明治大学総合数理学部

2016 年度

卒　業　研　究

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

学位請求者　先端メディアサイエンス学科

原田 玲央

目次

第 1 章

はじめに

1.1 研究背景

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な法整備や技術的な手法の確立が必要である.

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. 匿名加工情報とは, ビックデータをはじめとするパーソナルデータの利活用に向けて, 本人の同意に代わる一定の条件の下, 特定の個人を識別することができないように個人情報を加工したものを差す. 個人情報を適切に加工することにより, 収集したパーソナルデータを本人の同意なしに第三者企業に販売することが可能となった. しかし, 技術的に適切な加工手法, 評価手法が確立されておらず, 実用化に向けて研究が進められている. このような背景から, 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]. 参加者は, 共通の購買履歴データセットを用いて, 有効な匿名加工手法について競いあった.

1.2 研究目的

著者を含むグループは, 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.

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.

1.3 論文構成

本論文の構成は次のとおりである.

まず, 第 2 章で本研究の実験に使用した, 購買履歴データセットの分析と特性について, 第 3 章で購買履歴データに対する再識別手法の提案し, 評価を行う. 第 4 章では, 第 3 章で提案した再識別手法を問題点と捉え, 対抗する匿名加工手法について提案をする. 第 5 章で, 提案加工手法について評価と考察を述べ, 最後に, 第 6 章でまとめを述べる. 付録として, 本研究を進めるにあたり, 背景を理解するために行ったプチ PWSCUP の概要と, 3 年次に取り組んだ内容について触れる.

第 2 章

データセットの特性

2.1 Online Retail Dataset

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.

本データセットは, ギフト製品の取引における卸売業者の購買履歴であり, 顧客 ID, 国, 伝票 ID, 日時, 製品 ID, 単価, 数量の 7 属性で構成される. コンテストでは, クレンジングを行い, 顧客が行なった購買取引の履歴を表すトランザクションデータ T と, T をもとに合成した顧客マスターデータベース M が使用された. ただし, M の性別, 生年月日は架空に生成されたものである. 顧客マスター M と購買履歴データ T の例を表 2.1,2.2 にそれぞれ示す. M について, 顧客 ID は一意であり, 重複は認められない. 一方, T の顧客 ID は, 必ず M に存在し, 重複を許す.

2.2 データセットの分析と特性

2.2.1 変数定義

クレンジングされたデータセットは, 顧客数 $n = 400$, トランザクション数 $m = 38,087$, 出現する製品の種類数 $\ell = 2,781$ である. ここで, コンテストで使用されたデータセットの特性を示すために, 変数を定義する.

表 2.1 マスター M

顧客	性別	生年月日	国籍
12346	m	1976/02/24	UK
12347	f	1994/05/11	USA
12348	f	1994/08/12	UK
12349	f	1995/01/13	JPN
12350	f	1994/02/14	JPN

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

表 2.2　トランザクション T

顧客	伝票	取引日	時刻	商品	単価	数量
12346	10000	2011/02/17	10:30	13	500.0	1
12347	20000	2011/02/18	11:00	25	2.4	100
12347	20000	2011/02/18	11:00	65	15.6	50
12347	20000	2011/02/18	11:00	68	73.6	40
12350	30000	2011/02/19	12:00	14	3.2	55
12348	40000	2011/02/19	12:30	50	0.8	90
12347	50000	2011/02/19	12:30	50	0.8	75
12349	60000	2011/02/22	13:00	46	2.1	10

- $U = \{u_1, \ldots, u_n\}$: 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 が購入した商品種類の集合
- $b = \ell/n$: 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. ただし, ℓ, b は, 商品の種類数であり購買数を考慮したものではないことに注意したい.

2.2.2　顧客間における購買商品の類似性

データセットの特性として, 顧客同士の購買商品がどの程度類似しているのかを導出する. 顧客同士の類似性は, 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 平均値で示す.

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.

顧客同士の類似性を

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

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,

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

表 2.3　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
2 顧客間の商品集合の積の大きさ	h	3.9

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,

$$h = \frac{2b\mu}{b+\mu} \tag{2.1}$$

h を b, μ を用いて表すことができる.

2.2.3　データセットの統計量

表 2.3 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 2.1 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.

同様に, 図 2.2 は, $n = 4,333$ のフルデータセットにおける jaccard ヒストグラムである. jaccard 類似度の最大値は 1.0, 平均 $\mu = 0.03$ であり, $n = 400$ のデータセットと同様, 指数分布に従い, ほとんどの顧客が購入商品は相違しているデータセットである特性を持つ.

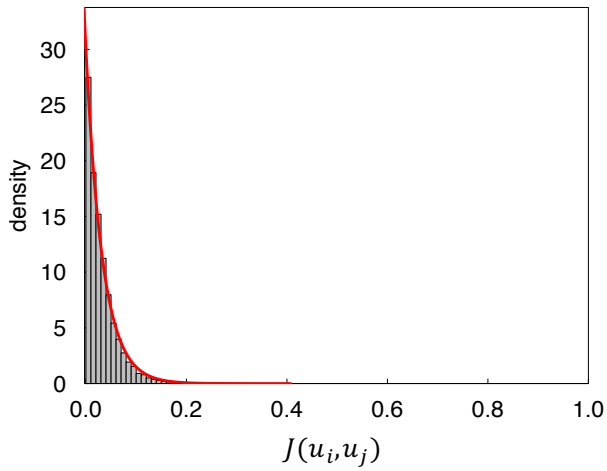

図 2.1　コンテストデータ ($n = 400$) における jaccard 類似度の分布

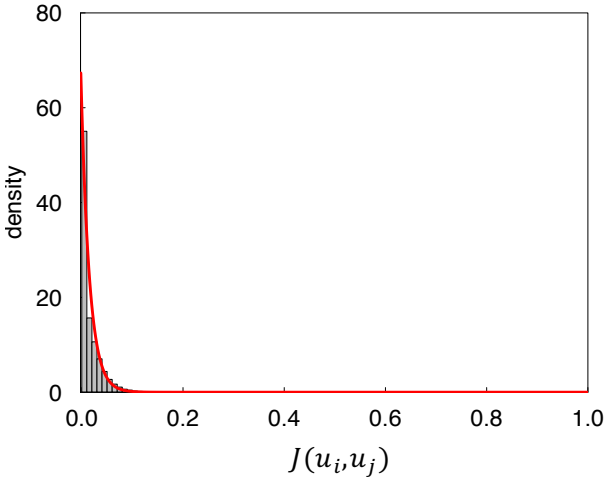

図 2.2　フルデータ ($n = 4,333$) における jaccard 類似度の分布

第 3 章

再識別

3.1 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 Algorithm 1 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.

表 3.1　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)	multi-jaccard 再識別 (c)
D_1	0.2225	*0.2225	0.2200
D_2	0.2375	*0.2375	*0.2375
D_3	0.2550	*0.2550	0.2325
D_4	0.2750	*0.2750	*0.2750
D_5	0.3025	*0.3025	*0.3025
D_6	0.3175	*0.3175	*0.3175
D_8	0.3725	0.2750	0.2600
D_9	0.3850	*0.3850	*0.3850
D_{10}	0.5500	*0.5500	0.5100

3.2　再識別評価結果

3.2.1　再識別率

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 3.1 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．(c) 列は Algorithm 1 を購入商品の数量も考慮する多重集合を用いた jaccard 再識別アルゴリズムによる評価である．本アルゴリズムを multi-jaccard 再識別とする．

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].

