Vol.2016-CSEC-73 No.20
Vol.2016-IOT-33 No.20
2016/5/27

秘密計算による分散医療統計システムの実装評価

濱田 浩気[1,a)]　木村 映善[2]　菊池 亮[1]　千田 浩司[1]　岡本 和也[3]　真鍋 史朗[4]　黒田 知宏[3]
松村 泰志[4]　武田 理宏[4]　三原 直樹[4]

概要：医療分野では異なる組織の臨床リポジトリを横断的に使った分析が試みられている．これまでプライバシー保護のために集約者を設置して情報開示を必要最低限にする工夫が行われてきたが，集約者を信頼することが必要であった．これに対しては秘密計算を用いることで集約者の権限を分散することができるが，複数のリポジトリへの攻撃や統計値からのデータ漏洩への対策は行われていなかった．我々は集約者を必要とせず，自身以外のすべての組織が結託しても自身のデータを守ることができる秘密計算を用いて，臨床研究で重要な統計計算を実装した．さらに，秘密計算上で統計的開示制御を実現し，統計値の開示の際にデータが漏えいするリスクについても対処した．地理的に離れた 3 組織間で実装したシステムの評価実験を行い，実用的な性能であることを確認した．

1. はじめに

生体情報や臨床データなど様々な情報を統合して新たな知見を得ようという動きが盛んになってきている．特に，それぞれの病院に蓄積されている患者データである臨床リポジトリを横断的に使うことによる，地域を跨いだ多数の症例を使った分析への期待は大きい．しかしながら臨床リポジトリの第三者への開示については，プライバシー侵害への懸念から慎重な意見があり進んでいない．臨床データなどの個人に関する機微な情報を利用する際のプライバシー侵害の懸念は大きく 2 つある．1 つめは分析結果から入力データの情報が推測できてしまうこと，2 つめは分析中に入力データが見えてしまうこと，である．

1 つめの懸念への対策は出力のプライバシー保護と呼ばれる．医療情報に関して多組織の情報を集約して出力のプライバシー保護を試みている取り組みに SHRINE [17] がある．SHRINE では集約者と呼ばれる主体がクエリに応じて必要な情報を参加組織から集め，統計的開示制御 (SDC) という仕組みでクエリ結果からのプライバシー侵害の恐れがないかどうかを判断し，問題がなければ利用者にクエリ結果を返す．この仕組みで出力のプライバシー保護が達成されるが，集約者にはデータが見えてしまうため，2 つめの懸念が残る．

2 つめの懸念への対策は入力のプライバシー保護と呼ばれる．入力のプライバシー保護を目指した技術としては，秘密計算と呼ばれる暗号学に基づいた手法がある．秘密計算はデータを暗号化などの方法で秘匿化したまま一度も元のデータに戻すことなく任意の計算を行うことを可能にする技術である．特に，結託耐性のある秘密計算を使うことで多組織の情報を集約した分析を入力のプライバシー保護をしたまま実現できる．しかしながら秘密計算には通常の計算機上での計算に比べて処理速度が非常に遅いという問題があり，Yao による基本的なアイディア [18] の発表から 30 年以上に渡り，処理速度を改善する研究が行われている．秘密計算で実現されている処理は限定的であり，出力のプライバシー保護など実用的な処理も行われていない．

1.1 貢献

本稿では，臨床リポジトリなどの機微な情報の組織横断の分析を可能にすることを目指し，秘密計算に基づく入力と出力のプライバシー保護を実現する分散医療統計システムを設計し，実装評価する．評価実験は提案システムを用いて地理的に離れた 3 組織間で臨床リポジトリを使用して行う．

評価実験により，結託に強い秘密計算に基づいた実用性

[1] NTT セキュアプラットフォーム研究所
NTT Secure Platform Laboratories, 3-9-11, Midori-cho, Musashino-shi, Tokyo 180–8585, Japan
[2] 愛媛大学
Ehime University, 454, Situkawa, Toon-shi, Ehime 791–0295, Japan
[3] 京都大学
Kyoto University, 54 Shogoin Kawahara-cho, Sakyo-ku, Kyoto 606–8507, Japan
[4] 大阪大学
Osaka University, 2-15, Yamada-oka, Suita-shi, Osaka 565–0871, Japan
[a)] hamada.koki@lab.ntt.co.jp

の高い計算が現実的な時間で正確に行えることを確認する．さらに，提案システム上で統計的開示制御の仕組みも実現し，結託に強い秘密計算を用いることで入力と出力のプライバシー保護を両立した実用的なシステムが現実的であることを示す．

