DOI：10.7652/xjtuxb201710023

针对机器学习中残缺数据的近似补全方法

曹卫权，褚衍杰，李显
（盲信号处理重点实验室，610041，成都）

摘要：针对机器学习中含残缺项的数据不能被有效利用，导致分类和回归准确率不高的问题，提出了一种近似补全方法——*k*-ANNO方法。给定残缺的数据样本，该方法首先通过离线构建的图结构来近似搜索与该样本最接近的*k*个近邻顶点，然后采用快速二次规划估计各近邻的最优权重，最后基于权重值来补全样本中的残缺项，用户可以根据实际需求在补全效率与准确性之间折中。*k*-ANNO方法较好地解决了机器学习中普遍存在的数据残缺问题，有效抑制了数据残缺对分类和回归精度的干扰。利用多份公开数据集评估了*k*-ANNO方法的补全效果，结果表明：当加速比在2～10之间时，*k*-ANNO方法的分类错误率比已有的均值补全、*C*均值补全、自组织映射补全方法低1%～4%，回归均方根误差比已有方法低约0.5～2.0；当样本规模为4 000时，在不同加速比参数下，*k*-ANNO方法的计算效率比朴素*k*近邻方法高约35%～320%。

关键词：机器学习；残缺项；二次规划；补全方法
中图分类号：TP391 **文献标志码**：A **文章编号**：0253-987X(2017)10-0142-07

Approximate Imputation Method for Missing Data in Machine Learning

CAO Weiquan, CHU Yanjie, LI Xian
(National Key Laboratory of Science and Technology on Blind Signal Processing, 610041, China)

Abstract: An approximate imputation method called *k*-ANNO is proposed to handle the problems of missing data in machine learning field given a missing sample. The proposed method begins by constructing an offline graph to approximately search nearest neighbors of the partially missing sample efficiently. Then a fast quadratic programming algorithm is utilized to determine the optimal weight for each neighbor. Finally, unmissed parts of the neighbors are used to impute the missing attributes by the estimated weights. Users get the freedom to weigh up between efficiency and imputation accuracy. The widespread data missing problems are well solved in this paper and *k*-ANNO is able to depress the impact of missing data effectively. Experiments on various well known datasets show that when the speedup rate parameters are between 2 and 10, *k*-ANNO method outperforms existing ones such as mean imputation or *C*-Means imputation etc. and the classification error and the regression error are 1% to 4% and 0.5-2.0 lower than those, respectively. Meanwhile, *k*-ANNO outperforms naïve *k*-NN imputation with a faster efficiency increased by 35%-320% faster.

Keywords: machine learning; missing attributes; quadratic programming; imputation method

---

收稿日期：2017-02-28。 作者简介：曹卫权(1989—)，男，博士，助理工程师；褚衍杰(通信作者)，男，博士，工程师。 基金项目：国家自然科学基金资助项目(U1536105)。
网络出版时间：2017-08-31 网络出版地址：http://www.cnki.net/kcms/detail/61.1069.T.20170831.1710.002.html

机器学习是一种挖掘数据中潜在规律的有效方法，能够对研究对象的未知类别或数值进行预测，因而被广泛应用在计算机视觉、智能家居[1]、问卷分析[2]、基因组分析[3]等领域。当机器学习方法的输入数据包含残缺项时，许多机器学习方法的预测精度会急剧下降，导致漏检、虚警甚至模型训练失败等问题。在现实世界中，数据残缺是非常常见的现象，例如传感器短时失效引起相应参数的采集中断[1]、问卷的被调查者可能拒绝回答部分问题[2]、基因序列采集不完整[3]等。因此，必须对数据中的残缺项进行适当的处理，以提升机器学习系统对实际非理想数据的适应能力。

针对残缺数据主要有 2 类处理方式，即设计专用的学习方法或对残缺数据进行补全预处理。常见的专用学习方法包括决策树、随机森林[3-4]等。这类方法能够容忍数据中的残缺项，从而对每个数据样本的未知类别进行预测，但仍存在很多缺陷。首先，专用学习方法限制了数据分析者对学习方法的选择空间，进而影响数据预测系统的计算效率和预测精度；其次，这类方法常用于解决机器学习的分类子问题，而不能解决回归子问题。常见的数据补全方法包括均值补全[5]、高斯混合模型（Gaussian Mixture Model，GMM）补全[6]、C 均值（C-Means）补全[7]、k 近邻（k Nearest Neighbors，k-NN）补全[8-10]等。基于这类方法对数据预处理后，分析者可以使用各种机器学习方法进行数据分类或回归，因此其适用范围比专用学习方法更广。均值补全[5]、GMM 补全[6]、C 均值补全[7]等方法通常假设数据的残缺项在统计上呈单峰或多峰高斯分布，这种强模型假设导致其补全精度通常低于 k-NN 补全方法[8-10]。k-NN 方法的主要缺陷则在于搜索 k-NN 步骤的复杂度较高，当数据体量较大时，其实用性较差。

