计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2024,33(3):195−205 [doi: 10.15888/j.cnki.csa.009441]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于混合数据类型相关性度量的非正态数据合成①

王春东, 张世鹏

(天津理工大学 计算机科学与工程学院, 天津 300384)
通信作者: 王春东, E-mail: michael3769@163.com

摘　要: 数据在机器学习、人工智能等领域的研究和开发工作中占据了极其重要的地位. 然而现实中存在的一些因素导致数据需求者无法获得符合工作要求的真实数据集, 例如隐私问题、数据稀缺和数据质量较差等. 针对此现状, 在 SI (sampling-iteration) technique 的基础上改进出一种非正态数据合成算法 (KMSI). 该算法使用混合类型相关系数矩阵以减小 SI technique 在目标设定、控制循环等步骤中的度量误差, 通过替换 Bootstrap 采样法为核密度估计采样法以避免使用真实数据. 实验结果表明, KMSI 相较 SI technique 能够应对复杂分布和混合类型的数据集, 且在合成结果中不包含真实数据; 相较于其他改进方法, KMSI 在合成数据集样本量上能够给予使用者更大的自定义空间.

关键词: 合成数据集; 隐私保护; 相关系数; 核密度估计

引用格式: 王春东,张世鹏.基于混合数据类型相关性度量的非正态数据合成.计算机系统应用,2024,33(3):195–205. http://www.c-s-a.org.cn/1003-3254/9441.html

Non-normal Data Synthesis Based on Mixed Data Type Correlation Measurement

WANG Chun-Dong, ZHANG Shi-Peng

(School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300384, China)

Abstract: Data plays an extremely important role in research and development in fields such as machine learning and artificial intelligence. However, some real-world factors prevent data consumers from obtaining real datasets that meet their work requirements, such as privacy issues, data scarcity, and poor data quality. In response to this situation, this study develops a non-normal data synthesis algorithm (KMSI) as an improvement to the sampling-iteration (SI) technique. This algorithm utilizes a mixed-type correlation coefficient matrix to reduce measurement errors in various steps of the SI technique, including target setting and control loops. It replaces Bootstrap sampling with kernel density estimation sampling to avoid using real data. Experimental results show that, compared to the SI technique, KMSI is capable of handling complex and mixed-type datasets and does not include real data in the synthetic results. Furthermore, compared to other enhancement methods, KMSI offers users more customization options for the sample size in synthetic datasets.

Key words: synthetic dataset; privacy protection; correlation coefficient; kernel density estimation

机器学习与人工智能领域的研究工作方兴未艾, 能否构建出以更高性能和准确度实现预测、分类、识别等模仿人类器官机能的数学模型, 极大程度上依赖着数据的质量, 因此愈发凸显其重要地位[1].

但是, 在现实中总是存在着各种因素, 使得数据需求者不能自由获取到满足自己实验或开发需求的真实数据集, 常见因素有以下 3 种: 第一, 数据集涉及法律限制的隐私问题或保密领域; 比如数据集中包含真实的姓名、住址、职业等信息; 不同国家和区域的法律出于保护公民隐私权的严格规定是此类障碍出现的直

① 基金项目: 国家自然科学基金联合基金 (U1536122); 天津市科委重大专项 (15ZXDSGX00030)
收稿时间: 2023-09-23; 修改时间: 2023-10-20; 采用时间: 2023-11-03; csa 在线出版时间: 2024-01-19
CNKI 网络首发时间: 2024-01-22

接原因. 第二, 数据集比较稀缺; 当研究人员对一些有偿数据集存在需求, 或者数据集的建立需要耗费超过研究人员能够承担的精力与时间成本时, 该因素是不可忽视的[2]. 第三, 数据集中的数据质量较差; 通常表现为数据不平衡、数据量少、特征冗余等问题, 这些问题的存在会直接影响模型的性能, 导致表现下降[3]. 当这些因素单独或叠加出现时, 使用真实数据集对研究工作将难以起到积极作用. 因此, 合成数据生成方法成为一个重要的研究方向, 旨在实现合成数据集与真实数据集的分布与相关性高度相近的目标.

目前关于多元非正态数据的合成方法, 已经有一些工作对该领域进行整理[4]与研究. Fleishman 提出的解决方案[5]基于三阶多项式变换并通过对方程组的求解把一个正态分布单变量转换为自定义非正态分布单变量. 在此基础之上, Vale 等人提出的 VM-transform[6]通过计算中间相关系数矩阵以调整算法对各个变量之间相关关系的影响将其成功运用于生成多变量数据上. VM-transform 确立了以“变换-计算”方式生成多元非正态数据集的基本框架, 之后沿用该框架并做出改进分布描述方式的工作例如文献[7–13]. “采样-迭代”框架即 SI technique, 是由 Ruscio 等人[14]提出并完善的[15]. “采样-迭代”框架对真实数据采样以实现对分布的描述, 并以迭代试错的方式逐步确定最优中间相关系数矩阵; 这消除了“变换-计算”框架对于分布矩的限制, 更有利于处理未定义矩和同矩不同分布的情况. Amatya 等人[16]提出的方法旨在合成混合类型数据集, 并强调了序数、连续型数据分别符合广义泊松分布与正态分布; Humski 等人[17,18]使用插值矩阵的方法进一步扩大了“变换-计算”框架合成样本的生成上限, 并将其应用到社交图谱等复杂数据集上. Foster[19]提出的模型能够以更简单的方式使用因子分析方法构建合成数据集; 该模型与 Bartholomew 等人[20]描述的较为相似.

现有工作对 SI technique 的改进集中在优化对变量分布的描述上, 但鲜有工作具体讨论数据相关性与隐私保护问题. 因此, SI technique 依旧存在改进空间: 第一, 单一相关系数矩阵难以准确度量多类型混合数据的相关性, 多类型混合、非线性数据等现象在真实数据中十分常见[21,22]. 第二, 采样方式无法在隐私保护的前提下重现真实特征列分布. SI technique 中的采样法有两种: Bootstrap 采样法和预定义分布采样法. 对真实数据集使用 Bootstrap 意味着采样结果均为真实数据, 增加了隐私暴露的风险, 且 Bootstrap 极大限制了采样数据量, 这对于使用者是不利的. 使用预定义分布采样虽然能够降低隐私暴露风险, 但预定义的分布并不总是能够符合真实分布的所有情况.

综上所述, SI technique 在应对合成多元非正态数据问题上具有更高的灵活度, 但在一些存在隐私保护要求的场景下有不足. 本文改进的 KMSI 算法在 SI technique 的基础上应用两项改进措施以弥补缺点: 1) 替换 Bootstrap 采样法为核密度估计采样法, 使其在采样结果无真实数据的条件下完成自定义样本量的采样, 且复现真实数据集的分布. 2) 替换原版算法中的皮尔逊相关系数矩阵为混合类型相关系数矩阵, 减小算法在应对混合类型数据集时的度量误差, 提高算法鲁棒性.

