Vol.2020-AL-177 No.2
2020/3/16

データ研磨アルゴリズムの挙動と収束性の解析

宇野 毅明[1,a)]

概要：データ研磨は, データのゆらぎを除去することで中小規模の構造を明確化し, マイニングアルゴリズムの効率や精度を高める手法である. 例えば, ネットワーククラスタリングの場合, グラフの密な部分をクリークにし, 疎な部分を独立集合とすることで, クラスタ構造を明確にする. 通常のクラスタリングが, 比較的大きなクラスタを見つけることが上手であるのに対して, データ研磨によるクラスタリングは, 小さくてまとまりの良いクラスタを網羅的に, かつ独立性高く, 適切な個数で見つけることができる. 実際にそのクラスタは意味解釈がしやすく, 新聞記事やツイッターのクラスタリングによりトピックを網羅的に見つけることが可能である. また, アルゴリズムの挙動も極めて安定しており, 大規模なデータでも数十の反復で収束に至ることがほとんどである. データ研磨のアルゴリズムの基本的なデザインはシンプルであり, 根拠となるデータに対する観察も明らかである. 一方, 収束性や得られた解と数理構造との対応は不明瞭であり, いわば実行可能仮説寄りのモデルである. 本稿では, データ研磨アルゴリズムの数理的な側面を明らかにすべく, その挙動に関する数理的な解析や確率的な解析を行い, データ研磨アルゴリズムの現実データでの挙動の良さを数理的に裏付ける.

キーワード：データクリーニング，データ解析，クリーク列挙，パターン発見，クラスタリング

Analyzing the Behaviour and the Convergence of the Data Polishing Algorithm

Abstract: Data polishing is a methodology to increase the efficiency and accuracy of data mining algorithms by removing the ambiguities of the data to clarify the small to middle size structures of the data. For example, data polishing clarifies the cluster structures in the given network data by changing a dense subgraph to a clique, and a sparse subgraph to an empty subgraph on the other hand. Clustering algorithms with the use of data polishing can find small concentrated clusters, relatively more completely, with keeping the independency of each cluster from the others so that we get proper number of clusters, while ordinary clustering algorithms are good at finding few relatively big clusters. Actually, the clusters given by these algorithms are good to interpretation, so that we can enumerate topics in the news paper articles or twitter data. The behaiour of the algorithm is quite stable and efficient such that the algorithm converges in few tens of steps even for very large scale data. The basic design of data polishing algorithm is quite simple, and the observation on the data is clear. On the other hand, there is no proof for the convergence, and the correspondence between the solutions and the mathematical structures are not clear. It is a kind of "feasible hypothesis based model". In this paper, we aim to clarify the mathematical aspects of data polishing algorithms. We analyze the structures of algorithms in mathematical ways that supports the practical efficiency of data polishing algorithms.

Keywords: data cleaning, data analysis, clique enumeration, pattern mining, clustering

1. Introduction

クラスタリングは, 教師無し学習における基本的かつ中心的な手法である. 科学分野から産業・商業応用まで幅広く使われており, 理論面の研究も進んでいる. クラスタリングは, データを分けることであるが, 同時に, 似たものが集まったグループを見つけることでもある. 両者は異なるものであるが, 同時に混在した目的として説明されることが多く, これまでの研究なり産業応用なりで, 両者の違いを

[1] 国立情報学研究所
2-1-2, Hitotsubashi, Chiyoda, Tokyo 101–8430, Japan
[a)] uno@nii.ac.jp

区別してモデル化なりデータ解析の方針なりを説明している文献は少ないように見える.

