SCIS 2017 2017 Symposium on Cryptography and Information Security
Naha, Japan, Jan. 24 - 27, 2017
The Institute of Electronics, Information and Communication Engineers

商品の特徴による再識別リスクとクラスタリングを用いた購買履歴データ匿名加工手法の提案

Proposal on Clustering based Anonymization Method for Transaction Data against Risk of Re-identification using Sets of Item

原田 玲央 * 伊藤 聡志 * 菊池 浩明 *
Reo Harada Satoshi Ito Hiroaki Kikuchi

あらまし 個人情報保護法の改正により新たに匿名加工情報が定義され, 匿名加工情報の実用化に向けた技術開発を試みる匿名加工・再識別コンテスト (PWSCUP2016) が 2016 年 10 月に開催された. 本コンテストを通じて, 商品集合の特徴をもとに再識別を行う手法を加工購買履歴データに適用して評価し, 高い精度で個人が特定できるリスクが存在することを明らかにした. これに対する加工手法の一つとして疑似レコードの追加があげられる. 本稿では, 購入商品の集合において, 顧客をクラスタリングし, クラスタ内で購入商品の特徴から個人を識別することができない様に疑似データを追加する新たな匿名加工手法を提案する. 提案手法をコンテストで使用された購買履歴データに適用して, その有用性と安全性を評価する.

キーワード 匿名加工, 再識別, 購買履歴データ, クラスタリング, TF-IDF

1 はじめに

インターネット技術やセンサ技術の進歩により多種多様なサービスから膨大な量のデータが生成されている. 企業は膨大なデータから新たな価値を見出すために, 購買履歴データや位置情報データといった個人に紐づく情報 (パーソナルデータ) を利活用しようと試みている. しかし, 一企業の枠を超えて社会全体でパーソナルデータを活用していくためには, 個人のプライバシ侵害に繋がらないように配慮し, 個人情報を適切に加工しなければならない.

2015 年 9 月の個人情報保護法の改正により, 匿名加工情報という新たな枠組みが定義された. それに伴い, 2016 年 10 月に安全で有用性の高い匿名加工技術の開発促進を目的に, 第二回匿名加工・再識別コンテスト PWSCUP2016 が開催された [1].

我々は, 本コンテストに参加し, 最も有用性の高い匿名加工データを最も正確に再識別した. 我々は, 顧客ごとの購入商品の集合に固有の特徴を有していることに注目し, 商品集合による再識別アルゴリズムを導入した. 本稿では, そのアルゴリズムを述べ, それによる再識別リスクを明らかにする.

この商品集合による再識別の問題に対する安全な加工方法について考える. ナイーブな方法として, 顧客毎の商品集合に個別の差が生じない様に, 購買履歴に疑似レコードを追加することが考えられる. 追加するレコードが多すぎると加工データの有用性を損なうので, 顧客集合を商品集合についていくつかのクラスタに分類し, 各クラスタ内で疑似レコードを追加すればよい. しかしながら, 購買履歴は商品数が多く高次元のため, 扱うには工夫が必要である [2]. 高次元データに単純な既存のクラスタリング (例えば k-means) を使うと,

問題 1. 少数の巨大なクラスタが生成される (多くの疑似レコードが必要)

問題 2. サイズ 1 の (識別されやすい) 小さなクラスタが, 大量に生成される

といった問題が生じる. そこで, これらの問題に対して, 本稿では,

1. 商品集合ベクトルの TF-IDF によるクラスタリング
2. 最小クラスタサイズを制約する新アルゴリズム

を提案する.

* 明治大学総合数理学部 〒 164-8525 東京都中野区中野 4-21-1. School of Interdisciplinary Mathematical Science, Meiji University, 4-21-1, Nakano, Nakano-ku, Tokyo 164-8525, Japan.

クラスタサイズを制約する手法として, 全てのクラスタサイズを均一にするまでクラスタを二分割していく手法 [3] が緒方らによって提案されている. 我々は均一ではなく最小クラスタサイズのみを設けるアプローチをとる. 提案手法によってクラスタリングされた顧客の購入商品を統一するように疑似データを追加し, 本コンテストで使用された購買履歴データを用いて, その有用性と安全性について評価する.

本論文では, 2 章で購買履歴データの特性と PWSCUP2016 によって明らかになった再識別リスクを示し, 3 章で, 再識別の対策として提案手法を示す. 4 章で実験結果を示し, 5 章でまとめを述べる.

2 再識別

2.1 購買履歴データの特性

PWSCUP2016 では共通データセットとして, Online Retail Dataset[4] が使用された. 本データセットは, 英国のオンライン店舗において 2010 年 12 月から約 1 年間に渡り実際に取引された購買履歴データで, UCI Machine Learning Repository[1]が公開している. ここで

- $U = \{u_1, \ldots, u_n\}$: 顧客の集合
- $I(U) = \{g_1, \ldots, g_\ell\}$: 全顧客が購入した商品の集合
- $I(u_i) \subseteq I(U)$: 顧客 u_i が購入した商品の集合
- b: 一人あたりの年間平均購買商品種類の数

と定義し, 加工データの顧客は U' とする. jaccard 値は,

$$J(u_i, u_j) = \frac{|I(u_i) \cap I(u_j)|}{|I(u_i) \cup I(u_j)|}$$

で定まる顧客 u_i, u_j 間の類似度である.