1 方法

1.1 核密度估计采样法

核密度估计 (kernel density estimation) 是一种非参数概率密度估计方法, 通常用于在给定数据的情况下估计未知概率密度函数[23]. 假设$X=\{x_1,\cdots,x_n\}$是一组独立同分布的特征值, 取自某未知的概率密度曲线 f, 拟合该组数据的概率密度函数为:

$$f'(x)=\frac{1}{nh}\sum_{i=1}^{n}K\left(\frac{x-x_i}{h}\right) \tag{1}$$

其中, K 代表核函数, h 代表窗宽. 拟合不同类型数据的核密度函数需要选取不同的核函数. 窗宽 h 对拟合结果亦影响较大, 窗宽决定了拟合函数曲线的形态. Botev 等人[24]提出的确定最优窗宽的方法 ISJ (improved Sheather-Jones) 在复杂分布的情况下被认为比 Scott 经验法则[25]拥有更高精度和健壮性, 在实践中取得了较好的效果. 在获得了每一列对应的拟合概率密度函数后使用反函数变换法来获得采样值. 相较 Bootstrap, 核密度估计采样法的采样值不直接来自原始数据, 降低了隐私泄露的可能; 其次, 它无需重复采样, 能够产生更多样的数值; 最后, 使用概率密度函数可以采样任意数量的合成数据, 这使得使用者能够自定义合成数据集大小.

1.2 混合类型相关系数矩阵

相关系数矩阵是 SI technique 算法过程中衡量特征变量之间相关性极为重要的工具. KMSI 中使用的混合类型相关系数矩阵是由斯皮尔曼秩相关系数、克莱姆相关系数和 CMCD (correlation measure between

continuous and discrete features) 相关性度量法共同构成的. 斯皮尔曼秩相关系数适用于度量连续型和连续型数据之间的相关性, 并且对于序数型数据亦有良好表现; 其计算公式如下:

$$\rho_{\text{Spearman}} = 1 - \frac{6 \cdot \sum d_i^2}{n(n^2-1)} \tag{2}$$

其中, d_i 表示第 i 个位次对的差值, n 表示样本数. 克莱姆相关系数适合用来度量离散型变量之间的相关性, 其计算公式如下:

$$\phi_{\text{Cramer}} = \sqrt{\frac{\chi^2}{n(k-1)}} \tag{3}$$

其中, χ^2 表示样本的卡方检测量, n 代表样本量, k 代表任意变量的较少类别数. CMCD 相关性度量法是由 Jiang 等人[26]基于类分离[16]理论提出的, 该理论相较离散化方法能够减少信息丢失并提高效率. 混合类型相关系数矩阵的完整计算方式如算法 2 所示; 由于相关系数适用的数据类型不尽相同, 因此需要输入数据集中每列特征的具体类型 *Type*.

算法 1. CMCD 相关性度量法

输入: 连续特征 $X=(x_1,x_2,\cdots,x_n)$, 离散特征 $Y=(y_1,y_2,\cdots,y_n)$
输出: 相似度 $\text{Sim}(X,Y)$

1) 组合两列特征 $(X,Y)=\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$
2) 根据 Y 中元素不同取值, 把 (X,Y) 划分为 r 组
3) 计算每组连续变量的均值 $U_1,U_2,\cdots,U_i,\cdots,U_r$, $U_i=\frac{1}{n_i}\sum_{j=1}^{n_i} x_{ij}$
4) 计算全部连续变量的均值 $M_0=\frac{1}{n}\sum_{i=1}^{r}\sum_{j=1}^{n_i} x_{ij}$
5) 计算各组中连续数据与对应 U_i 差值平方和 $S_{\text{ithintra}}=\sum_{j=1}^{n_i}(x_{ij}-U_i)^2$
6) 计算 U_i 与 M_0 差值的平方和 $S_{\text{inter}}=\sum_{i=1}^{r} n_i(U_i-M_0)^2$
7) 计算 S_{inter} 与各个 S_{ithintra} 的总和 S_T
8) 计算结果 $Sim(X,Y)=\begin{cases}\frac{S_{\text{inter}}}{S_T}, & \text{if } S_T\neq 0\\ 0, & \text{if } S_{\text{intra}}=S_{\text{inter}}=S_T=0\end{cases}$

算法 2. 混合类型相关系数矩阵计算法

输入: 数据集 $D=\{X_1,X_2,\cdots,X_n\}$, 数据类型 $Type=[t_1,t_2,\cdots,t_n]$
输出: 混合类型相关系数矩阵 M

1) i=0, M=array[n][n]
2) WHILE (i<n) DO
3) WHILE (j=i, j<n) DO
4) IF (*Type*[i]==*Type*[j]) DO
5) IF (*Type*[i]==“continue”) DO
6) $M[i][j]=M[j][i]=\text{Spearman}(X_i,X_j)$
7) END IF
8) IF (*Type*[i]==“discrete”) DO
9) $M[i][j]=M[j][i]=\text{Cramer}(X_i,X_j)$
10) END IF
11) ELSE
12) $M[i][j]=M[j][i]=\text{CMCD}(X_i,X_j)$
13) END IF
14) END WHILE
15) END WHILE
16) RETURN M

1.3 KMSI

SI technique 的核心理念是通过迭代的方式试错最优中间相关系数矩阵, 从而不断提高当前合成数据集和目标相关系数矩阵之间的相似程度, 直到两者之间的差距没有明显缩小为止, 最终使得合成数据集拥有与真实数据集高度一致的相关性表达. 在应用了核密度估计采样法与混合类型相关系数矩阵后, KMSI 具体计算方法如算法 3 所示.

算法 3. KMSI

输入: 核密度估计概率密度函数组 $P(f_1,f_2,\cdots,f_k)$, 因子数量 $N\_factors$, 最大循环次数 max_trials, 初始乘数 $init\_multi$, 特征列数据类型 $data\_type$, 合成数据集样本量 N_desire, 目标相关系数矩阵 M_{target}
输出: 合成数据集 $F'\left(X'_1,X'_2,\cdots,X'_m,C'\right)_{N\times k}$

1) $M_{\text{distribution}}$ =KDE_sampling(P) //核密度估计采样
2) $M_{\text{intermediate}}=M_{\text{target}}$
3) IF ($N\_factors$==Null||$N\_factors$==0) DO
4) $N\_factors$=Parallel_analysis($M_{\text{intermediate}}$,$M_{\text{distribution}}$,$data\_type$)
5) END IF
6) $M_{\text{shared\_comp}}$,$M_{\text{unique\_comp}}$=Normal_sampling(μ=0,σ^2=1)
7) trials_without_improvement =0
8) WHILE (trials_without_improvement<max_trials) DO
9) $M_{\text{shared\_load}}$ =Factor_analysis($M_{\text{intermediate}}$,$N\_factors$)
10) $M_{\text{unique\_load}}$ =Unique_load_calculation($M_{\text{shared\_load}}$)
11) M_{combin}=Combin($M_{\text{shared\_comp}}$,$M_{\text{shared\_load}}$,$M_{\text{unique\_comp}}$, $M_{\text{unique\_load}}$)
12) $F'_{\text{middel\_result}}$ =Data_replace(M_{combin},$M_{\text{distribution}}$)
13) $rmsr$=RMSR_calculation(M_{target},$F'_{\text{middle\_result}}$,$data\_type$)
14) IF ($rmsr$ is the lowest) DO
15) $M_{\text{best\_corr}}$ =$M_{\text{intermediate}}$
16) trials_without_improvement=0
17) ELSE DO
18) Update $M_{\text{intermediate}}$ by using init_multi
19) trials_without_improvement++
20) END IF
21) END WHILE
22) 使用 $M_{\text{intermediate}}$ 和 N_factors 按照步骤 9)–步骤 12) 得到最终结果 F'
23) return F'

KMSI 首先使用核密度估计采样法和混合类型相关系数矩阵以提取数据的分布与相关性信息, 保存在 Distribution 矩阵和 target 矩阵中, 分别对应步骤 1) 和

步骤 2). 在步骤 4)、步骤 9)、步骤 15)、步骤 21) 对应的平行分析、因子分析、循环控制、数据生成步骤中也需要使用了混合类型相关系数矩阵. 此外, 在步骤 13) 中使用了均方根残差 (root mean square residual, *RMSR*) 来监控中间相关系数矩阵与真实相关系数矩阵之间差距, 以实现循环控制. *RMSR* 的计算方法如式 (4)、式 (5) 所示.

$$r_{\text{residual}} = (C_{\text{real}} - C_{\text{magnify}} \text{ or } C_{\text{redual}})_{\text{UnderTriangularElements}} \tag{4}$$

$$RMSR = \sqrt{\frac{\sum r_{\text{residual}}^2}{k(k-1)/2}} \tag{5}$$

2 实验与评价

本节实验首先在若干 UCI 数据集上检验改进方式的有效性, 再使用两种机器学习算法在不同的训练集和测试集上构建模型, 以模型性能指标为基准完成对合成数据集质量和 KMSI 鲁棒性的评测.

