サービス事業者間データ連携における
分散匿名化手法の提案

竹之内 隆夫

電気通信大学大学院情報システム学研究科

博士（工学）の学位申請論文

2013年3月

サービス事業者間データ連携における
分散匿名化手法の提案

博士論文審査委員会

主査　　大須賀 昭彦　教授

委員　　田中 健次　教授

委員　　小池 英樹　教授

委員　　大森 匡　教授

委員　　川村 隆浩　客員准教授

Proposal of Distributed Anonymization Method for Data Federation between Service Providers

Takao Takenouchi

Abstract

Recently, it is expected that personal information stored by different service providers are combined securely and it will create a new service. However, there is a risk that a specific user record can be identified by the combined personal information, and the user's sensitive information is revealed. Also, the personal information collected by the service provider must not be disclosed to other service providers because of security and privacy issues. Thus, related researches have been conducted on distributed anonymization methods, which combine the personal information stored by the providers and sanitize it to ensure a policy of anonymity with the minimum disclosure.

However, in those researches, if sets of the users among the providers are different, a problem occurs that the users' presence in either provider may be revealed. Therefore, this paper proposes a new indicator, named *δ-site-presence*, which represents the probability of the users' presence being revealed. Also, this paper proposes an improved distributed anonymization protocol which satisfies the proposed indicator. This protocol uses dummy users who do not exist in the provider. The providers treat the dummy users as if they actually exist. By using the dummy users, it can anonymize the personal information without disclosing the users' presence.

We evaluate the security of the proposed protocol and proof that the protocol does not disclose any sensitive information. In addition, we evaluate the processing and communication cost of the protocol. The evaluation results show that the cost of the proposed protocol is not much higher than that of the existing protocols.

Moreover, we evaluate the utility of the proposed protocol with U.S. Census data and health data. Our evaluation results show that the proposed protocol can anonymize them

with lower information loss than the existing distributed anonymization method.

It is expected that our method combine not only census data and health data but also several types of the personal information and there is a possibility that a new service will be created.

サービス事業者間データ連携における分散匿名化手法の提案

竹之内 隆夫

概要

近年，複数のサービス事業者が保持するユーザのパーソナル情報を連携し，新たな知見を得ることによって，より良いサービスを創出することが期待されている．パーソナル情報にはユーザのプライバシに係る情報が含まれているため，パーソナル情報を必要最小限の開示に留めながら結合し，個人が特定されない形に加工する手法が求められている．そして，その手法として分散匿名化手法が注目されている．しかし，既存の分散匿名化手法では，双方のサービス事業者のユーザ集合が一致しない場合に，ユーザのパーソナル情報がそのサービス事業者に保持されているか否かというユーザ存在情報が，他方のサービス事業者に漏洩する問題があった．

そこで本論文では，このようなユーザ存在情報が漏洩する問題を軽減するために，新たに δ-*site-presence* というプライバシ指標を提案する．この指標によって，ユーザ存在情報が漏洩する可能性の許容範囲を示すことができる．そして，提案した指標を満たしつつ，データマイニング等での有用性を保った結合匿名テーブルを生成するための新たな分散匿名化のプロトコルを提案する．このプロトコルでは，存在するユーザと存在しないユーザの区別を困難にさせるダミーユーザを導入し，ユーザ存在情報の漏洩を軽減している．

そして，提案手法のプロトコルの安全性を暗号理論で用いられるシミュレータを用いた評価手法によって証明し，プライバシ性の高いパーソナル情報やユーザ存在情報が漏洩しないことを確認した．また，提案手法の計算量・通信量の評価を行い，双方の事業者が持つ情報を開示せずに単純な関数計算を行う既存のセキュア計算の計算量・通信量と比較した．その結果，提案手法の計算量・通信量は既存のセキュア計算の計算量・通信量と比較して，大幅な増加がないことを確認した．

さらに，提案手法を米国の国勢調査をもとに作成された評価データと実際のレセプトデータ（診療報酬明細情報）を用いて評価した．提案手法と既存の分散匿名化手法との実行結果を比較した結果，一定の条件下において提案手法は既存手法よりも大幅にデータの有用

性を保った匿名化が行えることを確認した．また，提案手法を既存の集中型のユーザ存在隠蔽の匿名化手法と比較し，提案手法は既存手法とほぼ同等に有用な匿名化が行えることを確認した．さらに，複数の医療機関が保持する医療データを結合・分析する利用場面を想定し，データ分析を行った際の集計誤差を計測した．結果，提案手法はユーザ存在情報の漏えいを軽減しながらも相対誤差 15%以下でデータ分析が可能であることがわかった．これは，近年言われている医療の効率化や医療サービスの質向上のための医学研究に適用できると考えられる．

提案手法を用いることによって，国勢調査データや医療データにとどまらず，様々な種類のパーソナル情報をサービス事業者間で安全にデータ連携することができ，新たなサービスが創出されることが期待できる．

目 次

図 目 次

表 目 次

第1章　序論

本章では，本研究の背景を述べた後，本論文の目的と貢献を説明する．その後，本論文の構成について述べる．

1.1　本研究の背景

近年，いくつかのサービス事業者は，ユーザのパーソナル情報を収集し，ユーザの好みに合わせたサービスを提供する等，収集したパーソナル情報を自事業者のサービスに利用している．今後これらのパーソナル情報は単一の事業者内で利用されるだけでなく，様々な事業者のパーソナル情報と組み合わせて利用されると考えられる．そして，組み合わせられたパーソナル情報を分析することで，新たな知見を得ることができ，より良いサービスが創出されることが期待されている [66, 68, 53, 56].

このような複数の事業者のパーソナル情報を連携 (データ連携) する利用場面として，例えば「(a) 医療機関のデータ連携」と「(b) 異業種のデータ連携」の 2 つが考えられる．以下にこれらの利用場面において，どのようなデータを連携し，どのような新たな知見を得ることが期待できるかについて説明する．

- (a) 医療機関のデータ連携

  医療機関が保持する患者の医療情報をデータ連携することにより，医学研究に有用なデータの分析が期待されている．例えば，日本のセンチネル・プロジェクトに関する提言 [53] では，複数の医療機関が保持するレセプトデータ (診療報酬明細書[1]) 等の医療情報を結合・分析することで,「ある医薬品の使用者における特定の副作用 (有害事

[1] レセプトデータ (診療報酬明細書) とは，患者が受診した医療費について医療機関が健康保険組合などの保険者に請求する際の明細書のことである．診療報酬明細書は，以前は紙であったが，現在は電子化が進んでいる [65].

象) の発生頻度を，当該医薬品を使用していない場合の有害事象の発生頻度と比較することが可能」になると言われている．

例えば，機関 A と機関 B は病院であり，診療した患者の診療情報として「被保険者番号」[2],「診療日」,「疾病情報」,「医薬品情報」を保持しているとする．そして，医学研究のために双方の機関の診療情報をデータ連携し，機関 A と機関 B が保持する診療情報を結合して公開することを想定する (図 1.1)．この場合，双方の機関が持つ「診療日」と「疾病情報」と「医薬品情報」を，共通の「被保険者番号」を用いて紐付けて結合したデータを生成することになる．これにより，ある患者について，機関 B で処方した「医薬品情報」と機関 A で受診した「疾病情報」が紐付くことになる．そして，この結合されたデータが開示されることにより，そのデータを受け取った研究機関 C は，機関 B で新しい薬品を注射した患者の集合のうち機関 A で副作用となる疾病を発症した患者の割合を計算できる．また，従来の薬品における同様の割合も計算することができる．これにより，新しい薬品と従来の薬品の使用に対する副作用の発生頻度を比較した副作用分析が可能になると考えられる．現状では，このような医療情報のデータ連携はプライバシ保護の観点で限定的となっているが，今後はプライバシを適切に保護した上で医療情報を副作用分析等の医学研究に利活用することが期待されている [53, 55] [3].

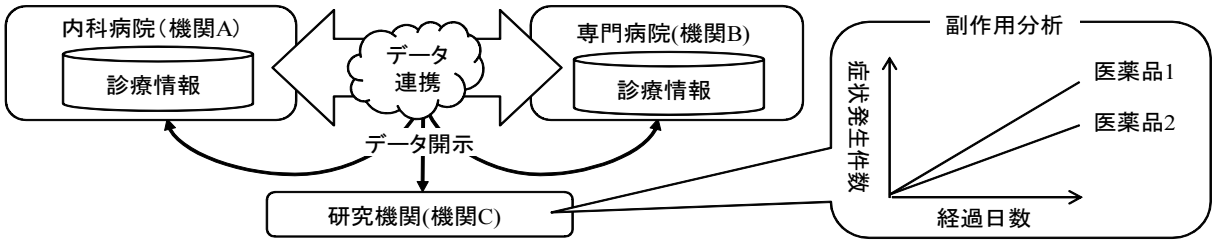


図 1.1: 「(a) 医療機関のデータ連携」の例

[2]被保険者番号とは，国民健康保険などの医療保険においてある保険者において被保険者を識別するための番号である．正確には扶養者がいる場合等は個人を一意に識別出来ないが，氏名などの他の情報との組み合わせることで一意に個人を識別できるとされているため，本論文では被保険者番号を個人の識別するための番号として用いる．

[3]2012 年度末において検討中となっている「医療個別法」[59] によって，公益目的での医療情報の利用規定が明確化され，匿名性や安全性が担保できる場合の利活用が促進される見通しとなっている

- (b) 異業種のデータ連携

  異なる業種が保持するユーザのパーソナル情報をデータ連携することで，新たなサービスが創出されることが期待されている [56]．例えば，オンデマンドビデオ配信サイト (機関 A) とローン会社 (機関 B) が連携し，機関 A が持つユーザの「視聴番組」及び「視聴時間帯」の情報と，機関 B が持つ「年収情報」を結合し，広告代理店 (機関 C) が番組視聴者の傾向分析を行う場合を考える (図 1.2)．この例では機関 A と機関 B は OpenID のような共通の認証サーバを利用しており，共通の認証 ID によって双方のパーソナル情報を結合する．このようにデータ連携することにより「昼間に視聴するユーザ群」,「夜間に視聴する比較的高収入のユーザ群」及び「夜間に視聴する比較的低収入のユーザ群」を見つけられるかもしれない．しかし，もしデータ連携を行わず「視聴情報」と「年収情報」が結合されなかったとしたら，単に「視聴時間帯」における「視聴番組」の分析程度しか行えず,「昼間に視聴するユーザ群」及び「夜間に視聴するユーザ群」しか見つけられないだろう．このように，機関 A と機関 B においてデータ連携することで，機関 C はより詳細な分析を行えることが期待される [56].

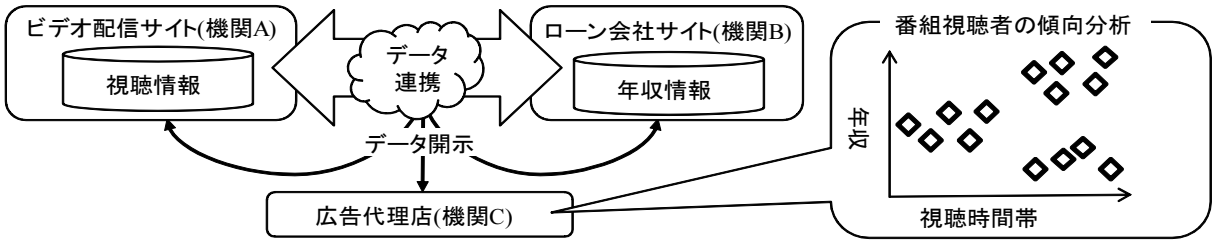


図 1.2: 「(b) 異業種のデータ連携」の例

なお，これら「(a) 医療機関のデータ連携」と「(b) 異業種のデータ連携」の利用場面において，各患者やユーザからは，個人情報保護法[4]における個人情報の利用についての同意を得ているものとする．具体的には，患者や顧客データの分析については許諾しているが，患者や顧客データの他機関への全公開はプライバシ上の懸念から許諾していないものとする．以上のような許諾内容については，通常のサービス利用において一般的な許諾内容と

[4]正確には「個人情報の保護に関する法律 (平成一五年五月三十日法律第五十七号)」

考えられる．

これら「(a) 医療機関のデータ連携」と「(b) 異業種のデータ連携」のようにパーソナル情報を結合することで，新たに有益な情報を得られる．しかしパーソナル情報を組み合わせると，その組み合わせからのユーザの特定が可能になり，他人に知られたくない情報が特定のユーザに紐付いてしまう恐れがある．例えば「(a) 医療機関のデータ連携」においては，[53] で指摘されているように，医療情報には「直接個人を特定できる情報を除去しても，個人の特定につながる可能性のある情報」が含まれている．つまり，先ほどのデータでは，たとえ「被保険者番号」のような直接個人を特定できる情報を削除したとしても，研究機関 C にいる研究員は，あるデータがだれのデータであるかを特定できてしまう可能性がある．例えばこの研究員が，患者 X さんは「1 月 1 日に機関 A に受診」し「2 月 2 日に機関 B に受診」したことを知っていたとする．そして，このような患者が全患者のなかで X さんの 1 名だけであったとする．するとこの研究員は，結合され公開されたデータのうち機関 A の「診療日」が 1 月 1 日で，機関 B の「診療日」が 2 月 2 日に該当する患者データが X さんのデータであると特定できてしまう．このように複数の情報の組合せから，あるデータがある個人のデータであるということを特定 (データの個人の特定) される恐れがある．そのため，機関 A,B は情報を開示する際の責務としてデータの個人の特定を防ぐための処理を行うべきであると言われている [53, 43]．つまり，「(問題 1) 機関 C においてデータの個人が特定される問題」の解決が必要である．

また，サービス事業者が保持するパーソナル情報は個人のプライバシに関する情報であるため，他の機関へ全開示して結合することはできない．例えば「(a) 医療機関のデータ連携」においては，米国の HIPAA(Health Insurance Portability and Accountability Act) 法における必要最小限の情報開示の要件 (minimum necessary requirements)[46] では，医療情報を開示する際には開示する情報を必要最小限にすることが求められている．つまり，医療情報を結合する際の情報開示は必要最小限にする必要がある．また「(b) 異業種のデータ連携」でも同様に，パーソナル情報はプライバシに関わる情報であると同時に，企業における情報資産とも考えられているため，パーソナル情報を他の機関へ全開示することは好ましくない．つまり，「(問題 2) 機関 A,B において必要以上にデータを開示してしまう問題」の解決が必要である．

そこで，機関 A,B が持つ情報を必要最小限の開示にとどめながら結合し，「(問題 1) 機関 C においてデータの個人が特定される問題」と「(問題 2) 機関 A,B において必要以上にデータを開示してしまう問題」の解決を行う手法として，分散匿名化手法が注目されている [15, 37, 47, 23, 24]．分散匿名化手法は，機関 A,B が持つ情報を必要最小限の開示に留めながら結合し，ユーザが特定されない形式に加工した結合匿名テーブルを生成・提供する手法である (図 1.3)．

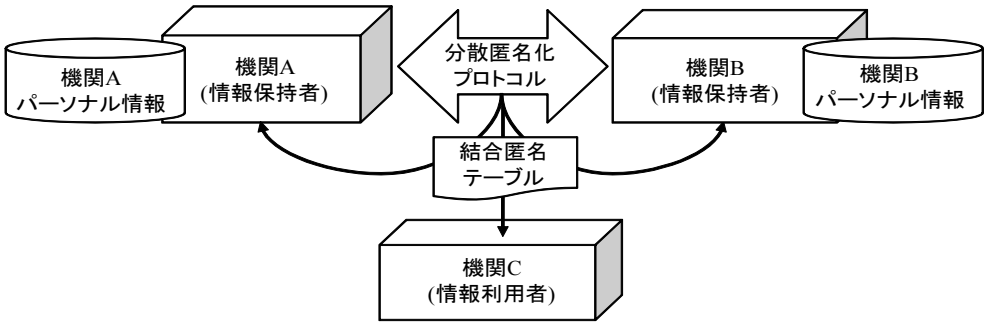


図 1.3: サービス事業者間のデータ連携と分散匿名化

しかし既存の分散匿名化の手法では，双方の機関のユーザ集合が一致しない場合に，結合匿名テーブルを参照することで，ユーザのパーソナル情報がその機関に「存在する/しない」というユーザ存在情報が，他方の機関に漏洩してしまう問題があった．例えば「(a) 医療機関のデータ連携」において機関 B が性病の専門病院であった場合，機関 A の医師が，結合匿名テーブルを参照することで，機関 A に風邪の診療の来た X さんは性病の専門病院である機関 B にも通院しているということを知ることができてしまう．このような専門病院への通院を他の一般の内科等の病院には知られたくないと考えられるため，ユーザ存在情報はユーザのプライバシに関わる情報といえる．同様に「(b) 異業種のデータ連携」の場合でも，オンデマンドビデオ配信サイト (機関 A) は，自機関のサイトを利用しているユーザがローン会社 (機関 B) に存在することを知ることになる．ユーザのパーソナル情報がローン会社に存在することは，そのユーザは借金をしていると推測される恐れがあるため，やはりユーザ存在情報はプライバシに関わる情報といえる．

また，ユーザ不在情報が知られると不利益となる場合もある．既存のユーザ存在情報の軽減を目指した研究 [39] では，企業の従業員等の採用候補者を絞り込む際に，糖尿病患者

でないことが確定している候補者と確定していない候補者がいる場合に，糖尿病患者でないことが確定している候補者を選ぶ傾向があると指摘している．これは，糖尿病患者でないことが確定していない候補者に対して不利益となる．つまり，ユーザ不在が確定することは，ユーザ不在が確定していないユーザにとって不利益になる場合がある．よって，ユーザ存在情報の漏洩だけを軽減するだけではなく，ユーザ不在情報の漏洩も同様に軽減する必要がある．

実際のアプリケーションにおいては，双方の機関でユーザ集合が一致することは稀であるため「(問題 3) 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題」は頻繁に発生すると考えられる．したがって，この「(問題 3) 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題」の解決は，分散匿名化手法を実際のアプリケーションに適用する上で重要である．

1.2 本研究の目的と貢献

本研究では，分散匿名化手法を実際のアプリケーションに適用するために，従来の分散匿名化が対象としている「(問題 1) 機関 C においてデータの個人が特定される問題」と「(問題 2) 機関 A,B において必要以上にデータを開示してしまう問題」だけでなく,「(問題 3) 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題」の解決も目指す．この問題 3 は，双方の機関が異なる属性のパーソナル情報を保持している際の分散匿名化において，双方の機関のユーザ集合が一致しない場合に発生する．実際のビジネスにおいては，双方の機関のユーザ集合が一致しない場合は多いため，この問題 3 の解決することは重要である．

本研究は，このようなユーザ存在情報が漏洩する問題の軽減を目的として行ったものであり，以下のような貢献が挙げられる．

- δ-*site-presence* という新たなプライバシ指標を提案する．この指標は，既存の集中型の匿名化におけるユーザ存在情報が知られる可能性を示した δ-*presence* [39] という指標を，分散匿名化のために拡張した指標である．この指標を用いることで，ユーザ存在情報が漏洩する可能性の許容範囲を示すことができる．

- 提案した δ-*site-presence* を満たしつつ，データマイニング等での有用性を保った結合

匿名テーブルを生成するための新たな分散匿名化手法のプロトコルを提案する．本プロトコルが目指すことは，*δ-site-presence* で示されたプライバシ要件を満たしつつ，可能な限り有用なデータを生成することである．提案プロトコルは，存在するユーザと存在しないユーザの区別を困難にさせるダミーユーザを導入することで，ユーザ存在情報の漏洩を軽減している．また，通信量と計算量を軽減させるために，双方の事業者が持つ情報を開示せずに単純な関数計算を行うセキュア計算 [32] を組合せて利用している．これにより，通信量と計算量を低く抑えながら，プライバシ性の高いデータの漏洩を防ぎつつ，ユーザ存在情報の漏洩を軽減した分散匿名化を実現できる．

- 提案プロトコルの計算量・通信量の評価を行い，既存のセキュア計算の計算量・通信量と比較して大幅に増加することは無いことを示す．これにより，データ規模が大きくなければ，適切に並列化を行うことで提案手法を実際のアプリケーションに適用可能であると考えられる．

- 提案手法を米国の国勢調査データと患者のレセプトデータを用いて評価し，提案手法の有用性を示す．レセプトデータを用いた評価では，ユーザ存在情報の漏えいを軽減しながらも相対誤差 15%以下でデータ分析が可能であることを確認している．これは，近年言われている医療の効率化や医療サービスの質向上のための医学研究に適用できると考えられる．

以上のような貢献により，本論文で提案する手法を用いることによって，国勢調査データや医療データにとどまらず，様々な種類のパーソナル情報をサービス事業者間で安全にデータ連携することができる．そして，本技術とデータを利用するための技術と連携することで，新たなサービス提供に必要な，データの生成から実際のサービス提供までを含めたアプリケーションのフレームワークを構築することできる (4.3 節)．その結果，新たなサービスが創出されることが期待できる．

1.3 本論文の構成

本論文の構成は次の通りである．まず，2 章で関連研究として，匿名化，分散匿名化，及びセキュア計算などの既存技術について説明する．次に，3 章で本論文における分散匿名

化を定義し，分散匿名化におけるユーザ存在情報が漏洩する課題について説明する．そして，4章にてユーザ存在情報の漏洩を軽減するための新たなプライバシ指標としてユーザ存在情報が漏洩する可能性の許容範囲を示す*δ-site-presence*を提案する．また，提案した*δ-site-presence*を満たしつつ，データマイニング等での有用性を保った結合匿名テーブルを生成するための新たな分散匿名化手法のプロトコルを提案する．続いて5章では，提案手法を米国の国勢調査データと実際の患者のレセプトデータを用いて評価し，提案手法の有用性を示す．そして6章では，提案手法の計算量・通信量を評価し，提案手法の計算量・通信量は既存のセキュア計算の計算量・通信量と比較して大幅な増加がないことを示す．さらに，提案手法の安全性を証明し，プライバシ性の高いデータが漏洩していないことを示す．最後に，7章で本論文をまとめる．

第2章　関連研究

本章では，本論文で提案するユーザ存在情報の漏洩を軽減した分散匿名化手法に関連する研究を説明する．まず 2.1 節において，匿名化の既存研究について説明する．続いて 2.2 節で，分散匿名化ではないが，ユーザ存在情報を隠蔽した匿名化について提案している既存研究を説明する．そして，2.3 節では分散環境における匿名化である分散匿名化の既存研究について説明する．さらに 2.4 節にてセキュア計算と Multi Party Computation について説明し，最後に 2.5 節で，プライバシを保持したデータマイニング手法である Privacy Preserving Data Mining について説明する．

2.1　匿名化とプライバシ指標

匿名化とは，あるパーソナル情報が誰に関する情報であるかを特定できないように，パーソナル情報を加工することである [15, 16]．ここでパーソナル情報とは，個人を特定することができる個人情報にとどまらず,「属性」と「属性値」として表現されるユーザ (病院や Web サービス等の利用者) に関する属性情報の集合とする．表 2.1(a) では，テーブルのレコードがユーザに，カラムが「属性」に，フィールドの値がユーザの属性の「属性値」にそれぞれ対応する．そして，単一の属性ではユーザを特定できないが，複数組み合わせるとユーザを特定できる可能性のある属性の組合せを準識別子 (Quasi-Identifier,QI) と呼ぶ．また，ユーザが特定された状態で開示されることが望ましくない属性をセンシティブ属性 (Sensitive Attribute,SA) と呼ぶ．表 2.1(a) の例では，年齢と性別という属性の組み合せが準識別子であり，病状という属性がセンシティブ属性とみなすことができる．例えば，ある病院が表 2.1(a) のような全患者 (user1〜user6) の病状を記録したテーブルを保持していたとする．そして，このテーブルを，医学研究を行う研究機関に公開するために，識別子を削除した表 2.1(b) のテーブルを作成したとする．つまり表 2.1(b) には，氏名など直接ユー

ザを識別できるような属性は含まれていない．しかし，もし表 2.1(b) を受け取った研究機関の研究員が，事前に「user6 はその病院に通院しており，年齢が 38 の女性である」ことを知っていたとする．すると，この研究員は表 2.1(b) の 6 番目のレコードが user6 のレコードであると知れてしまう．その結果この研究員は，user6 は心臓病ということを知ることができてしまう．つまり，たとえ識別子を削除したとしても，準識別子から個人特定ができる可能性がある．例えば，米国では zip コードと生年月日と性別の組合せから約 87%の米国国民を識別可能であると言われている [43].

表 2.1: k-匿名化の実行例

(a) 元テーブル

識別子	年齢	性別	疾病名
user1	12	男	かぜ
user2	18	女	ガン
user3	23	男	HIV
user4	26	男	かぜ
user5	32	女	かぜ
user6	38	女	心臓病

(b) 識別子を削除したテーブル

年齢	性別	疾病名
12	男	かぜ
18	女	ガン
23	男	HIV
26	男	かぜ
32	女	かぜ
38	女	心臓病

(c) 2-匿名化したテーブル

年齢	性別	疾病名
10-19	*	かぜ
10-19	*	ガン
20-39	男	HIV
20-39	男	かぜ
20-39	女	かぜ
20-39	女	心臓病

そこで，準識別子の属性値によってデータの個人が特定されることを防ぐために，準識別子の属性値を汎化 (generalize) して，より抽象的な値にする．このような加工により，準識別子の属性値の組合せによって識別されるレコードが少なくとも k 個以上あるテーブルを，k-匿名性 [43] を満たすという．表 2.1(c) は 2-匿名性を満たす．また，k-匿名性を満たすようにテーブルを加工することを，k-匿名化という．本論文では，単に識別子を削除することを匿名化というのではなく，準識別子の組合せから個人特定を防ぐために k-匿名化を行うことを匿名化と呼ぶ.

さらに k-匿名性の指標は拡張され，いくつかの新たな指標が提案されている．[34] ではセンシティブ属性の属性値の種類数も考慮した指標として ℓ-多様性を提案している．また [29] では，センシティブ属性の属性値の意味的な近さも考慮した指標として t-closeness を提案している．さらに，データが更新される前提におけるプライバシ指標として m-不変性

[50] なども提案されている．他にも，ノイズを付加することで k-匿名性や ℓ-多様性と同等の安全性を保つための指標として Pk-匿名性 [57] や $P\ell$-多様性 [58] も提案されている．また，位置情報における匿名化についても提案されている [36, 64, 44].

2.1.1 Top-down アプローチと Bottom-up アプローチによる匿名化

k-匿名化を行うアルゴリズムはいくつか提案されている [17, 27, 28, 26, 42, 6, 22, 49]. これらのアルゴリズムは，属性値を汎化する手法 [17, 27, 28, 26, 6, 49] や削除する手法 [6, 22, 42] など様々あるが，汎化する手法のほうがデータを削除するよりもデータの加工量が少ないとされている．そして汎化する手法は，大きく Top-down アプローチと Bottom-up アプローチに分けることができる．Top-down アプローチとは，準識別子の属性値を最も汎化されている状態から，k-匿名性を満たしている間，徐々に詳細化 (specialize) する手法である．それに対して，Bottom-up アプローチとは，準識別子の属性値を k-匿名性を満たすまで徐々に汎化していく手法である．一般に，Top-down アプローチは途中状態が常に k-匿名性を満たすため，途中で止めることが可能であることから，準識別子の数が多い場合など計算量が多くなる際でも有利とされる．

Top-down アプローチの k-匿名化を行うアルゴリズムとしては，[17, 27, 28] が良く知られている．Top-down アプローチは準識別子の属性値を徐々に詳細化するが，ここでの詳細化とは準識別子の属性値で識別されるユーザ集合を，ある境目で分割することを意味する．そして，この分割の境目となる属性値を分割点と呼ぶ．例えば，年齢を「30」という分割点で分割すると,「30 才以上」と「30 才未満」に分割することになる．そして，この分割点を決定する関数を分割点決定関数と呼ぶ.

