完全 k-concealment 匿名化を求める精度の高いアルゴリズムの評価

伊藤 聡志[1,2,a)]　菊池 浩明[3,b)]

概要：匿名加工は，個人が識別されることを防ぐために個人情報を加工する技術である．企業などの組織が収集したビッグデータ (顧客情報や位置情報など) を分析・利活用することにより，我々は様々な恩恵を得ることができるが，そのためにはデータの匿名加工とリスク評価が不可欠である．個人データの安全性を評価する指標として k-anonymity が広く知られているが，それを改善した指標である k-concealment が 2012 年に Tamir らによって提案されており，この指標を満たすようにデータを加工する（完全 k-concealment 化）ことによって，全ての個人が他の $k-1$ 人の個人と区別がつかないことが保証される．データを完全 k-concealment 化するためには，個人を点，個人間の類似度を辺とみなした二部グラフを加工の設計図として作成し，そのグラフ内で辺の重複しない k 種類の完全マッチングを探す必要がある．本稿では，それを解決するためにクラスタリングや巡回セールスマン問題の解法などを応用した 3 つの手法を提案し，位置情報データや購買履歴データを加工する実験を行うことによってそれらの性能を評価する．

キーワード：匿名加工, k-anonymity, k-concealment, 位置情報データ, 世帯収入データ

The evaluation of some accurate algorithms for complete k-concealment anonymization

Satoshi Ito[1,2,a)]　Hiroaki Kikuchi[3,b)]

Abstract: De-identification is a process to prevent individuals from being identified from original transaction data by processing personal identification information. Companies are required to assess the re-identification risks when employing big data extensively in their businesses. k-anonymity is a famous metrics to evaluate the privacy level of data and Tamir et al. proposed k-concealment method that is extension of k-anonymity in 2012. When we make data that satisfy k-concealment completely (complete k-concealment anonymization), it is ensured that all individuals in the data are not distinguished from other $k-1$ individuals. For the complete k-concealment anonymization, we must make a bipartite graph that has some nodes (individual) and some edges (similarity between individuals) as a blueprint of anonymization. When the bipartite graph has distinct k types perfect matchings, the processed data satisfy complete k-concealment. In this paper, we evaluate some methods to find low-cost distinct k types perfect matchings applying clustering and solutions for the traveling salesman problem.

Keywords: De-identification, k-anonymity, k-concealment, People Flow Data, Census Income Dataset

[1] 明治大学大学院先端数理科学研究科
Nakano, Nakano-ku, Tokyo 164–8525, Japan
[2] 日本学術振興会
Kojimachi, Chiyoda-ku, Tokyo 102–0083, Japan
[3] 明治大学総合数理学部
Nakano, Nakano-ku, Tokyo 164–8525, Japan
[a)] mmhm@meiji.ac.jp
[b)] kikn@meiji.ac.jp

1. はじめに

匿名加工は個人が識別されることを防ぐために個人情報を加工する技術である．購買履歴や位置情報などのビッグデータを匿名加工することによって，そのデータを安全に利活用することが可能になる．匿名加工されたデータの安

全性として，Sweeny によって提案された k-anonymity(k-匿名性)[1] が広く知られている．この指標はデータ中の少なくとも k 人の個人の区別がつかない（データが等しい）ことを保証するものであり，k-匿名性を満たすようにデータを加工することを k-匿名化と呼ぶ．

しかし，多くの研究者によって k-匿名性の弱点が指摘されている．Tamir らは，k-匿名化によって生じる $k+1$ 人以上のグループには加工の無駄があることを指摘し，それを解決するために新たな指標 k-concealment[2] を提案した．k-concealment は k-匿名性を拡張したもので，元データの個人を点とした二部グラフにおいて，全点（データ中の全個人）が**少なくとも** k 種類の辺の重複しない完全マッチングの辺を持つことを保証するものである．k-concealment を満たすように加工されたデータは，k-匿名化されたデータと同等の安全性を持ちながら，さらに有用性を向上することができる．

我々は，データ中の全個人が**等しく** k 種類の辺の重複しない完全マッチングの辺を持つように加工する処理（完全 k-concealment 化）に注目した．完全 k-concealment 化されたデータは，k-concealment を満たすように加工されたデータよりも加工の無駄が無くなり，さらに有用性が高まることが予想される．データを完全 k-concealment 化するためには，設計図の二部グラフ内に k 種類の辺の重複しない完全マッチングを作成すればよいが，そのためには個人数 n の階乗通り（全 $n!$ 通り）の完全マッチングの中からコストの低いものを探索する必要がある．

本稿では，データを完全 k-concealment 化するための 3 つのアルゴリズム（貪欲法，くじ引き法，TSP 解法手法）を提案する．貪欲法は距離の短い個人間に辺を張って完全マッチングを作成する手法であり，くじ引き法はランダムに複数回生成した完全マッチングのうち低コストのものを採用する手法であり，TSP（Traveling Salesman Problem）解法手法は巡回セールスマン問題の近似アルゴリズムを応用した手法である．我々は，これら 3 手法にクラスタリングを足した計 5 手法を用いて，100 人分の人流データと 1,000 人分の世帯収入データを完全 k-concealment 化する実験を行うことにより，どのような性質を持つデータに対してどの手法が有効であるかを評価する．