一般的にだが, 自然科学や商業での顧客分析など, データの大きな構造が知りたいときには, 「分ける」目的が主体となることが多いように思われる. また, 自動推薦や広告など, Web サービス分野や商業分野での顧客, ユーザの分類には, ユーザがなんらかのクラスタに所属していないと属性付与ができないため, 「分ける」アプローチが重要となる. 一方で, 新聞記事や SNS のクラスタからトピックを見つける, 顧客グループやコミュニティの特徴を捉えたいような場合は, 似た記事, 似た顧客だけが集まっていたほうが共通性が浮き上がりやすいため, 「似たものが集まったグループ」のアプローチが重要となる. 同じマーケッティングでも, 担当者が顧客を深く理解したい, 新商品開発や店舗レイアウト変更のために, 顧客の典型的な嗜好や動きが知りたい, というときなどに力を発揮する. 違う見方をすれば, 人間がクラスタに関しての知識を, 自らが考えることによって得たいと思う場合には, 似たものが集まったグループを用いるのが適切であり, コンピュータによる自動化で何かを行いたい場合には, 分けるアプローチのほうが適していることが多い, という見方もできる. もちろん, これはすべてではなく, 特徴の自動抽出をしたいときなどは似たものが集まったグループ型が適切となる.

既存のクラスタリングアルゴリズムの研究では, 多くが「分ける」問題を扱っている. 著名なアルゴリズム, k-means[5] や Girvan-Newman[1], グラフカット [3] などもこのタイプである. 一方, 似たものが集まったグループを見つけ, クラスタに所属しない要素があっても良い, とするタイプのクラスタリングは, むしろマイニングアルゴリズムとして研究されていることが多く, 例えばコミュニティマイニング, ランダムウォークを用いたコミュニティ発見, 列挙アルゴリズムを用いたパターンマイニング的アルゴリズムがその代表である. 「分ける」問題に対するアルゴリズムは, 少数のクラスタに分ける場合においては, 比較的質の高いクラスタが得られることが多く, 科学分野から産業応用まで盛んに用いられている. 一方で「似たもののグループ」を見つけるものに対しては, アルゴリズムの研究は多いものの, 実用上満足のいく解が得られていないためか, 利用の範囲は非常に限定的である. 実用上障害となる点は, 解の再現性がない, 結果が安定しない, 網羅性がない, 解が多すぎて扱えない, などである.

筆者らは「似たもののグループ」を見つける問題に対して, データ研磨というアルゴリズムを提案し, 安定性, 再現性, 網羅性, 解の数に対する問題を一気に解決した [6], [7], [8], [9]. データ研磨は質の面でも高いクラスタを出力するため, 新聞記事や SNS のトピック分類や顧客自動分類のための良い特徴を見つけるためにも使われている [2], [4]. 新聞記事のクラスタを直接観察してみると, 既存手法のクラスタリングが, 粒度も中身もばらばらのクラスタを見つけているのに対し, データ研磨のアルゴリズムは中身が非常にそろった, 適切な大きさのクラスタを見つけていることがわかる. 一般にこのようなクラスタは既存手法が求めるクラスタよりもサイズが小さく数が多いため, このようなクラスタをマイクロクラスタと呼ぶ. また, グラフクラスタリングのために設計したデータ研磨アルゴリズムをグラフ研磨とよぶ.

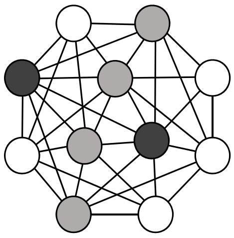

図 1 ある程度大きな密構造 (擬クリーク) に含まれる頂点の組には多くの頂点が共通して隣接する

一方で, データ研磨は, 他の手法と異なり, 解を数理的な言葉で直接的に説明できない. データ研磨のアイディアは, グラフの濃い部分をより濃く, 薄い部分をより薄くするという操作を繰り返して, クラスタ構造を明確化する, というものであり, 直感的にはわかりやすい. 実際には, ある種の反復操作を繰り返し行って収束したものが解になるのであるが, これはアルゴリズム的な説明であり, 解の持つ数理構造を説明したものではない. 既存クラスタリングの多くが, この数理的な説明を解の質の良さの説明のよりどころとしているのに対して, データ研磨のアルゴリズムはここに説明のよりどころがない. それは, データ研磨アルゴリズムの課題となりうるだろう. 収束性に保証がないところも, 数理的な説明のよりどころにできない点の一つである.