Top-down アプローチの動作の例を，表 2.2 に示す．この例では，表 2.1(a) のテーブルを 2-匿名性を満たすように加工している．この表で,「年齢」と「性別」の組みが準識別子である．まず，表 2.2(a) のように，表 2.1(a) の全ての準識別子の値を最も汎化されている状態にする．続いて，分割点決定関数を用いて分割点を決定する．この例では,「年齢」という属性の「20」という属性値が 1 回目の分割点として決定したとする．表 2.2(b) は，1 回目の分割点での分割後のテーブルである．この例で示したように,「年齢」が「*」という最も汎化された値が「20」で分割され,「10-19」と「20-39」という値に詳細化されている．ま

た，表 2.2(b) は 2-匿名性を満たしており，かつ user3,4,5,6 の 4 レコードはさらに 2 レコードに分割可能なので，さらに分割を行う．この例では，再度分割点決定関数を計算し，2 回目の分割点として「性別」という属性の「男」という属性値が選ばれている．なお，この例のように,「性別」のような数値ではないカテゴリ値である場合は，カテゴリ値を数値に変換させることで，カテゴリ値も数値として扱うことが出来る．この例では，男を 0，女を 1 と変換して，数値として扱っている．表 2.2(c) は，2 回目の分割点での分割後のテーブルである．そして，このテーブルはこれ以上の分割を行うと，2-匿名性を満たさなくなるので，分割を終了し識別子を削除したテーブルを出力する (表 2.1(c))．

続いて，Bottom-up アプローチの動作の例を，表 2.3 に示す．なお，この例でも元のテーブルは表 2.1(a) であり，2-匿名性を満たすという前提である．Bottom-up アプローチでは，元のテーブルの状態から，k-匿名性を満たすまで汎化を繰り返すという手法である．表 2.3(a) の例では，1 回目の汎化では「年齢」という属性を「10-19」と「20-39」という属性値に汎化した例である．しかし，このテーブルは，user1 と user2 のレコードが準識別子の属性値によって 2 レコード以下に識別出来てしまうので 2-匿名性を満たしていない．そのため，この 2 レコードをさらに汎化させる．表 2.3(b) は 2 回目の汎化後のテーブルである．この例では，user1 と user2 のレコードの「性別」の属性値を「*」に汎化させている．これにより，表 2.3(c) は 2-匿名性を満たすことが出来たので，識別子を削除したテーブルを出力する (表 2.1(c))．

2.2　ユーザ存在情報の漏洩を軽減した匿名化

分散匿名化ではないが公開テーブルと匿名テーブルにおいてユーザ存在情報の隠蔽を目指した匿名化の研究がおこなわれている．[39] では，*δ-presence* というユーザの存在の可能性を示す指標と，その指標を満たすための匿名化アルゴリズムを提案している．

δ-presence は，公開テーブル T_1 と匿名化されたテーブル T_2^* における，T_1 に存在するユーザのレコード内のデータが T_2^* にも存在する可能性を示した指標である．この T_2^* とは，T_1 の一部のレコードのデータから構成されたテーブル $T_2(T_2 \in T_1)$ を匿名化したテーブルである．

表 2.4 の例を用いて説明する．例えば表 2.4(a) の T_1 が，ある会社の社員名簿のテーブル

表 2.2: Top-down アプローチによる k-匿名化の例

(a) 初期状態のテーブル

識別子	年齢	性別	疾病名
user1	*	*	かぜ
user2	*	*	ガン
user3	*	*	HIV
user4	*	*	かぜ
user5	*	*	かぜ
user6	*	*	心臓病

(b) 1 回目の分割後のテーブル

識別子	年齢	性別	疾病名
user1	10-19	*	かぜ
user2	10-19	*	ガン
user3	20-39	*	HIV
user4	20-39	*	かぜ
user5	20-39	*	かぜ
user6	20-39	*	心臓病

(c) 2 回目の分割後のテーブル

識別子	年齢	性別	疾病名
user1	10-19	*	かぜ
user2	10-19	*	ガン
user3	20-39	男	HIV
user4	20-39	男	かぜ
user5	20-39	女	かぜ
user6	20-39	女	心臓病

表 2.3: Bottom-up アプローチによる k-匿名化の例

(a) 1 回目の汎化後のテーブル

識別子	年齢	性別	疾病名
user1	10-19	男	かぜ
user2	10-19	女	ガン
user3	20-39	男	HIV
user4	20-39	男	かぜ
user5	20-39	女	かぜ
user6	20-39	女	心臓病

(b) 2 回目の汎化後のテーブル

識別子	年齢	性別	疾病名
user1	10-19	*	かぜ
user2	10-19	*	ガン
user3	20-39	男	HIV
user4	20-39	男	かぜ
user5	20-39	女	かぜ
user6	20-39	女	心臓病

T_1 であり，社内で公開されているとする．表 2.4(b) が社員に対して HIV 検査を行った結果の非公開テーブル T_{priv} であるとする．そして，表 2.4(c) が HIV 検査の結果が陽性であった社員のリストを格納した非公開テーブル T_2 であるとする．当然，HIV に感染していることはプライバシに関わる情報であるので，ある社員が T_2 に存在するというユーザ存在情報はプライバシに関わる情報となる．

ここで T_2 を医学研究のために k-匿名化して研究者に公開することを考える．もし，T_2 を k-匿名化した結果のテーブル T_2^* が表 2.4(d) であった場合，T_1 と T_2^* を入手した研究者は，T_1 と T_2^* を比較することによりユーザ存在情報を推測出来てしまう．この場合，まず T_2^* に注目すると，年齢が「30-31」かつ性別が「男」のレコードは 2 つある．続いて，T_1 に注目すると，年齢が「30-31」かつ性別が「男」に該当するレコードは user1 と user2 の 2 名である．これにより，user1 と user2 は確実に T_2^* に存在することがわかり，user1 と user2 が HIV 患者であることを知ることができてしまう．

それに対し，もし，T_2 を k-匿名化した結果のテーブル T_2^* が，表 2.4(e) であった場合を考える．この場合 T_2^* に注目すると，年齢が「30-32」かつ性別が「*」(男性 or 女性) に該当するレコードは 2 つある．続いて，T_1 に注目すると，年齢が「30-32」かつ性別が「*」に該当するレコードは user1,user2,user3 の 3 名である．つまり，user1,2,3 の 3 名のうち 2 名が HIV 患者であることがわかるが，だれが HIV 患者であることまでは知ることは出来ない．なお，この時の，T_1 に存在するユーザが T_2^* にも存在する可能性は $\frac{2}{3}$ となる．[39] は，

このようなユーザ存在情報の可能性の許容範囲を指定することが出来るプライバシ指標として．*δ-presence* を提案している．そして，*δ-presence* で示されたユーザ存在情報の可能性の許容範囲を満たすように匿名テーブルを生成することで，ユーザ存在情報の漏洩を防ぐことを提案している．

表 2.4: δ-presence を満たす匿名化の実行例

(a) 公開テーブル (T_1)

社員 ID	年齢	性別
user1	30	男
user2	31	男
user3	32	女
user4	33	女
user5	34	女
user6	35	男

(b) 検査結果テーブル (T_{priv})

社員 ID	検査結果
user1	陽性
user2	陽性
user3	陰性
user4	陽性
user5	陽性
user6	陰性

(c) 感染者テーブル (T_2)

社員 ID	年齢	性別
user1	30	男
user2	31	男
user4	33	女
user5	34	女

(d) ユーザ存在情報が漏洩する匿名テーブル (T_2^*)

年齢	性別
30-31	男
30-31	男
33-34	女
33-34	女

(e) ユーザ存在情報が漏洩しにくい匿名テーブル (T_2^*)

年齢	性別
30-32	*
30-32	*
33-35	*
33-35	*

さらに [39] では，*δ-presence* を満たすような匿名化を実現するためのアルゴリズムとして，Single-Dimensional Presence Algorithm (SPALM) と，Multi-Dimensional Presence Algorithm (MPALM) を提案している．SPALM は Bottom-up のアルゴリズムであり，準

識別子の属性数が少ない場合に利用可能なアルゴリズムである．それに対し，MPALM は Top-down のアルゴリズムであり，準識別子の属性数が多い場合にも対応したアルゴリズムである．

しかし，これらのアルゴリズムは分散匿名化ではないため，双方の機関でユーザが異なる場合におけるユーザ存在情報の隠蔽課題には適用できない．また，提案されている *δ-presence* という指標は分散匿名化のための指標では無い．そこで，そこで本論文では，*δ-presence* を分散匿名化に適用した指標を *δ-site-presence* として新たに定義し，さらに *δ-site-presence* を満たすための分散匿名化のプロトコルを提案している．

2.3　分散匿名化

複数の機関が保持するテーブルを結合して匿名化する処理を分散匿名化 (Distributed Anonymization) と呼ぶ [37, 47, 23, 24]．分散匿名化は，パーソナル情報の分割形態の違いにより垂直分割と水平分割に分類される．垂直分割とは，本論文と同様にユーザのパーソナル情報が属性毎に異なる機関に保持されている分割形態である (表 2.5)．水平分割とは，ユーザのパーソナル情報がユーザ毎に異なる機関に保存されている分割形態である (表 2.6)．

垂直分割での分散匿名化としては [37, 47, 23] などが存在する．[37, 47] では，本論文と同じ Top-down アプローチとセキュア計算 (secure computation)[32, 51] を組み合わせた手法で，分散匿名化を実現している．Top-down アプローチでは，準識別子を詳細化することでグループを徐々に分割していくが，この分割後のユーザ集合のユーザ ID は，双方の機関で共有される (図 2.1)．そして *k*-匿名性が満たされている間，分割を続ける．最後に，分割した双方のテーブル (内部匿名テーブル) を結合して最終的な結合匿名テーブルを生成する．Top-down アプローチで分割点を決定するために，分割点決定関数というヒューリスティック関数が用いられる．分散匿名化では，この関数の計算にセキュア計算 [32] を用いる．セキュア計算とは，自機関が持つ属性値を相手の機関に秘密にしながら，大小比較などが行える暗号プロトコルである．セキュア計算を用いる事で，属性値を相手機関に隠蔽しながら分割点を決定することができる．

Bottom-up アプローチを用いた垂直分割での分散匿名化も提案されている [23]．[23] は，

表 2.5: 垂直分割データの分散匿名化の実行例

(a) 事業者 A(T_A)

userID	年収
user1	450 万
user2	300 万
user3	650 万
user4	550 万

(b) 事業者 B(T_B)

userID	時刻	番組
user1	16:15	Y ドラマ
user2	17:30	X アニメ
user3	14:45	Z ドラマ
user4	12:00	X アニメ

(c) 結合匿名テーブル (T^*)

年収 (万)	時刻	番組
500 未満	16:00-	Y ドラマ
500 未満	16:00-	X アニメ
500 以上	-15:59	Z ドラマ
500 以上	-15:59	X アニメ

表 2.6: 水平分割データの分散匿名化の実行例

(a) 事業者 A(T_A)

userID	年収	時刻	番組
user1	450 万	16:15	Y ドラマ
user3	650 万	14:45	Z ドラマ

(b) 事業者 B(T_B)

userID	年収	時刻	番組
user2	300 万	17:30	X アニメ
user4	550 万	12:00	X アニメ

(c) 結合匿名テーブル (T^*)

年収 (万)	時刻	番組
500 未満	16:00-	Y ドラマ
500 未満	16:00-	X アニメ
500 以上	-15:59	Z ドラマ
500 以上	-15:59	X アニメ

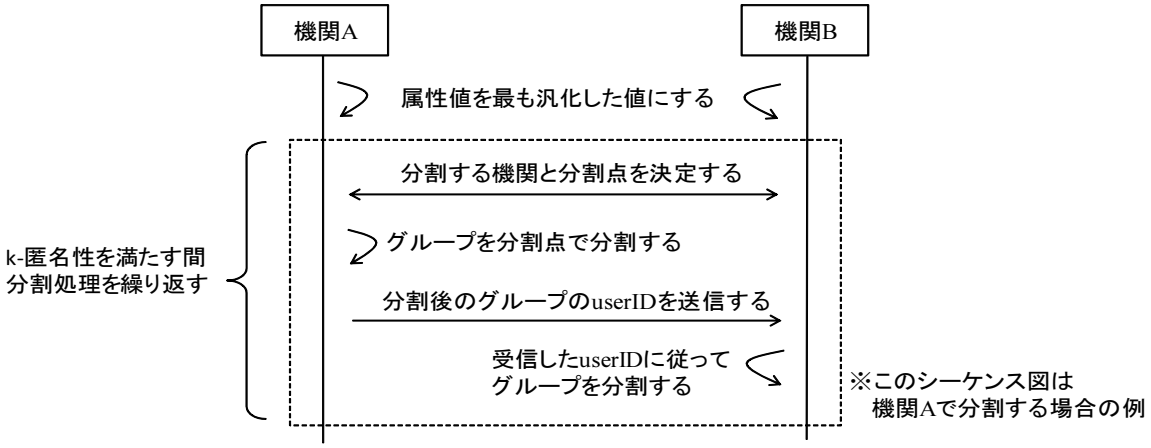


図 2.1: Top-down アプローチによる分散匿名化の処理シーケンス

それぞれの機関で個別に内部匿名テーブルを生成した後，結合匿名テーブルの匿名性が保たれることを確認しながら内部匿名テーブルを結合していく手法である．

水平分割での分散匿名化としては，[24] が知られている．[24] は水平分割での分散匿名化で発生するパーソナル情報の保存形式の違いから，情報の保存場所を知られてしまうという問題を，Top-down アプローチで解決している．また，この問題を解決するため *ℓ-site-diversity* という指標を提案している．さらに，提案した手法がプライバシ性の高いパーソナル情報を漏らしていないという安全性の評価を行っている．

2.4　セキュア計算と Multi Party Computation

セキュア計算とは，複数の機関が持つ値を，お互いに秘密にしながらそれらの値を入力とした関数を計算できる暗号プロトコルのことである [32]．このような，暗号プロトコルは，1986 年の Yao による研究 [51] が始まりとされている．[51] では，信頼のおける第三者 (Trusted Third Party, TTP) が存在しないという仮定において，2 つの機関がそれぞれ持つ秘密の値を，引数とする任意の関数を計算できることを示した[1]．これは，その後 [20, 19] において，複数機関が持つ秘密の値に対応するように拡張され，Multi Party Computation(MPC) と呼ばれている [7, 5]．MPC は，計算対象となる関数を AND と OR

[1]正確には，多項式時間で計算可能な任意の関数を計算できることを示した

の論理回路に変換し，AND や OR の論理回路の 1 つについて暗号理論を用いた手法を利用して，入力を秘密にしながら 1 つの論理回路の計算を行う方式で実現される [73].

セキュア計算は，このような任意の関数の計算が可能な MPC とは異なり，単純な関数の計算を可能とした暗号プロトコルにあたる．また，MPC は任意関数に対応するために関数を論理回路に変換して計算を行う．そのため，計算量と通信量が大きくなる問題がある．それに対しセキュア計算は，ある関数についての計算にだけ対応することで，MPC よりも計算量と通信量を抑えることができる[2].

セキュア計算のプロトコルはいくつか存在し，著者が知る限り以下の種類の関数計算を行うことができる [32, 1]．なお，これらのプロトコルでは暗号理論における安全性が証明されている．

- 大小比較 [51]
- 内積計算 [18, 67]
- 多項式計算 [38]
- 積集合計算 [14, 3, 62]
- 和集合計算 [25]
- log 計算 [30]

2.5 Privacy Preserving Data Mining

また複数の機関が持つ値を，お互いに秘密にしながらデータマイニングを行った結果を得るという研究が存在する [63, 68, 1, 52, 31, 2, 12, 10]．このような研究は，PPDM(Privacy Preserving Data Mining) と呼ばれる．PPDM は，匿名化とは違ってデータマイニングを行う点が大きな違いである．つまり，匿名化はデータを提供するだけで実際のデータマイニングまでは行わないが，PPDM はデータマイニングまで行う．そのため，匿名化は PPDM に対して，PPDP(Privacy Preserving Data Publishing) と呼ばれている [15, 16].

[2]積集合計算を実現するセキュア計算の実装 [11] では，要素数 5000 個の積集合の計算を約 2 秒で行える．

PPDM では，大きく Multi Party Computation やセキュア計算などの暗号プロトコルを利用する手法と，ノイズを付加する手法とが存在する．例えば暗号プロトコルを利用する手法 [52, 31, 2, 12, 10] では，セキュア計算を用いた近傍検索を行う手法 [52, 10, 2] や，分類木を作成する手法 [31] などが提案されている．

ノイズを付加する手法としては [4] が良く知られている．この手法では，ある確率分布のノイズを付加したデータから分類木を作成する手法である．まず，ある機関が持つ秘密の値 $\{x_1, \cdots, x_n\}$ に対して確率分布 Y の乱数 $\{y_1, \cdots, y_n\}$ を付加し，乱数が付加された値 $\{w_1 = x_1+y_1, \cdots, w_n = x_n+y_n\}$ を公開する．そして，この乱数が付加された値を受け取った機関は，確率分布 Y を知っている前提において，公開された $\{w_1 = x_1+y_1, \cdots, w_n = x_n+y_n\}$ から，元の値である $\{x_1, \cdots, x_n\}$ の確率分布を推定する．[4] では，ベイズの定理を用いて元の値の確率分布を推定する手法を提案している．つまり，たとえ乱数が付加されたとしても，乱数の確率分布を知っていれば元の値の分布を推定でき，分類木を作成可能である．

第3章　分散匿名化におけるユーザ存在情報の漏洩の課題

本章では，分散匿名化におけるユーザ存在情報の漏洩の課題を説明する．まず，3.1 節で本論文における分散匿名化を定義する．続いて，3.2 節では分散匿名化において，双方の機関のユーザ集合が一致しない場合に発生する，ユーザ存在情報の漏洩の課題について説明する．

3.1　分散匿名化の定義

本節では，本論文における分散匿名化を定義する．まず，3.1.1 節で本論文の分散匿名化における各テーブルの形式を定義する．その後 3.1.2 節で信頼モデルを定義する．

3.1.1　テーブル形式の定義

機関 A,B が保持するパーソナル情報のテーブル形式を定義する．本論文における分散匿名化は，垂直分割データの分散匿名化にあたる[1]．機関 A はテーブル T_A を，機関 B はテーブル T_B を保持するとする．そして，T_A は ID と QI_A(機関 A が持つ準識別子) という属性を保持し，T_B は ID と QI_B(機関 A が持つ準識別子) と SA(センシティブ属性) という属性を保持するテーブル形式である (図 3.1)．本論文では，このことを以下のように表記する．

$$T_A(ID, QI_A),\ T_B(ID, QI_B, SA)$$

ここで，ID は機関 A と機関 B において共通のユーザ ID である．本研究では，このように ID は機関 A と機関 B において共通であるという前提を置いてあるが，これは実際のアプリケーションにおいて十分現実的であると考える．例えば，1.1 節で説明した「(a) 医療

[1] 垂直分割データの分散匿名化については 2.3 節で説明している

機関のデータ連携」という利用場面においては,「被保険者番号」が機関 A と機関 B において共通の ID となる．また「(b) 異業種のデータ連携」においては，例えば機関 A と機関 B が同一の OpenID Provider[40] を用いている場合は，共通の ID を使うことができる．このような例から，ID が機関 A と機関 B において共通であるという前提は，十分現実的であると考える．

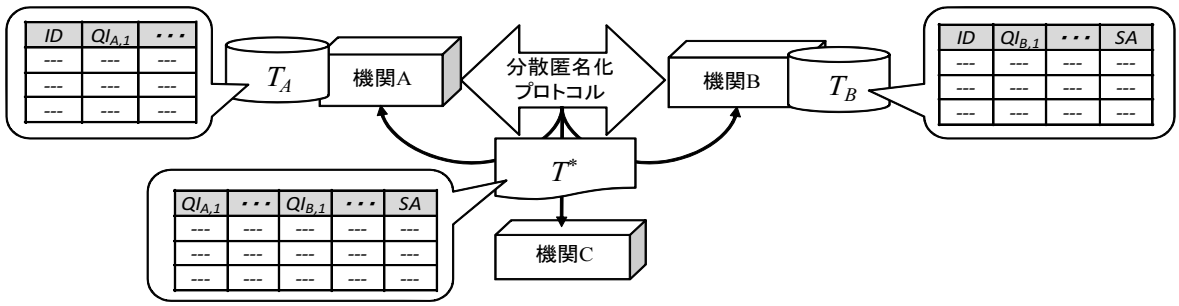


図 3.1: 分散匿名化の T_A，T_B，T^* の関係

分散匿名化によって生成される結合匿名テーブル T^* の形式は,

$$T^*(QI_A, QI_B, SA)$$

とする．ここで，T^* の QI_A と QI_B の属性値は，属性値の組合せからの個人特定を防ぐために，T^* が k-匿名性を満たすように加工されている．つまり T^* は以下で定義される k-匿名性を満たす.

定義 1 (T^* の k-匿名性) テーブル T^* において，QI_A と QI_B の属性値の組み合わせによって識別されるレコードの数が，少なくとも k 個以上あるとき，T^* は k-匿名性を満たすとする.

なお，本論文では，k-匿名性を満たすために属性値が汎化されるという前提とする.

このように T^* は k-匿名性を満たすので，1 章で説明した以下の問題 1 を解決することができる.

問題 1 機関 C においてデータの個人が特定される問題

3.1.2 信頼モデルの定義

続いて，本論文の分散匿名化における信頼モデルを定義する．本論文の分散匿名化では，既存の分散匿名化 [37, 47, 23, 24] と同様に双方の機関のテーブルの全開示をせずに，必要最小限の開示に留めながらテーブルを結合し匿名化を行うことを目指す．つまり，以下の問題 2 を解決することになる．

問題 2 機関 A,B において必要以上にデータを開示してしまう問題

ただし，既存の分散匿名化 [24] と同様に，プライバシ性の低い統計情報の開示は許すものとする．

ここで，プライバシ性の低い統計情報の開示を許しつつ必要最小限の開示に留る必要があるのは，異なる機関で完全な信頼関係を築くのは困難であり，テーブルを全て開示するのは危険であると考えられているためである．ただし，本論文では機関 A,B はある程度の信頼がおける機関であることを前提とし，各機関は semi-honest[21] で振舞うとする．semi-honest とは，各機関はプロトコルを介して得られた情報を解析して相手機関の情報を知ろうとするが，プロトコルを逸脱した攻撃は行わないという振る舞いモデルのことである．つまり，例えば機関 A が，機関 B に保持されているパーソナル情報を得るために機関 B に何度も分割を行わせるような，プロトコルを逸脱した攻撃は想定しない．

また，必要最小限の開示とは，ユーザが特定されない形式として開示された情報よりも詳しい情報が開示されていないということを意味する．例えば，結合匿名テーブルとして開示される診療日が年月レベルであった場合，機関 A,B の双方に開示される診療日は年月日レベルであってはならず，年月レベルにとどめなければならない．

3.2 ユーザ存在情報の漏洩の課題

既存の垂直分割の分散匿名化では，双方の機関のユーザ集合が一致している前提があった [37, 47, 23]．しかし，分散匿名化を実際のアプリケーションに適用することを考えると，様々な機関同士でのパーソナル情報の結合が期待されるため，ユーザ集合が一致しない場合への対応が必要となる．つまり，一部のユーザが片方の機関にだけ存在する場合にも対応する必要がある．

ユーザ集合が完全に一致せずに一部のユーザだけが一致する場合 (一部のユーザだけが共通ユーザとなる場合)，結合匿名テーブル T^* は機関 A と機関 B の共通ユーザのレコードだけとなり，片方の機関にだけ存在するユーザのレコードは含まれない．この場合，以下の問題 3 が発生する．

問題 3 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題

この問題 3 は，さらに以下の問題 3-1 と問題 3-2 に分割できる．

問題 3-1 結合匿名テーブルによるユーザ存在情報の漏洩問題

問題 3-2 ユーザ ID 通知によるユーザ存在情報の漏洩問題

以降の節でこれら問題 3-1 と問題 3-2 の詳細を説明する．なお，1.1 節で述べたとおりユーザ存在情報は患者のプライバシに関わる情報であるので，機関 A,B にユーザ存在情報が漏洩することを防ぐ必要がある．

3.2.1　結合匿名テーブルによるユーザ存在情報の漏洩

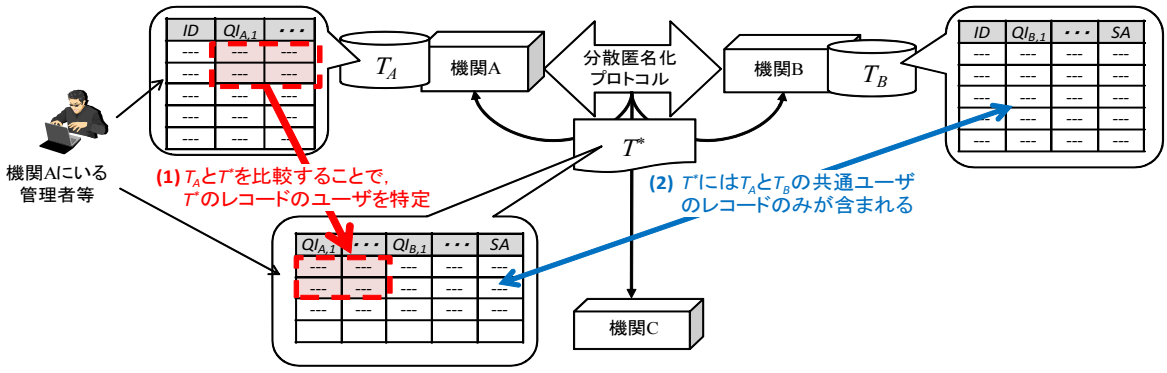


図 3.2: (問題 3-1) 結合匿名テーブルによるユーザ存在情報の漏洩問題

まず「(問題 3-1) 結合匿名テーブルによるユーザ存在情報の漏洩問題」について説明する．この問題は，自機関が持つテーブルと結合匿名テーブルの比較によってユーザ存在情報が漏洩してしまう問題である (図 3.2)．例えば機関 A がもつテーブル T_A が表 3.1(a)，機関 B がもつテーブル T_B が表 3.1(b)，結合匿名テーブル T^* が表 3.1(c) であったとする．

表 3.1: 結合匿名テーブルによるユーザ存在情報の漏洩

(a) 事業者 A(T_A)

userID	年収
user1	300 万
user2	400 万
user3	550 万
user6	600 万
user7	650 万
user8	700 万

(b) 事業者 B(T_B)

userID	時刻	番組
user1	16:00	X ドラマ
user2	17:00	Y アニメ
user4	17:30	X ドラマ
user5	16:30	Y アニメ
user6	15:00	X ドラマ
user7	12:00	Y アニメ
user9	14:00	Y アニメ
user10	14:30	X ドラマ

(c) ユーザ存在情報が漏洩する結合匿名テーブル (T^*)

年収 (万)	時刻	番組
500 未満	16:00 以降	X ドラマ
500 未満	16:00 以降	Y アニメ
500 以上	15:59 以前	X ドラマ
500 以上	15:59 以前	Y アニメ

(d) ユーザ存在情報が漏洩しにくい結合匿名テーブル (T^*)

年収 (万)	時刻	番組
600 未満	16:00 以降	X ドラマ
600 未満	16:00 以降	Y アニメ
600 以上	15:59 以前	X ドラマ
600 以上	15:59 以前	Y アニメ

表 3.1 では,「年収」と「時刻」を 3.1.1 節における QI_A と QI_B,「番組」を SA としている．つまり，T_A(表 3.1(a)) の「年収」と T_B(表 3.1(b)) の「時刻」の値は，汎化されていない元の値である．また，T^*(表 3.1(c)) の「年収」と「時刻」の値は，汎化された値である．例えば，T^*(表 3.1(c)) の年収の「500 万未満」という汎化された値は，年収の値が 500 万未満の範囲であることを意味する．

まず，結合匿名テーブル T^* が表 3.1(c) のように「500 万」で分割されていた場合を考える．この場合，T^*(表 3.1(c)) では年収 500 万未満は 2 名，T_A(表 3.1(a)) も年収 500 万未満は user1,2 の 2 名である．このことから事業者 A は，user1,2 の 2 名は確実に T^*(表 3.1(c)) に含まれていると推測できる (図 3.2 の (1))．さらに，T^*(表 3.1(c)) に含まれるユーザは事業者 A と事業者 B の双方に存在する共通ユーザである (図 3.2 の (2))．よって，事業者 A は，user1,2 の 2 名が確実に事業者 B にも存在することが推測できる．

それに対し結合匿名テーブル T^* が表 3.1(d) のように「600 万」で分割されていた場合を考える．この場合，T^*(表 3.1(d)) では年収 600 万未満は 3 名，T_A(表 3.1(a)) では年収 600 万未満は user1,2,3 の 3 名である．そのため，事業者 A は，user1,2,3 の 3 名のうちいずれか 2 名が事業者 B に存在することまでしか推測できない．

3.2.2　ユーザ ID 通知によるユーザ存在情報の漏洩

次に,「(問題 3-2)「ユーザ ID 通知によるユーザ存在情報の漏洩問題」について説明する．これは，プロトコル中のユーザ ID の通知によって，相手機関に自機関のユーザ存在情報が知られてしまう問題である．2.3 節で説明した既存の分散匿名化プロトコルは，グループを徐々に分割していくという Top-down アプローチのアルゴリズムになっており，分割後のユーザ集合のユーザ ID を相手事業者に通知するという方式である．そのため，もし単純に既存の分散匿名化プロトコルを適用してしまうと，分割後のユーザを相手機関に通知する際に，自機関に存在するユーザ ID だけを相手機関に通知することになる．すると，通知を受け取った機関は，通知されたユーザ ID のユーザは通知をしてきた機関に存在し，通知されなかったユーザは存在しないということを容易に推測できてしまう (図 3.3)．

また，この問題と関連して，事前に共通のユーザを双方の機関で共有しておく方法も考えられるが，これも容易にユーザ存在情報を知られてしまう．なぜなら，共通のユーザは

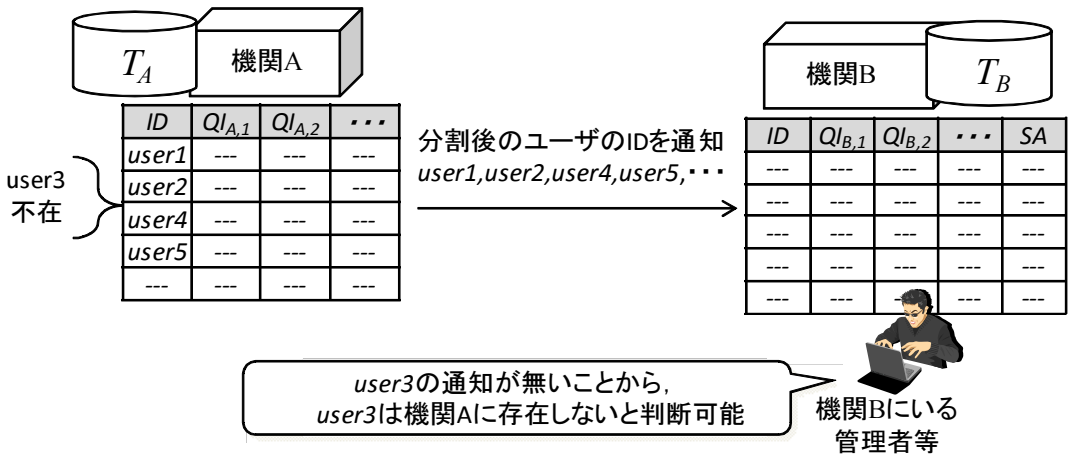


図 3.3: (問題 3-2) ユーザ ID 通知によるユーザ存在情報の漏洩問題

機関 A にも機関 B にも存在するユーザであるので，共通のユーザが明確になった時点でそのユーザは相手機関に存在することが確定してしまう．このように，既存の分散匿名化のプロトコルをそのまま用いることはできず，新たなプロトコルが必要となる．

第4章　ユーザ存在情報の漏洩を軽減した分散匿名化手法の提案

本章では，従来の分散匿名化が対象としている「(問題 1) 機関 C においてデータの個人が特定される問題」と「(問題 2) 機関 A,B において必要以上にデータを開示してしまう問題」[1]だけではなく,「(問題 3) 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題」[2]を解決するための，新たなプライバシ指標と分散匿名化手法を提案する．なお，3.2 節で説明したように，問題 3 は「(問題 3-1) 結合匿名テーブルによるユーザ存在情報の漏洩問題」と「(問題 3-2) ユーザ ID 通知によるユーザ存在情報の漏洩問題」に分割できる．

本章は以下のような構成になっている．まず 4.1 節にて「(問題 3-1) 結合匿名テーブルによるユーザ存在情報の漏洩問題」を解決するために *δ-site-presence* を提案する．その後，4.2 節にて *δ-site-presence* を満たしつつ「(問題 3-2) ユーザ ID 通知によるユーザ存在情報の漏洩問題」を解決する新たな分散匿名化手法のプロトコルを提案する．そして，4.3 節にて提案手法を用いたアプリケーション構築フレームワークを提案する．

4.1　*δ*-site-presence の提案

本節では,「(問題 3-1) 結合匿名テーブルによるユーザ存在情報の漏洩問題」を解決するために，分散匿名化におけるユーザ存在情報の可能性を示す指標として，*δ-site-presence* を提案する．既存の集中型での匿名化における 2 つのテーブルの比較からのユーザ存在情報の推測の可能性を示す指標として *δ-presence*[39] がある．そこで，この指標を分散匿名化に適用し，事業者におけるユーザ存在情報の推測の可能性を示す指標として *δ-site-presence* を定義する．

[1]問題 1 と問題 2 については，3.1 節にて詳しく説明している．
[2]問題 3 については，3.2 節にて詳しく説明している．

まず，*δ-presence* で定義されているユーザ存在情報の推測の可能性について説明する．あるテーブル T_1 と T_2 が存在し，T_2 は T_1 に存在する一部のユーザのレコード内のデータから構成されたテーブルとする．また，あるテーブル T のレコード数を $|T|$ と表現する．このとき [39] では，テーブル T_1 に存在するユーザのレコード内のデータが T_2 にも存在する可能性を $|T_2|/|T_1|$ と定義している．

そして，この定義を事業者が保持するテーブルと結合匿名テーブルとの比較の場合に適用する．例えば事業者 A のテーブル T_A が表 3.1(a)，結合匿名テーブル T^* が表 3.1(d) であった場合，T_A(表 3.1(a)) のうち「年収 600 万未満」に該当するレコードは user1,2,3 の 3 名分である．そして，T^*(表 3.1(d)) のうち「年収 600 万未満」のレコードは 2 名分である．このとき，T^* は T_A の一部のレコード内のデータから抜き出されたテーブルであるので，先ほどの定義より事業者 A の user1,2,3 が T^* にも存在する可能性は 2/3 となる．ここで，T^* は事業者 A,B の双方に存在する共通ユーザのテーブルであるため，user1,2,3 が事業者 B に存在する可能性は同じく 2/3 となる．同様に事業者 B のテーブル T_B が表 3.1(b)，結合匿名テーブル T^* が表 3.1(d) であった場合，T_B(表 3.1(b)) のうち「16:00 以降」「X ドラマ」に該当するレコードは user2,4 の 2 名分である．そして，T^*(表 3.1(d)) のうち「16:00 以降」「X ドラマ」のレコードは 1 名分である．よって，事業者 B からみて user2,4 が事業者 A に存在する可能性は 1/2 となる．他のレコードも同様に考えることができる．

このような双方の事業者のユーザ存在情報の推測の可能性を示した指標を，*δ-site-presence* として定義する．

定義 2 (*δ*-site-presence) T_A,T_B を機関 A,B が持つテーブル，T^* を結合匿名テーブルとする．そして，T^* のうち機関 $n \in \{A, B\}$ が持つ属性の属性値の組合せの集合を $\{v_{n,1}, \cdots, v_{n,m_n}\}$ とし，$v_{n,i} \in \{v_{n,1}, \cdots, v_{n,m_n}\}$ とおく．また，$v_{n,i}$ で識別されるテーブル T_n のレコード数を $|T_n[v_{n,i}]|$，$v_{n,i}$ で識別されるテーブル T^* のレコード数を $|T^*[v_{n,i}]|$ と表現する．この時，以下の式で示されるように，機関 $n \in \{A, B\}$ の各 $v_{n,i}$ によるユーザ存在情報の推測の可能性が $\delta_{max,n}$ 以下かつ $\delta_{min,n}$ 以上である時，T^* は $\{\delta_{min,A}, \delta_{max,A}, \delta_{min,B}, \delta_{max,B}\}$-*site-presence* を満たすと定義する．

$$\delta_{min,n} \leq \frac{|T^*[v_{n,i}]|}{|T_n[v_{n,i}]|} \leq \delta_{max,n} \quad \forall n \in \{A, B\} \tag{4.1}$$

