计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(2):30−36 [doi: 10.15888/j.cnki.csa.006195]

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

基于非时间属性关联的数据逼真生成算法①

张　锐, 肖如良, 倪友聪, 杜　欣, 蔡声镇

(福建师范大学 数学与信息学院, 福州 350117)
(福建师范大学 数字福建环境监测物联网实验室, 福州 350117)
通讯作者: 肖如良, E-mail: xiaoruliang@163.com

摘　要: 提出基于非时间属性关联的数据逼真生成算法. 该算法可以解决数据生成器研发中非时间属性关联构建的困难问题, 在大数据测评领域中对仿真数据生成有重要应用价值. 首先, 从数据集中提取关键的两个非时间属性, 对它们分别做两重频数统计. 然后, 根据两次统计结果计算最大信息系数值来评估相关性, 用拉伸指数分布进行拟合, 构建出关联模型. 最后, 通过模型参数构建约束, 在此约束的二维矩阵中生成数据. 实验结果表明, 该算法能够有效地模拟真实数据集的数据特征.

关键词: 数据逼真生成; 关联; 最大信息系数; 拉伸指数分布; 属性关联

引用格式: 张锐,肖如良,倪友聪,杜欣,蔡声镇.基于非时间属性关联的数据逼真生成算法.计算机系统应用,2018,27(2):30–36. http://www.c-s-a.org.cn/1003-3254/6195.html

Table Data Simulation Generating Algorithm Based on Not-Temporal Attribute

ZHANG Rui, XIAO Ru-Liang, NI You-Cong, DU Xin, CAI Sheng-Zhen

(College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China)
(Fujian Provincial Digit Fujian Internet-of-Things Laboratory of Environmental Monitoring, Fujian Normal University, Fuzhou 350117, China)

Abstract: A table data simulation generating algorithm is proposed based on not-temporal attribute correlation. This algorithm can overcome the difficulty in building not-temporal attribute correlation in the development of big data simulation generator, and play an important role in the field of measurement of the big data simulation generated. Firstly, we extract the two key not-temporal attributes from the data set, and make the statistics of twofold frequency. Then, based on the statistical results, we calculate the maximal information coefficient (MIC) value to measure dependence for two-variable relationships. We use the stretched exponential (SE) distribution to fit the relationship, and build the correlation model. Finally, we generate data in a two-dimensional matrix with this model. The experimental results show that this algorithm can effectively describe the data characteristics of the real data set.

Key words: data simulation generator; correlation; maximal information coefficient (MIC); stretched exponential distribution; attribute correlation

在大数据评测中, 因大数据集不易获取, 对大数据生成工具的研究引起了广泛关注. 在大数据生成工具研究中最关键的困难问题是如何刻画真实数据集中的数据特征. 学术界已经有大量的关于数据特征的工作, 可简单地划分为两类: 基于属性特征的仿真数据生成和基于属性关联特征仿真数据生成.

依据是否考虑时间变化因素, 基于属性特征的相关研究主要分为时间属性相关性质的研究和非时间属

① 基金项目: 福建省科技计划重大项目 (2016H6007); 福州市市校合作项目 (2016-G-40)
收稿时间: 2017-05-02; 修改时间: 2017-05-19; 采用时间: 2017-06-02; csa 在线出版时间: 2018-01-12

性相关性质的研究. 对于前者, 在很多领域已经有较为成熟的应用. 比如, 用于优化缓存性能的工作负载仿真数据生成[1]、在人类行为动力学领域研究人类行为事件时, 基于时间间隔分布的仿真数据生成[2]. 对于后者, 美国俄亥俄州立大学张晓东等[3]基于非时间相关属性的仿真数据生成时, 提出拉伸指数 (Stretched Exponential, SE) 分布比 Zipf-like 分布能更好的刻画频数与其排名的关系, 也能更好的刻画重尾特征.

而基于属性关联特征的仿真数据生成相关研究主要从相关性方面展开研究. 比如在大数据生成器研究中, 加拿大萨斯喀彻温大学 Busari[4]等研制了 ProWGen 数据生成器, 对属性关联采用正/负相关的属性关联方式进行了实现; 中科院计算所詹剑锋等研发的可扩展大数据生成器 BDGS[5]; 加拿大多伦多大学 Rabl[6]等研发的 PDGF 框架, 以及已在广泛使用的大数据评测标准 BigDataBench[7]等. 以上这些大数据生成工具很少有涉及到除正/负相关以外的属性关联特征的相关性方面, 普遍将重心放在对属性特征的研究层面, 而少有对属性间的关联特征的研究.