顧客 n 人から全ての異なる 2 人の組み合わせにおける jaccard 平均値を

$$\mu = \frac{1}{\binom{n}{2}} \sum_{i \neq j \in U} J(u_i, u_j)$$

とする. また, 2 顧客が購入した商品集合の積の大きさを $h = |I(u_i) \cap I(u_j)|$ と表すと,

$$\begin{aligned} \mu &= E\left(\frac{|I(u_i) \cap I(u_j)|}{|I(u_i) \cup I(u_j)|}\right) \\ &= \frac{E(|I(u_i) \cap I(u_j)|)}{E(|I(u_i)|) + E(|I(u_j)|) - E(|I(u_i) \cap I(u_j)|)} \\ &= \frac{h}{2b - h} \end{aligned}$$

と変形することができる. ここで, $E()$ は期待値 (平均値) である. これを解いて,

$$h = \frac{2b\mu}{a + \mu} \tag{1}$$

[1] https://archive.ics.uci.edu/ml/datasets/Online+Retail

表 1: コンテストで使用されたデータセットの統計量

項目	変数	値
顧客数	n	400
トランザクション	m	38,087
伝票数		1,763
製品数	ℓ	2,781
単価		$0.04 - 4161$
数量		$1 - 74215$
期間		2010/12/1 – 2011/12/9
平均購買商品種類の数	b	65
jaccard 平均値	μ	0.03

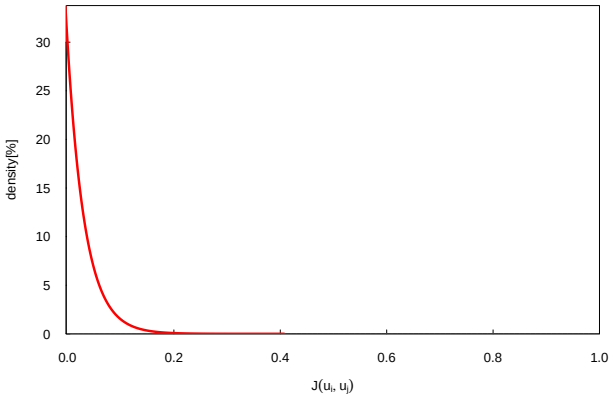

図 1: コンテストデータにおける jaccard 類似度の分布

と表すことができる.

表 1 に, 本データセット M, T の主な統計量を示す. 本データセットは, 顧客が商品を平均 $b = 65$ 個購入し, 無作為に選んだ 2 人について, $h = 4$ 個は他の顧客も購入している商品であることを意味している.

図 1 に, 本データセットの顧客 n 人から異なる 2 人を選んだ際の jaccard ヒストグラムを示す. 本データセットは, jaccard 類似度の最大値が 0.41, 平均 $\mu = 0.03$ であり, 最も似ている顧客同士でも高々0.41%しか類似せず, ほとんどの顧客の購入商品は相違している.

2.2 jaccard 再識別アルゴリズム

本コンテストにおいて, 匿名加工者は, 個人情報である顧客マスターデータ M と各顧客の購買取引の履歴を表すトランザクション T を加工して M', T' を作成し, u と u' の行置換を表す行番号 P を提出する. 加工手法として値の撹乱, レコード削除, 擬似レコード追加などが挙げられる. ここで, 元データのトランザクション数を m とし加工によるレコード数の増減を Δm と定義する.

すなわち, 疑似レコード追加後のトランザクション数 m' は,

$$m' = m + \Delta m$$

である. 再識別者は, 元データ M,T を頼りに, 加工された M',T' を解析して, 推定行番号 Q を導出する. Q と P を比較することにより再識別率を定める.

購買履歴データは, 時系列情報を含んだ複数のトランザクションから構成される動的データである. 動的データは観測期間が長期であるほど履歴の特徴から個人が一意に特定される可能性が高くなる. 本データセットにおいても, 1 年間に及ぶ顧客ごとの購入商品の組み合わせから再識別されるリスクが存在すると考えられる.

そこで, 商品集合の特徴量をもとにして特定を行う識別手法を考える. 元データと加工データのそれぞれについて過去に購入した商品リストを顧客ごとに算出し, 集合の類似度を示す jaccard 係数を用いて最も近い顧客同士を結びつける. 本 jaccard 再識別アルゴリズムを Algorithm1 に示す.

Algorithm 1 jaccard 再識別

Input: M, T, M', T'

Step 1.
元データ M,T と加工データ M',T' について顧客ごとに購入した商品集合を各々$I(u_i), I(u'_i)$ $(i = 1, \ldots, n)$ とする.