完全マッチングの探索は計算量的に困難であり，これを解決する効率的なアルゴリズムは知られておらず，そのため，近似アルゴリズムの提案や複数のデータを用いた性能評価には価値がある．また，完全 k-concealment 化のアルゴリズムは Tamir らによっても提案されていないため，加工手法の提案には新規性がある．完全 k-concealment 化は個人数 n に関わらず任意の k でデータを加工をすることができるため，データの一部にのみ適用することや，他の匿名加工手法と組み合わせる応用が可能であり，本稿で提案する加工手法は，いずれも個人数とレコード数が等しいデータにならば適用可能である．

表 1: 個人データの例 T_{ex}

ID	age	sex
Alice	10	F
Bob	20	M
Carol	40	M
David	50	F

本稿では，2 章で記号等の定義や既存研究の説明を行い，3 章で完全マッチングを作成するアルゴリズムを提案し，4 章で提案手法の性能評価のための実験を行う．

2. 基礎定義

2.1 データセット

本研究では，レコード（行）と属性（列）によって構成される個人データを考える．データ中の個人はレコードを一つのみ持ち，個人数とレコード数は常に等しくなる．記号等を以下のように定義する．

定義 2.1 （個人データ） 個人データを T とする．T のレコード数と個人数は n であり，ID 列を除く属性数は m である．T の個人集合を $U=\{u_1,\ldots,u_n\}$ とし，各個人は m 種類の属性にそれぞれ連続値または離散値の欠損値でない値を持つ．T の属性集合を $A=\{a_1,\ldots,a_m\}$ とし，個人 u_i が属性 a_x に持つ値を $v_{i,x}$ とする．

例 2.1 T の例として，表 1 に個人データ T_{ex} を示す．T_{ex} は 4 人の個人 $U=\{u_1=\mathrm{Alice}, u_2=\mathrm{Bob}, u_3=\mathrm{Carol}, u_4=\mathrm{David}\}$ と 2 つの属性 $A=\{a_1=\mathrm{age}, a_2=\mathrm{sex}\}$ を持つため，$n=4, m=2$ である．4 人の個人はそれぞれ age 属性の連続値と sex 属性の離散値を持ち，例えば $v_{1,1}$ は Alice が age 属性に持つ 10 を意味する．

定義 2.2 （個人間の距離） T の距離行列を

$$\mathrm{dist}(T)=\begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nn} \end{pmatrix}$$

とする，ここで，d_{ij} は個人 u_i と個人 u_j の間の距離であり，$d_{ij}=\sum_{x=1}^{m} d_{ij}^{a_x}$ である．ここで，$d_{ij}^{a_x}$ は属性 a_x についての u_i と u_j の間の距離であり，a_x が連続値属性の場合，

$$d_{ij}^{a_x}=|v_{i,x}-v_{j,x}|/\max_{1\le i,j\le n}(|v_{i,x}-v_{j,x}|)$$

であり，a_x が離散値属性の場合，

$$d_{ij}^{a_x}=\begin{cases} 0 & (v_{i,x}=v_{j,x}), \\ 1 & (\mathrm{otherwise}), \end{cases}$$

である．

例 2.2 表 1 の T_{ex} について，個人間の距離行列 $\mathrm{dist}(T_{\mathrm{ex}})$ を表 2 に示す．$\mathrm{dist}(T_{\mathrm{ex}})$ は，各属性についての距離行列の和で求められる．

表 2: T_{ex} の個人間の距離行列 $\mathrm{dist}(T_{\mathrm{ex}})$

	Alice	Bob	Carol	David
Alice	0	1.25	1.75	1.00
Bob	1.25	0	0.50	1.75
Carol	1.75	0.50	0	1.25
David	1.00	1.75	1.25	0

表 3: 加工データ例 $T'_{\mathrm{ex}\ 1}$

仮名	age	sex
1	[10, 20]	{F, M}
2	[10, 20]	{F, M}
3	[40, 50]	{F, M}
4	[40, 50]	{F, M}

表 4: 加工データ例 $T'_{\mathrm{ex}\ 2}$

仮名	age	sex
1	[10, 50]	{F, M}
2	[10, 50]	{F, M}
3	[10, 50]	{F, M}
4	[10, 50]	{F, M}

2.2 *k*-anonymity

一般的に，k-匿名化を行う際は，少なくとも k 個のレコードの値を同じに加工することによって，それらの個人の区別がつかないようにしている．複数個人のデータを等しく加工する手法として一般化やミクロアグリゲーションなどが挙げられるが，本稿では一般化のみに注目し，以下のように加工を行う．また，k-匿名化の手法として，特定の属性の値の頻度が k 未満である特異な個人を削除する（行削除）ものがあるが，本稿ではデータ中の個人は削除しないものとする．

定義 2.3　（一般化）（ℓ 人の個人 $u_1, \ldots, u_\ell$ の見分けがつかないようにするとき，各個人が持つ）属性 $a_1, \ldots, a_m$ の値を加工する．a_x が連続値属性の場合，$v_{1,x}, \ldots, v_{\ell,x}$ の一般化を閉区間 $[\min(v_{1,x}, \ldots, v_{\ell,x}), \max(v_{1,x}, \ldots, v_{\ell,x})]$ とする．a_x が離散値属性の場合，$v_{1,x}, \ldots, v_{\ell,x}$ の一般化をそれらの値の和集合 $\{v_{1,x}, \ldots, v_{\ell,x}\}$ とする．

例 2.3　T_{ex} を一般化で加工した例として，表 3, 4 に $T'_{\mathrm{ex}\ 1}$ と $T'_{\mathrm{ex}\ 2}$ を示す．$T'_{\mathrm{ex}\ 1}$ は Alice と Bob，Carol と David のレコードが一般化によって等しくなり，区別がつかなくなっているため 2-匿名性を満たしている．また，$T'_{\mathrm{ex}\ 2}$ は全てのレコードが一般化によって等しくなり，区別がつかなくなっているため 4-匿名性を満たしている．

