ARG WI2 No.8, 2016
WI2-2016-15

セキュアマルチパーティ秘密計算法によるベクトル量子化の実現

宮島 洋文[†,a]　重井 徳貴[††,b]　宮島 廣美[††,c]　宮西 洋太郎[†††,d]
北上 眞二[††††,e]　白鳥 則郎[†††,f]

† 長崎大学医歯薬学総合研究科　†† 鹿児島大学理工学研究科　††† (株) アイエスイーエム　†††† 早稲田大学

a) *k3768085@kadai.jp*　b) *shigei@eee.kagoshima-u.ac.jp*　c) *miya@eee.kagoshima-u.ac.jp*　d) *miyanisi@jade.dti.ne.jp*　e) *shinji@meltec.co.jp*　f) *norio@shiratori.riec.tohoku.ac.jp*

概要
秘匿データの分散処理方式としてセキュアマルチパーティ秘密計算法 (Secure Multi-party Computation : SMC) が知られている. 特に, データマイニングのようにデータの秘匿性が強く望まれる分野においては, 有望と考えられている. 特に, 個々のデータ自身を分割して記憶する新しい SMC 法は有効性が高いと期待されている. しかしながら, この分野の研究は始まったばかりであり, データマイニングのように, 複雑なデータ処理を必要とする分野への応用は行われていない. 本稿では, データマイニングの計算技術の一つとして知られる k-means と NG 法について, SMC による計算法を提案し, その有効性を数値シミュレーションで検証する.

キーワード　セキュアマルチパーティ秘密計算法, ベクトル量子化, k-means, ニューラルガス

1　はじめに

データマイニングに関して, データを秘匿したまま処理を実現する研究が行われている [1]. これらに関する多くの研究は, 匿名性や暗号を用いてデータ処理を実現する研究であり, いずれも秘匿データを集中管理する方法である. 一方, データを秘匿したまま分散管理する方式として, SMC に関する研究が行われている. 特に, 個々の秘匿データを分散して記憶や計算処理を実行する研究が注目されている [2, 3]. しかしながら, この分野の研究は始まったばかりであり, データマイニングのような複雑な計算処理についての研究はほとんど行われていない.

本稿では, データマイニングの計算技術の一つとして知られる k-means とニューラルガス (Neural Gas : NG) 法 [4] について, SMC による計算法を提案し, その有効性を数値シミュレーションで検証する.

2　予備概念

2.1　SMC

ここでは, 本稿で用いられる SMC のデータ表現法と計算方法の概略を説明する. 図 1 のようなクライアントと m 個のサーバ (パーティ) からなるモデルを使って, 1 個の実数データの処理について説明する.

はじめに, 1 つの実数データ x が m 個のデータに分解される. k 番目のサーバには, データ x の一部である x^k が送られる. ここに, $x = \sum_{k=1}^{m} x^k$ である. k 番目のサーバでは $f_k(x^k)$ を計算し, クライアント側に結果を送る. クライアント側では, 結果 $f_k(x^k)$ を加算して最終結果 $\sum_{k=1}^{m} f_k(x^k)$ を得る. 1 回の処理で結果が得られない場合は, この過程が複数回繰り返される. 問題は, どのような問題であればこのような計算処理 (部分的な計算の和が全体の結果, すなわち $f(x) = \sum_{k=1}^{m} f_k(x^k)$ となる) が可能かということになる.

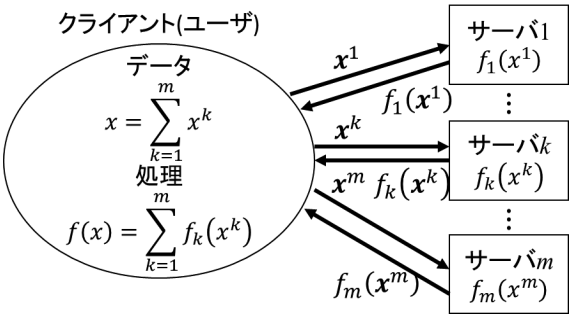

図 1 SMC の提案モデル.