本コンテストに参加されたチームのほぼ全てのデータが, 購入商品の特徴量をヒントに個人を識別するリスクが存在することが明らかとなった．

3.2.2　処理時間

jaccard 再識別アルゴリズムと multi-jaccard 再識別アルゴリズムの処理時間について比較を行う．

擬似レコード数 Δm に対する再識別の処理時間について図 3.1 に示す．両方ともに Δm の増加に伴い, 再識別にかかる処理時間は増加する．jaccard 再識別アルゴリズムに比べて, multi-jaccard 再識別アルゴリズムの処理時間は大幅にかかる．

再識別アルゴリズムの実験環境を表 3.2 に示す．本再識別アルゴリズムは python で実装され, ライブラリとして Numpy, Pandas を利用した．Numpy は, 数値計算を効率的に行うための拡張モジュールであり, 多次元のベクトル演算を高速に行うことができる．Pandas は, データ解析を支援する機能を提供するライブラリである．特に, 数表及び時系列データを操作するためのデータ構造と演算を提供している．

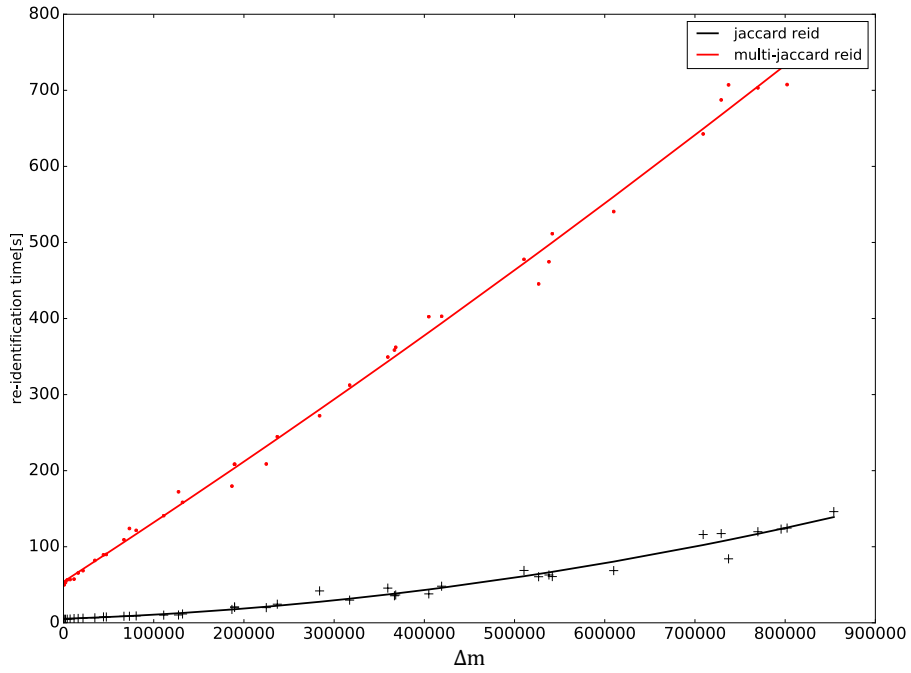

図 3.1　多重 jaccard と jaccard 再識別アルゴリズムの性能

表 3.2　実験環境

	環境
OS	OS X Sierra
メモリ	8GB
CPU	Intel Core i5
クロック	2.6GHz
使用言語	Python 2.7.10
ライブラリ	Numpy, Pandas

第 4 章

提案匿名加工手法

4.1 jaccard 再識別の対策

コンテストにおいて, 加工データの各レコードが, 元データにおけるどのレコードなのかといった情報がなければ, 適切に再識別評価を行うことができない. そこで, 匿名加工者は, 加工データと各レコードに対応する正解行番号を作成する. しかし, 正解行番号がスワップされていると, データ自体を加工していなくても, 安全であると評価されてしまう. このような加工手法は山岡匿名化と呼ばれる [1][6]. 本コンテストでは一定の範囲で山岡匿名化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.

本節では, 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 4.1 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).

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	$\{g_1, g_2, \underline{g_3}, \underline{g_4}, \underline{g_5}\}$
2	u'_2	$\{g_1, \underline{g_2}, g_3, \underline{g_4}, g_5\}$
3	u'_3	$\{\underline{g_1}, \underline{g_2}, \underline{g_3}, g_4, g_5\}$

(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	$\underline{g_3}$	*
u'_1	300	$\underline{g_4}$	*
u'_1	300	$\underline{g_5}$	*
u'_2	...	...	

* 疑似レコード

図 4.1　疑似レコードの追加方法

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.

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 $D_j = \{u_i \in U \big| I(u_i) \ni g_j\}$ 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: $X = \{x_1, x_2, \ldots, x_c\}$

4.2 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 4.2 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.

TF-IDF とは, 自然言語処理の分野において, 文書検索や分類に使われる指標の一種である. 各文書において出現頻度が高い単語は特徴的であると定義する TF 指標, 文書全体で登場回数が少ない単語は特徴的であると定義する IDF 指標の 2 つの指標を用いて, その文書を特徴づけている単語に対して重み付けを行う指標である. そこで, 文書と単語を, 顧客と購入商品に置き換え, ある顧客における購入商品の出現頻度を TF, 顧客全体における商品の登場回数の希少性を IDF とし, どの商品が顧客を特徴づけているのかを TF-IDF 指標による重み付けによって定義した.

重み付けをしない場合は, 顧客同士の距離が同程度に離れていて, 意図したように分類ができないのに対し, TF-IDF により重み付けすることによって, 顧客を特徴づけている商品が類似しているならば, 顧客同士は類似しているとみなし, クラスタサイズの偏りが改善された分類が可能である.

また, 図 4.3 を例として, 顧客 $U = \{u_1, u_2, u_3, u_4\}$ を 2 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).

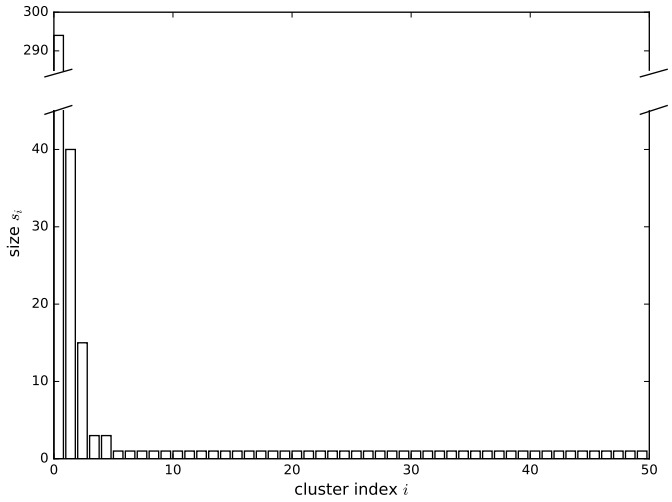

図 4.2　jaccard 距離によるクラスタサイズの分布

(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 4.3　TF-IDF を用いた類似顧客のクラスタリングの例

4.3　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 4.4 に示す. 図 4.2 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, $|x_{max}| = 32$, $|x_{min}| = 1$

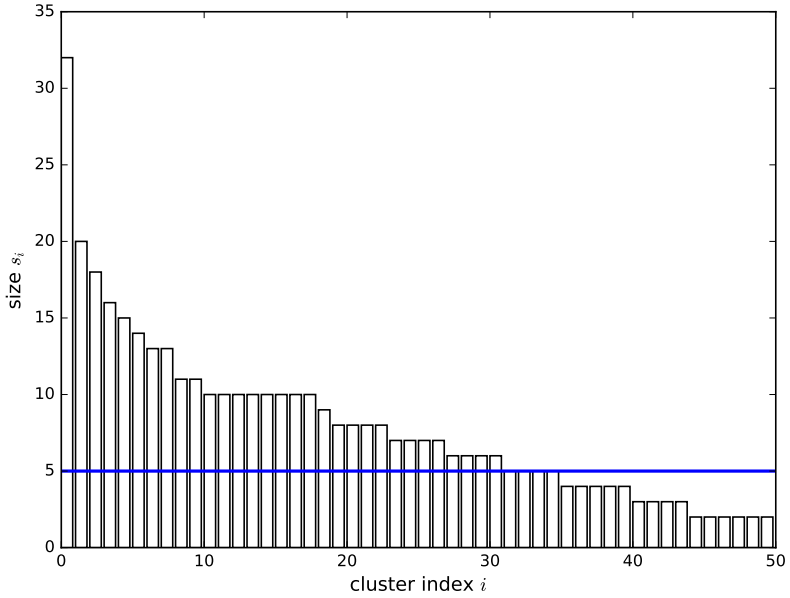

図 4.4　方式 1 における各クラスタサイズの分布 ($c = 50$)

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.