Step 2.
加工データの顧客 $j = 1, \ldots, n'$ について, jaccard 類似度が最大である元データの顧客

$$i^*_j = \underset{i \in \{1,\ldots,n\}}{\arg\max}\, J(I(u'_j), I(u_i))$$

と定める.

Output: 選択した顧客の行番号列 $Q = (i^*_1, i^*_2, \ldots, i^*_n)$ を返す.

本アルゴリズムは, $\mathcal{O}(n^2)$ の計算量である.

2.3 評価結果

PWSCUP2016 の本戦に参加した自チームを除く上位 9 チームから提出された購買履歴データを匿名加工したデータを $D_1, \ldots, D_6, D_8, \ldots, D_{10}$ とする. 表 2 に評価結果を示す. (a) 列はコンテストで最も高いチームの識別率,(b) 列は本アルゴリズムによるものである.

赤い数値 (*が付いている数値) は, 提案 jaccard 識別手法が加工データに対して最も再識別率成功率が高かったことを表す. コンテストのルールに則ると, 最も優秀な加工データ D_1 でも 22.25%の顧客が再識別されている [5].

表 2: 商品の特徴による再識別リスク

加工データ	最大再識別率 (a)	jaccard 再識別 (b)
D_1	0.2225	*0.2225
D_2	0.2375	*0.2375
D_3	0.2550	*0.2550
D_4	0.2750	*0.2750
D_5	0.3025	*0.3025
D_6	0.3175	*0.3175
D_8	0.3725	0.2750
D_9	0.3850	*0.3850
D_{10}	0.5500	*0.5500

3 提案加工手法

3.1 jaccard 再識別の対策

本コンテストでは一定の範囲で山岡匿名化 [1][6] をすることが許されていた. しかし山岡匿名化は, ルール上識別が困難なだけで, 実際には全ての個人を特定するリスクがあり安全といえない.

そこで, 山岡匿名化を考えずに, 複数顧客間で購入商品を統一することで jaccard 再識別手法を撹乱する対策を考える. 購入商品の統一する方法には,

1. 既存レコードを変更する方法 $(m' = m)$
2. 既存レコードを削除する方法 $(m' < m)$
3. 疑似レコードを追加する方法 $(m' > m)$

を考えることができる. 既存レコードの変更や削除する方法では, ある商品を実際に購入したという事実が残らないのに対し, 疑似レコード追加による手法は, 元データの購入商品について加工をしないので, 実際に購入したという事実を保証することができる.

本稿では, 山岡匿名化や元データのレコードを加工せず, 疑似レコードを追加するだけで個人特定リスクの一つである商品集合の特徴量を顧客間で統一し, jaccard 再識別手法を撹乱する匿名加工手法について検討する.

疑似レコードの追加アルゴリズムを図 2 に示す. 元データ M, T について顧客ごとの購入商品を集計する (a)(b).

顧客 u_1, u_2, u_3 の購入した商品集合を

$$\begin{aligned} I(u'_1) = I(u'_2) = I(u'_3) &= I(u_1) \cup I(u_2) \cup I(u_3) \\ &= \{g_1, g_2, g_3, g_4, g_5\} \end{aligned}$$

と共通にする (c). 例えば, u_1 の仮 ID に対応する u'_1 に商品 $\{g_3, g_4, g_5\}$ を新たな疑似レコードとして各顧客の適当な伝票 ID に追加する (d).

しかし, 全員が同じ商品を購入したとすると変更が大きすぎるので, 購入商品が類似している顧客をクラスタリングし, 各クラスタ毎に購入商品を統一する.

ここで, クラスタ数を c, 顧客クラスタの集合を $X = \{x_1, ..., x_c\}$, クラスタサイズを $s_i = |x_i|$ と定義する. ただし, 各クラスタ x_i は顧客 u の集合であり, $\bigcup_{i=1}^{c} x_i = U$ で

Algorithm 2 疑似レコード追加アルゴリズム

Input: $M, T, X = \{x_1, x_2, \ldots, x_c\}$

各クラスタ $x \in X$ において, 加工後の各顧客の商品集合が $I(x) = \bigcup_{u \in x} I(u)$ に統一されるように, x 内の顧客 u に $I(x) - I(u)$ の商品をもつ疑似レコードを u が持つ適当な伝票 ID に追加する. このとき, 単価 0.1-0.9, 数量 1 とした.

Output: M', T', P

(b)

	U	$I(U)$
1	u_1	$\{g_1, g_2\}$
2	u_2	$\{g_1, g_3, g_5\}$
3	u_3	$\{g_4, g_5\}$

(c)

P	U'	$I(U')$
1	u'_1	$\{g_1, g_2, g_3, g_4, g_5\}$
2	u'_2	$\{g_1, g_2, g_3, g_4, g_5\}$
3	u'_3	$\{g_1, g_2, g_3, g_4, g_5\}$

(a) T

U	伝票ID	商品
u_1	100	g_1
u_1	300	g_2
u_2	500	g_1
u_2	500	g_3
u_2	600	g_5
u_3	...	...