2.1 改进点测试

2.1.1 核密度估计采样

本节实验旨在测试核密度估计采样法生成的 Distribution 矩阵对真实数据集分布的复现程度. 表 1 展示了这些数据集的相关信息, 其中 C/D features 表示各个数据集中连续与离散型特征列的数量, sampling proportion 表示本节实验中对各个数据集的采样比例.

表 1 数据集信息

Dataset	C/D features	Size	Sampling proportion (%)	Class
Adult	2/8	32 561	5	2
Bank	7/10	45 211	5	2
Credit	6/10	690	400	2
Heart	5/9	303	200	2

分别对 4 个数据集的各个特征列使用核密度估计法拟合其概率密度函数, 并使用拟合得来的概率密度函数进行随机采样以得到不同采样比例的 Distribution 矩阵. 对于连续型特征, 采用高斯核函数赋予权重、ISJ algorithm 计算窗宽; 对于离散型特征, 采用 Delta 核函数赋予权重、Silverman 经验法则计算窗宽. 图 1–图 4 展示了一部分特征列与真实特征列的对比结果, 纵坐标 *P* 表示概率密度. 图 1 对应 Adult 数据集连续型特征“age”和离散型特征“marital-status”的分布情况, 图 2 对应 Bank 数据集连续型特征“duration”和离散型特征“job”的分布情况, 图 3 和图 4 则分别对应 Credit 和 Heart 两个数据集中不同类型特征列的分布情况. 真实特征列在其定义域范围内表现出多峰、长尾、偏移等现象, 这既表明真实情况下的数据分布较难满足正态分布的假设, 也证明了原版算法使用预定义的概率密度函数采样的做法存在不合理性. 相比较于核密度估计采样法得到的结果, 黄色直方图代表的采样数据与蓝色直方图代表的真实数据在分布上是基本一致的, 这证明核密度估计法具备还原真实数据集分布的能力.

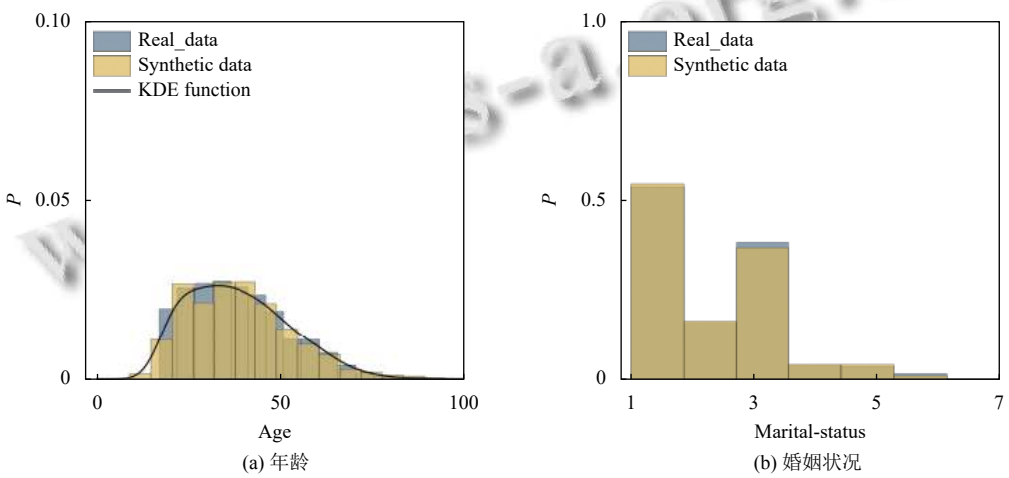
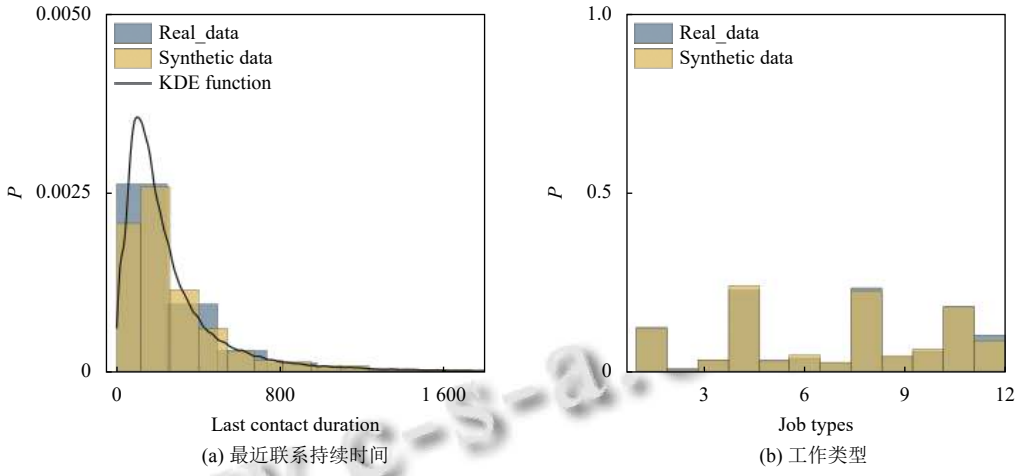


图 1 Adult 特征分布对比

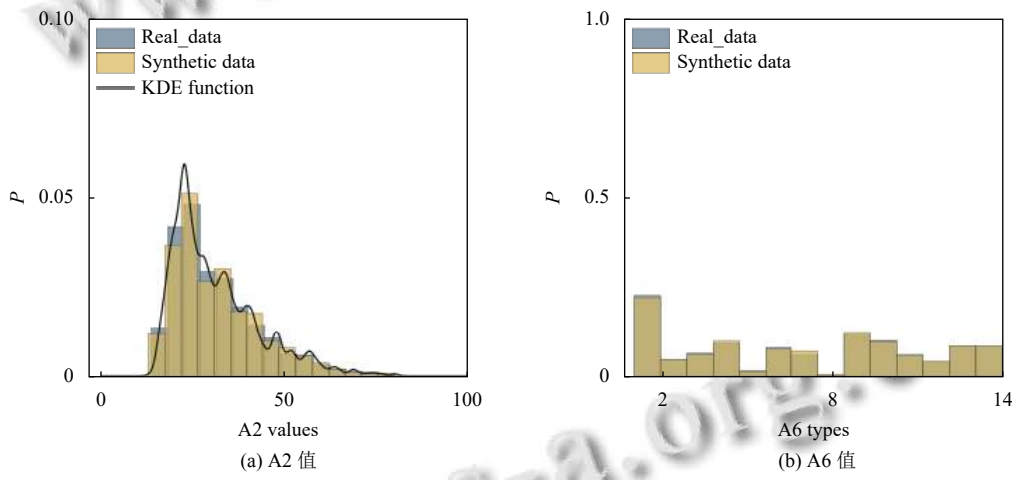
此外, 对于 Petricioli 等人[18]提出的使用分位数矩阵插值法生成 Distribution 矩阵同样进行了测试, 在上述数据集上计算了合成数据特征列与真实数据特征列之间的 KS-test 值, 以考察分位点数量 q 与 Distribution 矩阵复现分布效果之间的关系. 设定各个合成数据集的样本量与对应真实数据集相等以统一采样比例并消