1.2 関連研究

1.2.1 データ加工に基づくプライバシー保護

秘密計算とは別のプライバシー保護のアプローチとして，プライバシ侵害が起こらなくなる程度までデータを加工した上で分析を行う手法がある．k-匿名化 [14] や再構築法 [1] が代表的である．これらの手法は秘密計算に比べると計算コストが非常に小さい．また，データを非可逆な方法で加工することによりプライバシ保護を実現するため，加工後のデータを公開可能であるという特長を持つ．一方，その非可逆性から元のデータに対して行った場合と完全に同等の分析結果を得ることは不可能であり，加工後のデータを用いた分析結果の精度をいかにして担保するかが課題となっている．

1.2.2 秘密計算の実装と実験

秘密計算の実装例は FairPlayMP[2] や Sharemind[3]，VIFF[9]，SEPIA[5]，MEVAL[19][23]，などがあるが，まだ少ない．実験を行った例はさらに少なく，テンサイ (サトウダイコン) のオークション [4] や，ネットワークの異常検知 [5]，成人白血病の臨床研究 [20]，ゲノム分析 [11] が報告されている．

2. 秘密計算

2.1 結託に強いマルチパーティ計算

本稿で提案する秘密計算システムは Damgård らによる結託に強いマルチパーティ計算 [7] を使う．Damgård らのマルチパーティ計算は，N 者の加法的秘密分散に基づく秘密計算で，能動的な攻撃者による N 者のうちのどの $N-1$ 者の結託に対しても安全であるという特徴を持つ．この秘密計算は入力を受け取る前に事前計算を行うオフラインフェーズと入力を受け取った後に実際に所望の計算を行うオンラインフェーズの 2 つの段階に分けて動作する．オフラインフェーズは入力に非依存のため，事前計算を十分に行なっておけば，ユーザーに影響のある時間はオンラインフェーズの時間に限られる．そのため，本稿ではこの方式のオンラインフェーズの性能にのみ注目することとし，オフラインフェーズで作られる乱数はあらかじめ N 者に共有されたものとする．

2.2 秘密計算上の演算

2.2.1 秘密分散，復元，加算，乗算，シャッフル

秘密分散，復元，加算，乗算，シャッフルの各処理はは

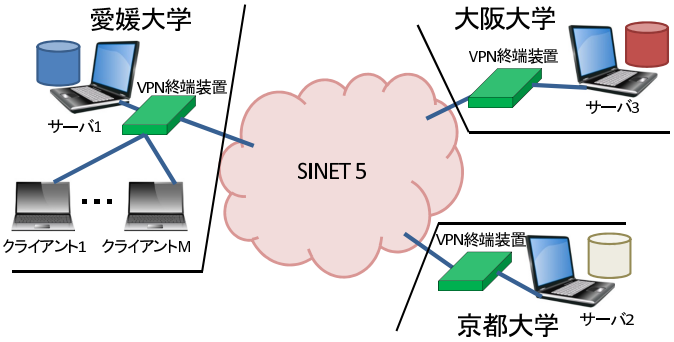

図 1 システム構成図

Damgård らの手法 [7] を使う．

2.2.2 ビット分解

ビット分解は Damgård らの手法 [6] を秘密分散，復元，加算，乗算の各演算の組み合わせにより実現する．

2.2.3 比較 ($<$，$=$，$\neq$)

比較の各演算は Nishide と Ohta の手法 [12] を秘密分散，復元，加算，乗算の各演算の組み合わせにより実現する．

2.2.4 ソート

ソートは Hamada らの手法 [10] を秘密分散，復元，加算，乗算，シャッフル，比較の各演算の組み合わせにより実現する．

2.2.5 浮動小数点数

浮動小数点数は，符号，指数，仮数それぞれに体の要素を一つ割り当てた三つ組で表現．浮動小数点数の秘密分散，復元，加算，乗算，除算，総和，整数からの変換の各演算を秘密分散，復元，加算，乗算，比較，ビット分解の各演算の組み合わせにより実現する．

3. システム

3.1 構成

本稿で提案する分散医療統計システムはマルチパーティ計算を行う N 台の秘密計算サーバと，M 台のクライアントにより構成される (図 1)．N 台の秘密計算サーバはどの 2 者も直接通信できるフルメッシュ型 VPN でつながっている．クライアントは N 台すべての秘密計算サーバとの間に通信路を持つ．

分析対象のデータは N 台の秘密計算サーバがそれぞれ保有し，患者ごとのデータを各行に，属性ごとのデータを各列にとったあらかじめ決められた行列の形をしたフォーマットである．