(d) T'

U'	伝票ID	商品	
u'_1	100	g_1	
u'_1	300	g_2	
u'_1	300	g_3	*
u'_1	300	g_4	*
u'_1	300	g_5	*
u'_2	...	...	

* 疑似レコード

図 2: 疑似レコードの追加方法

ある. クラスタ毎にレコード追加する手法を Algorithm2 に示す.

クラスタ内の商品集合を統一することで jaccard 再識別による個人の特定は, 元データの商品集合の要素数が最多な顧客のみに限定することができる. x における疑似レコード数は, $\Delta m = \sum_{u \in x} |I(x)| - |I(u)|$ である. また, 本稿では商品の数量を考慮する多重集合については議論しない.

3.2 レコード間距離の定義

顧客ごとの商品集合のデータは高次元であり, そのままクラスタリングに適用しても意図した結果が得られない. 例えば, jaccard 係数を 2 顧客間の距離として, k-means アルゴリズムによりクラスタリングした結果を図 3 に示す. 最大のクラスタのサイズが 294 個と極端に大きく, サイズが 1 のクラスタが 45 個生じており, クラスタサイズに大きな偏りが生じている.

そこで, 我々は文書をクラスタリング [7] する際に用いる TF-IDF を使い, 各商品に対して重み付けをしてクラスタリングを行う. Algorithm3 に TF-IDF を用いたクラスタリングの流れを示す.

また, 図 4 を例として, 顧客 $U = \{u_1, u_2, u_3, u_4\}$ を 2

Algorithm 3 TF-IDF による購入商品の重み付け

Input: 顧客 $u_i \in U$, 商品集合 $I(u_i), c$

Step 1. 顧客 u_i の全商品数 ℓ 次元の特徴ベクトルを $\boldsymbol{v}_i = (f_{i1}, f_{i2}, \ldots, f_{i\ell})$ と表す. ここで,

$$f_{ij} = \begin{cases} 1 & \text{if} \quad I(u_i) \ni g_j \\ 0 & \text{otherwise} \end{cases}$$

とする.

Step 2. ある商品 g_j を購入した全顧客の集合を $D_j = \{u_i \in U \,|\, I(u_i) \ni g_j\}$ と表す. f_{ij} の TF-IDF による重みを

$$f'_{ij} = \frac{f_{ij}}{\sum_{k=1}^{\ell} f_{ik}} (\log \frac{n}{|D_j|} + 1)$$

と定め, 重み付けした顧客 u_i の特徴ベクトルを $\boldsymbol{v}'_i = (f'_{i1}, f'_{i2}, \ldots, f'_{i\ell})$ で表す.

Step 3. 特徴ベクトル $\boldsymbol{v}'$ 間の cos 類似度を算出して顧客 U を k-means を使ってクラスタリングする.

Output: $X = \{x_1, x_2, \ldots, x_c\}$

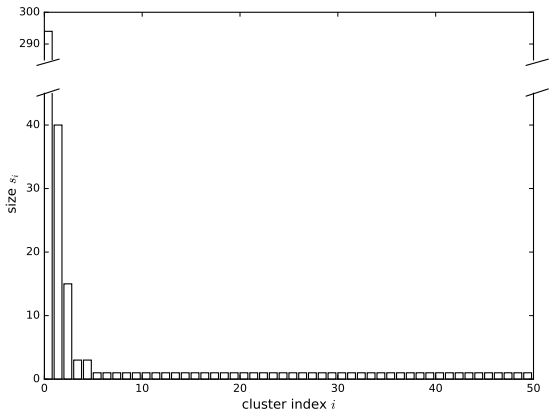

図 3: jaccard 距離によるクラスタサイズの分布

つのクラスタ $X = \{x_1, x_2\}$ に分類する手順について考える. 顧客ごとの商品集合 (a) において, 購入している商品を 1 とした 2 値行列に変換する (b). u_1 の購入商品 g_1 の特徴量は, TF$= \frac{1}{2}$, IDF$= 1$ から, 0.5 と重み付けされる (c). 顧客間の cos 類似度よりクラスタリングして $x_1 = \{u_1, u_2\}, x_2 = \{u_3, u_4\}$ を出力する (d).

3.3 方式 1(既存クラスタリングベース)

TF-IDF による商品を重み付けと cos 類似度を使った k-means によるクラスタリングを行い, 各クラスタ内で商品集合の和集合をとり疑似レコードを追加する手法を提案方式 1 とする.