针对上述问题，本文提出一种基于近似 k-NN 的数据补全方法，并设计了针对近邻点权重的最优估计方法来进一步提升补全精度。利用开源数据集验证了的实验结果表明，本文方法弥补了 k-NN 补全方法的效率缺陷，同时又保留了该方法补全精度高的优势，有效地解决了机器学习中常见的输入数据残缺问题。

1　k-NN 补全方法与搜索效率

1.1　k-NN 补全方法

为了下文描述方便，对于数据集 X，定义其中不包含缺失属性的数据集合为 X_c，其余为 X_m。给定 $\boldsymbol{x}\in X_m$，$\boldsymbol{x}$ 的缺失与非缺失属性的列索引集合分别为 A_m 与 A_c。

朴素的 k-NN 方法[8]搜索 $\boldsymbol{x}$ 在集合 X_c 中的 k 个最近邻，并按照一定的策略为各近邻赋予权重，进而估计 $\boldsymbol{x}$ 的缺失属性 A_m。文献[9]提出一种在查询点附近按象限搜索近邻点的象限均衡策略，以解决 k 近邻分布不均匀的问题。文献[10]基于自组织映射（self organizing map，SOM）和已检索到的 k 近邻来预测缺失属性，从而避免了 k 近邻权重的估计问题。

然而，上述几种 k-NN 补全方法的近邻点搜索效率均为 $O(|X_c|\log(k))$，当数据量较大时将出现效率瓶颈。

1.2　近似 k-NN 搜索方法

为提升 k-NN 的搜索效率，研究者提出了多种近似 k-NN 搜索方法[11-13]，这些方法不保证搜索到的点为精确的 k 近邻，一般为距离 $\boldsymbol{x}$ 较近的 k 条数据。已知的近似 k-NN 搜索方法可以分为以下 3 类。

（1）基于划分的方法[11]。该方法以树结构实现近似 k-NN 的快速搜索。

（2）基于局部敏感哈希的方法[12]。该方法利用相似数据的哈希函数比较接近的特点，构建多个哈希函数，来快速定位 $\boldsymbol{x}$ 的最近邻。

（3）基于 k-NN 图的方法[13]。首先构建有向图 $G_\rho(V,E)$，该图的基本构建原则为：顶点集合 $V\equiv X_c$，即每个顶点元素都是一条未缺失数据；任意顶点对 $(\boldsymbol{u},\boldsymbol{v})\in E$ 当且仅当顶点 $\boldsymbol{u}$ 为 $\boldsymbol{v}$ 的 ρ 近邻[13]。文献[13]给出了一种高效构建方法：在 k-NN 图基础上，通过在图中按照启发式策略沿边游走来查找与 $\boldsymbol{x}$ 最相似的 k 个顶点。

考虑到基于划分的方法和基于局部敏感哈希的方法均要求 $\boldsymbol{x}$ 的属性为完整的，不适用于查询点含缺失项的应用场景，因此本文通过对基于 k-NN 图的方法进行适当的修正，以满足近邻查询的需求。

2　近似 k-NN 高效补全方法

2.1　算法结构

基于近似 k-NN 的属性补全算法的工作流程如图 1 所示，点划线框出的部分为属性补全的作用范围。该算法基于离线的完整数据 X_c 构建 k-NN 图 $G(X_c)$，构建方法详见文献[13]。对于输入的缺失数据 $\boldsymbol{x}$，根据其已知属性集 A_c 和 k-NN 图，利用 2.2 节方法快速检索得到 $\boldsymbol{x}$ 的近似 k 近邻，随后利用 2.3 节方法获取各近邻点的最优权重 $\boldsymbol{w}_{k\times1}$，通过对

各近邻点线性组合得出各缺失属性$\{x_i \mid i \in A_m\}$的填补值。

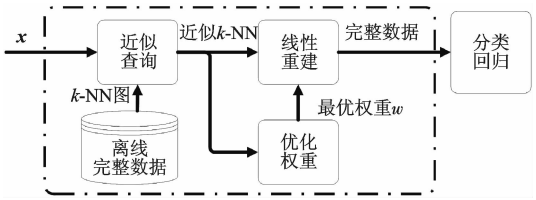


图 1 基于近似 k-NN 的属性补全算法流程

2.2 针对任意属性子集的近似 k-NN 搜索

在属性补全的背景下，仅已知 $\boldsymbol{x}$ 的部分属性 A_c，因此应定义图 G 中任意顶点 $\boldsymbol{v}$ 与 $\boldsymbol{x}$ 的距离如下

$$D(\boldsymbol{v},\boldsymbol{x}) = \sum_{i \in A_m} \| v_i - x_i \| \tag{1}$$

式中：v_i、x_i 分别为 $\boldsymbol{v}$、$\boldsymbol{x}$ 的第 i 项属性分量。式(1)表明，$D(\boldsymbol{v},\boldsymbol{x})$随 $\boldsymbol{x}$ 的缺失状况而变化。