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

4.4 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 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 4.5 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 4.4) から 16 個 (図 4.5) 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.

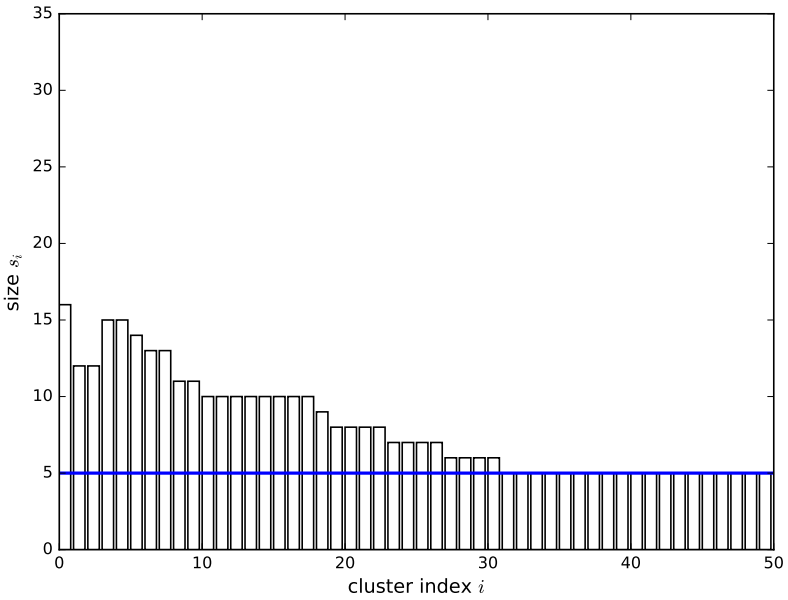

図 4.5　方式 2 における各クラスタサイズの分布 ($s_{min} = 5, c = 50$)

第 5 章

評価

5.1 Δm と他有用性指標との相関関係

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 5.1 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.

Δm に対するコンテストでの有用性指標 $U1$-cMAE の関係を図 5.1 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.

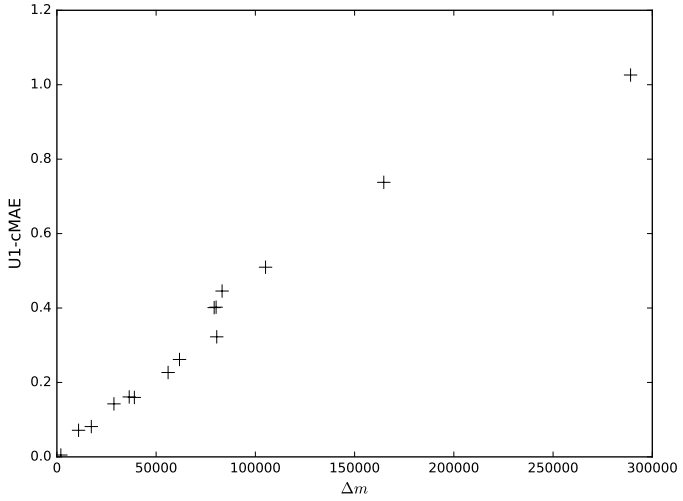

図 5.1 Δm と U1 の関係

表 5.1　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

5.2　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 5.2 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 5.2 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 2.2 節で求めた h, b は, a_i を使って

$$
\begin{aligned}
h &= a_2 + \sum_{i=1}^{s-2} \binom{s-2}{i} a_{i+2} \\
b &= a_1 + \sum_{i=1}^{s-1} \binom{s-1}{i} a_{i+1}
\end{aligned}
\tag{5.1}
$$

と表すことができる.

図 5.2 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,

$$
\Delta m(x, s) = \sum_{i=1}^{s} (s - i) \binom{s}{i} a_i \tag{5.2}
$$

と一般化される.

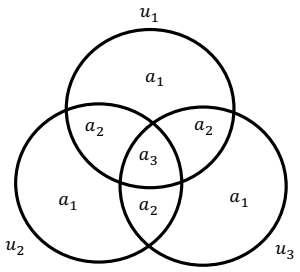

図 5.2　a_i の定義

よって, 全クラスタにおける追加レコード数の期待値は,

$$\begin{aligned} E(\Delta m) &= c\Delta m(x,s) \\ &= -\frac{hn^3}{2c^2} + (b+\frac{h}{2})\frac{n^2}{c} - bn \qquad (5.3) \\ &\geq (b+\frac{h}{2})\frac{n^2}{c} \qquad (5.4) \end{aligned}$$

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.

仮定として $a_i = 0 (i \geq 3)$ としたが, この場合, 2.2 節で求めたパラメータを使って $E(\Delta m)$ を求めることが可能である. ここで, $a_i = 0$ の仮定を置かず, データ分析の段階で a_i を算出することができれば, (5.2) 式をから, より正確に $E(\Delta m)$ を求めることができるが, 本論文では議論はしない.

表 5.2　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									

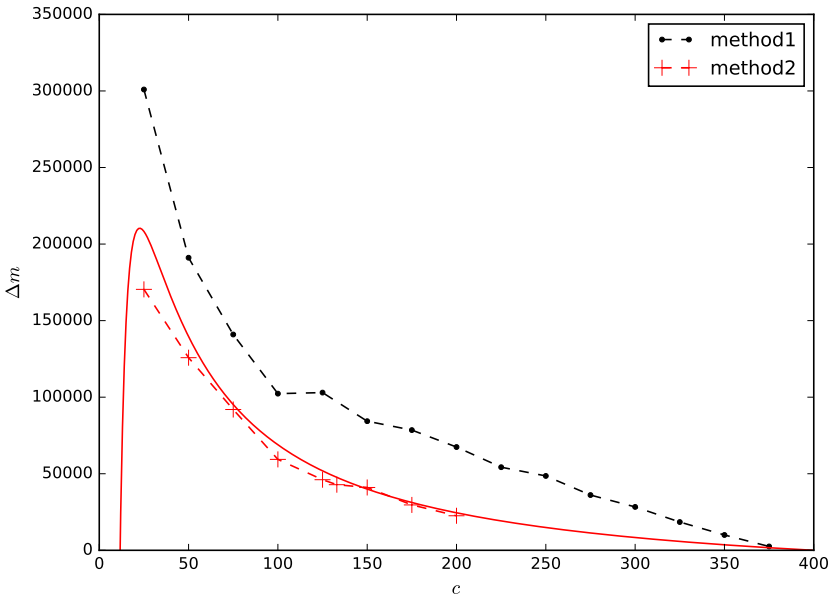

図 5.3　方式 1 と方式 2 の有用性の比較

5.3　有用性

s_{min} における追加疑似レコード数の関係を表 5.2 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 5.3 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, (5.3) 式による理論値である．(5.1) 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．