$c = 50$ としたときの, 各クラスタ $x_i \in X$ に属する顧客の数 s を図 5 に示す. 図 3 の jaccard 係数によるクラスタリング結果と比較して, 明らかにクラスタサイズの偏りが平均化され, TF-IDF を使うことでクラスタの偏

(a)

U	$I(U)$
u_1	$\{g_1, g_2\}$
u_2	$\{g_1, g_2\}$
u_3	$\{g_1, g_2, g_3, g_4\}$
u_4	$\{g_1, g_3, g_4\}$

(b)

	g_1	g_2	g_3	g_4	
u_1	1	1	0	0	$\boldsymbol{v_1}$
u_2	1	1	0	0	$\boldsymbol{v_2}$
u_3	1	1	1	1	$\boldsymbol{v_3}$
u_4	1	0	1	1	$\boldsymbol{v_4}$

(d)

U	cluster
u_1	0
u_2	0
u_3	1
u_4	1

(c)

g_1	g_2	g_3	g_4	
0.5	0.56	0	0	$\boldsymbol{v'_1}$
0.5	0.56	0	0	$\boldsymbol{v'_2}$
0.25	0.28	0.32	0.32	$\boldsymbol{v'_3}$
0.33	0	0.42	0.42	$\boldsymbol{v'_4}$

図 4: TF-IDF を用いた類似顧客のクラスタリングの例

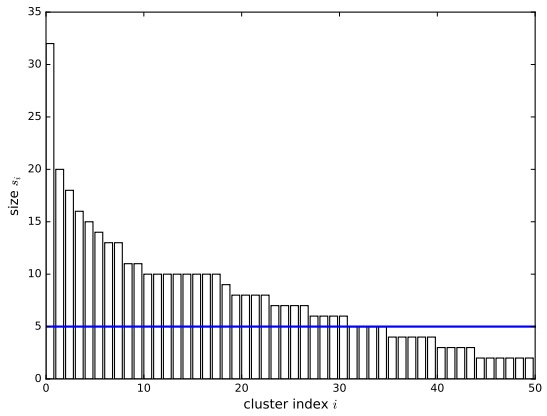

図 5: 方式 1 における各クラスタサイズの分布 ($c = 50$)

りを改善している. また, 最大クラスタ x_{max}, 最小クラスタ x_{min} とすると, $|x_{max}| = 32$, $|x_{min}| = 1$ であり, 大きいクラスタに属する顧客ほど, 追加すべき疑似レコード数は増える. 逆に, $s_i = 1$ のクラスタの顧客は疑似レコードを追加しないので一意に特定することができる.

3.4 方式 2(調整アルゴリズム)

方式 1 のクラスタサイズの偏りを改善するため, 全てのクラスタサイズが下限値 s_{min} を下回らないようにクラスタを調整するアルゴリズムを方式 2 を提案する. 本手法の操作を Algorithm4 に示す. 購入商品が最も類似する顧客を, 最大クラスタ x_{max} から s_{min} 未満のクラスタへ移動し, 全てのクラスタサイズが s_{min} 以上になるよう繰り返す.

取りうるクラスタサイズ s_{min} の下限値はクラスタ数

Algorithm 4 方式 2 調整アルゴリズム

Input: s_{min}, c, M, T
  方式 1 でクラスタリング
  クラスタの集合: $X = \{x_1, x_2, \ldots, x_c\}$
  for x **in** $\{x_i \in X \big| |x_i| < s_{min}\}$ **do**
    最大クラスタ: $x_{max} \in X$
    while $|x'| < s_{min}$ **do**
      $u_j = \underset{u_j \in x_{max}, u_i \in X}{\arg\max} J(I(u_i), I(u_j))$
      $x'_{max} \leftarrow x_{max} - \{u_j\}$
      $x' \leftarrow x \cup \{u_j\}$
    end while
  end for
  Algorithm2 へ
Output: M', T', P

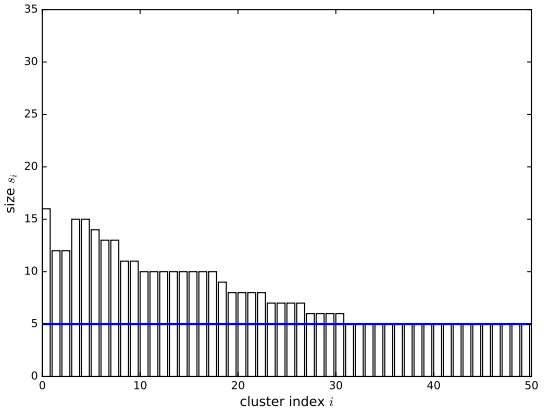

図 6: 方式 2 における各クラスタサイズの分布 ($s_{min} = 5, c = 50$)

c に依存し, その値域は

$$s_{min} \in \{2, 3, \ldots, \lfloor \frac{n}{c} \rfloor\}$$

である.

方式 1 に本手法を適用した時の, 各クラスタ x_i に対するサイズ s_i を図 6 に示す. $s_{min} = 5, c = 50$ としたとき, 最大クラスタサイズが 32 個 (図 5) から 16 個 (図 6) に減少し, 全てのクラスタのサイズが s_{min} を下回らないように改善した.