定義 2.4　（データの加工コスト）個人 u_i を加工したときのコスト $\mathrm{cost}(u_i)$ は，一般化によって u_i と区別がつかなくなった全個人と u_i の距離の和であるとする．また，データ T の加工コスト $\mathrm{cost}(T)$ は，全ての個人の加工コストの和であるとする．つまり，$\mathrm{cost}(T) = \sum_{i=1}^{n} \mathrm{cost}(u_i)$ である．

例 2.4　$T'_{\mathrm{ex}\ 1}$ の加工コスト $\mathrm{cost}(T'_{\mathrm{ex}\ 1})$ について考える．Alice は一般化によって Alice と Bob の区別がつかない仮名 1 に加工されているため，Alice の加工コスト cost(Alice) は 0(Alice と Alice の距離)+1.25(Alice と Bob の距離)= 1.25 である．全ての個人の加工コストの和がデータの加工コストであるため，$\mathrm{cost}(T'_{\mathrm{ex}\ 1})$ は $1.25 + 1.25 + 1.25 + 1.25 = 5$ となる．また，$T'_{\mathrm{ex}\ 2}$ の加工コスト $\mathrm{cost}(T'_{\mathrm{ex}\ 2})$ について考える．Alice は一般化によって全個人との区別がつかない仮名 1 に加工されているため，Alice の加工コスト cost(Alice) は $0 + 1.25 + 1.75 + 1 = 4$ であり，$\mathrm{cost}(T'_{\mathrm{ex}\ 2})$ は $4 + 3.5 + 3.5 + 4 = 15$ となる．

定義 2.5　（完全 *k*-匿名性）全ての個人が等しく $k-1$ 人の他の個人と区別がつかないことを保証する指標を，完全 k-匿名性とする．完全 k-匿名性を満たすようにデータを加工することを，完全 k-匿名化と呼ぶ．

例 2.5　$T'_{\mathrm{ex}\ 1}$ は全ての個人が他の 1 人の個人と区別がつかないデータであるため，完全 2-匿名化されたものである．また，$T'_{\mathrm{ex}\ 2}$ は完全 4-匿名化されたものである．

2.3 *k*-concealment

k-anonymity を満たすデータと k-concealment を満たすデータは，どちらも「少なくとも k 人の区別がつかない」という条件を満たしているが，後者の方が加工コストが低くなる場合もある．本稿では k-concealment を満たすようにデータを加工することを k-concealment 化と呼び，また，完全 k-匿名化と後述する完全 k-concealment 化について研究する．データを k-concealment 化するためには，まず加工の設計図として点（個人）と辺（個人間の対応関係）を持つ二部グラフを作成し，それに従って個人の情報を一般化すればよい．

定義 2.6　（完全 *k*-concealment）全ての点が等しく k 種類の辺の重複しない完全マッチングの辺（match）を持つデータを，完全 k-concealment と呼ぶ．完全 k-concealment を満たすようにデータを加工することを，完全 k-concealment 化と呼ぶ．

定義 2.7　（正解完全マッチング）同じ点同士を結ぶ辺を正解の辺とし，正解の辺のみで構成された完全マッチングを正解完全マッチングとする．完全 k-concealment 化を行うときに k 種類の完全マッチングを選ぶが，必ず一つの正解完全マッチングを含む．

例 2.6　図 1, 2 に，T_{ex} の 4 人の個人についての二部グラフ例 B_1, B_2 を示す．B_1 には 2 種類の完全マッチング（赤実線，緑実線）があり，全ての個人が等しく 2 本の match を持っている．この場合，赤実線の完全マッチングは正解完全マッチングである．B_2 は B_1 に 1 つの完全マッチング（青点線）が加えられたものであるが，既存の完全マッチングと一部の辺が重複しているので，Bob や Carol は match を 3 本持っているが，Alice や David は match を 2 本しかもっていない．

また，B_1 を基に加工した $T'_{\mathrm{ex}B_1}$ と B_2 を基に加工した $T'_{\mathrm{ex}B_2}$ を，表 5, 6 に示す．例えば，B_1 の右側の Alice は左側の Alice と David との間に辺を持つので，Alice と David との区別がつかない仮名 1 に一般化されている．$T'_{\mathrm{ex}B_1}$ は T_{ex} を完全 2-concealment 化したものであり，$T'_{\mathrm{ex}B_2}$ は T_{ex}

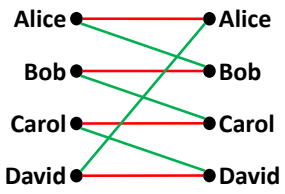

図 1: T_{ex} についての二部グラフ B_1

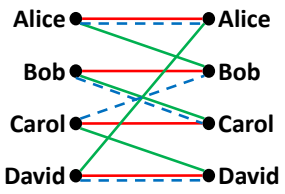

図 2: T_{ex} についての二部グラフ B_2

表 5: B_1 から加工した $T'_{\mathrm{ex}B_1}$

仮名	age	sex
1	[10, 50]	{F}
2	[10, 20]	{F, M}
3	[20, 40]	{M}
4	[40, 50]	{F, M}

表 6: B_2 から加工した $T'_{\mathrm{ex}B_2}$

仮名	age	sex
1	[10, 50]	{F}
2	[10, **40**]	{F, M}
3	[20, 40]	{M}
4	[40, 50]	{F, M}

を（不完全な）2-concealment 化したものである．二部グラフのみを見ると，B_2 は B_1 に辺を 1 本足したものなので，その分加工の度合いが大きくなっている（仮名 2 の age 属性）．