针对属性关联特征构建难的问题, 本文受张晓东等[3]研究的启发, 提出了基于非时间属性关联的数据逼真生成算法 (Table Data Simulation Generating Algorithm Based on Not-temporal Attribute Correlation, TDSA), 重点研究了非时间属性之间的关联特征, 通过仿真生成数据特征与真实数据特征的对比, 实验结果表明, 该算法能够有效的模拟真实数据集.

1 相关工作

为了逼真生成数据, 已经有了很多相关研究. 对单一属性特征的刻画一般有两种方式: 一种是通过随机、枚举或者数据字典的方式, 主要作用于非关键属性; 另一种是通过分布特征的方式, 主要用于关键属性. 比如, Gray[8]采用均匀分布、指数分布、正态分布、自相似分布刻画关键属性, 该项工作具有非常重要的意义; Rabl[6]等使用 β 分布、二项指数分布、对数正态分布、泊松分布等形式来模拟关键属性特征, 设计了数据生成框架 PDGF. 还有许多其他的重要工作, 如: 大数据测试基准 BigDataBench[7]、中科院计算所詹剑锋研发的 BDGS[5]框架、雅虎公司的 YCSB[9,10]等.

与时间有关的属性关联性研究, 需要为时间属性建模 (如自相似性、多分形性等), 通过模拟时间相关属性特征来生成数据. 比如, BURSE[11]工作负载数据生成器, 根据数据的周期性、突发性特征来模拟数据的自相似性; 法国凡尔赛大学 Laurent[12]利用多分形理论在不同单位时间内进行数据仿真; 美国新泽西理工学院 Ansari[13]采用 FARIMA 对 MPEG 中的 I、P 和 B 帧自相关结构进行建模. 较为成熟的产品, 加拿大西蒙菲沙大学 Jiang[14]收集蜂窝数字包数据网络中的业务数据, 运用工具 OPNET 建模和仿真分析.

在大数据关联关系度量领域, 美国加州大学伯克利分校 Speed 教授在《Science》杂志上发表论文所述, 从庞大数据集中发现数据之间潜在的重要有趣的关系变得十分重要, 21 世纪将是关联性学习的时代[15]. 所谓关联性学习就是发现存在于大量数据集中的关联关系或相关关系, 从而描述一个事物中某些属性同现的规律和模式. 这种同现关系可能表现为具有严格确定性的函数显示表达形式, 也可能是客观对象之间确实存在, 但在数量上不是严格对应的依存关系, 也可能是完全不存在内在联系的虚假相关关系[16].

对于多表之间的关联性研究, 加拿大多伦多大学 Rabl 等[17]提出了一种方法, 例如一个网络学习管理系统 (见图 1), 其中主要包括三个表: 学生信息表 (主键为 studentid)、课程信息表 (主键为 courseid)、学生选课信息表 (主键为 scid, 外键为 studentid 和 courseid). 首先生成学生选课信息表, 然后根据学生选课信息表中的 studentid 和 courseid 分别生成学生信息表和课程信息表. 这样能保证学生选课信息表中的 studentid 来自学生信息表, courseid 来自课程信息表. PDGF[6]框架中对多表之间的关联也采用该方法.

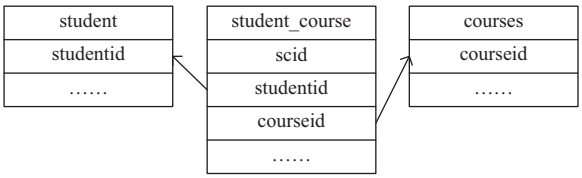


图 1 网络学习管理系统的一个实例

以上这些多表之间的关联方法、单属性刻画方法, 属性关联方法, 都已构建了应用工具. 但是, 对非时间属性相关性质研究中仍存在构建属性关联的困难问题. 本文依据大数据关联关系度量技术, 针对非时间属性的相关性, 提出 TDSA 算法, 逼真生成不同应用背景下

表格中非时间属性对数据.