4 評価

4.1 有用性と擬似レコード数 Δm の関係

提案方式の加工データの有用性は, 追加する疑似レコード数 Δm に大きく依存する. そこで, Δm が各有用性指標を代表する値であることを示すため, コンテストでの有用性指標と Δm との相関係数を表 3 に示す. Δm に対して有用性指標 $U1$-cMAE, $U2$-cMAE, $U3$-RFM には強い負の相関があり, 増加に伴い有用性が下がる. また, c と Δm の相関係数は-0.8454 であり, クラスタ数 c の増加に伴って, Δm が減少し, 再識別率が上がる.

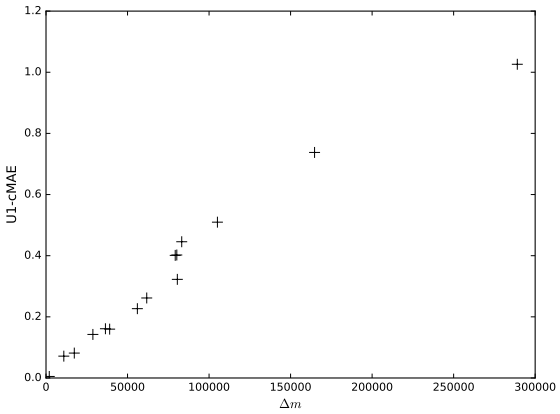

図 7: Δm と U1 の関係

Δm に対するコンテストでの有用性指標 $U1$-cMAE の関係を図 7 に示す. $0 < \Delta m \leq 300000$ の範囲で擬似レコードを追加したとき, 有用性は $0.0 \leq U1 \leq 1.02$ を示し, Δm の増加に伴い有用性は悪化した. Δm と各種有用性指標との間に強い相関関係があることが確かめられたので, 本節では, 各方式の手法に対して 10 回の試行を行い, Δm と jaccard 識別の二つの指標を用いて提案手法を評価する.

4.2 Δm の理論値

疑似レコード追加手法における Δm の理論値を求める.

理論値を求めるための準備として, 変数 a_i の定義を図 8 を用いて説明する. ある一つのクラスタ x に着目し, ある顧客に対して a_i を

- a_1: 自分だけが購入している商品の数
- a_2: 他 1 人の顧客も購入している商品の数
- a_3: 他 2 人の顧客も購入している商品の数

と定義する. 図 8 は, クラスタサイズ $s = 3$ の時の a_1, a_2, a_3 を表す. ただし, a_i はそれぞれクラスタ内の平均であり, $s > 3$ においても同様に a_i を定義することができる.

ここで h, b は, a_i を使って

$$h = a_2 + \sum_{i=1}^{s-2} \binom{s-2}{i} a_{i+2}$$
$$b = a_1 + \sum_{i=1}^{s-1} \binom{s-1}{i} a_{i+1} \quad (2)$$

と表すことができる.

図 8 を例に, 顧客 u_1, u_2, u_3 で構成されるクラスタ x における追加レコード数について考える. 顧客 u_1 に着目すると a_1 を他 2 人に, a_2 は 1 人と共通している商品

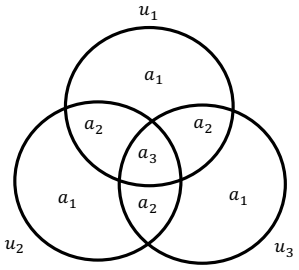

図 8: a_i の定義

なので残り 1 人に追加する. a_3 は全員が購入しているので新たに追加しない. この操作を顧客 u_2, u_3 についても行う.

従って, サイズ s のクラスタ x における追加レコード数は,

$$\Delta m(x, s) = \sum_{i=1}^{s} (s - i) \binom{s}{i} a_i$$

と一般化される.

よって, 全クラスタにおける追加レコード数の期待値は,

$$\begin{aligned} E(\Delta m) &= c\Delta m(x, s) \\ &= -\frac{hn^3}{2c^2} + (b + \frac{h}{2})\frac{n^2}{c} - bn \quad (3) \\ &\geq (b + \frac{h}{2})\frac{n^2}{c} \quad (4) \end{aligned}$$

であり, データセットの特性を示す b, μ, n をパラメータとした c の式で近似することができる. ただし, $a_i \geq 0$ であり, $a_i = 0 (i \geq 3)$, クラスタサイズが一定 $(s = \frac{n}{c})$ と仮定をおいたことに注意したい.

4.3 有用性

s_{min} における追加疑似レコード数の関係を表 4 に示す. 各 c について $s_{min} = \lfloor \frac{n}{c} \rfloor$ の時, Δm は最小をとる. また, 方式 2 を適用することによる jaccard 類似度の標準偏差は c, s_{min} の値に対して 0.01 未満を示し, 安定している.

次に, $n = 400$ の購買履歴データにおける, c に対する Δm の関係を図 9 に示す. ここで, 方式 2 の Δm は, $s_{min} = \lfloor \frac{n}{c} \rfloor$ の時の加工データである. 方式 2 は方式 1 の追加手法に比べて, Δm を約 53%と大幅に抑えることができている.

実線は, (3) 式による理論値である. (2) 式は, s が大きくなると $a_1 \geq 0$ を満たさなくなってしまう. すなわち, 理論式の定義域は $n = 400$ のデータセットにおいて $c \geq 23$ である. $c \geq 23$ のとき, 実測値は理論値に沿っている.