定義 2.8　（二部グラフの辺） k-concealment 化の加工コスト $\mathrm{cost}(T)$ は，基になった二部グラフの辺の距離の総和である．

例 2.7　$T'_{\mathrm{ex}B_1}$ の加工コスト $\mathrm{cost}(T'_{\mathrm{ex}B_1})$ は，基にした B_1 の 8 本の辺の距離の和であるため，$0+0+0+0+1.25+0.50+1.25+1=4$ である．また，B_2 は B_1 に左側の Carol から右側の Bob に向かう辺を 1 本足したものなので，$T'_{\mathrm{ex}B_2}$ の加工コスト $\mathrm{cost}(T'_{\mathrm{ex}B_2})$ は $4+0.5=4.5$ である．

定義 2.9　（左右対称完全マッチング）正解の辺を含まない左右対称の完全マッチングを，左右対称完全マッチングと呼ぶ．

左右対称完全マッチング 1 種と正解完全マッチングを組み合わせた二部グラフをもとにすれば，データを完全 2-匿名化することができる．n が奇数のとき，左右対称完全マッチングは存在しない．

命題 2.1　（完全マッチングの種類数について） n 人の個人についての完全マッチングは全 $n!$ 通りあり，そのうち左右対称完全マッチングは $\prod_{i=1}^{n/2}(n-2i+1)$ 種類ある．また，$u_1 \to u_2 \to \cdots \to u_n (\to u_1)$ のように辺が全体で循環する完全マッチングは $(n-1)!$ 種類ある．

例 2.8　$n=4$ の場合，完全マッチングは $4!=24$ 通りあり，そのうち左右対称完全マッチングは $\prod_{i=1}^{4/2}(4-2i+1)=(4-2+1)(4-4+1)=3$ 種類である．また，$n=10$ の場合，完全マッチングは $10!=3,628,880$ 通りあり，そのうち左右対称完全マッチングは $\prod_{i=1}^{10/2}(10-2i+1)=(10-2+1)(10-4+1)(10-6+1)(10-8+1)(10-10+1)=945$ 種類である．

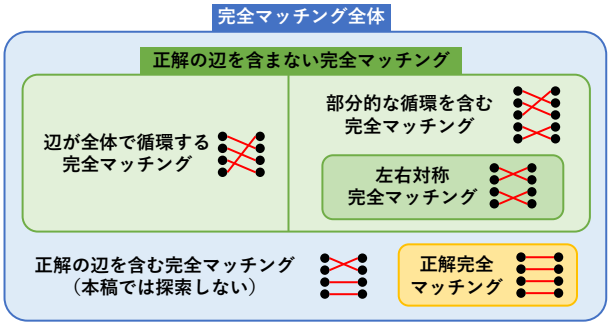

図 3: 完全マッチングのベン図

完全マッチングの分類（左右対称完全マッチングや，辺が全体で循環する完全マッチングなど）の関係を図 3 に示す．

定義 2.10　（完全マッチングの表記） n 人の個人についての完全マッチングを，置換 $P=(P[1],\ldots,P[n])$ と表記する．P には値 $\{1,\ldots,n\}$ が一度ずつ記録されており，$P[i]=j$ であるとき u_i から u_j に辺が張られている．

定義 2.11　2 種類の完全マッチング P_1, P_2 があり，$P_1[i]=P_2[i]$ であるとき，P_1 と P_2 は辺が重複している．P_1 と P_2 の辺が重複していることを $\mathrm{over}(P_1,P_2)$ と表記する．

例 2.9　T_{ex} についての二部グラフ B_1（図 1）に注目する．この二部グラフには 2 種類の完全マッチングが含まれており，$U=\{u_1=\mathrm{Alice}, u_2=\mathrm{Bob}, u_3=\mathrm{Carol}, u_4=\mathrm{David}\}$ とすると，赤実線のものは $P_1=(1,2,3,4)$，緑実線のものは $P_2=(2,3,4,1)$ と表記される．また，B_2 の青点線の完全マッチングは $P_3=(1,3,2,4)$ と表記できるが，$P_1[1]=P_3[1], P_1[4]=P_3[4], P_2[2]=P_3[2]$ であるため，この完全マッチングは前者 2 つと辺が重複している．

定理 2.1　（完全 k-concealment 化の加工コストについて） 完全 k-concealment 化の最小加工コストは完全 k-匿名化の最小加工コスト以下である．

(証明)　完全 k-匿名化されたデータは完全 k-concealment であることを示す．完全 k-匿名なデータは k 人の個人 $u_1,\ldots,u_k$ が同じデータを持っており，区別がつかなくなっているため，これは二部グラフ上で $u_1,\ldots,u_k$ それぞれから $u_1,\ldots,u_k$ 全てに辺が張られている状態である．ここで，$P_1=(1,2,\ldots,k), P_2=(2,\ldots,k,1),\ldots,P_k=(k,1,\ldots,k-1)$ という k 種類の完全マッチングを考える．これらの完全マッチングはいずれも辺が重複しておらず，全てを重ねると $u_1,\ldots,u_k$ それぞれから $u_1,\ldots,u_k$ 全てに辺が張られている二部グラフとなる．つまり，完全 k-匿名化されたデータの k 人の個人 $u_1,\ldots,u_k$ は，全て等しく k 本の match を持っているといえるため，完全 k-concealment も満たしている．よって，完全 k-匿名化されたデータの最小加工コストは完全 k-concealment 化されたデータの加工