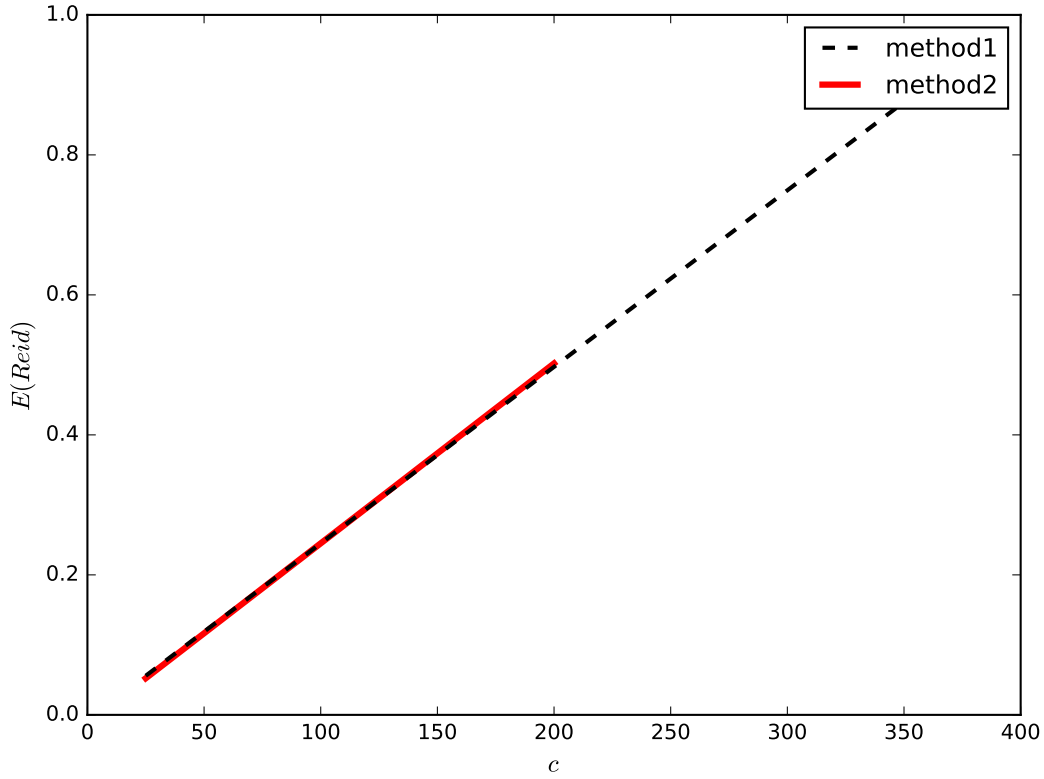

図 5.4　方式 1 と方式 2 の安全性の比較

5.4　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 5.4 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 5.2 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と考える．

5.5 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,

$$\frac{\alpha E(\Delta m) + E(Reid)}{2} \tag{5.5}$$

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 5.5 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.

顧客数 n についてのクラスタ数の最適値 c^* を考えよう. (5.3) 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, (5.4) 式を (5.5) 式に代入した極小値から最適値

$$c^* = \sqrt{\alpha(b + \frac{h}{2})n^3} \tag{5.6}$$

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 5.6 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.

(5.5) 式と α を与えたとき, (5.6) 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.

5.6 考察

5.6.1 再識別について

購買履歴データには, 各個人ごとの履歴の特徴から個人を特定するリスクが存在することが分かった. 購買履歴だけでなく位置情報データのようなその他の動的データにも同じようなリスクが存在すると言えるだろう.

提案した jaccard 再識別アルゴリズムにおいて, 数量を考慮する multi-jaccard 再識別と数量を考慮しない jaccard 再識別との再識別率の違いは小さい. 詳細は表 3.1 に示す. 同じ結果を示したデータもあれば multi-jaccard 再識別の方が少し劣っているデータも存在している. これは, multi-jaccard の方が優れているという予想に反する結果だ. アルゴリズムの性質上, ランダムによる識別率の誤差が生じた可能性も考えられる. 結果データによる評価結果の違いについて議論する余地はある.

5.6.2 匿名加工手法について

提案匿名加工手法では, 既存のレコードを加工せず, 擬似レコードを追加するのみであった. 擬似レコード数が膨大になるというデメリットはあるが, 既存の履歴情報 (購買履歴データでは, 顧客が実際にある商品を購入したという事実) が残るというメリットが挙げられると考える.

ここで, 加工された購買履歴データをもとに自社商品をレコメンドする例を考えよう.

顧客 A:{ ゲーム } 関連商品をよく購入している人

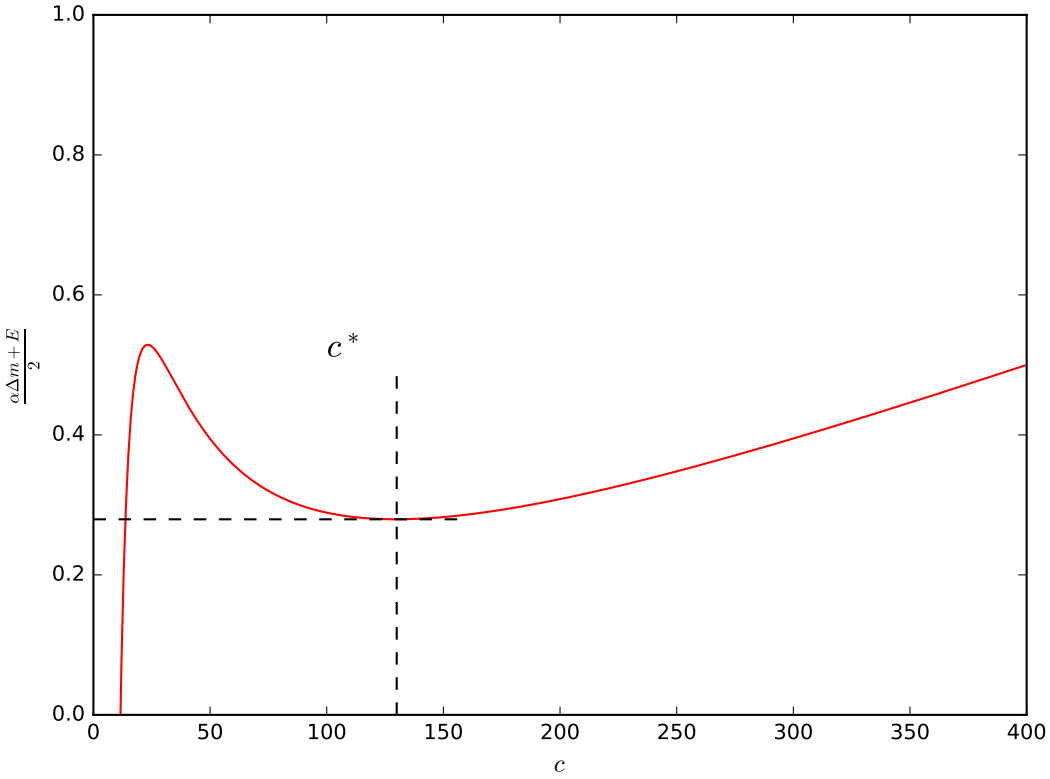

図 5.5　方式 2 の最適値 c^*

顧客 B:{ アウトドア } 関連商品をよく購入している人

と仮定する. ここで,

顧客 A:{ ゲーム } 関連商品をよく購入している人
顧客 B:{ ゲーム } 関連商品をよく購入している人

といったように既存レコードの削除や書き換えを行うと, 顧客 B には自分には全く関連のない商品がレコメンドされてしまう (=クレームに繋がる可能性). しかし, 提案手法に則って加工すると,

顧客 A:{ ゲーム, アウトドア } 関連商品をよく購入している人
顧客 B:{ ゲーム, アウトドア } 関連商品をよく購入している人

となり, 顧客 A,B 共にゲームとアウトドア関連商品のレコメンドを受けることとなる. つまり, 前者のように全く関係ない商品のレコメンドを受けることはない.

このような点で, 商品集合を統一するにあたり, ユースケースに応じて, レコードの追加, 削除, 書き換えを考えていかなければならない.