2 理论基础

2.1 相关性度量标准: MIC

变量对之间的相关性度量, 在大数据分析领域已经有很多的研究[18], 比如, 识别线性关系的 Person 相关系数, 识别单调函数的 Spearman 相关系数和 Kendall 相关系数等等. 在《Science》杂志上, Reshef[19]等提出 MIC 度量方式. 此方法不仅能刻画线性关系, 还能很好地度量非线性关系, 甚至是多种函数的叠加, 具有广泛性; 对于不同关系类型, 若噪声相同, 则 MIC 值也相同, 具有公平性.

假设有 n 个变量对的数据集 D, 根据坐标轴把 D 分为$(x\times y)$等分表示为 G, 用动态规划算法求解的每次结果为$D|_G$, 那么 D 按照 G 这种划分方式的最大互信息为:

$$I^*(D,x,y)=\max I(D|_G) \tag{1}$$

根据划分的方式不同, 可以得到一个的矩阵, 对这个矩阵标准化得到式 (2).

$$M(D)_{x,y}=\frac{I^*(D,x,y)}{\log\min\{x,y\}} \tag{2}$$

在网格划分细度下, 矩阵中的最大值即为 MIC 值:

$$\mathrm{MIC}(D)=\max_{xy<B(n)}\left\{M(D)_{x,y}\right\} \tag{3}$$

由于$0\le I^*(D,x,y)\le\log\min\{x,y\}$, 可得$0\le\mathrm{MIC}(D)\le 1$. 当 MIC 值越接近 1 表示相关性越强, 反之越弱.

2.2 分布拟合模型: SE 分布

Zipf-like 分布广泛用于描述频数与其排名的分布特征. 但是美国俄亥俄州立大学张晓东等[1,3]对 Web 工作负载上不同类型的 16 个数据集进行分析, 发现 Zipf-like 分布不适合描述此分布特征, 而 SE 分布能更好地刻画.

Zipf 分布函数为:

$$y=\frac{c}{x^a} \tag{4}$$

为了方便用最小二乘法拟合, 将函数变换成:

$$\ln y=\ln c-a\ln x \tag{5}$$

SE 分布的分布函数为:

$$y=e^{-(\frac{x}{x_0})^c} \tag{6}$$

其中 c 为广延参数, 其参数范围在 (0, 1), x_0 为尺度参数. 为方便用最小二乘法拟合, 将分布函数变换成式 (7).

$$y^c=-a\ln x+b \tag{7}$$

其中$a=x_0^c$, $b=y_1^c$.

MIC 值具有广泛性和公平性, 本文将考虑用 MIC 值将作为 TDSA 中度量属性间关联性的依据. Zipf 分布广泛用于描述频数与其排名的分布特征, 张晓东证明 SE 分布具有更好的效果[3]. 本文提出的 TDSA 算法将分别采用 Zipf 分布和 SE 分布作为描述属性间关系的函数, 最终通过实验对比, 采用了 SE 分布.

3 基于属性关联的数据逼真生成算法

本文提出基于属性关联的数据逼真生成算法 TDSA. 该算法分为三个阶段: 第一阶段, 提取真实数据集模型参数; 第二阶段, 设置仿真数据集的规模, 检验规模设置的合理性, 生成模型为下一阶段准备; 第三阶段, 生成一个带约束条件的二维矩阵, 生成填充矩阵的方式, 生成逼真数据集. 算法描述如图 2 所示. 其中步骤 1 至步骤 6 是第一阶段, 步骤 7 至步骤 10 为第二阶段, 步骤 11 至步骤 13 为第三阶段.

3.1 提取模型参数

首先从数据集中提取关键属性, 做两重频数统计和一次在群体上的平均总和统计. 然后, 根据统计结果计算 MIC 值来评估相关性, 并采用 SE 分进行拟合, 提取出模型参数.

下面将采用被广泛使用的 MovieLens-1M 数据集说明此提取过程. MovieLens-1M 数据集中用户对电影评分表, 此表包含 6040 位用户对 3900 部电影的 1 000 209 条评分记录. 此表由用户 id、电影 id、评分等级和时间戳构成. 提取出用户 id 和电影 id 这一对非时间属性对. 对用户 id 做频数统计得到用户的活跃度, 电影 id 做频数统计得到电影的流行度. 活跃度降序排列得到相应的排名, 流行度降序排列得到相应的排名. 对活跃度做频数统计得到活跃度与其出现的频数, 对流行度做频数统计得到流行度与其出现的频数. 根据上述的操作之后得到 4 个关系: 活跃度与其排名关系; 活跃度与其频数的关系; 活跃度簇与其流行度平均总和关系, 流行度与其排名关系.

