

筑波大学大学院博士課程

システム情報工学研究科修士論文

最適化に基づくマージン付きサイズ均等
クラスタリングアルゴリズム

石田 紗知子

修士（工学）

（リスク工学専攻）

指導教員 遠藤 靖典

2017年3月

概要

クラスタリングとは、データ解析の一手法であり、外的基準なしに自動的に個体の集合の分類を行う、教師なし分類の一種である。この手法を用いることにより、対象となるデータ間の類似性や関連性を明らかにし、そのように分類されたデータから、有益な情報を得ることが出来るとされており、現在ではマーケティングなど幅広い場面で利用されている。

そのようなクラスタリング手法の1つにサイズ均等クラスタリングがある。サイズ均等クラスタリングは、各クラスタの持つ個体数が均等かつクラスタ内距離の総和を最小にするようなクラスタリング手法であり、最適化に基づく手法として、最適化に基づくサイズ均等クラスタリング (Even-sized Clustering Based on Optimization : ECBO) が提案された。ECBO は、 k -means の目的関数と制約条件に、新たにクラスタサイズを均等にするという制約を加え、シンプレックス法に基づき、目的関数の最適化を行うことにより、各クラスタに所属する個体数（クラスタサイズ）を均等にするクラスタリング手法である。通常のクラスタリング手法では、クラスタサイズを任意の均等な数に分割することが出来ないのに対し、ECBO では、任意のクラスタ数・クラスタサイズで均等に分割できるという利点がある。そのため、この手法は、荷物の配送計画を組む際に、配送対象となる住宅地域を分割するといったスケジューリング問題や、企業のタスクを分配するといった最適化問題に有用であると考えられる。

現在提案されている ECBO は、クラスタサイズを完全に均等にするという制約のもと分類を行うため、クラスタサイズにある程度のあそびを許し、完全に均等な個体数に分ける必要が無いような場合には、不便である。

そこで、本研究では、既存のクラスタリング手法のアルゴリズムを利用し、クラスタサイズに一定の幅を持たせた、最適化に基づくマージン付きサイズ均等クラスタリング (Even-sized with Margin Clustering based on Optimization : α -ECBO) のアルゴリズムの手法を提案する。

本論文では、サイズ均等クラスタリングの一種である K -Member Clustering (KMC)、既存手法である ECBO の手法・課題を示し、それらの問題に対するアプローチとして、既存のクラスタリング手法のアルゴリズムを利用した、最適化に基づくマージン付きサイズ均等クラスタリングの手法を 6 つ提案する。そして、人工データや Iris データ等の実データを用いた数値例を通して、提案する手法の有効性の評価・考察を行う。

目 次

第 1 章 序論	1
1.1 背景	1
1.2 目的	2
1.3 本論文の構成	2
第 2 章 クラスタリングの様々な手法	3
2.1 クラスタリングとは	3
2.2 類似性尺度	4
2.2.1 ユークリッド距離を用いた非類似性尺度	4
2.2.2 コサイン相関を用いた類似性尺度	5
2.2.3 L_1 ノルムを用いた非類似性尺度	5
2.3 k -means	6
2.4 k -means++	7
2.5 k -medoids	7
2.6 Kernel k -means	8
2.6.1 カーネル関数を利用したクラスタリング	9
2.7 L_1 k -means	10
第 3 章 サイズ均等クラスタリング	12
3.1 K -Member Clustering	12
3.1.1 K -匿名化について	12
3.1.2 Greedy K -member Clustering (GKC)	13
3.1.3 One-pass K -means Algorithm for k -anonymization (OKA)	13
3.1.4 Clustering-Based k -Anonymity (CBK)	14
3.2 最適化に基づくサイズ均等クラスタリング	15
第 4 章 提案手法	17
4.1 α -ECBO : k -meansに基づくアルゴリズム	17
4.2 α -ECBO++ : k -means++に基づくアルゴリズム	18
4.3 α -MECBO : k -medoidsに基づくアルゴリズム	19
4.4 α -KECBO : Kernel k -meansに基づくアルゴリズム	19
4.5 α - L_1 ECBO : L_1 ノルムを用いたアルゴリズム	19

4.6 α -cosine-ECBO : コサイン類似度を用いたアルゴリズム	20
第 5 章 数値例	24
5.1 Adjusted Rand Index	24
5.2 人工データ	24
5.2.1 密度の異なる円状のデータ	25
5.2.2 2重円データ	29
5.2.3 個体数が異なる標準正規分布に従ったデータ	32
5.2.4 密度・半径の異なる円状のデータ	37
5.2.5 3次元データ	41
5.3 実データ	43
5.3.1 Fisher's Iris データ	43
5.3.2 Wisconsin Breast Cancer データ	45
第 6 章 結論	48
謝辞	50
参考文献	51

図 目 次

5.1	密度の異なる円状のデータ（個体数 255 個）	25
5.2	k -means による分類結果	26
5.3	ECBO による分類結果	26
5.4	α -ECBO による分類結果 ($\alpha=27$)	27
5.5	α -MECBO による分類結果 ($\alpha=27$)	27
5.6	α -KECBO による分類結果 ($\alpha=38$)	27
5.7	α - L_1 ECBO による分類結果 ($\alpha=26$)	27
5.8	α -cosine-ECBO による分類結果 ($\alpha=10$)	27
5.9	α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.10	α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.11	α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.12	α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.13	α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.14	α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	28
5.15	α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	28
5.16	α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	28
5.17	α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	28
5.18	α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	29
5.19	α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	29
5.20	α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	29
5.21	2 重円データ（個体数 250 個）	29
5.22	Kernel k -means による分類結果	30
5.23	KECBO による分類結果	30
5.24	α -ECBO による分類結果 ($\alpha=15$)	31
5.25	α -ECBO++による分類結果 ($\alpha=11$)	31
5.26	α -MECBO による分類結果 ($\alpha=18,28$)	31
5.27	α -KECBO による分類結果 ($\alpha=39$)	31
5.28	α - L_1 ECBO による分類結果 ($\alpha=19$)	31
5.29	α -cosine-ECBO による分類結果 ($\alpha=1$)	31
5.30	α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32

5.31 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32
5.32 α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32
5.33 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32
5.34 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32
5.35 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	32
5.36 個体数が異なる標準正規分布に従ったデータ（個体数 380 個）	33
5.37 k -means による分類結果	34
5.38 ECBO による分類結果	34
5.39 α -ECBO による分類結果 ($\alpha=15 \sim$)	34
5.40 α -MECBO による分類結果 ($\alpha=15 \sim$)	34
5.41 α -KECBO による分類結果 ($\alpha=16$)	35
5.42 α - L_1 ECBO による分類結果 ($\alpha=16$)	35
5.43 α -cosine-ECBO による分類結果 ($\alpha=22$)	35
5.44 α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	35
5.45 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	35
5.46 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	35
5.47 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	36
5.48 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	36
5.49 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	36
5.50 α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	36
5.51 α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	36
5.52 α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	36
5.53 α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	36
5.54 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	36
5.55 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の 変遷	36
5.56 密度・半径の異なる円状のデータ（個体数 510 個）	37
5.57 k -means による分類結果	38
5.58 ECBO による分類結果	38
5.59 α -ECBO による分類結果 ($\alpha=20$)	38
5.60 α -MECBO による分類結果 ($\alpha=20$)	38
5.61 α -KECBO による分類結果 ($\alpha=25$)	39
5.62 α - L_1 ECBO による分類結果 ($\alpha=17$)	39
5.63 α -cosine-ECBO による分類結果 ($\alpha=28$)	39
5.64 α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	39
5.65 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	39
5.66 α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	39
5.67 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	40

5.68 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	40
5.69 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	40
5.70 α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	40
5.71 α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	40
5.72 α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	40
5.73 α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	40
5.74 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	40
5.75 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の 変遷	40
5.76 球状の 3 次元データ（個体数 411 個）	41
5.77 α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.78 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.79 α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.80 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.81 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.82 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	42
5.83 α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	43
5.84 α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	43
5.85 α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	43
5.86 α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	43
5.87 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	43
5.88 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の 変遷	43
5.89 α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.90 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.91 α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.92 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.93 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.94 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	44
5.95 α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	45
5.96 α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	45
5.97 α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	45
5.98 α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	45
5.99 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	45
5.100 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の 変遷	45
5.101 α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46
5.102 α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46

5.103 α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46
5.104 α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46
5.105 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46
5.106 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷	46
5.107 α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	47
5.108 α -ECBO++を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	47
5.109 α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	47
5.110 α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	47
5.111 α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷	47
5.112 α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の 変遷	47

表 目 次

2.1 本論文における記号表記一覧	4
5.1 Comparing partitions U and V	24
5.2 密度の異なる円状のデータに対する各手法の実行結果	26
5.3 2重円データに対する各手法の実行結果	30
5.4 個体数が異なる標準正規分布に従ったデータに対する各手法の実行結果	34
5.5 密度・半径の異なる円状のデータに対する各手法の実行結果	38
5.6 3次元データに対する各手法の実行結果	42
5.7 Fisher's Iris データに対する各手法の実行結果	44
5.8 Breast Cancer データに対する各手法の実行結果	46

第1章 序論

1.1 背景

インターネットやネットワークを活用したサービス・技術の普及に伴い、誰でも気軽にネットワークと繋がり、ITを活用できるような時代となってきている。それに伴い、企業は、ITを活用して得られた情報を様々なビジネスに役立てようとする動きが活発になってきている。そういった流れから、ビッグデータという言葉が生まれた[5]。ビッグデータとは、誰でも簡単に繋がることの出来るようなインターネット環境の普及と、IT技術の進化によって生まれた言葉であり、企業などがこれまで扱ってきたデータ以上に、より大容量かつ多種多様なデータのことを指す。そのようにして扱われるビッグデータの種類は非常に多岐に渡り、数値や文字列、電子メール、通信ログといったデータや、文章、音声、動画などマルチメディアデータなど様々なものが含まれている。そして、このようなデータを活用することで、顧客のマーケティングリサーチなどを行う動きが、Amazon[2], Facebook[3]などでより活発になってきている。顧客データを分析し、活用しているAmazonのリコメンデーションシステムやトヨタ自動車のビッグデータ交通情報サービスなどが一例である[4]。また、最近では、人工知能の技術開発にもビッグデータを利用した研究が盛んである。

このように、膨大なデータを分析して価値を生み出すために、各データの属性間の類似性に基づき分析を行うデータ分析手法の一つとして、クラスタリングがある[1]。クラスタリングとは、データ解析手法の一つであり、外的基準なしに自動的に分類を行う手法である。通常のクラスタリング手法では、クラスタ数 c を与え、各個体間の類似度に基づきグループ分けを行う。

ここで、個人の購買履歴や住所などのパーソナルなビッグデータが与えられた際、個人情報を含むため、その活用には十分な配慮が必要である。そして、政府は約10年ぶりとなる2015年に、個人情報保護法を改定し、個人を特定できないような状態での情報の公開や利用を促進しようとしている[6]。そのような条件を満たすデータ分析手法の一つに、 K -匿名化がある[7]。 K -匿名化とは、あるデータから特定の個人の識別を困難にするデータ処理技術の一つであり、ビッグデータ活用の動きが盛んになるにつれ、プライバシー保護の有益な手段として、重要性が高まっている[8][9][10][11]。 K -匿名性を満たすとは、属性内で同じ組み合わせを持つような個体が K 個以上存在するということをいう。このような匿名化技術への応用を目的として提案されているクラスタリング手法の一つに、サイズ均等クラスタリングがある。サイズ均等クラスタリングの一種である、 K -Member Clustering(KMC)とは、各クラスタのクラスタサイズを K 以上にするかつクラスタ内距離の総和が最小となるようにデータセットの分割を行うクラスタリング手法である[14]。従来のKMCの手法は、クラスタサイズが

K 個となるように個体の分割を行うことが出来るが、最適化に基づく手法はなかった。そこで、最適化に基づく手法として、Even-sized Clustering Based on Optimization(ECBO) [15] が提案された。ECBO は、目的関数および制約条件を設定し、個体の分類を行うことで、均等なクラスタサイズを持つようなグループ分けを可能とするような手法である。しかしながら、現在提案されている ECBO は、クラスサイズを各クラスタ間で完全に均等にするという制約条件に基づき分類を行うため、ある程度幅を持たせた分類結果を得たい場合には、不便であり、不自然な分類結果となってしまう点が問題である。

1.2 目的

本研究では、既存のサイズ均等クラスタリングの一種である ECBO に対し、新たにクラスタサイズに幅を持たせるという概念を導入した新たなアルゴリズムを提案する。それにより、ECBO で挙げられた課題である、クラスタサイズにある程度あそびを許したいような場合の、データセットの分類を可能にする。

提案する手法は、既存研究である ECBO に対し、他のクラスタリング手法のアルゴリズムを基にしたものとなっている。提案手法に用いたクラスタリングアルゴリズムは、 k -means, k -means++, k -medoids, Kernel k -means, $L1$ k -means, cosine k -means の 6 つの手法である。それぞれの手法を導入する理由としては、非線形なデータの分類、ノイズを含むデータ、初期値依存に対する課題の解決として種々のクラスタリング手法によるアプローチを考える。これらのクラスタリング手法を導入した提案手法を、人工データ・実データを用いて結果の比較をし、手法の評価を行う。

1.3 本論文の構成

本論文では、まず、第 2 章でクラスタリングの様々な手法・個体間の類似度の定義について示す。次に、第 3 章では、関連研究であるサイズ均等クラスタリングや K -匿名化について述べるとともに、既存研究である ECBO のアルゴリズム・手法の説明を行う。第 4 章では、ECBO のクラスタサイズに関する制約条件に、新たに幅を持たせるという概念を導入した、6 つのクラスタリング手法の提案を行う。第 5 章では、提案した手法と既存のクラスタリング手法を、人工データ・実データ等の数値例を用いて比較を行うことで、アルゴリズムの評価・考察を行う。最後に、第 6 章では、本研究により得られた結果についてのまとめを述べる。

第2章 クラスタリングの様々な手法

本章では、まず、クラスタリングの概要について説明するとともに、現在提案されている一般的なクラスタリング手法について、アルゴリズムを用いて各手法の違いを説明する。

2.1 クラスタリングとは

クラスタリングとは、データ解析手法の一つであり、外的基準なしに自動的に分類を行う手法である。

外的基準なしに自動的に分類を行う方法は、パターン認識の分野では、教師なし分類と呼ばれる。その一方で、外的基準のもとで分類を行う方法は、教師あり分類と呼ばれる。

クラスタリングは、外的基準がないため、分類対象となる個体の集合であるデータセットの間に定義された類似性や距離に基づいて、クラスタと呼ばれるグループにデータの集合を分割することである。

クラスタリングの方法は、[1]において、次のように定められている。

分類すべき個体の集まりにおいて、任意の2つの個体間に類似性あるいは非類似性を表す測度が与えられていると仮定する。クラスタリングとは、個体の集まりをいくつかのクラスタ（部分集合）に分割し、それぞれのクラスタの中では個体同士の類似度が大きく、異なるクラスタについては類似度が小さくなるようにすることである。

クラスタリングの手法は、大きく分けると、最短距離法や最長距離法などの階層的クラスタリング（Agglomerative Hierarchical Clustering:AHC）と k -means や Fuzzy-c-means などの非階層的クラスタリングの2つに分類される。

ここで、本論文において提案しているクラスタリング手法は、非階層的クラスタリングに基づく手法であり、その中の代表的な手法である k -means クラスタリングのように、クラスタ数 c をあらかじめ指定し、個体を c 個のクラスタに分割するという分類を、設定した目的関数の値を最小化するようにクラスタリングを行う。そのため、本論文では、階層的クラスタリングに関する詳細な説明は省略する。

以下では、クラスタリング分析において、データ間の関連性の尺度を定義するために用いる類似性尺度について説明するとともに、非階層的クラスタリングにおける代表的な手法について紹介する。

これ以降、クラスタリングの対象となる個体を $\{x_1, x_2, \dots, x_n\}$ で表す。また、これらの個体の集まりを X で表し、 $X = \{x_1, x_2, \dots, x_n\}$ となる。

本論文で用いる文字の定義を表2.1に示す。

表 2.1: 本論文における記号表記一覧

x_k ($k = 1, \dots, n$)	: クラスタリングの対象となる個体
n	: 個体数
$X = \{x_1, \dots, x_n\}$: 個体の集まり
c	: クラスタ数
C_i ($i = 1, \dots, c$)	: i 番目のクラスタ
v_i	: クラスタ i のクラスタ中心
u_{ki}	: 個体 k のクラスタ i への帰属度
K	: クラスタサイズ

2.2 類似性尺度

クラスタリングのアルゴリズムは、類似度あるいは非類似度の計算とクラスタ生成の 2 つのステップに大別される。ここで、類似性尺度とは、個体を 2 つ与えたとき、その個体間に對して決まる実数であり、類似度 (similarity) は、その値 $s(x, y)$ が大きければ大きいほど個体 x, y は類似しており、非類似度 (dissimilarity) の場合、その値 $d(x, y)$ が小さければ小さいほど、2 つの個体 x, y は互いに類似しているということになる。類似度と非類似度の関係性は (2.1)~(2.4) のように定義される [1]。

$$s(x, x) \geq s(x, y) \quad \forall y \in X \quad (2.1)$$

$$d(x, x) \leq d(x, y) \quad \forall y \in X \quad (2.2)$$

$$s(x, y) = s(y, x) \quad \forall x, y \in X \quad (2.3)$$

$$d(x, y) = d(y, x) \quad \forall x, y \in X \quad (2.4)$$

データの型は、2 値データ (binary data)、名義データ (nominal data)、実数データ (real data) のように区別されるが、本研究で用いた数値例のデータは、個体 x_i が、ある属性に対してとるスコアが実数値もしくは整数値となる実数データのみであるため、ここでは、実数データの非類似度尺度としてよく用いられるユークリッド距離を用いた非類似度と距離以外の類似性尺度としてよく用いられるコサイン類似度、各ベクトルの各成分の絶対値の和で表される L_1 ノルムについて述べる。

2.2.1 ユークリッド距離を用いた非類似性尺度

2 つの個体 $x, y \in X$ における非類似度を $d(x, y)$ とする。個体 $x, y \in X$ が実ベクトル空間の点のとき、その空間に定義された距離が個体 x, y の非類似性となる。個体間の距離尺度の計算で最もよく用いられるのがユークリッド距離であり、 x と y の距離は、(2.5) のように表される。

$$\|x - y\| = \sqrt{(x - y)^2} \quad (2.5)$$

ここで, $\|\cdot\|$ はベクトル \cdot のユークリッドノルムを表す.