除样本数量带来的影响; 为优化展示效果, 仅对部分结果绘制成图, 如图 5–图 7 所示. 随着分位点数量 q 的增加, 两种不同类型特征列的 KS-test 值均逐渐下降到较低水平并不断震荡, 伴随着 p 值逐渐上升到 0.05 以上, 这代表了两个分布的相似程度在不断增加并趋于稳定, 总体 Distribution 矩阵的分布情况越来越接近真实数据集.



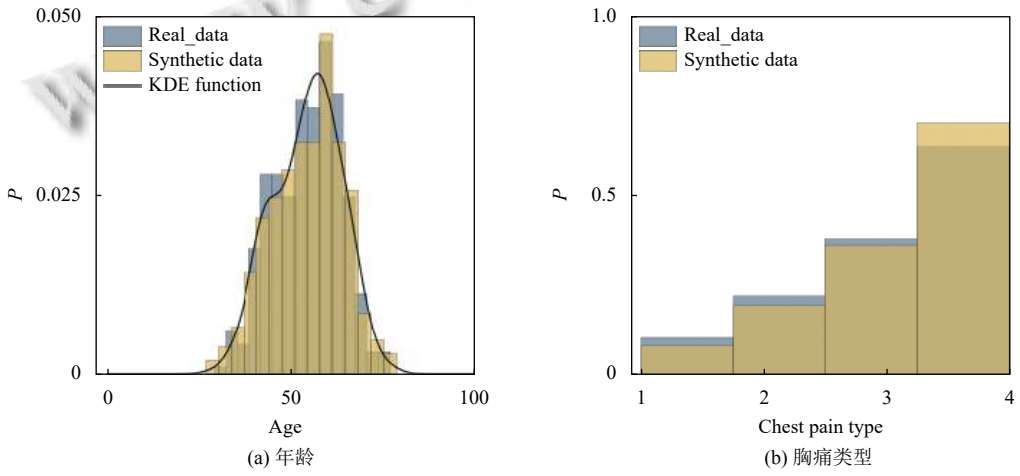
(a) 最近联系持续时间 (b) 工作类型

图 2 Bank 特征分布对比



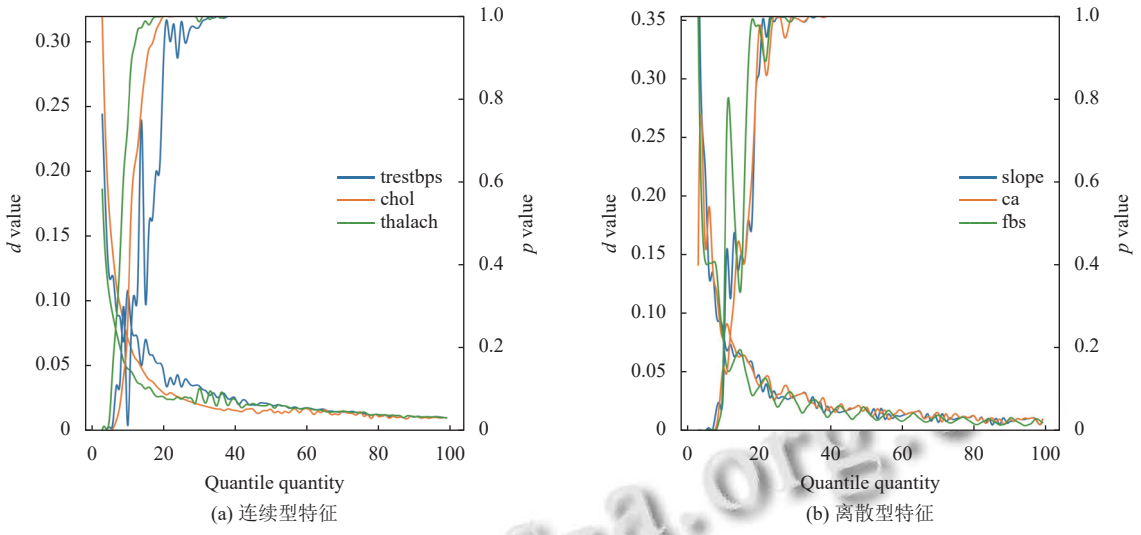
(a) A2 值 (b) A6 值

图 3 Credit 特征分布对比



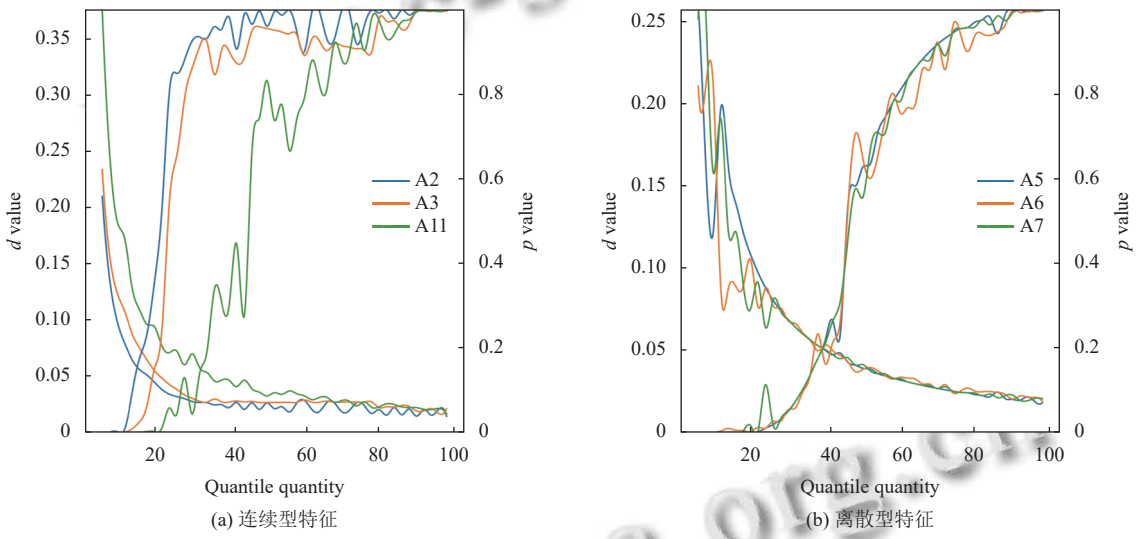
(a) 年龄 (b) 胸痛类型

图 4 Heart 特征分布对比



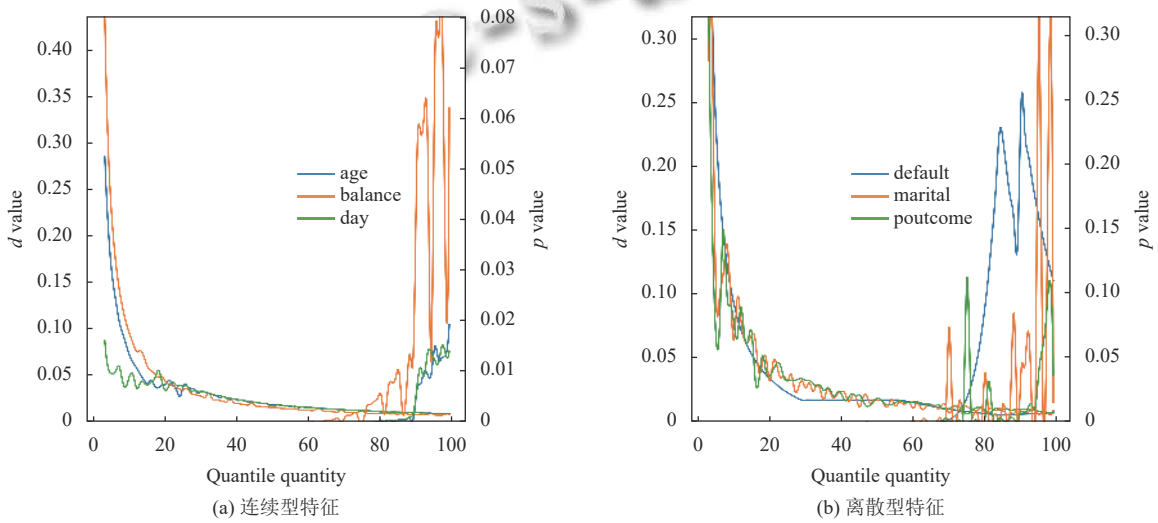
(a) 连续型特征 (b) 离散型特征

图 5 Heart K-S 检验



(a) 连续型特征 (b) 离散型特征

图 6 Credit K-S 检验



(a) 连续型特征 (b) 离散型特征

图 7 Bank K-S 检验

虽然该方法在获取扩大的 Distribution 矩阵上的确有良好表现, 但当希望得到一个缩小的 Distribution 矩阵时, 分位点数量 q 必须谨慎选择. 设定以 Adult 数据集为基础的合成数据量为原数据集的 0.1% (32 例样本), 测试 q 取值范围为[2, 500]时 KS-test 的变化; 如图 8 所示, 当 q 值超过所需样本量时, KS-test 值出现了明显的宽幅波动, 在个别取值点甚至出现了高于起始值的现象, 这代表着该方法生成的合成数据无法稳定还原真实数据的分布. 此外, 当 q 值超过所需样本量时, 论文中用于扩大样本量的插值模块将被跳过, 使得 Distribution 矩阵里将只包含来自真实数据集的样本, 隐私安全问题也就成为该算法的致命缺点.

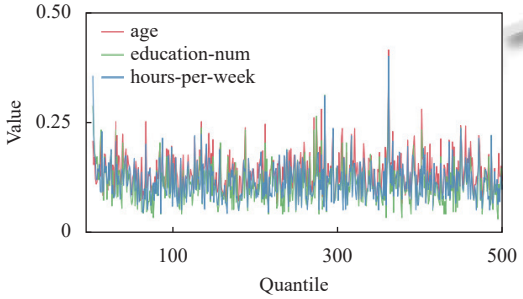


图 8 Adult K-S 检验

2.1.2 混合类型相关系数矩阵