用 MIC 值对上述的 4 个关系进行相关性度量, 然后用 SE 分布函数刻画这 4 个关系, 就可以提取模型参数.

3.2 检验规模合理性

如果这两个属性间存在单向选择的关系, 比如用户看电影, 用户买商品, 用户听音乐等等, 那么此类关系都可以采用下面的方法来提取关联关系. 此关联关

系在算法中用于检验规模的合理性.

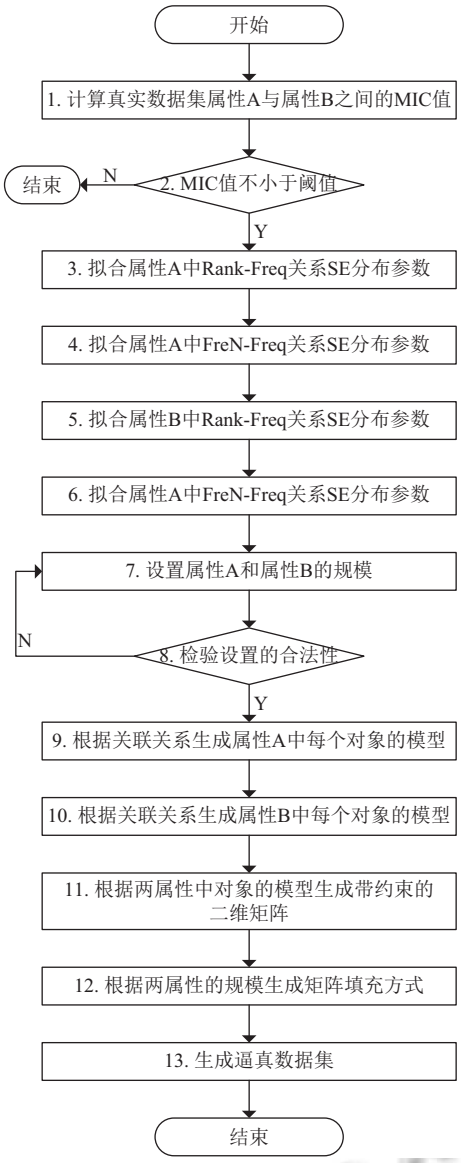


图 2 TDSA 算法流程图

以 MovieLens-1M 数据集为例说明此提取过程. 通过用户活跃度与其频数关系, 可以得到用户活跃度对应的频数 (*FreqFreq*), 所有活跃度对应的频数总和是用户数量 (*UserCount*).

$$\sum FreqFreq = UserCount \tag{8}$$

通过电影流行度与其频数关系, 可以得到电影流行度对应的频数 (*PopuFreq*), 所有流行度对应的频数总和是电影数量 (*ItemCount*).

$$\sum PopuFreq = ItemCount \tag{9}$$

因为真实数据集中每条记录包含一个 Userid 和一个 Itemid, 所以所有用户活跃度总和要和电影流行度总和相等.

$$\sum Ferq = \sum Popu \tag{10}$$

如果给出用户的规模和电影的规模, 那么可以根据此关联关系得到每一位用户的模型和每一部电影模型. 用户模型包括用户 id 和用户活跃度, 电影模型包括电影 id 和电影流行度.

3.3 生成逼真数据集

以 MovieLens-1M 数据集为例, 在二维矩阵中横轴是用户、纵轴代表电影, 交叉式的生成一行或者一列数据. 在生成一个用户的数据时可能需要多次填充用一部电影, 此时是按照从大到小顺序补充. 此填充方式会对数据的噪点产生作用, 并不影响总体上刻画数据特征.

4 实验结果与分析

为了验证 TDSA 是否能有效刻画真实数据集中的数据特征, 实验分为三步: 第一步, 验证 4 个关系关联度的度量, 此关联度用 MIC 值来评估; 然后, 对这 4 个关系的拟合效果比较, 此效果采用决定系数 R^2 进行评估; 最后, 验证生成数据集的逼真效果, 此效果采用逼真数据集模型参数和真实数据集模型参数进行对比.