ユーザが分析を行う際には，クライアント端末から演算の種類と計算対象のデータを指定する．指定された情報は各秘密計算サーバに伝えられ，各秘密計算サーバは必要なデータを抽出して秘密分散し，秘密計算を実行する．秘密計算が完了すると各秘密計算サーバからクライアントに秘密分散のシェアが送られ，クライアントはこれを復元する

ことにより計算結果を得る．

各サーバはクエリに必要な値だけを秘密分散するため，例えば Pearson の相関係数のように計算の大半を個別の組織内で行える演算は非常に効率よく実現できる．

3.2 提供する分析

秘密計算の提供する基本演算の組み合わせにより，実用上重要な分析を実現した．提供する分析は統計的開示制御 (SDC) 機能付きの頻度，総和，平均，分散，最小値，最大値，中央値の各統計量の計算，Pearson の相関係数，Spearman の順位相関係数，一元配置分散分析，Kruskal-Wallis 検定，である．また，対象とするデータは条件式で指定することができる．

3.3 統計的開示制御 (SDC)

提案システムでは出力からの患者及び大学病院のプライバシー侵害を防止するため，しきい値ルールと占有ルールと呼ばれる統計的開示制御を秘密計算で実現した．

しきい値ルールは統計値を計算する際に，対象となるデータの度数が，あらかじめ設定しているしきい値 t を下回る場合に統計値を出力しないというルールである．本システムではデータを提供する施設が出力を見た場合 (データのうち 1 施設分を知っている者が出力を見た場合) にもこの基準を満たすよう，しきい値ルールが満たされる条件を，N 施設のうち，どの $N-1$ 施設の提供するデータの度数の和も t 以上であること，とする．

占有ルールは総和の計算を行う際に，m 施設の提供する値の総和が全体の総和の k%以上を占める場合に統計値を出力しないというルールである．本システムではデータを提供する施設が出力を見た場合にもこの基準を満たすよう，占有ルールが満たされる条件を，N 施設のうち，すべての $N-1$ 施設 S とすべての m 施設 $X \subset C$ に対して (X の総和)/(S の総和) $\leq k/100$ であること，とする．

4. 評価実験

臨床リポジトリを使った分析を想定したシナリオを設定し，これに従って秘密計算で分析を行った．正確性と高速性の 2 つの観点から実用性を評価した．

4.1 シナリオ

提案システムの有効性の検証のため，HbA1c と腎機能の関係を見るシナリオを設定した．腎機能の指標としては，連続値である血清 Cr 値と推定糸球体ろ過量 (eGFR)，非連続値である尿たんぱくと慢性腎臓病 (CKD) の重症度，を使用した．

使用した患者データは 2012 年 4 月 1 日から 2014 年 3 月 31 日に取得されたデータのうち，30 日以内にすべての項目が取得されている 20 才以上の患者のものとした．使用した患者データの項目は，

- 年齢
- 性別
- HbA1c
- 尿たんぱく質
- 血清クレアチニン値 (Cr)
- eGFR
- CKD 重症度

の 7 項目であり，このうち年齢，性別，HbA1c，尿タンパク質，血清クレアチニン値 (Cr) は各大学の EMR から抽出した．eGFR は年齢，性別，Cr から，CKD 重症度は eGFR と尿たんぱくから，それぞれ計算した．

患者は年齢による 3 群への分類 (全体，65 才以上，20 才以上 65 才未満) と性別による 3 群への分類 (全体，男性，女性) の組み合わせによる 9 群に分けて評価した．また，eGFR は 15 未満，$[15, 30)$，$[30, 45)$，$[45, 60)$，$[60, 90)$，90 以上の 6 区間に，HbA1c は 6.2 未満，$[6.2, 6.9)$，$[6.9, 7.4)$，$[7.4, 8.4)$，8.4 以上の 5 区間に，それぞれ分類して評価した．

4.2 計算対象

評価実験では，実際の患者データを使ったシナリオに沿った計算と，性能測定のためのランダムに生成したデータを入力とする各処理の計算を行う．

4.2.1 シナリオに沿った計算

4.2.1.1 基本統計量

SDC 付き基本統計量の計算の検証として，Cr に関して頻度，総和，平均，分散，最小値，最大値，中央値の各基本統計量を eGFR と HbA1c の区間の組み合わせ 30 通りについて計算した．

4.2.1.2 多群の検定

HbA1c と腎機能を表す指標に関係があるかどうかを見るため，HbA1c の各区間によって分けた 5 群に対して一元配置分散分析および Kruskal-Wallis 検定を尿たんぱく，CKD 重症度，eGFR のそれぞれについて行った．

4.2.1.3 相関係数

HbA1c と腎機能を表す指標の関係の強弱を測るため，HbA1c と eGFR の Pearson 相関係数と HbA1c と尿たんぱくの Spearman 順位相関係数を計算した．