为了检验 KMSI 在生成不同大小 Distribution 矩阵时的鲁棒性, 模拟实验设置如下: 首先按照算法 2 计算数据集的混合类型相关系数矩阵 C_{real}, 再使用核密度估计采样法得到不同比例放大和缩小的 Distribution 矩阵 D_{magnify} 和 D_{reduce}, 然后分别计算二者的混合类型相关矩阵 C_{magnify}、C_{reduce}, 最后使用可视化工具讨论 C_{real} 和 C_{magnify}、C_{reduce} 之间均方根残差的稳定性.

考虑到 4 个数据集的样本量不尽相同, 故 Adult 和 Bank 数据集重点测试 C_{reduce} 的变化; Credit 和 Heart 数据集重点测试 C_{magnify} 的变化. 表 2 列出了该模拟实验中各个数据集的测试点.

表 2 模拟实验设置

Dataset	Size	Testing sampling proportion (%)
Adult	32 561	0.1, 0.15, 0.22, 0.34, 0.50, 0.75, 1.13, 1.71, 2.56, 3.84, 5.76, 8.64, 12.97, 19.46, 29.19, 43.78, 65.68, 98.52
Bank	45 211	0.1, 1.1, 2.1, 3.1, 4.1, 5.1, 6.1, 7.1, 8.1, 9.1
Credit	690	100, 300, 500,⋯, 3700, 3900
Heart	303	100, 300, 500,⋯, 3700, 3900

图 9、图 10 展示了这 4 个数据集在各个测试点的 *RMSR* 变化, 可以看出, 在扩大和缩小情况下真实数据集和 Distribution 矩阵的混合类型相关系数矩阵之间的差距并没有发生较大变动, 总体上维持在±0.1 范围内; 这说明混合类型相关系数矩阵在数据充足和稀缺的条件下具有健壮性, 与核密度估计采样法结合后无不良表现.

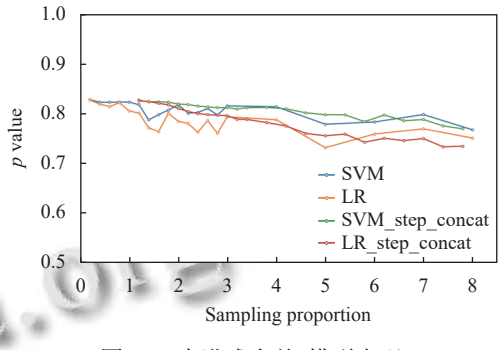
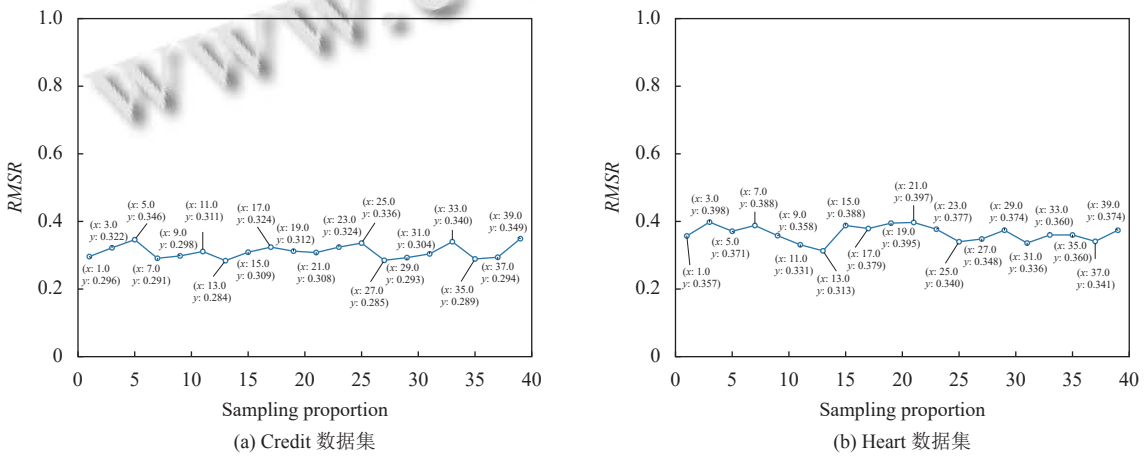


图 9 “步进式合并”模型表现



(a) Credit 数据集 (b) Heart 数据集

图 10 Credit 与 Heart 数据集的 *RMSR* 波动

2.2 总体测试

本节实验通过检验合成数据的质量来展示 KMSI 算法的总体表现.