4.1 数据集描述

实验选取 4 个真实数据集: MovieLens-1M、MovieLens-20M、Amazon-Movie、Amazon-Music. 这些数据集具有较好的代表性. 主要表现在: (1) 数据集来源于可靠而权威的机构或组织, 比如, 明尼苏达大学的社会计算研究; (2) 在各自所在的应用领域内, 数据作为常用数据源被多次使用, 如 MovieLens 数据集在推荐系统实验中广泛使用; (3) 来自同一系统的不同数据集不同大小, 不同时间段. 比如 MovieLens 不同时期不同大小的数据 1 M 和 20 M 数据集, 亚马逊的用户对电影和音乐的评论信息. 表 1 对各个数据集进行了简单的介绍.

表 1 真实数据集

数据集	评价主体数	被评价主体数	评价数
MovieLens-1M	6040	3900	1 000 209
MovieLens-20M	138 493	27 278	20 000 263
Amazon-Movie	2 088 620	200 940	4 607 047
Amazon-Music	478 235	266 414	836 006

MovieLens-1M 数据集是结构化数据, 包含 2003 年

2 月期间 6040 位用户对 3900 部电影的 1 000 209 条评分记录.

MovieLens-20M 数据集是结构化数据, 包含 1995 年 1 月至 2015 年 3 月期间 138 493 位用户对 27 278 部电影的 20 000 263 条评分记录.

亚马逊电影评论 (Amazon-Movie) 数据集是半结构化数据集, 包含 1996 年 5 月到 2014 年 7 月期间的 4 607 047 条评论记录.

亚马逊数字音乐评论 (Amazon-Music) 数据集是半结构化数据集, 包含 1996 年 5 月到 2014 年 7 月期间的 836 006 条评论记录.

4.2 实验结果与分析

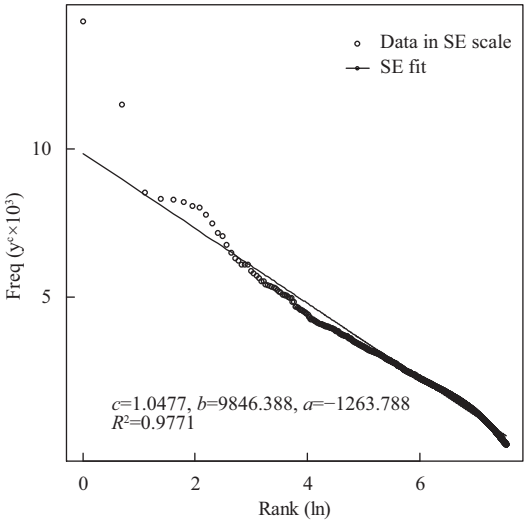
关联模型的 4 个关系表示为, Rank-Freq, Rank-Popu, FreN-Freq, Popu-Freq. 以 MovieLens-1M 数据集为例, 与此相应的关系分别是: 活跃度与其排名关系、流行度与其排名关系、活跃度与其频数的关系、流行度与其频数关系.

4 个数据集的 4 个关系 MIC 值, 实验结果如表 2 所示. 实验结果表明, MIC 值取值一般在 0.7 以上, 说明用 MIC 度量的这 4 个关系的相关性比较强. 特别是前两个关系, MIC 值接近于 1, 说明有很强的相关性.

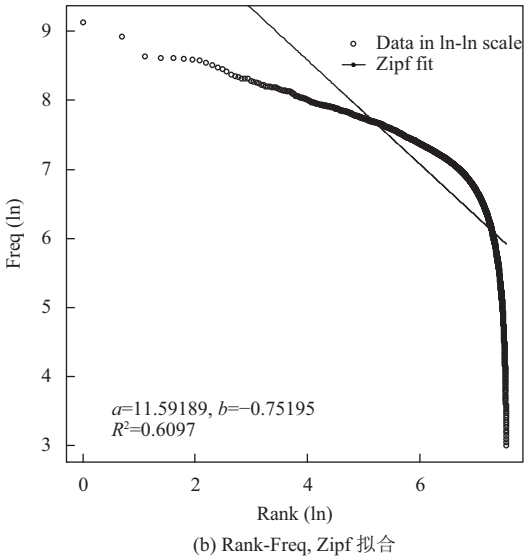
表 2 4 个数据集中 4 个关系的 MIC 值

关系对	MovieLens-1M	MovieLens-20M	Amazon-Movie	Amazon-Music
Rank-Freq	1.000	1.000	1.000	1.000
Rank-Popu	1.000	1.000	1.000	1.000
FreN-Freq	0.702	0.871	0.795	0.907
Popu-Freq	0.525	0.541	0.783	0.825