例えば表 3.1(d) のうち機関 A が持つ属性 (年収) の属性の属性値の組合せの集合 $\{v_{A,1}, v_{A,2}\}$ は {600 万未満, 600 万以上 } である．まず,「600 万未満」について考える．T^*(表 3.1(d)) のうち年収が「600 万未満」であるレコードは 2 つであるので，$|T^*[v_{A,1}]|$=2 である．そして，T_A(表 3.1(a)) のうち年収が「600 万未満」を満たすレコードは 3 つであるので，$|T_A[v_{A,1}]|$=3 である．よって，年収の「600 万未満」についてはユーザ存在情報の推測の可能性は 2/3 である．同様に「600 万以上」についても，ユーザ存在情報の推測の可能性は 2/3 である．続いて，機関 B が持つ属性 (時刻, 分類) の属性値の組合せの集合についても同様に計算すると，表 3.1(d) は $\{\frac{2}{3}, \frac{2}{3}, \frac{1}{2}, \frac{1}{2}\}$-*site-presence* を満たすテーブルであることがわかる．

このように *δ-site-presence* は，*δ-presence* のようにテーブルにおけるユーザ存在情報を示しているのではなく，機関におけるユーザ存在情報を示している．これは例えば，*δ-site-presence* を 3 つ以上の機関へ拡張した場合 (4.1.2 節で説明)，単に結合匿名テーブルにユーザが存在するか否かを意味するのではなく，自機関に存在するユーザが他の 2 つの機関の両方にも存在するか否かを意味する．このように *δ-site-presence* は分散匿名化において重要な，機関におけるユーザ存在情報を表すことが出来る．

4.1.1 δ-site-presence の設定の指針

本節では，*δ-site-presence* の $\delta_{min,n}, \delta_{max,n} (n \in \{A, B\})$ をどのような指針で設定するかについて説明する．まず $\delta_{min,n}, \delta_{max,n}$ に設定可能な理論上の限界について説明し，その後設定の指針について述べる．

δ-site-presence の理論限界

δ-site-presence の $\delta_{min,n}, \delta_{max,n}$ に設定できる値には理論上の限界が存在し，ユーザ人数 (レコード数) から求めることができる．*δ-presence* の研究 [39] で述べられているように，ユーザ存在情報の確率は共通ユーザ数 (T^*) と機関 n のユーザ数 (T_n) によって，ある程度決定される．例えば，表 3.1 のように機関 A のユーザ数 ($|T_A|$) が 6 で，共通ユーザ数 ($|T^*|$) が 4 であった場合を考える．この場合，機関 A の 6 人のうち 4 人が共通ユーザであることから，T_A のレコードが T^* に存在する可能性は $4/6 = 2/3$ となる．つまり，機関 A のユー

ザが機関 B にも存在する可能性は少なくとも 2/3 であると言える．これは，表 3.1 の場合，$\delta_{max,A}$ を $\frac{2}{3}$ よりも小さくすることは出来ないことを意味する．$\delta_{min,A}$ についても同様なことが言え，$\delta_{min,A}$ を $\frac{2}{3}$ よりも大きくすることはできない．このように，$\delta_{max,n}$ は $|T^*|/|T_n|$ よりも小さく出来ず，$\delta_{min,n}$ は $|T^*|/|T_n|$ よりも大きく出来ない．つまり，$\delta_{min,n}$ と $\delta_{max,n}$ は以下の範囲で設定される必要がある．

$$0 \leq \delta_{min,n} \leq \frac{|T^*|}{|T_n|} \leq \delta_{max,n} \leq 1 \qquad (n \in A, B) \tag{4.2}$$

本論文では，$|T^*|/|T_n|$ を δ-*site-presence* の理論限界値と呼ぶ．

設定の指針

δ-*site-presence* の $\delta_{min,n}, \delta_{max,n}$ に設定するべき値は，扱うパーソナル情報の種類に依存する．例えば，ユーザ存在情報が漏洩してもプライバシ侵害の被害が小さいと考えられるような場合は $\delta_{max,n}$ の値は大きく設定し，ある程度のユーザ存在情報の漏洩を許容するようにしてもよい．逆に，例えば犯罪者データベースに存在するかどうかという情報のように，ユーザ存在情報が漏洩した際のプライバシの侵害が大きい場合は $\delta_{max,n}$ の値は小さく設定するべきである．特に，米国政府の「Centers for Medicare & Medicaid Service」[3] という医療情報の提供サービスでは，個人特定が困難なようにデータを加工する際には，個人を 10 人以下に特定されないようにすることが定められている [60]．この考えをユーザ存在確率に当てはめて考えると，ユーザ存在確率が $\frac{1}{10}$ 以下になることを禁じていると考えられる．

このようなことから，ユーザ存在情報が漏洩した際のプライバシ侵害の被害が大きい場合は，$\delta_{max,n}$ を 0.1 以下に設定する必要があると考える．逆に，ユーザ存在情報が漏洩した際にプライバシの被害が小さい場合は，$\delta_{max,n}$ を 0.9 付近に設定し，ユーザ存在が確定しないように設定すれば十分と考える．$\delta_{min,n}$ についても $\delta_{max,n}$ と同様の考え方で，ユーザ不在情報が漏洩した際のプライバシの被害の大きさから設定値を決めると良い．

また他の設定方針として，ユーザ存在情報が推測された際の被害額をもとに，これらの値を設定する方法もある．例えば，既存研究の [39] では，糖尿病患者であるかどうかを他

[3] `http://www.cms.gov/`

人に知られた場合における被害額から，許容するユーザ存在情報の推測確率を求める方法が提案されている．この方法は，ユーザ存在情報が推測される確率から被害額を算出する式を求め，その式を用いて許容する被害額から許容するユーザ存在情報の推測確率を逆算するというものである．

以上のような設定方針を踏まえつつ，さらにデータの有用性も考慮したうえで適切な *δ-site-presence* の設定を行う．また実際の運用では，ユーザや関連する事業者との対話をとおして決定していくことが望ましい．

4.1.2 3 つ以上の機関への拡張の検討

本論文で提案する *δ-site-presence* は 2 機関に限定した指標となっているが，この指標を拡張し 3 機関以上でも適用可能であることを示す．まず，例として 3 事業者の場合を考える．この場合，例えば機関 A, 機関 B と機関 C が存在し，それぞれが持つ T_A,T_B,T_C を結合して匿名化した T^* を生成するとする．この時，T^* には T_A,T_B,T_C に含まれる共通ユーザのレコードのみとなる．そして，例えば機関 A から見た場合，機関 A のユーザが機関 B,C の両方にも存在する可能性は，T_A で識別されるレコードのうち，どのくらいのレコードが T^* に存在するかという可能性になる．つまり，3 機関の場合の *δ-site-presence* は，ある機関のユーザが他の 2 機関にも存在する可能性を指定する指標となる．

3 機関の場合の *δ-site-presence* の拡張方法を踏まえ，3 機関以上の場合の *δ-site-presence* について定義する．

定義 3 (3 機関以上の場合の δ-site-presence) $\{T_1, \cdots, T_N\}$ を機関 $n \in \{1, \cdots, N\}$ が持つテーブル，T^* を $\{T_1, \cdots, T_N\}$ の結合匿名テーブルとする．そして，T^* のうち機関 n が持つ属性の属性値の組合せの集合を $\{v_{n,1}, \cdots, v_{n,m_n}\}$ とし，$v_{n,i} \in \{v_{n,1}, \cdots, v_{n,m_n}\}$ とおく．また，$v_{n,i}$ で識別されるテーブル T_n のレコード数を $|T_n[v_{n,i}]|$，$v_{n,i}$ で識別されるテーブル T^* のレコード数を $|T^*[v_{n,i}]|$ と表現する．この時，以下の式で示されるように，機関 n の各 $v_{n,i}$ によるユーザ存在情報の推測の可能性が $\delta_{max,n}$ 以下かつ $\delta_{min,n}$ 以上である時，T^* は $\{\delta_{min,1}, \delta_{max,1}, \cdots, \delta_{min,N}, \delta_{max,N}\}$-*site-presence* を満たすと定義する．

$$\delta_{min,n} \leq \frac{|T^*[v_{n,i}]|}{|T_n[v_{n,i}]|} \leq \delta_{max,n} \quad \forall n \in \{1, \cdots, N\} \tag{4.3}$$

このように，*δ-site-presence* を 3 つ以上の機関に拡張することは可能であるが，本論文で提案している手法をそのまま 3 つ以上の事業者で用いることは出来ない．これは，提案手法で用いているセキュア計算のいつくかは 2 機関限定となっているためである．しかし，3 機関でも動作可能なセキュア計算の研究 [32] や，3 つの機関以上の機関における分散匿名化手法 [37, 24] を参考にすることで，提案手法を 3 つ以上の機関に対応するように拡張可能であると考える．

4.1.3　簡易版指標 (*δ*-max-site-presence) の提案

本節では，提案した *δ-site-presence* の簡易的な指標として，*δ-max-site-presence* を提案する．*δ-site-presence* は，さまざまな場面のユーザ存在情報の隠蔽に対応できるような指標となっているが，その反面指定するパラメータが多くなっている．そこで，以下のような前提がある場合のための簡易的な指標として，設定するパラメータが少ない *δ-max-site-presence* を提案する．

- ユーザが存在することの隠蔽は行うが，ユーザが存在しないことの隠蔽は行わない場合．
- 機関 A,B のユーザ数がほぼ同じであり，機関 A,B におけるユーザ存在情報の漏洩の確率はほぼ同じである場合．

δ-max-site-presence は，*δ-site-presence* と比較して 2 つの違いがある．1 つ目は，ユーザ存在情報の漏洩の可能性の最大値のみを指定できるという点である．*δ-site-presence* は，ユーザ存在情報の漏洩の可能性の最大値と最小値を示すことができる指標であったが，もしユーザが存在することの隠蔽は行うが，ユーザが存在しないことの隠蔽は行わない場合は，ユーザ存在情報の漏洩の可能性の最小値の設定は不要である．そこで，*δ-max-site-presence* では，ユーザ存在情報の漏洩の可能性の最大値のみを指定できる指標としている．

2 つ目の違いは，機関 A から見たユーザ存在情報の可能性と機関 B から見た可能性に同じ値を設定するという点である．*δ-site-presence* では，ユーザ存在情報の漏洩の可能性を別々に示すような指標であったが，もし，機関 A,B のユーザ数がほぼ同じであり，機関 A,B におけるユーザ存在情報の漏洩の確率はほぼ同じである場合は，別々に指定する必要は無

い．そこで，*δ-max-site-presence* では，機関 A から見たユーザ存在情報の可能性と機関 B から見た可能性として同じ値を設定する指標としている．

このような *δ-max-site-presence* を以下のように定義する．

定義 4 (δ-max-site-presence) T_A,T_B を事業者 A,B が持つテーブル，T^* を結合匿名テーブルとする．但し，T_A,T_B にユーザ *ID* 以外の同一の属性は無いものとする．そして，T^* のうち事業者 $n \in \{A, B\}$ が持つ属性の属性値の組合せの集合を $\{v_{n,1}, \cdots, v_{n,m_n}\}$ とし，$v_{n,i} \in \{v_{n,1}, \cdots, v_{n,m_n}\}$ とおく．また，$v_{n,i}$ で識別されるテーブル T_n のレコード数を $|T_n[v_{n,i}]|$，$v_{n,i}$ で識別されるテーブル T^* のレコード数を $|T^*[v_{n,i}]|$ と表現する．この時，以下の式で示されるように，事業者 n の各 $v_{n,i}$ によるユーザ存在情報の推測の可能性が δ 以下である時，T^* は *δ-max-site-presence* を満たすと定義する．

$$\frac{|T^*[v_{n,i}]|}{|T_n[v_{n,i}]|} \leq \delta \quad \forall v_{n,i} \in \{v_{n,1}, \cdots, v_{n,m_n}\} \quad \forall n \in \{A, B\} \tag{4.4}$$

例えば表 3.1(d) では，T^* のうち事業者 A の属性の属性値の組合せの集合 $\{v_{A,1}, v_{A,2}\}$ は {年収 600 万未満, 年収 600 万以上} である．そのうち，結合匿名テーブル T^*(表 3.1(d)) に「年収 600 万未満」に該当するレコードは 2 名分なので $|T^*[v_{A,1}]| = 2$ となり，事業者 A のテーブル T_A(表 3.1(a)) に「年収 600 万未満」に該当するレコードは 3 名分なので $|T_A[v_{A,1}]| = 3$ となる．表 3.1(d) は 2/3-*max-site-presence* を満たす．

4.2 ダミーユーザプロトコルの提案

本節では，*δ-site-presence* を満たしつつ,「(問題 3-2) ユーザ ID 通知によるユーザ存在情報の漏洩問題」を解決するための分散匿名化のプロトコルを提案する．

問題 3-2 は，ユーザ ID を通知する際に，通知をする機関に存在するユーザ ID だけを通知することにより発生してしまう．そこで，存在しないユーザのユーザ ID も通知するために，ダミーユーザを導入する．ダミーユーザは，自機関に存在しないユーザを，あたかも存在するかのように扱うユーザのことである．なお，ダミーユーザに対して，存在するユーザを存在ユーザと呼ぶ．ダミーユーザを導入することにより，通知されるユーザ ID がダミーユーザなのか存在ユーザなのかの区別を困難にでき，問題 3-2 を解決することができる．

このようなダミーユーザを用いた提案手法は，問題 1,2,3 を満たすために以下の要件を満たしつつ，できるだけ詳細な結合匿名テーブル T^* を出力する必要がある．

(要件 1) T^* は k-匿名性を満たすこと

(要件 2) プロトコルの通信内容から，T^* から推測される以上の詳しい情報が極力漏れないこと

(要件 3) T^* は δ-*site-presence* を満たすこと

(要件 4) プロトコルの通信内容から，T^* から推測される以上の詳しいユーザ存在情報が極力漏れないこと

ここで，要件 1 と要件 2 は既存の分散匿名化の要件と同じであり，問題 1 と問題 2 の解決のための要件であたる．そして要件 3 と要件 4 は，問題 3 の解決のために追加された要件であり，それぞれ問題 3-1 と問題 3-2 の解決のための要件にあたる．

そこで，要件 1 と要件 2 だけでなく要件 3 と要件 4 も満たすために，既存の Mondrian[27] を拡張し，ダミーユーザを導入したダミーユーザプロトコルを提案する．なお，Mondrian とは，k-匿名化を行うための Top Down アプローチの匿名化アルゴリズムとして広く利用されているアルゴリズムであり，既存の [24] の分散匿名化手法でも採用されている．そして，提案するダミーユーザプロトコルでは，k-匿名化だけでなく δ-*site-presence* も満たす必要があるため，既存の Mondrian の分割点決定関数を拡張する．

ダミーユーザプロトコルは，[24] の分散匿名化プロトコルと同様に分割プロトコルと結合プロトコルで構成される (図 4.1)．まず，事業者 A,B が分割プロトコルを実行し，各事業者内で内部匿名テーブル $T_n^*(n \in \{A, B\})$ を生成する．その後，事業者 C が結合プロトコルを実行し，事業者 A,B が持つ T_n^* を単純に結合した T^* を取得する．T_n^* の分割と T^* の例を表 4.1 に示す．この例では，事業者 A は $T_A(userID,$ 年収) を，事業者 B は $T_B(userID,$ 視聴開始時刻 , 視聴番組) を保持している．そして，年収と視聴開始時刻を準識別子，視聴番組をセンシティブ属性として結合匿名テーブル T^*(年収 , 視聴開始時刻 , 視聴番組) を作成している．

以降の節では，4.2.1 節で，ダミーユーザプロトコルの分割プロトコルと結合プロトコルの詳細について説明する．4.2.2 節では，既存の Mondrian の分割点決定関数を拡張した，

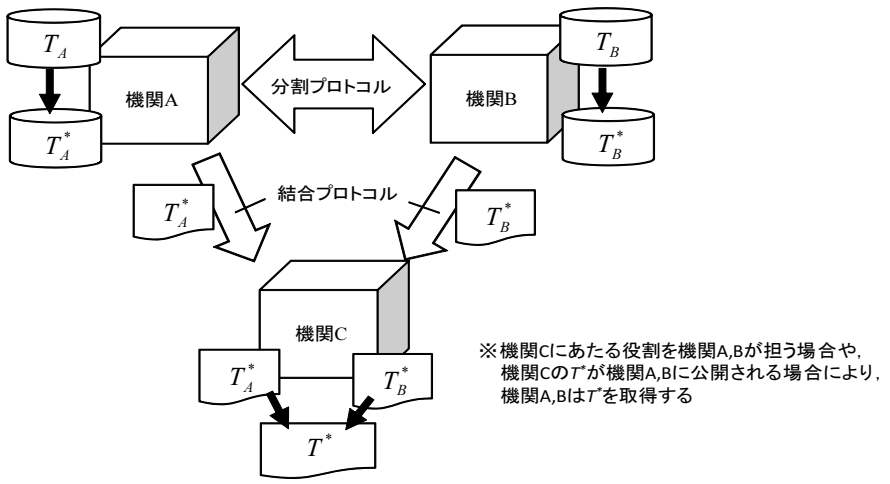


図 4.1: ダミーユーザプロトコルの分割プロトコルと結合プロトコル

ダミーユーザプロトコルの分割プロトコルの分割点決定関数について説明する．さらに，4.2.3 節ではダミーユーザプロトコルにおけるセキュア計算の利用について説明する．そして，4.2.4 節でダミーユーザの割り当て方法と母集団の要件について述べる．

4.2.1 ダミーユーザプロトコルの分割プロトコルと結合プロトコルの動作

本節では，ダミーユーザプロトコルの分割プロトコルと結合プロトコルの詳細について説明する．ダミーユーザプロトコルの分割プロトコルは，大きく 3 つの Step で動作を行う (図 4.2)．これらの分割プロトコルの各 Step の動作の詳細と，結合プロトコルの動作の詳細を説明する．

分割プロトコルの Step1:ダミーユーザの割当と T_n^* の初期化

分割プロトコルでは最初に，事業者 A と事業者 B が，自事業者のダミーユーザを割り当てる．本提案手法では，双方の事業者のユーザを包含する母集団ユーザ集合 U を事前に知っているという前提を置く．ここで U は，事業者 A に存在するユーザ集合を U_A，事業者 B に存在するユーザ集合を U_B，事業者 A,B のどちらにも存在しないユーザ集合を U_O としたとき $U = U_A \cup U_B \cup U_O$ $(U_O \neq \phi, U_A \cap U_B \neq \phi)$ となる．このような前提は，例えば事業者 A,B が Open ID[40] のような同一の認証サーバを利用している場合に成立する．この

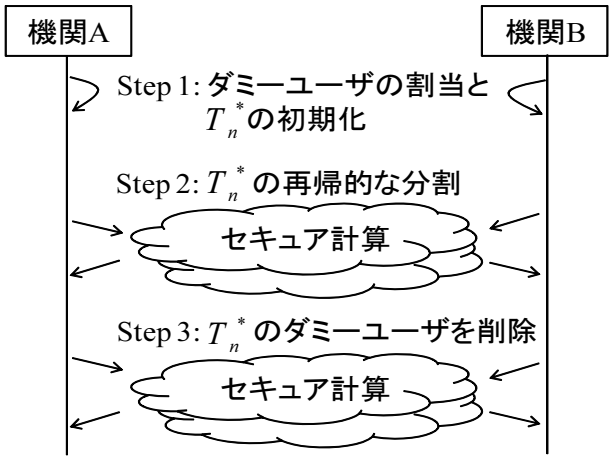


図 4.2: ダミーユーザプロトコルの分割プロトコルの概要

場合，認証サーバに存在する全ユーザが U となる．そして事業者 A と事業者 B は，事業者 A のダミーユーザを $U - U_A$，事業者 B のダミーユーザを $U - U_B$ と割り当てる．

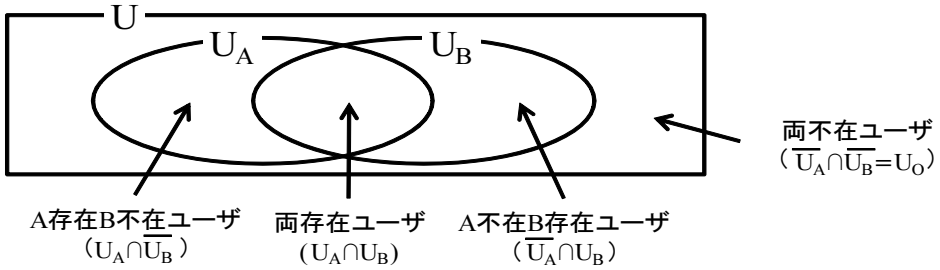


図 4.3: ダミーユーザと存在ユーザの関係

また，これらのユーザ集合の関係を図 4.3 に示す．本論文では，事業者 A,B の両方に存在するユーザを「両存在ユーザ」(つまり「共通ユーザ」)，事業者 A に存在するが事業者 B に存在しないユーザを「A 存在 B 不在ユーザ」，逆に事業者 B に存在するが事業者 A に存在しないユーザを「A 不在 B 存在ユーザ」，事業者 A,B の両方に存在しないユーザを「両不在ユーザ」と呼ぶ．この図に示したように，事業者 A のダミーユーザ $(U - U_A)$ は A 不在 B 存在ユーザ $(\overline{U_A} \cap U_B)$ と両不在ユーザ $(\overline{U_A} \cap \overline{U_B})$ の和集合となり，事業者 B のダミーユーザ $(U - U_B)$ は A 存在 B 不在ユーザ $(U_A \cap \overline{U_B})$ と両不在ユーザ $(\overline{U_A} \cap \overline{U_B})$ の和集合となる．

function split(U_p:分割対象となるユーザ集合の IDs)
1: U_p のダミーユーザのダミー値を更新
2: $d \Leftarrow$ 分割点決定関数を用いて分割点を決定 (セキュア計算を利用)
3: d で分割した際に k-匿名性と δ-*site-presence* を満たすか確認 (セキュア計算を利用)
4: **if** k-匿名性と δ-*site-presence* を満たせない **then**
5: U_p についての split 処理終了
6: **endif**
7: **if** d は自機関の T_n^* の分割点 **then**
8: T_n^* を d で分割し，分割後の IDs を相手の機関へ送信
9: **else**
10: 相手から分割後の IDs を受信し，T_n^* を分割
11: **endif**
12: $U_{hi}, U_{low} \Leftarrow$ 分割後の IDs
13: split(U_{hi}),split(U_{low}) を再帰呼出し

図 4.4: 分割プロトコルの Step2 のアルゴリズム

次に内部匿名テーブル T_n^* を初期化し，最も一般化された状態にする (表 4.1(a)(b))．各事業者は内部匿名テーブルの分割を繰り返すことで匿名化を行う．内部匿名テーブルが持つ属性は $T_A^*(userIDs, QID_A)$，$T_B^*(userIDs, QID_B, userCounts)$ である．$userIDs$ とは，T_n^* のレコードに該当するユーザ ID の集合である．QID_A と QID_B とは，事業者 A,B が持つ準識別子である．$userCounts$ とは，$userIDs$ で示されたユーザ集合におけるセンシティブ属性の各属性値の共通ユーザ数であり，分割が全て完了してから計算される．なお，T_n^* の初期化時は，ダミーユーザの準識別子 (QID_A，QID_B) の属性値は，各属性の最小値が割り当てられているとして扱われる．そして T_n^* の $userIDs$ には，ダミーユーザが含まれるように初期化が行われる．

分割プロトコルの Step2:T_n^* の再帰的な分割

続いて，機関 A,B 間で通信を行い，事業者 A の主導により T_A^* と T_B^* を分割していく分割処理を行う (図 4.4)．この分割処理は Mondrian と同様に，分割対象となるユーザ集合を分割後に，分割後のユーザ集合を次の分割対象として再帰的に処理を呼び出す．

まず，ダミーユーザの準識別子の属性値に適切な値を割り当てる．この値をダミー値と

表 4.1: 内部匿名テーブル T_A^*,T_B^* と結合匿名テーブル T^*

(a) 事業者 A の T_A^*(初期)

GID	$userID$s	年収
1	user1-15	200-499

(b) 事業者 B の T_B^*(初期)

GID	$userID$s	視聴時刻	$userCounts$
1	user1-15	17:00-20:59	-

(c) 事業者 A の T_A^*(1 回目)

GID	$userID$s	年収
2	user1-10	200-499
3	user11-15	200-499

(d) 事業者 B の T_B^*(1 回目)

GID	$userID$s	視聴時刻	$userCounts$
2	user1-10	17:00-18:59	-
3	user11-15	19:00-20:59	-

(e) 事業者 A の T_A^*(2 回目)

GID	$userID$s	年収
4	user1-5	200-399
5	user6-10	400-499
3	user11-15	200-499

(f) 事業者 B の T_B^*(2 回目)

GID	$userID$s	視聴時刻	$userCounts$
4	user1-5	17:00-18:59	X アニメ:1 Y ドラマ:1
5	user6-10	17:00-18:59	X アニメ:1 Y ドラマ:1
3	user11-15	19:00-20:59	X アニメ:1 Y ドラマ:1

(g) 最終の結合匿名テーブル T^*

年収	視聴時刻	視聴番組
200-399	17:00-18:59	X アニメ
200-399	17:00-18:59	Y ドラマ
400-499	17:00-18:59	X アニメ
400-499	17:00-18:59	Y ドラマ
200-499	19:00-20:59	X アニメ
200-499	19:00-20:59	Y ドラマ

呼ぶ．ダミーユーザは，相手事業者からみて存在ユーザなのかダミーユーザなのか区別がつかないようにする必要があるため，分割対象のユーザにおける存在ユーザ (U_A, U_B) の準識別子の属性値の分布に沿ってダミー値を割り当てる．また，ダミー値を分割毎に設定しなおしている．これにより，機関 A と機関 B の準識別子に相関があったとしても，ある程度その相関に沿ってダミー値を補正することができる．

次に，分割点決定関数を用いて分割点を決定する．この処理の詳細は 4.2.2 節で説明する．そして，決定した分割点で分割しても k-匿名性と δ-*site-presence* を満たせるかを，セキュア計算を用いて確認する．詳細は，4.2.3 節で説明する．

そして，指標を満たしている場合のみ T_A^*，T_B^* を分割する．1 回目の分割の様子を表 4.1(c)(d) に示す．この分割の分割点は「事業者 B」の「視聴開始時刻」の「19:00」である．この場合，まず分割点の準識別子を持つ事業者 B の T_B^* を分割する (表 4.1(d))．そして，分割前の $userIDs$ と，分割後の $userIDs$ を事業者 A に送信する．事業者 A は，受け取った $userIDs$ に従って T_A^* を分割する (表 4.1(c))．最後に，分割後の $userIDs$ に対して再帰的に上記の分割処理を繰り返していく．2 回目の分割の例を表 4.1(e)(f) に示す．この例は,「事業者 A」の「年収」を「400 万」で分割した例である．

分割プロトコルの Step3:T_n^* のダミーユーザの削除

全ての分割処理が完了後，セキュア計算を用いてダミーユーザを削除し，$userCounts$ を計算する．詳細は，4.2.3 節で説明する．例えば表 4.1(f) の user1-5 のレコードでは，事業者 A の存在ユーザのユーザ ID の集合と「X アニメ」を視聴した事業者 B の存在ユーザのユーザ ID の集合が入力として与えられ，積集合の個数が 1 として出力された場合である．

以上のような Step1〜3 までの分割プロトコルによって，事業者 A,B は内部匿名テーブル T_A^*，T_B^* を分割していく．

結合プロトコル：T_n^* の結合

最後に，結合匿名テーブル T^* を取得する事業者 C が，事業者 A,B から T_A^*，T_B^* を取得して結合を行う．まず，事業者 A,B は T^* の $userIDs$ を削除する．続いて，GID から分割の順番を知られないように，事業者 A が主導して GID をランダムに並び変え再度シーケ

ンシャルな番号を振り直し，GID の振り直し指示を事業者 B に送信する．そして事業者 B は，指示に従って T_B^* の GID を更新する．その後，事業者 C は $userIDs$ が削除され GID が振り直しされた T_A^*，T_B^* を受信し，GID をキーにして結合を行うことで結合匿名テーブル T^* を得る (表 4.1(g))．

4.2.2　ダミーユーザプロトコルの分割点決定関数

ダミーユーザプロトコルのための分割点決定関数を提案する．従来の Mondrian の分割点決定関数は，各属性の正規化済みの値域 (normalized range) が最大となる属性を選択し，その属性の中央値 (median) を分割点にしている．この従来の分割点決定関数を拡張し，新たに *δ-site-presence* も満たしやすい分割点が選ばれるようにする．そのためには，分割後のユーザ集合にダミーユーザが偏りなく入る分割点が選ばれると良いと考えられる．例えば表 3.1(c) では，事業者 A から見ると年収 500 万未満は user1,2，年収 500 万以上は user3,6,7,8 である．このうち事業者 B のダミーユーザは user3,8 であるため，user1,2 の 2 名ではダミーユーザは 0 名であるのに対し，user3,6,7,8 の 4 名ではダミーユーザは 2 名となっている．つまり，表 3.1(c) では，ダミーユーザは偏っている．それに対し表 3.1(d) では，年収 600 万未満は user1,2,3，年収 600 万以上は user6,7,8 である．よって，user1,2,3 の 3 名のうち 1 名がダミーユーザであり，user6,7,8 の 3 名のうち 1 名がダミーユーザとなっており，ダミーユーザは偏っていない．

そこで，ダミーユーザのエントロピー (シャノンの平均情報量) を導入する．エントロピーは，事象全体における各事象の発生確率の偏りが小さいほど大きな値になる．ダミーユーザのエントロピー (Dummy Entropy,DE) を，以下のように定義する．

$$DE(c,n) = -\sum_{U_i \in \{U_{hi}, U_{low}\}} \frac{|dummy(n,U_i)|}{|U_i|} log(\frac{|dummy(n,U_i)|}{|U_i|}) \tag{4.5}$$

ここで c は分割点候補であり，分割前のユーザ集合 U_p を上位 U_{hi} と下位 U_{low} へ分割する属性値を意味する．また，$dummy(n,U_i)$ はユーザ集合 $U_i \in \{U_{hi}, U_{low}\}$ から事業者 n のダミーユーザを抜き出したユーザ集合である．このように定義することで，分割後のユーザ集合におけるダミーユーザの偏りが小さくなる時に DE の値が大きくなる．

この DE を利用して，ダミーユーザプロトコルの分割点決定の分割点決定関数を定義する．まず，従来の Mondrian と同様に normalized range が最大となる属性を選ぶ．そして，その属性における分割点の候補となる属性値 $(x_i \in X)$ を分割点候補 c_i として，以下のように定義したスコア値 S を計算する．

$$S(c_i) = (1-\alpha)\left(\frac{-L(c_i)}{\max_{x_j \in X}(L(x_j))}\right) + \alpha\frac{1}{2}\sum_{n \in A,B}\left(\frac{DE(c_i,n)}{\max_{x_j \in X}(DE(x_j,n))}\right) \quad (4.6)$$

$$L(c_i) = \sum_{x_j \in X}|x_j - c_i| \quad (4.7)$$

ここで $\alpha(0 \leq \alpha \leq 1)$ は，DE の影響を調整するための重みであり，α が大きいほど DE の影響が大きくなる．また，L は c_i の属性の各属性値 x_i と c_i の距離の和を意味する．median とは L が最小となる点と言い換えることができるため，α=0 とした時は c_i が median の時に S が最大となり，従来の Mondrian と同様に median が分割点に決定される．スコア値 S は，L と事業者 A,B についての DE を正規化して，重み付で足した値となる．そして S を最大化させる分割点で分割を行うことで，分割後のユーザ集合における，ユーザ数に対する事業者 A,B のダミーユーザ数の割合の偏りがほぼ無くなるように分割が行われ，結果的に *δ-site-presence* を満たしつつ多くの分割が可能になることが期待される．