本稿では, データ研磨のもつこれらの数理的に不明瞭な部分を解決すべく, アルゴリズムの収束性の根拠となるようないくつかの定理を証明する. 具体的には, 密な部分グラフがクリークになること, 密な部分グラフに属さない枝が, 高い確率で除去されることなどを証明する.

2. 記法

グラフの**クリーク**とは，そのグラフの頂点部分集合で，全ての頂点間に枝がある物である．クリークは通常部分グラフとして定義されることが多いが，ここでは頂点集合で定義していることを注意しておく．他のクリークに含まれないクリークを**極大クリーク**という．

グラフの密度を，全ての頂点対に対する枝の割合，つまり $\frac{|E|}{|V|(|V|-1)/2}$ とする．頂点集合の密度を，その頂点集合が誘導する部分グラフの密度とする．ただし，頂点集合 U が誘導する部分グラフとは，G の枝で U の頂点同士を結ぶ物を集めてできるグラフである．**擬クリーク**とは，ある

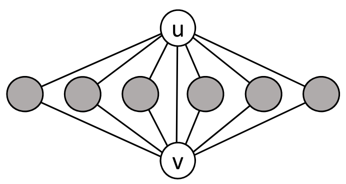

図 2　2 つの頂点は 6 つの共通近傍を持っている

程度以上の密度を持つ頂点集合のことである．密度が δ 以上のクリークのことを δ-**擬クリーク**とよぶ．

G の頂点 v に対し，v と枝で結ばれている頂点 u は，v に**隣接する**といい，そのような u を v の**近傍**という．v の近傍の集合を $N(v)$ と表記する．v の**次数** $d(v)$ は，v に隣接する頂点の数，つまり $|N(v)|$ である．$N[v]$ は $N(v) \cup \{v\}$ のことであり，**閉近傍**という．u と v の両方に隣接する頂点を u と v の**共通近傍**とよぶ．

3. グラフクラスタリングに対するデータ研磨アルゴリズム

データ研磨は, データの不明瞭な部分を修正して明確にすることにより, データが内包する局所的な部分構造を明らかにするデータ解析手法である.

グラフクラスタリングでは，通常，局所的に枝が密である部分をグループ（クラスタ）であると考える．しかし，多くの実データでは密な部分の境界が曖昧であり，取りようによっていくらでも類似するグループ構造が見つかってしまう．逆に，全てのグループがクリークになっていて，グループに含まれない部分は疎になっているのであれば，境界がはっきりしており，明確に構造を発見できる．つまり，密度の高い部分はクリークに，そうでない部分には枝がない，というグラフが，グループ構造が明確になっているグラフと考えることができる．

このような「明確」なグラフは，同じクラスタに含まれる頂点の間には必ず枝があり，同じクラスタに含まれない頂点の間には枝がないようなグラフ，と考えることができる．そこで, グラフ G の 2 つの頂点 v と u が同じクラスタに含まれそうであれば枝があり, そうでないならば枝がないようなグラフ G' を作ることにより, クラスタ構造を明確化しようというのが, データ研磨の狙いである. v と u が同一のクラスタに含まれるかどうかは, 以下の仮説によって推測する.

近傍類似性仮説 v と u が同一のクラスタに含まれるなら, v の閉近傍の集合 $N[v]$ と u の閉近傍の集合 $N[u]$ は類似している. (類似度が θ 以上である)

u と v が同一のクラスタに含まれるならば, u と v はそのクラスタに対応する密な部分グラフに含まれるので, その密な部分グラフの頂点は u と v 両方との間に枝を持つ可能性が高く, 共通隣人の数が増える (図 1). すると, u の閉近傍集合と v の閉近傍集合の類似度が高くなる, ということである. グラフのすべての頂点対 v と u に対して, その閉近傍集合の類似度を $\mathrm{sim}(u,v)$ で表記する. sim は, 例えば閉近傍集合の共通部分 $N[u] \cap N[v]$ や Jaccard 係数 $\frac{N[u]\cap N[v]}{N[u]\cup N[v]}$ で定義する. ここで行うことは, u と v が同一のクラスタに含まれそうであれば枝が張られているグラフ G' を, 近傍類似性仮説から, $\mathrm{sim}(u,v) \geq \theta$ であれば u, v 間に枝があるグラフで定義することである. 2 つの頂点が多くの共通近傍を持つ場合を, 図 2 に示した.