在 4 个数据集中选取 MoiveLens-20M 数据集来说明 SE 分布于 Zipf 分布拟合效果如图 3 至图 6 所示. 实验结果表明, 对这 4 个关系的拟合, SE 分布比 Zipf 分布的决定系数更大, 说明 SE 分布比 Zipf 分布能更有效地刻画这 4 个关系. 验证 TDSA 生成数据集的逼真效果, 验证方法是提取逼真数据集的模型参数和真实数据集的模型参数, 然后把这两组参数进行对比. 以 Amazon-Movie 数据集为例, 实验结果如表 3 所示, c, b, a 代表 SE 分布的相关参数, T 代表真实数据集, S 在逼真生成的数据集. 从实验结果可以看出, 逼真数据集模型参数和真实数据集模型参数完全一样. 在其他 3 个数据集的实验结果也是同样的情况, 逼真数据集模型参数和真实数据集模型参数完全一样. 表明当模型一样时, TDSA 能生成和真实数据集有相同特征的逼真数据集, 即说明 TDSA 能够有效地模拟真实数据集数据特征.



(a) Rank-Freq, SE 拟合

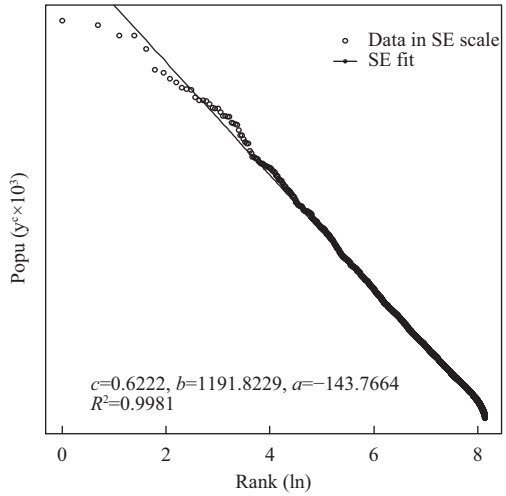


(b) Rank-Freq, Zipf 拟合

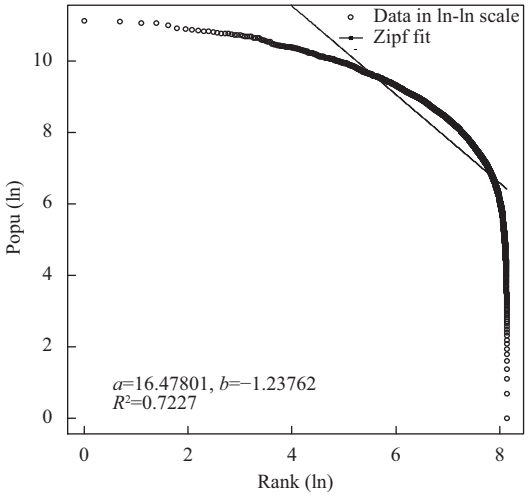
图 3 MovieLens-20M 数据集 Rank-Freq 散点图

5 结论

属性间关联特征的构建是表格数据生成的关键, 本文提出的数据逼真生成算法 TDSA, 它首先分析真实数据集提取出关联模型, 然后设置数据规模, 最后在一个代约束的二维矩阵中生成数据. TDSA 算法可以在一定程度保持真实数据特征, 有助于对表格数据的逼真生成. 但大数据软件的评测, 存在许多不同的实际应用背景, 对于不同的数据生成速度及数据的分布式分发等数据提供问题是今后努力的研究工作.