在本节的搜索算法中，需维护近似 k-NN 集合 S_k，该集合存储已搜索到距 $\boldsymbol{x}$ 最近的 k 个点。点 $\boldsymbol{v}$ 可以成功插入 S_k，当且仅当

$$D(\boldsymbol{v},\boldsymbol{x}) < \max_{\boldsymbol{u} \in S_k} D(\boldsymbol{u},\boldsymbol{x}) \tag{2}$$

采用堆排序法，S_k 的插入效率为 $O(\log k)$。基于已建立的图 G，按照如下步骤设计贪心搜索算法：

(1)初始化迭代计数值 $m=0$，在图 G 中随机选择起点 $\boldsymbol{v}^{(m)}$（$\boldsymbol{v}^{(m)}$ 表示 $\boldsymbol{v}$ 第 m 轮迭代的取值。），若 $\boldsymbol{v}^{(m)}$ 满足式(2)，则将其插入 S_k；

(2)将 $\boldsymbol{v}^{(m)}$ 所有满足式(2)的邻接顶点插入 S_k；

(3)获取邻接顶点中距 $\boldsymbol{x}$ 最近的顶点 $\boldsymbol{v}^*$；

(4)若达到最大搜索次数，则终止计算，S_k 为近似 k-NN；若 $D(\boldsymbol{v}^*,\boldsymbol{x}) \leqslant D(\boldsymbol{v}^{(m)},\boldsymbol{x})$，则令 $\boldsymbol{v}^{(m+1)} = \boldsymbol{v}^*$，返回步骤 2，否则返回步骤 1 重新开始搜索。

定义 k-NN 图 G 中顶点数量与该算法最大搜索次数的比值为加速比，用 r_s 表示。一般地，加速比越小，则搜索效率越低[13]。

2.3 近邻点权重

本文借鉴局部线性重建思想[8]，通过优化式(3)所示的目标函数，获取各个近邻点的最优权重 $\boldsymbol{w}$

$$\begin{aligned} &\min_{\boldsymbol{w}} f(\boldsymbol{w}) = \frac{1}{2} \| \boldsymbol{P}\boldsymbol{w} - \boldsymbol{x} \|^2 \\ &\text{s.t.} \quad \boldsymbol{w} \geqslant 0, \ | \boldsymbol{w} | = 1 \end{aligned} \tag{3}$$

式中：$\boldsymbol{w}$ 的维度为 $k \times 1$；$\boldsymbol{x}_{d\times 1}$ 为查询点的已知属性 A_c 构成的列向量，A_c 的维数为 d；矩阵 $\boldsymbol{P}_{d\times k}$ 的各列为每个近似 k 近邻的 A_c 属性。

上述优化问题需要针对每条缺失数据求解，因此求解效率尤为重要，而利用经典的内点法、有效集法等直接求解该问题的效率很低。注意到该问题的不等式约束为简单的上下界约束，因此可以利用序列最小优化（sequential minimal optimization，SMO）[14]实现快速求解。

2.3.1 Karush-Kuhn-Tucker 条件 设优化问题的对偶变量为 $\boldsymbol{\alpha}_{k\times 1}$ 和 $\boldsymbol{\beta}_{1\times 1}$，分别对应不等式约束和等式约束，则该问题对应的拉格朗日函数为

$$\begin{aligned} &L(\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\beta}) = \frac{1}{2} \| \boldsymbol{P}\boldsymbol{w} - \boldsymbol{x} \|^2 - \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{w} - \boldsymbol{\beta}(\boldsymbol{1}^{\mathrm{T}}\boldsymbol{w} - \boldsymbol{1}) \\ &\text{s.t.} \quad \boldsymbol{\alpha} \geqslant 0 \end{aligned} \tag{4}$$

对应的 Karush-Kuhn-Tucker(KKT)条件为

$$\left.\begin{aligned} &\partial L/\partial \boldsymbol{w} = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{w} - \boldsymbol{P}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{\alpha} - \boldsymbol{1}\boldsymbol{\beta} = 0 \\ &\partial L/\partial \boldsymbol{\beta} = -\boldsymbol{1}^{\mathrm{T}}\boldsymbol{w} + \boldsymbol{1} = 0 \\ &\alpha_i w_i = 0, \quad i = 1,2,\cdots,k \end{aligned}\right\} \tag{5}$$