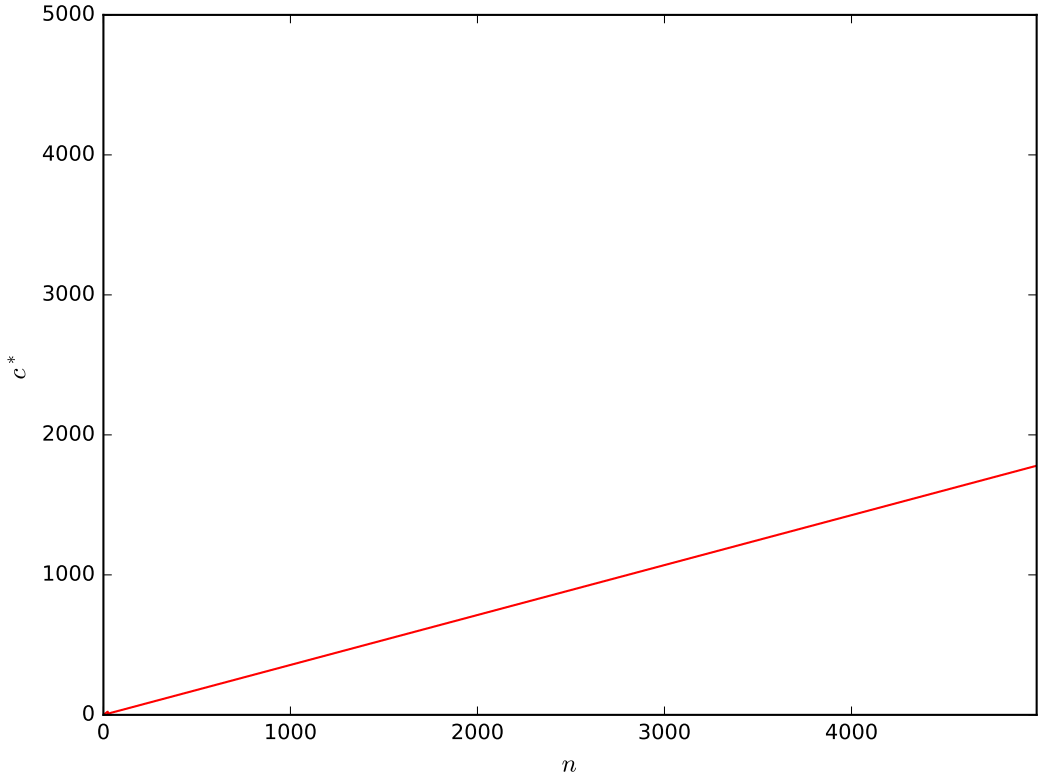

図 5.6　n に対する c の最適値 c^*

5.6.3　評価について

表 5.2 について, クラスタ数 c ごとの方式 1 と方式 2 における jaccard 類似度の変化は, 微小であった. 方式 2 によって, jaccard 類似度を大きく損失させず, Δm を最小限に抑えることができたが, 購買履歴データセットの jaccard 平均値 $\mu = 0.03$ と小さく, ほとんどの顧客の購入商品は相違していたことから, 顧客をクラスタ間で移動させたことによる jaccard 類似度への影響が小さかったと考えられる. つまり, 方式 2 による jaccard 類似度の変化は, データセットの特性に依存すると考え, 議論の余地がある.

第 6 章

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必要がある.

謝辞

本研究を進めるにあたり, ご指導を頂いた卒業論文指導教員の菊池浩明教授に感謝致します. また, 研究を通じて活発な議論にお付き合い頂いた菊池研究室伊藤聡志氏, 岡本健太郎氏, 田中司氏, 及び菊池研究室の皆様に感謝致します.

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.

[8] 原田玲央, 伊藤聡志, 菊池浩明, “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”, SCIS2017(発表予定) , 2017.

付録 A

プチ PWSCUP 評価プラットフォームの実装

匿名加工, 再識別手法の模索, 議論の促進, 理解を深めるために, 2016 年 8 月, 匿名加工班のメンバー (伊藤聡志, 岡本健太郎, 田中司) でプチ PWSCUP を開催した. プチ PWSCUP を円滑に進めるために, プチ PWSCUP 評価プラットフォームを実装し, 菊池研究室の学生 32 人から収集した実際の suica データを使用して, 匿名加工手法を競った. 本章では, 評価プラットフォームの概要と機能について述べる.

A.1 概要

加工データ, 有用性・安全性評価スクリプトの投稿, 評価値の算出を円滑に行えるように, サーバー上[*2]に評価プラットフォームを構築した.

ユーザは, 匿名加工データと指標スクリプトをアップロードする (図 A.2). プラットフォームは, 元データ, 加工データと評価スクリプトから, 有用性・安全性が評価され, 結果をデータベースに格納する. 評価結果は, リアルタイムに更新され, 結果画面から参照できる (図 A.3). 評価結果をデータベースに格納することにより, プチ PWSCUP の結果分析を容易に行えるようにした.

匿名加工データは csv 形式, 有用性・安全性評価指標スクリプトは R, python が対応している. 実装言語は, PHP, Javascript である.

A.2 機能

本節では, 評価プラットフォームの主な機能について述べる.

- ドラッグ&ドロップでのファイルアップロード機能
- 評価結果のリアルタイム反映
- ランキングソート機能
- 評価結果エクスポート機能

[*2] https://windy.mind.meiji.ac.jp/~reoh/evaluationPlatform

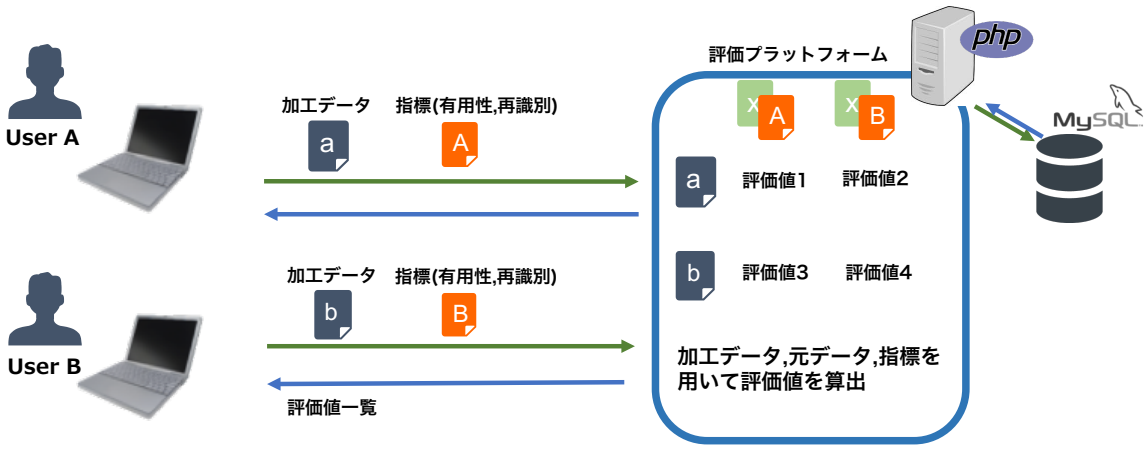

図 A.1　プチ PWS 評価プラットフォーム

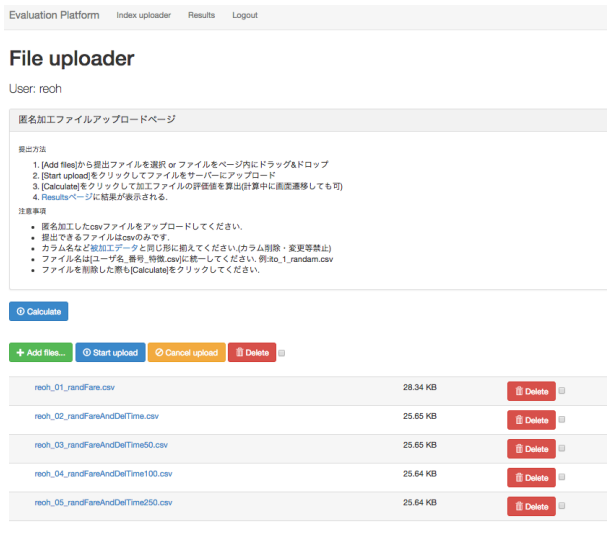

図 A.2　プチ PWS 評価プラットフォーム (投稿画面)