4.2.1.4 正解の計算

計算結果の正確性の検証のため，各大学病院のデータを集約して代表的な統計ソフトウェアである R [15] および SAS [13] で秘密計算と同様の計算を行った．

4.2.2 ランダムな入力による性能測定

入力の大きさによる処理時間の違いを見るため，ランダムな入力に対する秘密計算の提供する各演算の処理時間を測定した．計算対象は，秘密分散，復元，乗算，ビッ

ト分解，比較 (定数との $<$，$<$，$=$，$\neq$)，最大値，最小値，シャッフル，ソート，浮動小数点数の演算 (秘密分散，復元，整数からの変換，加算，乗算，除算，総和)，Pearson の相関係数，Spearman の順位相関係数，一元配置分散分析，Kruskal-Wallis 検定，である．

4.3 実験環境

秘密計算サーバは愛媛大学病院，大阪大学病院，京都大学病院の 3 箇所に設置した．すべてのサーバは SINET 5 (Science Information NETwork 5) [21] のレイヤー 2 ネットワーク上にフルメッシュ型 VPN を構築指定，相互に接続した IPSec ルータ FITELnet F2000 を使って IPSec による VPN を構築した．秘密計算サーバとして使用した計算機は，CPU が Intel Core i5 2540M 2.60GHz (2 コア)，RAM が 8GB，SSD が 128GB (TOSHIBA THNSNC12)，OS が Ubuntu 12.04 である．サーバプログラムは C++ で作成され，コンパイラは g++ 4.6.3 を使用した．ネットワークのパフォーマンスの評価には iperf と ping コマンドを使った．

統計的開示制御のパラメータは，以下のように設定した．しきい値ルールのパラメータは $t = 10$ とした．これは，米国政府の Department of Health and Human Services のエージェント CMS[8] など国内外の公的統計で一般的に用いられる基準のうち，最も厳しいものが $t = 10$ であったためである．占有ルールのパラメータは $m = 1, k = 80$ とした．瀧による文献 [22] では，「k は大きい値たとえば $k = 80$」と例示されている．最近ではフィンランドのパーソナルデータ保護のガイドライン [16] において $k = 80$ の例示がある．

秘密計算の有限体の位数は $p = 2^{61} - 1$ とした．

4.4 測定結果

4.4.1 ネットワーク性能

VPN 上での愛媛，京都，大阪の 3 病院の各 2 病院間での通信帯域と遅延 (ラウンドトリップタイム) の測定を行った．

表 1 VPN 上でのサーバ間の通信帯域 (Mbps)

	単一	同時
愛媛から京都	599.6 Mbps	183.5 Mbps
京都から愛媛	603.8 Mbps	213.3 Mbps
京都から大阪	601.2 Mbps	146.5 Mbps
大阪から京都	652.2 Mbps	269.5 Mbps
大阪から愛媛	567.0 Mbps	232.8 Mbps
愛媛から大阪	520.8 Mbps	162.7 Mbps

通信帯域の測定には，iperf コマンドを用いた．測定は (1) 各通信が単一で行われる理想的な場合と，(2) 実際の秘密計算の実行時に近い，複数の通信が同時に行われる場合の 2 つの場合で測定した．測定結果を表 1 に示す．表中の各値は大学 A から大学 B へ TCP で 10 秒間送信を行った場合の平均転送量で，単位は Mbps である．表 1 で単一と書いた列は各通信を一つずつ行った場合の通信帯域であり，各 5 回測定した平均である．表 1 で同時と書いた列は 6 方向すべての通信を同時に行った場合の通信帯域である．開始タイミングのずれを考慮し，6 方向すべての通信が行われている状況での通信帯域を測定するため，20 回連続して実行したうちの 6 回目から 15 回目までを対象として，10 回の平均を求めた．

遅延 (ラウンドトリップタイム) の測定には，ping コマンドを用いた．測定は (1) 理想的な ping コマンドしか実行しない場合，(2) 実際の秘密計算実行中に近い場合，(3) 実際より厳しいと思われる通信帯域を限界まで使っている場合，の 3 つの場合で測定した．測定結果を表 2 に示す．表中の各値は大学 A から大学 B へ行って A へ戻るラウンドトリップタイムで，単位はミリ秒である．それぞれ 100 回実行した結果の最小値，平均，最大値，平均偏差 (平均との差の絶対値の平均) を求めた．表 2 で負荷なしと書いた部分は ping コマンドのみを実行した場合の測定値である．秘密計算中と書いた部分は，秘密計算で全患者データを対象に Spearman の順位相関係数を計算中に実行した場合の測定値である．帯域全使用と書いた部分は，通信帯域測定で同時と書いた部分の測定と同様に，iperf コマンドで 6 方向の送信を同時実行中に実行した場合の測定値である．