表 3: Δm と各種有用性の相関係数

	Δm	$U1$	$U2$	$U3$	$U4$	Y1	jaccard	Reid	c
Δm	1.0000								
$U1$-cMAE	0.9798	1.0000							
$U2$-cMAE	0.9798	1.0000	1.0000						
$U3$-rfm	0.9547	0.9876	0.9876	1.0000					
$U4$-topitems	-	-	-	-	-				
$Y1$-subset	0.6690	0.6798	0.6798	0.7030	-	1.0000			
jaccard	-0.8586	-0.9327	-0.9327	-0.9494	-	-0.7349	1.0000		
Reid	-0.8489	-0.9247	-0.9247	-0.9432	-	-0.7434	0.9996	1.0000	
c	-0.8454	-0.9220	-0.9220	-0.9406	-	-0.7461	0.9994	0.9999	1.0000

表 4: s_{min} に対する Δm の関係

	$c = 50$			$c = 75$			$c = 100$			$c = 125$		
	Δm	jaccard	Reid	Δm	jaccard	Reid	Δm	jaccard	Reid	Δm	jaccard	Reid
方式 1	182897	0.1728	0.1235	141696	0.2402	0.1858	128568	0.3060	0.2488	97581	0.3692	0.3120
$s_{min} = 2$	183902	0.1729	0.1223	136526	0.2403	0.1860	99228	0.3061	0.2475	60492	0.3687	0.3105
$s_{min} = 3$	175449	0.1726	0.1222	112781	0.2394	0.1855	68357	0.3041	0.2480	*46101	0.3667	0.3102
$s_{min} = 4$	162474	0.1723	0.1218	*91946	0.2382	0.1855	*59374	0.3044	0.2465			
$s_{min} = 8$	*125798	0.1681	0.1218									

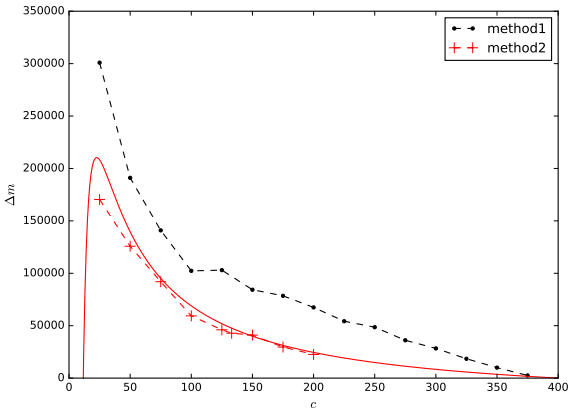

図 9: 方式 1 と方式 2 の有用性の比較

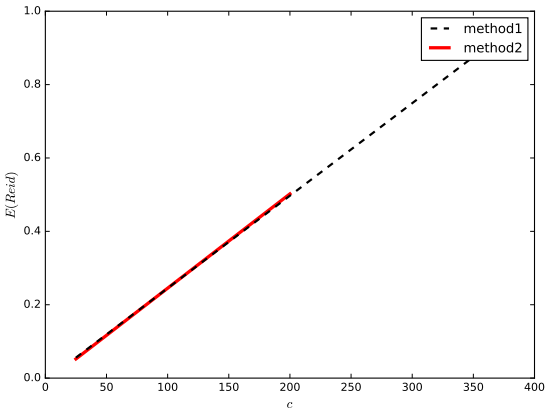

図 10: 方式 1 と方式 2 の安全性の比較

4.4 安全性

2 章で述べた jaccard 再識別アルゴリズムを方式 1 と方式 2 による加工データに適用したときの再識別率を図 10 に示す. 本手法による加工データに対して jaccard 再識別を行うと, クラスタ内のどの顧客 $u' \in x$ も, 元データの商品要素数が最多な顧客 $u \in x$ に識別される. よって, 方式 1, 方式 2 ともに再識別率の期待値は

$$E(Reid) = \frac{c}{n}$$

である.

表 4 に, 各 c における再識別率の実測値 $Reid$ を示す. 実測値 $Reid$ と期待値 $E(Reid)$ の誤差は, 商品要素数が最多となる顧客 u が複数存在したことによるものだろう.

4.5 最適クラスタ数

データを加工すると, 一般的に有用性が悪くなり, 安全性が高くなる. しかし, この 2 つの指標を総合的に評価するにはユースケースやデータ構成に依存する. 本稿では, コンテストでの総合評価に使用された $\frac{U+E}{2}$ の U を Δm と置き換え,

$$\frac{\alpha E(\Delta m) + E(Reid)}{2} \tag{5}$$

を用いてクラスタの最適値 c^* を定める. ここで, α は Δm を $0 \leq E(\Delta m) \leq 1$ に正規化する係数とする. 図 11 に最適値 c^* を示す. $n = 400, b = 65, \mu = 0.03$ のデータセットを方式 2 の手法に適用すると, 評価値が極小となるのは, $c^* = 130$ の時である.