このユークリッド距離をクラスタリングのための非類似性尺度として用いると,

$$d(x, y) = \|x - y\| \quad (2.6)$$

となる.

また, これ以降で紹介する k -means クラスタリングや本論文で提案する手法等では, ユークリッド距離の 2 乗を個体間の非類似度としており, その場合,

$$d(x, y) = \|x - y\|^2 = (x - y)^2 \quad (2.7)$$

となり, この値が小さければ小さいほど, 個体 x と y は類似しているということになる.

2.2.2 コサイン相関を用いた類似性尺度

実数データに対する個体間の関連性を定義する尺度として, ユークリッド距離以外でよく用いられるものに, 角度と相関を用いたコサイン類似度がある.

ベクトル x と y が p 次元ユークリッド空間でなす角を $\theta_{x,y}$ とすると,

$$\cos \theta_{x,y} = \frac{\langle x, y \rangle}{\|x\| \|y\|} = \frac{\sum_{j=1}^p x^j y^j}{\sqrt{\left\{ \sum_{j=1}^p (x^j)^2 \right\} \left\{ \sum_{j=1}^p (y^j)^2 \right\}}} \quad (2.8)$$

である. このコサイン相関をクラスタリングのための類似性尺度として用いると,

$$s_{\cos}(x, y) = \cos \theta_{x,y} \quad (2.9)$$

となり, この類似度 $s_{\cos}(x, y)$ をコサイン類似度と呼び, この値が大きければ大きいほど, 個体 x と y は類似しているということになる.

2.2.3 L_1 ノルムを用いた非類似性尺度

ユークリッド距離のように, 2 点間を直線で結ぶのではなく, 格子状に分かれたマス目に沿って移動するときの距離を表す L_1 ノルム (マンハッタン距離) がある. ユークリッド距離が各ベクトルの成分の差の 2 乗和の平方根で表されるのに対し, L_1 ノルムは, 各ベクトルの各成分の差の絶対値の和で表される.

ベクトル $\mathbf{x} = (x_1, x_2)$, $\mathbf{y} = (y_1, y_2)$ の距離は, L_1 ノルムを用いると,

$$\|\mathbf{x} - \mathbf{y}\| = |(x_1 - y_1)| + |(x_2 - y_2)| \quad (2.10)$$

のように表される.

L_1 ノルムを非類似性尺度として用いた場合, この値が小さければ小さいほど個体 \mathbf{x} と \mathbf{y} は類似しているということになる. L_1 ノルムを非類似度として用いる利点としては, ユークリッド距離の 2 乗を用いる場合よりも, 外れ値の影響を受けづらい点が挙げられる.

2.3 *k*-means

k-means クラスタリングは、非階層的クラスタリングの最も代表的な手法である [12]. この手法では、あらかじめクラスタ数 c を設定し、 c 個のクラスタ中心、あるいは各個体の帰属度をランダムに与え、各個体とそれが属するクラスタのクラスタ中心との距離の総和が最小となるようにデータセットの分割を行う。

目的関数 J は、(2.11) のように定義され、この J を最小化するようにクラスタ分割の最適化を行う。

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|x_k - v_i\|^2 \quad (2.11)$$

また、制約条件は(2.12) のように表される。ここで、 u_{ki} は、個体 x_k のクラスタ C_i への帰属度を表し、個体 x_k がクラスタ C_i に属する場合、 $u_{ki} = 1$ 、属さない場合 $u_{ki} = 0$ である。

$$\text{s.t.} \quad \sum_{i=1}^c u_{ki} = 1 \quad (k = 1, \dots, n) \quad (2.12)$$

k-means クラスタリングのアルゴリズムとしては、最初に c 個のクラスタ中心の選択、あるいは初期帰属度をランダムに与える。その後、目的関数 (2.11) を最小化するように、各個体のクラスタへの割り当ての最適化と、クラスタ中心の最適化を交互に行う。

クラスタ中心は、そのクラスタの代表点であり、クラスタに属する個体の重心となる。クラスタ中心の計算式は(2.13) のように求めることが出来る。

$$v_i = \frac{\sum_{k=1}^n u_{ki} x_k}{\sum_{k=1}^n u_{ki}} \quad (2.13)$$

k-means クラスタリングのアルゴリズムを Algorithm 1 に示す。

Algorithm 1 *k*-means

- 1: 初期のクラスタ中心を c 個、もしくは帰属度をランダムに与える。
 - 2: 各個体を最も近いクラスタ中心に割り当てる。
 - 3: クラスタ中心 V を(2.13) で更新する。
 - 4: 前回と V が変化しない、もしくはすべての個体の割り当てがひとつ前のステップと変化しなければ終了。そうでなければ、更新した各クラスタの中心を新しい中心として、Step3 に戻る。
-

k-means 法の特徴としては、個体のクラスタへの割り当てとクラスタの代表点であるクラスタ中心の更新を交互に繰り返し、目的関数を最小化するというように交互最適化を行い、データセットのクラスタ分割を行うという点である。また、*k*-means クラスタリングは、最初にランダムに与えるクラスタ中心もしくは個体の帰属度の初期値によって結果が大きく影響されるということが挙げられる [13]。また、各クラスタに含まれる個体数がある程度均等になるように分割されるという特徴もある。

2.4 k -means++

Algorithm 1 では、初期クラスタ中心もしくは初期のクラスタ分割をランダムに与えて、最も近いクラスタ中心に各個体が分類されるように、クラスタリングを行っている。しかしながら、ランダムに選択された初期値のために、その与えられるクラスタ中心や帰属度の初期値によっては、クラスタリングの結果に悪い影響を及ぼす場合もある。そこで、 k -means の初期値依存の問題を軽減する手法として提案されたのが、 k -means++ [16] である。 k -means++ は、初期のクラスタ中心を選択する初期値計算フェーズとデータセットの分割を行うクラスタリングフェーズに分けられる。 k -means クラスタリングとの違いとしては、初期値依存の問題の軽減のために、すでに選択されたクラスタ中心との最短距離が大きいデータが新しいクラスタ中心として選択されるように、(2.14) の大きい順に、確率的に初期のクラスタ中心を選択する。クラスタ中心を選択後は、通常の k -means と同様にクラスタリングを行う。

k -means++ の初期値計算フェーズとクラスタリングフェーズのアルゴリズムを Algorithm 2, Algorithm 3 に示す。

Algorithm 2 k -means++ (初期値計算フェーズ)

- 1: ランダムに個体を 1 つ選び、クラスタ中心 v_1 とする。
- 2: (2.14) の確率で個体を 1 つ選び、新しいクラスタ中心 v_i とする。

$$\frac{D(x)^2}{\sum_{x \in X} D(x)^2} \quad (2.14)$$

ここで、 $D(x)$ は個体 x とすでに選択した最も近いクラスタ中心との距離である。

- 3: c 個のクラスタ中心を選ぶまで Step 2 を繰り返す。
-

Algorithm 3 k -means++

- 1: クラスタ数 c を設定する。
 - 2: 初期クラスタ中心 $V = (v_i)$ ($i = 1, \dots, c$) を Algorithm 2 により決定する。
 - 3: 帰属度 $U = (u_{ki})$ ($k = 1, \dots, n, i = 1, \dots, c$) を更新する。
 - 4: クラスタ中心 V を更新する。
 - 5: 前回と V が変化しなければ終了。そうでなければ、Step 4 へ。
-

2.5 k -medoids

k -means は、クラスタ中心を各クラスタに所属する個体の重心とする手法である。そのため、ノイズや外れ値の影響を受けやすい。そこで、その問題を軽減する手法として提案されたのが、 k -medoids クラスタリングである [17]。 k -medoids は、medoid と呼ばれるクラスタ内を代表する個体そのものを各クラスタの中心とする手法である。そのため、 k -means に比べ、

ノイズや外れ値の影響を受けにくいとされている。また、個体そのものがクラスタ中心となるため、各個体間の距離さえ与えられていればクラスタリングを行うことが出来る。

medoid とは、各クラスタにおいて、任意の個体とその他のクラスタ C_i に属する全ての個体とのユークリッド距離の総和が最も小さくなるように選択された個体である。 k -medoids クラスタリングでは、 k -means クラスタリングと同様、medoids の更新と個体の帰属度の更新を交互に行うことで、最適化を行う。

k -medoids クラスタリングのアルゴリズムを Algorithm 4 に示す。

Algorithm 4 k -medoids

1: 初期値選択

1. 全ての個体同士の距離 $\|x_i - x_j\|$ を計算する。
2. 全ての個体 x_j に関して、 w_j を (2.15) を用いて計算する。

$$w_j = \sum_{i=1}^n \frac{\|x_i - x_j\|}{\sum_{k=1}^n \|x_k - x_j\|} \quad (2.15)$$

3. w_j を昇順にソートし、 w_j の値が最も小さい k 個の個体を初期のメドイドとして選択する。
4. 各個体を最も近いメドイドのクラスタに所属させる。
5. すべての個体とそれが所属するクラスタのメドイドとの距離の総和を求める。

2: メドイドの更新

各クラスタにおいて、任意の個体とその他のクラスタに属する全ての個体との距離の総和が最も小さい個体を選び、新たなメドイドとする。

3: 個体の帰属の更新

1. 各個体に最も近いメドイドのクラスタに所属させる。
 2. 全ての個体とそれが所属するクラスタのメドイドとの距離の総和を求める。
-

2.6 Kernel k -means

k -means クラスタリングでは、データセットは線形分離の形となるように分割される。そのため、非線形なデータセットの場合、上手くクラスタリングを行うことが出来ない。しかしながら、実データの中には線形分離できないものも多い。そこで、そのような問題を解決するために提案された手法が Kernel k -means クラスタリングである [18]。Kernel k -means は、個体をカーネル関数と呼ばれる非線形関数によって高次元に写像し、クラスタリングを行う。

この手法では、個体を元の空間より高次元の特徴空間に写像し、クラスタリングを行うことで、非線形な分割を行えるようにする手法である。つまり、データ点を $x \in X$ 、非線形関数を ϕ としたとき、 $\phi(x)$ に対してクラスタリングを行う。非線形関数であるカーネル関数には、以下に示すような (2.16), (2.17), (2.18) がある。

$$\text{多項式カーネル } \kappa(x, y) = (1 + \langle x, y \rangle)^d \quad (2.16)$$

$$\text{ガウシアンカーネル } \kappa(x, y) = \exp(-\gamma \cdot \|x - y\|^2) \quad (2.17)$$

$$\text{シグモイドカーネル } \kappa(x, y) = \tanh(\gamma \langle x, y \rangle + \theta) \quad (2.18)$$

ここで、 $\gamma > 0$ は、データの分散を調整するパラメータである。

2.6.1 カーネル関数を利用したクラスタリング

カーネル法を用いたクラスタリングでは、個体を高次元特徴空間に写像してから個体間の非類似度を計算し、クラスタリングを行う。高次元特徴空間における非類似度 $d_\Phi(x, y)$ は、

$$d_\Phi(x, y) = d(\Phi(x), \Phi(y)) = \|\Phi(x) - \Phi(y)\|^2 \quad (x, y \in X) \quad (2.19)$$

で表される。しかしながら、高次元特徴空間内での x と y の距離の 2 乗は、カーネル関数を用いることで、写像 Φ の形が分からなくても求めることが出来る。すなわち、

$$d_\Phi(x, y) = \|\Phi(x) - \Phi(y)\|^2 \quad (2.20)$$

$$= \|\Phi(x)\|^2 - 2 \langle \Phi(x), \Phi(y) \rangle + \|\Phi(y)\|^2 \quad (2.21)$$

$$= \langle \Phi(x), \Phi(x) \rangle - 2 \langle \Phi(x), \Phi(y) \rangle + \langle \Phi(y), \Phi(y) \rangle \quad (2.22)$$

$$= \kappa(x, x) - 2\kappa(x, y) + \kappa(y, y) \quad (2.23)$$

よって、Kernel k -means における非類似度は、(2.23) の x, y に個体の座標を入力すれば求められることが分かる。ゆえに、高次元特徴空間での個体 x_k とクラスタ中心 v_i^Φ の距離は、(2.24) のように計算することが出来る。

$$\|\Phi(x_k) - v_i^\Phi\|^2 = \kappa(x_k, x_k) - \frac{2}{|C_i|} \sum_{x_l \in C_i} \kappa(x_k, x_l) + \frac{1}{|C_i|^2} \sum_{x_l \in C_i} \sum_{x_m \in C_i} \kappa(x_l, x_m) \quad (2.24)$$

Kernel k -means クラスタリングのアルゴリズムを Algorithm 5 に示す。

Algorithm 5 Kernel k -means

- 1: クラスタ数 c を設定する.
- 2: 初期帰属度 U をランダムに設定する.
- 3: U の値を用いて, (2.24) を更新する.
- 4: 更新した非類似度を用いて, (2.25) を解き, U を更新する.

$$\text{minimize} \quad J(U, V) = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|\Phi(x_k) - v_i^\Phi\|^2 \quad (2.25)$$

- 5: 前回のステップと U が変化しなくなれば終了. そうでなければ, Step3 へ.
-

2.7 $L_1 k$ -means

$L_1 k$ -means は, k -means の類似性尺度に L_1 ノルムを用いた非階層的クラスタリング手法の一つである. L_1 ノルムは, 各次元での個体間の距離の差の絶対値の総和によって定義される. 非類似度にユークリッド距離の 2 乗を用いたアルゴリズムに対し, L_1 ノルムは, 外れ値に対してロバストであるとされている. L_1 ノルムを用いた k -means クラスタリングは, 距離尺度にユークリッド距離を用いた k -means と同様, 目的関数 J を最小化するよう, クラスタ中心 V と帰属度 U の交互最適化を行う. $L_1 k$ -means の目的関数を以下に示す.

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|x_k - v_i\| \quad (2.26)$$

$$= \sum_{k=1}^n \sum_{i=1}^c \sum_{j=1}^p |x_{kj} - v_{ij}| \quad (2.27)$$

ここで, 帰属度 U を固定したときの V の最適化では, 各クラスタの各成分ごとに独立で最適化を行う. よって, 準目的関数は (2.28) のように定義される.

$$J_{ij} = \sum_{k=1}^n u_{ki} |x_{kj} - v_{ij}| \quad (2.28)$$

(2.28) を用いると, 目的関数 (2.27) は, (2.29) のように書き換えられる.

$$J = \sum_{i=1}^c \sum_{j=1}^p J_{ij} \quad (2.29)$$

よって, 準目的関数 J_{ij} が最適となるとき, 目的関数 J も最適となる. Algorithm 6 に $L_1 k$ -means のアルゴリズムを示す.