4.4.2 ディスク読み込み速度

提案システムではあらかじめディスクに書き込んだ乱数を使用して秘密計算を実行する．ディスクからの乱数の読み込みの時間を測定した．愛媛大学に設置したサーバ上で 48×10^7 バイトのデータの読み込みを 3 回行い，読み込み速度の平均は 440.944 MB/秒であった．

4.4.3 秘密計算の処理時間

まず，患者データを用いた際の秘密計算での処理時間を測定した．処理時間は一般に処理対象となるデータの量に依存する．そのため，Pearson の相関係数，Spearman 順位相関係数，一元配置分散分析，Krsukal-Wallis 検定については 33,552 件の全患者データを対象とする場合を，SDC 付きの頻度，総和，平均，分散，最小値，最大値，中央値の各統計量については該当する患者数が 11,183 人と最も多くなった eGFR が 60 以上 90 未満かつ HbA1c が 6.2 未満である場合を，それぞれ対象として処理時間を測定した．各 5 回の測定を行い，その平均を求めた．測定結果を表 3 に示す．

処理時間は送信処理の時間 (表 3 の送信)，受信処理の時間 (表 3 の受信)，ディスク読み込み処理の時間 (表 3 のディスク)，その他サーバ内での計算の時間 (表 3 のローカル) の 4 項目に分類してそれぞれ集計した．ただし，提案システムの実装は送信部分及び乱数読み込み部分を別ス

表 2 VPN 上でのサーバ間の遅延 (ラウンドトリップタイム)．単位はミリ秒．

	負荷なし				秘密計算中				帯域全使用			
	最小	平均	最大	平均偏差	最小	平均	最大	平均偏差	最小	平均	最大	平均偏差
愛媛から京都	7.687	7.881	8.054	0.115	7.472	10.512	18.821	3.027	9.332	16.623	23.693	3.143
京都から愛媛	7.617	7.802	7.945	0.121	7.538	10.355	17.780	3.104	7.889	16.109	27.100	3.618
京都から大阪	3.332	3.857	27.860	2.452	3.243	3.899	7.231	0.687	5.577	12.452	23.286	3.487
大阪から京都	3.336	3.837	39.101	3.545	3.218	3.881	6.679	0.591	7.772	13.154	21.159	3.004
大阪から愛媛	7.508	7.793	9.270	0.295	7.360	10.416	16.994	2.824	8.554	15.817	27.125	4.454
愛媛から大阪	7.499	7.714	8.591	0.209	7.322	10.930	25.383	3.566	7.882	16.176	34.845	5.253

表 3 患者データに対する処理時間とその内訳，各演算の通信ラウンド，全体で送信 (受信) を行ったデータ量，各サーバが使用するディスク上の乱数．

		処理時間内訳 (秒)				通信	全体の	各サーバの
	処理時間 (秒)	ローカル	送信	受信	ディスク	ラウンド	送信 (受信) 量	ディスク読込み
Pearson の相関係数	12.068	0.551	0.041	11.232	0.244	2550	13.928 MB	10.592 MB
Spearman の順位相関係数	619.432	141.519	7.976	463.537	6.399	8971	38.525 GB	29.750 GB
一元配置分散分析	18.393	1.140	0.090	17.154	0.009	3790	42.304 MB	32.094 MB
Kruskal-Wallis 検定	406.647	85.308	4.802	311.662	4.875	8037	23.433 GB	17.723 GB
頻度	1.114	0.005	0.001	0.882	0.225	139	0.204 MB	0.160 MB
総和	1.432	0.020	0.003	1.409	0.001	234	0.608 MB	0.479 MB
平均	6.757	0.352	0.027	6.374	0.003	1494	2.780 MB	2.097 MB
分散	10.142	0.500	0.041	9.596	0.005	2216	7.562 MB	5.746 MB
最小値	1.282	0.025	0.004	1.252	0.001	260	0.340 MB	0.267 MB
最大値	1.302	0.027	0.004	1.270	0.001	260	0.340 MB	0.267 MB
中央値	98.678	22.138	1.369	74.359	0.812	2044	5.395 GB	4.179 GB

レッドで実行しているため，送信処理は送信を行うスレッドへデータのデータの受け渡し，ディスク読み込み処理は乱数をディスクから読み込むスレッドへのデータの受け渡し，にそれぞれ要した時間である．