Algorithm 1 提案手法 1：貪欲法

Input: $\mathrm{dist}(T), k$
  n：個人数
  d_{ij}：u_i と u_j の間の距離
  L：完全マッチングを記録するためのリスト
  $L[1] \leftarrow (1, 2, \ldots, n)$
  for i **in** $\{2, \ldots, k\}$ **do**
    P：n 次元の置換
    U_1, U_2：個人集合 $U_1 = U_2 = \{u_1, \ldots, u_n\}$
    while $|U_1| > 0$ **do**
      U_1 から，ランダムに u_x を選ぶ.
      $u_y = \arg\min_{u_y \in U_2} d_{xy}$
      $U_1 \leftarrow U_1 - \{u_x\}$, $U_2 \leftarrow U_2 - \{u_y\}$, $P[x] \leftarrow y$
    end while
    if $\forall_{j \in [1, i-1]} \overline{\mathrm{over}(P, L[j])}$ **then**
      $L[i] \leftarrow P$
    else
      i 番目のループをやり直す.
    end if
  end for
Output: L

Algorithm 2 提案手法 2：くじ引き法

Input: $\mathrm{dist}(T), k$
  n：個人数
  d_{ij}：u_i と u_j の間の距離
  L：完全マッチングを記録するためのリスト
  t：試行回数
  count：成功回数
  $L[1] \leftarrow (1, 2, \ldots, n)$
  for i **in** $\{2, \ldots, k\}$ **do**
    for count **in** $(1, \ldots, t)$ **do**
      ランダムに完全マッチング P を生成する.
      if $\forall_{j \in [1, i-1]} \overline{\mathrm{over}(P, L[j])}$ **then**
        $P_{\mathrm{count}} \leftarrow P$
      end if
    end for
    $P_{\min}$：$P_1, \ldots, P_t$ のうち，コストが最小であるもの
    $L[i] \leftarrow P_{\min}$
  end for
Output: L

コストでもあるため，これが完全 k-concealment 化の最小加工コストを下回ることはない. (Q.E.D)

3. 提案手法

加工コストが低い完全 k-concealment 化をするためには，辺が重複しないようにコストが低い完全マッチングを k 種類選んで二部グラフを作成し，それをもとにデータを加工すればよい．しかしながら，前述したように n 人の個人についての完全マッチングは全 $n!$ 通りあり，それらの加工コストを全て求めて最適解を見つけるのは計算量的に困難であるため，近似解を検討する．本章では，精度が高く加工コストが低い完全 k-concealment 化をするための近似アルゴリズムを提案する.

3.1 提案手法 1：貪欲法

貪欲法による提案手法のアルゴリズム 1 に示す．貪欲法はランダムな個人から最も近い個人へ辺を張って完全マッチング P を k 種類作成し，それらを記録したリスト L を出力するシンプルな手法である．定義 2.7 より，正解完全マッチングが必ず最初に選ばれていることを定義しているため，$k-1$ 回しか完全マッチングを作成していない点に注意せよ．また，この手法では張れる辺が無くなってしまう「詰み」が発生してしまう場合があることに注意せよ.

3.2 提案手法 2：くじ引き法

提案手法の一つであるくじ引き法をアルゴリズム 2 に示す．くじ引き法は完全マッチング P をランダムに t 回生成し，そのうちコストが最小のものを二部グラフに加えていく手法である．P が生成される際に，既存の完全マッチングと辺の重複が無いかを確認し，重複がある場合は再生成する．ループが進んでリスト L の要素が増えていくたびに，生成した P がチェックに通る確率が下がるため，k や t の値によって計算時間が増加していくことに注意せよ．また，完全マッチングの種類数は $n!$ であるため，個人数 n が増えるにつれてランダム生成によって低コストのマッチングをひける確率が下がり，加工コストも大きくなる.

3.3 提案手法 3：TSP 解法手法

巡回セールスマン問題の解を求める近似アルゴリズムを応用したものをアルゴリズム 3 に示す．TSP は，セールスマンが全ての都市を 1 回ずつ巡回する場合の最短経路を求める問題であり，NP 困難であることが知られている．我々は，巡回経路を辺が循環する完全マッチングに置き換えられることに注目し，既存の TSP 解法を低コストの完全マッチングの探索に応用した.

例えば，T_{ex} についての二部グラフ B_1（図 1）に注目する．B_1 の緑実線の完全マッチングでは，辺が Alice → Bob → Carol → David → Alice と循環している．このような辺が循環する完全マッチングを都市の巡回経路に，個人間の距離を都市間の距離に置き換えることによって，TSP 解法を用いて低コストの完全マッチングを高速に探索することができる．しかし，この手法によって検索できるのは，完全マッチング全体（$n!$ 種類）のうち，辺が循環する完全マッチング $(n-1)!$ 種類のみであることに注意せよ.