4.2.3 ダミーユーザプロトコルにおけるセキュア計算の利用

本節では，ダミーユーザプロトコルにおいてセキュア計算をどのように利用しているかを説明する．

セキュア計算を用いた分割点の決定方法

本節では，ダミーユーザプロトコルの分割プロトコルの Step2 における分割点決定関数の計算をにおいて，どのようにセキュア計算を用いているかについて説明する．提案する分割点決定関数は，属性値やユーザ存在情報を隠蔽したまま計算する必要があるため，3 種類のセキュア計算を用いる (図 4.5)．まず，分割点の属性を選ぶ処理で *secure comparison*[51] を用いる (図 4.5(1))．これは，機関 A,B が持つ値を秘密にしながら大小関係を求めるプロトコルである．*secure comparison* を用いて，機関 A,B がローカルで計算した最大の normalized

range を比較し，どちらが大きいかを求め，分割を行う機関 (分割機関) と行わない機関 (非分割機関) を決定する．

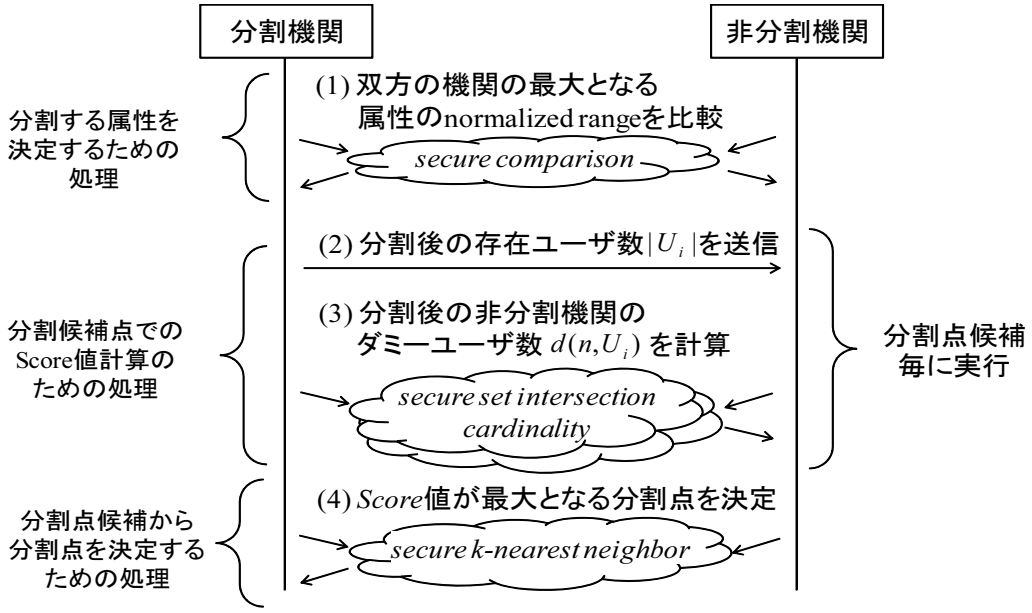


図 4.5: Step 2 の分割点決定関数の処理シーケンス

次に，機関 A,B で分割点候補 c_i の DE を計算する．ここで，非分割機関は分割後のユーザ集合を知らないので，分割後のユーザ数 ($|U_i|$) と非分割機関 n のダミーユーザ数 ($|d(n, U_i)|$) をローカルで計算できない．そこで $|U_i|$ は分割機関から取得する (図 4.5(2))．$|d(n, U_i)|$ については，*secure set intersection* の *cardinality* を用いて，分割後のユーザ集合 U_i と非分割機関 n のダミーユーザの積集合の要素数を得ることで計算する (図 4.5(3))．以上により，分割機関の c_i の分割点の属性値や分割後のユーザ集合を知ることなく，DE の計算に必要な情報を得ることができたため，DE を機関内でローカルに計算できる．

最後に，機関 A,B は *secure k-nearest neighbor*[52] というセキュア計算のプロトコルを用いて，分割点を決定する (図 4.5(4))．これは，機関 A,B がローカルで計算した正規化した DE と L について，それらを足した $S(c_i)$ が最大となる分割点候補を得る処理になる．以上のように，属性値やユーザ存在情報を相手機関に秘密にしながら分割点を決定することができる．

セキュア計算を用いた指標の確認方法

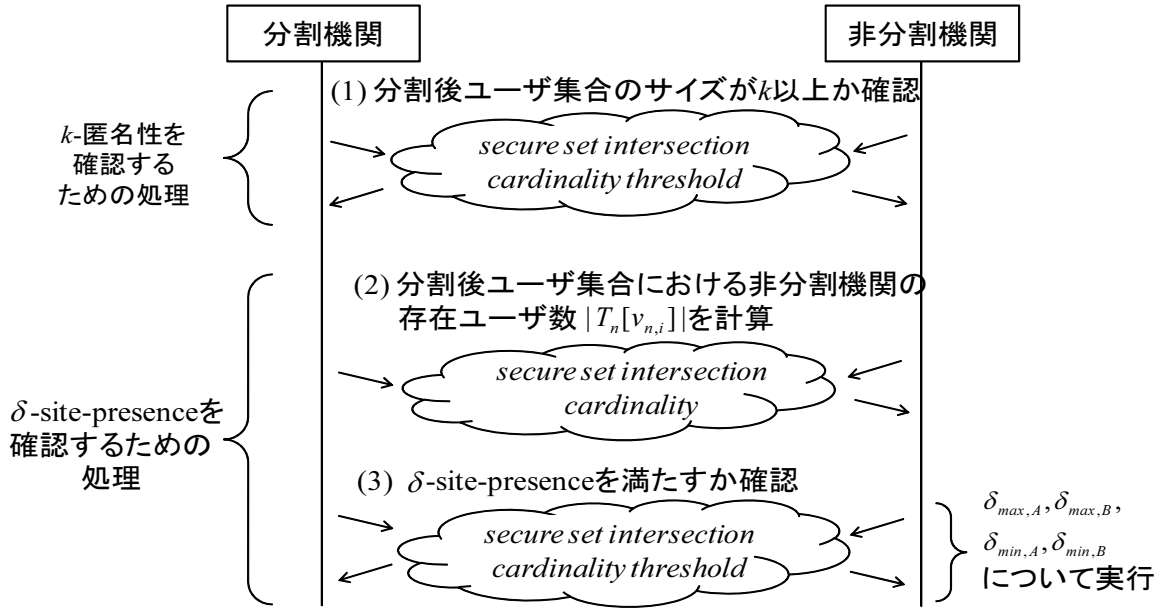


図 4.6: Step 2 の各指標確認の処理シーケンス

本節では，ダミーユーザプロトコルの分割プロトコルの Step2 における，k-匿名性と δ-*site-presence* を満たしているかの確認処理を，セキュア計算をどのように用いているかについて説明する．これらの指標を満たしているかの確認には相手機関にユーザ存在情報を知られてはいけないので，*secure set intersection*[14] というセキュア計算 [32] のプロトコルを用いる (図 4.6)．このプロトコルは，機関 A,B が持つ集合をお互いに隠蔽しながら，それらの集合の積集合や，積集合の要素数 (*cardinality*) や，積集合の要素数と指定した値との大小関係 (*cardinality threshold*) を求めることができる．分割後のグループで k-匿名性を満たしているかを確認するためには，機関 A,B は存在ユーザのユーザ ID の集合を入力として *cardinality threshold* を実行し，積集合の人数 ($|T^*[v_{n,i}]|$) が k 以上であるかを求めればよい (図 4.6(1))．δ-*site-presence* を満たしているかを確認にするには，例えば $\delta_{max,A}$ の確認の場合は，先ほどと同様に *cardinality threshold* を用いて「$|T^*[v_{A,i}]| \leq \delta_{max,A}|T_A[v_{A,i}]|$」を確認すればよい．ただし，機関 A が非分割機関であった場合は，機関 A は分割点候補の分割後のユーザ集合を知らないので $|T_A[v_{A,i}]|$ をローカルで計算できない．そこで機関 A は，機関 B の分割後グループの *ID*s と機関 A の存在ユーザの *ID*s を入力として *cardinality* を実行

し，$|T_A[v_{A,i}]|$ を得る (図 4.6(2))．そして，$\delta_{max,B}$，$\delta_{min,A}$，$\delta_{min,B}$ についても同様に計算し，*cardinality threshold* を用いて，*δ-site-presence* を満たしているかを確認にする (図 4.6(3))．

セキュア計算を用いたダミーユーザの削除方法

本節では，ダミーユーザプロトコルの分割プロトコルの Step3 におけるダミーユーザの削除の処理において，どのようにセキュア計算を用いているかについて説明する．ダミーユーザの削除の処理は，*secure set intersection* の *cardinality* を用いて，T_n^* の各レコードの SA の各属性値 s について，機関 A の存在ユーザのユーザ ID の集合，機関 B で s を持つ存在ユーザのユーザ ID の集合との積集合の個数を求めればよい．例えば表 4.1(f) の user1-5 のレコードでは，機関 A の存在ユーザのユーザ ID の集合と,「視聴番組」が「X アニメ」の機関 B の存在ユーザのユーザ ID の集合を入力として与えた結果，積集合の個数が 1 として出力された例である．

4.2.4　ダミーユーザの割り当て方法と母集団の要件

提案手法は，各機関で予め共有されている母集団からダミーユーザを割り当てるという手法を取っている．本節では，なぜこのようなダミーユーザの割り当て方法を採用しているかの理由を述べる．そして，ダミーユーザを割り当てるために必要な，母集団が満たすべき要件について説明する．

ダミーユーザの割り当て方法とその理由

提案手法では，4.2.1 節で述べたように，母集団 U から自機関のユーザ (U_A，U_B) を除いた全ユーザを各事業者のダミーユーザとして割り当てる (図 4.7)．つまり，このダミーユーザの割り当て方法では，機関 A のユーザ (U_A) と機関 A のダミーユーザの和集合と，機関 B のユーザ (U_B) と機関 B のダミーユーザの和集合は，母集団ユーザと一致することになる．これにより，ユーザ ID を相手事業者に通知する際は，母集団ユーザの全ユーザ ID が通知されることになり，ユーザ存在情報やユーザ不在情報を隠すことができる．

ここで，提案手法のように母集団に一致するようにダミーユーザを割り当てる方法ではなく，母集団からランダムにダミーユーザを選択するような方法を考えてみる．この

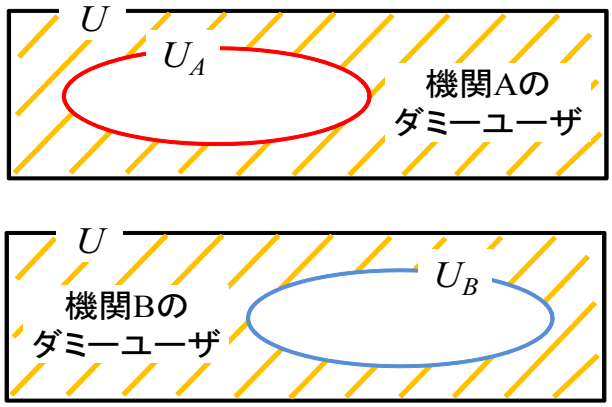


図 4.7: 機関 A,B におけるダミーユーザの割り当て方法

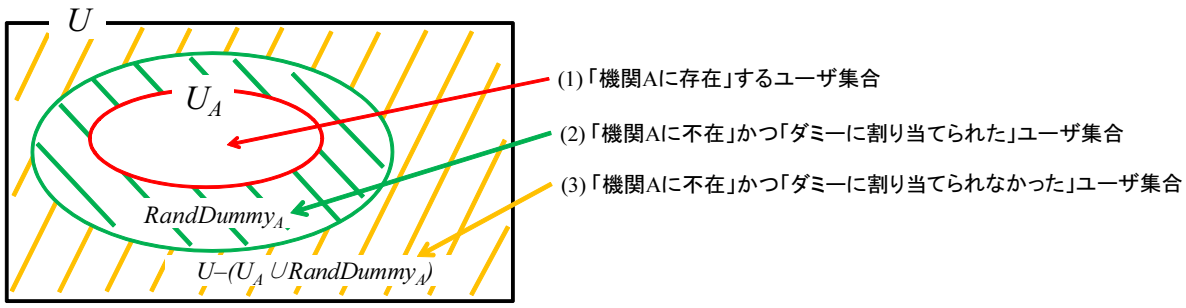


図 4.8: ランダムにダミーを割り当てる方法 (機関 A の場合)

方法では，機関 A は母集団 (U) から自事業者に存在するユーザ (U_A) を引いたユーザ集合 ($U - U_A$) から，一部をダミーユーザとしてランダムに抽出する．この抽出したダミーユーザを $RandDummy_A$ とする (図 4.8)．すると，機関 A から機関 B に分割後のグループのユーザ ID を通知する際に，$U_A \cup RandDummy_A$ のユーザ集合がのユーザ ID が通知されることになる．ここで，U_A は「機関 A に存在するユーザ」(図 4.8 の (1))，$RandDummy_A$ は「機関 A に存在しないユーザ」かつ「ダミーに割り当てられたユーザ」である (図 4.8 の (2))．よって，残りの $U - (U_A \cup RandDummy_A)$ となるユーザ集合は,「機関 A に存在しないユーザ」かつ「ダミーに割り当てられなかったユーザ」である (図 4.8 の (3))．ここで，機関 A が $U_A \cup RandDummy_A$ のユーザ集合のユーザ ID を機関 B に通知するということは，機関 B は $U - (U_A \cup RandDummy_A)$ を計算出来ることを意味する．すると，$U - (U_A \cup RandDummy_A)$ は少なくとも機関 A に存在しないユーザであるので，機関 B はユーザ不在を知ることが出来てしまう．このように，ランダムにダミーユーザを選択する方法では，相手機関にユーザ不在を知られてしまう．そこで提案手法では，母集団から自機関のユーザを除いた全ユーザをダミーユーザとして割り当てる方法を取っている．

母集団が満たすべき要件

提案手法のダミーユーザの割り当て方法では，もし母集団が適切に設定されておらずダミーユーザ数が少な過ぎる場合，ユーザ存在を知られてしまう恐れがある．そこで，以下に母集団が満たすべき要件を整理する．

まず，母集団のユーザ数の下限の要件を説明する．母集団のユーザ数があまりに少ないと，ダミーユーザ数が小さくなり，結果として通知するユーザのほとんどが存在ユーザになってしまう．例えば，機関 A に存在するユーザが 100 名であり，母集団が 120 名であった場合を考える．この場合，本手法でダミーユーザを割り当てると，機関 A のダミーユーザは 20 名となる．すると，機関 A から機関 B に通知するユーザ ID にうち，機関 A に存在するユーザの割合は，母集団 120 名のうち 100 名がダミーユーザであるので $100/120 \approx 0.83$ となる．これは，もし機関 B が機関 A のユーザ数と母集団ユーザ数を知っていたとすると，機関 B が知ることが出来てしまうユーザ存在確率は約 0.83 であることを意味する．つまり，たとえ δ-*site-presence* の $\delta_{max,A}$ を 0.7 と設定したとしても，ユーザ ID の通知から知ら

れてしまうユーザ存在確率が 0.83 であるので，ユーザ存在確率を 0.7 以下に抑えることができなくなってしまう．

このように，設定する $\delta_{max,n}$ $(n \in A, B)$ を満たすことができるような母集団のユーザ数が必要である．つまり，

$$\frac{|U_n|}{|U|} \leq \delta_{max,n} \quad (n \in A, B) \tag{4.8}$$

を満たす必要がある．母集団のユーザ数 $|U|$ は，式 4.8 を式変換した以下の関係を満たす必要がある．

$$\frac{|U_n|}{\delta_{max,n}} \leq |U| \quad (n \in A, B) \tag{4.9}$$

また，母集団ユーザ集合があまりにも大きくなってしまうと，計算量・通信量が大きくなってしまうため，上記の要件を満たしつつ小さな母集団となっていることが望ましい．なお，実際のアプリケーションを考えた際の母集団ユーザ数の上限値は 5.6 節で評価している．

提案手法では，以上のような要件を満たす母集団を機関 A,B において共有しているという前提をおいているが，これは実際のアプリケーションにおいて十分あり得るケースであると考える．例えば,「(a) 医療機関のデータ連携」の場合では，特定の健康保険組合が管理する被保険者番号を母集団とすれば良い．また,「(b) 異業種のデータ連携」では，事業者 A,B が同一の認証プロバイダを利用しているような場合がある．その場合は，事前に母集団をプロバイダから受け取ることが考えられる．このように，提案手法を実行するために必要な母集団は適切に設定することが可能であると考えられる．

4.3 提案手法を用いたアプリケーション構築フレームワーク

本節では，提案手法とデータを利用するための技術とを用いて，サービス提供に必要なデータの生成から実際のサービス提供までを含めたアプリケーション構築のためのフレームワークを説明する．このフレームワークを用いることで，様々な種類のパーソナル情報やその他の大量の情報を用いたサービスを提供することができる．

提案手法は，複数の事業者がもつプライバシに関わるデータを結合し匿名化することで，プライバシ性の低いデータに加工する技術である．実際のアプリケーションでは，この技術で生成したデータをデータマイニングなどの手法を用いて分析し，分析結果を用いてユーザにサービスを提供することになる．特に近年では，様々なセンサ情報 (RFID による物のトレース情報，GPS や加速度センサによる情報など) を取得することが可能であることから，今後は複数の事業者がもつプライバシ情報をデータ連携した情報だけでなく，様々なセンサ情報を用いたサービス提供を行うことになると考えられる．すると，分析対象となるデータが大量になり，ユーザのその時その時の状況に合わせたサービス提供が困難となってしまう恐れがある．したがって，提案手法を用いてアプリケーションを提供するためには，大量データを対象とした分析のための技術も必要となる．

そこで，大量データの分析のための技術として著者の研究 [72, 45] を利用すると良い．著者の研究 [72, 45] は，ビックデータから重要なデータを抜き出すフィルタリング技術である．この技術は，Semantic Web[8] の技術を用いて，Time(時間)，Place(場所)，Occation(状況)，Personalization(個人の好み) によって，データをソートして切り出している．この技術により，大量のデータからユーザに必要な必要な情報だけを抜き出しつつユーザのコンテキスト (状況) に応じた適切なサービスを提供することができる．このように，この技術と提案技術を組み合わせることにより，本論文で提案した手法を用いて生成された大量のデータ (ビックデータ [48]) をもちいて，個々のユーザに適したサービスを提供するアプリケーション構築フレームワークを実現することができる．

図 4.9 は，このフレームワークの全体を示した図である．まず，ユーザに関する情報はサービス事業者に蓄積される (図 4.9 の 1)．この情報は，プライバシに関わる情報であるため，本論文で提案した技術 (図 4.9 の (a)) を用いて，安全なデータに加工する (図 4.9 の 2)．そして，著者の研究 [72, 45] の技術 (図 4.9 の (b)) を用いて，安全に加工されたデータを含むビックデータを用いたサービス提供を行う (図 4.9 の 3)．このように，本論文で提案した技術と，著者の研究 [72, 45] の技術を組み合わせることで，アプリケーション構築のフレームワークを実現することができ．様々な情報を大量に用いた新たなサービスを提供することが期待できる．

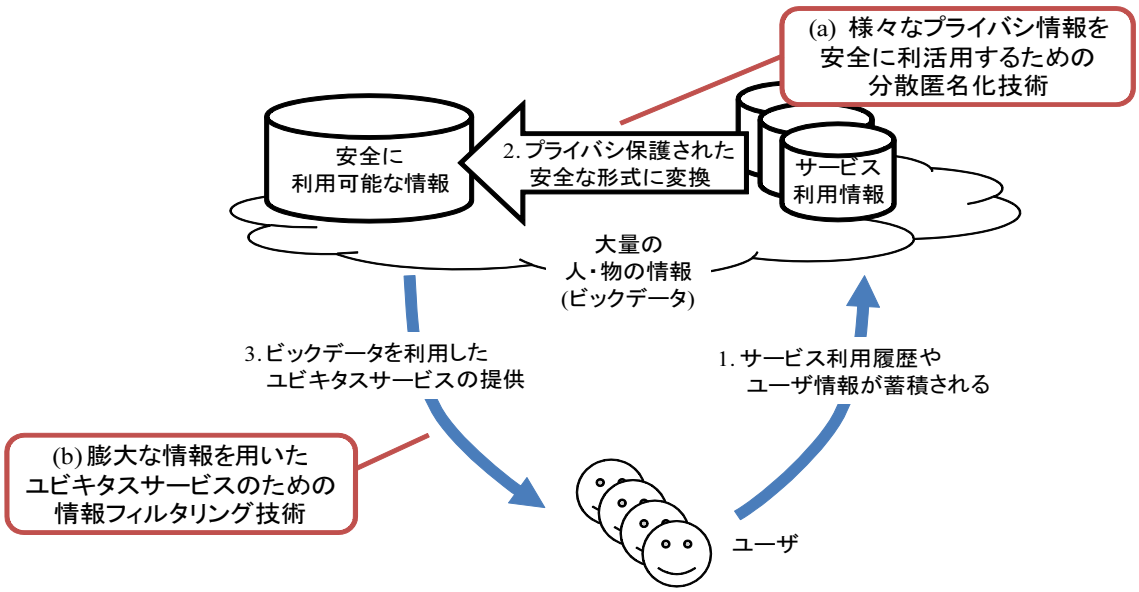


図 4.9: 提案手法を用いたアプリケーション構築フレームワーク

第5章　評価実験

本章では，提案プロトコルを実装し，評価を行った結果を示す．まず 5.1 節で，評価に用いたデータセットについて説明する．続いて 5.2 節では，評価で用いる評価指標を説明する．そして 5.3 節で評価内容を説明し，5.4〜5.7 節でそれらの評価内容の結果を示す．最後に，5.8 節で評価結果をまとめ，考察する．

提案プロトコルの基本的な実装は Java で行い，いくつかのセキュア計算のためのライブラリを呼び出すように実装している．表 5.1 に本実装に用いたライブラリの詳細を示す．なお，本実装はシングルスレッドで動作するような実装になっているため，マルチスレッドで動作するように改良し，マルチコア環境で動作させることで，さらなる速度改善が見込める．

表 5.1: 利用してるセキュア計算のライブラリ

セキュア計算	ライブラリ
secure set intersection	野島らによる *secure set intersection*[14] の C++実装 [71] [1]
secure k-nearest neighbor	Kun Liu による Paillier 暗号 [41] の Java 実装 [33] [2] と Apache Thrift 0.6.1 [13] [3]を利用して実装
secure comparison	Fairplay 1.0[35](MPC の Java 実装) [4]

そして，実装したプログラムを表 5.2 に示した実行環境で動作させた．ただし，匿名化結果データの有効性評価などの速度計測以外の評価をする際は，セキュア計算を用いて通信する必要が無いので，仮想的にセキュア計算を実行するように動作させて評価を行った．

[1] `http://fnp.sourceforge.net/` からダウンロード可能
[2] `http://www.csee.umbc.edu/~kunliu1/research/Paillier.html` からダウンロード可能
[3] `http://thrift.apache.org/` からダウンロード可能
[4] `http://www.cs.huji.ac.il/project/Fairplay/` からダウンロード可能

表 5.2: 評価環境

項目	マシン 1	マシン 2
CPU	Intel Xeon E5645@2.4GHz	Intel Xeon E5420@2.5GHz
Memory	8Gbytes	
Network	Gigabit Ethernet	
OS	CentOS 6.0	
言語	Java 1.6.0_24	

5.1　評価データ

本節では，評価に用いたデータについて説明する．評価では，以下の 2 種類のデータを用いた．

レセプトデータ　日本のレセプトデータ (診療報酬明細書)

国勢調査データ　米国の国勢調査データ

以降の節でこれらのデータの詳細を説明する．

5.1.1　レセプトデータ

評価実験で用いたレセプトデータは，JMDC(株式会社日本医療データセンター)[5]が提供している実際のレセプトデータの一部である．JMDC では，いくつかの特定の健康保険組合に加入している約 10 万人の糖尿病患者の糖尿病以外の過去の疾病を含むレセプトデータを提供している．このデータは，個人の特定はできないが複数の医療機関での個人データの結合はできるように，氏名や地域に関する情報は別コードに置き換えられている．評価では，このデータから異なる診療科の医療機関のうち共通の患者数が一番多い医療機関を，それぞれ機関 A，機関 B として抽出し，評価用のレセプトデータとした．

抽出したレセプトデータにおける機関 A は約 300 人患者 (U_A) のデータを持つ耳鼻科の病院，機関 B は約 3500 人の患者 (U_B) のデータを持つ内科の病院である[6]．そして，これら

[5] http://www.jmdc.co.jp/

[6] 本評価で用いているデータは実際のレセプトデータであることから，病院の特定が困難になるように各病院の人数はおおよその値として表記する

機関の共通の患者 ($U_A \cap U_B$) は約 230 人である．つまり，評価データは内科と耳鼻科のレセプトデータである．評価では，これらの患者とは別に，機関 A,B に通院していない患者 (U_O) を約 1430 人抜き出し，母集団の患者 (U) を 5000 人とした (図 5.1).

そして，2 つの機関で患者の疾病履歴を結合して病気の相関を調べるというユースケースを想定し，以下の形式の機関 A,B のテーブル (T_A, T_B) を生成した.

$$T_A(ID, \text{病名 } A1, \text{病名 } A2),\quad T_B(ID, \text{病名 } B1, \text{病名 } B2, \text{分類})$$

ここで ID は患者の識別子 (機関 A と機関 B で共通),「病名 A1」と「病名 A2」は機関 A における直近に診療した 2 件の疾病の疾病コードを数値に変換した値である．機関 B の「病名 B1」と「病名 B2」も同様である．また,「分類」はガンなどの疾病の進行を想定しており，今回の評価結果に影響が無いため疑似的に {「I」,「II」,「III」,「IV」} のいずれかの値をランダムに生成した．なお，結合テーブルの形式は

$$T^*(\text{病名 } A1, \text{病名 } A2, \text{病名 } B1, \text{病名 } B2, \text{分類})$$

である．ここで {「病名 A1」,「病名 A2」,「病名 B1」,「病名 B2」} が準識別子,「分類」がセンシティブ属性である.

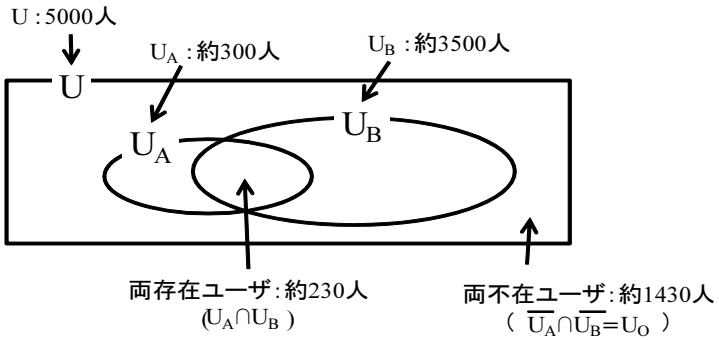


図 5.1: レセプトデータのユーザ数

5.1.2 国勢調査データ

2 つ目の評価データとして，[37] の分散匿名化プロトコルの評価と同様に，米国の国勢調査データをもとにして作成された UCI(University of California, Irvine) が提供している

Adult データ [9] を 2 事業者に分割したデータを用いた．Adult データは，14 種類の属性と 1 種類の年収分類 (class)($50K 以上 or 未満) を持つ約 3 万レコードのデータである． なお，Adult データには最終学歴や 1 週間の労働時間などある程度偏りがあるような属性も含まれている．そして，14 種類の属性を準識別子，年収分類をセンシティブ属性とし，約 3 万のレコードをランダムに並び変えた上位レコードから事業者 A と事業者 B に存在するユーザ (両存在ユーザ，$U_A \cap U_B$)，事業者 A と事業者 B の片方だけに存在するユーザ ((A 存在 B 不在ユーザ，$U_A \cap \overline{U_B}$)，(A 不在 B 存在ユーザ，$\overline{U_A} \cap U_B$))，最後に双方に存在しないユーザ (両不在ユーザ，$\overline{U_A} \cap \overline{U_B} = U_O$) を選択した．つまり，事業者 A の存在ユーザは両存在ユーザと A 存在 B 不在ユーザとなり，これらのユーザのレコードが T_A に格納される．同様に事業者 B の存在ユーザは両存在ユーザと A 不在 B 存在ユーザとなり，これらのユーザのレコードが T_B に格納される．また，ダミーユーザは 4.2.1 節で説明した方法で割り当てられ，ダミー値は各事業者の存在ユーザの準識別子の分布に沿って割り当てられる．

特に断りが無い限り本評価では両存在ユーザ数 1200 名，A 存在 B 不在ユーザ数 1200 名，A 不在 B 存在ユーザ数 1200 名，両不在ユーザ数 1200 名として評価を行った (図 5.2)．母集団数は 4800 名である．そして，データ生成を含めて 5 回計測を行い，評価値はその平均とした．なお，Mondrian アルゴリズムの研究 [27] と同様にカテゴリ値は数値として扱った．

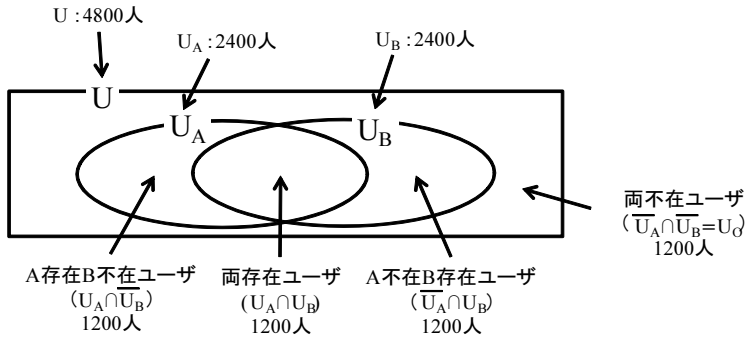


図 5.2: 国勢調査データのユーザ数

5.2　評価指標

続いて，評価の指標について説明する．本評価実験では以下の 2 つの評価指標を用いた．

- ユーザ数カウントのクエリ結果の誤差
- Discernibility Metric[6]

これらの指標は，両方とも匿名化による精度の低下を表しているが，それぞれ計測している観点が異なる．まず「ユーザ数カウントのクエリ結果の誤差」は，データマイニングの結果の精度を計測するという観点の指標である．それに対し,「Discernibility Metric」(DM) はデータそのもの精度を計測するという観点での指標である．

既存のユーザ存在情報を隠蔽手法 [39] や提案手法のベースとなっている Mondrian アルゴリズム [27] では，DM を用いて評価をしているが，実際のデータマイニング等での有用性を評価する場合は，DM を用いた評価指標は直感的に何を表しているかを判断することが難しいと考える．そこで，本研究ではユーザ数カウントのクエリ結果の誤差を計測する評価指標を主な評価指標として用い，DM は既存手法との比較のための参考として利用する．なお，参考にしている分散匿名化の既存技術 [24] や既存の匿名化の研究 [50] においても，ユーザ数カウントのクエリ結果の誤差を利用している．以降の節では，これらの指標について詳しく説明する．

5.2.1 ユーザ数カウントのクエリ結果の誤差

「ユーザ数カウントのクエリ結果の誤差」の評価指標は，結合匿名テーブル (T^*) に対してデータマイニングを行った場合に，マイニング結果にどの程度の誤差が発生するのかという観点の指標である．この指標は，既存の匿名化の研究 [50] と同様に，ある条件に合致するユーザ数をカウントするクエリ (“select count(*) from T^* where 条件部”) の結果の相対誤差 (*relative error*) を計測することで求める．なお，このようなユーザ数をカウントするクエリはデータマイニングにおける基本的な集約クエリ (*aggregate query*) とされている．

この評価手法では，まずカウントされるユーザ数の割合の期待値 (*expected selectivity*) を $\theta(0\% < \theta < 100\%)$ とおいて，条件部に指定する検索範囲が全体の θ 倍になるようなクエリをランダムに生成する．つまり，T^* に含まれるユーザ数が 1200 であった場合，θ=10%としたクエリで検索されるユーザ数 (レコード数) は約 120 となる．そして，生成したクエリを用いて，匿名化前の結合テーブル T_{AB}(T_A と T_B を単純に内部結合したテーブル) に対し

て得られたユーザ数を act，結合匿名テーブル T^* に対して得られたユーザ数を est とし，その相対誤差を $|act - est|/act$ で計算する．なお，est はクエリの条件部に記載された範囲と，汎化された値の重なり度合いに応じて算出する．例えば T^* に「20〜29 才」というレコードが 5 個であり，クエリが「20〜21 才」であった場合は，このクエリは「20〜29 才」の 20%が重なっているので，$est = 5 \times 20\% = 1$ となる．なお，条件部に利用する属性は 2 つとし，ランダムに選択した．また各評価値は，ランダムなクエリを 10000 回生成し相対誤差を計測した値の平均である．

5.2.2 Discernibility Metric

続いて，Discernibility Metric(DM)[6] という指標を説明する．この指標は，既存の集中型の匿名化手法 [39] や提案手法のベースとなる Mondrian アルゴリズム [27] で用いられている評価指標であり，この指標は先ほどの相対誤差と同様に匿名化による精度の低下を表す指標である．つまり，DM が小さいほど良い匿名化データであると言える．DM の値は，$qids$ を T^* における準識別子の属性値の集合とおくと，以下の式で計算される[7]．

$$DM = \sum_{q_i \in qids} |T^*[q_i]|^2 \tag{5.1}$$

例えば，1200 人のデータがきれいに 8 名[8]ごとに 150 個に分割されている場合は $8^2 \times 150 = 9600$ となる．また，DM 値は人数を 2 乗しているため分割に偏りが有ると急激に悪い値となる．

5.3 評価内容

本評価実験の評価内容について説明する．評価は以下の 4 つの観点で行った．

1. 有効性の評価

[7][6] ではレコード削除 (suppression) をした際の DM 値も定義されているが，提案手法ではレコード削除は行わないので無視している．

[8]データマイニングでは大まかな傾向がわかれば良いため，8 名程度の分割であってもマイニング結果に影響が少ないと考えられる．

提案手法と既存手法を実行し，提案手法が既存手法と比べてどの程度有効であるかの評価を行う．そして，レセプトデータや国勢調査データのような実際のデータにおいて有効である事を示す.