また，基本演算を含めた各演算について，入力の大きさの違いによる処理時間の変化を見るため，入力の大きさを 10，10^2，10^3，10^4 と変えてランダムな入力に対する処理時間を測定した．測定結果を表 4 に示す．測定は各演算について入力の大きさごとにそれぞれ 5 回行い，その平均を測定結果とした．

4.4.4 計算結果の正確性

表 5 統計ソフトウェアごとの出力の桁数．ただし，R は有効数字，SAS は小数点以下の桁数．

	R	SAS
Pearson の相関係数	7 桁	5 桁
Spearman の順位相関係数	7 桁	5 桁
一元配置分散分析	5 桁	2 桁
Kruskal-Wallis 検定	5 桁	4 桁
平均	7 桁	6 桁
分散	7 桁 または 8 桁	6 桁
その他	全桁	全桁

患者データを使用したすべての計算結果を R および SAS で計算した値と比較を行った結果，すべての値で一致することが確認できた．比較は R および SAS と出力の桁が一致するように秘密計算による計算結果を四捨五入して行った．R と SAS の出力の桁数を表 5 に示す．

また，SDC による統計値の開示制御が正しく動作していることも確認できた．今回の患者データでは eGFR が 90 以上かつ HbA1c が 8.4 以上の場合に度数が $t = 10$ を下回る 2 病院が存在するためしきい値ルールが適用されるはずであったが，秘密計算で統計値を計算した結果，eGFR が 90 以上かつ HbA1c が 8.4 以上の場合に出力が抑制された．

4.5 考察

4.5.1 正確性

検証を行ったすべての計算結果について，標準的な統計ソフトウェアである R や SAS の結果と完全に一致しており，本稿で提案した分散医療統計システムの正確性が確認できた．また，統計的開示制御についても，患者データの範囲で正しく動作していることが確認できた．

4.5.2 高速性

今回のシナリオで必要とされた計算については，最も時間のかかる全体に対する Spearman の順位相関係数の計算でも約 11 分で完了しており，実用的な時間で統計分析が行えたと言える．

4.5.3 ボトルネックの分析

より多数の患者データを対象とした統計分析を行うためには，さらなる高速化が必要である．本節では提案システムのボトルネックを調べる．

表 4 ランダムな入力を使った各演算の時間 (秒)．n は入力の数で，最大値，最小値，シャッフル，ソート，浮動小数点数総和では入力のベクトルの大きさを，Pearson の相関係数，Spearman の順位相関係数，一元配置分散分析，Kruskal-Wallis 検定では各サーバの持つデータ数を，他の単項 (二項) 演算では同時に実行する入力 (入力の対) の数をそれぞれ表す．“-” は未測定であることを示す．

	$n = 10$	$n = 10^2$	$n = 10^3$	$n = 10^4$
秘密分散	0.056	0.057	0.061	0.113
復元	0.089	0.091	0.092	0.114
乗算	0.016	0.016	0.017	0.026
ビット分解	0.271	0.742	1.833	13.320
定数との比較 (<)	0.075	0.111	0.515	4.052
比較 (<)	0.471	0.750	1.197	9.487
比較 (=)	0.106	0.116	0.184	0.619
比較 (≠)	0.144	0.161	0.230	0.673
最大値	0.885	1.557	3.232	10.423
最小値	0.921	1.561	3.415	9.724
シャッフル	0.648	0.762	1.215	9.261
ソート	1.563	4.465	10.444	81.323
浮動小数点数秘密分散	0.050	0.053	0.065	0.174
浮動小数点数復元	0.091	0.101	0.109	0.124
整数 → 浮動小数点数変換	0.526	1.156	2.934	21.415
浮動小数点数加算	1.691	2.237	12.115	85.457
浮動小数点数乗算	1.107	2.329	6.956	54.180
浮動小数点数除算	6.065	10.756	46.001	348.617
浮動小数点数総和	1.917	3.252	7.027	37.610
Pearson の相関係数	11.978	12.251	11.516	11.380
Spearman の順位相関係数	13.959	23.513	64.723	-
一元配置分散分析 (5 群)	20.032	18.961	18.467	18.819
Kruskal-Wallis 検定 (2 群)	19.804	28.298	83.991	-
Kruskal-Wallis 検定 (3 群)	21.018	31.279	122.938	-
Kruskal-Wallis 検定 (4 群)	22.536	37.038	139.703	-
Kruskal-Wallis 検定 (5 群)	23.399	36.279	175.482	-