2.2.1 相关信息

该部分实验使用 Adult 数据集, 为了检验改进算法在数据不平衡情况下的性能, 本节实验在数据预处理阶段采用了文献[27]中构建的数据划分工具 FedLab 对真实数据集划分出 4 种不同类型的子数据集; 通过使用合成数据集构建的分类模型性能来讨论算法效果. 对于在生成合成数据阶段的数据集不对离散型特征进行特殊编码, 仅用自然数表示不同类别; 在模型验证阶段则需要对离散特征进行特殊编码以获得较高的模型性能. 实验代码使用 Golang 和 Python 共同编写完成.

2.2.2 实验评估

在完成数据预处理等步骤后, KMSI 以一个从真实数据集中计算得来的目标相关系数矩阵 C_{target} 和使用核密度估计采样法得来的初始 Distribution 矩阵为算法输入. KMSI 的超参数 iter_without_improvement 限定了算法在得到新的最小值的最大步数, init_multi 则起到学习率的作用. 在实际实验中发现 iter_without_improvement 取 5 左右, init_multi 取 2 左右比较合适.

实验使用真实数据集训练逻辑回归和支持向量机分类模型, 以分类模型的准确度作为评价数据质量的标准. 使用机器学习指标判断更加客观具体, 且借助高准确度的机器学习模型有利于判别合成数据集的总体质量. 表 3–表 7 展示了机器学习分类器在各个合成数据集上的表现, 为探究用户自定义样本量对算法的影响, 表 4 中的合成数据子集样本数量等同于对应的真实数据子集, 其他表格对应的合成数据子集样本数量均为 500. 使用真实数据集训练的支持向量机模型与逻辑回归模型的准确度分别是 0.837、0.828; 从实验数据可以看出, 两种分类模型在真实数据集上的准确度整体上明显高于在合成数据集上的准确度, 这说明合成数据集中的确存在不符合真实特征关系的样本. 但通过对比不同子集划分方式 (表 3, 表 5–表 7) 和不同样本数量 (表 3, 表 4) 的结果, 可以看出分类模型在合成数据集上的准确度并没有出现明显变化, 始终保持在 0.7 左右. 这种现象说明 KMSI 在生成任意大小的合成数据集时能够保持性能稳定, 具备了在满足数据隐私的前提下应对特征列复杂分布和数据不平衡问题的鲁棒性, 能够给予使用者在样本量上更大的自定义空间.

表 3 基于 Dirichlet 分布的标签偏移子集在模型上的表现

idx	Quantity		SVM	LR	idx	Quantity		SVM	LR
	real	syn				real	syn		
0	742	500	0.746	0.698	10	3511	500	0.758	0.748
1	618	500	0.742	0.716	11	2598	500	0.738	0.686
2	1334	500	0.726	0.696	12	576	500	0.744	0.706
3	464	500	0.77	0.744	13	704	500	0.728	0.694
4	49	500	0.756	0.736	14	1569	500	0.72	0.718
5	295	500	0.722	0.692	15	2004	500	0.726	0.702
6	43	500	0.73	0.664	16	1048	500	0.684	0.668
7	2416	500	0.77	0.748	17	228	500	0.726	0.678
8	2743	500	0.754	0.746	18	7317	500	0.76	0.75
9	2069	500	0.73	0.722	19	2233	500	0.776	0.744

表 4 基于 Dirichlet 分布的数量偏移子集在模型上的表现

idx	Quantity		SVM	LR	idx	Quantity		SVM	LR
	real	syn				real	syn		
0	13	13	0.717	0.676	10	1636	1636	0.731	0.714
1	5851	5851	0.757	0.720	11	2755	2755	0.744	0.727
2	6388	6388	0.72	0.694	12	2423	2423	0.786	0.765
3	38	38	0.73	0.71	13	1628	1628	0.711	0.681
4	150	150	0.77	0.754	14	890	890	0.755	0.724
5	2314	2314	0.745	0.674	15	359	359	0.730	0.704
6	648	648	0.734	0.7	16	855	855	0.733	0.702
7	187	187	0.740	0.703	17	286	286	0.698	0.636
8	605	605	0.745	0.720	18	1971	1971	0.730	0.713
9	2994	2994	0.724	0.693	19	558	558	0.746	0.752

表 5 基于 Dirichlet 分布的数量偏移子集在模型上的表现

idx	Quantity		SVM	LR	idx	Quantity		SVM	LR
	real	syn				real	syn		
0	13	500	0.836	0.698	10	1636	500	0.763	0.724
1	5851	500	0.734	0.732	11	2755	500	0.796	0.79
2	6388	500	0.734	0.730	12	2423	500	0.744	0.72
3	38	500	0.792	0.778	13	1628	500	0.738	0.72
4	150	500	0.756	0.71	14	890	500	0.708	0.7
5	2314	500	0.772	0.776	15	359	500	0.774	0.74
6	648	500	0.75	0.732	16	855	500	0.718	0.698
7	187	500	0.746	0.716	17	286	500	0.776	0.68
8	605	500	0.728	0.704	18	1971	500	0.714	0.7
9	2994	500	0.768	0.7	19	558	500	0.744	0.692

表 6 独立同分布子集在两种模型上的表现

idx	Quantity		SVM	LR	idx	Quantity		SVM	LR
	real	syn				real	syn		
0	1628	500	0.73	0.726	10	1628	500	0.722	0.714
1	1628	500	0.772	0.746	11	1628	500	0.72	0.72
2	1628	500	0.748	0.728	12	1628	500	0.746	0.724
3	1628	500	0.776	0.75	13	1628	500	0.732	0.736
4	1628	500	0.768	0.736	14	1628	500	0.712	0.682
5	1628	500	0.728	0.688	15	1628	500	0.714	0.698
6	1628	500	0.724	0.708	16	1628	500	0.728	0.696
7	1628	500	0.784	0.756	17	1628	500	0.742	0.694
8	1628	500	0.706	0.702	18	1628	500	0.74	0.714
9	1628	500	0.75	0.754	19	1628	500	0.72	0.78

表 7 基于样本量的标签偏移子集在两种模型上的表现

idx	Quantity		SVM	LR	idx	Quantity		SVM	LR
	real	syn				real	syn		
0	2472	500	0.742	0.72	10	2472	500	0.764	0.738
1	785	500	0.782	0.766	11	785	500	0.76	0.742
2	2472	500	0.706	0.698	12	2472	500	0.766	0.736
3	785	500	0.78	0.732	13	785	500	0.75	0.74
4	2472	500	0.76	0.702	14	2472	500	0.736	0.71
5	785	500	0.792	0.762	15	785	500	0.652	0.636
6	2472	500	0.724	0.702	16	2472	500	0.74	0.684
7	785	500	0.742	0.736	17	785	500	0.734	0.748
8	2472	500	0.75	0.732	18	2472	500	0.722	0.72
9	785	500	0.7692	0.676	19	785	500	0.736	0.706

