化曲线, 由于 DWNB 和本文所提学习框架的时间消耗主要在训练阶段, 因此本文通过训练时间变化来观测模型的时间成本变化. 从图 6(b) 可以看出, 经学习框架改进后 DWNB 的训练时间和原始训练时间相差甚微, 这表明本文所提学习框架具有高效性, 学习框架的有效性并不依赖于消耗高额的时间成本.

结合图 6(a) 和 (b), 在迭代次数为 10 轮时, 经本文所提学习框架改进后 DWNB 的分类精度已经显著高于原始 DWNB 的最终收敛结果, 并且训练时间显著低于原始 DWNB 的收敛所需时间, 这表明优化迭代模型在学习框架下有望通过更短的学习时间, 获得不低于原始模型收敛后的分类结果. 因此可知, 以 DWNB 为代表的优化迭代模型, 收敛速度随着迭代次数的增加而变缓, 此时提升少许的分类性能需要增加高额的迭代次数. 本文所提学习框架有望以较低的迭代次数, 使得优化迭代模型更快达到相同或者更好的分类性能.

4.4　参数分析

本文所提学习框架包含了一个经验参数 γ, 上述实验中 γ 均默认设置为 1. 本小节尝试通过实验, 探讨当 γ 取不同值时, 对学习框架下模型分类精度的影响. 为此, 本节实验中的基础模型、实验数据和

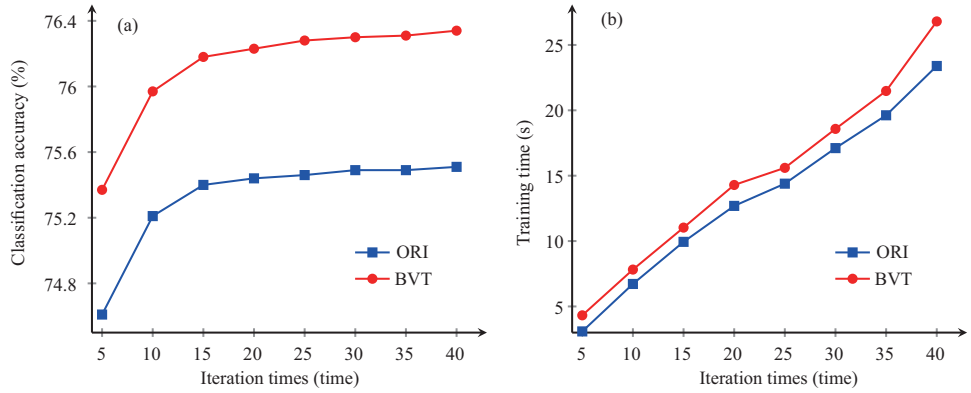

图 6 (网络版彩图) 分类精度、训练时间与迭代次数的关系示意图

Figure 6 (Color online) The diagrams of the relationship between (a) classification accuracy with iteration times and (b) training time with iteration times

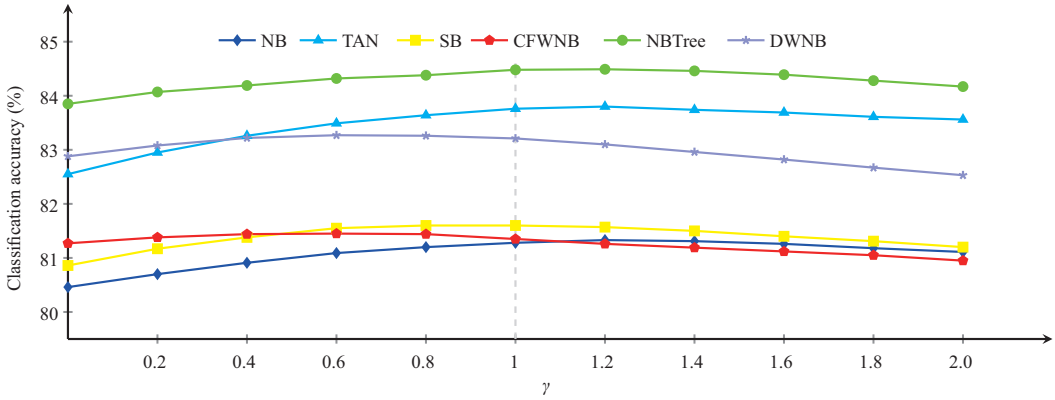

图 7 (网络版彩图) 分类精度与 γ 的关系示意图

Figure 7 (Color online) The diagrams of the relationship between classification accuracy and γ

实验方法仍与第 4.2 小节保持一致, 学习框架的参数 γ 在区间 [0, 2] 内以 0.2 为间隔进行取值. 最终, 通过汇总不同 γ 取值下基础模型在学习框架下的平均分类精度, 得到如图 7 所示的分类精度曲线图. 图 7 中横坐标为 γ 的不同取值, 纵坐标为学习框架下各基础模型在全部 60 个数据集上的平均分类精度.