Algorithm 6 $L_1 k$ -means (クラスタリングフェーズ)

- 1: クラスタ数 c を設定する.
 - 2: 初期クラスタ中心 $V = (v_i) (i = 1, \dots, c)$ をデータセットの中からランダムに選択する.
 - 3: (2.29) を最小化するように、帰属度 $U = (u_{ki}) (k = 1, \dots, n, i = 1, \dots, c)$ を更新する.
 - 4: クラスタ中心 V を、Algorithm 7 を用いて更新する.
 - 5: 前回と V が変化しなければ終了。そうでなければ、Step 4 へ.
-

Algorithm 7 $L_1 k$ -means (クラスタ中心 V 最適化フェーズ)

- 1: 個体 x_{kj} を各次元 j ごとに昇順にソートし、 $x_{q(k)j}$ とする.
 - 2: $S = -\frac{1}{2} \sum_{k=1}^n u_{q(k)j}, r = 0$ とする.
 - 3: S の符号が $-$ から $+$ に変わるまで、 $S = S + u_{q(k)j}, r = r + 1$ を計算する.
 - 4: 最適解 $v_{ij} = x_{q(r)j}$ を更新する.
-

第3章 サイズ均等クラスタリング

サイズ均等クラスタリングとは、各クラスタの持つ個体数が均等かつクラスタ内距離を最小にするようなクラスタリング手法であり、KMCを基として提案されたクラスタリング手法である。*k-means*等の通常のクラスタリング手法では、クラスタ数のみを指定し目的関数を最小化するようにデータセットの分割を行うため、各クラスタに含まれる個体数（クラスタサイズ）を全てのクラスタ間で均等に分割することは出来ない。しかしながら、サイズ均等クラスタリングの場合、帰属度に関する制約条件に加え、クラスタサイズに関する制約を設定することで、クラスタサイズを全てのクラスタ間で均等にするよう分割を行うことが出来る。

以下では、サイズ均等クラスタリングに関連する研究である *K-Member Clustering* (KMC) の代表的な手法 3 つの特徴とアルゴリズム、*K*-匿名化について説明する。そして、サイズ均等クラスタリングに関する既存研究である最適化に基づく手法について述べる。

3.1 *K*-Member Clustering

K-Member Clustering (KMC) は、各クラスタのクラスタサイズを K 以上にするかつクラスタ内距離の総和が最小となるようにデータセットの分割を行うクラスタリング手法である [14]。一般的に、プライバシー保護に関する重要な性質である、*K* 匿名性を確保することが出来るという利用方法で用いられる。以下では、*K*-匿名化および KMC の代表的な手法 3 つについて述べる。

3.1.1 *K*-匿名化について

K-匿名化とは、 K 人以上が同じ準識別子を持つようなデータへ元のデータを変換することで、属性内で同じ組み合わせを持つデータが K 個以上存在するような状態を *K*-匿名性を満たすという [8]。

ここで、名前やマイナンバーなどその属性の値だけで個人を直接特定することが可能な属性のことを識別子といい、性別、年齢などそれ単体では個人を特定することが出来ないが、それらの属性を複数組み合わせることで個人の特定が可能となるような属性群のことを準識別子という [11]。

このような匿名化技術は、企業や様々な組織が、マーケティング等の様々なサービスに個人情報や顧客データなどを利用する際に、データ内の情報から個人を特定できないような形でプラ

イバシーを守りながら用いるための、プライバシー保護データマイニング（Privacy-Preserving Data Mining）の分野で活用されている。

3.1.2 Greedy K-member Clustering (GKC)

GKC は、Byun ら [20] によって提案された KMC の手法である。この手法では、まず初期値をひとつランダムに選択し、この個体に対する非類似度の小さい個体 K 個を選択し、それをひとつのクラスタとする。そして、クラスタに属していない個体をランダムに選択し、最も近い個体に所属させるということを、すべての個体がどれかひとつのクラスタに属するまで繰り返す。GKC のアルゴリズムを Algorithm 8 に示す。

Algorithm 8 GKC

- 1: クラスタサイズ K を設定する。
 - 2: データセットの中から個体ひとつをランダムに選択し、 r とする。
 - 3: r に対し、非類似度の小さい個体 K 個を同じクラスタ C_i とする。
 - 4: クラスタに属していない個体数が K 個未満なら、Step 5へ。そうでなければ、 C_i から最も遠い個体を r とし、Step 2に戻る。
 - 5: クラスタに属していない個体をランダムにひとつ選択し、最も近いクラスタに所属させる。これをクラスタに属していない全ての個体に対して行う。
-

この手法の特徴としては、最初に出来るクラスタは理想的なクラスタとなるが、最後に分割されたクラスタほどまとまりが悪く、非類似度の大きい個体同士が同じクラスタに分割されることがある。

3.1.3 One-pass K -means Algorithm for k-anonymization (OKA)

OKA のアルゴリズムは、クラスタリングフェーズと調整フェーズに分けられる [21]。まず、クラスタリングフェーズでは、One pass k -means というクラスタリング手法により、通常のクラスタリングを行った後、調整フェーズでは、出来たクラスタを、非類似度を用いてクラスタサイズが K 個以上になるように調整を行う。その際に、クラスタサイズが K に満たない各クラスタについて、クラスタサイズが K よりも大きいものから個体を移動する。OKA のアルゴリズムを、Algorithm 9, Algorithm 10 に示す。

OKA は、GKC と異なり、通常のクラスタリングを行ってからクラスタサイズの調整を行う。そのため、GKC よりもまとまりの良いクラスタを得ることが出来る。しかしながら、クラスタリングフェーズにおける初期クラス中心選択の初期値依存の問題が挙げられる。

Algorithm 9 OKA : クラスタリングフェーズ (One-pass k -means)

-
- 1: クラスタサイズ K を設定する.
 - 2: データセットの中から, 初期クラスタ中心 $\lfloor \frac{n}{K} \rfloor$ 個をランダムに選択する.
 - 3: 残りの個体を最も非類似度の小さいクラスタ中心のクラスタ C_i に所属させる.
 - 4: クラスタ中心を更新する.
-

Algorithm 10 GKC : クラスタサイズ調整フェーズ

-
- 1: Algorithm 9 で形成された各クラスタに対して, クラスタサイズが K より大きい全てのクラスタに対して, 以下を行う.
 - 2: 各クラスタで, C_i 内の個体をクラスタ中心 v_i からの距離でソートする.
 - 3: 最もクラスタ中心から遠い個体 $|C_i| - K$ 個をそのクラスタから取り除く.
 - 4: 取り除いた全ての個体の中から個体をランダムに選択し, $|C_i| < K$ のクラスタがある場合には, その中から v_i との非類似度が最も小さくなるクラスタ C_i にその個体を所属させる. そうでない場合, 単純に最も近いクラスタに所属させる.
-

3.1.4 Clustering-Based k -Anonymity (CBK)

CBK は, データセットの 2 分割を繰り返し行うことで, 個体の分割を行う手法である [22].

まず, データセット全体を 1 つのクラスタとする. 次に, クラスタサイズが $2K$ 以上のクラスタにおいて, 初期値をランダムに 2 つ選択し, それをクラスタ中心とする. 各クラスタ中心と各個体間の非類似度を計算し, 非類似度がより小さいほうのクラスタへ個体を所属させる. その後, $|C_i| < K$ となるクラスタがあれば, 2 つのクラスタがどちらも K 個以上の個体数を持つように調節する. クラスタサイズが $2K$ 以上のクラスタがなくなるまで, これを繰り返す. CBK のアルゴリズムを Algorithm 11 に示す.

Algorithm 11 CBK

-
- 1: クラスタサイズ K を設定する.
 - 2: クラスタサイズが $2K$ 以上のクラスタをひとつ選び C とする.
 - 3: クラスタ C から個体 2 つランダムに選択し, クラスタ中心 (v_1, v_2) とする.
 - 4: クラスタ C 内の全ての個体に対して, クラスタ中心 v_1, v_2 との非類似度を計算し, 小さいほうに個体を所属させる.
 - 5: クラスタ C_1, C_2 が K 個以上の個体を持つように調整を行う.
-

クラスタ数が多いほどクラスタ内距離の総和は小さくなるため, KMC では最終的に形成されるクラスタ数が多いほうが良い. しかし, CBK の場合は, クラスタを 2 分割するように個体の分類を行うアルゴリズムであり, 本手法は, 分割後のクラスタサイズが K 個未満の場合のみクラスタサイズの調整を行うため, 最終的なクラスタ数が最大になるとは限らない.

3.2 最適化に基づくサイズ均等クラスタリング

KMC やサイズ均等クラスタリングでは、アルゴリズムの中で最適化を用いていない。しかしながら、クラスタリングの代表的な手法である *k-means* などは、目的関数の最適化に基づき構成された手法である。そこで、サイズ均等クラスタリングに最適化を用いた手法として提案されたのが、最適化に基づくサイズ均等クラスタリング (Even-sized Clustering Based on Optimization : ECBO) [15] である。ECBO は、*k-means* の目的関数と制約条件に、新たにクラスタサイズを K 以上 $K+1$ 以下にするという制約を加えることにより、クラスタサイズを均等にすることを目的としたクラスタリング手法である。

クラスタリングは、(3.2)～(3.4) の最適化問題を解くことで行う。

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|x_k - v_i\|^2 \quad (3.1)$$

$$\text{s.t.} \quad \sum_{i=1}^c u_{ki} = 1 \quad (i = 1, \dots, c) \quad (3.2)$$

$$\sum_{i=1}^c u_{ki} = 1 \quad (k = 1, \dots, n) \quad (3.3)$$

$$K \leq \sum_{k=1}^n u_{ki} \leq K + 1 \quad (i = 1, \dots, c) \quad (3.4)$$

この制約条件 (3.3)～(3.4) は、帰属度 u_{ki} に関して線形であるため、線形計画法の解法であるシンプレックス法を用いて U の最適解を求めることが出来る。また、クラスタ中心 V の最適化は、*k-means* 同様、クラスタの重心を求めるこで行う。

また、クラスタサイズ K またはクラスタ数 c の決定は、指定した変数を用いて行い、 K もしくは c の先にどちらを指定し、下記のように計算を行う。

クラスタサイズ K を指定した場合

K を最初に指定する場合、クラスタ数を最大にするために K の値によっては制約条件を満たさない場合がある。そのため、最終的にクラスタサイズ K または $K+1$ のクラスタに分割できる個体数 n と K の関係は (3.5) で表される。

$$n \leq (K+1) \cdot \frac{n - (n \bmod K)}{K} \quad (3.5)$$

(3.5) が成り立つとき、クラスタ数 c は (3.6) で表される。

$$\frac{n - (n \bmod K)}{K} = \left\lfloor \frac{n}{K} \right\rfloor \quad (3.6)$$

クラスタ数 c を指定した場合

n 個の個体をクラスタサイズが K もしくは $K+1$ の c 個のクラスタに分割する際の条件は,

$$Kc \leq n < (K+1)c \quad (3.7)$$

である。また、床関数 $\lfloor a \rfloor$ は、その定義より $\lfloor a \rfloor \leq a < \lfloor a \rfloor + 1$ という性質をもつ。ここで、 $Kc \leq n < (K+1)c$ において、全ての辺を c で割ると、

$$K \leq \frac{n}{c} < K+1 \quad (3.8)$$

よって、 K は $\frac{n}{c}$ の床関数となり、この場合、クラスタサイズ K は、

$$K = \left\lfloor \frac{n}{c} \right\rfloor \quad (3.9)$$

となる。

ECBO のアルゴリズムを Algorithm 12 に示す。

Algorithm 12 ECBO

- 1: クラスタサイズ K , c を設定する.
 - 2: クラスタ中心 $V = (v_i) (i = 1, \dots, c)$ をランダムに選択する.
 - 3: シンプレックス法により帰属度 $U = (u_{ki}) (k = 1, \dots, n, i = 1, \dots, c)$ を求める.
 - 4: クラスタの重心を求め、 V を更新する.
 - 5: V もしくは U が前回と変化しなければ終了。そうでなければ Step 3 へ.
-

ECBO は、最適化に基づきデータセットの分割を行うため、KMC の手法よりもより自然な分割結果を得やすい。しかしながら、クラスタサイズを K 以上 $K+1$ 以下にするという制約のもと分類を行うため、データセットによっては不自然な分類となることが考えられる。

第4章 提案手法

本章では、提案手法であるクラスタサイズに幅を持たせた、最適化に基づくマージン付きサイズ均等クラスタリングのアルゴリズムにおいて、クラスタリングフェーズが k -means, k -medoids, k -means++, Kernel k -means, 非類似度に L_1 ノルムを用いたもの、コサイン類似度を用いたものに基づく 6 つの手法について述べる。

4.1 α -ECBO : k -means に基づくアルゴリズム

ECBO のクラスタサイズに関する制約に対し、幅 α , α' を持たせた制約条件を以下に示す。個体数を n , クラスタサイズに関するパラメータ K は (4.4) を満たすものとする。そのとき、クラスタ数 c は (4.4) で表され、 $\lfloor \frac{n}{K} \rfloor$ である。帰属度 u_{ki} は、個体 x_k がクラスタ C_i に属する場合 $u_{ki}=1$, 属さない場合 $u_{ki}=0$ と定義する。よって、目的関数と制約条件は以下の式となる。

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|x_k - v_i\|^2 \quad (4.1)$$

$$\text{s.t.} \quad \sum_{i=1}^c u_{ki} = 1 \quad (k = 1, \dots, n) \quad (4.2)$$

$$\alpha' \leq \sum_{k=1}^n u_{ki} \leq \alpha \quad (i = 1, \dots, c) \quad (4.3)$$

$$\begin{cases} \alpha' \leq K \\ \alpha \geq K + 1 \end{cases}$$

(4.3) はクラスタサイズに関する制約条件である。 u_{ki} の計算には ECBO と同様にシンプレス法を適用する。

また、クラスタサイズ K を設定した場合のクラスタ数 c は、(4.4) を用いて計算される。

$$\frac{n - (n \bmod K)}{K} = \left\lfloor \frac{n}{K} \right\rfloor \quad (4.4)$$

最適化に基づくマージン付きサイズ均等クラスタリングのアルゴリズムを Algorithm 13 に示す。

Algorithm 13 α -ECBO

-
- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
 - 2: クラスタサイズに関する制約条件 (4.3) の幅 α, α' を設定する.
 - 3: クラスタ中心 $V = (v_i) (i = 1, \dots, c)$ をランダムに選択する.
 - 4: シンプレックス法により帰属度 $U = (u_{ki}) (k = 1, \dots, n, i = 1, \dots, c)$ を更新する.
 - 5: クラスタ中心 V を (4.5) で更新する.

$$v_i = \frac{\sum_{k=1}^n u_{ki} x_k}{\sum_{k=1}^n u_{ki}} \quad (4.5)$$

-
- 6: 前回と V が変化しなければ終了. そうでなければ, Step 4 へ.
-

4.2 α -ECBO++ : k -means++に基づくアルゴリズム

Algorithm 13 では, 初期クラスタ中心をランダムに与えてクラスタリングを行っている. しかしながら, 従来の k -means と同様に, 与えられる初期値によっては, クラスタリング結果に悪い影響を及ぼす場合がある. 一方, k -means の初期値依存性を軽減するために, k -means++ [16] という手法が提案されている. そこで, 本提案アルゴリズムへの k -means++の適用を考える.

ここで, 目的関数, 制約条件は, 先ほどの Algorithm 13 と同様のものを用いる. まず, k -means++と同様, Algorithm 14 により, クラスタ中心の初期値を計算する. その後, 得られた初期値を用いて, Algorithm 13 を実行する. k -means++をクラスタリングフェーズに用いた最適化に基づくマージン付きサイズ均等クラスタリングのアルゴリズムを Algorithm 15 に示す.

Algorithm 14 k -means++ (初期値 V 計算フェーズ)

-
- 1: ランダムに個体を 1 つ選び, クラスタ中心 v_1 とする.
 - 2: (4.6) の確率で個体を 1 つ選び, 新しいクラスタ中心 v_i とする.

$$\frac{D(x)^2}{\sum_{x \in X} D(x)^2} \quad (4.6)$$

- ここで, $D(x)$ は個体 x とすでに選択した最も近いクラスタ中心との距離である.
-
- 3: c 個のクラスタ中心を選ぶまで Step 2 を繰り返す.

Algorithm 15 α -ECBO++

- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
 - 2: クラスタサイズに関する制約条件 (4.3) の幅 α, α' を設定する.
 - 3: 初期クラスタ中心 $V = (v_i) (i = 1, \dots, c)$ を Algorithm 14 により決定する.
 - 4: シンプレックス法により, 帰属度 $U = (u_{ki}) (k = 1, \dots, n, i = 1, \dots, c)$ を更新する.
 - 5: クラスタ中心 V を更新する.
 - 6: 前回と V が変化しなければ終了. そうでなければ, Step 4 へ.
-

4.3 α -MECBO : k -medoids に基づくアルゴリズム

k -means は, クラスタ中心を各クラスタの重心とする手法であるが, k -medoids [17] は, medoid と呼ばれるクラスタ内を代表する個体そのものをクラスタ中心とする手法である. そのため, k -means よりも外れ値の影響を受けにくいとされている. また, 個体そのものがクラスタ中心となるため, 個体同士の距離さえ与えればクラスタリングを行うことができる. k -medoids を用いたアルゴリズムを Algorithm 16 に示す.

4.4 α -KECBO : Kernel k -means に基づくアルゴリズム

通常, k -means などのクラスタリング手法では, データセットの分類境界はボロノイズのような線形に分類される形となる. しかしながら, 実データや 2 重円データなど, データによっては線形分離できないものも多い. そこで, 元の空間では非線形に分類できないデータセットを高次元特徴空間に写像してからクラスタリングを行うことで, 非線形なデータの分割を得る手法であるカーネルクラスタリングが提案された.

そこで, 本提案手法のクラスタリングフェーズにも, カーネル法を取り入れることを考える. 目的関数は通常の Kernel k -means と同様のものを用い, 制約条件にクラスタサイズに関するものを追加する. クラスタリングフェーズに Kernel k -means を用いた提案手法のアルゴリズムを Algorithm 17 に示す.

4.5 α - L_1 ECBO : L_1 ノルムを用いたアルゴリズム

k -means 等のクラスタリング手法では, 多くの場合, ユークリッド距離の 2 乗を用いている. その場合, 個体を直線的に結んだときの距離の 2 乗となっているため, 外れ値の影響を受けやすいという問題が挙げられる. しかしながら, L_1 ノルムを距離尺度として用いると, 各次元での個体間の差の絶対値の総和によって定義されるため, 非類似度に L_1 ノルムを用いることで, 外れ値の影響を受けづらいという利点が挙げられる. そこで, 本提案手法にも L_1 ノルムを非類似度に用いた k -means クラスタリングを導入することを考える.