为了探究这部分不符合真实特征关系的样本对模型的具体影响, 实验把合成数据子集和对应的真实数据子集合并, 分别用合并前后的数据集训练分类模型, 在完整数据集上测试准确度, 结果如表 8 所示; 为了更直观地展示 KMSI 在产生更宽样本量范围的合成数据集上的能力, 以第 13 号真实数据子集为例生成不同比例的合成数据子集, 使用上述方式测试了分类模型的准确度, 结果如图 11 所示. 从表格和折线图中观察到在生成缩小和扩大的合成数据集时, 模型的准确度分别维持在 80% 和 75% 左右. 结合前一部分的实验结果可以推断出, 若进一步扩大合成数据在合并后数据集中的比例, 模型准确度将维持在 70% 左右, 接近表 3–表 7 所示的结果. 鉴于 KMSI 在生成缩小的合成数据集上拥有较好表现, 本实验也尝试用“步进式合并(step-merging)”的方法, 即用数个合成样本比例为 20% 的合成数据集与真实数据集合并来产生不同大小的扩大数据集; 结果展示在图 12 中. 同样以第 13 号真实数据子集为材料, “步进式”方法在整体稳定性上有了明显优化, 尤其是在合成样本比例为[1.1, 3.0]范围内表现出优于直接合成方法的稳定性. 因此, 在应对生成扩大合成数据集任务时, 使用步进式合并的方式是更具鲁棒性的方式.

表 8 使用合并前后数据子集训练的两种模型在真实数据集上的表现

No.	Dir label skew quantity=500				Dir label skew equal quantity				Dir quantity skew quantity=500				IID quantity=500				Quantity skew quantity=500			
	Before		After		Before		After		Before		After		Before		After		Before		After	
	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR	SVM	LR
0	0.825	0.825	0.822	0.816	0.824	0.826	0.801	0.789	0.820	0.783	0.761	0.728	0.825	0.826	0.826	0.824	0.827	0.826	0.826	0.824
1	0.812	0.823	0.811	0.810	0.813	0.825	0.811	0.808	0.831	0.828	0.831	0.828	0.832	0.827	0.830	0.820	0.827	0.826	0.824	0.817
2	0.823	0.826	0.820	0.822	0.824	0.825	0.808	0.799	0.829	0.827	0.828	0.826	0.827	0.827	0.825	0.825	0.834	0.828	0.831	0.825
3	0.825	0.825	0.817	0.807	0.823	0.823	0.818	0.804	0.817	0.809	0.733	0.720	0.821	0.826	0.817	0.816	0.826	0.825	0.824	0.812
4	0.804	0.807	0.804	0.770	0.804	0.808	0.810	0.760	0.813	0.827	0.788	0.770	0.837	0.829	0.831	0.819	0.832	0.828	0.831	0.826
5	0.822	0.817	0.806	0.788	0.822	0.819	0.818	0.810	0.833	0.828	0.832	0.828	0.831	0.828	0.827	0.822	0.819	0.826	0.812	0.818
6	0.810	0.821	0.759	0.752	0.806	0.820	0.770	0.754	0.821	0.824	0.818	0.813	0.827	0.825	0.827	0.823	0.832	0.827	0.830	0.821
7	0.831	0.826	0.831	0.827	0.827	0.827	0.821	0.804	0.816	0.826	0.796	0.798	0.828	0.827	0.827	0.828	0.829	0.827	0.825	0.817
8	0.833	0.827	0.833	0.821	0.833	0.828	0.827	0.805	0.824	0.828	0.812	0.803	0.830	0.825	0.830	0.820	0.829	0.825	0.828	0.820
9	0.826	0.826	0.825	0.818	0.828	0.828	0.826	0.806	0.831	0.828	0.829	0.825	0.830	0.828	0.828	0.827	0.826	0.825	0.819	0.816
10	0.833	0.830	0.825	0.826	0.833	0.828	0.824	0.805	0.824	0.827	0.823	0.818	0.825	0.827	0.825	0.823	0.829	0.828	0.828	0.826
11	0.826	0.825	0.826	0.821	0.828	0.827	0.825	0.812	0.826	0.824	0.827	0.822	0.830	0.828	0.826	0.824	0.826	0.828	0.821	0.821
12	0.825	0.824	0.819	0.815	0.823	0.824	0.819	0.809	0.832	0.827	0.831	0.825	0.830	0.828	0.829	0.823	0.832	0.828	0.830	0.825
13	0.822	0.828	0.820	0.807	0.823	0.828	0.817	0.794	0.827	0.826	0.823	0.819	0.832	0.826	0.828	0.827	0.829	0.824	0.820	0.809
14	0.829	0.827	0.827	0.822	0.827	0.826	0.822	0.807	0.828	0.825	0.827	0.820	0.828	0.830	0.824	0.823	0.832	0.828	0.829	0.824
15	0.830	0.828	0.829	0.826	0.826	0.828	0.818	0.801	0.817	0.826	0.818	0.796	0.823	0.827	0.822	0.816	0.824	0.827	0.816	0.807
16	0.827	0.829	0.823	0.822	0.828	0.829	0.818	0.800	0.829	0.828	0.822	0.814	0.831	0.824	0.832	0.820	0.828	0.827	0.825	0.824
17	0.795	0.822	0.769	0.766	0.795	0.822	0.781	0.745	0.823	0.824	0.819	0.798	0.829	0.827	0.827	0.822	0.826	0.826	0.824	0.814
18	0.831	0.828	0.832	0.828	0.833	0.828	0.824	0.808	0.831	0.828	0.829	0.826	0.829	0.826	0.824	0.819	0.827	0.827	0.826	0.823
19	0.830	0.825	0.825	0.818	0.832	0.825	0.818	0.800	0.810	0.813	0.806	0.786	0.829	0.828	0.827	0.818	0.826	0.824	0.819	0.805

3 总结与展望

SI technique 是一种基于概率统计学构建的合成数据生成算法, 它以迭代的方式不断缩小采样数据与真实数据相关系数矩阵之间的距离, 进而找到合适的中间相关矩阵以合成数据. KMSI 对原版采样方法和相关性度量方法做出调整, 使其能够应对复杂分布特征列和混合数据类型数据集; 其合成结果不包含真实数据, 并降低了原版算法对合成样本量的限制. 通过对机器学习模型准确度的观察, KMSI 在获得较真实数据集更小的合成数据集上有良好表现, 实验数据也佐证了使用“步进式”方法是获得扩大的合成数据集的较优选择.