采用数值解法[15]求得满足式(5)所定义 KKT 条件的解，即为问题(3)的最优解。

2.3.2 最优权重快速求解 在本小节描述的算法中，令 $\boldsymbol{v}^{(m)}$ 表示 $\boldsymbol{v}$ 第 m 次迭代的结果，v_i 表示向量 $\boldsymbol{v}$ 的第 i 个元素；对任意矩阵 $\boldsymbol{M}$，令 $\boldsymbol{M}^{(m)}$ 表示 $\boldsymbol{M}$ 的第 m 次迭代结果，$\boldsymbol{M}_{i\rightarrow}$、$\boldsymbol{M}_{i\downarrow}$ 分别表示矩阵的第 i 个行、列向量。

利用 SMO 算法的思想，一次迭代仅优化 $\boldsymbol{w}$ 的 2 维分量，同时结合式(3)问题的 KKT 条件，设计最优权重的快速求解算法。算法步骤如下：

(1)查询任一破坏式(5)KKT 条件的分量 w_i，并随机选取另一分量 w_j，若未找到 w_i 则优化终止；

(2)限制 $w_i^{(m+1)} + w_j^{(m+1)} = w_i^{(m)} + w_j^{(m)} = C$，限制其他 $w_{l\neq i,j}^{(m+1)}$ 保持不变；

(3)采用式(7)所示的解析法[14]直接优化函数 $f(w_i^{(m+1)})$；

(4)将最优解 $w_i^{(m+1)}$ 限制在区间$[0,C]$；

(5)参照式(9)、(10)，更新 $\boldsymbol{\beta}$、$\boldsymbol{\alpha}$ 等参数，进入下一轮迭代。

为了保证上述算法可以复现，需要分别在步骤 1 明确如何确定破坏 KKT 条件的 w_i，在步骤 3 明确如何优化函数 $f(w_i^{(m+1)})$，在步骤 5 明确如何更新 $\boldsymbol{\beta}$ 与 $\boldsymbol{\alpha}$。

在式(5)定义的 KKT 条件中，条件 1 通过步骤 5 强制满足，条件 2 通过步骤 2 强制满足。因此，在步骤 1 中，可以通过仅检查 $|\alpha_i w_i| \neq 0$ 来确定破坏条件 3 的分量 w_i。

当按照步骤 2 约束 w_j 及其他分量时，由于除 w_i 和 w_j 外，其他分量的值保持不变，因此目标函数简化为式(6)定义的一元函数

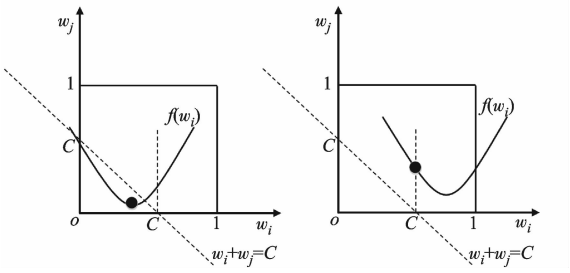
$$2f(\boldsymbol{w}) = \left\| \sum_{l=1}^{k} \boldsymbol{P}_{l\downarrow} w_l - \boldsymbol{x} \right\|^2 = \| (\boldsymbol{P}_{i\downarrow} - \boldsymbol{P}_{j\downarrow}) w_i + \boldsymbol{\eta} \|^2 \quad (6)$$

式中：$\boldsymbol{\eta}_{d\times 1}$为与 w_i 无关的常向量，但需在每步迭代中更新。观察式(6)不难发现，$f(\boldsymbol{w})$实际上是关于 w_i 的二次函数(抛物线函数)。

无约束条件下，最小化上述一元二次函数，可得步骤 3 的最优 w_i 如下

$$w_i^* = -\frac{(\boldsymbol{P}_{i\downarrow} - \boldsymbol{P}_{j\downarrow})^{\mathrm{T}} \eta}{\| \boldsymbol{P}_{i\downarrow} - \boldsymbol{P}_{j\downarrow} \|^2} \quad (7)$$

考虑到步骤 2 对 w_i 和 w_j 的约束，需要将 w_i^* 限制在区间$[0,C]$之内。采用图 2 所示的最优权重分量方法来求解 w_i 的最优值 w_i^*。当 w_i^* 在$[0,C]$之内时，w_i^* 即为最优的 w_i 值；当 w_i^* 的取值位于区间$[0,C]$之外时，距离 w_i^* 最近的区间边界即对应$f(\boldsymbol{w})$在区间约束下的最小值。



(a)最优 w_i 在区间内　　(b)最优 w_i 在区间边界上

图 2　最优权重分量求解原理

式(7)中 $\boldsymbol{\eta}$ 的更新策略如下

$$\boldsymbol{\eta}^{(m+1)} = \boldsymbol{P}\boldsymbol{w}^{(m)} - \boldsymbol{x} - (\boldsymbol{P}_{i\downarrow} - \boldsymbol{P}_{j\downarrow}) w_i^{(m)} \quad (8)$$