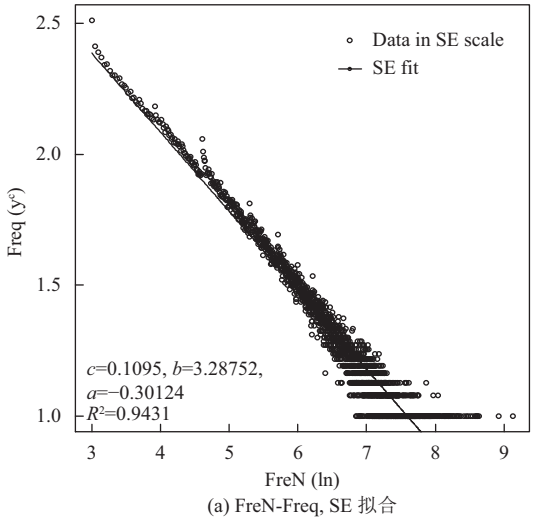


(a) Rank-Popu, SE 拟合

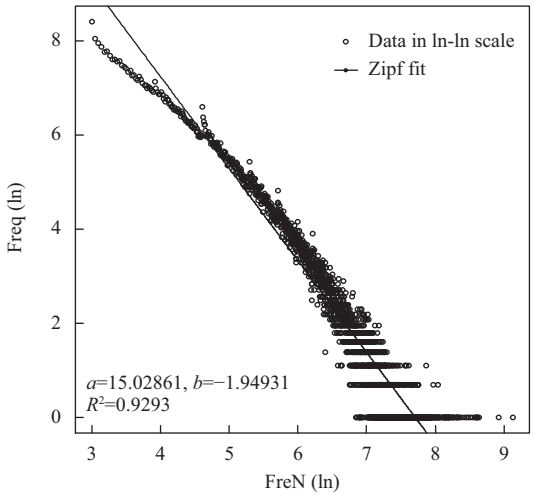


(b) Rank-Popu, Zipf 拟合

图 4 MovieLens-20M 数据集 Rank-Popu 散点图

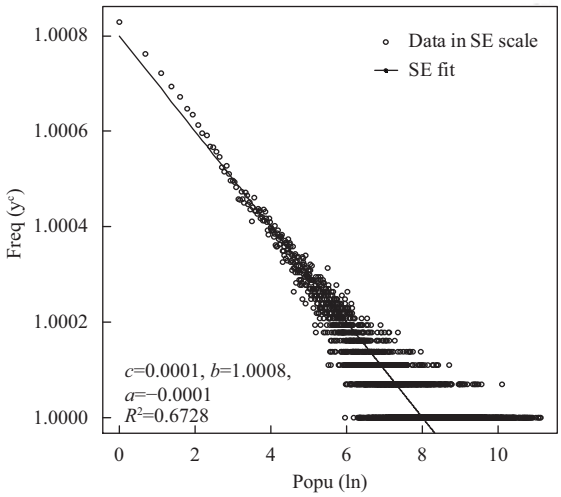


(a) FreN-Freq, SE 拟合

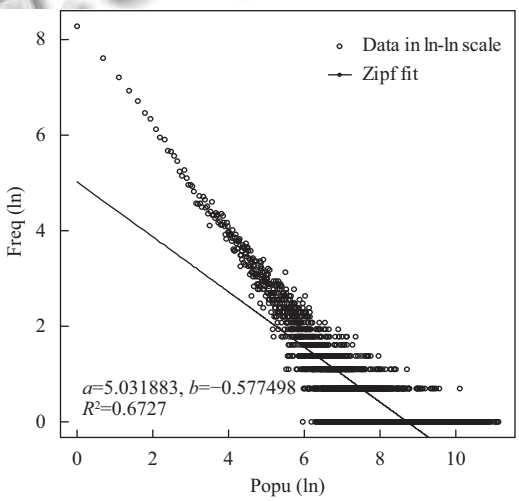


(b) FreN-Freq, Zipf 拟合

图 5 MovieLens-20M 数据集 FreN-Freq 散点图



(a) Popu-Freq, SE 拟合



(b) Popu-Freq, Zipf 拟合

图 6 MovieLens-20M 数据集 Popu-Freq 散点图

表 3 Amazon-Movie 实验数据

关系对	MIC		*c*		*b*		*a*	
	T	S	T	S	T	S	T	S
Rank-Freq	1.000	1.000	0.7959	0.7959	606.4566	606.4566	92.6307	92.6307
Rank-Popu	1.000	1.000	0.6464	0.6464	377.8185	377.8185	49.7366	49.7366
FreN-Freq	0.795	0.795	0.0001	0.0001	1.0011	1.0011	0.0002	0.0002
Popu-Freq	0.783	0.783	0.0001	0.0001	1.0011	1.0011	0.0001	0.0001

参考文献

1 Guo L, Tan EH, Chen SQ, *et al*. The stretched exponential distribution of internet media access patterns. Proceedings of the Twenty-Seventh ACM Symposium on Principles of Distributed Computing. Toronto, Canada. 2008. 283–294.