非類似度に L_1 ノルムを用いたときの、目的関数は(4.11)のように表され、制約条件は他のアルゴリズムのものと同様となる。

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|x_k - v_i\| \quad (4.10)$$

$$= \sum_{k=1}^n \sum_{i=1}^c \sum_{j=1}^p |x_{kj} - v_{ij}| \quad (4.11)$$

ここで、帰属度 U を固定したときの V の最適化では、各クラスタの各成分ごとに独立で最適化を行う。よって、準目的関数は(4.12)のように定義される。

$$J_{ij} = \sum_{k=1}^n u_{ki} |x_{kj} - v_{ij}| \quad (4.12)$$

(4.12) を用いると、目的関数(4.11)は、(4.13)のように書き換えられる。

$$J = \sum_{i=1}^c \sum_{j=1}^p J_{ij} \quad (4.13)$$

よって、準目的関数 J_{ij} が最適となるとき、目的関数 J も最適となる。

非類似度が L_1 ノルムに基づく k -means クラスタリングを導入した、最適化に基づくマージン付きサイズ均等クラスタリングのアルゴリズムを Algorithm 18, Algorithm 19 に示す。

4.6 α -cosine-ECBO : コサイン類似度を用いたアルゴリズム

コサイン類似度は、 n 次元空間でのベクトルの類似性を表し、ベクトルの向きの近さを類似性の尺度としている。本提案手法にも、距離尺度にコサイン類似度を用いた k -means クラスタリングを導入することを考える。

類似性尺度にコサイン相関を用いたときの、最適化に基づくマージン付きサイズ均等クラスタリング手法の目的関数は、(4.14) で表される。

$$\text{minimize} \quad J = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \left(\frac{3}{2} - \cos \theta_{x_k, v_i} \right) \quad (4.14)$$

ここで、帰属度とクラスタサイズに関する制約条件はユークリッド距離の 2 乗を非類似度に用いた場合と同様となる。提案手法におけるクラスタリングフェーズで、コサイン相関を用いた k -means を導入した場合の最適化に基づくマージン付きサイズ均等クラスタリングのアルゴリズムを Algorithm 20 に示す。

Algorithm 16 α -MECBO

- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
 - 2: クラスタサイズに関する制約条件 (4.3) の幅 α, α' を設定する.
 - 3: 初期 medoids の選択
 1. 全ての個体同士の距離 $\|x_i - x_j\|$ を計算する.
 2. 全ての個体 x_j に関して, w_j を (4.7) を用いて計算する.
$$w_j = \sum_{i=1}^n \frac{\|x_i - x_j\|}{\sum_{k=1}^n \|x_k - x_j\|} \quad (4.7)$$
 3. w_j を昇順にソートし, w_j の値が最も小さい k 個の個体を初期の medoids として選択する.
 4. 各個体を最も近い medoid のクラスタに所属させる.
 5. すべての個体とそれが所属するクラスタの medoid との距離の総和を求める.
 - 4: medoids の更新
各クラスタにおいて, 任意の個体とその他のクラスタに属する全ての個体との距離の総和が最も小さい個体を選び, 新たな medoid とする.
 - 5: 個体の帰属の更新
シンプレックス法を用いて, 以下の要領で各個体の帰属の更新を行う.
 1. 各個体に最も近い medoid のクラスタに所属させる.
 2. 全ての個体とそれが所属するクラスタの medoid との距離の総和を求める.
 - 6: 収束判定
各個体の帰属が変化しなくなれば終了. そうでなければ, Step 4 へ.
-

Algorithm 17 α -KECBO

- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
- 2: クラスタサイズに関する制約条件 (4.3) の幅 α, α' を設定する.
- 3: 初期帰属度 U をランダムに設定する.
- 4: U の値を用いて, (4.8) を更新する.

$$\|\Phi(x_k) - v_i^\Phi\|^2 = \kappa(x_k, x_k) - \frac{2}{|C_i|} \sum_{x_l \in C_i} \kappa(x_k, x_l) + \frac{1}{|C_i|^2} \sum_{x_l \in C_i} \sum_{x_m \in C_i} \kappa(x_l, x_m) \quad (4.8)$$

- 5: 更新した非類似度を用いて, シンプレックス法を用いて (4.9) を解き, U を更新する.

$$\text{minimize } J(U, V) = \sum_{k=1}^n \sum_{i=1}^c u_{ki} \|\Phi(x_k) - v_i^\Phi\|^2 \quad (4.9)$$

- 6: 前回のステップと U が変化しなくなれば終了. そうでなければ, Step 4 へ.
-

Algorithm 18 $\alpha - L_1$ ECBO

- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
 - 2: クラスタサイズに関する制約条件 (4.3) の幅 α, α' を設定する.
 - 3: 初期クラスタ中心 $V = (v_i) (i = 1, \dots, c)$ をデータセットの中からランダムに選択する.
 - 4: (4.13) を最小化するように, シンプレックス法により帰属度 $U = (u_{ki}) (k = 1, \dots, n, i = 1, \dots, c)$ を更新する.
 - 5: クラスタ中心 V を, Algorithm 19 を用いて更新する.
 - 6: 前回と V が変化しなければ終了. そうでなければ, Step 4 へ.
-

Algorithm 19 $\alpha - L_1$ ECBO (クラスタ中心最適化フェーズ)

- 1: 個体 x_{kj} を各次元 j ごとに昇順にソートし, $x_{q(k)j}$ とする.
 - 2: $S = -\frac{1}{2} \sum_{k=1}^n u_{q(k)j}, r = 0$ とする.
 - 3: S の符号が $-$ から $+$ に変わるまで, $S = S + u_{q(k)j}, r = r + 1$ を計算する.
 - 4: 最適解 $v_{ij} = x_{q(r)j}$ を更新する.
-

Algorithm 20 α -cosine-ECBO

- 1: クラスタサイズ K を設定し, (4.4) に従ってクラスタ数 c を計算する.
- 2: クラスタサイズに関する制約条件 (4.3) の幅 α , α' を設定する.
- 3: クラスタ中心 $V = (v_i)$ ($i = 1, \dots, c$) をランダムに選択する.
- 4: (4.14) を最小化するように, シンプレックス法により帰属度 $U = (u_{ki})$ ($k = 1, \dots, n$, $i = 1, \dots, c$) を更新する.
- 5: クラスタ中心 V を (4.15) で更新する.

$$v_i = \frac{\sum_{k=1}^n u_{ki} x_k}{\sum_{k=1}^n u_{ki}} \quad (4.15)$$

- 6: 前回と V が変化しなければ終了. そうでなければ, Step 4 へ.
-

第5章 数値例

5.1 Adjusted Rand Index

クラスタリング結果の標準的な性能評価基準として知られる Adjusted Rand Index (ARI) [23] の値を比較して評価する。ARI は、分類の正解率を表し、表 5.1 の値を用いて (5.1) で表現される。

表 5.1: Comparing partitions U and V

		partition V	
		same group	different group
partition U	same group	a	b
	different group	c	d

$$\text{ARI} = \frac{nC_2(a+d) - [(a+b)(a+c) + (c+d)(b+d)]}{nC_2^2 - [(a+b)(a+c) + (c+d)(b+d)]} \quad (5.1)$$

ARI は、正解クラスタの分割と実際のクラスタ分割の近さを表している。 n はデータセットの個体数を表す。実際に得た分類結果からあるペアを取り出し、それらが同じクラスタに属しているかつ正解の分割でも同じクラスタに属す場合、 a に 1 を加える。また、取り出したペアが互いに異なるクラスタに属しており、正解の分割でも異なるクラスタに属す場合、 d に 1 を加える。

仮に、2つのクラスタリング結果が同じである場合は、 $\text{ARI} = 1$ となる。その逆に、ARI の値が 0 に近いほどクラスタ分割は一致していないということを示す。

5.2 人工データ

4つのデータセットを用いて、 k -means, ECBO と提案手法でクラスタリングを行った結果を、目的関数・目的関数の分散・ARI の図とグラフで示す。なお、Kernel k -means をクラスタリングフェーズを用いた手法では、カーネル関数にガウシアンカーネルを用いており、データの分散を調整するパラメータである γ の値は、 $\gamma = 1.0$ として、実験を行った。Centroid は、各クラスタのクラスタ中心を表し、Medoid も同様、 k -medoids に基づく手法の結果の場合におけるクラスタ中心を表す。

5.2.1 密度の異なる円状のデータ

個体 155 個の一様乱数を生成した半径の小さな円と, 個体 100 個の一様乱数を生成した半径 3 倍の大きな円の, 合計 255 個の個体を持つデータセットに対して, 各手法でクラスタリングを行う. このデータセットを図 5.1 に示す. このデータに対して, 各手法について $c = 2$ で初期値を 10 回変えて, クラスタリングを行った.

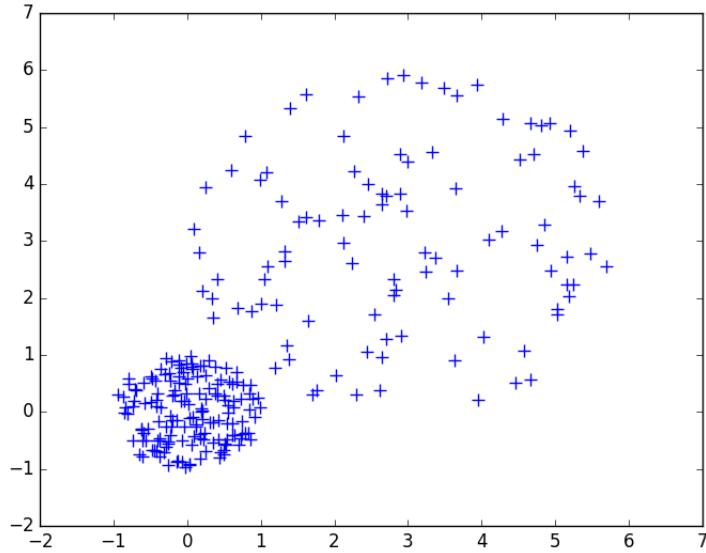


図 5.1: 密度の異なる円状のデータ（個体数 255 個）

以下に, 各提案手法および既存手法を用いたときのクラスタリング結果を図で示すとともに, 各手法において, ARI の値が最大となったときの目的関数, 目的関数の分散, ARI 値, 分割結果を表 5.2 およびグラフに示す.

k -means によるクラスタリングでは, クラスタの境界が, 半径の大きな円の方に引き寄せられ, 分割される領域がクラスタ間で均等になるように分類されており, 不自然な分類結果となっている. また, ECBO を用いた結果も, 個体数を均等に分類するという制約のため, クラスタ中心が個体数の多い方に引き寄せられ, $ARI = 0.62$ となっている. しかしながら, クラスタサイズにマージンを持たせた提案手法 (α -ECBO, α -ECBO++, α -MECBO, α -KECBO) を用いることで, データの密度や個体数が異なるような場合でも, 半径の小さな円と大きな円にクラスタリングされており, $ARI = 1.0$ となった. また, ECBO と提案手法 (α -ECBO) の目的関数を比較すると, 提案手法の方がより小さい値を得ることが出来ている. その理由としては, マージンを持たせたことでクラスタサイズに関する制約条件が緩和され, よりまとまりのあるクラスタを得ることが出来たためだと考えられる. また, $ARI = 1.0$ となったときの, 提案手法の基づくクラスタリングフェーズごとに結果を比較すると, α -KECBO のみク

ラスタサイズのマージンが $\alpha = 38$ となるときに最適な分割結果となっており、サイズに関する制約を他のクラスタリングフェーズに基づく提案手法より、更に緩めたときに自然な結果となった。目的関数の分散は、 α -ECBO++で、ほぼ 0 に等しい値を取っている。これは、初期値選択が改善されたことで、各試行において平均的に安定した結果を得られたためである。

その一方で、 α -L₁ECBO, α -cosine-ECBO では、ARI の値が 1.0 となるような最適な分類結果を得ることが出来ておらず、 α -cosine-ECBO に至っては、 k -means や ECBO よりも悪い結果となってしまった。 α -cosine-ECBO がこのような結果となったのは、ひとつのクラスタ中心が原点 (0, 0) となるように選択されており、そこからの角度に基づいて分割されたからだと考えられる。

表 5.2: 密度の異なる円状のデータに対する各手法の実行結果

手法	分割結果	Margin(α)	目的関数 Min(J)	目的関数の分散 Var(J)	Max(ARI)
k -means	(79,176)	—	4.87E ⁺²	1.52	0.69
ECBO	(127,128)	—	7.86E ⁺²	7.71	0.62
α -ECBO	(100,155)	27	5.57E ⁺²	5.25E ⁺⁴	1.00
α -ECBO++	(100,155)	27	5.57E ⁺²	1.29E ⁻²⁶	1.00
α -MECBO	(100,155)	27	5.75E ⁺²	1.45E ⁺²	1.00
α -KECBO	(100,155)	38	1.72E ⁺²	2.11	1.00
α -L ₁ ECBO	(101,154)	26	3.58E ⁺²	1.12E ⁺²	0.95
α -cosine-ECBO	(117,138)	10	98.1	26.2	0.45

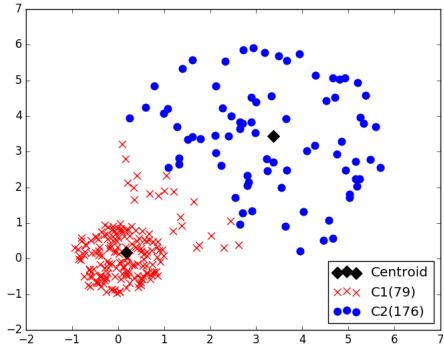


図 5.2: k -means による分類結果

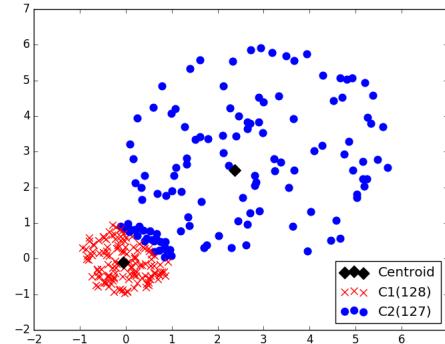


図 5.3: ECBO による分類結果

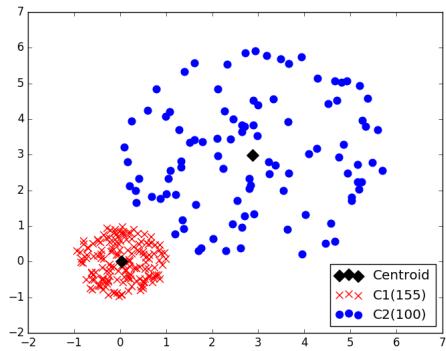


図 5.4: α -ECBO による分類結果 ($\alpha=27$)

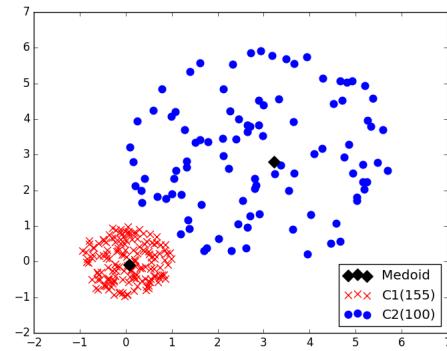


図 5.5: α -MECBO による分類結果 ($\alpha=27$)

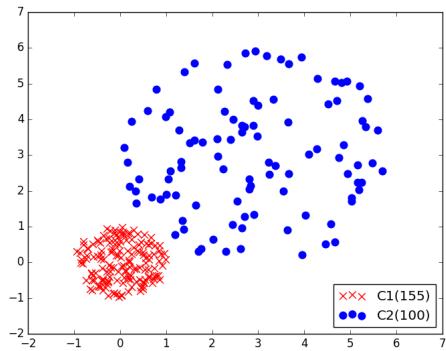


図 5.6: α -KECBO による分類結果 ($\alpha=38$)

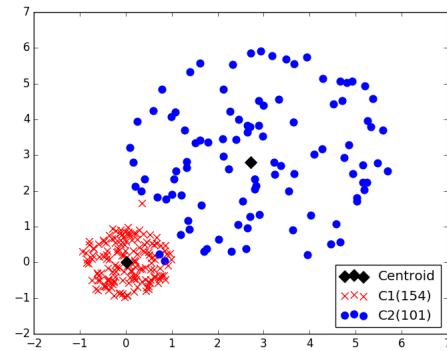


図 5.7: α - L_1 ECBO による分類結果 ($\alpha=26$)

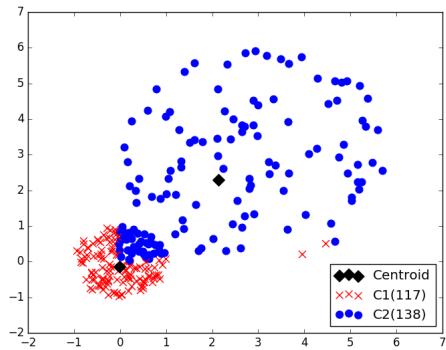


図 5.8: α -cosine-ECBO による分類結果 ($\alpha=10$)

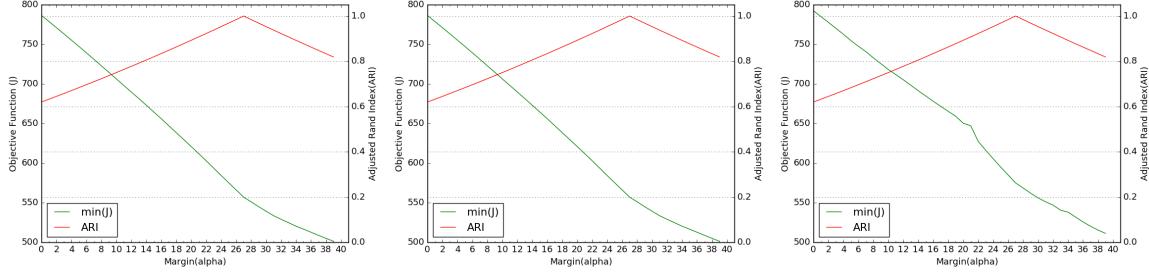


図 5.9: α -ECBO を用いたとき 図 5.10: α -ECBO++を用いたと 図 5.11: α -MECBO を用いたと
の Margin の変化に伴う目的関数の Margin の変化に伴う目的 関数と ARI の変遷
の Margin の変化に伴う目的関数と ARI の変遷