对于步骤 5，为保证解满足 KKT 条件 1，需令

$$\boldsymbol{\alpha}^{(m+1)} = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{w}^{(m+1)} - \boldsymbol{P}^{\mathrm{T}}\boldsymbol{x} - \mathbf{1}\boldsymbol{\beta}^{(m+1)} \quad (9)$$

考虑 KKT 条件 3 要求 α_i 与 w_i 至少一项为 0，且 $\alpha_i \geqslant 0$。为使得 $\boldsymbol{\alpha}^{(m+1)}$ 的零项尽可能多，$\boldsymbol{\beta}^{(m+1)}$ 应取值为向量 $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{w}^{(m+1)} - \boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}$ 的最小元素值，如式(10)所示

$$\boldsymbol{\beta}^{(m+1)} = \min(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{w}^{(m+1)} - \boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}) \quad (10)$$

采用上述解法有 2 项显著优点：一是求解速度更快；二是当 $\boldsymbol{P}$ 非列满秩，即 $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}$ 为半正定矩阵时，可以回避内点法的矩阵奇异问题[15]。

2.4　基于近邻点权重的属性补全

基于 2.2 节获取的近似 k 近邻集合 S_k 和 2.3 节获取的最优权重 $\boldsymbol{w}$，确定查询点 $\boldsymbol{x}$ 缺失的数值型属性补全值

$$x_j = \sum_{v\in S_k, l\in[1,k]} w_l v_j, \quad \forall j \in A_{\mathrm{m}} \quad (11)$$

对于非数值型属性，则采用近邻点加权投票的方式补全。

2.5　时间复杂度分析

本文方法的时间复杂度由 2 部分构成：搜索近似 k-NN 与最优权重计算。对于前者，由 2.2 节的贪心算法流程，易知其复杂度为 $O(|X_{\mathrm{c}}|d(\log k)/r_{\mathrm{s}})$；对于后者，观察 2.3.2 节的优化过程，需要预先计算 $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}$ 与 $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}$，时间复杂度为 $O(k^2 d)$。在每次迭代中，步骤 1 的复杂度为 $O(k)$，步骤 2、4 的复杂度为 $O(1)$；对于步骤 3，分析式(7)与式(8)，由于每次迭代仅更新 $\boldsymbol{w}$ 的 2 个分量，因此更新 $\boldsymbol{\eta}$ 的复杂度为 $O(d)$，求取 $\boldsymbol{w}_i^*$ 的复杂度为 $O(d)$；对于步骤 5，更新 $\boldsymbol{\beta}$、$\boldsymbol{\alpha}$ 的复杂度分别为 $O(1)$、$O(k)$、$O(k)$。若设最大迭代次数为 M，则该环节的总复杂度为 $O(k^2 d + M(k+d))$。当 k 相对较大时，算法的瓶颈仅受限于 $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}$ 乘法操作。

综上所述，本文方法的总复杂度为 $O\left(\frac{1}{r_{\mathrm{s}}}|X_{\mathrm{c}}| \cdot d\log k + k^2 d + M(k+d)\right)$，此复杂度的示意图如图 3 所示。图 3 中 T_0 表示优化权重的时间，即 $O(k^2 d + M(k+d))$。

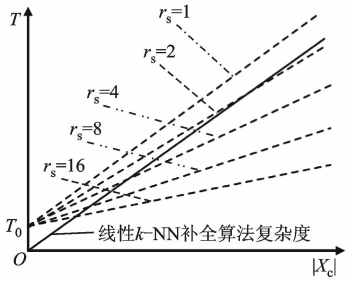


图 3　本文方法的时间复杂度示意图

3　实验分析

3.1　数据准备

分别采用 UCI 和 Torgo 机器学习数据集来测试本文方法在分类与回归 2 项机器学习任务上的补全效果。实验用数据集情况如表 1 所示，其中前 2 项来自 UCI 数据集，后 2 项来自 Torgo 数据集。测试数据同时包含样本数稀疏、丰富的情况，用以模拟不同的应用场景。

基于每项数据集，分别随机选择 5%～50%的条目为缺失数据，其中每条缺失数据的缺失情况从 1 项到一半属性随机不等。每次实验均重复生成 30 次缺失数据，以保证实验结果统计可信。

表 1 实验用数据集简介

数据集名称	条目数	属性数	学习类型
sonar	208	61	分类
satellite	4 435	37	分类
stock	950	10	回归
cpu activity	8 192	22	回归