依据图 7 可知, 随着 γ 从 0 开始逐渐增大, 各基础模型在学习框架下的平均分类精度均先增大后减小, 这表明学习框架的有效性受 γ 取值的影响. 综合各曲线图走势, 当 γ 取值为 1 时, 各模型均可取得较好的平均分类精度, 因此本文其余章节中 γ 的默认取值为 1.

5 总结与展望

本文以 BVT 作为切入点, 总结了改进 NB 的 5 类方法, 理论分析了在贝叶斯分类学习中做 BVT 的可行性, 探讨了保证可行性的关键因素. 根据关键因素, 本文构建了回归任务来学习贝叶斯分类模型的后验概率损失, 从而调控关键因素的变化. 基于上述工作, 本文提出了一种基于 BVT 的贝叶斯分类学习框架, 成功通过 BVT 原则提升了 NB 及其改进模型的泛化性能. 本文通过丰富的比较实验验证

了本文所提学习框架的有效性和高效性.

为了保证本文所提学习框架的简洁性, 目前的版本采用最简单的线性回归模型来拟合贝叶斯分类模型的后验概率损失. 同时, 为了简化回归任务, 只拟合了模型在实例真实类别上的后验概率损失. 在未来的工作中, 我们将利用更加复杂的回归模型, 进一步完善目前学习框架中的回归任务. 当然, 更加复杂的回归模型也意味着更复杂的理论推导过程和更高的计算复杂性. 除此之外, 本文希望通过讨论部分的补充实验引导读者将目前学习框架的原理应用于深度学习或者复杂的参数优化模型. 依据本文的实验结果, 当优化迭代模型收敛速度减慢后, 是通过增加高额的迭代次数来提升少许的分类性能, 还是通过学习模型的后验概率损失来提升模型的性能, 是个值得探讨的问题.

参考文献

1 Han J W, Kamber M, Pei J. Data Mining: Concepts and Techniques. 3rd ed. San Francisco: Morgan Kaufmann, 2011
2 Wu X D, Kumar V, Quinlan J R, et al. Top 10 algorithms in data mining. Knowl Inf Syst, 2008, 14: 1–37
3 Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn, 1997, 29: 131–163
4 Webb G I, Boughton J R, Wang Z H. Not so naive Bayes: aggregating one-dependence estimators. Mach Learn, 2005, 58: 5–24
5 Li N, Yu Y, Zhou Z H. Semi-naive exploitation of one-dependence estimators. In: Proceedings of the 9th IEEE International Conference on Data Mining, Miami, 2009. 278–287
6 Jiang L X, Zhang H, Cai Z H. A novel Bayes model: hidden naive Bayes. IEEE Trans Knowl Data Eng, 2009, 21: 1361–1371
7 Wang L M, Liu Y, Mammadov M, et al. Discriminative structure learning of Bayesian network classifiers from training dataset and testing instance. Entropy, 2019, 21: 489
8 Langley P, Sage S. Induction of selective Bayesian classifiers. In: Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence, Seattle, 1994. 399–406
9 Jiang L X, Zhang H, Cai Z H, et al. Evolutional naive Bayes. In: Proceedings of the 1st International Symposium on Intelligence Computation and Applications, 2005. 344–350
10 Ratanamahatana C, Gunopulos D. Feature selection for the naive Bayesian classifier using decision trees. Appl Artif Intell, 2003, 17: 475–487
11 Chen S L, Martínez A M, Webb G I, et al. Sample-based attribute selective An DE for large data. IEEE Trans Knowl Data Eng, 2017, 29: 172–185
12 Jiang L X, Kong G G, Li C Q. Wrapper framework for test-cost-sensitive feature selection. IEEE Trans Syst Man Cybern Syst, 2021, 51: 1747–1756
13 Zhang H, Sheng S L. Learning weighted naive Bayes with accurate ranking. In: Proceedings of the 4th International Conference on Data Mining, Brighton, 2004. 567–570
14 Hall M. A decision tree-based attribute weighting filter for naive Bayes. Knowl Based Syst, 2007, 20: 120–126
15 Zaidi N A, Cerquides J, Carman M J, et al. Alleviating naive Bayes attribute independence assumption by attribute weighting. J Mach Learn Res, 2013, 14: 1947–1988
16 Jiang L X, Zhang L G, Yu L J, et al. Class-specific attribute weighted naive Bayes. Pattern Recogn, 2019, 88: 321–330
17 Lee C H. An information-theoretic filter approach for value weighted classification learning in naive Bayes. Data Knowl Eng, 2018, 113: 116–128
18 Zhang H, Jiang L X, Yu L J. Class-specific attribute value weighting for naive Bayes. Inf Sci, 2020, 508: 260–274
19 Jiang L X, Zhang L G, Li C Q, et al. A correlation-based feature weighting filter for naive Bayes. IEEE Trans Knowl Data Eng, 2019, 31: 201–213
20 Kohavi R. Scaling up the accuracy of naive Bayes classifer: a decision-tree hybrid. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, 1996. 202–207
21 Frank E, Hall M A, Pfahringer B. Locally weighted naive Bayes. In: Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence, Acapulco, 2003. 249–256
22 Jiang L X, Cai Z H, Wang D H. Improving naive Bayes for classification. Int J Comput Appl, 2010, 32: 328–332