図 5.12: α -KECBO を用いたと 図 5.13: α - L_1 ECBO を用いたと 図 5.14: α -cosine-ECBO を用いた
きの Margin の変化に伴う目的関数の Margin の変化に伴う目的 関数と ARI の変遷
きの Margin の変化に伴う目的関数と ARI の変遷

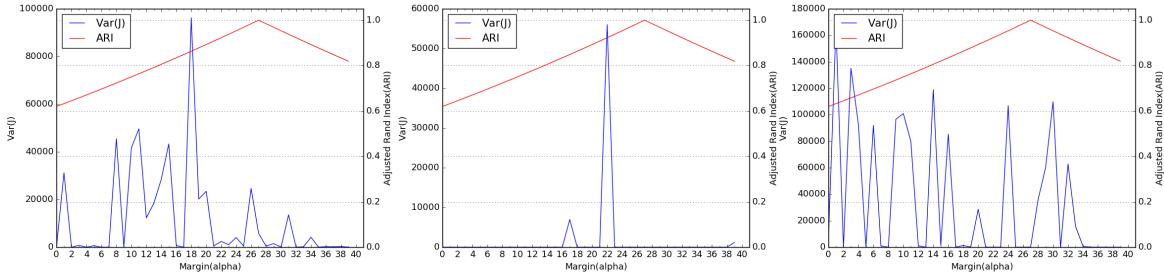


図 5.15: α -ECBO を用いたとき 図 5.16: α -ECBO++を用いたと 図 5.17: α -MECBO を用いたと
の Margin の変化に伴う目的関数の Margin の変化に伴う目的 関数の分散と ARI の変遷
の Margin の変化に伴う目的関数の分散と ARI の変遷

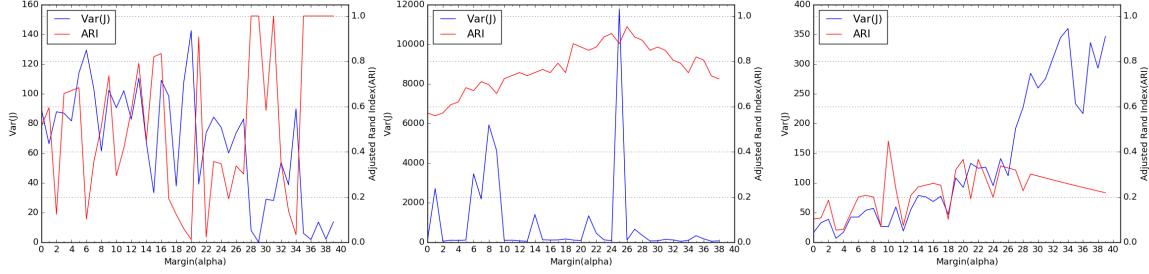


図 5.18: α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.19: α -L₁ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.20: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

5.2.2 2重円データ

半径が小さな円の内側に個体 100 個、半径が 2 倍の大きな円の円上に個体 150 個の、合計 250 個の個体を持つデータセットに対して、各手法でクラスタリングを行う。このデータセットを図 5.21 に示す。このデータに対して、各手法について $c = 2$ で初期値を 10 回変えて、クラスタリングを行った。

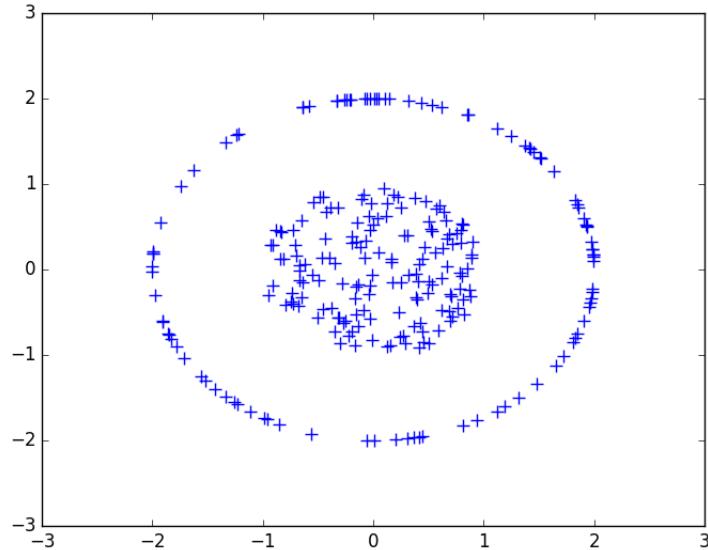


図 5.21: 2 重円データ（個体数 250 個）

以下に、各提案手法および既存手法を用いたときのクラスタリング結果を図で示すとともに

に、各手法において、ARI の値が最大となったときの目的関数、目的関数の分散、ARI 値、分割結果を表 5.3 およびグラフに示す。

線形分離不可能なデータでは、Kernel k -means および α -KECBO を除いて、不自然な分類結果となっており、分割されたときの各クラスタが形成する面積がクラスタ間で均等になっている。KECBO で、内側の円と外側の円を上手く分離できない理由としては、円の内側と外側の個体数の差が 50 個あるにも関わらず、クラスタサイズを均等にするという制約条件によるためである。 α -KECBO では、マージンを持たせることで、円の内側と外側にきれいに分類することが出来た。しかし、目的関数や ARI の値は、マージンによって変化が大きく、クラスタサイズに関する制約に結果が強く左右されていることが、Margin の変化に伴う目的関数と ARI の変遷のグラフから分かる。

表 5.3: 2 重円データに対する各手法の実行結果

手法	分割結果	Margin(α)	目的関数 Min(J)	Max(ARI)
Kernel k -means	(100,150)	—	1.66E ⁺²	1.00
KECBO	(125,125)	—	3.27E ⁺²	0.49
α -ECBO	(114,136)	15	3.13E ⁺²	0.04
α -ECBO++	(114,136)	11	3.15E ⁺²	0.04
α -MECBO	(117,133)	18/28	3.18E ⁺²	0.03
α -KECBO	(100,150)	39	1.66E ⁺²	1.00
α - L_1 ECBO	(106,144)	19	1.72E ⁺²	0.11
α -cosine-ECBO	(124,126)	1	87.9	0.04E ⁻¹

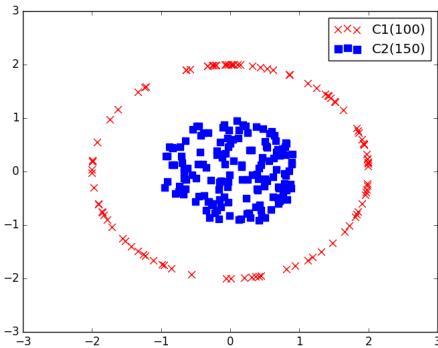


図 5.22: Kernel k -means による分類結果

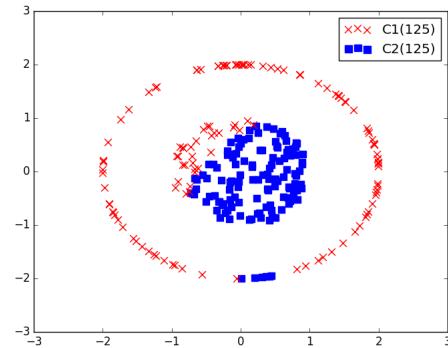


図 5.23: KECBO による分類結果

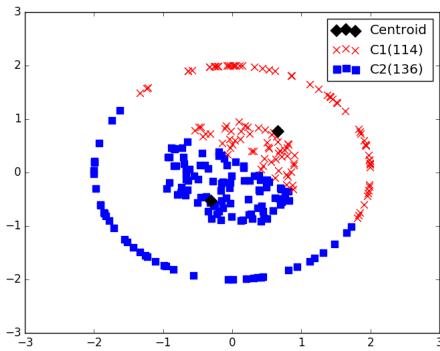


図 5.24: α -ECBO による分類結果 ($\alpha=15$)

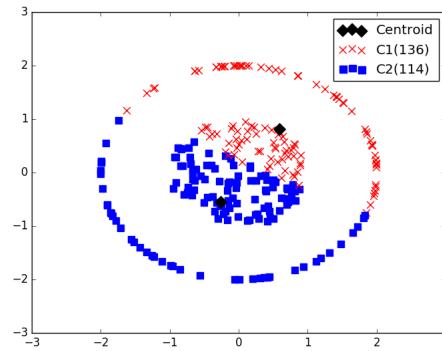


図 5.25: α -ECBO++による分類結果 ($\alpha=11$)

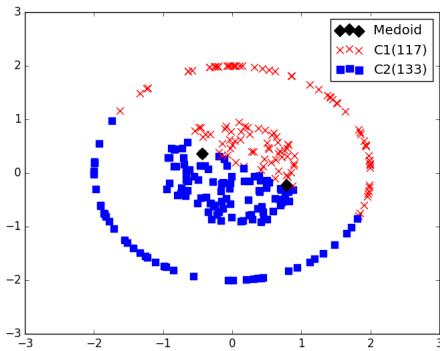


図 5.26: α -MECBO による分類結果 ($\alpha=18,28$)

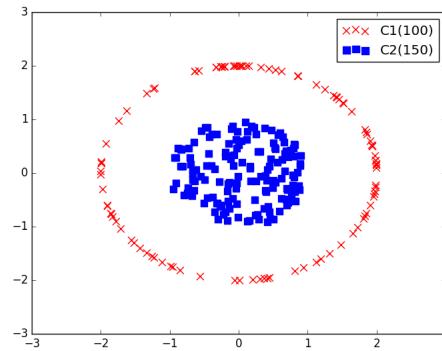


図 5.27: α -KECBO による分類結果 ($\alpha=39$)

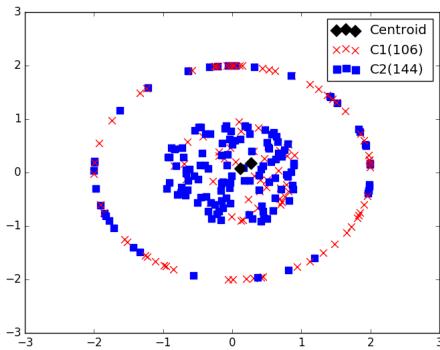


図 5.28: α - L_1 ECBO による分類結果 ($\alpha=19$)

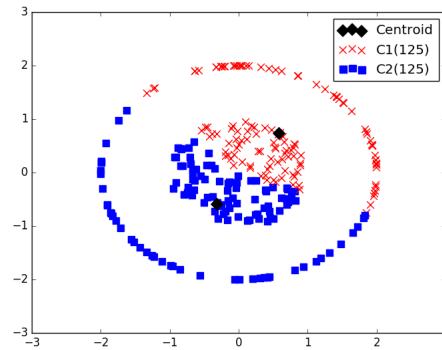


図 5.29: α -cosine-ECBO による分類結果 ($\alpha=1$)

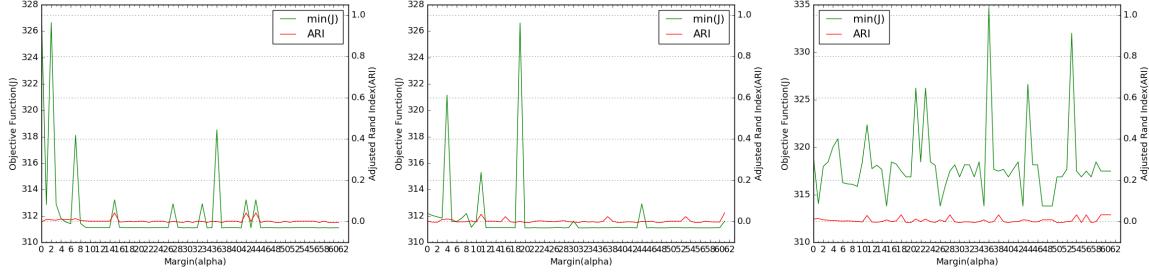


図 5.30: α -ECBO を用いたとき 図 5.31: α -ECBO++を用いたと 図 5.32: α -MECBO を用いたと
の Margin の変化に伴う目的関数の Margin の変化に伴う目的関数と ARI の変遷 関数と ARI の変遷

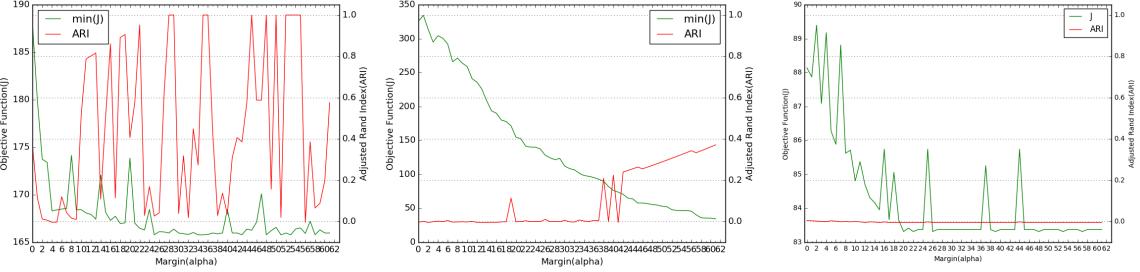


図 5.33: α -KECBO を用いたと 図 5.34: α -L₁ECBO を用いたと 図 5.35: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

5.2.3 個体数が異なる標準正規分布に従ったデータ

標準正規分布に従った乱数を、個体数 80 個, 90 個, 100 個, 110 個の合計 380 個の個体を持つデータセットに対して、各手法でクラスタリングを行う。このデータセットを図 5.36 に示す。このデータセットに対して、 $c = 4$ で初期値を 10 回変えてクラスタリングを行った。

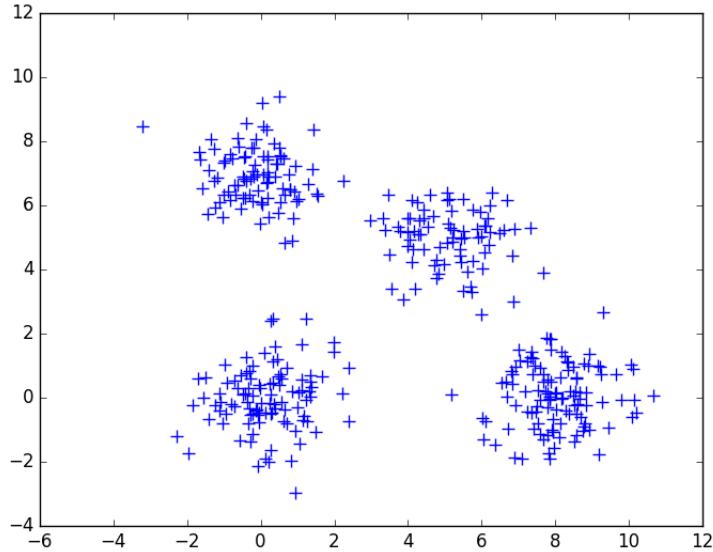


図 5.36: 個体数が異なる標準正規分布に従ったデータ（個体数 380 個）

以下に、各提案手法および既存手法を用いたときのクラスタリング結果を図で示すとともに、各手法において、ARI の値が最大となったときの目的関数、目的関数の分散、ARI 値、分割結果を表 5.4 およびグラフに示す。

k -means でも ARI= 1.0 となる最適な分割結果を得ることが出来たが、ECBO の場合、クラスタサイズに関する制約条件のため、個体が各クラスタで均等になるように分割され、目的関数の分散も非常に大きな値を取っており、各試行において初期値に強く依存していることが分かる。提案手法である α -ECBO, α -ECBO++, α -MECBO では、マージン α を 15 以上にすると、ARI= 1.0 となり、必ず最適な分類結果となっている。また、目的関数の分散もおよそ 0 で、この 3 手法では、初期値に依らず大域的最適解に収束していくことが分かる。 α -cosine-ECBO で上手く分類できなかった理由としては、密度の異なる 1 つ目のデータセット同様、ひとつのクラスタ中心は原点が (0, 0) となるように選択されているため、このような分類結果になってしまったと思われる。

表 5.4: 個体数が異なる標準正規分布に従ったデータに対する各手法の実行結果

手法	分割結果	Margin(α)	目的関数 Min(J)	目的関数の分散 Var(J)	Max(ARI)
k -means	(80,90,100,110)	—	6.77E ⁺²	6.04E ⁺⁵	1.00
ECBO	(95,95,95,95)	—	9.44E ⁺²	6.08E ⁺⁵	0.86
α -ECBO	(80,90,100,110)	15~	6.76E ⁺²	1.29E ⁻²⁶	1.00
α -ECBO++	(80,90,100,110)	15~	6.76E ⁺²	1.29E ⁻²⁶	1.00
α -MECBO	(80,90,100,110)	15~	6.82E ⁺²	0.00	1.00
α -KECBO	(80,88,100,112)	16	2.93E ⁺²	24.9	0.91
$\alpha - L_1$ ECBO	(82,90,102,106)	16	5.18E ⁺²	2.43E ⁺²	0.93
α -cosine-ECBO	(73,90,99,118)	22	45.6	5.97	0.50