Algorithm 3 提案手法 3：TSP 解法手法

Input: $\mathrm{dist}(T), k$
  n：個人数
  L：完全マッチングを記録するためのリスト
  $L[1] \leftarrow (1, 2, \ldots, n)$
  for i **in** $\{2, \ldots, k\}$ **do**
    if i が偶数 **then**
      L の各要素の置換と辺の重複がないように，9 種類の TSP 解法で完全マッチング P を作成し，コストの最も低いものを P_i とする．
    else
      ひとつ前のループで作成された P_{i-1} の逆回りの完全マッチングを作成し，P_i とする．
    end if
    $L[i] \leftarrow P_i$
  end for
Output: L

表 7: TSP 近似アルゴリズムの概要

ID	手法名	概要
1	identity	巡回経路を ID 順に出力する
2	random	巡回経路をランダムに出力する
3	nearest insertion	ある都市から始まる巡回経路に，まだ含まれていない都市を挿入して巡回経路を作っていく Nearest：経路上のある都市に最も近い都市を挿入する．
4	cheapest insertion	Cheapest：経路の増加具合が最も少ない都市を挿入する．
5	farthest insertion	Farthest：経路上のある都市に最も遠い都市を挿入する．
6	arbitrary insertion	Arbitrary：経路に含まれていない都市をランダムに挿入する．
7	nn	ランダムな都市から経路を始め，経路の終端の都市に最も近い都市を後ろに追加していく
8	repetitive nn	Repetitive：全ての都市をスタート地点として計算を行い，最も距離が短い経路を返す．
9	two opt	経路の一部の辺を交換 (順番を逆にする) して距離を短くしていく手法

本稿では，R 言語 [4] の TSP パッケージ [5] を用いて TSP 解法手法を実装した．TSP の近似アルゴリズムには，TSP パッケージに含まれている表 7 の 9 種類の手法を用いる．k 回の探索 1 回ごとに，全ての手法を用いて完全マッチングを作成し，最もコストの低いものを採用している．また，コストの低い巡回経路を見つけたとき，その逆回りの経路も同様にコストの低い経路であることを利用し，計算時間短縮のために検索を一部で省いている．

3.4 提案手法 2,3+クラスタリング

提案手法 2 のくじ引き法には，n が増えるにしたがって加工コストが大きく増加してしまう問題点があり，提案手法 3 の TSP 解法手法には，完全マッチング全 $n!$ 種類のうち，辺が循環する完全マッチング $(n-1)!$ 種類だけしか探索できないという問題点があった．そこで我々は，問題を小規模の部分問題に分解して効率よく解き，部分解を統合して大きな解を得るという分割統治法の考えに基づき，部分問題への分解にクラスタリングを用いてこれらの問題点の解決を試みる．

Algorithm 4 提案手法 + クラスタリング

Input: $\mathrm{dist}(T), k$
  n：個人数
  L：完全マッチングを記録するためのリスト
  各クラスタの要素数が k を下回らないようにクラスタリングを行い，n 人の個人を c 個のデータ $T_1, \ldots, T_c$ に分ける．
  for i **in** $\{1, \ldots, c\}$ **do**
    $k, \mathrm{dist}(T_i)$ をアルゴリズム 2, または 3 に入力し，出力としてリスト L_i を得る．
  end for
  for j **in** $\{1, \ldots, k\}$ **do**
    $L_1[j], \ldots, L_c[j]$ を結合し，$L[j]$ に記録する．
  end for
Output: L

アルゴリズム 4 にクラスタリングを組み合わせる方法を示す．n 人の個人をクラスタリングによって c 個のデータに分割し，各データについて提案手法 2 や 3 で完全マッチングリスト $L_1, \ldots, L_c$ を出力し，最後にこれらを結合してリスト L を作成する．データを分割することによって，疑似的に完全マッチング種類数 $n!$ を減らすことができるので，提案手法 2 で低コストのマッチングをひける確率が高くなり，加工コストを下げることができる．また，クラスタリングと提案手法 3 を組み合わせることによって，TSP 解法単体では探索できない完全マッチングも作成できるようになる．本稿では，クラスタリング手法にウォード法 [6] を用いている．

4. 評価実験

4.1 データセット

本章では，2 つのデータセット（疑似人流データ，世帯収入データ）を用いて提案手法 1,2,3 や提案手法 + クラスタリングの評価を行う．

疑似人流データ [7] はナイトレイ社より公開されている位置情報データである．このデータは都市圏周辺の数千人分の人流を，SNS の地域解析に基づいて疑似的にデータ化したものであり，緯度や経度などの 9 つの属性が含まれている．疑似人流データには各個人の 5 分刻みの位置情報が記録されているため，レコード数 $\neq$ 個人数のデータであるが，我々は 0 時 0 分のレコードを用いることにより，定義 2.1 を満たすように処理している．また，本稿ではランダムに抽出した個人 100 人と，緯度と経度の連続値属性 2 つのみのデータ $T_{\text{人流}}$ を用いる（$n = 100, m = 2$）．

世帯収入データには，UCI より公開されている Adult Data Set[8] を用いる．このデータは国勢調査によって作成された 32,561 レコードのデータであり，職種などの離散値属性 9 つと，年齢などの連続値属性 6 つを含んでいる．Adult Data Set はレコード数 = 個人数のデータであ

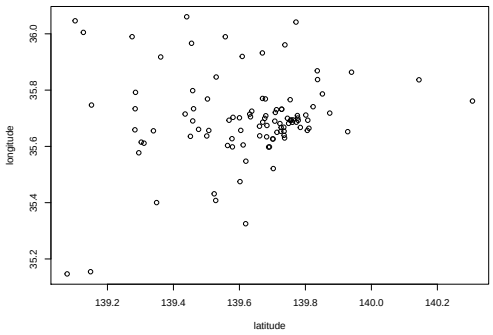

図 4: $T_{\text{人流}}$ の位置情報

り，我々はランダムに抽出した個人 1,000 人分の全ての属性を用い，これを $T_{\text{世帯}}$ とする（$n = 1,000, m = 15$）．

4.2 実験方法

4.2.1 実験 1：$T_{\text{人流}}$ を用いた手法比較

実験 1 では，提案手法 1,2,3 に加え，提案手法 2,3 にクラスタリングを加えた計 5 手法の完全 2-concealment 化手法と，貪欲法を用いた完全 2-匿名化手法の比較を行う．我々はこれら 6 種類の手法で加工された $T_{\text{人流}}$ の加工コストを求め，性能の評価を行う．

