

集合間の距離に基づくクラスタの分布を用いた重み付きカーネル
k-平均法の初期クラスタ探索法

柳 信 一

北海道情報大学

A Search Algorithm for Initial Clusters of Weighted Kernel
k-means Method using the Distribution for Clusters Based on
the Distance between the Set

Shinichi YANAGI

Hokkaido Information University

平成28年11月

北海道情報大学紀要 第28巻 第1号別刷

〈論文〉

集合間の距離に基づくクラスタの分布を用いた重み付きカーネル
k-平均法の初期クラスタ探索法

柳 信一*

A Search Algorithm for Initial Clusters of Weighted Kernel
k-means Method using the Distribution for Clusters Based on
the Distance between the Set

YANAGI SHINICHI*

要旨

本論文では、集合間の距離に基づくクラスタの分布を用いた重み付きカーネル k -平均法の初期クラスタを計算する手法を提案する。重み付きカーネル k -平均法は、代表的なクラスタリング手法である k -平均法に基づいており、クラスタ同士の境界が非線形であるデータの分類が可能である。しかし、計算結果は入力である初期クラスタに強く依存するという問題点がある。提案手法は、集合間の距離に基づくクラスタの分布を用いて初期クラスタを決定する。いくつかの人工データと実データを用いて計算機実験を行い、提案手法の有効性を示す。

Abstract

This paper proposes a weighted kernel k -means algorithm to compute initial clusters using the distribution for clusters based on the distance between the set. The weighted kernel k -means algorithm is based on the k -means algorithm which is a well known clustering algorithm, and makes it possible to identify clusters that are non-linearly separable in input space. The proposed algorithm identifies clusters with the original weighted kernel k -means algorithm after searching initial clusters using the distribution for clusters based on the distance between the set. An experimental comparison in some artificial and real data set shows that a performance of the proposed algorithm is efficient.

キーワード

クラスタリング (clustering), k -平均法 (k -means method), カーネル法 (kernel method), 初期クラスタ (initial clusters)

1. まえがき

クラスタリングとは、データ集合をある基準によって分類することである。 k -平均法 (MacQueen, 1967) は代表的なクラスタリング手法であり、アルゴリズムが単純で、事前に必要とするパラメータがクラスタ数のみであるという利点のため、クラスタリングを必要とする計算過程において利用されることが多い。しかし、距離計算にユークリッド距離を使用するた

め、各クラスタの分離境界が非線形となるデータの分類に適していない。

重み付きカーネル k -平均法 (Dhillon, Guan and Kulis, 2004) は、 k -平均法に対してカーネル法 (Muller, Mika, Ratsch, Tsuda and Scholkopf, 2001) と重み付き距離計算を適用した手法であり、非線形に分離されているクラスタの分類が可能である。しかし、基となったカーネル k -平均法 (Dhillon, Guan and Kulis, 2004) 同様、計算結果が距離計算に使用するカーネル関数のパラメータに敏感であり、さらに、初期クラスタへの依存が強いという問題点がある。

* 北海道情報大学経営情報学部システム情報学科講師,
Lecturer, Department of Business and Information Systems, HIU

一方で、重み付きカーネル k -平均法と数学的に等価な手法として、いくつかの種類のスペクトラルクラスタリング (Chan, Schlag and Zien, 1994; Shi and Malik, 2000; Ng, Jordan and Weiss, 2002; Yu and Shi, 2003) が提案されている。スペクトラルクラスタリングは与えられたデータ集合をデータ間の類似度を定義することでグラフとみなし、クラスタリングをグラフのカット問題に帰着し、最終的な計算過程で k -平均法を利用する手法であり、カーネル k -平均法や重み付きカーネル k -平均法同様、非線形に分離されているクラスタの分類が可能である。スペクトラルクラスタリングにも、類似度関数に利用するパラメータ依存の問題が存在するが、適切なパラメータを選択する手法 (Zelnik-Manor and Perona, 2004; Yu, Tranchevent, De Moor and Moreau, 2011) が提案されている。初期クラスタ依存に関しては、重み付きカーネル k -平均法に比較すると軽減されているものの、固有値問題を解く必要があるため、計算量が大きく、さらに、数値計算を必要とする。

重み付きカーネル k -平均法のカーネルパラメータ依存に関しては、文献 (柳, 2016) において、集合間の距離を基に初期クラスタを決定し、初期クラスタを基準にパラメータ探索を行う手法が提案されている。文献 (柳, 2016) では、重み付きカーネル k -平均法およびスペクトラルクラスタリングとの比較実験を行っており、ある程度の有効性を示しているものの、クラスタが部分的に分離していたり内部が密集していないクラスタが混在しているデータに対しては結果が不安定となる場合がある。