2. ユーザ存在情報の隠蔽の限界の評価

提案手法においてどの程度までユーザ存在情報の隠蔽が可能であるかの評価を行う．そして，レセプトデータを用いた実際の分析例において，意義のあるユーザ存在情報の隠蔽が可能である事を示す.

3. 対応可能ユーザ数の限界の評価

提案手法においてどの程度のユーザ数まで対応可能であるかの評価を行う．そして，10000 人以下のユーザ数におけるデータ連携において，現実的な時間で処理可能であることを示す.

4. 分割におけるダミーユーザの偏りの評価

提案した分割点決定関数によってダミーユーザが偏りなく分割されているかを評価する．そして，分割点決定関数が意図したとおりに動作していることを示す.

以降の 5.4〜5.7 節でこれらの評価結果を示し，最後に 5.8 節で評価結果を考察する.

5.4 有効性の評価

本節では，提案手法と既存手法を実行し，提案手法が既存手法と比べてどの程度有効であるかの評価を行う．まず 5.4.1 節で，重み α の適切な設定値を調べ，評価で用いる重み α の値を決定する．なお，重み α とは，4.2.2 節で提案した分割点決定関数で用いているダミーユーザのエントロピー (DE) の影響を調整するためのパラメータである．続いて 5.4.2 節で，既存の分散匿名化手法と比較した結果を示す．さらに，5.4.3 節で，既存の集中型のユーザ存在情報の漏洩軽減手法と比較した結果を示す.

5.4.1　重み α の適切な設定の評価

分割点決定関数の重み α の最適値を調べるために，α を変化させて評価を行った．評価は，レセプトデータと国勢調査データの両方のデータに対して行い，それぞれのデータにおける重み α の影響を調べた．

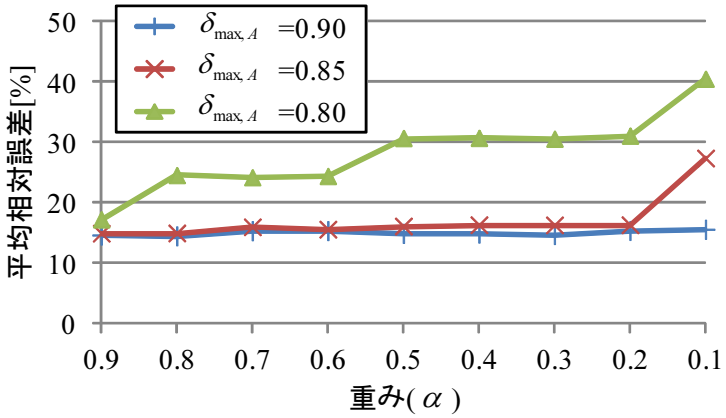


図 5.3: 重み α の影響の評価 (レセプトデータ)

まず，レセプトデータに対して評価した結果を図 5.3 に示す．この評価は，$\delta_{max,A}$ を $\{0.90, 0.85, 0.80\}$，α を $\{0.9, \cdots, 0.1\}$ に変化させて，θ=3%として相対誤差を計測した結果である．なお，$\delta_{max,A}$ の値による相対誤差の変化を見やすくするために，$\delta_{max,A}$ 以外の δ の設定値は $\delta_{max,B} = 0.99, \delta_{min,A} = 0.01, \delta_{min,B} = 0.01$ とユーザ存在/不在情報を隠蔽するための設定を緩く (δ_{max} を大きく，δ_{min} を小さく) 設定している．

この結果が示すように，$\delta_{max,A}$ を小さく設定した場合 ($\delta_{max,A}$=0.80) は α の重みが重要になり，α が小さいほど相対誤差が小さくなる傾向がある．これは，レセプトデータでは α が限界値に近い場合，ダミーユーザのわずかな偏りで δ-*site-presence* を満たさなくなるためだと考えられる．そのため，DE の影響が大きくなるように設定したほうが相対誤差が小さくなる．

続いて，国勢調査データに対して評価した結果を図 5.4 に示す．この評価では $\delta_{max,A}$ を $\{0.75, 0.70, 0.65\}$ とおいて評価している．この結果が示す通り，国勢調査データでは重み α の値による相対誤差の変化は 1〜2%程度と小さく，重み α の影響は小さいことが解る．これは，国勢調査データはレセプトデータほどダミーユーザの偏りによる影響が小さいこと

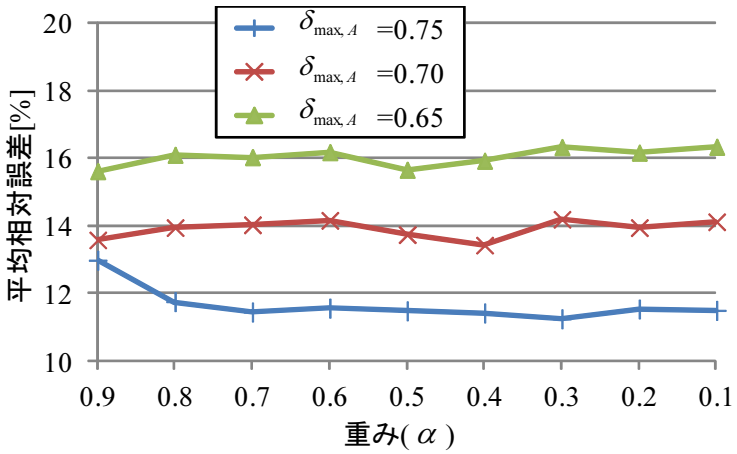


図 5.4: 重み α の影響の評価 (国勢調査データ)

を意味する.

以上の評価の結果，レセプトデータの場合は重み α は 0.90 に設定すると良い事が解り，国勢調査データの場合は重み α の値による影響が小さいことが分かった．このことから，データによって α がデータの有効性に与える影響は異なるが，α を 0.9 付近の大きい値に設定すると良いことが分かった．以降の評価では，$\alpha = 0.9$ として設定し，計測を行う.

5.4.2 既存の分散匿名化手法との比較評価

続いて，提案手法の有効性を評価するために，既存手法となる Mondrian を単純に分散環境に対応させた分散対応 Mondrian との比較を行う．この分散対応 Mondrian は，提案手法と比較するために k-匿名性だけでなく δ-*site-presence* も満たしている際に分割を行い，最終結果では共通ユーザだけを出力する分散匿名化手法であり，ベースラインとなる手法である.

図 5.5 と図 5.6 に，レセプトデータと国勢調査データにおいて k=2，$\theta = \{3\%, 5\%, 10\%, 20\%\}$ として平均相対誤差を計測した結果を示す．なお，既存手法との差を明確にするために，各 δ の値は緩く設定し，$\delta_{max,A} = 0.99$，$\delta_{max,B} = 0.99$，$\delta_{min,A} = 0.01$，$\delta_{min,B} = 0.01$ とした.

まず，図 5.5 のレセプトデータにおける評価結果について説明すると，θ=3%の時の既存手法の相対誤差は約 40%であるのに対し，提案手法の相対誤差は約 15%であり，相対誤差

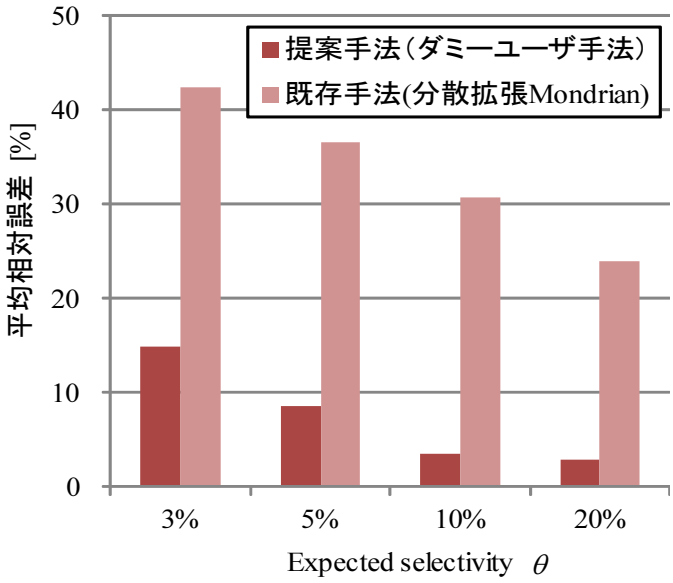


図 5.5: 既存の分散匿名化手法との比較評価 (レセプトデータ)

が約 25 ポイント小さくなっている．$\theta = \{5\%, 10\%, 20\%\}$ についても，同様に提案手法のほうが約 25 ポイントほど相対誤差が小さくなっている．これは，ダミーユーザのエントロピー (DE) の追加や分割後のダミー値の更新により，ユーザ存在情報が隠蔽できるような分割点が選ばれるようになり，その結果分割回数が増え，より詳細な情報の開示が可能になったためである．なお，θ が小さいほど相対誤差が大きくなる傾向があるが，これは匿名化を行うことにより値が汎化され曖昧な値になってしまうので，θ を小さくしてレコードを選択する際のクエリ条件の範囲を狭くすると，正しいレコードを選択しづらくなるためである．

続いて，図 5.6 の国勢調査データにおける評価結果について説明する．こちらでは，θ=3% の時の既存手法の相対誤差は約 70% であるのに対し，提案手法の相対誤差は約 20% であり，相対誤差が約 50%小さくなっている．$\theta = \{5\%, 10\%, 20\%\}$ についても同様である．

以上の既存の分散匿名化との比較評価の結果から，提案手法は既存手法よりも相対誤差を約 25%〜50%を低下させることができ，有効であることが分かった．

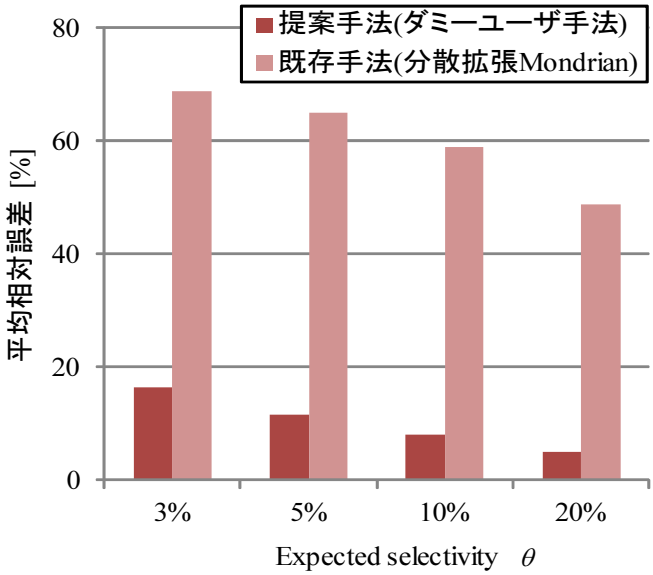


図 5.6: 既存の分散匿名化手法との比較評価 (国勢調査データ)

A:急性気管支炎 ⇒B:急性副鼻腔炎 [sup=3.1%,conf=71.4%]
A:急性気管支炎 ⇒B:アレルギー性鼻炎 [sup=3.1%,conf=57.1%]
A:急性上気道炎 ⇒B:急性咽頭喉頭炎 [sup=2.2%,conf=60.0%]

図 5.7: 機関 A(内科) と機関 B(耳鼻科) の疾病の相関ルール

θ と相対誤差のデータマイニング結果における意味

ここで，相対誤差や θ の値が，実際のデータマイニング結果においてどのような意味を持つのかについて考察を行う．例えば，レセプトデータの評価結果 (図 5.5) において提案手法の相対誤差が $\theta = 3\%$ のときに約 15%であった．この相対誤差は，例えば相関ルールマイニングを行った際に得られる相関ルールにおいて，支持度 (support) や確信度 (confidence) が約 3%であった時に，その値の相対誤差が約 15%程度発生することを意味している．

図 5.7 は，匿名化前の機関 A と機関 B のテーブル (T_A，T_B) を内部結合した結合テーブル T_{AB} に対して相関ルールマイニングを行い，支持度が 2%以上，確信度が 50%以上となる疾病についての相関ルールを，支持度が高い順に出力した結果である[9]．この結果に示したように，支持度 (support) が 3.1%と 2.2%の相関ルールが得られている．ここで，$\theta = 3\%$

[9]図 5.7 に示した疾病は，鼻や咽喉頭の炎症が気道や気管支に到達した際に起こる合併症としてよく知られている．

において約 15%の相対誤差があるということは，匿名結合テーブル (T^*) に対して相関ルールマイニングを行った場合は，これらの相関ルールの支持度に 15%の誤差が入ることになるので，3.1%と 2.2%の相関ルールの支持度は約 2.6〜3.6%と約 1.9〜2.5%になる．

レセプトデータの例におけるこの誤差は，図 5.7 の相関ルールの支持度の大小関係が逆転するようなことは少ない程度の誤差である．よって，少なくともレセプトデータの例においては，得られた相関ルールに大きな差は無いと考えることができる．

5.4.3　既存の集中型の手法との比較評価

次に，集中型 (非分散環境) の匿名化におけるユーザ存在情報の隠蔽手法である δ-*presence* を満たすための MPALM アルゴリズム [39] と比較し，分散型 (分散環境の分散匿名化) に対応した提案手法の有用性が集中型とほぼ同等であることを示す．集中型での既存手法は，あるテーブルと匿名テーブルにおけるユーザ存在情報を隠蔽する手法であり，提案手法のように機関 A と機関 B の双方からみた，ユーザ存在情報の推測を防ぐというものではない．そこで，公平な評価を行うために機関 B 側から見た $\delta_{min,B}$ と $\delta_{max,B}$ を設定せずに評価を行った．

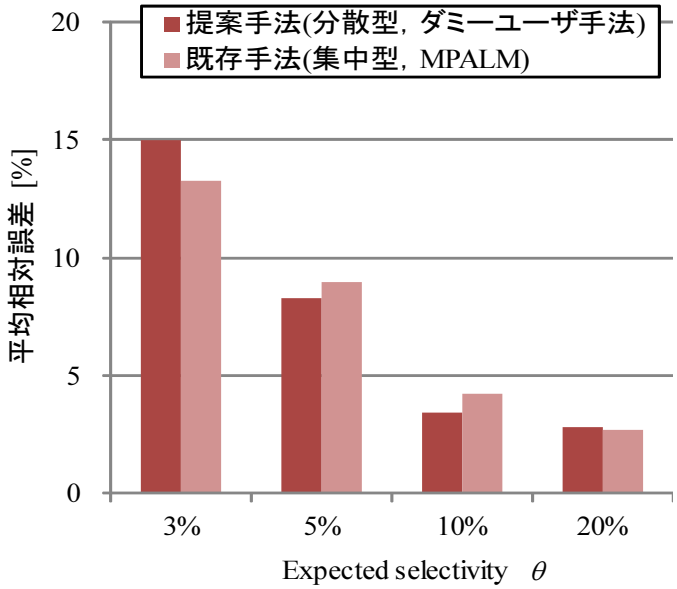


図 5.8: 集中型匿名化のユーザ存在情報の隠蔽手法との比較 (レセプトデータ)

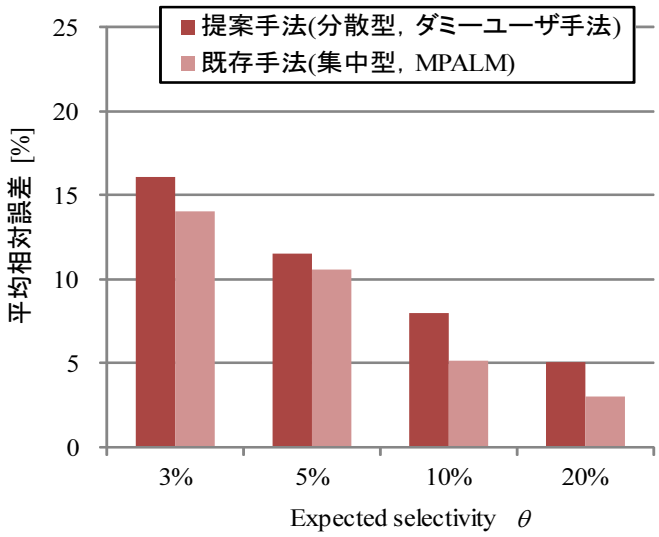


図 5.9: 集中型匿名化のユーザ存在情報の隠蔽手法との比較 (国勢調査データ)

図 5.8 と図 5.9 に，レセプトデータと国勢調査データに対して，提案手法と既存手法の平均相対誤差の値を計測した結果を示す．なお，その他のパラメータは 5.4.2 節と同じにした．まず図 5.8 のレセプトデータの評価結果について説明する．この結果では，θ が 3%と 20%の時は提案手法のほうが既存手法よりも 1〜2 ポイントほど誤差が大きく，悪い結果となっている．しかし，数ポイント程度の差はデータ分析に与える影響は小さいと考える．

また，この結果ではそ θ が 5%と 10%の時は提案手法のほうが既存手法よりも数%ほど誤差が小さく，良い結果となっている．このような θ の値によって提案手法と既存手法の相対誤差の善し悪しが逆転する現象は，既存手法の分割点を探索するアルゴリズムに原因があると考える．集中型の既存手法では，分割候補に対してユーザ存在情報を隠蔽可能であるかを順番に確認し，最初に隠蔽可能であることが見つかった分割候補で分割を行うというアルゴリズムとなっている．このアルゴリズムは，最終的な分割の回数が増えるというメリットがあるが，分割点が端に偏る傾向がある．一般に，分割の回数が増えることはデータの精度が向上することを意味するので，良い評価結果になりそうであるが，本評価で用いている相対誤差の計測方法のように，データの全体から一部を抜き出してカウントを取るような場合には良い評価結果になるとは限らない．そのため，結果的に提案手法のように Mondrian の分割点決定関数を拡張して分割点を探索するアルゴリズムの方が良い

評価結果になる場合もある．

次に，図5.9の国勢調査データについて説明する．この結果では，θが{3%，5%，10%，20%}のいづれの値であっても，提案手法のほうが既存手法よりも数%ほど誤差が大きく，悪い結果となっている．しかし，やはり2〜3ポイントほどの差であるためデータ分析に与える影響は小さいと考える．

このように，θの値によって多少の相対誤差の善し悪しはあるものの，レセプトデータと国勢調査データでの評価結果では，既存手法と提案手法との相対誤差には大きな差は無い．この結果から，提案手法は集中型の既存手法の匿名化結果と大きな差がなく，集中型の既存手法と同等の有効な匿名化が行えることがわかった．

DMを用いた既存の集中型との比較

表5.3: DMを用いた既存の集中型との比較

評価データ	提案手法 (分散型)	既存手法 (集中型)
レセプトデータ	1535	1435
国勢調査データ	10508	5512

また，参考に評価指標としてDiscernibility Metric(DM)を用いた場合の評価結果を，表5.3に示す．この結果から分かるように，提案手法は集中型の既存手法と比べてDM値が大きく，悪い結果となっていることが解る．

先ほどの相対誤差の指標を用いたレセプトデータの評価結果 (図5.8) では，θが5%と10%の時には提案手法の方が相対誤差が小さく提案手法の方が良い結果となっていたが，DM値で比較すると提案手法の方が悪い結果となる．これは，DMは分割の回数が多い場合に良い結果となりやすい評価指標となっているためである．しかし，実際のデータマイニング等での有用性を評価する場合は，DMを用いた評価指標は直感的に何を表しているかを判断することが難しいと考えられるため，本研究ではレコード数をカウントする際の相対誤差を計測する評価指標を主な評価指標として用いている．

5.5 ユーザ存在情報の隠蔽の限界値の評価

続いて，*δ-site-presence* の $\delta_{min,A}$ と $\delta_{max,A}$ と $\delta_{min,B}$ と $\delta_{max,B}$ の設定を変化させた際の相対誤差を計測し，これらの値を設定できる限界がどの程度であるかを評価した．まず，5.5.1 節で評価結果を示する．その後 5.5.2 節において評価結果をまとめ，5.5.3 節で実際のアプリケーションにおける意義を考察する．

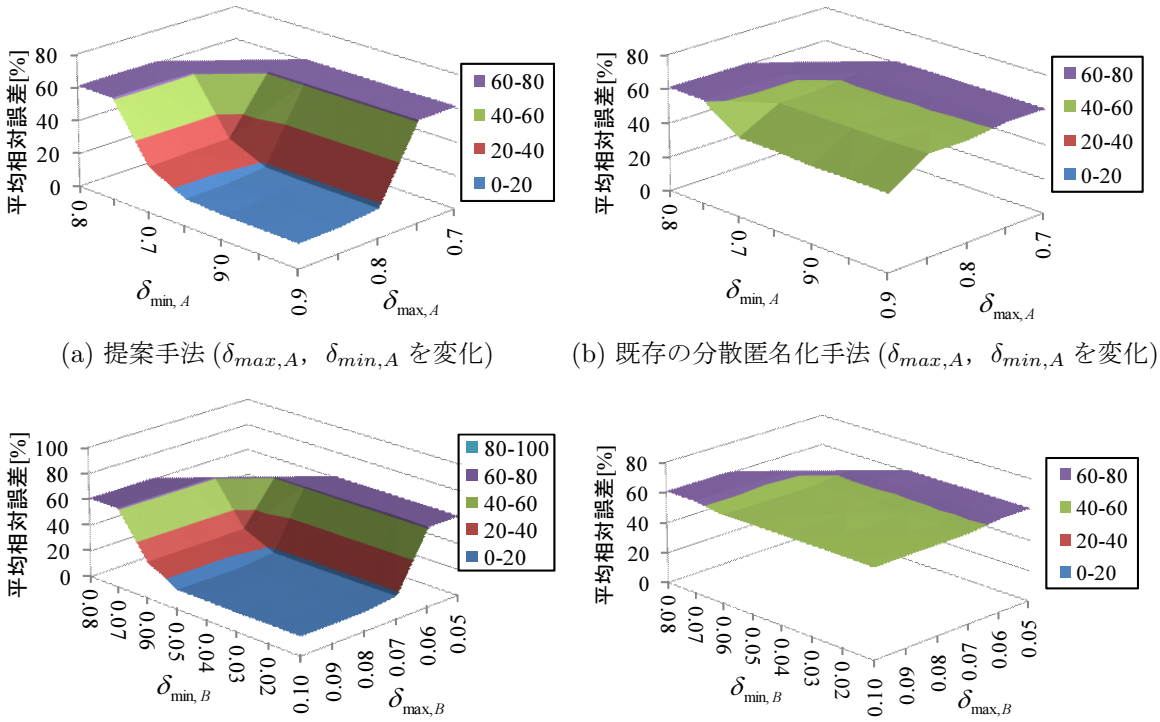
5.5.1 評価結果

レセプトデータにおける評価結果

まず，図 5.10 にレセプトデータについて評価を行った結果を示す．この評価では，θ=3%，$\delta_{max,A} = \{0.9, \cdots, 0.7\}$，$\delta_{min,A} = \{0.8, \cdots, 0.5\}$，$\delta_{max,B} = \{0.10, \cdots, 0.05\}$，$\delta_{min,B} = \{0.01, \cdots, 0.08\}$ と設定し，5.2.1 節で説明したユーザ数カウントクエリ結果の平均相対誤差を計測している．図 5.10 は (a)〜(c) の 4 つのグラフがあるが，(a) と (b) は $\delta_{max,A}$ と $\delta_{min,A}$ を変化させたグラフであり，(a) は提案手法を用いた場合の計測結果，(b) は分散対応 Mondrian を用いた場合の計測結果である．そして，(c) と (d) は $\delta_{max,B}$ と $\delta_{min,B}$ を変化させた際の同様の計測結果である．また，$\delta_{max,A}$ と $\delta_{min,A}$ を変化させているグラフでは，$\delta_{max,A}$ と $\delta_{min,A}$ の影響が明確になるように，$\delta_{max,B}$ と $\delta_{min,B}$ は最も緩い設定である $\delta_{max,B}$=0.99，$\delta_{min,B}$=0.01 としている．$\delta_{max,B}$ と $\delta_{min,B}$ を変化させているグラフも同様である．なお，これらの 3 次元グラフの横軸 $\delta_{min,A}$ や $\delta_{max,A}$ などは，手前よりも奥の値のほうがよりユーザ存在情報やユーザ不在情報を隠蔽する厳しい設定となっている．

まず，$\delta_{max,A}$ と $\delta_{min,A}$ を変化させて提案手法を評価した結果である図 5.10(a) について見てみる．この結果から解るように，左側の横軸の $\delta_{min,A}$ が 0.65〜0.5，右側の横軸の $\delta_{max,A}$ が 0.9〜0.8 の範囲では縦軸の平均相対誤差が 20%以下と小さくなっている．そして，$\delta_{min,A}$ が 0.65 よりも大きい値に設定した場合や，$\delta_{max,A}$ が 0.8 よりも小さい値に設定した場合あたりからは急激に結果が悪化し最終的には平均相対誤差は約 60%まで上昇している．それに対して，図 5.10(b) の分散拡張 Mondrian を用いた結果では，たとえ $\delta_{max,A}$ と $\delta_{min,A}$ を緩く設定したとしても平均相対誤差は，40%以上もある．

このような傾向は，図 5.10(c) や図 5.10(d) の $\delta_{max,B}$ と $\delta_{min,B}$ を変化させた際の結果で



(a) 提案手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)　(b) 既存の分散匿名化手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)

(c) 提案手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)　(d) 既存の分散匿名化手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)

図 5.10: δ を変化させた際の提案手法と既存手法の相対誤差 (レセプトデータ)

も言える．この結果では左側の横軸の $\delta_{min,B}$ が 0.06〜0.01，右側の横軸の $\delta_{max,B}$ が 0.10〜0.07 の範囲では縦軸の平均相対誤差が 20%以下と小さく，$\delta_{min,B}$ が 0.06 よりも大きい値に設定した場合や，$\delta_{max,B}$ が 0.07 よりも小さい値に設定した場合あたりからは急激に結果が悪化している．それに対して図 5.10(d) の平均相対誤差は，40%以上もある．

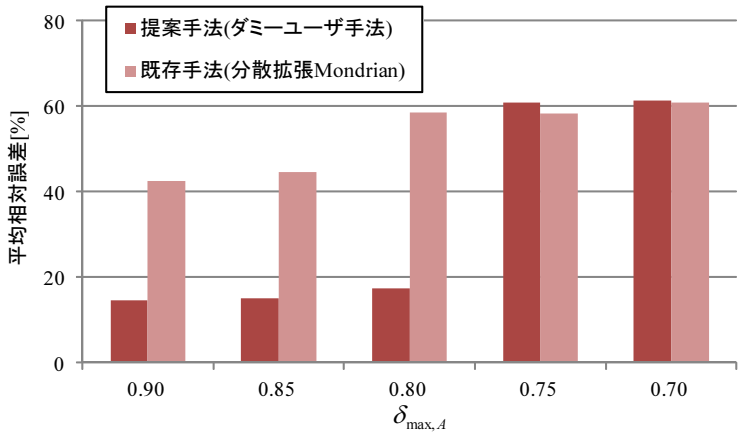
このように，提案手法において $\delta_{max,A}$，$\delta_{min,A}$，$\delta_{max,B}$，$\delta_{min,B}$ をある値よりも厳しく設定すると急激に悪化するのは，ユーザ存在情報の隠蔽の限界値 (4.1.1 節) が関係している．レセプトデータでは，機関 A のユーザ数が約 300 人で共通ユーザ数が約 230 人であるので，機関 A から見たユーザ存在情報の隠蔽の限界値 ($\delta_{max,A}$ として設定できる値の最小値，$\delta_{min,A}$ として設定できる値の最大値は，0.76($\simeq$ 230/300) である．そのため，図 5.10(c) では，$\delta_{max,A}$ や $\delta_{min,A}$ が理論限界である 0.76 に近づくと，急激に平均相対誤差が悪化している．

同様に，機関 B から見たユーザ存在情報の隠蔽の限界値は 0.066($\simeq$ 230/3500) である．図 5.10(c) においても，$\delta_{max,B}$ や $\delta_{min,B}$ が理論限界である 0.066 に近づくと，急激に平均相対誤差が悪化していることが解る．

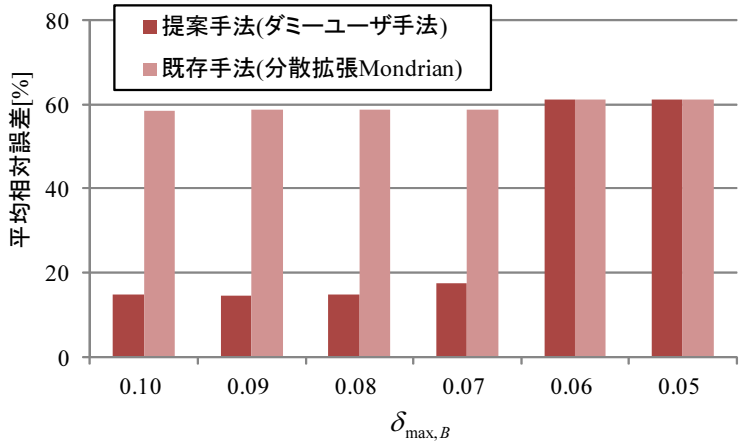
理論限界に近づいた際に急激に相対誤差が悪化することについて，どの程度の δ の設定から悪化が始まるかを詳しく見るために，図 5.10 の評価結果のうち，$\delta_{min,A}$=0.5 とした所を抜き出したグラフを図 5.11(a) に，$\delta_{min,B}$=0.01 とした所を抜き出したグラフを図 5.11(b) に示す．このグラフから分かる通り，図 5.11(a) では，限界値である $\delta_{max,B}$=0.76 から 0.1 ほど余裕を持たせた $\delta_{max,B}$=0.85 を超えたあたりから相対誤差が増加してくる．同様に図 5.11(b) では，限界値である $\delta_{max,B}$=0.06 から 0.01 ほど余裕を持たせた $\delta_{max,B}$=0.07 を超えたあたりから相対誤差が増加してくる．

また，参考に図 5.12 と図 5.13 に DM 値を用いた同様な評価結果を示す．DM 値においても先ほどと同様な傾向がみられるが，特に図 5.13 の結果をみると，限界値付近で急激に値が悪化していく様子が解りやすい．

以上のようなレセプトデータにおけるユーザ存在情報の隠蔽の限界値の評価の結果，提案手法は $\delta_{max,A}$ と $\delta_{min,A}$ を理論限界値の 0.76 から 0.1 ほど余裕を持たせた値に設定すれば，相対誤差が 20%以下の有効な匿名化結果が得られることが分かった．

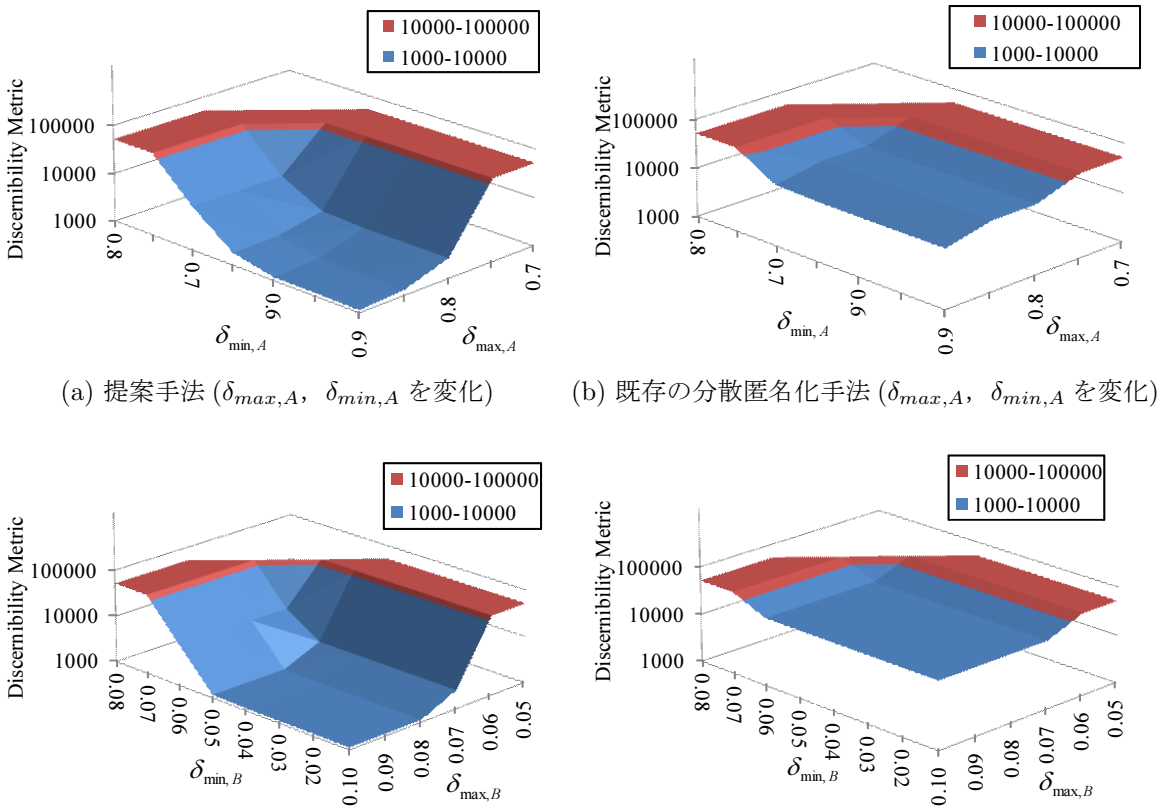


(a) $\delta_{max,A}$ を変化させた際の相対誤差の比較 ($\delta_{min,A}$=0.5)