グラフ研磨は, この操作を反復的に行う, つまり G' にも適用してさらに新しいグラフを得て, そのグラフにも適用し, ということを続けて, グラフを明確化する. この反復は, 最終的にグラフが変化しなくなるまで, あるいは一定数の反復を行っても変化がなくならず, 収束の見込みがなくなるまで続ける. このデータ研磨アルゴリズム, 特にグラフ研磨と呼ぶ, は以下のように記述できる.

ALGORITHM GraphPolishing ($G = (V, E)$, sim(), θ, τ)
1. **for** $i := 1$ **to** τ
2. $G' := \{V, \{(u,v) \mid \mathrm{sim}(u,v) \geq \theta\}$
3. **if** $G' = G$ or $i == \tau$ **then output** G; **exit**
4. $G := G'$
5. bf next

著者らの計算実験 [6], [9] で, データ研磨は大きさが小さく類似性が高いものが集まったクラスタを適度な量見つける目的には非常に高い効果が示されている.

4. 研磨操作の性質

k-共通部分研磨や k-Jaccard 研磨によって, 密な部分グラフはクリークに近いグラフになると考えられる. よって, 研磨操作を何回も適用することで, 密な部分グラフがクリークになったグラフが得られると考えられる. この節では, この考察を補強すべく, 研磨操作による密部分グラフの密度の変化に関する定理をいくつか証明する. 以下では, 2 項分布を正規分布とポワソン分布で近似し, その近似した分布によって確率を抑える. 以下, 定理での言明では, 確率を抑えるということは, この近似した確率を抑えるものとする.

定理 1　頂点数 n が十分大きく, 枝が確率 $p < 1/n^{1/2}$ で独立に張られているランダムグラフ G において, 頂点 u と v が $|N[u] \cap N[v]| \geq 6$ を満たす確率は 1/1000 以下である.

証明: v が u と共通隣人を k 人以上持つ確率を考える. 頂点 w が u と v の共通隣人になる確率は p^2 である. よって, u と v の共通隣人数は 2 項分布となり, その期待値は np^2, 分散は $np^2(1-p^2)$ となる. 両者とも十分小さいので, $\lambda = np^2$ のポワソン分布で近似できる.

よって, 不完全ガンマ関数を用いて $P(X \le k) = \Gamma(k+1,\lambda)/k!$ であり, これは $k=3$ のときに 0.9810 よりも大きくなり, $k=5$ のときに 0.9994 よりも大きくなる. よって u と v との共通隣人が 4 人以上である確率は 0.019 以下, 6 人以上である確率は, 0.0006 以下である.

よって, u と v が枝で結ばれ, かつ $|N[u] \cap N[v]| \ge 6$ である確率は, $0.019/n^{1/2}$ 以下, u と v が枝で結ばれてなく, かつ $|N[u] \cap N[v]| \ge 6$ である確率は, 0.0006 以下である.

∎

確率が不完全ガンマ関数であらわされているため, 一般の n と p に対してより精密な解析的な結果を求めることは難しいが, 十分大きな n と十分小さな p に対して, 上記不完全ガンマ関数の値はポワソン分布の確率密度と非常に近くなり, 共通近傍数が確率的に小さいことが証明できる.

定理 2 頂点数 n が十分大きく $(n > 3000)$, 枝が確率 $p, 13/(n-2) \le p \le 1/n^{1/2}$ で独立に張られているランダムグラフ G において, 頂点 u と v に対する $N[u]$ と $N[v]$ の Jaccard 指数が 1/7 以上になる確率は, 1/1000 以下である.

$13(n-2)$ は, 平均次数が 13 以上であることとほぼ同じであることを記しておく.