本論文では、文献 (柳, 2016) の手法において、初期クラスタの探索基準を改良したアルゴリズムを提案する。いくつかの人工データと実データを用いて比較実験を行い提案手法の有効性を示す。

以下、本論文は次のように構成されている。2章で従来の重み付きカーネル k -平均法の定義と従来手法に関する説明を行う。3章では新たに重み付きカーネル k -平均法の初期クラスタ探索法を提案する。4章では従来手法と提案手法の比較実験を行い、提案手法の有効性を検討する。5章では本論文のまとめを行う。

2. 重み付きカーネル k -平均法

クラスタリングとは、対象とする n 個の

データからなるデータ集合 $X = \{x_i \mid i = 1, 2, \dots, n, \forall x_i \in R^d\}$ を何らかの基準により k 個の部分集合 $C_j = \{x \mid \forall x \in X\}$ に分類することであり、この部分集合 $C_j (j = 1, 2, \dots, k)$ をクラスタと呼ぶ。一般的にクラスタリングを必要とする場面では、データ集合のみが与えられ、クラスタ数 k は未知であることもあるが、本論文では k を既知のものとして取り扱う。

2-1 k -平均法

k -平均法は各データ $x_i \in X (i = 1, 2, \dots, n)$ と、クラスタの中心 $m_j \in R^d (j = 1, 2, \dots, k)$ との距離の総和を最小とする最適化問題として式 (1) で定式化される。

$$\begin{aligned} \min_{C_j} \sum_{j=1}^k \sum_{x_i \in C_j} \|x_i - m_j\|^2, \\ \text{s.t. } \bigcup_{j=1}^k C_j = X, \\ C_j \cap C_l = \emptyset, 1 \leq j, l \leq k, j \neq l. \end{aligned} \quad (1)$$

式 (1) の $\|\cdot\|$ は L_2 ノルムである。式 (1) の解はクラスタ $C_j (j = 1, 2, \dots, k)$ であり、クラスタ中心 m_j は式 (1) より、 C_j に属するデータ数 $|C_j|$ を用いて式 (2) のように求まる。

$$m_j = \frac{1}{|C_j|} \sum_{x \in C_j} x, (j = 1, 2, \dots, k). \quad (2)$$

式 (1) の定式化は暗に各クラスタが凸形状となっていることを仮定しているため、線形分離不可能なクラスタからなるデータの分類に適していない。

2-2 重み付きカーネル k -平均法

重み付きカーネル k -平均法 (Dhillon, Guan and Kulis, 2007) は k -平均法にカーネル法を適用した手法である。

カーネル法は、無限次元まで含めた、現在の空間の次元数 d よりはるかに大きい $D (\gg d)$ 次元空間上に、写像 ϕ により $x \in R^d$ を非線形写像し、空間 R^D 上でデータの線形分離を試みる方法である。最大の特徴は、計算量削減のため実際の写像 $\phi(x)$ 自体の計算は行わず、式 (3) のカーネル関数を利用して $\phi(x)$ と $\phi(x')$ の内積のみを使用することで問題を解くことである。

$$K(x, x') = \phi(x) \cdot \phi(x'). \quad (3)$$

具体的なカーネル関数としては、カーネル関数が高次元空間における内積となっていることを保証するマーセルカーネル (Máté, 1989) が望ましく、特に多くの研究では、代表的なマーセルカーネルである式 (4) のガウスカーネルが使用されている。

$$K(x, x') = \exp\left(\frac{-\|x - x'\|^2}{\delta^2}\right). \quad (4)$$

式 (4) において、 δ はスケールパラメータを表す。

重み付きカーネルk-平均法は式 (1) の最適化問題の目的関数を式 (5) のように、高次元空間へ写像されたデータを用い、さらに、各データ x_i の重み w_i を考慮した形式に変換した手法である。

$$\begin{aligned} \min_{C_j} \sum_{j=1}^k \sum_{x_i \in C_j} w_i \|\phi(x_i) - M_j\|^2, \\ \text{s.t. } \bigcup_{j=1}^k C_j = X, \\ C_j \cap C_l = \emptyset, 1 \leq j, l \leq k, j \neq l. \end{aligned} \quad (5)$$