3.2 面向分类与回归任务的对比实验

分别对比本文方法与 MEI[5]、C-Means[7]、GMM[6]、象限均衡[9]、SOM[10]、朴素 k-NN[8] 等补全算法在分类与回归任务中的补全效果。其中，仅采用 2.2 节近似 k-NN 的补全算法标记为 k-ANN，本文方法标记为 k-ANNO。近似 k-NN 的加速比设置为 2。对于分类学习任务，选择分类错误率作为评价指标，学习工具为支持向量机、k-NN、决策表、C4.5 决策树；对于回归学习任务，选择回归均方根误差(RMSE)作为评价指标，学习工具为 k-NN、决策表、线性回归、M5P 回归。2 种学习任务均采用 10 折交叉方式验证。对各缺失率及补全方法，均取对应 4 种学习工具 30 次重复试验结果的均值作为测试结果。7 种补全算法在不同数据集上的补全效果对比如图 4～图 7 所示。

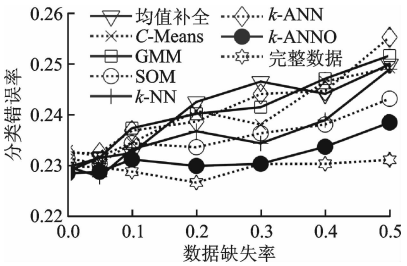


图 4 各种补全方法在 sonar 数据下的分类错误率对比

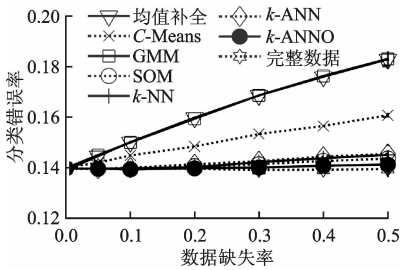


图 5 各种补全方法在 satellite 数据下的分类错误率对比

观察实验结果，MEI 补全[5]作为一种简捷高效的数据补全方法，在缺失数据呈单峰、低方差分布等典型情况下，其补全效果不劣于 C-Means[7]、GMM[6]等方法。相比于朴素 k-NN 方法，象限均衡方法[9]较适用于低维情况。随着数据维数增加，象限数呈指数增长，该方法补全精度较差，见图 7 中 cpu activity 的补全效果。此外，当数据维数过高时(如 sonar、satellite)，该方法无法求解，相应地图 4、图 5 中亦未标出象限均衡法，体现了其局限性。

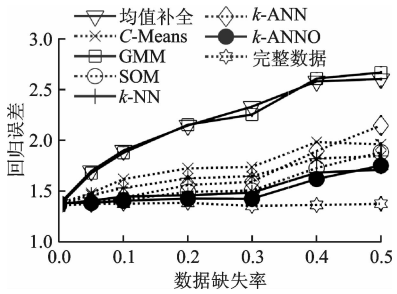


图 6 8 种补全方法在 stock 数据下的回归误差对比

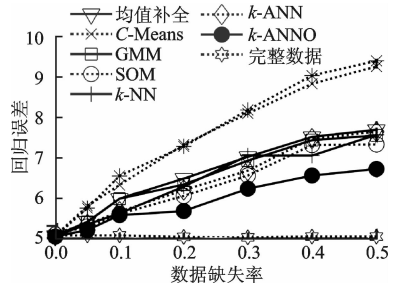


图 7 8 种补全方法在 cpu activity 数据下的回归误差对比

SOM 方法[10]给出了一种基于 k 近邻来预测待补全属性的解决方案，但实验发现，该方法的补全效果在部分数据下(如 stock)并不优于朴素 k-NN 方法。此外，该方法的检索复杂度与朴素 k-NN 相同，补全效率较低。

若仅采用近似 k-NN 进行属性补全，尽管效率有所提升，但补全性能相较精确 k-NN 下降明显，特别是在 sonar 数据下的补全效果接近简单的 MEI 补全。通过估计最优权重，近似 k-NN 算法的补全效果得到显著改善，在各类数据下均优于已有方法。当加速比在 2～10 之间时，基于该方法补全数据，分类错误率比其他方法低 1%～4%，回归均方根误差低约 0.5～2.0。

根据数据各属性之间相关程度的强弱，不同的补全算法在对残缺数据进行补全时，很难全面占优。从图 4～图 7 也可以看出，除去本文方法外，图 4～图 7 中有些 k-ANN 占优，有些 C-Means 占优。本文方法虽然性能提升有限，但却可以在多个数据集下都优于或不劣于已有方法，显示了较广的适用范围。

此外，在同一缺失率下，不同数据条目的缺失对属性补全结果也有一定的影响。以 sonar 数据集为例，分别在缺失率为 0%、10%、30%、50%下开展实

验。对每项缺失率，重复 1 000 次随机抽取对应比率的条目设置为缺失条目，统计各缺失率下基于本文补全方法进行数据分类实验的错误率分布情况，如图 8 所示。当缺失率较低时，数据补全后，多次重复实验的分类性能较为接近，分布图呈“瘦高”状。随着缺失率的增加，对数据进行补全操作，补全效果的不确定性逐渐增大，分布图逐渐呈现“矮胖”状。