証明: 頂点 v か u のどちらかのみに隣接する頂点の数が h であるとする. このとき, それ以外の頂点が u と v 両方に隣接する確率は $p^2/(1-2p+p^2)$ であり, その分布は 2 項分布, その期待値は $(n-2-h)p^2/(1-2p+p^2)$ である.

h の期待値は $(n-2)(2p-p^2)$ であり, 分散は $(n-2)(2p-p^2)(1-2p+p^2)$ であるので, 2 項分布を正規分布で近似し, 標準偏差から,

$$h < (n-2)(2p-p^2) - 3.9((n-2)(2p-p^2)(1-2p+p^2))^{1/2},$$
$$h < (n-2)(2p-p^2) - 2.7(n(2p-p^2)(1-2p+p^2))^{1/2},$$
$$h < (n-2)(2p-p^2) - 1.57((n-2)(2p-p^2)(1-2p+p^2))^{1/2}$$

となる確率は 0.00005 以下, 0.0035 以下, 0.0583 以下である.

これらの式の右辺

$$T = (n-2)(2p-p^2) - b((n-2)(2p-p^2)(1-2p+p^2))^{1/2}$$

を調査するため, 定数 b を係数部分に導入すると,

$$\begin{aligned}T &= (n-2)(2p-p^2) - b((n-2)(2p-p^2)(1-2p+p^2))^{1/2} \\ &= 2(n-2)p - (n-2)p^2 \\ &\quad -b((n-2)(2p-p^2-4p^2+2p^3+2p^3-p^4))^{1/2} \\ &= 2(n-2)p - (n-2)p^2 \\ &\quad -b(2p(n-2)-5p^2(n-2)+4p^3(n-2)-p^4(n-2))^{1/2}\end{aligned}$$

である. 定数 t に対して, $p \ge t/(n-2)$ であれば,

$$\begin{aligned}T &= 2(n-2)p - (n-2)p^2 \\ &\quad -b(2p(n-2)-5p^2(n-2)+4p^3(n-2)-p^4(n-2))^{1/2} \\ &= 2t - tp - b(2t - 5tp + 4tp^2 - tp^3)^{1/2} \\ &> 2t - tp - b(2t + 4tp^2)^{1/2}\end{aligned}$$

である. t はグラフの次数の期待値とほぼ同じである. ここで, $p \le 1/n^{0.612}, n >= 3000, t >= 13$ を仮定すると, $2t - tp - b(n(2p-p^2)(1-2p+p^2))^{1/2}$ は $n = 3000, p = 13/2998, t = (n-2)p = 13$ のときに最小になる. そのときの T の値は,

$$\begin{aligned}T &= 2t - tp - b((n-2)(2p-p^2)(1-2p+p^2))^{1/2} \\ &> 26 - 13 * 13/2998 - b(13(1.9935)(0.9871))^{1/2} \\ &> 25.943 - b(5.058)\end{aligned}$$

である. よって, $b = 3.9$ のときに, $T > 6$ より真に大きくとなり, $b = 2.7$ のときに $T > 12$, $b = 1.57$ のときに $T > 18$ となる.

まず, u と v が枝でつながれていない場合を考える. $N[u]$ と $N[v]$ の Jaccard 係数が 1/7 以上になる場合は,

(1) $k = 1, h \le 6$
(2) $k = 2, h \le 12$
(3) $k = 3, h \le 18$
(4) $k \ge 4$

のどれかが成り立つ確率よりも小さい. 正規分布で近似したことと, 不完全ガンマ関数の計算により, それぞれの場合がおこる確率は,

(1) $k = 1$ となる確率が 0.140808 以下, $h \le 6$ となる確率が 0.00005 以下であるので, 0.0000088 以下
(2) $k = 2$ となる確率が 0.011708 以下, $h \le 12$ となる確率が 0.0035 以下であるので, 0.0000673 以下
(3) $k = 3$ となる確率が 0.0006489 以下, $h \le 18$ となる確率が 0.0583 以下であるので, 0.0000379 以下
(4) は, 0.0000279 以下

である. よって, すべての和をとると, Jaccard 係数が 1/7 より真に大きくなる確率が 0.0001419 以下であることがわかる.