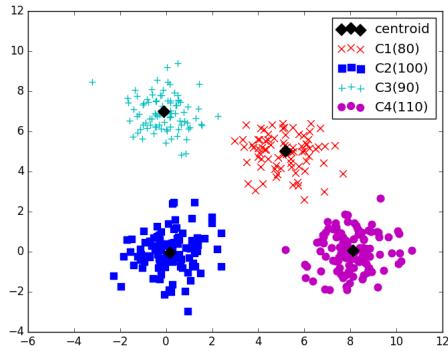


図 5.37: k -means による分類結果

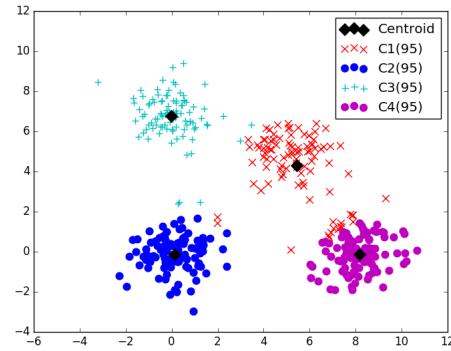


図 5.38: ECBO による分類結果

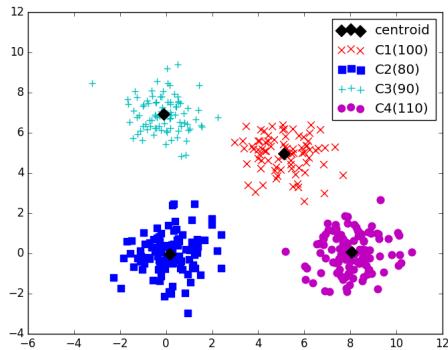


図 5.39: α -ECBO による分類結果 ($\alpha=15\sim$)

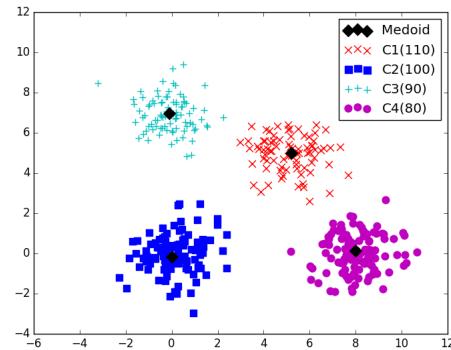


図 5.40: α -MECBO による分類結果 ($\alpha=15\sim$)

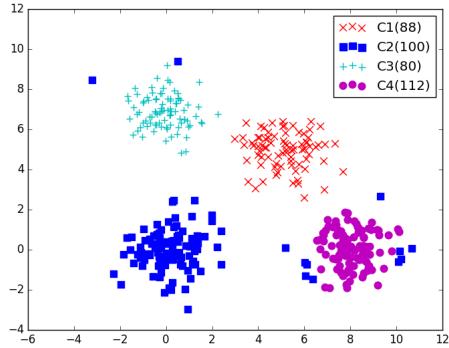


図 5.41: α -KECBO による分類結果 ($\alpha=16$)

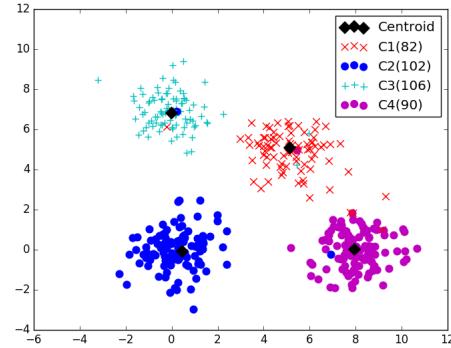


図 5.42: α -L₁ECBO による分類結果 ($\alpha=16$)

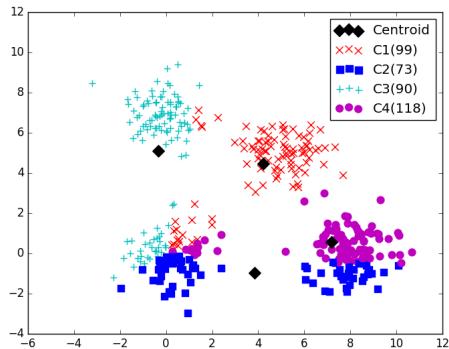


図 5.43: α -cosine-ECBO による分類結果 ($\alpha=22$)

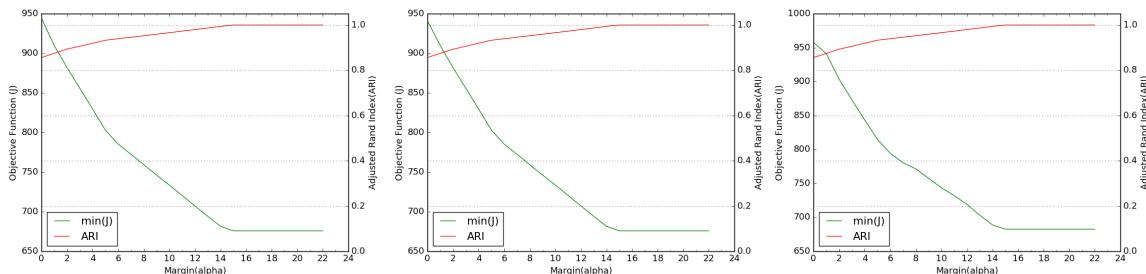


図 5.44: α -ECBO を用いたとき 図 5.45: α -ECBO++を用いたと 図 5.46: α -KECBO を用いたと
の Margin の変化に伴う目的関数の Margin の変化に伴う目的関数の変遷
数と ARI の変遷 関数と ARI の変遷

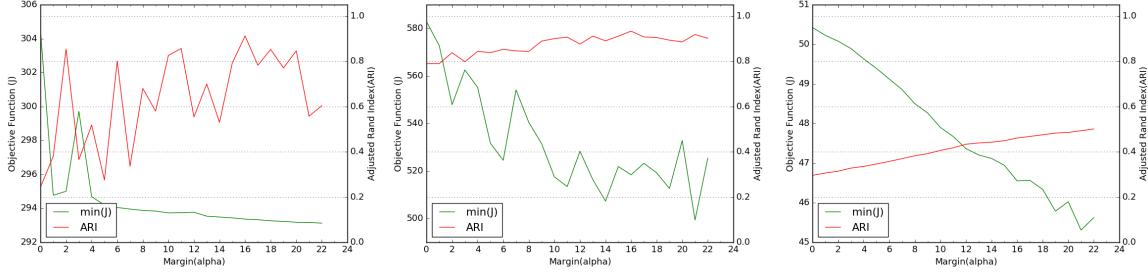


図 5.47: α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷
 図 5.48: α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷
 図 5.49: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

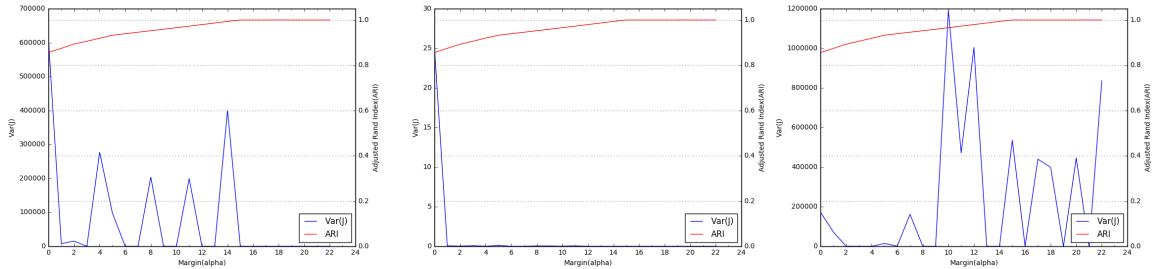


図 5.50: α -ECBO を用いたとき 図 5.51: α -ECBO++を用いたと 図 5.52: α -MECBO を用いたとの Margin の変化に伴う目的関数の分散と ARI の変遷
 の Margin の変化に伴う目的関数の分散と ARI の変遷
 の Margin の変化に伴う目的関数の分散と ARI の変遷

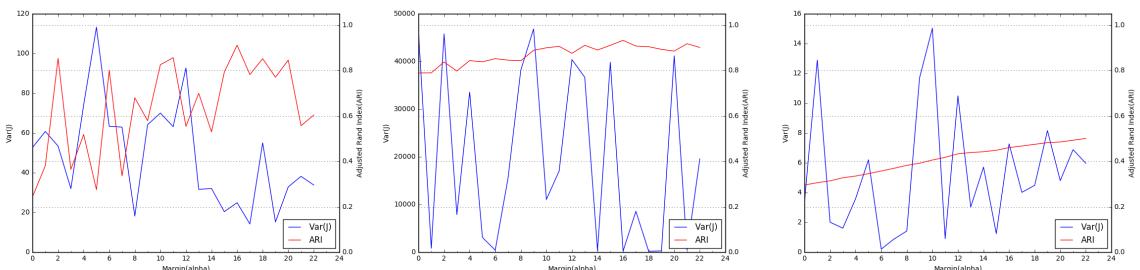


図 5.53: α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.54: α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.55: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

5.2.4 密度・半径の異なる円状のデータ

個体 180 個の一様乱数を生成した半径の小さな円 2 つと、個体 150 個の一様乱数を生成した半径 3 倍の大きな円の、密度・半径の異なる個体数合計 510 個のデータセットに対して、各手法でクラスタリングを行う。このデータセットを図 5.56 に示す。このデータに対して、 $c = 3$ で初期値を 10 回変えてクラスタリングを行った。

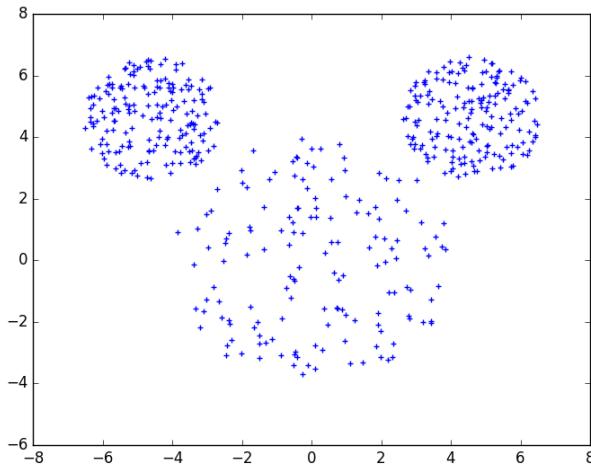


図 5.56: 密度・半径の異なる円状のデータ（個体数 510 個）

以下に、各提案手法および既存手法を用いたときのクラスタリング結果を図で示すとともに、各手法において、ARI の値が最大となったときの目的関数、目的関数の分散、ARI 値、分割結果を表 5.5 およびグラフに示す。

k -means, ECBO の両手法では上手く分類できないデータセットであるが、マージンがデータセットの各クラスタが形成する個体数の差異に対してちょうど一致する $\alpha = 20$ となったとき、最適な分類結果を得ることが出来ていることが、 $ARI = 1.0$ となる結果から明らかである。マージンが大きくなるにつれて、クラスタサイズに関する制約が弱まるため、 k -means を用いたときのクラスタリング結果に近づいていくことが分かる。目的関数の分散は、 k -means++に基づく手法で、マージンに関係なく 0 に近い結果となっており、確率的に初期値を選択するクラスタリングの特徴が上手く効いており、各試行で安定した結果を得られたことが分かる。

表 5.5: 密度・半径の異なる円状のデータに対する各手法の実行結果

手法	分割結果	Margin(α)	目的関数 Min(J)	目的関数の分散 Var(J)	Max(ARI)
k -means	(132,188,190)	—	1.93E ⁺³	5.17E ⁺⁴	0.90
ECBO	(170,170,170)	—	2.33E ⁺³	1.16E ⁺⁶	0.89
α -ECBO	(150,180,180)	20	1.97E ⁺³	1.57E ⁺⁶	1.00
α -ECBO++	(150,180,180)	20	1.97E ⁺³	5.17E ⁻²⁶	1.00
α -MECBO	(150,180,180)	20	2.01E ⁺³	1.25E ⁺⁵	1.0
α -KECBO	(145,180,185)	25	4.37E ⁺²	99.3	0.60
α - L_1 ECBO	(153,177,180)	17	1.08E ⁺³	4.44E ⁺²	0.96
α -cosine-ECBO	(142,169,199)	28	89.1	35.2	0.34

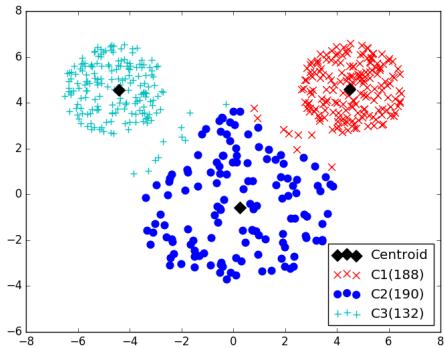


図 5.57: k -means による分類結果

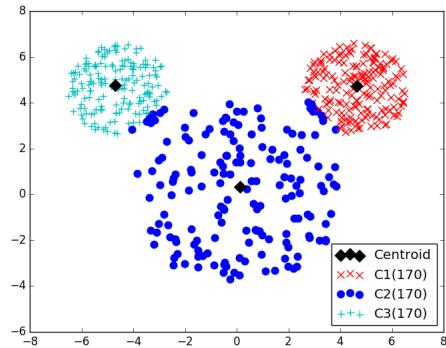


図 5.58: ECBO による分類結果

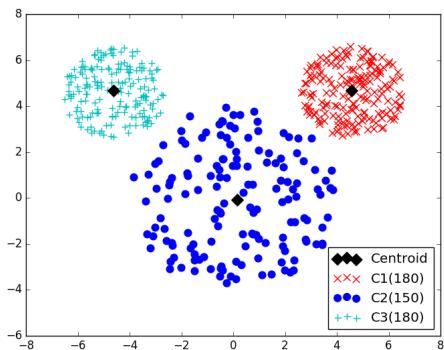


図 5.59: α -ECBO による分類結果 ($\alpha=20$)

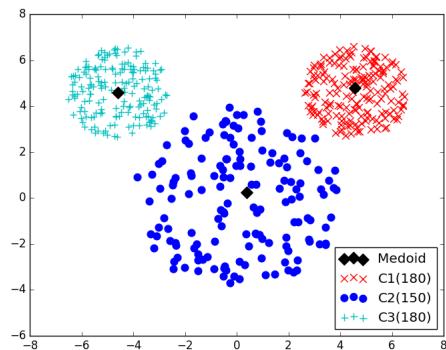


図 5.60: α -MECBO による分類結果 ($\alpha=20$)

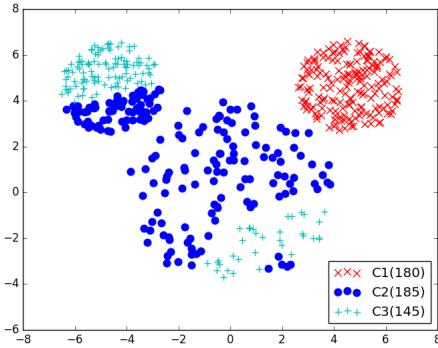


図 5.61: α -KECBO による分類結果 ($\alpha=25$)

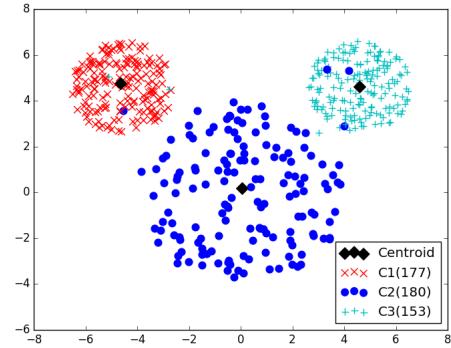


図 5.62: α - L_1 ECBO による分類結果 ($\alpha=17$)

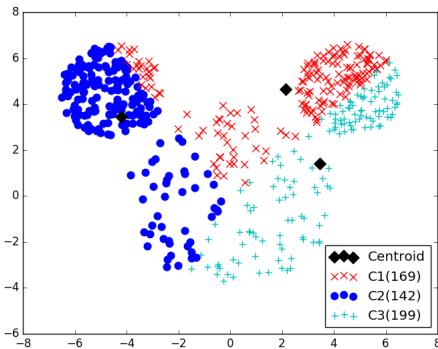


図 5.63: α -cosine-ECBO による分類結果 ($\alpha=28$)

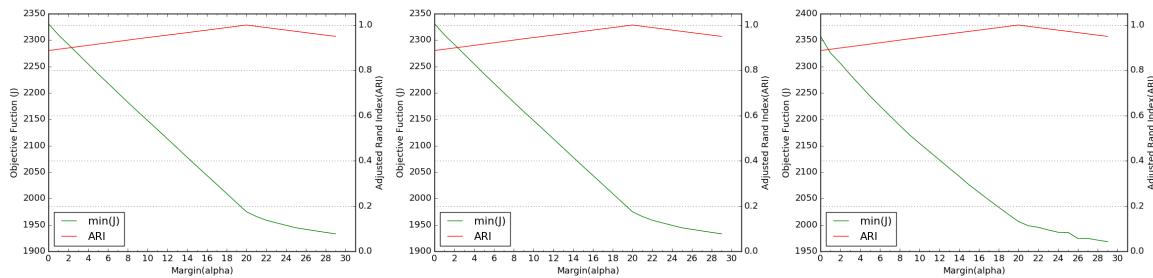


図 5.64: α -ECBO を用いたとき 図 5.65: α -ECBO++を用いたと 図 5.66: α -MECBO を用いたと
の Margin の変化に伴う目的関数の Margin の変化に伴う目的関数の Margin の変化に伴う目的
数と ARI の変遷 関数と ARI の変遷 関数と ARI の変遷