(b) $\delta_{max,B}$ を変化させた際の相対誤差の比較 ($\delta_{min,B}$=0.01)

図 5.11: 提案手法と既存手法の相対誤差の比較 (レセプトデータ)



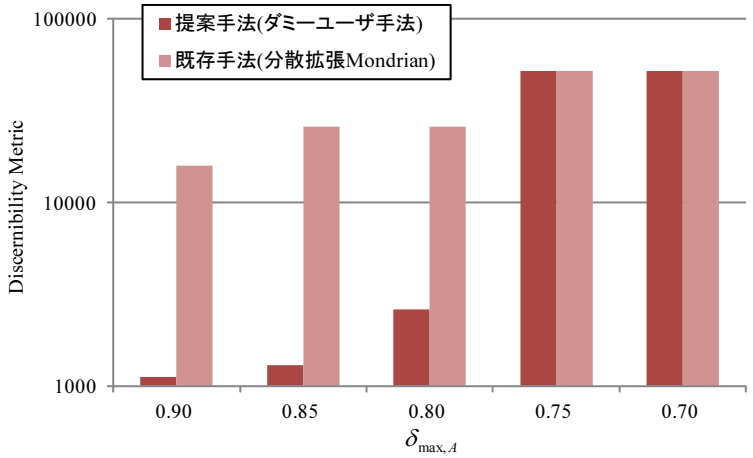
(a) 提案手法 ($\delta_{max,A}$, $\delta_{min,A}$ を変化)

(b) 既存の分散匿名化手法 ($\delta_{max,A}$, $\delta_{min,A}$ を変化)

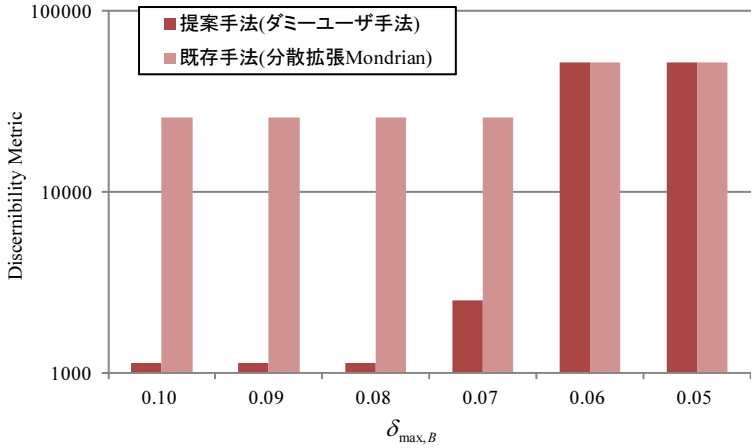
(c) 提案手法 ($\delta_{max,B}$, $\delta_{min,B}$ を変化)

(d) 既存の分散匿名化手法 ($\delta_{max,B}$, $\delta_{min,B}$ を変化)

図 5.12: δ を変化させた際の提案手法と既存手法の DM 値 (レセプトデータ)



(a) $\delta_{max,A}$ を変化させた際の相対誤差の比較 ($\delta_{min,A}$=0.5)



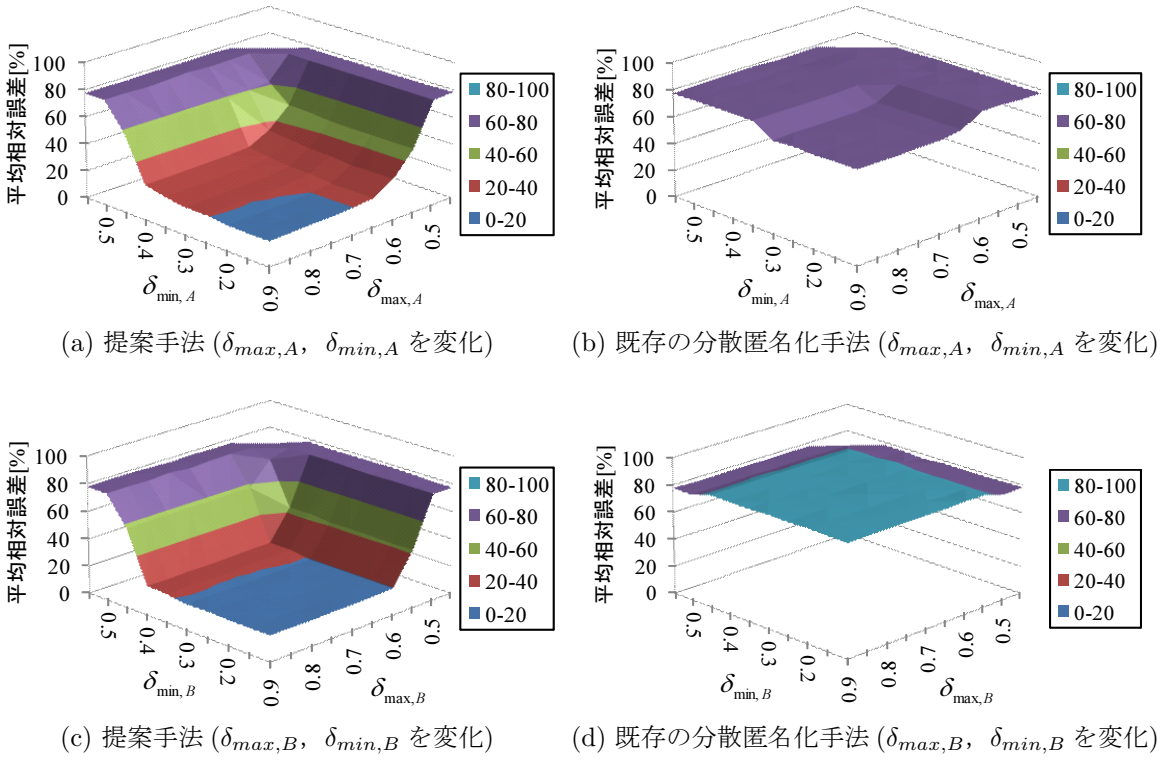
(b) $\delta_{max,B}$ を変化させた際の相対誤差の比較 ($\delta_{min,B}$=0.01)

図 5.13: 提案手法と既存の分散匿名化手法の DM 値の比較 (レセプトデータ)

国勢調査データにおける評価結果

続いて，図 5.14 に国勢調査データについての評価結果を示す．この評価では，$\theta = 3\%$，$\delta_{max,A} = \{0.9, \cdots, 0.45\}$，$\delta_{min,A} = \{0.1, \cdots, 0.55\}$，$\delta_{max,B} = \{0.9, \cdots, 0.45\}$，$\delta_{min,B} = \{0.1, \cdots, 0.55\}$ と設定している．この国勢調査データでは，機関 A に存在するユーザ (U_A) は 2400 名，機関 B に存在するユーザ (U_B) は 2400 名，共通のユーザ $(U_A \cap U_B)$ は 1200 名である．よって，δ の理論上の限界値は $1200/2400 = 0.5$ である．

この評価結果から解るように，先ほどのレセプトデータの場合と同様に，提案手法は δ_{max} や δ_{min} を理論値の 0.5 付近に設定しなければ，約 20%程度の平均相対誤差となることがわかる．そして，δ_{max} や δ_{min} を理論値付近に設定すると急激に誤差が大きくなっている (図 5.14(a) と図 5.14(c))．それに対し，既存の分散匿名化手法の評価結果である図 5.14(b) と図 5.14(d) では，δ_{max} や δ_{min} をどの値に設定しても約 80%ほどの平均相対誤差となっている．



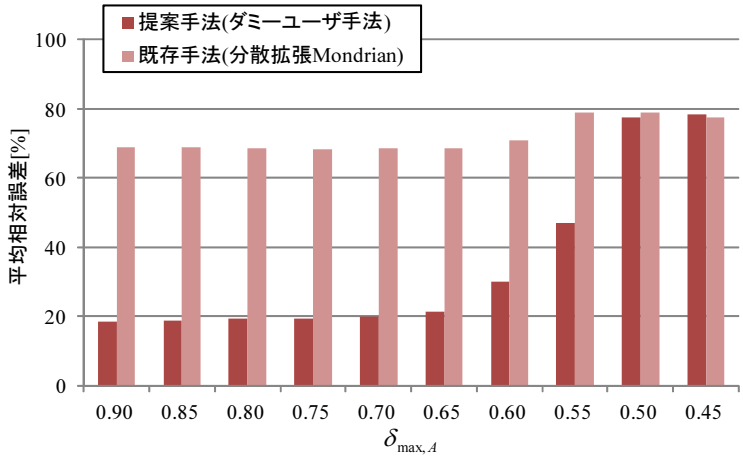
(a) 提案手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)　(b) 既存の分散匿名化手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)

(c) 提案手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)　(d) 既存の分散匿名化手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)

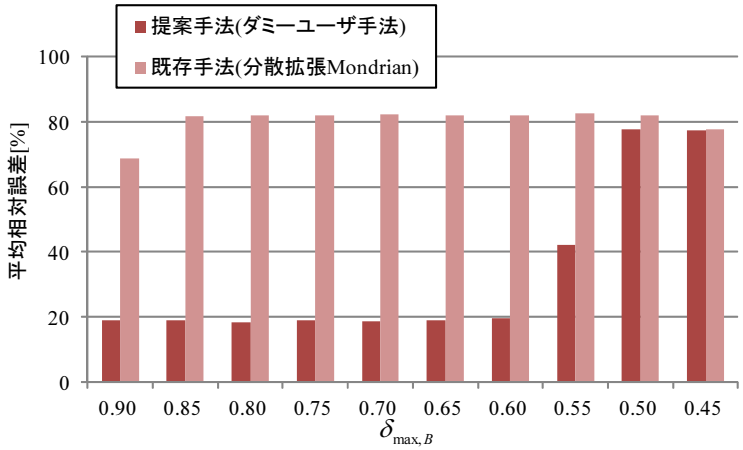
図 5.14: δ を変化させた際の提案手法と既存手法の相対誤差 (国勢調査データ)

さらに，先ほどと同様に評価結果を詳しく見るために，図 5.14 の評価結果のうち，$\delta_{min,A}$

= 0.1 とした所を抜き出したグラフを図 5.15(a) に，$\delta_{min,B} = 0.1$ とした所を抜き出したグラフを図 5.15(b) に示す．この結果が示すように，国勢調査データでは $\delta_{max,A}$ や $\delta_{max,B}$ が理論値の 0.5 から 0.1 ほどの余裕を持たせた 0.6 付近から急激に相対誤差が増加していることがわかる．



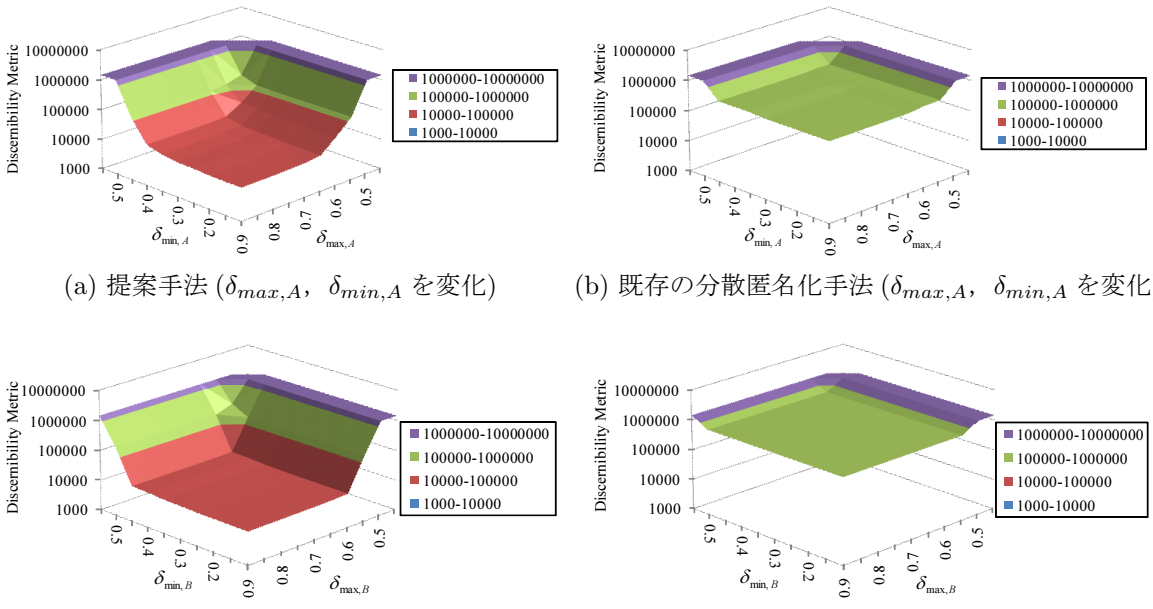
(a) $\delta_{max,A}$ を変化させた際の相対誤差の比較 ($\delta_{min,A}$=0.1)



(b) $\delta_{max,B}$ を変化させた際の相対誤差の比較 ($\delta_{min,B}$=0.1)

図 5.15: 提案手法と既存手法の相対誤差の比較 (国勢調査データ)

また，参考に図 5.16 と図 5.17 に DM 値を用いた同様な評価結果を示す．先ほどのレセプトデータの場合と同様に，DM 値においても先ほどと同様な傾向がみられる．



(a) 提案手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)　(b) 既存の分散匿名化手法 ($\delta_{max,A}$，$\delta_{min,A}$ を変化)

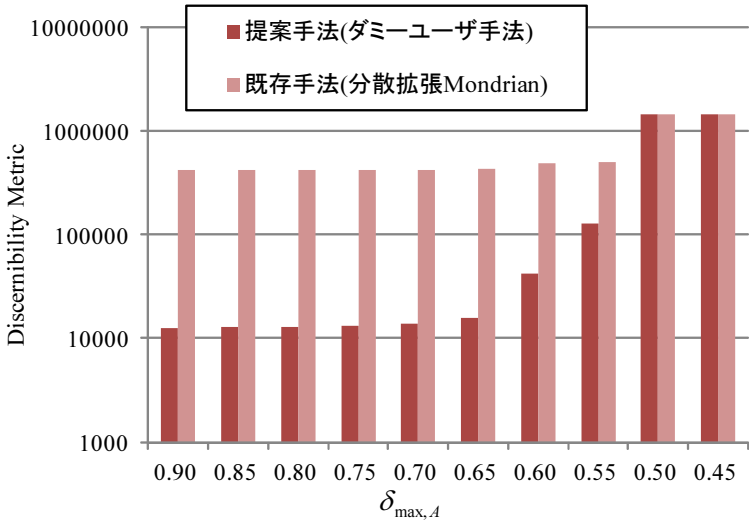
(c) 提案手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)　(d) 既存の分散匿名化手法 ($\delta_{max,B}$，$\delta_{min,B}$ を変化)

図 5.16: δ を変化させた際の提案手法と既存手法の DM 値 (国勢調査データ)

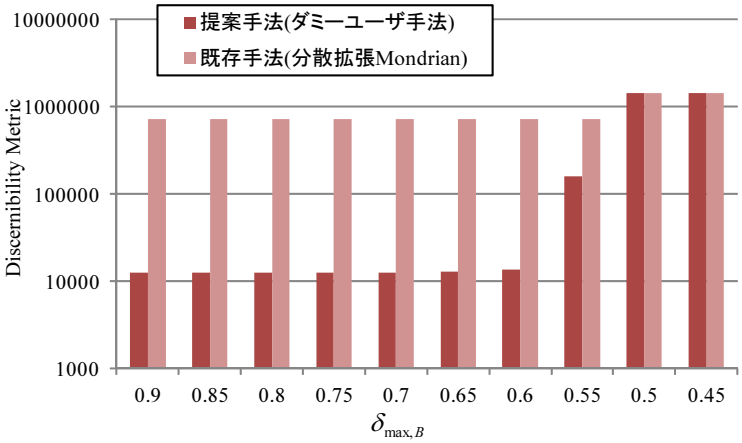
5.5.2 評価結果の考察と実用上の限界値

以上のようなレセプトデータと国勢調査データのユーザ存在情報の隠蔽の限界値の評価結果から，δ_{max} や δ_{min} を理論限界値 (4.1.1 節) 付近に設定すると，ユーザ数カウントのクエリ結果の誤差が大きくなり，データマイニング等で有効なデータを生成できなくなることがわかった．つまり，δ_{max} や δ_{min} として設定可能な実用上の限界は，4.1.1 節で説明した理論限界値よりも少し余裕を持たせた値であると考えられる．

実用上の限界がどの程度であるかを考察するために，表 5.4 にレセプトデータと国勢調査データを用いた評価結果から分かった，理論上の限界と実用上の限界を整理する．この結果をみると，レセプトデータと国勢調査データともユーザ存在情報を隠蔽する δ の実用上の限界値は，理論上の限界値から約 10〜20%ほどの余裕を持たせた値であると考えられる．つまり，限界値が 0.76 である場合は $0.76 \times 0.2 \approx 0.1$，限界値が 0.06 である場合は $0.06 \times 0.2 \approx 0.01$，限界値が 0.5 である場合は $0.5 \times 0.2 \approx 0.1$ の余裕を持たせた設定値が実用上の限界であると考えられる．



(a) $\delta_{max,A}$ を変化させた際の相対誤差の比較 ($\delta_{min,A}$=0.1)



(b) $\delta_{max,B}$ を変化させた際の相対誤差の比較 ($\delta_{min,B}$=0.1)

図 5.17: 提案手法と既存手法の DM 値の比較 (国勢調査データ)

表 5.4: ユーザ存在情報隠蔽の理論上の限界値と実用上の限界値

評価データ	δ	理論上の限界値	実用上の限界値	理論上と実用上の限界値の差 (約)
レセプトデータ	$\delta_{max,A}$	0.76	0.85	+0.1
	$\delta_{min,A}$	0.76	0.65	−0.1
	$\delta_{max,B}$	0.066	0.08	+0.01
	$\delta_{min,B}$	0.066	0.05	−0.01
国勢調査データ	$\delta_{max,A}$	0.50	0.60	+0.1
	$\delta_{min,A}$	0.50	0.40	−0.1
	$\delta_{max,B}$	0.50	0.60	+0.1
	$\delta_{min,B}$	0.50	0.40	−0.1

5.5.3 実際のアプリケーションにおける意義

本節では，実際のアプリケーションを想定した評価データであるレセプトデータにおける実用上の限界値が，実際のレセプトデータを用いた利用場面において有効であるかについて考察を行う．

今回のレセプトデータを用いた利用場面では，専門病院への通院している可能性を隠したいと考えられる．4.1.1 節で示した *δ-site-presence* の設定指針のとおり，専門病院への通院は比較的プライバシ性が高い情報であると考えられるので，$\delta_{max,B}$=0.1 と設定することが望ましい．また，通院していないことが確定しないことが望ましいので，$\delta_{min,B}$=0.01 と設定することが望ましい．ここで，先ほどのユーザ存在情報隠蔽の限界値の結果，レセプトデータにおいて $\delta_{max,B}$=0.1，$\delta_{min,B}$=0.01 が実用上の限界であった．

よって，提案手法はレセプトデータを用いたユースケースにおいて，十分意のあるユーザ存在情報の隠蔽が可能であると考える．

5.6 対応可能ユーザ数の評価

本節では，提案手法の処理速度を計測し，現実的な時間で処理可能なユーザ数を調べ，提案手法における対応可能なユーザ数の限界を評価する．まず，5.6.1 節で提案手法の処理

速度の計測結果を述べ，続いて 5.6.2 節で提案手法で対応可能なユーザ数におけるサービス例を述べる．

5.6.1　処理速度の評価結果

表 5.5: 速度評価の結果

データ種類	母集団ユーザ数 (U)	U_A	U_B	$U_A \cap U_B$	計測結果 (分)
国勢調査データ	300	150	150	75	15
	600	300	300	150	39
	1200	600	600	300	80
	2400	1200	1200	600	253
	4800	2400	2400	1200	1066
	9600	4800	4800	2400	4824
レセプトデータ	5000	約 3500	約 300	約 230	1321

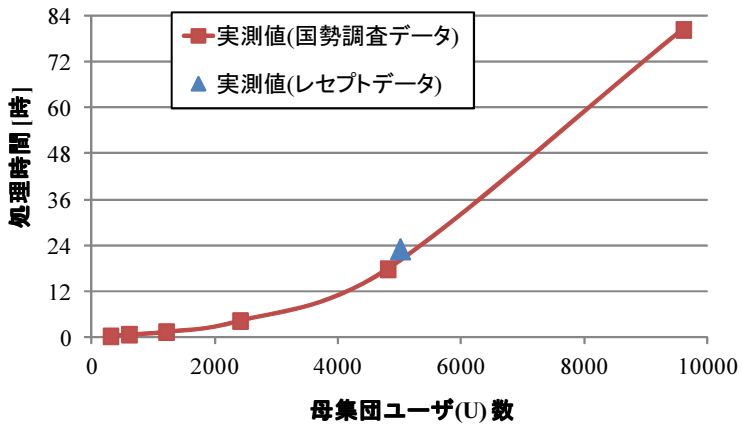


図 5.18: 動作速度 (レセプトデータ)

本節では，提案手法の処理速度を計測し，現実的な時間で処理可能なユーザ数を調べ，提案手法における対応可能なユーザ数の限界を評価する．表 5.5 に提案手法の処理速度を計測した際の母集団ユーザ数 (U) と処理速度の計測結果を示す．なお，計測値はそれぞれのデータサイズについて 5 回計測した平均値である．つまり国勢調査データの場合は，デー

タの生成と速度の計測を行う作業を 1 回の作業として計 5 回行い，それらの平均を計算している．レセプトデータの場合は，データを生成し直すことはしないので，そのデータに対して 5 回速度計測を行った平均である．

また，図 5.18 にこれらの計測結果をプロットしたグラフを示す．なお，提案手法の計算量と通信量は 6.1.3 節で評価を行っており，主に母集団ユーザ数によって計算量と通信量が決まり，母集団ユーザ数を N とおいたときに計算量が $O(N^2 loglogN)$，通信量が $O(N)$ となる．

この結果から分かるとおり，約 5000 人の母集団ユーザであれば 1 日程度，約 10000 人の母集団ユーザであれば 3 日程度で処理が可能である．この処理速度において，どのようなサービスが提供可能であるかについて，次節で説明する．

5.6.2 対応可能なサービスの例

本節では，母集団ユーザが 10000 人以下で，3 日程度でデータ提供のサービスの例を示し，提案手法によって新たなサービス提供が可能である事を示す．そして，現状の匿名データを提供するサービスと比較して，提案手法によって提供されるサービスが有意義であることを明確にする．

まず，以下にサービス例を 2 つ示す．

- 企業内の会社社員の健康状態分析のための匿名データ提供サービス

  提案手法を用いることによって，従業員数が 10000 人以下の企業の従業員の健康状態を分析するために，スポーツセンターと専門病院のデータを連携するサービスが可能であると考えられる．このサービスでは，例えば，企業が提携しているスポーツセンターが保有する利用者の運動時間に関する情報と，企業が提携している病院の患者の疾病情報に関する情報に対して，提案手法を用いてデータ連携する．そして，データ連携して匿名化されたデータを，医学研究を行う研究機関へ提供することで，運動量と疾病の相関関係などを分析し，従業員の健康促進の活動に生かすというサービスである．この例では性病等の専門病院への通院をスポーツセンターに知られたくないと考えられるため，提案手法によってユーザ存在情報を隠しながら分散匿名

化を行う必要がある．また，この例における母集団は，ある企業の従業員の10000名であり，スポーツセンターと病院において個人を識別する共通のIDとして，健康保険の保険者番号と記号と被保険者番号を用いる．一般に民間企業における健康保険組合では，保険者番号と記号によって企業(事業所)を一意に識別することになる[10]．よって，スポーツセンターと病院では，あらかじめ保険者番号と記号と被保険者番号の値の範囲を共有しておくことで，10000名の母集団を共有することができる．

- 病院と専門病院における医学分析のための匿名データ提供サービス

  提案手法を用いることによって，大規模な病院と専門病院における医学分析のための匿名データを提供するサービスが可能であると考える．このサービスでは，例えば，ある大病院のある期間における10000名以下の患者の医薬品・疾病情報と，ある専門病院の医薬品・疾病情報に対して，提案手法を用いてデータを結合し，匿名化する．そして，匿名化されたデータを医学研究を行う機関へ提供することで，例えば，専門病院で処方された医薬品と大病院における疾病の相関関係を分析することで，医薬品の副作用分析ができると考えられる．そして，この例においても，専門病院への過去の通院を大病院に知られたくないと考えられるため，提案手法によってユーザ存在情報を隠しながら分散匿名化を行う必要がある．なお，病床数が1000床にもなる大規模な病院における1日平均入院患者は約1000人[11]であり，全国における一般病床における平均入院日数は約18日である[61]．よって，大病院の約半年(6カ月)分の入院患者を母集団とした場合，以下のように約10000人となる．

$$\begin{aligned}\text{ある期間における入院患者数} &= 1\text{日平均入院患者数} \times \text{期間}/\text{平均入院日数} \\ &= 1000\text{人} \times (30\text{日} \times 6\text{カ月})/18\text{日} = 10000\text{人} \quad (5.2)\end{aligned}$$

このように，母集団ユーザ数が10000人以下となるようなサービス例は十分存在する．また，上記に上げた例は医学分野における例であるが，それ以外にもWebサービス事業者間のデータ連携など様々なサービス例が存在する．

[10]例えば「NEC健康保険組合」の場合，保険者番号と記号の組によって「日本電気株式会社」などの会社が識別される．そして，会社の社員番号が被保険者番号となっている．

[11]例えば，病床数が1162床の大病院に分類される東京大学医学部付属病院における平成23年度の1日平均入院患者数は1049人である．(出展：東京大学医学部付属病院ホームページ `http://www.h.u-tokyo.ac.jp/about/beds/index.html`)

また，現状において匿名データを提供するサービスとしては，独立行政法人統計センター[12]による，公的統計の調査票情報を加工した匿名データの提供が知られている．このサービスでは，政府が取得した統計情報のうち必要な情報を，統計法で定められた匿名化方法によってデータを切り落としたりすることによる匿名化 (k-匿名化ではない) を行い，匿名化データを提供するサービスである．これは，独立行政法人統計センターが保持しているデータに対して匿名化処理を行うものであるので，集中型の匿名化手法が用いられている．そして，現状のこのサービスでは，データの依頼依頼があってから 14 日以内にデータを提供するとされている [70].

また，位置情報を匿名化するサービスとして株式会社エヌ・ティ・ティ・ドコモ[13]による「モバイル空間統計」というサービスが知られている．このサービスでは，携帯電話の基地局などを利用して取得した位置情報を，匿名化して提供している．そして，この匿名化処理には数日を要するとされている [54][14].