次に, u と v が枝でつながれている場合を考える. このようなことが起きるのは, 0.0074474 以下の確率である. そのとき, $N[u]$ と $N[v]$ の Jaccard 係数が 1/7 より真に大きくなる場合は,

(1) $k = 0, h < 12$
(2) $k = 1, h < 18$
(3) $k \ge 2$

のどれかが成り立つ確率よりも小さい. それぞれの場合がおこる確率は,

(1) は, $k = 0$ となる確率が 0.846811 以下, $h < 12$ となる

確率が 0.0035 以下であるので, 0.0000221 以下

(2) は, $h < 18$ となる確率が 0.0583 以下, $k = 1$ となる確率が 0.140808 以下であるので, 0.0000612 以下

(3) は, $k \geq 2$ となる確率が 0.012385 以下であるので, 0.0000923 以下

である. よって, すべての和をとると, Jaccard 係数が 1/7 より真に大きくなる確率が 0.000176 以下であることがわかる.

ここで次に, $p \geq 1/n^{0.612}, n >= 3000, t >= 13$ である場合を考える. $2t - tp - b(n(2p - p^2)(1 - 2p + p^2))^{1/2}$ は $n = 3000, p = 1/n^{0.612}, t = (n-2)p = 22.$ のときに最小になり, $b = 4$, $b = 3$, $b = 2.1$ のときに, それぞれ $T > 18$, $T > 24$, $T > 30$, となり, よって $h \leq 18, h \leq 24, h \leq 30$ となる確率はそれぞれ 0.0000318, 0.00135, 0.017864 以下となる.

まず, u と v が枝でつながれていない場合を考える. $N[u]$ と $N[v]$ の Jaccard 係数が 1/7 より大きくなる場合は,

(1) $k \leq 3, h \leq 18$
(2) $k = 4, h \leq 24$
(3) $k = 5, h \leq 30$
(4) $k = 6, h \leq 36$
(5) $k \geq 7$

のどれかが成り立つ確率よりも小さい. それぞれがおこる確率は,

(1) は, $k \leq 3$ となる確率が 0.981011 以下, $h \leq 18$ となる確率が 0.0000318 以下であるので, 0.0000312 以下

(2) は, $k = 4$ となる確率が 0.01532 以下, $h \leq 24$ となる確率が 0.00135 以下であるので, 0.0000207 以下

(3) は, $k = 5$ となる確率が 0.00307 以下, $h \leq 30$ となる確率が 0.01787 以下であるので, 0.0000549 以下

(4) は, $k = 6$ となる確率が 0.00051176 以下, $h \leq 36$ となる確率が 0.102043 以下であるので, 0.0000523 以下

(5) $k \geq 7$ が成り立つ確率は 0.00008325 以下である.

よって, すべての和をとると, Jaccard 係数が 1/7 より真に大きくなる確率が 0.0002424 以下であることがわかる.

次に, u と v が枝でつながれている場合を考える. 枝でつながれる確率は, 0.0182575 以下である. $N[u]$ と $N[v]$ の Jaccard 係数が 1/7 以上になる場合は,

(1) $k \leq 1, h \leq 18$
(2) $k = 2, h \leq 24$
(3) $k = 3, h \leq 30$
(4) $k = 4, h \leq 36$
(5) $k \geq 5$

のどれかが成り立つ確率よりも小さい. それぞれがおこる確率は,

(1) は, $k = 1$ となる確率が 0.3681 以下, $h \leq 18$ となる確率が 0.0000318 以下であるので, 0.000000214 以下である.

(2) は, $k = 2$ となる確率が 0.18396 以下, $h \leq 24$ となる確率が 0.00135 以下であるので, 0.00000454 以下となる.

(3) は, $k = 3$ となる確率が 0.061324 以下, $h \leq 30$ となる確率が 0.01787 以下であるので, 0.0000201 以下となる.

(4) は, $k = 4$ となる確率が 0.015327 以下, $h \leq 36$ となる確率が 0.102043 以下であるので, 0.0000286 以下となる.

(5) は, $k \geq 5$ となる確率が 0.003664 以下であるので, 0.000121 以下となる.