23 Zheng Z J, Webb G I. Lazy learning of Bayesian rules. Mach Learn, 2000, 41: 53–84
24 El Hindi K M, Aljulaidan R R, AlSalman H. Lazy fine-tuning algorithms for naïve Bayesian text classification. Appl Soft Comput, 2020, 96: 106652
25 Elkan C. Boosting and Naive Bayesian Learning. Technical Report CS97-557, 1997
26 Jiang L X, Wang D H, Cai Z H. Discriminatively weighted naive Bayes and its application in text classification. Int J Artif Intell Tools, 2012, 21: 1250007
27 Xu W Q, Jiang L X, Yu L J. An attribute value frequency-based instance weighting filter for naive Bayes. J Exp Theor Artif Intell, 2019, 31: 225–236
28 Zhou Z H. Machine Learning. Beijing: Tsinghua University Press, 2016 [周志华. 机器学习. 北京: 清华大学出版社, 2016]
29 Geman S, Bienenstock E, Doursat R. Neural networks and the bias/variance dilemma. Neural Comput, 1992, 4: 1–58
30 Kohavi R, Wolpert D H. Bias plus variance decomposition for zero-one loss functions. In: Proceedings of the 13th International Conference on Machine Learning, Bari, 1996. 275–283
31 Lagrange J L. Mecanique Analytique. Paris: Mallet-Bachelier, 1853
32 Witten I H, Frank E, Hall M A. Data Mining: Practical Machine Learning Tools and Techniques. 3rd ed. San Francisco: Morgan Kaufmann, 2011
33 Webb G I. MultiBoosting: a technique for combining boosting and wagging. Mach Learn, 2000, 40: 159–196

Bayesian classification learning framework based on bias–variance trade-off

Wenjun ZHANG[1], Liangxiao JIANG[1,2*], Huan ZHANG[1] & Chengyu HU[1]

1. *School of Computer Science, China University of Geosciences, Wuhan* 430074*, China*;
2. *Key Laboratory of Artificial Intelligence, Ministry of Education, Shanghai* 200240*, China*
* Corresponding author. E-mail: ljiang@cug.edu.cn

Abstract Due to its simplicity, efficiency, and efficacy, naive Bayes (NB) continues to be one of the top ten data mining algorithms. However, its attribute-conditional independence assumption rarely holds true in real-world applications. In order to alleviate the need for this assumption, scholars have proposed five types of improved approaches, including structure extension, attribute selection, attribute weighting, instance selection, and instance weighting. Although these existing improved approaches reduce the bias of the model to some extent, they also increase the variance of the model and thus limit the generalization of the model. The bias–variance trade-off is one of the core principles of machine learning, which requires a model to have low bias and variance at the same time. This paper is focused on how to introduce the bias–variance trade-off into Bayesian classification learning, obtain lower bias and variance at the same time, and improve the generalization of the model. Therefore, we first theoretically analyze the feasibility of introducing the bias–variance trade-off into Bayesian classification learning and determine the key factor that ensures feasibility. Then, we learn the posterior probability losses of the Bayesian classification models by constructing regression tasks to control the change in the key factor. Finally, we propose a Bayesian classification learning framework based on the bias–variance trade-off and re-implement NB and its various improved models under the proposed learning framework. The experimental results on a large number of classical UCI standard datasets show that under the proposed learning framework, the classification performance of existing various state-of-the-art Bayesian classification models is significantly better than their original performance.

Keywords naive Bayes, attribute conditional independence assumption, bias–variance trade-off, posterior probability loss, learning framework