本稿では, データマイニングの基本的な手法である k-means や NG 法について, このような計算処理が可能であることを示す. また, この処理過程においては, サーバ側では部分的な情報しか知る事ができず, データの秘匿性が保持される.

2.2　ニューラルガス法

多くのデータを少数のデータで近似または, 特徴抽出する方法をベクトル量子化といい, データマイニングの基本的計算法として知られている. ベクトル量子化には多くの方法が知られているが, k-means や NG はその代表的な手法である [4]. 以下では, NG とその特別な場合である k-means 法について説明する. すべてのデータからなるベクトル集合 $\boldsymbol{X}$ を少数の参照ベクトル集合 W で近似する問題を考える. はじめに, ベクトル $\boldsymbol{x}\in\boldsymbol{X}$ に対して, $\boldsymbol{W}$ の各要素 $\boldsymbol{w}^i$ が $\boldsymbol{x}$ に何番目に近いかを与える近傍ランク $e_i(\boldsymbol{x}, \boldsymbol{w}^i)$ をベクトル間のユークリッド距離を使って定義する. この近傍ランクを使って, $\boldsymbol{W}$ の各

要素 $\boldsymbol{w}^i$ を以下の式により更新する.

$$\triangle \boldsymbol{w}^i = \varepsilon \cdot h_\lambda \left(e_i(\boldsymbol{x}, \boldsymbol{w}^i)\right) \cdot (\boldsymbol{x} - \boldsymbol{w}^i) \quad (1)$$

$$h_\lambda(e_i(\boldsymbol{x}, \boldsymbol{w}^i)) = \exp\left(-e_i(\boldsymbol{x}, \boldsymbol{w}^i)/\lambda\right) \quad (2)$$

ここに, $\varepsilon \in [0, 1]$ かつ λ は正の実数である. この式は, 入力 $\boldsymbol{x}$ に近い $\boldsymbol{W}$ の要素ほど $\boldsymbol{x}$ に大きく近づけることを意味する. 与えられた $\boldsymbol{x}$ に対して, 近傍ランクを求め, 更新を繰り返す事により, 集合 $\boldsymbol{W}$ が $\boldsymbol{X}$ を近似することを実現する.

また, $\lambda \to 0$ である場合が k-means 法である. すなわち, 入力 $\boldsymbol{x}$ に最も近い $\boldsymbol{W}$ の要素のみを $\boldsymbol{x}$ に近づけることを意味する. 与えられた $\boldsymbol{x}$ に対して, , 最も近い $\boldsymbol{W}$ の要素に対して更新を繰り返す事により, 集合 $\boldsymbol{W}$ が $\boldsymbol{X}$ を近似することを実現する.

3 SMC を用いた NG 法の実現

NG を実現する SMC システム (プロトコル) を提案する. k-means 法はその特別な場合として実現できる. NG を実現するには, 1)$\boldsymbol{W}$ の各要素の $\boldsymbol{x}$ に対する近傍ランクの決定と, 2) 参照ベクトルの更新を行う 2 つのステップを分散処理方式により実現する. 初期条件としては, 入力 $\boldsymbol{x}$ と参照ベクトル $\boldsymbol{W}$ の各要素は, サーバに記憶されているとする. ここに, $\boldsymbol{x} = (x_1, \cdots, x_p, \cdots, x_n)$, $x_p^i = \sum_{k=1}^m (x_p^i)^k$, $\boldsymbol{w}^i = (w_1^i, \cdots, w_p^i, \cdots, w_n^i)$, $w_p^i = \sum_{k=1}^m (w_p^i)^k$ である.