したがって、式 (5) よりクラスタ中心は式 (6) となる。

$$M_j = \frac{\sum_{x_i \in C_j} w_i \phi(x_i)}{\sum_{x_i \in C_j} w_i}, M_j \in R^D (j = 1, 2, \dots, k). \quad (6)$$

式 (6) より、クラスタ中心 $M_j \in R^D$ とデータ $\phi(x_i) \in R^D$ の距離は式 (7) で計算できる。

$$\begin{aligned} & \|\phi(x_i) - M_j\|^2 \\ &= \|\phi(x_i)\|^2 - 2M_j \cdot \phi(x_i) + \phi(x_i) \cdot \phi(x_i) \\ &= \frac{\sum_{x_t, x_l \in C_j} w_t w_l K(x_t, x_l)}{(\sum_{x_t \in C_j} w_t)^2} - \frac{2 \sum_{x_t \in C_j} w_t K(x_t, x_i)}{\sum_{x_t \in C_j} w_t} \\ & \quad + K(x_i, x_i), (j = 1, 2, \dots, k). \end{aligned} \quad (7)$$

式 (5) の問題は、重み w_i と $K(x_i, x_j)$ を i 行 j 列の要素とするカーネル行列 K を適切に設定することにより、いくつかの種類の目的関数のスペクトラルクラスタリングと数学的に等価になることが証明されている (Dhillon et al., 2007)。本論文では、 $w_i = 1 / \sum_{j=1}^n K(x_i, x_j)$ とした。式 (5) の問題は、可能な C_j の選択に関して組合せ的な探索を必要とする離散最適化であるため NP 困難となる。したがって、局所解

の出力を認める発見的アルゴリズムを用いて計算する。重み付きカーネルk-平均法のアルゴリズムを Algorithm 1 に示す。Algorithm 1

Algorithm 1 入力: $X, k, \delta, C_1^{(1)}, C_2^{(1)}, \dots, C_k^{(1)}$

- 1 繰返し番号 t を $t = 1$ で初期化する。
 - 2 各データに対して各クラスタ $C_j^{(t)}$ の中心との距離を式 (7) により計算し、最も近いクラスタ中心のクラスタに割り当てる。
 - 3 2 の結果から式 (5) の目的関数を再計算し、変化が無ければ $C_j^{(t)} (j = 1, 2, \dots, k)$ を出力して計算を終了する。そうでなければ $t = t + 1$ として 2 へ戻る。
-

の入力 $C_j^{(1)} (j = 1, 2, \dots, k)$ は初期クラスタである。重み付きカーネルk-平均法は初期クラスタ依存度が強く、初期クラスタとして、対象データ $x \in X$ をランダムに割り当ててもうまく分類できない。そこで、Dhillon らは、データをグラフの節点とみなし適切な枝でグラフを構成した後、グラフの粗大化 (Karypis and Kumar, 1998) と局所探索 (Dhillon, Guan and Kogan, 2002) により初期クラスタを決定している。

グラフの粗大化は、節点集合 V 、枝集合 E 、データ間の類似度を要素とする類似度行列 K からなるグラフ $G(V, E, K)$ が入力として与えられたとき、グラフ G の規模が小さくなるように粗く表現したグラフ $G'(V', E', K') (|V| > |V'|)$ を作成し、問題を解く手法である。任意の2つの節点の結合を段階的に繰返し、節点数が $5k$ 以下であれば、その節点集合をデータ集合とみなして重み付きカーネルk-平均法を実行する。次に、得られたクラスタリング結果に関して、粗大化を一段階解除し、前段階のクラスタリング結果を初期クラスタとして、再び重み付きカーネルk-平均法を実行する。以下、粗大化が全て解除され、元のデータ集合が復元されるまで繰り返す。さらに、各段階で重み付きカーネルk-平均法を実行した後、局所探索を行う。局所探索は、目的関数が最小となるよう、ただ一つのデータに対して、現在のクラスタから他の異なるクラスタに割り当て変更を行う。この処理を目的関数が減少しなくなるまで繰返す。