全体の処理時間を決定づける要素としては，ローカルの計算，送信 (受信)，通信遅延，ディスク読み込み，の 4 つがある．これらは互いに独立したリソースを要する処理であるので，依存関係を無視して理想的に並列化することで，これらのうち最も時間がかかる処理の時間が全体の処理時間となる．提案システムの実装では，高速化のため送信とディスク読み込みを別スレッドにて行うようにしており，残るは受信部分の並列化が必要である．表 3 の結果を見ると，Pearson の相関係数と頻度のディスク読み込みを除いていずれも全体の 2%以下と非常に小さくなっており，並列化の効果は確認できる．Pearson の相関係数と頻度については，起動直後に実行されたため，ディスク読み込み用スレッドのキャッシュ準備と実行が重なり，ディスク読み込み処理の時間が本来よりも大きくなったと考えられる．

理想的に並列化された場合のボトルネックを，送受信データ量と送受信回数の関係が異なる代表的な 2 つの演算について分析する．

4.5.3.1 一元配置分散分析のボトルネック

一元配置分散分析は送受信データ量に対して送受信回数が非常に多い演算である．全体の送信量が 42.304 MB，ディスク読み込み必要量が各サーバで 32.094 MB に対し，通信ラウンドは 3790 である．秘密計算の送信 (受信) の帯域を表 1 の同時の場合の総和である 1208.3 Mbps, 遅延を表 2 の秘密計算中の場合の平均の平均の半分である 4.166 ミリ秒，ディスク読み込みの速度を測定結果の 440.944 MB/秒とすると，理想的に実装された場合の処理時間は送信 (受信) に 0.28 秒，ディスク読み込みに 0.073 秒，通信遅延に 15.789 秒がそれぞれ見込まれる．現状ではローカルの計算に 1.140 秒 を要しており，それぞれが完全に並列に行われたとすると，通信遅延が次いで大きいローカルの計算の 13.85 倍と圧倒的に大きく，全体の高速化のためには，サーバ間遅延の短縮か，通信ラウンドの削減が有効である．

4.5.3.2 Spearman の順位相関係数のボトルネック

Spearman の順位相関係数は送受信回数に対して送受信データ量が非常に大きい演算である．全体の送信量が 38.525 GB，ディスク読み込み必要量が各サーバで 29.750 GB に対し，通信ラウンドは 8971 である．帯域を 1208.3 Mbps，遅延を 4.166 ミリ秒，ディスク読み込みの速度を

440.944 MB/秒とすると，秘密計算で理想的に実装された場合の処理時間は送信 (受信) に 261.191 秒，ディスク読み込みに 69.088 秒，通信遅延に 37.373 秒がそれぞれ見込まれる．ローカルの計算に 141.519 秒を要しており，それぞれが完全に並列に行われたとすると，送信 (受信) に要する 261.191 秒が最大である．従って，全体の高速化のためには通信帯域の拡大や送信 (受信) 量の削減が有効である．

5. おわりに

本研究では医療分野での利用を想定し，集約者を必要とせずに異なる組織のデータを横断的に使った分析を可能にする秘密計算システムを提案し，実装評価を行った．提案したシステムは高い結託耐性を持つとともに，統計的開示制御を秘密計算上で実現して計算結果からの情報漏えいリスクにも対処した．

愛媛大学，京都大学，大阪大学にそれぞれ秘密計算サーバを設置して 33,552 人の患者データを用いて実装したシステムの評価実験を行った．臨床リポジトリを使った分析を想定したシナリオを設定し，統計ソフトウェア R や SAS との比較により計算結果が正確であること，最も時間のかかる Spearman の順位相関係数でも計算時間が 11 分と実用的な処理性能であること，を確認した．