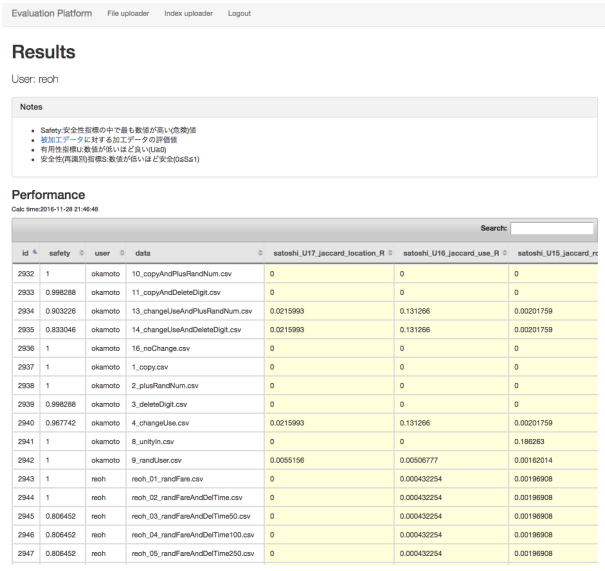

図 A.3　プチ PWS 評価プラットフォーム (結果画面)

付録 B

PWSCUP2015 可視化システムの実装

B.1 概要

2015 年 10 月, 匿名加工情報の技術を競う”匿名加工・再識別コンテスト”(PWSCUP) が初めて開催された [6]. 本戦競技中の匿名加工による防御と再識別による攻撃の様子を可視化することで各チームの攻守を盛り上げるために, PWSCUP2015 可視化システムの実装を行った.

本可視化システムは, クライアント上で動作するアプリケーションである. 匿名加工データの有用性と安全性を評価する匿名加工フェーズと, 各チームの加工データに対して正確に個人情報を復元する再識別フェーズについて, 提出された加工データに基づく評価結果を運営サーバーが算出し, 本可視化システムに反映させる.

B.2 匿名加工フェーズ

各チームは, あらかじめ用意された有用性指標と安全性指標の評価をできる限り下げないように, 元データから匿名加工データを生成し, サーバーに提出する.

提出された加工データの評価順位を可視化した 3 次元グラフを図 B.1 に示す. 立方体が加工データを表し, x 軸の低い方が有用性が高く, z 軸が高いほど安全であることを示している. y 軸は, 有用性と安全性の総合順位を表している.

B.3 再識別フェーズ

再識別に挑む攻撃者は, 有用性指標を元に匿名加工者がどう加工したかを推測していく. 本フェーズでは, 匿名加工フェーズで提出された加工データに対し, 再識別を行い, 識別に成功した行数の多さを競う. 本フェーズによって, 加工データの安全性は下がっていく.

本フェーズにおける可視化の様子を図 B.2 に示す. 六角柱は, 攻撃者の識別に成功した行数の多さを示し, 再識別に成功すると, そのチームを中心に波紋が広がる. それに伴って, 加工データの安全性を示す z 軸は低くなる.

本可視化システムは, コンテストの名称 Ice&Fire にちなんで, 識別成功による波紋のエフェクトを Fire, 安全性を表す z 軸を Ice と例え, Fire が Ice を溶かす関係をイメージした.

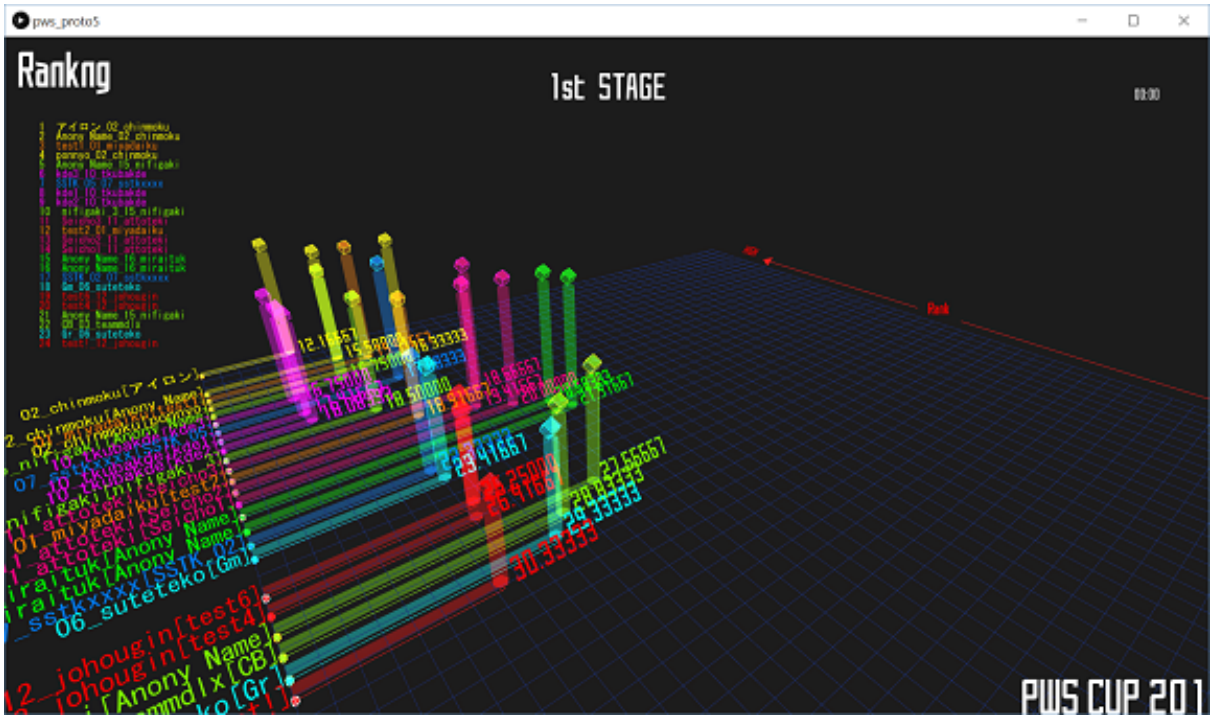

図 B.1　PWSCUP2015 可視化システム (匿名加工フェーズ)

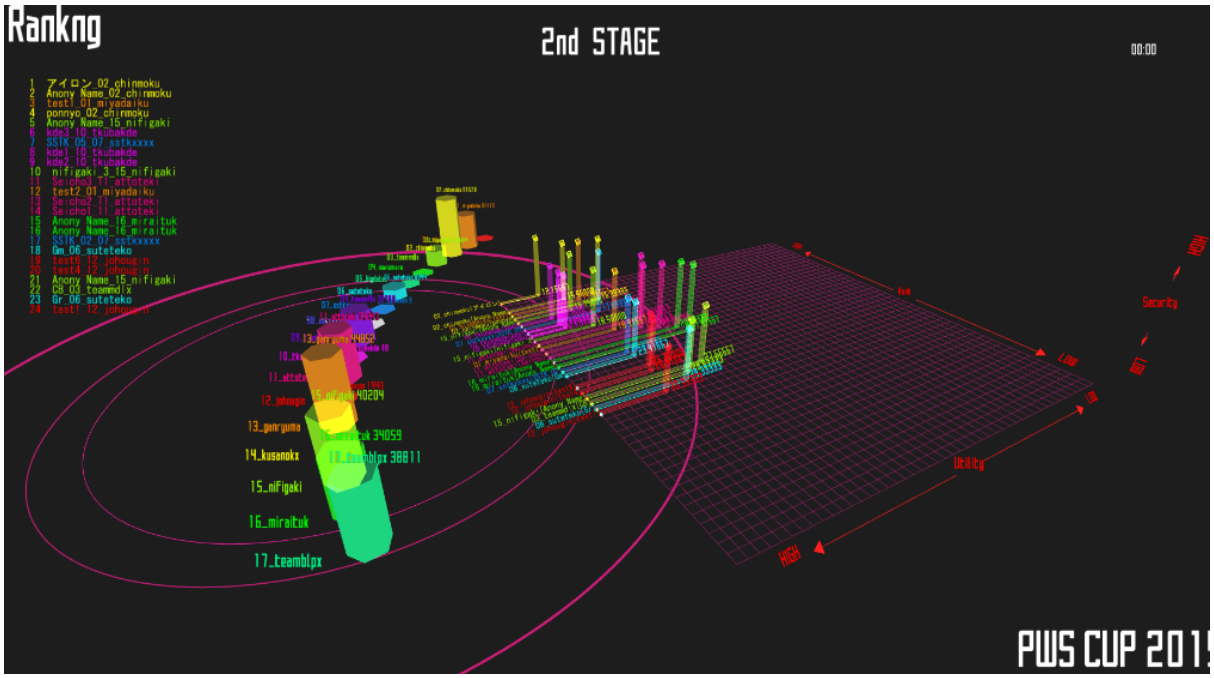

図 B.2　PWSCUP2015 可視化システム (再識別フェーズ)