2-3 文献 (柳, 2016) の手法

文献 (柳, 2016) の手法は、初期クラスタと初

期パラメータ δ を入力として、各2対の初期クラスタ間を分離できるようなパラメータを決定し、重み付きカーネル k -平均法を行う。初期クラスタは、データ集合間、すなわち、クラスタ間の距離が十分離れていることを基準に決定する。まず、任意の有限集合 $A = \{a \mid a \in R^d\}$ と $B = \{b \mid b \in R^d\}$ の集合間の距離 $d(A, B)$ を式(8)により定式化する。

$$d(A, B) = \min_{\forall a \in A, \forall b \in B} \|a - b\|. \quad (8)$$

式(8)は任意の有限集合 A, B, C に関して必ずしも三角不等式 $d(A, B) \leq d(A, C) + d(C, B)$ を満たすとは限らないため、距離の公理を満たしておらず、厳密な意味での距離とはならないが、本論文では距離と呼ぶことにする。次に、クラスタ間の距離が十分に離れているデータ集合という基準は式(9)を満たすデータ集合 X の $k(\geq 2)$ 個の部分集合を要素とする集合を $disjoint(X, k)$ と定義することで定量化する。

$$\begin{aligned} disjoint(X, k) = \{ & X_i \mid \bigcup_{i=1}^k X_i = X, \\ & X_i \cap X_j = \emptyset, \\ & \max_{\forall A \subseteq X_i} d(A, X_i - A) \leq d(X_j, X_i), \\ & 1 \leq i, j, t \leq k, j \neq t\}. \end{aligned} \quad (9)$$

式(9)は $disjoint(X, k)$ の要素を任意に2分割した部分集合同士の距離が、 $disjoint(X, k)$ の要素同士の距離より小さいことを意味する。また、集合 Y を要素とする集合 $Z = \{Y_1, Y_2, \dots, Y_n\}$ があたえられたとき、 $\forall Y \in Z$ の最近傍要素 $NN(Y)$ を式(10)で定義する。

$$NN(Y) = S = \arg \min_{\forall S \in Z - \{Y\}} d(Y, S). \quad (10)$$

N の各要素の最近傍要素に関して距離が大きい上位 k 個の要素からなる集合を式(11)のように $NN_k(N)$ と定義する。

$$\begin{aligned} NN_k(N) = \{ & T \mid |NN_k(N)| = k, \\ & d(T, NN(T)) \geq d(T', NN(T')), \\ & \forall T' \in N - NN_k(N)\}. \end{aligned} \quad (11)$$

X の部分集合 $T \subseteq X$ からなる集合 $N = \{T \mid T \subseteq X, T \cap T' = \emptyset, \forall T' \in N\}$ が与えられたとき、 N の部分集合 S を要素とする集合 $D_k(N)$ を式(12)で定義する。

$$\begin{aligned} D_k(N) = \{ & S \mid S = \{T \mid T \in N, \\ & (NN(T) \in S) \vee \\ & ((\exists T' \in S) \wedge (T' \notin NN_k(N)), \\ & NN(T') = T)\}, \\ & \bigcup_{\forall S \in D_k(N)} S = N\}. \end{aligned} \quad (12)$$

式(12)は N の各要素 T を結合して S を構成することで得られる集合を意味し、任意の1つの要素 $T \in N$ を $S = \{T\}$ とした後、逐次的に構成できる。 T の最近傍要素 $NN(T)$ がすでに S に含まれているとき、または、 T がすでに S に含まれている要素 T' の最近傍要素であるときに $T \in S$ となる。ただし、 $T' \in S$ が $NN_k(N)$ の要素のときは、 T を S の要素としない。2対の要素 T と T' が互いに最近傍とは限らない、すなわち、 $NN(T) = T'$ かつ $NN(T') = T$ とは限らないため、式(12)のような定式化となる。

式(12)を用いて X の部分集合を要素とする集合 $D_{i,k}$ を式(13)のように再帰的に定義する。

$$D_{i,k} = \begin{cases} D_k(X) & (i = 1) \\ D_k(D_{i-1,k}) & (i > 1) \end{cases} \quad (13)$$

最終的に、 $|D_{i,k}| = k$ を満たすとき、 $D_{i,k} = D_k$ を初期クラスタとすることで $disjoint(X, k) = D_k$ を満たす。もし X の任意の2点間の距離が全て異なるのであれば、 D_k は一意に定まる。一方で X が外れ値を含んでいたり、クラスタ内の分布が極端に異なるようなデータ集合の場合、 $disjoint(X, k)$ がクラスタリングの意味で適切な分割とはならない可能性がある。そこで、上記のような状況に対応できるよう、整数値 r を用いて $D_{i,k}$ を式(14)のように $D_{i,k}^r$ に緩和する。

$$D_{i,k}^r = \begin{cases} D(X) & (i = 1, r > 1) \\ D_k(X) & (i = 1, r = 1) \\ D(D_{i-1,k}^r) & (i < r) \\ D_k(D_{i-1,k}^r) & (i \geq r) \end{cases}. \quad (14)$$