よって, すべての和をとると, Jaccard 係数が 1/7 以上になる確率が 0.0002424 以下であることがわかる. ∎

この定理は, 確率の上界を押さえているのだが, n と p の閉じた式の形で上界を表現することが難しいため, 上述のように多少荒い算定を行っている. そこで, n と $(n-2)p$ (次数) を変化させたときの実際の確率を表 3 にまとめた. 実際の値を見ると, グラフが疎であるときに Jaccard 係数が 1/7 を超える確率は非常に小さいことがわかる.

次に, グラフ研磨の操作によって枝が張られる場合について解析を行う. 以下の定理で, 大きさ n のクラスタ内に一様にランダムに枝が張られている場合, その共通隣人の数が高い確率で, ある一定以上になることを示す.

定理 3 頂点数 n, 枝が確率 p で独立に張られているランダムグラフ G において, $(n-2)p^2 \geq 5$ であれば, 頂点 u と v が $|N[u] \cap N[v]| \leq (n-2)p^2 - 3.1((n-2)p^2(1-p^2))^{1/2}$ を満たす確率は 1/1000 以下である.

証明: v が u と共通隣人を k 人以下しか持たない確率を考える. 頂点 w が u と v の共通隣人になる確率は p^2 である. よって, u と v の共通隣人数は 2 項分布となり, その期待値は $(n-2)p^2$, 分散は $(n-2)p^2(1-p^2)$ となる. $(n-2)p^2$ が 5 以上であれば, これは正規分布で近似できる.

$p^2(1-p^2)$ は, $p^2 = 0.5$ で最大値となり, その値は 1/4 であるので, この分布の標準偏差 $((n-2)p^2(1-p^2))^{1/2}$ は, $n^{1/2}/4$ より必ず同じか小さい. 正規分布において, 標準偏差の 3.1 倍以上, 平均値より値が小さくなる確率は 1/1000 未満であるので, u と v の共通隣人の数が $(n-2)p^2 - 3.1((n-2)p^2(1-p^2))^{1/2}$ より小さくなる確率は 1/1000 未満であり, よって題意を得る. ∎

定理 4 頂点数が 45 以上, 枝が確率 $p \geq 1/2$ で独立に張られているランダムグラフ G において, 頂点 u と v が $|N[u] \cap N[v]| < 6$ を満たす確率は 1/1000 以下である.

証明: 上記定理と同様に, v が u と共通隣人を k 人以下しか持たない確率を考える. まず, v と u が枝で結ばれていない場合を考える. これがおこる確率は $(1-p)$ である. このとき, u と v の共通隣人の数が 6 未満である確率は, $(n-2)p^2 \geq 10.75$, 標準偏差が 1.63936 より小さいことから, 0.0009419 以下である.

次に, v と u が枝で結ばれているときを考える. これが

図 3 ランダムグラフの頂点数 n と平均次数 d の変化に対する，2 つの頂点の閉近傍の Jaccard 係数が 1/7 を超える確率

$d \diagdown n$	1000	2000	4000	8000	16000	32000	64000	128000
5	0.006491802	0.003187322	0.001578883	0.000785738	0.000391935	0.000195740	0.000097811	0.000048897
10	0.000978246	0.000351608	0.000143965	0.000064347	0.000030306	0.000014699	0.000007231	0.000003592
20	0.000035067	0.000003460	0.000000408	0.000000079	0.000000021	0.000000007	0.000000003	0.000000009
40	0.006745979	0.000208886	0.000004621	0.000000489	0.000000009	0.000000005	0.000000002	0.000000000
80	0.628150236	0.109967285	0.006393139	0.000196616	0.000004335	0.000000483	0.000000009	0.000000005
160	0.999999253	0.988410352	0.622203291	0.107687697	0.006216719	0.000190481	0.000004192	0.000000481
320	1.000000000	1.000000000	0.999999197	0.988094154	0.619229818	0.106547903	0.006128509	0.000187414
$1/n^{4/5}$	0.007625058	0.003492859	0.001474704	0.000561281	0.000187803	0.000053827	0.000012878	0.000002504
$1/n^{3/4}$	0.005515838	0.001797775	0.000466104	0.000091026	0.000012766	0.000002118	0.000000078	0.000000010
$1/n^{2/3}$	0.000978246	0.000091819	0.000004160	0.000000079	0.000000001	0.000000009	0.000000002	0.000000004
$1/n^{1/2}$	0.000691351	0.000662736	0.000642658	0.000628461	0.000618422	0.000611324	0.000606304	0.000602755