このような現状を考えると，現状の集中型の匿名データ提供サービスを拡張し提案手法を導入することで，新たに分散型の匿名データ提供サービスを提供することが考えられる．10000 人規模であれば 3 日程度で処理が完了するため，現状のサービスのデータ提供の速度を低下させることなく，異なる機関が保持するデータに対してもデータ提供が可能になる．

以上のように，母集団ユーザ数が 10000 人以下で 3 日間程度で処理が可能でる提案手法は，様々なサービス案が考えられ，提案手法は十分意義があると考える．

5.7 分割におけるダミーユーザの偏りの評価

本節では，提案の分割点決定関数が有効に機能しているかを評価するために，決定された分割点における分割後のユーザ集合のダミーユーザの偏りがどの程度であるかを評価した．図 5.19 に，分割前のユーザ集合のユーザ数に対する，事業者 A,B のダミーユーザの偏りを示す．このグラフではダミーユーザの偏り b を，以下の式で示したように，分割点 c で分割後の 2 つのユーザ集合 (U_{hi}，U_{low}) における，ユーザ数 ($|U_{hi}|$，$|U_{low}|$) に対する事業者 $n \in \{A, B\}$ のダミーユーザ数 ($|dummy(n, U_{hi})|$，$|dummy(n, U_{low})|$) の割合の差を計算し，

[12] http://www.nstac.go.jp/
[13] http://www.nttdocomo.co.jp/
[14] ホームページにおける説明では「モバイル空間統計の作成には、数日を要します」と記載されている

その値の事業者 A,B での平均としている．

$$b(c) = \sum_{n \in \{A,B\}} \left| \frac{|dummy(n, U_{hi})|}{|U_{hi}|} - \frac{|dummy(n, U_{low})|}{|U_{low}|} \right| /2 \tag{5.3}$$

このグラフは，提案手法を 10 回実行し，分割前のユーザ集合のユーザ数を適切な区間で区切って偏り b の平均値，最大値，最小値を計算した結果である．なお，$\delta_{max,A}$=0.7 とし，$\delta_{min,A}$ と $\delta_{max,B}$ と $\delta_{min,B}$ は設定していない．また，その他のパラメータは 5.4.2 節の評価と同じにしている．

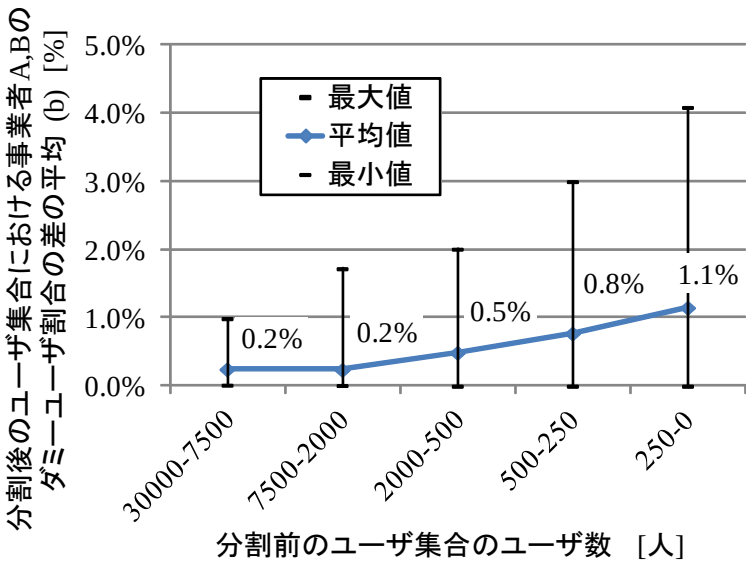


図 5.19: ダミーユーザの偏りの評価

図 5.19 に示したように，分割が進み，ユーザ集合が小さくなるに従って，ダミーユーザの偏り b が大きくなる傾向がある．これは，分割対象のユーザ集合が小さいと分割点候補が少なくなってしまうので，偏りが小さくなるような分割点を選べなくなるためである．しかし，ダミーユーザの偏り b の平均は約 1%程度であり，小さいと考えられる．

このように，提案の分割点決定関数によって，分割後のユーザ集合における事業者 A,B のダミーユーザの偏りが小さくなるような分割点が選ばれることが確かめられた．よって，提案の分割点決定関数は設計時に意図したとおりに有効に機能していることがわかった．

5.8 評価結果のまとめと考察

本節では，5.4〜5.7 節で示した評価結果に対して考察を行う．5.4 節の有効性評価では，まず 5.4.1 節で重み α の適切な設定値を評価し，データによって α がデータの有効性に与える影響は異なるが α を 0.9 付近の大きい値に設定すると良いことが分かった．そして，5.4.2 節で既存の分散匿名化手法と比較し，提案手法は既存手法よりも約 25〜50 ポイント相対誤差を低下させることができることを確認した．さらに 5.4.3 節では，既存の集中型の手法と有効性を比較した結果，提案手法は既存の集中型の手法とほぼ同等の良い結果となることが分かった．これらの有効性評価の結果から，提案手法は分散型に対応しつつも，既存の集中型と同等に有効であることが分かった．

続いて，5.5 節のユーザ存在情報隠蔽の限界値の評価の結果，提案手法において設定できる δ の値は，理論上の限界値から約 10〜20%ほどの余裕を持たせた値であることが分かった．これは，例えば今回のレセプトデータを用いたユースケースにおいては，プライバシを保つことができるとされる $\delta_{max,B}$=0.1 を十分満たすものであり，意義のあるユーザ存在情報の隠蔽が可能であることが分かった．

さらに 5.6 節での評価結果より，提案手法は母集団ユーザ数が 10000 人以下で 3 日間程度で処理が可能であることが分かった．また，母集団ユーザ数が 10000 人以下で 3 日間程度でデータ提供が行えるというサービスはいくつか考えられるため，提案手法は実際のアプリケーションに適用可能であると考える．また，5.7 節での評価では，提案の分割点決定関数が有効に機能していることを確認した．

以上の結果から，本論文で提案する手法を用いることによって，国勢調査データや医療データにとどまらず，様々な種類のパーソナル情報をサービス事業者間で安全にデータ連携することができると考えられる．

第6章　計算量・通信量と安全性の評価

本章では，提案手法の計算量・通信量と安全性の評価結果について述べる．まず 6.1 節で，提案手法の計算量・通信量を算出し，既存のセキュア計算と比較して大幅な増加が無いことを示す．続いて 6.2 節で，提案手法の安全性を評価し，プライバシ性の高い情報が漏洩していないことを示す．

6.1　計算量・通信量の評価

本節では，提案手法の平均的な計算量と通信量のオーダーを算出する．提案手法では，Step2 と Step3 において複数のセキュア計算を実行している．このセキュア計算は，Step1〜3 の他の処理に比べて計算量と通信量が大きく，提案手法の計算量と通信量のオーダーを算出する際には，無視できるくらいに小さい．そこで，まず 6.1.1 節と 6.1.2 節で Step2 と Step3 において実行しているセキュア計算を整理し，Step2 と Step3 のそれぞれの平均計算量と平均通信量を算出する．その後，6.1.3 節で，Step2 と Step3 の平均計算量と平均通信量を合計することで，提案手法の平均計算量と平均通信量を算出する．最後に，6.1.4 節で，算出した平均計算量と平均通信量に対して考察を行う．

6.1.1　Step2 の計算量と通信量の算出

本節では，Step2 において実行しているセキュア計算を整理し，Step2 の平均計算量と平均通信量を算出する．

Step2 で実行しているセキュア計算の整理

Step2 では，図 4.4 に示したように，「(1) 分割点を決定する処理」(図 4.4 の 2 の処理) と「(2) 指標を確認する処理」(図 4.4 の 3 の処理) の 2 箇所でセキュア計算を利用している．

「(1) 分割点を決定する処理」では，4.2.3 節の図 4.5 のシーケンス図に示したように，*secure comparison*，*secure set intersection*，*secure k-nearest neighbor* の 3 種類のセキュア計算が用いられる.「(2) 指標を確認する処理」では，4.2.3 節の図 4.6 のシーケンス図に示したように，*secure set intersection* が用いられる．これらのセキュア計算の計算量・通信量は，入力となる値のデータサイズ (入力データサイズ) によって変わってくる．そこで，これらのセキュア計算の入力データサイズと実行回数を整理する．表 6.1 は，Step2 における 1 回の分割において実行されるセキュア計算の入力データサイズと実行回数を整理した結果である．なお，*secure comparison* の入力データサイズは比較する値の個数であり，*secure set intersection* と *secure k-nearest neighbor* の入力データサイズは入力となる集合の要素の個数である．

表 6.1: Step2 における 1 回の分割において実行されるセキュア計算

処理内容	実行するセキュア計算	入力データサイズ	実行回数
(1) 分割点を決定する処理	*secure comparison*	2	1
	secure set intersection	分割前のユーザ数	分割点候補数
	secure k-nearest neighbor	分割点候補数	1
(2) 指標を確認する処理	*secure set intersection*	分割後のユーザ数	12

まず「(1) 分割点を決定する処理」で実行する *secure comparison* について整理する．このセキュア計算は，双方の機関が持つ値の大小を比較する処理であるため，*secure comparison* の入力データサイズは 2 である．また，このセキュア計算は 1 回の分割において最初に実行されるだけであるため，処理回数は 1 である．

続いて,「(1) 分割点を決定する処理」で実行する *secure set intersection* について整理する．このセキュア計算は，各分割点候補のスコア値を計算するために用いられる．このセキュア計算の入力データは，分割前のグループのユーザ ID の集合であるため，入力データサイズは分割前のグループのユーザ数となる．そして，この処理は分割点候補毎に行われる．

さらに,「(1) 分割点を決定する処理」で実行する *secure k-nearest neighbor* について整理

する．このセキュア計算はスコア値が最大となる分割点候補を選ぶために用いられ，入力データは，各分割点候補について計算したスコア値の部分計算結果である．よって，入力データサイズは分割点候補数となり，処理回数は 1 である．

最後に,「(2) 指標を確認する処理」で実行しているセキュア計算を整理する．この処理では，4.2.3 節の図 4.6 のシーケンス図に示したように，*k*-匿名性の確認と *δ-site-presence* の確認の 2 つの処理において，*secure set intersection* を実行している．まず *k*-匿名性の確認の処理における *secure set intersection* の入力サイズと実行回数を整理する．この処理では，分割後のグループに対して *secure set intersection* を実行している．分割後のグループは 2 つあるので，*k*-匿名性の確認の処理では，入力サイズは分割後のグループのユーザ数で，実行回数は 2 回となる．続いて，*δ-site-presence* の確認の処理の入力サイズと実行回数を整理する．この処理では，*secure set intersection* を途中計算のために 1 回と，$\delta_{max,B}$，$\delta_{max,B}$，$\delta_{min,A}$，$\delta_{min,B}$ の確認のために 4 回実行している．この確認も分割後の 2 つのグループについて行うので，*δ-site-presence* の確認のための実行する *secure set intersection* は，$(1+4)\times 2 = 10$ 回である．よって,「(2) 指標を確認する処理」で実行している *secure set intersection* は，*k*-匿名性の確認の処理の 2 回と，*δ-site-presence* の 10 回の合計 12 回となる．そして，入力データサイズは分割後のグループのユーザ数となる．

Step2 の分割後のグループサイズとユーザ数

先ほどの表 6.1 の整理の結果，*secure comparison* の入力データサイズと実行回数は一定であるが，*secure set intersection* と *secure k-nearest neighbor* は分割が進むにつれて入力データサイズと実行回数が変化することが分かった．よって分割が進むにつれて，分割後のグループのユーザ数や分割点候補がどのように変化していくかを整理する．

分割の 1 回目は分割対象のグループのサイズ (ユーザ数) は母集団ユーザ数 $|U|$ となる．なお，簡略化のため $|U|$=N とおく．本手法の分割では多少の偏りはあるが平均的に中央で分割される[1]ので 2 回目以降のグループサイズは $\frac{N}{2}, \frac{N}{4}, \cdots$ となる (図 6.1)．また，これらのグループの個数は $2, 4, \cdots$ となる．さらに，各グループにおける分割点候補は，多くてもグループのサイズと同じなので $N, \frac{N}{2}, \frac{N}{4}, \cdots$ となる (表 6.2)．

[1]提案手法の分割点決定関数では中央値を考慮して分割点が選ばれるので，多少のずれはあるが平均的に中央で分割される．

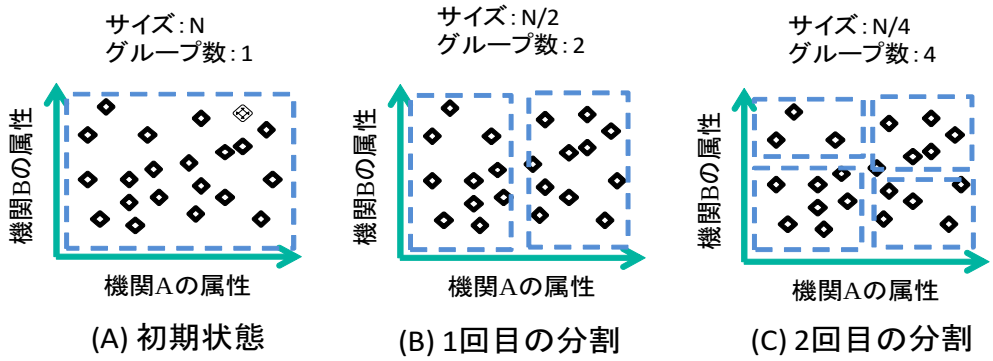


図 6.1: 分割後のグループ数とユーザ数

表 6.2: 各分割におけるグループ数とユーザ数と分割点候補数

分割 回数	分割前の グループ数	分割前の ユーザ数	分割 候補数	分割後の ユーザ数
1 回目	1	N	N	$N/2$
2 回目	2	$N/2$	$N/2$	$N/4$
3 回目	4	$N/4$	$N/4$	$N/8$
$log(N/k) - 1$ 回目	$N/(2k)$	$2k$	$2k$	k

なお，最も多く分割出来るようなデータセットであった場合，分割後のユーザ数が k-匿名性の k の値と同じくなった際に分割が止まる．よって，最も多く分割出来る場合は，$log(N/k) - 1$ 回目の分割で分割が止まる．そして，最も多く分割出来る場合における，最後の分割前のユーザ数は $2k$ であり，分割前のグループ数は $N/(2k)$ となる．

なお，分割の回数の最大値は QI(準識別子) の個数には依存しない．仮に QI の個数が増えた場合は分割する属性の候補が増えることになり分割できる可能性は高くなるが，分割が止まる際の条件はあくまで分割後のグループのレコード数が重要となるため，仮に QI の個数が増えたとしても分割の回数の最大値が増えることは無い．つまり QI の個数は分割の回数の最大値には影響は無い．

以降，これらの整理をもとに「(1) 分割点を決定する処理」と「(2) 指標を確認する処理」の平均計算量・通信量を個別に算出し，最後に Step2 の平均計算量・通信量を算出する．

Step2 の「(1) 分割点を決定する処理」の平均計算量

Step2 の「(1) 分割点を決定する処理」の平均計算量を算出する．まず,「(1) 分割点を決定する処理」の *secure comparison* の計算量を考える．表 6.1 の整理の結果,「(1) 分割点を決定する処理」の *secure comparison* の入力データサイズと処理回数が一定である．このような場合は，計算量はほぼ一定であると考えられるので，計算量は $O(1)$ とする．次に,「(1) 分割点を決定する処理」における *secure set intersection* の平均計算量を算出する．*secure set intersection* の計算量は，入力データサイズとなる 2 つの集合の要素数を両方とも M とおいたとき $O(MloglogM)$ となる [14]．表 6.1 と表 6.2 の整理の結果，1 回目の分割での *secure set intersection* の計算量は，サイズ N の 1 つのグループに対する計算量を N 個の分割点候補分行うので

$$O(NloglogN) \times 1 \times N \tag{6.1}$$

となる．また，2 回目の分割ではサイズ $\frac{N}{2}$ の 2 つのグループに対する計算量を $\frac{N}{2}$ 個の分割点候補分行うので

$$O(\frac{N}{2}loglog\frac{N}{2}) \times 2 \times \frac{N}{2} \tag{6.2}$$

となる．そして，3 回目は,

$$O(\frac{N}{4}loglog\frac{N}{4}) \times 4 \times \frac{N}{4} \tag{6.3}$$

となる．ここで，*log* は単調増加関数であることから，これらを足した値は以下の関係を満たす.

$$\begin{aligned}
&N^2loglogN + \frac{N^2}{2}loglog\frac{N}{2} + \frac{N^2}{4}loglog\frac{N}{4} + \cdots \\
&< N^2loglogN + \frac{N^2}{2}loglogN + \frac{N^2}{4}loglogN + \cdots \\
&< (1 + \frac{1}{2} + \frac{1}{4} + \cdots)N^2loglogN \\
&< 2N^2loglogN
\end{aligned} \tag{6.4}$$

つまり，2 回目以降の分割の計算量の合計は 1 回目の計算量よりも小さい．よって,「(1) 分割点を決定する処理」における *secure set intersection* の平均計算量は

$$O(N^2loglogN) \tag{6.5}$$

である．

同様に,「(1) 分割点を決定する処理」における *secure k-nearest neighbor* の平均計算量を算出する．*secure k-nearest neighbor*[52] は，入力データサイズとなる比較対象の要素数を M とおいたときに $O(M^2)$ の計算量である．よって，1 回目の分割における計算量は

$$O(N^2) \times 1 \tag{6.6}$$

となる．また，2 回目は,

$$O((\frac{N}{2})^2) \times 2 \tag{6.7}$$

となり，3 回目は,

$$O((\frac{N}{4})^2) \times 4 \tag{6.8}$$

となる．そして，先ほどと同様にこれらを足した値は以下の関係があるため，2 回目以降の分割の計算量の合計は 1 回目の計算量よりも小さい.

$$\begin{aligned} & N^2 + (\frac{N}{2})^2 \times 2 + (\frac{N}{4})^2 \times 4 + \cdots \\ & < (1 + \frac{1}{2} + \frac{1}{4} + \cdots)N^2 \\ & < 2N^2 \end{aligned} \tag{6.9}$$

ゆえに,「(1) 分割点を決定する処理」における *secure k-nearest neighbor* の平均計算量は

$$O(N^2) \tag{6.10}$$

である.

Step2 の「(1) 分割点を決定する処理」の平均通信量

続いて，Step2 の「(1) 分割点を決定する処理」の平均通信量を算出する．先ほどと同様に,「(1) 分割点を決定する処理」の *secure comparison* の通信量は $O(1)$ となるので，ここでは *secure set intersection* と *secure k-nearest neighbor* の通信量を算出する.

まず「(1) 分割点を決定する処理」の *secure set intersection* の平均通信量を算出する．*secure set intersection*[14] の通信量は，2 つの集合の要素数を両方とも M とおいたとき

$O(M)$ である．まず，1 回目の分割での通信量は，サイズ N の 1 つのグループに対する通信を N 個の分割点候補分行うので

$$O(N) \times 1 \times N \tag{6.11}$$

となる．そして，2 回目は

$$O(\frac{N}{2}) \times 2 \times \frac{N}{2} \tag{6.12}$$

となる．また，3 回目は，

$$O(\frac{N}{4}) \times 4 \times \frac{N}{4} \tag{6.13}$$

となる．

よって，先ほどと同様に，これらを足した値は以下の関係があり，2 回目以降の分割の通信量の合計は 1 回目の通信量よりも小さい．

$$\begin{aligned} & N \times 1 \times N + \frac{N}{2} \times 2 \times \frac{N}{2} + \frac{N}{4} \times 4 \times \frac{N}{4} && (6.14) \\ & < (1 + \frac{1}{2} + \frac{1}{4} + \cdots)N^2 \\ & < 2N^2 && (6.15) \end{aligned}$$

ゆえに，「(1) 分割点を決定する処理」の *secure set intersection* の平均通信量は

$$O(N^2) \tag{6.16}$$

である．

同様に，「(1) 分割点を決定する処理」の *secure k-nearest neighbor* の通信量を算出する．*secure k-nearest neighbor* は，比較対象の要素数を M とおいたときに $O(M^2)$ の通信量である [52]．これも，さきほどと同様に計算すると，

$$O(N^2) \tag{6.17}$$

となる．

Step2 の「(2) 指標を確認する処理」の平均計算量

さらに,「(2) 指標を確認する処理」における *secure set intersection* の平均計算量を算出する．この処理は 1 回目の分割における入力データサイズは分割後のユーザ数である $N/2$, 実行回数は 12 回であるので，1 回目の分割における計算量は,

$$O(\frac{N}{2}loglog\frac{N}{2}) \times 1 \times 12 \tag{6.18}$$

となる．また，2 回目は 2 つのグループに対して行うので,

$$O(\frac{N}{4}loglog\frac{N}{4}) \times 2 \times 12 \tag{6.19}$$

となる．同様に，3 回目は 4 つのグループに対して行うので,

$$O(\frac{N}{8}loglog\frac{N}{8}) \times 4 \times 12 \tag{6.20}$$

となる．そして，最も上手く分割出来た場合は $log(N/k) - 1$ 回目まで分割が可能であり, その際の計算量は

$$O(kloglogk) \times \frac{N}{2k} \times 12 \tag{6.21}$$

となる．これらを足した値は，$k << N$ という前提で考えると，以下のような関係が成り立つ.

$$\begin{aligned}
&(\frac{N}{2}loglog\frac{N}{2} + \frac{N}{4}loglog\frac{N}{4} \times 2 + \frac{N}{8}loglog\frac{N}{8} \times 4 + \cdots + kloglogk \times \frac{N}{2k}) \times 12 \\
&= (\frac{N}{2}loglog\frac{N}{2} + \frac{N}{2}loglog\frac{N}{4} + \frac{N}{2}loglog\frac{N}{8} + \cdots + \frac{N}{2}loglogk) \times 12 \\
&< 12(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \cdots + \frac{1}{2})NloglogN \\
&= 6(log(N/k) - 1)NloglogN \\
&< 6NlogNloglogN
\end{aligned} \tag{6.22}$$

となる．ゆえに,「(2) 指標を確認する処理」における *secure set intersection* の平均計算量は,

$$O(NlogNloglogN) \tag{6.23}$$

である.

Step2 の「(2) 指標を確認する処理」の平均通信量

同様に「(2) 指標を確認する処理」における *secure set intersection* の平均通信量を算出する．*secure set intersection*[14] の通信量は，2 つの集合の要素数を両方とも M とおいたとき $O(M)$ であるため，1 回目の分割の通信量は

$$O(\frac{N}{2}) \times 1 \times 12 \tag{6.24}$$

となる．また，2 回目の分割の通信量は

$$O(\frac{N}{4}) \times 2 \times 12 \tag{6.25}$$

となり，3 回目の分割の通信量は

$$O(\frac{N}{8}) \times 4 \times 12 \tag{6.26}$$

となる．そして，最も上手く分割出来た場合の $log(N/k) - 1$ 回目の通信量は

$$O(k) \times \frac{N}{2k} \times 12 \tag{6.27}$$

となる．

これも先ほどと同様に計算すると，

$$\begin{aligned}
&(\frac{N}{2} + \frac{N}{4} \times 2 + \frac{N}{8} \times 4 + \cdots + k \times \frac{N}{2k}) \times 12 \\
&= 12(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \cdots + \frac{1}{2})N \\
&= 6(log(N/k) - 1)N \\
&< 6NlogN
\end{aligned} \tag{6.28}$$

となる．よって，「(2) 指標を確認する処理」における *secure set intersection* の平均通信量は，

$$O(NlogN) \tag{6.29}$$

である．

表 6.3: Step2 における平均計算量と通信量

処理内容	実行するセキュア計算	平均計算量	平均通信量
(1) 分割点を決定する処理	*secure comparison*	$O(1)$	$O(1)$
	secure set intersection	$O(N^2 loglogN)$ (式 6.5)	$O(N^2)$ (式 6.16)
	secure k-nearest neighbor	$O(N^2)$ (式 6.10)	$O(N^2)$ (式 6.17)
(2) 指標を確認する処理	*secure set intersection*	$O(NlogNloglogN)$ (式 6.23)	$O(NlogN)$ (式 6.29)

Step2 の平均計算量と通信量のまとめ

Step2 の平均計算量と通信量の算出結果をまとめると，表 6.3 のようになる．そして，これらを合計した Step2 の平均計算量は，

$$\begin{aligned} &O(1) + O(N^2 loglogN) + O(N^2) + O(NlogNloglogN) \\ &= O(N^2 loglogN) \end{aligned} \tag{6.30}$$

となる．同様に Step2 の平均通信量は，

$$\begin{aligned} &O(1) + O(N^2) + O(N^2) + O(NlogN) \\ &= O(N^2) \end{aligned} \tag{6.31}$$

となる．

6.1.2　Step3 の計算量と通信量の算出

続いて，Step3 の平均計算量と通信量を算出する．

Step3 の平均計算量

まず Step3 の平均計算量を算出する．Step3 では，4.2.3 節で示したように，*secure set intersection* が実行される．*secure set intersection* は，分割後の各グループについて，各セ

ンシティブ属性の属性値について実行される．また分割後のグループは，最も多く分割できた場合，サイズが k のグループが N/k 個できる．よって，センシティブ属性の属性の個数を S とおくと，Step3 で *secure set intersection* の計算量は

$$O(kloglogk) \times N/k \times S \tag{6.32}$$

となる．ここで，$1 < S << N$，$1 < k << N$ であるので，

$$\begin{aligned} & kloglogk \times N/k \times S \\ & = NSloglogk \approx N \end{aligned} \tag{6.33}$$

という関係が成り立つ．ゆえに，もっとも多く分割できるようなデータセットの場合における，Step3 の平均計算量は，

$$O(N) \tag{6.34}$$

となる．

Step3 の平均通信量

続いて Step3 の平均通信量を算出する．*secure set intersection* の通信量は，2 つの集合の要素数を両方とも M とおいたとき $O(M)$ である．そのため Step3 の *secure set intersection* の通信量は，先ほど同様に $1 < S << N$ と考えると，

$$\begin{aligned} & k \times N/k \times S \\ & = N \times S \approx N \end{aligned} \tag{6.35}$$

となる．

そして，もっとも多く分割できるようなデータセットの場合における，Step3 の平均通信量は，

$$O(N) \tag{6.36}$$

となる．

6.1.3　提案手法の平均計算量と通信量

以上の計算により，Step2 の平均計算量 (式 6.30) と Step3 の平均計算量 (式 6.34) を足した，提案手法の平均計算量は，

$$\begin{aligned} & O(N^2 loglogN) + O(N) \\ & = O(N^2 loglogN) \end{aligned} \tag{6.37}$$

となる．

また，Step2 の平均通信量 (式 6.31) と Step3 の平均通信量 (式 6.36) を足した，提案手法の平均通信計算量は，

$$\begin{aligned} & O(N^2) + O(N) \\ & = O(N^2) \end{aligned} \tag{6.38}$$

となる．

6.1.4　計算量・通信量の評価結果の考察

以上の評価結果より，提案手法の平均計算量は $O(N^2 loglogN)$，平均通信量は $O(N^2)$ であることが分かった．そして，これらの計算量は既存のセキュア関数計算よりも大幅に増加しているわけではない．例えば，提案手法の平均計算量は $O(N^2 loglogN)$ であり，*secure set intersection* の $O(NloglogN)$ の平均計算量の N 倍となっている．また，提案手法の平均通信量は $O(N^2)$ であり，*secure k-nearest neighbor* の $O(N^2)$ の平均通信と同じである．このことから，データ規模が大きく無ければ，適切に並列化を行うことで提案手法を実際に動かすことが可能であると考える．

6.2　安全性の評価

本節では，提案手法の安全性について評価を行う．ここで安全であるとは，提案手法のプロトコルの通信内容から，想定されている以上の情報を得る事ができないことを言う．まず，機関 A,B 間のプロトコルの通信内容から得られる情報が，プロトコルの実行結果で

ある内部匿名テーブル $T_n^*(n \in A, B)$ と，途中計算の結果である 2 つの中間情報だけであることを証明する．次に，これらの情報はプライバシ上の問題が小さいことを示す.

6.2.1 安全性の定義と証明

機関 A,B のプロトコルの通信内容から，機関 A が機関 B の秘密の情報を得ることができないことを証明するには，機関 A が受信する通信内容 (機関 B が送信する通信内容) をシミュレートするシミュレータ S_A が存在し，S_A に対する入力として機関 A が機関 B との通信内容から得られると想定されている情報と機関 A が元々持っている情報を与え，S_A が通信内容をシミュレートできることを示せば良い [24, 21, 69]．なぜなら，S_A がシミュレートした通信内容には入力として与えられた情報以外の情報が一切含まれていないため，シミュレートされた通信内容を受信する機関 A は，S_A に入力された情報以外の情報を得ることができないからである．また，提案のプロトコルではセキュア計算を用いているため，Composition Theorem[21] を用いて証明を行う [24]．Composition Theorem とは，プロトコル F を安全なプロトコルブロック $f_1 \ldots f_n$ で構成できるとした時，$f_1 \ldots f_n$ を信頼できる第三者 (Trusted Third Party, TTP) を介した通信に置き換えたものが安全であれば F も安全であるという定理である．本証明では，まず提案手法のプロトコルのセキュア計算を TTP を介した通信に置き換えたプロトコルにおいて，プロトコルの通信内容をシミュレートできることを証明する．その後，Composition Theorem によって TTP を介した通信を各種セキュア計算で置き換えても，同様にプロトコルが安全であることを示す.

定理 1 機関 $n \in \{A, B\}$ は，提案手法のプロトコルの通信内容から，T_n^* と中間情報 *1,2* 以外の情報を知ることができない.

- **中間情報 *1***: 相手機関の分割点候補における，分割後グループのユーザ数と自機関のダミーユーザ数 *(*ユーザの *ID* は漏洩しない*)*
- **中間情報 *2***: キャンセルされた分割点における，分割点の分割機関，分割後グループのユーザ数と自機関のダミーユーザ数 *(*相手機関で分割をする場合*)*，分割点の属性と属性値 *(*自機関で分割をする場合*)*，満たされなかった指標 *(k*-匿名性 *or δ-site-presence)*

証明 1 機関 $n \in \{A, B\}$ が元々持つ T_n と T_n^* と中間情報 *1,2* から，シミュレータ S_n が，機関 n が受信する通信内容をシミュレートできる事を示せば良い．初めに，T_A と T_A^* と中間情報 *1,2* から，機関 A が受信する通信内容を S_A がシミュレートできることを示す．まずシミュレータ S_A は，T_A^* の各レコードの GID の値がシーケンシャルに割り当てられていることを利用して，分割を逆順に辿ることでどのような分割が行われたかを分析する．この分析では，T_A^* のうち GID の値が最も大きい *2* つのレコードが最後に行われた分割であり，この分割の分割前のレコードは歯抜けになっている GID のうち最も大きな値のレコードであると判断する *(*例：表 *4.1(e)* では *2* が歯抜けになっている最も大きな値であるため，「*2* ⇒ *4,5*」という分割が行われたことが判る*)*．そして，この *2* つのレコードを比較することで，分割が行われた機関と分割後の IDs を判断できる．また，分割が機関 A で行われた場合は分割点の属性と属性値も分かる *(*例：表 *4.1(e)* では「事業者 A」で「年収」「*400*」で分割され，*user1*〜*5* と *user6*〜*10* に分割されたことが分かる*)*．この処理を繰り返すことで，最初の分割まで辿ることが可能であり，どのような分割が行われたかを分析できる．ここでは，この分析結果を「分割情報」と呼ぶ．

次に，S_A は分析した分割情報を使って通信内容のシミュレートを開始する．提案手法では，*Step2* と *Step3* で通信が行われる．特に *Step2* では，分割点決定関数の計算と，指標確認と，ID の通知の際に通信を行う．最初に S_A は，*Step2* の *1* 回目の分割における機関 A が受信する情報をシミュレートする．*Step2* の分割点決定関数の計算で行われる通信内容 *(*図 *4.5)* は，以下の *4* つである．

- *(1)* 分割を行う機関 *(*双方の機関が受信*)*
- *(2)* 分割点候補における分割後のユーザ数
- *(3)* 自機関のダミーユーザ数 *(*非分割機関が受信*)*
- *(4)* 決定した分割点の属性と属性値 *(*分割機関が受信*)*

これらの情報のうち，*(1)* と *(4)* については分析した分割情報から情報を知ることができ，*(2)(3)* については中間情報 *1* から知ることができる．そしてシミュレータ S_A は，これらの情報から機関 A が受信する情報を抜き出してシミュレートを行う．続いて，分析した分割

情報から分割がさらに続くかを判断する．もし分割後の *GID* についてさらに分割が続く場合は，*Step2* の指標確認を *OK* としてシミュレートする．そして，この分割が機関 B で行われていた場合は，*Step2* の分割後のユーザ *ID* の通知をシミュレートする．その後，分割後の *GID* について上記処理を再帰的に繰り返す.

もし分割が続かない場合は，一度指標確認を *OK* としてシミュレートした後，中間情報 *1* と *2* を使って，先ほどと同様に分割点決定関数の計算で行われる通信内容をシミュレートする．但し先ほどとは違って，分析した分割情報に，*(1)(2)(3)(4)* の情報は含まれていないため，代わりに中間情報 *2* を利用する．そして，その後の指標確認では中間情報 *2* を使って k-匿名性か δ-*site-presence* を *NG* とするシミュレートを行う．このような処理を繰り返すことで，S_A は機関 A が受信する通信内容をシミュレートできる.

続いて，T_B と中間情報 *1,2* と T_B^* から，機関 B が受信する通信内容を，シミュレータ S_B がシミュレートできることを示す．*Step2* については，先ほどと同様にシミュレートできる．*Step3* の *userCount* については，T_B^* に情報があるためシミュレートできる.

以上のように，*TTP* を利用した場合の提案手法のプロトコルで，S_n は機関 $n \in \{A, B\}$ が持つ T_n と中間情報 *1,2* と T_n^* から，機関 n が受信する通信内容をシミュレートできるため，プロトコルの通信内容から T_n^* と中間情報 *1,2* 以外の情報の漏洩は無い．また *Composition Theorem* により，*TTP* による計算を各種セキュア計算に置き換えても提案手法のプロトコルは安全であると言える．よって，機関 n は提案手法のプロトコルの通信内容から，T_n^* と中間情報 *1,2* 以外の情報を知ることはできない. □

6.2.2 安全性の評価結果の考察

T_n^* は機関 n の相手機関がもつ属性は含まれない．また，中間情報 1,2 はユーザ ID が含まれないため，どのユーザの情報であるかを知ることはできない．よって，中間情報 1,2 や内部匿名テーブルの情報からのセンシティブ属性の属性値やユーザ存在情報の確定はないことから，プライバシ上の問題は低いと考える.