2 韩筱璞, 汪秉宏, 周涛. 人类行为动力学研究. 复杂系统与复杂性科学, 2010, 7(2): 132–144.

3 Guo L, Tan EH, Chen SQ, *et al*. Analyzing patterns of user content generation in online social networks. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, France. 2009. 369–378.

4 Busari M, Williamson C. ProWGen: A synthetic workload generation tool for simulation evaluation of Web proxy caches. Computer Networks, 2002, 38(6): 779–794. [doi: 10.1016/S1389-1286(01)00285-7]

5 Ming ZJ, Luo CJ, Gao WL, *et al*. BDGS: A scalable big data generator suite in big data benchmarking. In: Rabl T, Raghunath N, Poess M, *et al*, eds. Advancing Big Data Benchmarks. Cham, Swizerland: Springer, 2014. 138–154.

6 Rabl T, Frank M, Sergieh HM, *et al*. A data generator for cloud-scale benchmarking. Proceedings of the Second TPC Technology Conference on Performance Evaluation, Measurement and Characterization of Complex Systems. Berlin, Heidelberg, Germany. 2010. 41–56.

7 詹剑锋, 高婉铃, 王磊, 等. BigDataBench: 开源的大数据系统评测基准. 计算机学报, 2016, 39(1): 196–211. [doi: 10.11897/SP.J.1016.2016.00196]

8 Gray J, Sundaresan P, Englert S, *et al*. Quickly generating billion-record synthetic databases. Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data. Minneapolis, MN, USA. 1994. 243–252.

9 Cooper BF, Silberstein A, Tam E, *et al*. Benchmarking cloud serving systems with YCSB. Proceedings of the 1st ACM Symposium on Cloud Computing. Indianapolis, IN, USA. 2010. 143–154.

10 Abramova V, Bernardino J, Furtado P. Evaluating Cassandra scalability with YCSB. International Conference on Database and Expert Systems Applications. Springer International Publishing 2014. 199–207.

11 Yin JW, Lu XJ, Zhao XK, *et al*. BURSE: A bursty and self-similar workload generator for cloud computing. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(3): 668–680. [doi: 10.1109/TPDS.2014.2315204]

12 Akrour N, Mallet C, Barthes L, *et al*. A rainfall simulator based on multifractal generator. EGU General Assembly Conference Abstracts. Vienna, Austria. 2015.

13 Ansari N, Liu H, Shi YQ, *et al*. On modeling MPEG video traffics. IEEE Transactions on Broadcasting, 2002, 48(4): 337–347. [doi: 10.1109/TBC.2002.806794]

14 Jiang M, Nikolic M, Hardy S, *et al*. Impact of self-similarity on wireless data network performance. Proceedings of IEEE International Conference on Communications. Helsinki, Finland. 2001. 477–481.

15 Speed T. A correlation for the 21st century. Science, 2011, 334(6062): 1502–1503. [doi: 10.1126/science.1215894]

16 Fan JQ, Han F, Liu H. Challenges of big data analysis. National Science Review, 2014, 1(2): 293–314. [doi: 10.1093/nsr/nwt032]

17 Rabl T, Lang A, Hackl T, *et al*. Generating shifting workloads to benchmark adaptability in relational database systems. In: Nambiar R, Poess M, eds. Performance Evaluation and Benchmarking. Berlin Heidelberg, Germany: Springer, 2009. 116–131.

18 钱宇华, 成红红, 梁新彦, 等. 大数据关联关系度量研究综述. 数据采集与处理, 2015, 30(6): 1147–1159.

19 Reshef DN, Reshef YA, Finucane HK, *et al*. Detecting novel associations in large data sets. Science, 2011, 334(6062): 1518–1524. [doi: 10.1126/science.1205438]