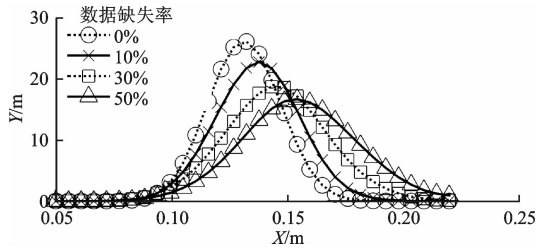


图 8　不同条目缺失下分类错误率分布情况

3.3　近似级别与补全性能

近似 k-NN 的加速比决定了该算法在离线数据中的搜索范围。通过实验测试了不同加速比对近似 k-NN 补全性能的影响，如图 9 所示。从图 9 可见，除本文算法外，朴素 k-NN 算法补全效果比较稳定，因此本小节以该方法作为参考。

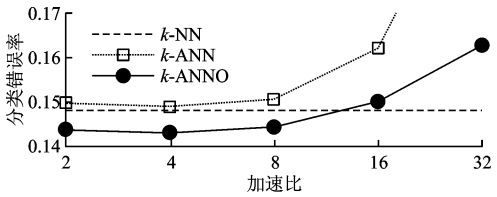


图 9　加速比对近似 k-NN 补全效果的影响

实验采用 satellite 数据集作为测试样本，其余实验流程与 3.2 节一致。随着加速比增加，2 种近似 k-NN 补全算法的效果都有所下降。比较虚线与实线，在不同近似级别下，最优权重始终对近似 k-NN 算法的性能有正增益。

针对 satellite 数据集，本文 k-ANNO 算法在加速比约小于 10 时补全效果优于朴素 k-NN 算法。当加速比大于 10 时，由于近似性的增加，本文方法补全效果逐渐降低，但补全效率随之提升，下面将分析算法的效率问题。

3.4　计算效率

式(12)及图 3 表明，求解最优权重带来了额外计算量，因此其计算效率尤为关键。图 10 对比了本文方法与内点法的优化效果。

图 10a、b 两图分别比较了两种方法的优化时间与目标函数随问题规模 k 的变化情况。本文方法比

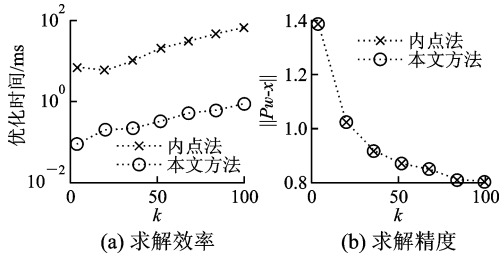


图 10　本文最优权重快速解法与内点法求解效果对比

内点法快大约 2 个数量级，同时在不同规模的优化问题下，两种方法的求解质量一致。

图 11 对比了 k-NN 补全方法与本文方法在不同加速比下，补全一条缺失数据的平均时间。

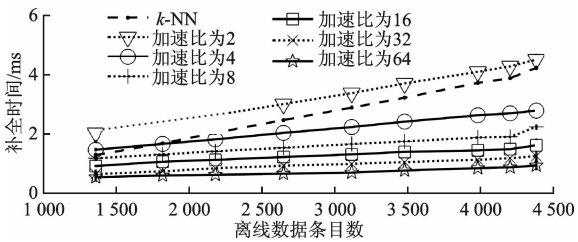


图 11　本文方法与 k-NN 补全方法的时间效率对比

随着数据规模 $|X_c|$ 的增加，近似 k-NN 补全方法的运行效率逐渐优于 k-NN 方法；当样本规模为 4 000 时，依加速比参数的不同，该方法的计算效率比朴素 k 近邻方法高约 35%～320%。若采用传统内点法求解最优权重，则该环节将成为本文方法的效率瓶颈，使得本文算法失去其效率优势。

通过 3.2 节至 3.4 节的实验可以发现：一方面本文方法在加速比为 4～10 之间时，补全性能接近或略优于现有补全算法，且补全效率较高；另一方面，本文方法可以通过加速比参数，灵活调节补全效率与准确性，从而适应更多的应用场景。

4　结　语

本文提出一种基于近似 k-NN 的属性补全方法，并基于快速二次规划来求解各近邻点的最优权重，进一步提升了补全性能。实验结果表明，设定适当的加速比，本文方法的补全效果优于已有方法，且计算效率较精确 k-NN 方法更高。同时，本文通过加速比参数，给出一种灵活的数据补全方案，用户可以根据实际需求在补全效率与准确性之间恰当地折中。下一步的研究内容包括如何针对任意已知属性子集，实现面向超大规模离线数据的近似 k-NN 搜索，拓宽本文方法的适用范围。

参考文献：

[1] 杨雷，李贵鹏，张萍. 无线传感网络数据缺失下的通信优化仿真 [J]. 计算机仿真，2013，30(12)：249-252.
YANG Lei，LI Guipeng，ZHANG Ping. Simulation on communication optimization of wireless sensor networks under missing data [J]. Computer Simulation，2013，30(12)：249-252.