1) 入力 $\boldsymbol{x}$ に対する参照ベクトル $\boldsymbol{W}$ の各要素 $\boldsymbol{w}^i$ の近傍ランク $e_i(\boldsymbol{x}, \boldsymbol{w}^i)$ を計算する. この過程は, 以下の式に従って各サーバの差分 D を求め, これをクライアント側で統合する事により, ランクを決定する.

$$D_p^k = \left((x_p^i)^k - (w_p^i)^k\right) \ (p = 1, \cdots, n) \quad (3)$$

$$||\boldsymbol{x}^q - \boldsymbol{w}^i||^2 = \sum_{p=1}^{n} \left(\sum_{k=1}^{m} D_p^k \right)^2 \quad (4)$$

2)$\boldsymbol{x}$ に対する $\boldsymbol{W}$ の各要素の更新量を以下の式を使って更新する.

$$\triangle \left(w_p^i\right)^k = \frac{\partial E}{\partial (w_p^i)^k} = \frac{\partial E}{\partial w_p^i} \frac{\partial w_p^i}{\partial (w_p^i)^k}$$

$$= \varepsilon \cdot h_\lambda \left(e_i(\boldsymbol{x}, \boldsymbol{w}^i)\right) \cdot (x_i - w_p^i) \quad (5)$$

$$h_\lambda(e_i(\boldsymbol{x}, \boldsymbol{w}^i)) = \exp\left(-e_i(\boldsymbol{x}, \boldsymbol{w}^i)/\lambda\right) \quad (6)$$

3) 近似誤差が十分に小さいか, または決められた回数だけ 1) と 2) を繰り返す.

この計算法の正当性は, 式 (5) より明らかである.

4 数値実験

ここでは, 提案手法が従来手法と比較して十分な精度を実現できることを示すために, Iris, Wine, Sonar, BCW[5] の 4 種類のベンチマークデータを, 従来の k-means とニューラルガス法, 本稿で提案を行った SMC における k-means とニューラルガス法により分類を行う. ここで, Iris, Wine, Sonar, BCW のデータ数はそれぞれ 150, 178, 208, 683, クラス数は Iris, Wine が 3, Sonar と BCW は 2 である. 表 1 に各データに対する誤分類率 (%) を示す. ここで, 表中の各値は 5-fold 交差検定に対する 20 回試行の平均である. 表 1 において, 提案手法はベクトル量子化の従来手法と同等の誤分類率 (精度) を示している. また, データを格納するサーバの数 (m) が 3 個の場合と 10 個の場合, どちらも同程度の誤分類率を示している.

表 1 データの誤分類率 (%)

		Iris	Wine	Sonar	BCW
従来法	k-means	11.3	19.7	45.6	3.9
	NG	6.7	10.6	44.7	10.1
提案手法 (k-means)	($m = 3$)	12.8	15.1	46.6	3.9
	($m = 10$)	15.4	12.1	45.4	4.0
提案手法 (NG)	($m = 3$)	6.8	10.8	44.7	10.1
	($m = 10$)	5.6	11.2	44.7	10.1

5 結論

本稿では, 秘匿データに対してベクトル量子化を実現する SMC システムの提案を行った. また, 提案手法をデータの分類問題に適用し, 提案手法が従来手法と同等の精度を実現すること, および, データを格納するサーバの数が増えても十分な精度を実現することを示した.

今後の課題として, 本手法の改善や他のデータマイニング手法と組み合わせた手法への SMC システムの提案を行う.

参考文献

[1] C. C. Aggarwal et al., "Privacy-Preserving Data Mining: Models and Algorithms", ISBN 978-0-387-70991-8, Springer-Verlag, 2009.

[2] Y. Miyanishi et al., "New Methods to Ensure Security to Increase User's Sense of Safety in Cloud Services", Proc. of The 14th IEEE Int. Conference on Scalable Computing and Communications (ScalCom-2014), pp.859-865, Bali, Dec.2014.

[3] H. Miyajima et al., "A Proposal of Back Propagation Learning for Secure Multi-Party Computation Methods", Proc. of the IMECS, Vol I, pp.381-386, 2016.

[4] T. M. Martinetz et al., "Neural Gas Network for Vector Quantization and Its Application to Time-series Prediction", IEEE Trans. Neural Network, Vol.4, No.4, pp.558-569, 1993.

[5] UCI Repository of Machine Learning Databases and Domain Theories, ftp://ftp.ics.uci.edu/pub/machine learning-Databases.