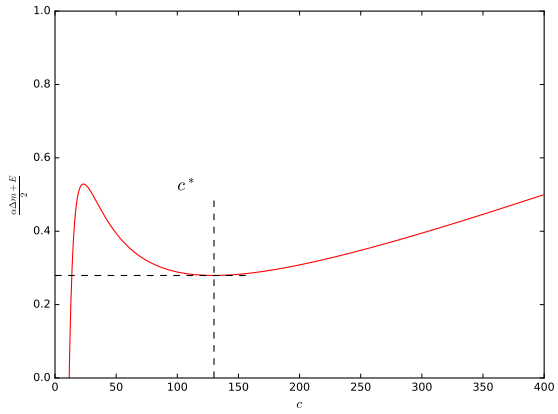

図 11: 方式 2 の最適値 c^*

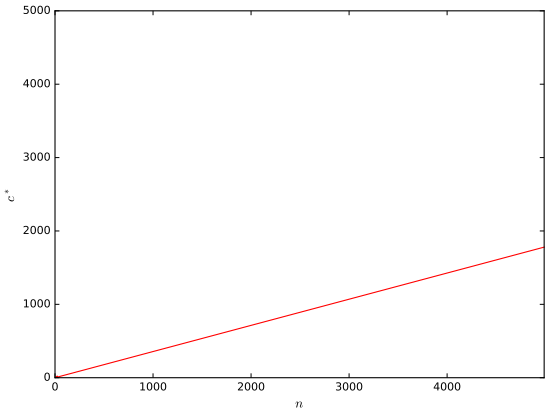

図 12: n に対する c の最適値 c^*

顧客数 n についてのクラスタ数の最適値 c^* を考えよう. (3) 式の $E(\Delta m)$ において, $\frac{1}{c}$ が支配的な項であることから, (4) 式を (5) 式に代入した極小値から最適値

$$c^* = \sqrt{\alpha(b + \frac{h}{2})n^3} \tag{6}$$

を得る. ただし, α は n に依存する変数であることに注意したい. 図 12 に $b = 65, \mu = 0.03, n$ 人の特性を持つデータセットに対する, 最適値 c^* の変化を示す.

(5) 式と α を与えたとき, (6) 式を用いて, データセットの特性 b, μ, n から, 方式 2 における最適値 c^* を導出することができる. 例えば, $n = 4000, b = 65, \mu = 0.03$ のデータセットに対しては, $c^* = 1427$ より, 再識別率 $E(Reid) = 0.3567$, $E(\Delta m) = 490650$ が方式 2 における最適な加工である.

5 おわりに

PWSCUP2016 の結果に基づき, 購入商品の特徴を用いた再識別手法における特定リスクを明らかにした. その対策として疑似レコードを追加する匿名加工手法を提案した. 提案手法は, 商品の類似している顧客を TF-IDF による重み付けを取り入れてクラスタリングし, クラスタサイズの下限値を設けることで追加疑似レコード数を抑える. また, 提案方式における追加疑似レコード数と再識別率の理論値を求めた.

今後の課題として, 別のデータセットを適用したときの効果の確認およびクラスタリングの精度評価などがあげられる. また, 有用性を下げすぎないよう疑似レコードの追加だけでなく, 削除や書き換えによる手法についても考えていく.

謝辞

コンテストにおける再識別手法について, ともに議論した明治大学菊池研究室の岡本健太郎氏, 田中司氏に感謝する.

参考文献

[1] 菊池浩明, 小栗 秀暢, 野島 良, 濱田 浩気, 村上 隆夫, 山岡 裕司, 山口 高康, 渡辺 知恵美, “PWSCUP:履歴データを安全に加工せよ”, CSS 2016, pp. 271-278, 2016.

[2] 長谷川聡, 菊池亮, 正木彰伍, 濱田浩気, “行列分解を利用した確率的 k-匿名性を満たす高次元データ公開法”, CSS 2016, pp. 936-942, 2016.

[3] 緒方悠人, 遠藤靖典, “K-Member Clustering 問題に関する一考察”, FSS 2013, pp. 61-66, 2013.

[4] Daqing Chen, Sai Liang Sain, and Kun Guo, “Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining,” Journal of Database Marketing and Customer Strategy Management, Vol. 19, No. 3, pp. 197-208, 2012.

[5] PWS 実行委員会, “PWSCUP 匿名加工・再識別コンテスト”, (https://pwscup.personal-data.biz), 2016 年 12 月参照.

[6] 菊池浩明, 山口高康, 濱田浩気, 山岡裕司, 小栗秀暢, 佐久間 淳, “匿名加工再識別コンテスト Ice&Fire の設計”, CSS 2015, pp. 363-370, 2015.

[7] Rakesh Chandra Balabantaray, Chandrali Sarma and Monica Jha, “Document Clustering using K-Means and K-Medoids”, arXiv preprint arXiv:1502.07938, 2015.