[2] 孟杰，李春林. 基于随机森林模型的分类数据缺失值插补 [J]. 统计与信息论坛，2014(9)：86-90.
MENG Jie，LI Chunlin. Missing data imputation for categorical data based on random forest model [J]. Statistics & Information Forum，2014(9)：86-90.

[3] 吴小姣，李高明，易大莉，等. 基因表达谱的非参缺失森林填补算法研究 [J]. 中国卫生统计，2016，33(6)：1068-1070.
WU Xiaojiao，LI Gaoming，YI Dali，et al. Study on the algorithm of non-parameter deletion forest filling for gene expression profiles [J]. Chinese Journal of Health Statistics，2016，33(6)：1068-1070.

[4] 张晓琴，程誉莹. 基于随机森林模型的成分数据缺失值填补法 [J]. 应用概率统计，2017，33(1)：102-110.
ZHANG Xiaoqin，CHENG Yuying. Imputation of missing values for compositional data based on random forest [J]. Chinese Journal of Applied Probability and Statistics，2017，33(1)：102-110.

[5] FARHANGFAR A，KURGAN L，DY J G，et al. Impact of imputation of missing values on classification error for discrete data [J]. Pattern Recognition，2008，41(12)：3692-3705.

[6] OUYANG M，WELSH W J，GEORGOPOULOS P. Gaussian mixture clustering and imputation of microarray data [J]. Bioinformatics，2004，20(6)：917-923.

[7] DING Y，ROSS A. A comparison of imputation methods for handling missing scores in biometric fusion [J]. Pattern Recognition，2012，45(3)：919-933.

[8] KANG P. Locally linear reconstruction based missing value imputation for supervised learning [J]. Neurocomputing，2013，118(11)：65-78.

[9] 张孙力，杨慧中. 基于改进的K近缺失数据补全 [J]. 计算机与应用化学，2015，32(12)：1499-1503.
ZHANG Sunli，YANG Huizhong. Missing data completion based on an improved *K*-neighbor algorithm [J]. Computers and Applied Chemistry，2015，32(12)：1499-1503.

[10] LIU Z G，PAN Q，DEZERT J，et al. Adaptive imputation of missing values for incomplete pattern classification [J]. Pattern Recognition，2016，52(C)：85-95.

[11] MUJA M，LOWE D G. Scalable nearest neighbor algorithms for high dimensional data [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2014，36(11)：2227-2240.

[12] LV Q，JOSEPHSON W，WANG Z，et al. Multi-probe LSH：efficient indexing for high-dimensional similarity search [C]// 2007 33rd International Conference on Very Large Data Bases. San Jose，CA，USA：VLDB Endowment，2007：950-961.

[13] WAGAMAN A. Efficient *k*-NN graph construction for graphs on variables [J]. Statistical Analysis & Data Mining，2013，6(6)：443-455.

[14] ALVARO B，DORRONSORO J R. Momentum sequential minimal optimization：an accelerated method for support vector machine training [C]// International Joint Conference on Neural Networks. Piscataway，NJ，USA：IEEE，2011：370-377.

[15] BOYD S，VANDENBERGHE L. Convex optimization [J]. IEEE Transactions on Automatic Control，2006，51(11)：1859-1876.

[本刊相关文献链接]

王博，张为，刘艳艳，等. 采用机器学习的火焰前景提取算法. 2017，51(8)：26-32. [doi：10.7652/xjtuxb201708005]

韩江涛，张英杰，李程辉，等. 面向光学三维测量的非均匀条纹生成方法. 2017，51(8)：12-18. [doi：10.7652/xjtuxb201708003]

徐思雨，蔡佳妮，祝继华，等. 自适应多位编码量化的哈希图像检索方法. 2017，51(8)：19-25. [doi：10.7652/xjtuxb201708004]

翟夕阳，王晓丹，李睿，等. 采用多类代价指数损失函数的代价敏感 AdaBoost 算法. 2017，51(8)：33-39. [doi：10.7652/xjtuxb201708006]

付学中，方宗德，关亚彬，等. 采用 NSGA-Ⅱ 算法的面齿轮副小轮拓扑修形多目标优化. 2017，51(7)：98-104. [doi：10.7652/xjtuxb201707015]

高智勇，董荣光，高建民，等. 采用聚类特征的基本概率分配生成方法及应用. 2016，50(10)：8-14. [doi：10.7652/xjtuxb201610002]

（编辑　刘杨）