また，我々は提案手法の加工度合いの可視化も試みる．$T_{\text{人流}}$ の 100 人の個人の位置情報（緯度と経度）を図 4 に示す．$T_{\text{人流}}$ のような位置情報を一般化で加工すると，後述する図 5 のように個人の位置情報が点から四角形の範囲になる．例えば，図 4 中の左下に位置する 2 人の個人の区別をつかないようにすると，この 2 点を対角の頂点とする四角形に一般化してやればよく，この四角形の面積が小さいほど有用性が高いといえる．

4.2.2 実験 2：$T_{\text{世帯}}$ を用いた手法比較

実験 2 では，提案手法 2,3 とクラスタリングを組み合わせた計 4 手法で $T_{\text{世帯}}$ を完全 k-concealment 化し，それらの性能の比較を行う．この実験では，$T_{\text{世帯}}$ を $k = 2, \ldots, 7$ で完全 k-concealment 化したときの加工コストを評価する．

4.3 実験結果

4.3.1 実験 1 の結果

6 つの加工データのコストを表 8 に示す．$T_{\text{人流}}$ を $k = 2$ で加工する場合，提案手法 2+クラスタリングが最もコストが低くなった．また，提案手法 1,2,3 単体はいずれも完全 2-匿名化に加工コストで劣っているが，クラスタリングを組み合わせることによって提案手法 2,3 は完全 2-匿名化より有用性の高いデータを生成した．

まず，貪欲法での完全 2-匿名化と完全 2-concealment 化の比較を行う．図 5, 6 に，$T_{\text{人流}}$ を貪欲法にて完全 2-匿名化した結果と完全 2-concealment 化した結果を示す．どちらも個人 2 人の区別がつかないデータであるが，前者の加工コストは 0.2145，後者の加工コストは 0.2328 であり，貪欲法では完全 2-匿名化の方が有用性が高くなった．

表 8: $T_{\text{人流}}$ の加工コスト（$k = 2$）

加工手法	加工コスト
完全 2-匿名化（貪欲法）	0.2145
提案手法 1（貪欲法）	0.2328
提案手法 2（くじ引き法）	3.1490
提案手法 2+クラスタリング	**0.1493**
提案手法 3（TSP 解法手法）	0.3186
提案手法 3+クラスタリング	0.1640

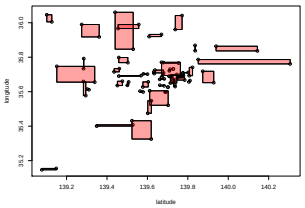

図 5: 完全 2-匿名化された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 0.2145$）

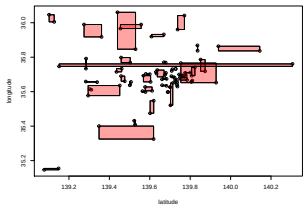

図 6: 提案手法 1 で加工された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 0.2328$）

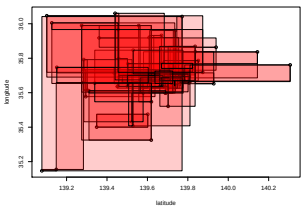

図 7: 提案手法 2 で加工された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 3.1490$）

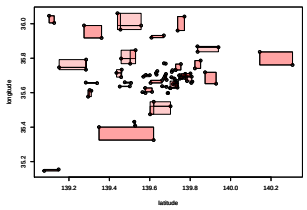

図 8: 提案手法 2+クラスタリングで加工された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 0.1493$）

次に，提案手法 2（くじ引き法，$t = 100,000$）のクラスタリングの有無で比較を行う．図 7, 8 に，$T_{\text{人流}}$ をクラスタリング無しで加工した結果とクラスタリング有りで加工した結果を示す．提案手法 2 のみでは一般化による四角形の面積が大きくなっている一方で，クラスタリングをした場合では距離が近い個人が一般化されていることがわかる．前者の加工コストは 3.149，後者の加工コストは 0.1493 であった．

最後に，提案手法 3（TSP 解法手法）のクラスタリングの有無で比較を行う．図 9, 10 に，$T_{\text{人流}}$ をクラスタリング無しで加工した結果とクラスタリング有りで加工した結果を示す．提案手法 3 のみでは，図 9 のようにデータ全体を循環する経路を探してしまうため，四角形の面積は大きくなってしまい，加工コストは 0.3186 である．一方，クラスタリングを組み合わせた場合は加工コストが 0.1640 まで下がっている．

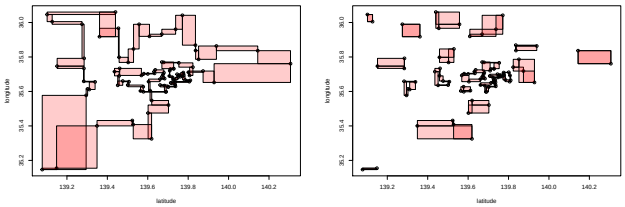

図 9: 提案手法 3 で加工された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 0.3186$）

図 10: 提案手法 3+クラスタリングで加工された $T_{\text{人流}}$（$\text{cost}(T_{\text{人流}}) = 0.1640$）