第7章　結論

本章では，本研究の課題，提案内容，評価内容についてまとめる．そして最後に，今後の課題について述べる．

7.1　まとめ

本節では，本論文が対象とした分散匿名化におけるユーザ存在情報の漏洩の課題，その課題を解決するための提案内容とその特徴，さらに評価内容と結果についてまとめる．

7.1.1　本研究の課題

本論文の課題は，分散匿名化手法を実際のアプリケーションに適用するために，既存の垂直分割データの分散匿名化が対象としている「(問題 1) 機関 C においてデータの個人が特定される問題」と「(問題 2) 機関 A,B において必要以上にデータを開示してしまう問題」だけでなく，双方の機関のユーザ存在情報が一致しない場合に発生する「(問題 3) 機関 A,B の双方に対してユーザ存在情報が漏洩してしまう問題」も解決することである．この問題 3 は，以下の 2 つに分割される．

問題 3-1 結合匿名テーブルによるユーザ存在情報の漏洩問題

問題 3-2 ユーザ ID 通知によるユーザ存在情報の漏洩問題

本論文では，これらの問題を解決し，ユーザ存在情報の漏洩を低減した分散匿名化手法を実現することを目的としている．

7.1.2　提案の内容と特徴

本論文では，問題 3-1 と問題 3-2 を解決するために以下のような提案を行った．

1. 問題 3-1 を解決するために，*δ-site-presence* という新たなプライバシ指標を提案した．

   δ-site-presence は，既存の集中型の匿名化におけるユーザ存在情報が知られる可能性を示した *δ-presence* [39] という指標を，分散匿名化のために拡張した指標である．この指標を用いることで，分散匿名化におけるユーザ存在情報が漏洩する可能性の許容範囲を示すことができる．

2. 問題 3-2 を解決しつつ *δ-site-presence* を満たした分散匿名化手法を実現するために，ダミーユーザを導入した新たな分散匿名化手法のプロトコルとして，ダミーユーザプロトコルを提案した．

   ダミーユーザプロトコルは，ダミーユーザによって，存在するユーザと存在しないユーザの区別を困難にできるプロトコルである．ダミーユーザプロトコルを用いることで，*δ-site-presence* を満たし，ユーザ存在情報の漏洩を軽減した分散匿名化を実現できる．

次に，これらの提案内容の特徴をまとめる．まず，*δ-site-presence* には以下のような特徴がある．

- *δ-site-presence* の特徴は，機関にユーザが存在するか否かを表現できる点である (4.1 節)．そのため，3 機関以上における分散匿名化において，あるユーザが他の機関に存在するか否かを示すこともできる．それに対し既存の *δ-presence* は，あるテーブルにユーザが存在するか否かまでしか表現できないため，3 機関以上における分散匿名化において，あるユーザが他の機関に存在するか否かを表現することは出来ない．

続いて，ダミーユーザプロトコルには以下のような特徴がある．

1. ダミーユーザプロトコルの第一の特徴は，ダミーユーザのダミー値を分割を行う度に補正している点である (4.2.1 節)．この補正を行うことにより，たとえ機関 A,B の準識別子の属性間に相関があったとしても，ある程度相関に沿ったダミー値を設定で

きる．これは，既存の摂動法 [4] のように単にランダムに値を設定する方法とは異なり，Top-down アプローチによる分散匿名化の手法に特化した方法である．

2. ダミーユーザプロトコルの第二の特徴は，分割点決定関数においてダミーユーザのエントロピーが高くなるような分割点が選ばれるようにしている点である (4.2.2 節)．これにより，既存の k-匿名性だけでなく，新たに δ-*site-presence* も満たしやすい分割点が選択され，結果として有用性の高い結合匿名テーブルの生成を可能としている．

3. ダミーユーザプロトコルの第三の特徴は，通信量・計算量と安全性のバランスを考慮してセキュア計算を組み合わせて設計している点である (4.2.3 節)．もし，既存の MPC を用いてプロトコルを設計すると，一切の情報を機関 A,B に漏らさずに分散匿名化を実現できるが，通信量・計算量が多くなりすぎてしまう．そこで，ダミーユーザプロトコルでは，プライバシ性の低い情報が漏洩することを許容する代わりに通信量・計算量を減らすような設計を行い，実際のサービスに適用を目指している．

7.1.3 評価の内容と結果

本論文では，δ-*site-presence* とダミーユーザプロトコルを用いた提案手法を評価するために，提案手法の有効性の評価と，ダミーユーザプロトコルの計算量・通信量と安全性の評価を行った．

まず有効性評価では，米国の国勢調査データと患者のレセプトデータを用いた評価を行った．その結果，ユーザ存在情報の漏えいを軽減しながらも，既存の分散匿名化手法よりも有用性の高い結合匿名テーブルが生成できることを確認した．また，提案手法と既存の集中型のユーザ存在隠蔽の匿名化手法との比較を行い，既存手法は既存手法とほぼ同等に有用な匿名化が行えることを確認した．さらに，複数の医療機関が保持する医療データを結合・分析する場面での利用を想定し，データ分析を行った際の集計誤差を計測した．結果，提案手法はユーザ存在情報の漏えいを軽減しながらも相対誤差 15%以下でデータ分析が可能であることがわかった．これは，近年言われている医療の効率化や医療サービスの質向上のための医学研究に適用できると考えられる．

また，ユーザ存在情報隠蔽の限界値の評価の結果では，提案手法において設定できる δ

の値は，理論上の限界値から約20%ほどの余裕を持たせた値であることが分かった．これは，例えば今回のレセプトデータを用いたユースケースにおいては，プライバシを保つことができるとされる$\delta_{max,B}$=0.1を十分満たすものであり，意義のあるユーザ存在情報の隠蔽が可能であることが分かった．さらに実行速度を計測した結果，提案手法は母集団ユーザ数が10000人以下で3日間程度で処理が可能であることが分かった．母集団ユーザ数が10000人以下で3日間程度でデータ提供が行えるというサービスはいくつか考えられるため，提案手法は実際のアプリケーションに適用可能であると考える．

そして，提案手法のプロトコルの安全性を暗号理論で用いられるシミュレータを用いた評価手法によって証明し，プライバシ性の高いパーソナル情報やユーザ存在情報が漏洩しないことを確認した．また，提案手法全体の計算量・通信量の評価を行い，双方の事業者が持つ情報を開示せずに単純な関数計算を行う既存のセキュア計算の計算量・通信量と比較した．その結果，提案手法全体の計算量・通信量が既存のセキュア計算の計算量・通信量よりも大幅に増加することは無いことを確認した．

以上のような評価結果から，本論文で提案する手法を用いることによって，国勢調査データや医療データにとどまらず，様々な種類のパーソナル情報をサービス事業者間で安全にデータ連携することができ，新たなサービスが創出されることが期待できる．

7.2 今後の課題

本節では，提案手法について今後解決するべき課題を述べる．

- データ量が増えた際の通信量と計算量の低減

  提案手法では，セキュア計算を組み合わせて用いることで通信量と計算量を軽減した手法ではあるが，既存のセキュア計算における問題と同様に，データ量が増えた場合は通信量と計算量が増加してしまう問題がある．今後は，より通信量と計算量が少ないセキュア計算を用いることや，適切に並列化を行う手法を用いることで，スケーラビリティを向上させる必要がある．

- 複数事業者への拡大

提案手法は2事業者に限定したプロトコルとなっている．今後より多くの利用場面に適用することを考えると，3事業者以上の複数事業者にも対応したプロトコルを検討する必要がある．本論文で提案している手法をそのまま3つ以上の事業者で用いることは出来ないのは，提案手法で用いているセキュア計算のいつくかは2機関限定となっているためである．しかし，3機関でも動作可能なセキュア計算の研究 [32] や，3つの機関以上の機関における分散匿名化手法 [37, 24] を参考にすることで，提案手法を3つ以上の機関に対応するように拡張可能であると考える．

- 有効性の向上

  提案プロトコルの分割点決定関数では集中型の手法より若干有効性が落ちる (5.4.3節の評価結果)．そのため，提案プロトコルの分割点決定関数やダミーユーザのダミー値の設定方法を改良し，より有用性の高い結合テーブルを生成可能にすることが望まれる．

謝辞

本研究の一部は，経産省の「平成 23 年度次世代高信頼・省エネ型ＩＴ基盤技術開発・実証事業 (レセプト情報等の利活用基盤の開発)」プロジェクトの成果である．

本研究にあたり，ご多忙の中適切なご指導をくださった大須賀 昭彦 教授，川村 隆浩 客員准教授，田原 康之准教授に感謝いたします．また，様々な協力をしてくださった大須賀・田原研究室の皆様に感謝の意を表します．そして，投稿した論文等に対して，国内外の多くの査読者から様々なコメントをいただきました．

審査を快く引き受けてくださいました大学院 情報システム学研究科の田中 健次 教授，小池 英樹 教授，大森 匡 教授に感謝申し上げます．先生方には，論文のまとめ方や技術の評価方法などに関して多大なご指導をいただきました．

本研究は日本電気株式会社において，多くの方々のご指導とご協力を得て行ったものに基づいています．日本電気株式会社に在籍のまま社会人博士課程への就学を許可いただき便宜を図って戴いた，情報・ナレッジ研究所 所長 野口誠氏，情報・ナレッジ研究所 部長 宮内幸司氏，情報・ナレッジ研究所 主任研究員 森拓也氏に心から感謝申し上げます．また研究活動を進める上でご指導をくださり，さまざまなご支援とご配慮を頂きました情報・ナレッジ研究所の先輩，同僚，後輩の方々，さらに本研究の初期段階において，ご在職中にご指導をくださった伊東直子氏に心から御礼申し上げます．

最後に，日々の研究活動を心身両面に渡って支えてくれた妻 真由美と娘 美優に心から感謝します．

参考文献

[1] Charu C. Aggarwal and Philip S. Yu. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer-Verlag, 2008.

[2] Gagan Aggarwal, Nina Mishra, and Benny Pinkas. Secure computation of the kth-ranked element. In Avdances in Cryptology - Proc. of Eurocyrpt ' 04, pp. 40–55. Springer-Verlag, 2004.

[3] Rakesh Agrawal, Alexandre Evfimievski, and Ramakrishnan Srikant. Information sharing across private databases. In *Proc. SIGMOD'03*, pp. 86–97. ACM, 2003.

[4] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *Proc. SIGMOD'00*, pp. 439–450. ACM, 2000.

[5] Gilad Asharov and Yehuda Lindell. A full proof of the bgw protocol for perfectly-secure multiparty computation. *IACR Cryptology ePrint Archive*, pp. 136–136, 2011.

[6] Roberto J. Bayardo and Rakesh Agrawal. Data privacy through optimal k-anonymization. In *Proc. ICDE'05*, pp. 217–228. IEEE Computer Society, 2005.

[7] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proc. STOC '88*, pp. 1–10. ACM, 1988.

[8] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 2001.

[9] C L Blake and C J Merz. Uci repository of machine learning databases., 1998.

[10] Chris Clifton. Privately computing a distributed knn classifier. In *In Proceedings of the Eighth European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2004.

[11] Emiliano De Cristofaro and Gene Tsudik. Experimenting with fast private set intersection. In *Proceedings of the 5th international conference on Trust and Trustworthy Computing*, TRUST'12, pp. 55–73. Springer-Verlag, 2012.

[12] Cynthia Dwork and Kobbi Nissim. Privacy-preserving datamining on vertically partitioned databases. In *In CRYPTO*, pp. 528–544. Springer, 2004.

[13] Apache Software Foundation. Apache thrift. `http://thrift.apache.org/`.

[14] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *Proc. EUROCRYPT'04*, pp. 1–19. Springer-Verlag, 2004.

[15] B.C.M. Fung, K. Wang, A.W.C. Fu, and P.S. Yu. *Privacy-Preserving Data Publishing: Concepts and Techniques*, chapter 11–12. CRC Press, 2010.

[16] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Comput. Surv.*, Vol. 42, No. 4, pp. 14:1–14:53, 2010.

[17] Benjamin C. M. Fung, Ke Wang, and Philip S. Yu. Top-down specialization for information and privacy preservation. In *Proc. ICDE'05*, pp. 205–216. IEEE Computer Society, 2005.

[18] Bart Goethals, Sven Laur, Helger Lipmaa, and Taneli Mielikainen. On private scalar product computation for privacy-preserving data mining. In *Proc. ICISC'04*, pp. 104–120. Springer-Verlag, 2004.

[19] O. Goldreich. Secure multi-party computation, working draft, version 1.3, 2001.

[20] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *Proc. STOC'87*, pp. 218–229. ACM, 1987.

[21] Oded Goldreich. *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press, 2004.

[22] Vijay S. Iyengar. Transforming data to satisfy privacy constraints. In *Proc. KDD '02*, pp. 279–288. ACM, 2002.

[23] Wei Jiang and Chris Clifton. Privacy-preserving distributed k-anonymity. In *Proc. DBSec'05*, pp. 166–177. Springer, 2005.

[24] Pawel Jurczyk and Li Xiong. Distributed anonymization: Achieving privacy for both data subjects and data providers. In *Proc. DBSec'09*, pp. 191–207. Springer, 2009.

[25] Murat Kantarcioglu and Chris Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. *IEEE Trans. on Knowl. and Data Eng.*, Vol. 16, No. 9, pp. 1026–1037, 2004.

[26] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Incognito: efficient full-domain k-anonymity. In *Proc. SIGMOD'05*, pp. 49–60. ACM, 2005.

[27] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Mondrian multidimensional k-anonymity. In *Proc. ICDE'06*, p. 25. IEEE, 2006.

[28] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Workload-aware anonymization. In *Proc. KDD'06*, pp. 277–286. ACM, 2006.

[29] Ninghui Li and Tiancheng Li. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Proc. ICDE07*, 2007.

[30] Y. Lindel and B. Pinkas. Privacy preserving data mining. *Journal of Cryptology*, Vol. 15, No. 3, pp. 177–206, 2002.

[31] Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. In *JOURNAL OF CRYPTOLOGY*, pp. 36–54. Springer-Verlag, 2000.

[32] Yehuda Lindell and Benny Pinkas. Secure multiparty computation for privacy-preserving data mining. *Journal of Privacy and Confidentiality*, Vol. 1, pp. 59–98, 2009.

[33] Kun Liu. Paillier's cryptosystem in java. `http://www.csee.umbc.edu/~kunliu1/research/Paillier.html`.

[34] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data*, Vol. 1, No. 1, 2007.

[35] Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. Fairplay a secure two-party computation system. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, pp. 20–20. USENIX Association, 2004.

[36] Shinya Miyakawa, Nobuyuki Saji, and Takuya Mori. Location l-diversity against multifarious inference attacks. In *SAINT*, pp. 1–10, 2012.

[37] N. Mohammed, B. C. M. Fung, K. Wang, and P. C. K. Hung. Privacy-preserving data mashup. In *Proc. EDBT'09*, pp. 228–239. ACM, 2009.

[38] Moni Naor and Benny Pinkas. Oblivious transfer and polynomial evaluation. In *Proc. STOC '99*, pp. 245–254. ACM, 1999.

[39] Mehmet Ercan Nergiz, Maurizio Atzori, and Chris Clifton. Hiding the presence of individuals from shared databases. In *Proc. SIGMOD'07*, pp. 665–676. ACM, 2007.

[40] OpenID Foundation. *OpenID Authentication 2.0*, 2007.

[41] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of the 17th international conference on Theory and appli-*

cation of cryptographic techniques, EUROCRYPT'99, pp. 223–238. Springer-Verlag, 1999.

[42] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. on Knowl. and Data Eng.*, Vol. 13, No. 6, pp. 1010–1027, 2001.

[43] Latanya Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, Vol. 10, pp. 557–570, 2002.

[44] Tsubasa Takahashi and Shinya Miyakawa. Cmoa: continuous moving object anonymization. In *IDEAS*, pp. 81–90, 2012.

[45] Takao Takenouchi, Naoyuki Okamoto, Takahiro Kawamura, Akihiko Ohsuga, and Mamoru Maekawa. Development of knowledge-filtering agent along with user context in ubiquitous environment. In *Proc. International Symposium on Ubiquitous Intelligence and Smart Worlds (UISW'05)*, pp. 71–80. Springer, 2005.

[46] U.S. National Archives and Records Administration. Standards for privacy of individually identifiable health information. *Federal Register*, Vol. 67, No. 157, pp. 53182–53273, 2002.

[47] K. Wang, B. C. M. Fung, and G. Dong. Integrating private databases for data analysis. In *Proc. of the 2005 IEEE International Conference on Intelligence and Security Informatics (ISI)*, Vol. 3495, pp. 171–182, 2005.

[48] Tom White. *Hadoop: The Definitive Guide, 3rd Edition*. O'Reilly Media/Yahoo Press, 2012.

[49] Raymond Chi-Wing Wong, Jiuyong Li, Ada Wai-Chee Fu, and Ke Wang. (a, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. In *Proc. KDD '06*, pp. 754–759. ACM, 2006.

[50] Xiaokui Xiao and Yufei Tao. m-invariance: Towards privacy preserving re-publication of dynamic datasets. In *Proc. SIGMOD'07*, pp. 689–700. ACM, 2007.

[51] Andrew C. Yao. Protocols for secure computations. In *Proc. SFCS'82*, pp. 160–164. IEEE Computer Society, 1982.

[52] Justin Zhan, Liwu Chang, and Stan Matwin. Privacy preserving k-nearest neighbor classification. *International Journal of Network Security*, Vol. 1, No. 1, pp. 46–51, 2005.

[53] 厚生労働省医薬品の安全対策における医療関係データベースの活用方策に関する懇談会. 電子化された医療情報データベースの活用による医薬品等の安全・安心に関する提言（日本のセンチネル・プロジェクト）について, 2010.

[54] 株式会社エヌ・ティ・ティ・ドコモ. モバイル空間統計の作成手順. `http://www.nttdocomo.co.jp/corporate/technology/rd/tech/main/mobile_spatial_statistics/how_to_produce/`.

[55] 経済産業省. 「平成 23 年度次世代高信頼・省エネ型ＩＴ基盤技術開発・実証事業 (レセプト情報等の利活用基盤の開発)」に係る補助事業者募集要領, 2011.

[56] 経済産業省. 平成 24 年 6 月 1 日ＩＴ融合フォーラム有識者会議 - 配付資料-「ＩＴ融合新産業の創出に向けて」, 2012.

[57] 五十嵐大, 千田浩司, 高橋克巳. k-匿名性の確率的指標への拡張とその適用例. コンピュータセキュリティシンポジウム 2009 (CSS2009) 論文集, pp. 763–768, 2009.

[58] 五十嵐大, 千田浩司, 高橋克巳. Pl-多様性 : 属性推定に対する再構築法のプライバシーの定量化. コンピュータセキュリティシンポジウム 2010 (CSS2010) 論文集, pp. 813–818, 2010.

[59] 厚生労働省. 医療等分野における情報の利活用と保護のための環境整備のあり方に関する報告書 (案) 平成 24 年 9 月 12 日, 2012.

[60] 厚生労働省. 第 2 回レセプト情報等の提供に関する有識者会議（22 年 10 月 28 日）資料 1, 2012.

[61] 厚生労働省. 平成 23 年 (2011) 医療施設（静態・動態）調査・病院報告の概況, 2012.

[62] 千田浩司, 五十嵐大, 濱田浩気, 高橋克巳. エラー検出可能な軽量 3 パーティ秘匿関数計算の提案と実装評価. 情報処理学会論文誌, Vol. 52, No. 9, pp. 2674–2685, 2011.

[63] 菊池浩明. データマイニングと個人情報保護. 情報科学技術フォーラム FIT2004 講演論文集, 2004.

[64] 高橋翼, 宮川伸也, 伊東直子. 移動軌跡ストリームに対するリアルタイム k-匿名化手法の提案. 日本データベース学会論文誌, Vol. 10, No. 1, pp. 37–42, 2011.

[65] 社会保険診療報酬支払基金. レセプト電算処理システム. http://www.ssk.or.jp/rezept/index.html.

[66] 佐久間淳, 高橋克巳. クラウドストレージにおける個人情報の利活用とプライバシー保護. 情報処理, Vol. 52, No. 6, pp. 706–715, 2011.

[67] 佐久間淳, 小林重信. プライバシを保護した内積比較プロトコルの提案. 電子情報通信学会技術研究報告. ISEC, 情報セキュリティ, Vol. 106, No. 176, pp. 27–34, 2006.

[68] 佐久間淳, 小林重信. プライバシ保護データマイニング. 人工知能学会誌, Vol. 24, No. 2, pp. 283–294, 2009.

[69] 有田正剛. 知識の証明と暗号技術 (代数系アルゴリズムと言語および計算理論). 数理解析研究所講究録, Vol. 1655, pp. 75–95, 2009.

[70] 独立行政法人統計センター. 匿名データ利用の手引（学術研究・高等教育目的関係）. http://www.nstac.go.jp/services/anonymity.html.

[71] 野島良, 門林雄基. On the security and the performance of a practical two-party set-intersection protocol. 暗号と情報セキュリティシンポジウム SCIS 2008, 2008.

[72] 竹之内隆夫, 岡本直之, 川村隆浩, 大須賀昭彦, 前川守. ユビキタス環境において動的なコンテクストに応じて知識情報をフィルタリングする推論エージェントの開発. 電子情報通信学会論文誌 D-I, Vol. 88, No. 9, pp. 1428–1437, 2005.

[73] 岡本龍明, 太田和夫. 暗号・ゼロ知識証明・数論. 共立出版, 1995.

研究業績

学術雑誌

1. **竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦: ユーザ存在の特定を困難にした分散匿名化の提案～２診療機関のレセプトデータを用いた有効性の評価～, 電子情報通信学会論文誌 D, Vol.J96-D, No.3, 2013 年 3 月.（採録決定）

2. **竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦 : クラウド上での事業者間データ連携のための分散型パーソナル情報保護エージェント, 情報処理学会論文誌, Vol.53, No.11, pp.2432-2444, 2012 年 11 月.

3. **竹之内 隆夫**, 岡本 直之, 川村 隆浩, 大須賀 昭彦, 前川 守: ユビキタス環境において動的なコンテキストに応じて知識情報をフィルタリングする推論エージェントの開発, 電子情報通信学会論文誌 (D-I)，Vol. J88-D-I, No. 9, pp. 1428-1437, 2005 年 9 月.

- **Takao Takenouchi**, Naoyuki Okamoto, Takahiro Kawamura, Akihiko Ohsuga , Mamoru Maekawa: Development of knowledge-filtering agent along with user context in ubiquitous environment Systems and Computers in Japan, Systems and Computers in Japan, John Wiley & Sons, Inc., Vol.38, No.8, pp.11-19, April 2007. (3 の論文の英訳版)

4. 岡本 直之, **竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦, 前川 守: 放送番組に対してパブリックオピニオン・メタデータを生成する視聴支援エージェントの開発 ～ネットコミュニティからの雰囲気成分の抽出とユーザ間での流通による洗練化～，電子情報通信学会 (D-I)，Vol.J88-D-I, No.9, pp.1477-1486, 2005 年 9 月.

国際会議

5. **Takao Takenouchi**, Takahiro Kawamura, Akihiko Ohsuga: Hiding of User Presence for Privacy Preserving Data Mining, Proceedings of 3rd IIAI International Conference on e-Services and Knowledge Management (ESKM 2012), pp.133-138, IEEE, September 2012.

6. **Takao Takenouchi**, Takahiro Kawamura, Akihiko Ohsuga: Distributed Data Federation without Disclosure of User Existence, Proceedings of 26th Annual WG 11.3 Conference on Data and Applications Security and Privacy (DBSec 2012), pp.282-297, Springer, July 2012.

7. **Takao Takenouchi**, Takahiro Kawamura, Akihiko Ohsuga: Development of Knowledge-filtering Agent along with User Context in Ubiquitous Environment, Proceedings of 2nd International Symposium on Ubiquitous Intelligence and Smart Worlds (UISW2005), pp.71-80, Springer, December 2005.

査読付国内シンポジウム

8. **竹之内 隆夫**, 伊東 直子, 川村 隆浩, 大須賀 昭彦: クラウド上での事業者間データ連携のための分散型パーソナル情報保護エージェント，合同エージェントワークショップ＆シンポジウム 2011（JAWS2011）論文集, 2011 年 10 月.

9. **竹之内 隆夫**, 南澤 岳明, 伊東 直子: 既知ユーザ攻撃によるユーザ情報の漏洩リスクを低減した条件マッチング連係方式, マルチメディア，分散，協調とモバイル (DICOMO2011) シンポジウム論文集, pp.528-535, 2011 年 7 月.

10. **竹之内 隆夫**, 岡本 直之, 川村 隆浩, 大須賀 昭彦, 前川 守: ユビキタス環境における過渡的な知識を考慮した意思決定支援エージェントの開発, 合同エージェントワークショップ＆シンポジウム 2004(JAWS2004) 論文集, 2004 年 11 月.

11. 岡本 直之, **竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦, 前川 守: 仮想パブリックビューイングを実現する視聴支援エージェントの開発, 合同エージェントワークショップ＆シンポジウム 2004(JAWS2004) 論文集, 2004 年 11 月.

国内大会・研究会

12. **竹之内 隆夫**，側高 幸治，豊田 由起，高橋 翼，森 拓也: 部分データセットとの突合に対する耐性を有するレセプト匿名化方式，第 32 回医療情報学連合大会論文集, 2-F-2-4, pp.778-781, 2012 年 11 月.

13. 側高 幸治，高橋 翼，豊田 由起，**竹之内 隆夫**，森 拓也，興梠 貴英: レセプト匿名化システムの実証と評価，第 32 回医療情報学連合大会論文集, 2-F-2-1, pp.766-769, 2012 年 11 月.

14. 高橋 翼，側高 幸治，豊田 由起，**竹之内 隆夫**，森 拓也，興梠 貴英: 患者識別子の突合による匿名性破綻を防ぐレセプト匿名化方式，第 32 回医療情報学連合大会論文集, 2-F-2-2, pp.770-773, 2012 年 11 月.

15. 豊田 由起，側高 幸治，高橋 翼，**竹之内 隆夫**，森 拓也，興梠 貴英: 制約と優先度を考慮したレセプト匿名化方式，第 32 回医療情報学連合大会論文集, 2-F-2-3, pp.774-777, 2012 年 11 月.

16. **竹之内 隆夫**，川村 隆浩，大須賀 昭彦: ユーザ存在／不在確率の範囲を限定した分散匿名化手法と医療データによる評価, コンピュータセキュリティシンポジウム 2012(CSS2012) 論文集, 2D3-3, 2012 年 10 月.

17. **竹之内 隆夫**，川村 隆浩，大須賀 昭彦: プライバシ保護データマイニングのための分散匿名化プロトコルの提案，2012 年度人工知能学会全国大会（第 26 回）論文集, 3I2-OS-20-6, 2012 年 6 月.

18. 高橋 翼, **竹之内 隆夫**, 側高 幸治: 時系列データに対する ℓ-多様化方式の提案, 第 4 回データ工学と情報マネジメントに関するフォーラム (DEIM Forum 2012), A7-6, 2012 年 3 月.

19. **竹之内 隆夫**, 豊田 由起, 南澤 岳明: ユーザ情報漏洩リスクを軽減した条件マッチング連係, 電子情報通信学会総合大会講演論文集 2010, B-7-98, 2010 年 3 月.

20. 豊田 由起, **竹之内 隆夫**, 南澤 岳明: 条件マッチング連係におけるユーザ情報漏洩リスク軽減の改善手法, 電子情報通信学会総合大会講演論文集 2010, B-7-99, 2010 年 3 月.

21. **竹之内 隆夫**, 南澤 岳明: マルチレイヤ仮名通信の仮名 ID 管理機能の実現, 電子情報通信学会総合大会講演論文集 2009, B-7-35, 2009 年 3 月.

22. 南澤 岳明, 蔦澤 奈津子, **竹之内 隆夫**, 伊東 直子, 吉田 万貴子: 匿名性と利便性を両立したコミュニケーションサービスの提供, 電子情報通信学会総合大会講演論文集 2009, B-7-36, 2009 年 3 月.

23. **竹之内 隆夫**, 蔦澤 奈津子, 南沢 岳明, 伊東 直子, 吉田万 貴子: 仮名 ID の生成・管理機能の提案, 電子情報通信学会総合大会講演論文集 2008, B-7-38, 2008 年 3 月.

24. 蔦澤 奈津子, 南沢 岳明, **竹之内 隆夫**, 伊東 直子: コミュニケーションにおける匿名 ID ライフサイクル管理, 電子情報通信学会総合大会講演論文集 2008, B-7-37, 2008 年 3 月.

25. 渡部 正文, 伊東 直子, **竹之内 隆夫**, 吉田 万貴子: プライバシーに配慮したコンテキスト開示制御, 電子情報通信学会総合大会講演論文集 2008, B-20-46, 2008 年 3 月.

26. 渡部 正文, **竹之内 隆夫**, 伊東 直子: ユーザ情報を活用したサービス制御基盤の検討: コンテキストに基づいたユーザ情報通知ポリシの実装と評価, 電子情報通信学会 情報ネットワーク研究会 (IN) , Vol.106 No.118 pp.49-54, 2006 年 6 月.

27. **竹之内 隆夫**, 岡本 直之, 川村 隆浩, 大須賀 昭彦, 前川 守: ユビキタス環境において動的なコンテキストに追従して知識情報をフィルタリングする推論エージェントの開発, 情報処理学会ユビキタスコンピューティングシステム研究会 (UBI) 研究報告, 2005-UBI-7(Vol.2005, No.28), pp.217-224, 2005 年 3 月

28. 岡本 直之, **竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦, 前川 守: 放送番組に対してパブリックオピニオン・メタデータを生成する視聴支援エージェントの開発，電子情報通信学会第 2 回 Web インテリジェンスとインタラクション研究会 (WI2)，電子情報通信学会技術研究報告, WI2-2005-23, 2005 年 3 月.

書籍（分担執筆）

- SQuBOK 策定部会: ソフトウェア品質知識体系ガイド—SQuBOK Guide 改訂版, オーム社, 2013 年度発行予定. (プライバシ保護の節を担当)

登録特許

- **Takao Takenouchi**, Naoko Ito: COMMUNICATION TERMINAL DEVICE, COMMUNICATION SYSTEM, RELAYING-DEVICE SELECTING DEVICE, COMMUNICATION METHOD, AND PROGRAM, 米国特許 8,284,710, 登録 2012 年 10 月.

関連論文の印刷公表の方法及び時期

学術雑誌

1. 全著者名：**竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦
   論文題目：ユーザ存在の特定を困難にした分散匿名化の提案～２診療機関のレセプトデータを用いた有効性の評価～
   印刷公表の方法及び時期：電子情報通信学会論文誌 D, Vol.J96-D, No.3, 2013 年 3 月（採録決定済）
   （第 4 章及び第 5 章及び第 6 章と関連）

2. 全著者名：**竹之内 隆夫**, 川村 隆浩, 大須賀 昭彦
   論文題目：クラウド上での事業者間データ連携のための分散型パーソナル情報保護エージェント
   印刷公表の方法及び時期：情報処理学会論文誌, Vol.53, No.11, 2012 年 11 月
   （第 4 章及び第 5 章及び第 6 章と関連）

3. 全著者名：**竹之内 隆夫**, 岡本 直之, 川村 隆浩, 大須賀 昭彦, 前川 守
   論文題目：ユビキタス環境において動的なコンテキストに応じて知識情報をフィルタリングする推論エージェントの開発
   印刷公表の方法及び時期：電子情報通信学会 (D-I)，Vol. J88-D-I, No. 9, pp. 1428-1437, 2005 年 9 月
   （第 4 章と関連）

国際会議

4. 全著者名：**Takao Takenouchi**, Takahiro Kawamura, and Akihiko Ohsuga
   論文題目：Hiding of User Presence for Privacy Preserving Data Mining
   印刷公表の方法及び時期：Proceedings of 3rd IIAI International Conference on e-Services and Knowledge Management (ESKM 2012), pp.133-138, IEEE, September 2012
   （第 4 章及び第 5 章と関連）

5. 全著者名：**Takao Takenouchi**, Takahiro Kawamura, and Akihiko Ohsuga
   論文題目：Distributed Data Federation without Disclosure of User Existence
   印刷公表の方法及び時期：Proceedings of 26th Annual WG 11.3 Conference on Data and Applications Security and Privacy (DBSec 2012), pp.282-297, Springer, July 2012
   （第 4 章及び第 6 章と関連）

6. 全著者名：**Takao Takenouchi**, Takahiro Kawamura, and Akihiko Ohsuga
   論文題目：Development of Knowledge-filtering Agent along with User Context in Ubiquitous Environment
   印刷公表の方法及び時期：Proceedings of 2nd International Symposium on Ubiquitous Intelligence and Smart Worlds (UISW2005), pp.71-80, Springer, December 2005
   （第 4 章と関連）

本論文との関連の詳細

章	節		関連論文番号	関連する内容
3 章			2	分散匿名化におけるユーザ存在情報の漏洩課題の定義
4 章	4.1 節		4	δ-*site-presence* の提案
		4.1.1〜4.1.2 節	1	δ-*site-presence* の設定指針と拡張
		4.1.3 節	2	δ-*max-site-presence* の提案
	4.2 節	4.2.1〜4.2.2 節	2	ダミーユーザの提案
		4.2.3〜4.2.4 節	5	セキュア計算を利用したプロトコルの提案
	4.3 節		3	大量データのフィルタリング処理の提案
			6	フィルタリング処理の詳細な評価
5 章	5.4〜5.5 節		2	国勢調査データの DM を用いた評価
			4	国勢調査データのクエリ結果相対誤差を用いた評価
			1	レセプトデータのクエリ結果相対誤差を用いた評価
	5.7 節		2	ダミーユーザの偏りの評価
6 章	6.1 節		1	計算量・通信量の評価
	6.2 節		2	MPC を用いた場合の安全性の評価
			5	セキュア計算を用いた場合の安全性の評価

著者略歴

竹之内 隆夫（たけのうち たかお）

1980 年 5 月 11 日	新潟県柏崎市に生まれる
1999 年 3 月	新潟県立柏崎高等学校 卒業
1999 年 4 月	国立電気通信大学 電気通信学部 情報工学科 入学
2003 年 3 月	国立電気通信大学 電気通信学部 情報工学科 卒業
2003 年 4 月	国立電気通信大学 大学院 情報システム学研究科 情報システム設計学専攻 博士前期課程 入学
2005 年 3 月	国立電気通信大学 大学院 情報システム学研究科 情報システム設計学専攻 博士前期課程 修了
2005 年 4 月	日本電気株式会社 入社
2011 年 4 月	国立大学法人 電気通信大学 大学院 情報システム学研究科 社会知能情報学専攻 博士後期課程 入学
2013 年 3 月	国立大学法人 電気通信大学 大学院 情報システム学研究科 社会知能情報学専攻 博士後期課程 修了予定