謝辞 本研究は愛媛大学医学部倫理委員会 (承認番号: 愛大医病倫 1501009 号)，京都大学医学部倫理委員会，大阪大学医学部倫理委員会の承認を得て行われた．

参考文献

[1] Agrawal, R. and Srikant, R.: Privacy-Preserving Data Mining, *SIGMOD Conference* (Chen, W., Naughton, J. F. and Bernstein, P. A., eds.), ACM, pp. 439–450 (2000).
[2] Ben-David, A., Nisan, N. and Pinkas, B.: FairplayMP: a system for secure multi-party computation, *ACM Conference on Computer and Communications Security* (Ning, P., Syverson, P. F. and Jha, S., eds.), ACM, pp. 257–266 (2008).
[3] Bogdanov, D., Laur, S. and Willemson, J.: Sharemind: A Framework for Fast Privacy-Preserving Computations, *ESORICS* (Jajodia, S. and López, J., eds.), LNCS, Vol. 5283, Springer, pp. 192–206 (2008).
[4] Bogetoft, P., Christensen, D. L., Damgård, I., Geisler, M., Jakobsen, T. P., Krøigaard, M., Nielsen, J. D., Nielsen, J. B., Nielsen, K., Pagter, J., Schwartzbach, M. I. and Toft, T.: Secure Multiparty Computation Goes Live, *Financial Cryptography* (Dingledine, R. and Golle, P., eds.), LNCS, Vol. 5628, Springer, pp. 325–343 (2009).
[5] Burkhart, M., Strasser, M., Many, D. and Dimitropoulos, X. A.: SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics, *USENIX Security Symposium*, USENIX Association, pp. 223–240 (2010).
[6] Damgård, I., Fitzi, M., Kiltz, E., Nielsen, J. B. and Toft, T.: Unconditionally Secure Constant-Rounds Multi-party Computation for Equality, Comparison, Bits and Exponentiation, *TCC*, pp. 285–304 (2006).
[7] Damgård, I., Keller, M., Larraia, E., Pastro, V., Scholl, P. and Smart, N. P.: Practical Covertly Secure MPC for Dishonest Majority - Or: Breaking the SPDZ Limits, *Computer Security - ESORICS 2013 - 18th European Symposium on Research in Computer Security, Egham, UK, September 9-13, 2013. Proceedings* (Crampton, J., Jajodia, S. and Mayes, K., eds.), Lecture Notes in Computer Science, Vol. 8134, Springer, pp. 1–18 (online), DOI: 10.1007/978-3-642-40203-6 (2013).
[8] DEPARTMENT OF HEALTH AND HUMAN SERVICES CENTERS FOR MEDICARE & MEDICAID SERVICES: INSTRUCTIONS FOR COMPLETING THE DATA USE AGREEMENT (DUA) FORM CMS-R-0235 (2006).
[9] Geisler, M.: Cryptographic Protocols: Theory and Implementation, PhD Thesis, University of Aarhus (2010).
[10] Hamada, K., Kikuchi, R., Ikarashi, D., Chida, K. and Takahashi, K.: Practically Efficient Multi-party Sorting Protocols from Comparison Sort Algorithms, *ICISC* (Kwon, T., Lee, M.-K. and Kwon, D., eds.), LNCS, Vol. 7839, Springer, pp. 202–216 (2012).
[11] Kamm, L., Bogdanov, D., Laur, S. and Vilo, J.: A new way to protect privacy in large-scale genome-wide association studies, *Bioinformatics*, Vol. 29, No. 7, pp. 886–893 (online), DOI: 10.1093/bioinformatics/btt066 (2013).
[12] Nishide, T. and Ohta, K.: Multiparty Computation for Interval, Equality, and Comparison Without Bit-Decomposition Protocol, *PKC*, pp. 343–360 (2007).
[13] SAS Institute Inc: SAS 9.4 Software — SAS.
[14] Sweeney, L.: k-Anonymity: A Model for Protecting Privacy, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 10, No. 5, pp. 557–570 (2002).
[15] The R Foundation: R: The R Project for Statistical Computing.
[16] Tilastokeskus Statistikcentralen: Guidelines on the protection of tabulated personal data (2013).
[17] Weber, G. M., Murphy, S. N., McMurry, A. J., MacFadden, D., Nigrin, D. J., Churchill, S. and Kohane, I. S.: The Shared Health Research Information Network (SHRINE): a prototype federated query tool for clinical data repositories, *Journal of the American Medical Informatics Association*, Vol. 16, No. 5, pp. 624–630 (2009).
[18] Yao, A. C.-C.: How to Generate and Exchange Secrets (Extended Abstract), *FOCS*, pp. 162–167 (1986).
[19] 濱田浩気，五十嵐大，菊池　亮，千田浩司，諸橋玄武，冨士　仁，高橋克巳：実用的な速度で統計分析が可能な秘密計算システム MEVAL，コンピュータセキュリティシンポジウム 2013 論文集，pp. 1–8 (2013).
[20] 濱田浩気，大竹茂樹，五十嵐大，竹之内大地，千田浩司，冨士　仁，高橋克巳，村田節子，熊田総佳：秘匿関数計算システムによる医療データのプライバシ保護統計分析，信学技報，LOIS2011-102, Vol. 111, pp. 177–181 (2012).
[21] 国立情報学研究所：学術情報ネットワーク (SINET 5、サイネット・ファイブ).
[22] 瀧　敦弘：集計表におけるセル秘匿問題とその研究動向，統計数理，Vol. 51, No. 2, pp. 337–350 (2003).
[23] 菊池　亮，五十嵐大，濱田浩気，千田浩司：改ざん検知機能付きの実用的な秘密計算システム MEVAL2，*SCIS*, pp. 1–8 (2015).