表 9: $T_{\text{世帯}}$ の加工コスト（$k = 2, \ldots, 7$）

k	提案手法 2	+クラスタリング	提案手法 3	+クラスタリング
2	5535.23	1819.31	1683.76	1718.13
3	11066.13	4083.07	3367.52	3443.10
4	16600.02	7413.35	5416.70	5656.75
5	22153.13	10041.37	7465.88	7858.74
6	27682.26	12741.39	9718.28	10364.58
7	33205.50	15973.04	11970.68	12826.53

4.3.2 実験 2 の結果

表 9 に，4 つの手法の加工コストを示す．まず，提案手法 2（$t = 10,000$）の性能評価を行う．例えば，提案手法 2 のみで完全 2-concealment 化を行うと加工コストは 5535.23 であるが，クラスタリングを組み合わせると加工コストが 1819.31 まで減少する．

次に，提案手法 3 の性能評価を行う．$T_{\text{世帯}}$ の場合，提案手法 3 は提案手法 2 よりも全ての k で加工コストが優れている．また，$T_{\text{人流}}$ の場合とは異なり，クラスタリングをしない方が有用性の高い加工ができており，例えば $k = 7$ のときはクラスタリングをすることによって加工コストが 1.07 倍ほどに増加している．

4.4 考察

$T_{\text{人流}}$ を用いた実験 1 の結果では，クラスタリングを組み合わせたほうが提案手法 2,3 ともに加工コストが低くなっていたが，$T_{\text{世帯}}$ を用いた実験 2 の結果では，クラスタリングをしない方が提案手法 3 の加工コストが低くなった．この原因として，データによる個人間の距離の分布の違いが考えられる．図 11,12 に，各データの個人間距離の分布を示す．図からわかるように，$T_{\text{人流}}$ には近い距離に個人が多く分布しており，クラスタリングをすることによって局所的な最適解を求めやすくなると考える．一方，$T_{\text{世帯}}$ は属性数が多いためか，個人間の距離が大きく，このようなデータに対してはクラスタリングは効果が薄いと考えられる．

5. おわりに

本稿では，レコード数と個人数が等しいデータを完全 k-concealment 化する研究を行った．完全 k-concealment 化を行うためには，データ中の個人を点とした二部グラフ

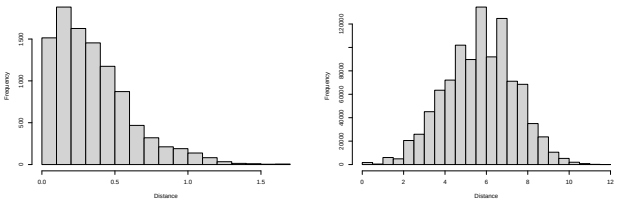

図 11: $T_{\text{人流}}$ における個人間距離の分布

図 12: $T_{\text{世帯}}$ における個人間距離の分布

を加工の設計図に見立て，その中に辺の重複しない k 本の完全マッチングを作成する必要がある．そこで，我々は辺の重複しない k 本の完全マッチングを作成する 3 つのアルゴリズム（1:貪欲法，2:くじ引き法，3:TSP 解法手法）を提案した．

また，100 人分の人流データと 1,000 人分の世帯収入データを用いて提案手法の性能評価を行ったところ，以下の結果が得られた．(1) 人流データを $k = 2$ で加工するとき，提案手法 2,3 とクラスタリングを組み合わせると，完全 2-匿名化されたデータよりも有用性の高い加工をすることができる．(2) 世帯収入データを完全 k-concealment 化するときは，提案手法 3 のみの加工が最もコストが低く，クラスタリングを組み合わせるとコストが 1.07 倍まで増加する．

本稿で提案した完全 k-concealment 化手法は，レコード数と個人数が等しいならばどのようなデータに対しても適応可能であるが，なかでも近い距離に多くの個人が分布しているデータに対しては，クラスタリングを組み合わせると加工コストの向上が期待できる．

謝辞 本研究は JSPS 特別研究員奨励費 21J13050 の助成を受けたものです．

参考文献

[1] L. Sweeny, “k-anonymity: a model for protecting privacy”, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5), pp.557–570, (2006).

[2] Tamir Tassa, Arnon Mazza, Aristides Gionis, “k-Concealment: An Alternative Model of k-Type Anonymity”, TRANSACTIONS ON DATA PRIVACY 5, pp. 189–222, (2012).

[3] 高橋磐郎，藤重悟，「離散数学」，岩波講座情報科学 17, pp.147–162, (1981).

[4] The R Project for Statistical Computing, `https://www.r-project.org/`, 2021 年 8 月 6 日参照.

[5] Michael Hahsler, Kurt Hornik, “Package TSP”, `https://cran.r-project.org/web/packages/TSP/TSP.pdf`, 2021 年 8 月 6 日参照.

[6] Ward, J. H., Jr., “Hierarchical Grouping to Optimize an Objective Function”, Journal of the American Statistical Association, 58, pp. 236–244, (1963).

[7] 疑似人流データ，ナイトレイ社，`https://nightley.jp/archives/1954/`，2021 年 8 月 6 日参照.

[8] Adult Data Set, UCI Machine Learning Repository, `https://archive.ics.uci.edu/ml/datasets/adult`, 2021 年 8 月 6 日参照.