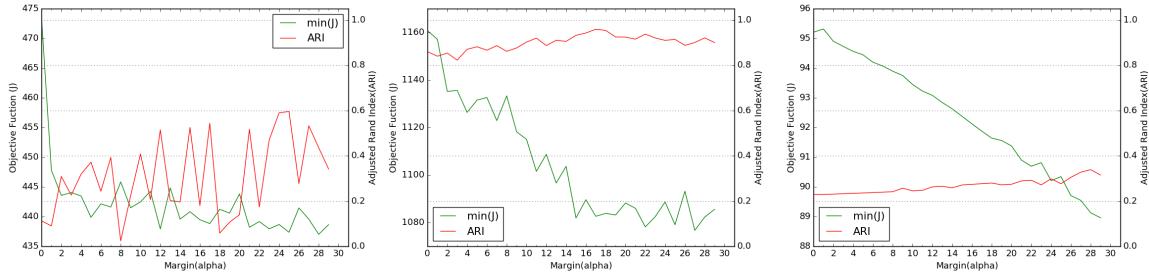


図 5.67: α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.68: α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.69: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

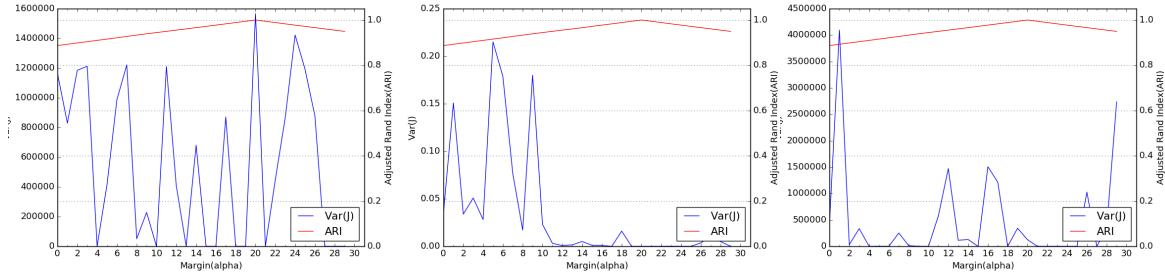


図 5.70: α -ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.71: α -ECBO++ を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.72: α -MECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

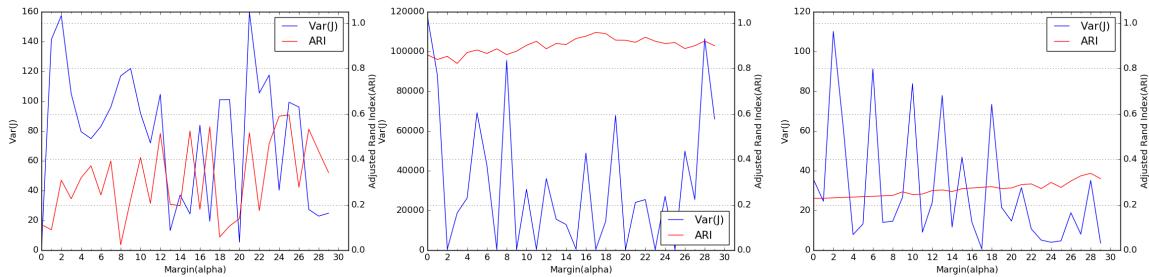


図 5.73: α -KECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.74: α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷
 図 5.75: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

5.2.5 3次元データ

平均 $\mu = 0$, 標準偏差 $\sigma = 1$ の標準正規分布 (5.2) に従う 3 次元の橈円上の個体を 90 個, 105 個, 106 個, 110 個の個体数合計 411 個のデータセットに対して, 各手法でクラスタリングを行う. このデータセットを図 5.76 に示す. このデータに対して, $c = 4$ で初期値を 10 回変えてクラスタリングを行った.

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (5.2)$$

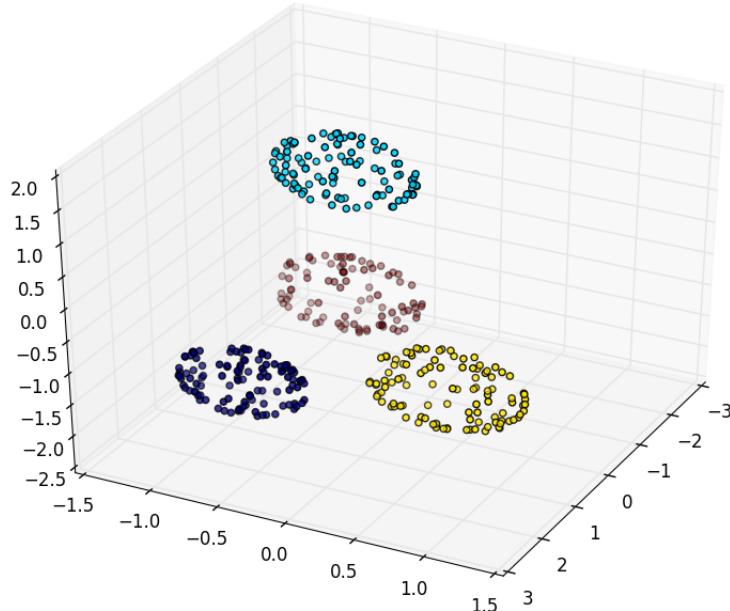


図 5.76: 球状の 3 次元データ (個体数 411 個)

以下に, 各提案手法および既存手法を用いたときのクラスタリング結果を図で示すとともに, 各手法において, ARI の値が最大となったときの目的関数, 目的関数の分散, ARI 値, 分割結果を表 5.6 およびグラフに示す.

クラスタサイズにマージンをつけた提案手法を用いることで α - L_1 ECBO を除いた全ての手法で ARI=1.0 となるクラスタリングを得ることが出来た. α -cosine-ECBO では, マージン α が 12 以上になると必ず自然な分割結果となり, 類似性尺度に $\cos \theta_{x,y}$ を用いたことで, 各クラスタ中心と個体のベクトル間の角度でクラスタリングを行えたことによるものであると思われる. α - L_1 ECBO のみ, 1 回の試行において Max100 回のクラスタ中心の最適化・シンプルレックス法による帰属度の更新を行っても, 収束しなかった. そのため, 収束しなかった中でも最も良い結果となった結果を図とグラフに示す.

表 5.6: 3 次元データに対する各手法の実行結果

手法	分割結果	Margin(α)	Min(J)	Var(J)	Max(ARI)
k -means	(90,105,106,110)	—	97.8	2.81E ⁺⁴	1.00
ECBO	(102,103,103,103)	—	1.47E ⁺²	2.15E ⁺²	0.88
α -ECBO	(90,105,106,110)	12~	1.05E ⁺²	2.02E ⁻²⁸	1.00
α -ECBO++	(90,105,106,110)	12~	1.03E ⁺²	2.02E ⁻²⁸	1.00
α -MECBO	(90,105,106,110)	15/17/20	1.92E ⁺²	2.39	1.00
α -KECBO	(90,105,106,110)	12~	1.51E ⁺²	0.00	1.00
α - L_1 ECBO	(99,102,105,105)	3	4.16E ⁺²	2.96E ⁺³	0.80
α -cosine-ECBO	(90,105,106,110)	12~	14.0	3.16E ⁻³⁰	1.00

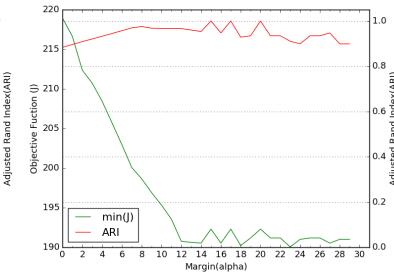
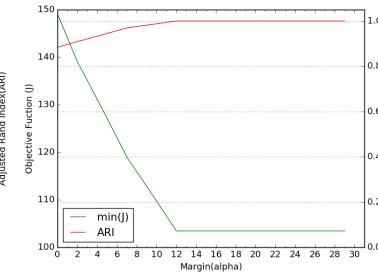
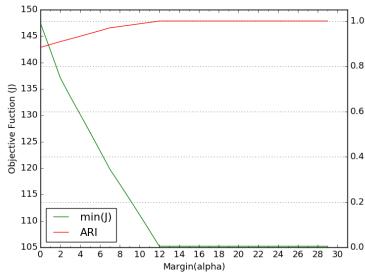


図 5.77: α -ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

図 5.78: α -ECBO++を用いたときの Margin の変化に伴う目的関数と ARI の変遷

図 5.79: α -MECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

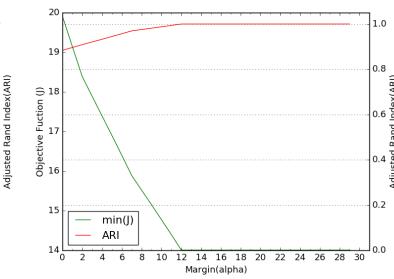
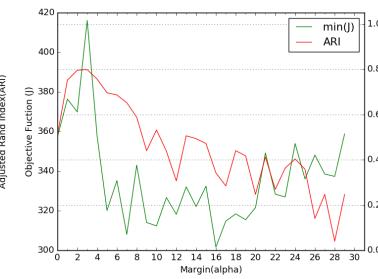
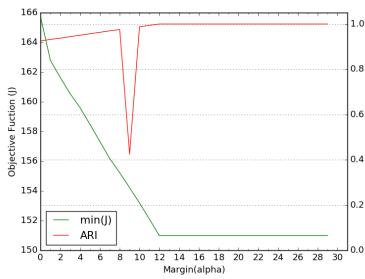


図 5.80: α -KECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

図 5.81: α - L_1 ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

図 5.82: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

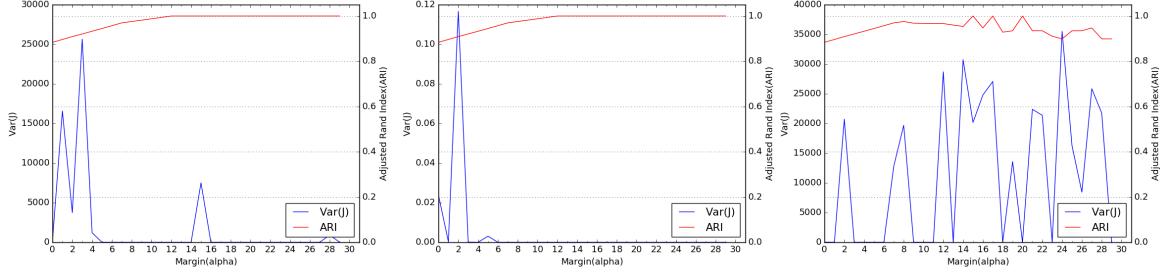


図 5.83: α -ECBO を用いたとき 図 5.84: α -ECBO++を用いたと 図 5.85: α -MECBO を用いたと
の Margin の変化に伴う目的関数の分散と ARI の変遷 関数の分散と ARI の変遷 関数の分散と ARI の変遷

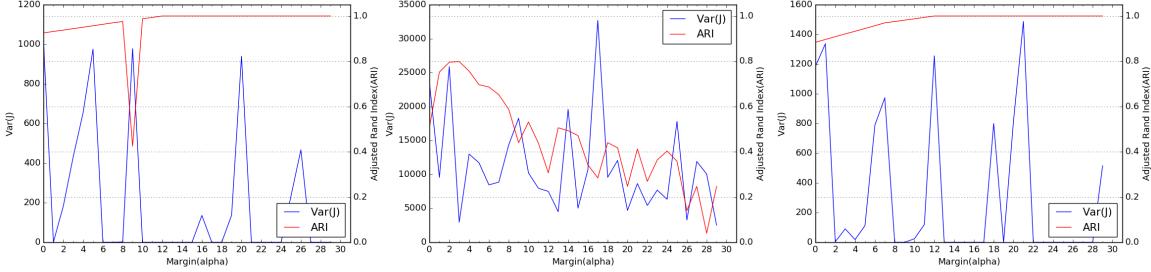


図 5.86: α -KECBO を用いたと 図 5.87: α - L_1 ECBO を用いたと 図 5.88: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷 関数の分散と ARI の変遷 関数の分散と ARI の変遷

5.3 実データ

5.3.1 Fisher's Iris データ

Fisher's Iris データ [24] は、品種の異なる 3 種類のアヤメ各 50 の花の、がく片の長さと幅、花弁の長さと幅の 4 個の計測値を与えた 4 次元のデータセットである。各手法において、ARI の値が最大となったときの目的関数、目的関数の分散、ARI 値、分割結果を表 5.7 およびグラフに示す。

Iris データは、各クラスタの個体数が全て 50 個と均一である。そのため、クラスタサイズの制約条件に幅を持たせていない ECBO で最良の ARI 値が得られると思ったが、クラスタサイズに幅を持たせた α -ECBO の方が、0.01 だけであるが、より良い結果となった。また、 α -cosine-ECBO で、最も高い ARI 値となっており、目的関数の分散もほぼ 0 で、初期値に依らず安定した結果を得ることが出来ていることが分かる。これは、4 次元の個体からなる Iris

データのため, n 次元空間でのベクトルの類似性を測る $\cos \theta_{x,y}$ がデータセットの特徴に上手く一致したためであると思われる。

表 5.7: Fisher's Iris データに対する各手法の実行結果

手法	分割結果	Margin(α)	目的関数 Min(J)	目的関数の分散 Var(J)	Max(ARI)
k -means	(35,50,65)	—	80.5	9.77E ⁺²	0.75
ECBO	(50,50,50)	—	81.4	1.14E ⁺³	0.79
α -ECBO	(49,50,51)	1	81.2	0.01	0.80
α -ECBO++	(48,50,52)	2	80.9	0.05	0.79
α -MECBO	(49,50,51)	1	85.2	0.54	0.80
α -KECBO	(48,50,52)	5	76.0	58.8	0.89
α - L_1 ECBO	(44,50,56)	8	1.44E ⁺²	5.45E ⁺³	0.89
α -cosine-ECBO	(47,50,53)	3	0.16	5.80E ⁻⁸	0.94

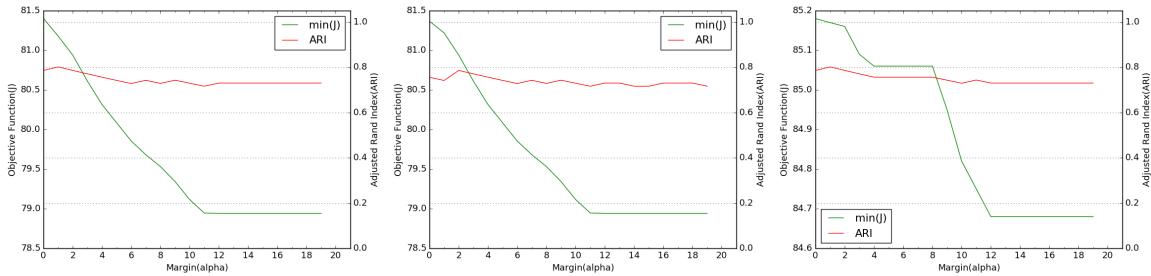


図 5.89: α -ECBO を用いたとき 図 5.90: α -ECBO++を用いたと 図 5.91: α -MECBO を用いたとの Margin の変化に伴う目的関数と ARI の変遷

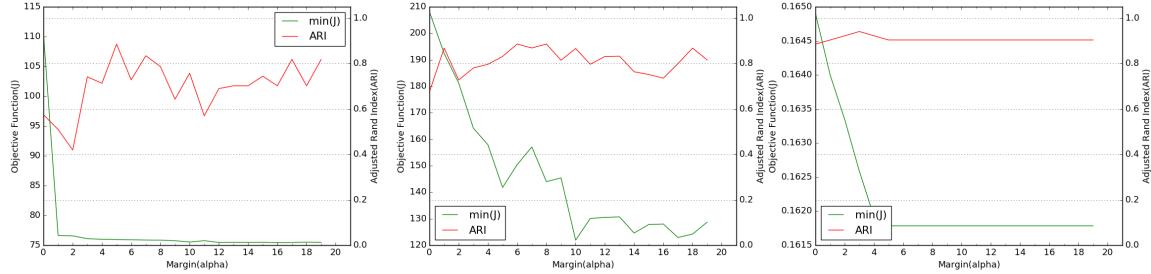


図 5.92: α -KECBO を用いたと 図 5.93: α - L_1 ECBO を用いたと 図 5.94: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数と ARI の変遷

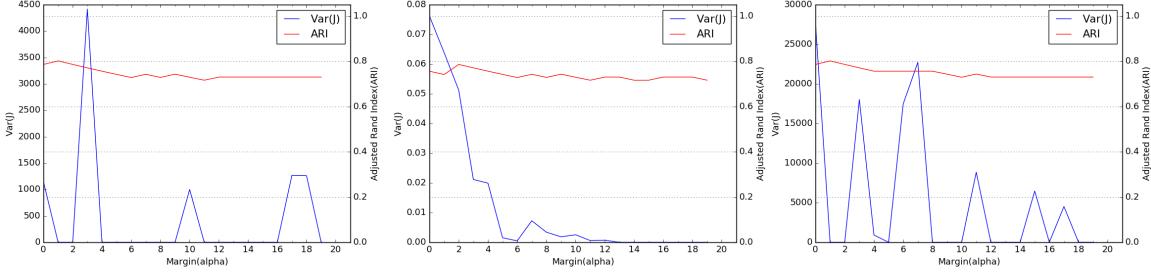


図 5.95: α -ECBO を用いたとき 図 5.96: α -ECBO++を用いたと 図 5.97: α -MECBO を用いたと
の Margin の変化に伴う目的関数の分散と ARI の変遷 関数の分散と ARI の変遷 関数の分散と ARI の変遷