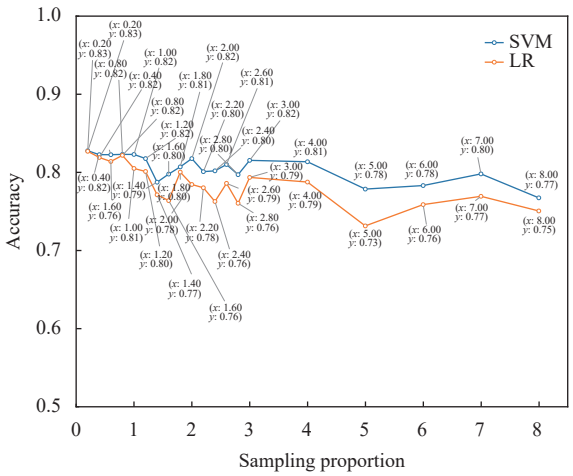
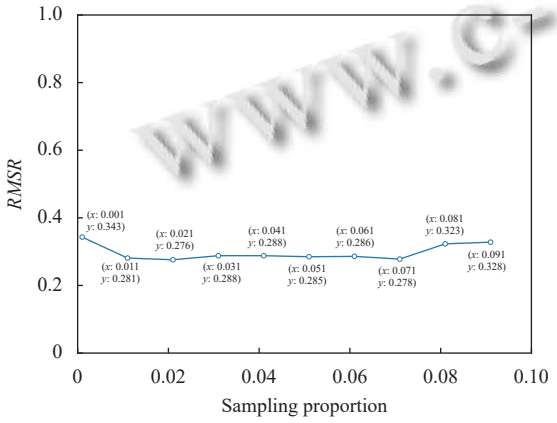
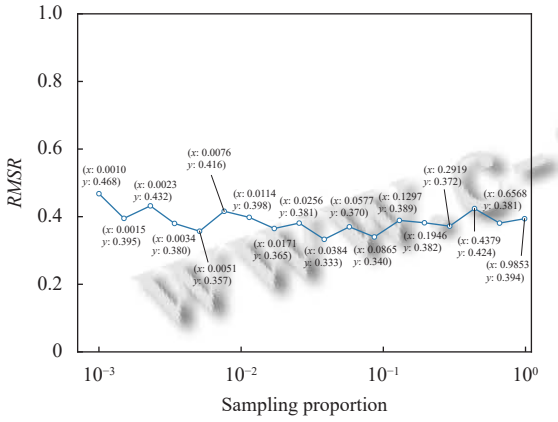


图 11 合并后模型表现



(a) Bank 数据集



(b) Adult 数据集

图 12 Adult 与 Bank 数据集的 *RMSR* 波动

生成合成数据的意义不仅在于其本身, 也在于它暗示了另一条改进数据集不平衡问题的道路, 即有针对性地对一个数据集中的少数类数据使用合成数据生成方法, 通过产生少数类合成数据扩充数据集来达到平衡类别的目的. 后续工作将按照该思路把 KMSI 与联邦学习相结合, 以探索新的应用场景.

参考文献

1 Lu YZ, Shen MJ, Wang HZ, *et al*. Machine learning for synthetic data generation: A review. arXiv:2302.04062, 2023.

2 Babbar R, Schölkopf B. Data scarcity, robustness and extreme multi-label classification. Machine Learning, 2019, 108(8–9): 1329–1351. [doi: 10.1007/s10994-019-05791-5]

3 Pipino LL, Lee YW, Wang RY. Data quality assessment. Communications of the ACM, 2002, 45(4): 211–218. [doi: 10.1145/505248.506010]

4 Olvera OL. Issues, problems and potential solutions when simulating continuous, non-normal data in the social sciences. https://conferences.lnu.se/index.php/metapsychology/article/view/2117. (2020-07-31).

5 Fleishman AI. A method for simulating non-normal distributions. Psychometrika, 1978, 43(4): 521–532. [doi: 10.1007/BF02293811]

6 Vale CD, Maurelli VA. Simulating multivariate nonnormal distributions. Psychometrika, 1983, 48(3): 465–471. [doi: 10.1007/BF02293687]

7 Headrick TC. Fast fifth-order polynomial transforms for generating univariate and multivariate nonnormal distributions. Computational Statistics & Data Analysis, 2002, 40(4): 685–711.

8 Nagahara Y. A method of simulating multivariate nonnormal distributions by the Pearson distribution system and estimation. Computational Statistics & Data Analysis, 2004, 47(1): 1–29.

9 Foldnes N, Olsson UH. A simple simulation technique for nonnormal data with prespecified skewness, kurtosis, and covariance matrix. Multivariate Behavioral Research, 2016, 51(2-3): 207–219. [doi: 10.1080/00273171.2015.1133274]

10 Fialkowski A, Tiwari H. SimCorrMix: Simulation of correlated data with multiple variable types including continuous and count mixture distributions. The R Journal, 2019, 11(1): 250–286. [doi: 10.32614/RJ-2019-022]

11 Ferrari PA, Barbiero A. Simulating ordinal data. Multivariate Behavioral Research, 2012, 47(4): 566–589. [doi: 10.1080/00273171.2012.692630]

12 Demirtas H, Gao R. Mixed data generation packages and related computational tools in R. Communications in Statistics—Simulation and Computation, 2022, 51(8): 4520–4563. [doi: 10.1080/03610918.2020.1745841]

13 Foldnes N, Grønneberg S. Non-normal data simulation using piecewise linear transforms. Structural Equation Modeling: A

Multidisciplinary Journal, 2022, 29(1): 36–46. [doi: 10.1080/10705511.2021.1949323]

14 Ruscio J, Ruscio AM, Meron M. Applying the bootstrap to taxometric analysis: Generating empirical sampling distributions to help interpret results. Multivariate Behavioral Research, 2007, 42(2): 349–386. [doi: 10.1080/00273170701360795]

15 Ruscio J, Kaczetow W. Simulating multivariate nonnormal data using an iterative algorithm. Multivariate Behavioral Research, 2008, 43(3): 355–381. [doi: 10.1080/00273170802285693]

16 Amatya A, Demirtas H. Concurrent generation of multivariate mixed data with variables of dissimilar types. Journal of Statistical Computation and Simulation, 2016, 86(18): 3595–3607. [doi: 10.1080/00949655.2016.1177530]

17 Humski L, Pintar D, Vranić M. Analysis of Facebook interaction as basis for synthetic expanded social graph generation. IEEE Access, 2019, 7: 6622–6636. [doi: 10.1109/ACCESS.2018.2886468]

18 Petricioli L, Humski L, Vranić M, *et al*. Data set synthesis based on known correlations and distributions for expanded social graph generation. IEEE Access, 2020, 8: 33013–33022. [doi: 10.1109/ACCESS.2020.2970862]

19 Foster R. Simulating factor structures from continuous and discrete distributions using mixture of means within a hierarchical model. https://osf.io/b43yq. (2022-11-28).

20 Bartholomew D, Knott M, Moustaki I. Latent variable models and factor analysis: A unified approach. 3rd ed., Chichester: John Wiley & Sons, Ltd., 2011.

21 Kairouz P, McMahan HB, Avent B, *et al*. Advances and open problems in federated learning. Foundations and Trends® in Machine Learning, 2021, 14(1–2): 1–210.

22 Solorio-Fernández S, Carrasco-Ochoa JA, Martínez-Trinidad JF. A survey on feature selection methods for mixed data. Artificial Intelligence Review, 2022, 55(4): 2821–2846. [doi: 10.1007/s10462-021-10072-6]

23 Kamalov F, Elnagar A. Kernel density estimation-based sampling for neural network classification. Proceedings of the 2021 International Symposium on Networks, Computers and Communications (ISNCC). Dubai: IEEE, 2021. 1–4.

24 Botev ZI, Grotowski JF, Kroese DP. Kernel density estimation via diffusion. The Annals of Statistics, 2010, 38(5): 2916–2957.

25 Scott DW. Multivariate density estimation: Theory, practice, and visualization. 2nd ed., Hoboken: John Wiley & Sons, Ltd., 2015.

26 Jiang SY, Wang LX. Efficient feature selection based on correlation measure between continuous and discrete features. Information Processing Letters, 2016, 116(2): 203–215. [doi: 10.1016/j.ipl.2015.07.005]

27 Li QB, Diao YQ, Chen Q, *et al*. Federated learning on non-IID data silos: An experimental study. Proceedings of the 38th IEEE International Conference on Data Engineering (ICDE). Kuala Lumpur: IEEE, 2022. 965–978.

(校对责编: 牛欣悦)