おこる確率は p である．このとき, u と v の共通隣人の数が 4 未満である確率は, 0.0000192 以下である．

よって題意は示された．■

5. まとめ

本稿では, グラフクラスタリングに対するデータ研磨アルゴリズムの収束性の良さを明らかにすべく, その確率的な挙動を解析した．まず, 疎でクラスタ構造がないランダムグラフ, 頂点数 n に対して平均次数が $n^{1/2}$ 未満であるようなグラフでは, データ研磨アルゴリズムは非常に低い確率でしか枝を張らない, つまり任意の 2 頂点をあやまって同一クラスタに所属すると認識する確率が非常に低いことを示した．続いて, このような疎なグラフに枝密度がある程度以上高いクラスタが埋め込まれた場合, クラスタ内の２つの頂点は非常に高い確率で枝が張られる, つまりデータ研磨アルゴリズムは, この２つの頂点を高い確率で同一クラスタに所属していると認識することを示した．また, それぞれの頂点が複数のクラスタに所属しうるソフトクラスタリングのインスタンスにおいても, データ研磨アルゴリズムは非常に高い確率で同一クラスタに所属している頂点を認識することが示された．

データ研磨アルゴリズムは, 実際には本稿で証明したようなある程度の大きさグラフで, ある程度の大きさのクラスタではないクラスタも, 実用上では効率よく見つけ, 実際に収束している．このような場合の解析には, より精細, あるいはある程度の荒さを持つ代わりにより本質をついたようなモデルの考案と, それの確率的解析が必要になるだろう．

謝辞

この研究は科学技術振興機構 CREST, 課題番号 JP-MJCR1401 の補助を受けている．

参考文献

[1] M. Girvan and M. E. J. Newman: Community Structure in Social and Biological Networks, Proc. Natl. Acad. Sci. USA **99**, pp. 7821–7826 (2002)
[2] Takako Hashimoto, Takeaki Uno, Tetsuji Kuboyama, Kilho Shin, Dave Shepard: Time Series Topic Transition Based on Micro-Clustering, BigComp 2019: 1-8 (2019)
[3] G. Karypis, V. Kumar: METIS–Unstructured Graph Partitioning and Sparse Matrix Ordering System, version 2.0. (1995).
[4] 中原孝信, 丸橋弘明, 羽室行信, 宇野毅明: グラフ研磨を利用した顧客クラスタリングによる多様性を考慮した特徴抽出オペレーションズ・リサーチ 64(2) 102 - 109 2019 年 2 月
[5] H. Steinhaus: Sur la division des corps materiels en parties, Bull. Acad. Polon. Sci. 4 (12), pp. 801–804, (1957) (French).
[6] Takeaki Uno, Hiroki Maegawa, Takanobu Nakahara, Yukinobu Hamuro, Ryo Yoshinaka, and Makoto Tatsuta: Micro-Clustering by Data Polishing, IEEE Big Data 2017 (2017).
[7] 宇野毅明, 岩崎幸子, 中原孝信, 中元政一, 羽室行信: 乱数シード依存のクラスタリング手法の安定化に対するアプローチ, 人工知能基本問題研究会 105, pp. 58–62 (2018).
[8] 宇野毅明, 岩崎幸子, 中原孝信, 中元政一, 羽室行信: データ研磨によるコンセンサスクラスタリングの精緻化, 人工知能基本問題研究会 106, pp. 43–50 (2018).
[9] 宇野毅明, 中原孝信, 前川浩基, 羽室行信: データ研磨によるクリーク列挙クラスタリング, 第 146 回情報処理学会アルゴリズム研究会 146, pp. 1–8 (2014).