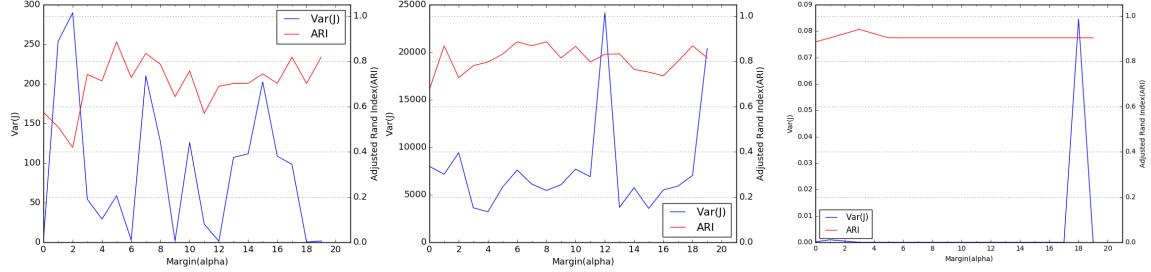


図 5.98: α -KECBO を用いたと 図 5.99: α - L_1 ECBO を用いたと 図 5.100: α -cosine-ECBO を用いたときの Margin の変化に伴う目的関数の分散と ARI の変遷 関数の分散と ARI の変遷 いたときの Margin の変化に伴う目的関数の分散と ARI の変遷

5.3.2 Wisconsin Breast Cancer データ

Wisconsin Breast Cancer データ [25] は、ウィスコンシン大学病院から得た乳がんに関するデータで、良性（がん細胞でない）、悪性（がん細胞である）の 2 つのクラスタに分類される。オリジナルのデータセットは、全 699 個のデータからなるが、今回の実験では、重複データや欠損データを全て排除し、全データ数 449 個に対してクラスタリングを行った。各個体は、9 つの属性を持つ。

以下に、各提案手法および既存手法において、ARI の値が最大となったときの目的関数、目的関数の分散、ARI 値、分割結果を表 5.8 およびグラフに示す。

α - L_1 ECBO を除き、マージンがある一定上になると、分割結果・ARI の値は一定で、大きく変化はない。よって、このようなデータセットでは、クラスタサイズに関する制約条件は結果にあまり影響を与えないと言える。

表 5.8: Breast Cancer データに対する各手法の実行結果

手法	分割結果	Margin(α)	Min(J)	Var(J)	Max(ARI)
k -means	(217,232)	—	1.79E ⁺²	1.18E ⁻⁸	0.73
ECBO	(224,225)	—	1.80E ⁺²	1.27E ⁺³	0.77
α -ECBO	(223,226)	1	1.80E ⁺²	1.16E ⁺³	0.76
α -ECBO++	(223,226)	1	1.80E ⁺²	4.50	0.76
α -MECBO	(224,225)	2/5~7/9/11/12/15/16/19	2.08E ⁺²	8.08E ⁻²⁸	0.77
α -KECBO	(211,238)	18	2.01E ⁺²	1.59E ⁻⁶	0.81
α - L_1 ECBO	(222,227)	2	6.93E ⁺²	7.34E ⁺⁴	0.77
α -cosine-ECBO	(223,226)	1	34.3	2.35	0.25

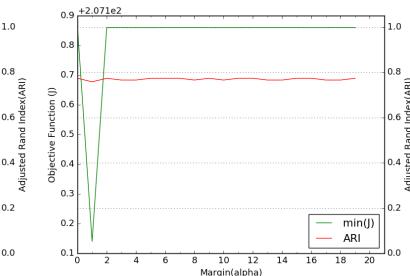
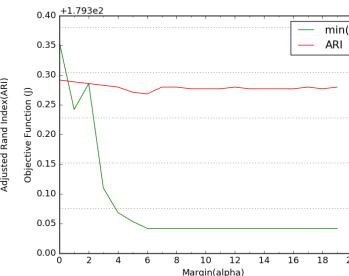
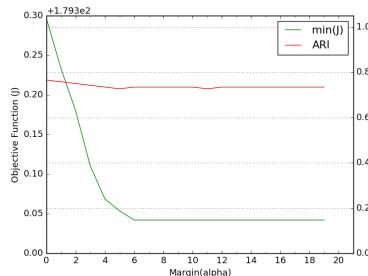


図 5.101: α -ECBO を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

図 5.102: α -ECBO++を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

図 5.103: α -MECBO を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

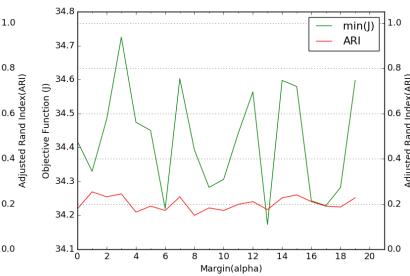
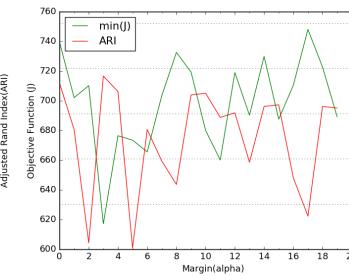
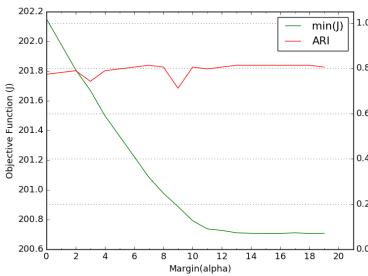


図 5.104: α -KECBO を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

図 5.105: α - L_1 ECBO を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

図 5.106: α -cosine-ECBO を用いたときの Margin の変化に伴う目的的関数と ARI の変遷

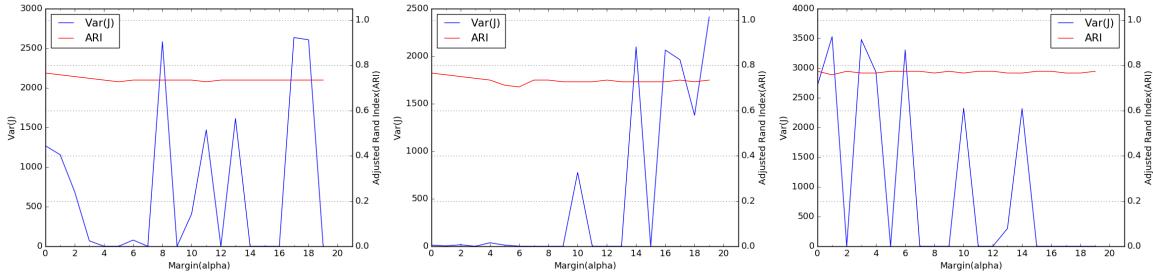


図 5.107: α -ECBO を用いたと 図 5.108: α -ECBO++を用いた 図 5.109: α -MECBO を用いた
きの Margin の変化に伴う目的 ときの Margin の変化に伴う目 ときの Margin の変化に伴う目
的関数の分散と ARI の変遷 的関数の分散と ARI の変遷 的関数の分散と ARI の変遷

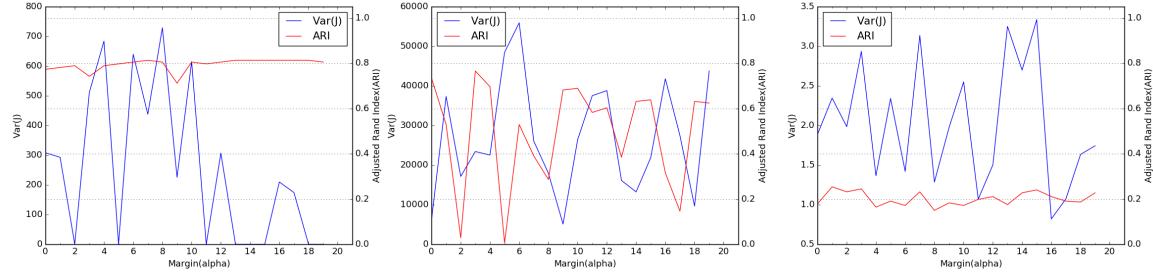


図 5.110: α -KECBO を用いた 図 5.111: α - L_1 ECBO を用いた 図 5.112: α -cosine-ECBO を用
ときの Margin の変化に伴う目 ときの Margin の変化に伴う目 いたときの Margin の変化に伴
的関数の分散と ARI の変遷 的関数の分散と ARI の変遷 う目的関数の分散と ARI の変
遷

第6章 結論

本研究では、既存手法である ECBO に対し、新たにクラスタサイズに幅を持たせるという概念を導入した、最適化に基づくマージン付きサイズ均等クラスタリングの手法を、既存のクラスタリング手法である k -means, k -means++, k -medoids, Kernel k -means, L_1 k -means, cosine k -means の 6 つの手法に基づくアルゴリズムの提案を行った。

線形分離が出来ない 2 重円のデータセットを除いた全てのデータで、提案手法である α -ECBO, α -ECBO++, α -MECBO の場合には、マージンの与え方によっては、安定した良い結果を得ることが出来ており、ECBO よりも目的関数、ARI ともに良い数値を得ることが出来た。

α -cosine-ECBO の結果は、人工データセットにおいて、3 次元のデータセットを除き、 $ARI = 1.0$ となるような良い分類結果を得ることが出来なかった。しかしながら、3 次元の人工データセットにおいては、コサイン相関を用いたクラスタリング手法の、ベクトル間の角度に基づき個体間の相関を取るという特徴により、データセットの各集合が形成する個体数に上手く一致するマージンの値からそれ以上になると、必ず大域的最適解に収束した。

提案手法の α - L_1 ECBO では、すべてのデータセットにおいて正解ラベルと同様の結果は得ることが出来なかった。しかしながら、実データとして用いた Iris データでは、 α -ECBO, α -MECBO, α -ECBO++よりも良い結果を得ることが出来ていることが、Margin の変遷のグラフから分かる。 L_1 k -means の特徴であるノイズに対してロバストな点が上手く作用したと思われる。

線形分離では上手くクラスタリングを行うことの出来ないデータセットとして、2 重円データを用いて、実験を行ったところ、 α -KECBO では、クラスタサイズのマージンによっては、円の内側と外側でデータを分割することが出来た。しかしながら、他のデータにこのアルゴリズムを適用したところ、望むような結果を得ることが出来なかった。理由としては、データの分散を調整するカーネル関数のパラメータ γ に、クラスタリングの結果が大きく依存するためだと思われる。

ECBO が初期値に依存しやすいのに対し、全ての提案手法において、マージンの与え方がデータセットに上手く一致した場合には、初期値や繰り返し回数に関係なく、 $ARI = 1.0$ となるような安定した結果を得ることが出来た。その理由としては、ECBO に比べ、クラスタサイズに関する制約が緩和されたためだと思われる。

また、人工データ・実データ全てのデータセットに共通して、マージン α が大きくなればなるほど、目的関数 J は右肩下がりに減少した。これは、クラスタサイズに関する制約条件が緩和されたことで、目的関数をより最小化するような解に収束するためである。また、制

約が緩くなることで、基づくクラスタリング手法を用いたときの結果に近づいていくことが、 ARI の変化に伴う目的関数の変遷のグラフから明らかである。

本研究に用いた数値例では、2次元の人工データや緯度経度などの数値データに適用したものを見た。しかしながら、第3章でも述べたように、サイズ均等クラスタリングは、個人情報利用時のプライバシ保護技術である、 K -匿名化のための利用が非常に有用であると考えられる。そこで、今後の課題としては、名前や性別、居住地、年収などからなるカテゴリカルデータを対象に、レコードを K 個以上に分割するという目的で利用することが考えられる。カテゴリカルデータを利用する際には、個体間の類似性尺度の定義が実数データとは異なってくるため、そういうデータを対象とした関連性の定義も必要であると考えられる。

また、今回の研究では、クラスタサイズのマージンを任意に与え、クラスタリングを行っていたが、与えるマージンによってクラスタリング結果がどれだけ大きく変化するのかということを、数理計画問題において、問題の係数が変化したときに最適解の変化を扱う問題である Sensitive Analysis（感度分析）[26] を用いて、マージンの変化がクラスタリング結果に与える影響の検証を行うこともできる。

更には、今回の提案手法におけるクラスタサイズの上限・下限は定数としていたが、クラスタサイズの上限・下限そのものをファジィ化してしまうことで、より制約を緩めた手法の提案も出来ると考えられる。

本研究で提案した手法は、全てのデータセット・解析に適しているとはいえないが、データを扱う側が望むある程度決まった個体数に分かれるクラスタリング結果がほしい場合には、有用な手法であるといえる。

謝辞

本論文の作成にあたり、指導教員である遠藤靖典教授には、日頃から研究に関するアドバイスやディスカッション、学会への参加だけでなく、大学院生活など多岐に渡り熱心なご指導を賜り、大変お世話になりました。研究室の一員として受け入れていただきましたこと、心より感謝致します。

副査を受け持って頂いた、ソフトコンピューティング基礎グループのイリチュ（佐藤）美佳教授には、中間発表での研究に関するご助言、データマイニングの授業やグループ内でのイベント等で大変お世話になりました。ありがとうございました。

また、同じく副査を受け持って頂きました、知能機能システム専攻濵谷長史助教には、達成度評価委員会等におきましても、貴重なアドバイスを頂き、深く感謝致します。

そして、日本学術振興会特別研究員であるソフトコンピューティング基礎グループの木下尚彦氏には、クラスタリングに関する基礎知識や研究の在り方、学会投稿論文の書き方等多くのご指導を頂きました。ここに感謝致します。

ソフトコンピューティング基礎グループの先輩、後輩、そして同期の皆さまには、研究室で共に過ごす仲間として、大変お世話になり、途中からにも関わらず、研究メンバーとして快く受け入れて頂けたことを心から嬉しく思います。本当にありがとうございました。

最後に、これまで24年もの間私を育て見守っていてくれた両親、どんな時でも私の支えとなってくれた兄と弟には、心から感謝しています。これからは、社会人として立派に成長し、恩返しをしていきます。これからも、よろしくお願い致します。

参考文献

- [1] 宮本 定明. クラスター分析入門. 森北出版株式会社, 1999.
- [2] Amazon. <https://www.amazon.co.jp/>.
- [3] How Facebook is Using Big Data - The Good, the Bad, and the Ugly.
<https://www.simplilearn.com/how-facebook-is-using-big-data-article>
- [4] トヨタ自動車 次世代テレマティクス.
https://www.toyota.co.jp/jpn/tech/smart_mobility_society/next_generation_telematics/.
- [5] 日立製作所 ビッグデータへの道.
<http://www.hitachi.co.jp/products/it/bigdata/column/column01.html>.
- [6] 経済産業省 個人情報保護法.
http://www.meti.go.jp/policy/it_policy/privacy/downloadfiles/01kaiseikojinjohopamphlet.pdf.
- [7] 個人情報とは、匿名化とは何か.
<http://business.nikkeibp.co.jp/article/opinion/20140722/269015/?rt=nocnt>.
- [8] 松崎和賢. データ匿名化の現状に関する一考察.
http://www-erato.ist.hokudai.ac.jp/docs/seminar/20110708_public.pdf. 2011.
- [9] Agrawal, R., Srikant, R. Privacy-preserving data mining. ACM Sigmod Record (Vol. 29, No. 2, pp. 439-450), 2000.
- [10] Lindell, Yehuda, and Benny Pinkas. Privacy preserving data mining. Annual International Cryptology Conference. Springer Berlin Heidelberg, 2000.
- [11] 新井淳也, 鬼塚真, 塩川浩昭. クラスタリングと空間分割の併用による効率的な k -匿名化. DBSJ Japanese journal 日本データベース学会和文論文誌 13.1, 2014:72-77.
- [12] J. MacQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp.281-297, 1967.
- [13] 小野田崇, 坂井美帆, 山田誠二. k -means 法の様々な初期値設定によるクラスタリング結果の実験的比較. 人工知能学会, 1J1-OS9-1 33, 2011.

- [14] 緒方 悠人, 遠藤 靖典. *K-Member Clustering* 問題に関する一考察. 第 29 回ファジィシステムシンポジウム (FSS2013), 2013.
- [15] 平野翼, 遠藤靖典. 最適化に基づくサイズ均等クラスタリング. 筑波大学大学院システム情報工学研究科修士論文, 2016.
- [16] D. Arthur, S. Vassilvitskii. *k-means++: The Advantages of Careful Seeding*. Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027-1035, 2007.
- [17] H.S. Park, C.H. Jun. A simple and fast algorithm for *K*-medoids clustering. *Expert Systems with Applications* 36.2, pp.3336-3341, 2009.
- [18] M. Girolami. Mercer Kernel-Based Clustering in Feature Space. *IEEE Trans. on Neural Networks*, vol.13, no.3, pp.780-784, 2002.
- [19] H. Kashima, et al. K-means clustering of proportional data using L1 distance. *Pattern Recognition*, 2008. ICPR 2008. 19th International Conference on. IEEE, 2008.
- [20] B. Ji-Won, et al. Efficient K-Anonymization Using Clustering Techniques. *International Conference on Database Systems for Advanced Applications*. Springer Berlin Heidelberg, 2007.
- [21] Lin, J. L., Wei, M. C. An efficient clustering method for k-anonymization. In *Proceedings of the 2008 international workshop on Privacy and anonymity in information society* (pp. 46-50). ACM, 2008.
- [22] He, X., Chen, H., Chen, Y., Dong, Y., Wang, P., Huang, Z. Clustering-Based k-anonymity. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 405-417). Springer Berlin Heidelberg, 2012.
- [23] L. Hubert, P. Arabie. Comparing partitions. *Journal of Classification* 2, pp.193–218, 1985.
- [24] R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, vol1.7, No.2, pp.179–188, 1936.
- [25] W. H. Wolberg. Breast Cancer Wisconsin (Original) Data Set.
[https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+\(Original\)](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)).
- [26] Sensitive Analysis-MIT. <http://web.mit.edu/15.053/www/AMP-Chapter-03.pdf>.