付録 C

BLE デバイスからのプライバシー問題に関する安全性の評価

C.1 調査背景

私たちは常にスマートホンをはじめとするデバイスを持ち歩き, インターネットにいつでも接続できる環境下にある. さらに Bluetooth Low Energy(以後, BLE と呼ぶ) が普及したことにより容易に様々な情報の交換がされるようになった. 今後, イヤホンやスマートウォッチなどユーザー一人に対して身につける BLE デバイス数が増加することが想定される.

Bluetooth デバイスにはデバイス固有のアドレス (MAC アドレス) が振られているので, プライバシーに関するリスクも考えなければならない. 高木は, 山手線の車内で観測できる Bluetooth デバイスの MAC アドレスを収集することによって, 得られたデータから乗降パターンが追跡されてしまうリスクを提示している [1]. このように, デバイス固有の情報をスキャンすることで Bluetooth デバイスを利用しているユーザーの行動パターンというプライバシー関わる情報が第三者に漏洩してしまう恐れがある.

そこで, 近年普及してきている BLE に注目し, 通信から Bluetooth デバイスの情報がどのくらい取得できるか検証し, Bluetooth デバイスによるプライバシー問題に関する安全性について明らかにすることを本研究の目標とする.

C.2 Bluetooth 通信観測実験

C.2.1 概要

本実験は, PC やスマートホンなどのデバイスの BLE 通信をスプーフィングすることで, デバイスの情報を取得できるか検証する.

図 C.1 に本実験の構成を示す. アドバタイジングパケットをブロードキャストするビーコン A を用意し, それらを受け取ったデバイスが返送する ADV_SCAN_REQ パケットを CC2540 USB 評価モジュール・キット [2] で拾うことにより, 周囲にある全ての Bluetooth デバイスの MAC アドレスを収集する. CC2540 で収集したデータは専用のパケットスニッファ形式である psd ファイル (packet sniffer data) に保存される. 観測データをあらかじめ調べておいた Bluetooth デバイスの MAC アドレスのリスト (以後, 評価用データと示す) と照合し, 取得できるデバイス数を評価する.

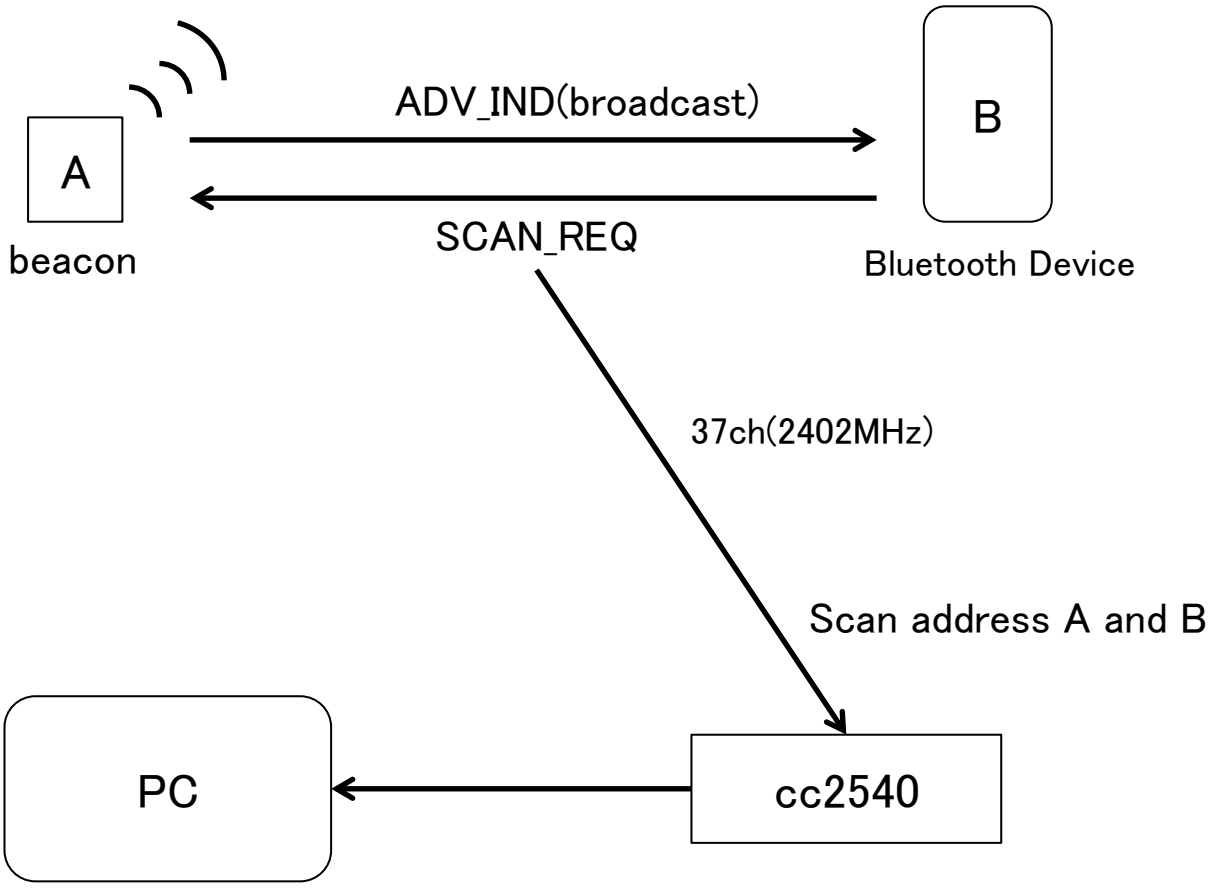

図 C.1　デバイス情報の収集方法

C.2.2　変換プログラム psd2csv

CC2540 を用いて取得できたパケットは独自フォーマットである psd 形式で保存される．本データを R などのツールを用いて解析できるように, csv として出力する psd2csv を processing で実装した．図 C.2 に psd2csv の変換の様子を示す．psd2csv は csv 形式の評価用データと BLE パケットが含まれる psd ファイルを入力として, 検出された ADV_SCAN_REQ パケットと検知されたデバイスを集計する csv ファイルを出力する．

C.2.3　データ形式

表 C.1 に CC2540 で収集した情報, C.2.2 節で示した評価用データ, psd2csv で出力する情報を示す．psd2csv は CC2540 が取得した情報から本実験で扱う情報のみを抽出し, 評価用データと結びつけて出力する．

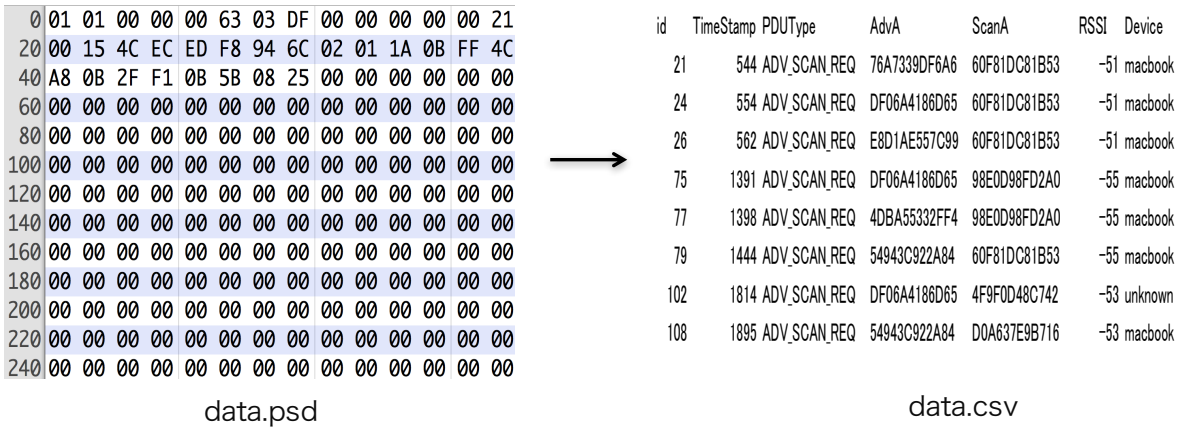

図 C.2　変換プログラムの入出力

表 C.1　各データファイルの取得情報

取得情報	CC2540	評価データ	psd2csv
Time stamp	○	×	○
PDU Type	○	×	○
Adv Address	○	×	○
Scan Address	○	○	○
RSSI	○	×	○
Access Address	○	×	×
PDU header	○	×	×
Channel	○	×	×
CRC	○	×	×
FCS	○	×	×
Device name	×	○	○
User	×	○	○

C.3　評価

C.3.1　取得できたデバイス数

2015 年 11 月 17 日に実験室にて研究室の学生が持つデバイス 19 台を対象に実験を行った．観測地点の周囲（半径 3m 以内）に Bluetooth デバイスをランダムに配置する．検証時間を 100s とし, BLE のアドバタイジングチャンネルである 37ch(2402MHz) におけるパケットを収集した．デバイスは Bluetooth を ON 状態とし, 観測中の位置は動かさない．

あらかじめ調査した対象デバイスの MAC アドレスに対して, BLE パケットから得られた Scan Address が一致したデバイスの数を表 C.2 の B に示す．BLE パケットから得られたデバイスは 4 台 (4/19=21%) である．デバイスの種類で見ると, MacBook と Mac mini が検出されたのに対して Android 端末や iPhone と

表 C.2　評価用データ及び検出数

デバイスの種類	デバイス数 (A)	検出数 (B)
Android	4	0
iPhone(iOS)	6	0
Mac Book (OS X)	4	3
Mac mini (OS X)	1	1
Laptop PC (Windows)	2	0
Mobile phone	1	0
Fit-bit	1	0

表 C.3　評価用データ及び検出数

デバイス	標準偏差 (C)	変動係数 (D)
Mac mini (a)	2.44	5.8%
Mac Book 1 (b)	3.40	7.1%
Mac Book 2 (c)	3.22	6.5%
Mac Book 3 (d)	6.79	11.9%

いったスマートホンやレガシー携帯電話は検出されなかった.

C.3.2　検出時間と RSSI

検出されたデバイスの時間変動に対する BLE 通信の受信信号強度 RSSI(Received Signal Strength Indication) の安定性を明らかにするため, C.3.1 節で psd2csv により取得したデータを R で解析した. 図 C.3 に, 検出されたパケットの RSSI の時間推移を示す. 評価用データとアドレスが一致したデバイスのパケットを・で示し, それ以外のアドレスのパケットを×で示す. また, 表 C.3 の C に RSSI の標準偏差, D に変動係数を示し, 観測デバイス a,b,c,d とする. デバイス d は, abc に比べて RSSI のばらつきが大きく, 周囲の人の動きなどの外的要因があると推測される. しかし, d を除いて固定されたデバイスの RSSI の標準偏差は 2.44 から 3.40 であり安定していると考えられるが, 高精度な位置関係を導出するには十分な安定性とはいえない. また, 評価用データとアドレスが一致しない unknown アドレスが検出された. Unknown アドレスは-90 dBm 付近に分布しており, 実験室外のデバイスが検出されたと推測される.

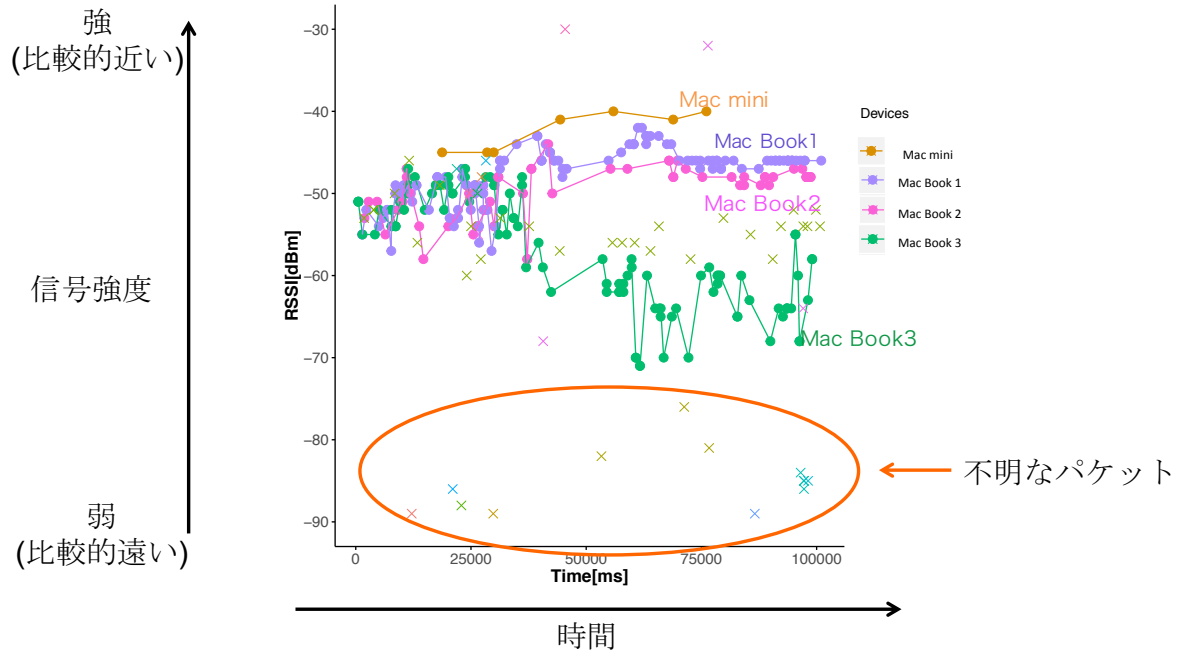

図 C.3　検出されたパケットの RSSI

C.4　まとめ

BLE の ADV_SCAN_REQ パケットから周辺デバイス情報の取得が可能であることを検証した．C.2 節での手法では一部のデバイスしか読み取ることができなかったが, Bluetooth デバイスからユーザーの行動パターンを読み取る脅威が存在するのではないかと考える．

一方, 本実験において検出されなかったデバイスが多数みられた．本実験は 3 つのアドバタイジングチャンネルのうち 1 つの通信経路における調査結果であり, 本実験で検出できなかったデバイスが他のチャンネルに流れている可能性があげられる．もしくは, 検出が容易にできないように BLE 規格が設計されているのではないかと考える．

参考文献

[1] 高木浩光, “Bluetooth で山手線の乗降パターンを追跡してみた”, (http://takagi-hiromitsu.jp/diary/20090301.html,2015 年 6 月参照).

[2] TEXAS INSTRUMENTS, “CC2540 USB 評価モジュール・キット”, (http://www.tij.co.jp/tool/jp/cc2540emk-usb,2015 年 5 月参照).

[3] 鄭立, “Bluetooth 入門”, 秀和システム, 2014.

[4] 折尾彰吾, 上田浩, 上原哲太郎, 津田侑, “ワイヤレスデバイスのもたらすロケーションプライバシー問題に関する一察”, CSS 2012, pp. 262-269, 2012.