ここで、式(14)の $D(X)$ は式(12)において $NN_k(N) = \emptyset$ を強要すること、つまり、 T を最近傍要素とする T' が S に含まれていれば、かならず $T \in S$ とすることを意味する。式(14)の $D_{i,k}^r$ が $|D_{i,k}^r| = k$ を満たすとき、 $D_{i,k}^r = D_k^r$ と定義し初期クラスタとする。 $r = 1$ のときは、 $D_k^r = D_k$ となる。また、 $r = 2$ のときは、一つのデータが D_k^r の要素となること、すなわ

ち, $\exists\{x\} \in D_k^r$ となることを避けることができるため, 外れ値対策となる。 $D \in D_{r-1,k}^r$ ならば $D \in D_{r,k}^r$ とはならないため, 以下 r が増加するにつれて距離が大きい集合間の結合が速くなると考えられる。 D_k^r を求めるアルゴリズムを Algorithm 2 に示す。

Algorithm 2 入力: $r, k, X = \{x_1, x_2, \dots, x_n\}$

- 1 $j \leftarrow 1$ とする。
- 2 一つのデータを初期クラスタとしてクラスタ集合 $D_k = \{C_i \mid C_i = \{x_i\}, i = 1, 2, \dots, n\}$ を作成する。
- 3 $|D_k| > k$ なら $DSAVE \leftarrow D_k$ とする。
- 4 $|D_k| = k$ なら D_k を出力して終了する。
- 5 $|D_k| < k$ なら $D_k \leftarrow DSAVE, r \leftarrow 1$ とする。
- 6 $T \leftarrow \emptyset$ とする。
- 7 $\forall C \in D_k$ に対して $NN(C)$ を求める。
- 8 $j \geq r$ ならば, $d(C, NN(C))$ が大きい上位 k 個の C を T の要素とし, $D_k \leftarrow D_k - T$ とする。
- 9 $D \leftarrow \emptyset$ とする。
- 10 $\forall C \in D_k$ を一つ選択する。
- 11 以下の更新を行う。

$$\begin{aligned}
 D &\leftarrow D \cup \{C\} \\
 D_k &\leftarrow D_k - \{C\} \\
 C &\leftarrow NN(C)
 \end{aligned}$$

- 12 $C \in D_k$ なら 11 へ戻る。
 - 13 $C \notin S, \forall S \in T$ なら $T \leftarrow T \cup \{D\}$ とし, そうでなければ $C \in S$ を満たす S に以下の処理を行う。
- $$\begin{aligned}
 T &\leftarrow T - \{S\} \\
 T &\leftarrow T \cup \{S \cup \{D\}\}
 \end{aligned}$$
- 14 $D_k \neq \emptyset$ なら 9 へ戻る。そうでなければ, $D_k \leftarrow T, j \leftarrow j + 1$ とし 3 へ戻る。
-

次に, 初期クラスタ集合 D_k^r を分離するパラメータの計算に関して, パラメータ探索区間 $[d_{\min}, d_{\max}]$ を式 (15) により定める。

$$\begin{aligned}
 d_{\min} &= \epsilon (> 0), \\
 d_{\max} &= \max_{\forall x, x' \in C_i \cup C_j, 1 \leq i, j \leq k, i \neq j} \frac{\sqrt{d(x, x')}}{2}. \quad (15)
 \end{aligned}$$

D_k^r の各 2 対のクラスタ $\forall C_a, C_b \in D_k^r$ に対して, 初期パラメータ $\delta_1 = d_{\min}$ から始めて, $\delta_q > d_{\max}$ となるか, または, 式 (18) を満たすまで, 式 (16), 式 (17) により δ_q を逐次更新していく。

$$\delta_q = \rho_{q-1} \delta_{q-1}. \quad (16)$$

$$\rho_q = \sqrt{\frac{1}{1 - \frac{\|M_a^{(q-1)} - x_a\|^2 - \|M_b^{(q-1)} - x_a\|^2}{3}}}. \quad (17)$$

$$\|M_a^{(q-1)} - \phi(x_a)\|^2 \leq \|M_b^{(q-1)} - \phi(x_a)\|^2. \quad (18)$$

ここで, $M_a^{(q-1)}$ は $q-1$ 番目の $\delta_{(q-1)}$ により定まるクラスタ C_a の中心点である。

δ を更新するためのアルゴリズムを Algorithm 3 に示す。

Algorithm 3 入力: $C_a, C_b, d_{\min}, d_{\max}$

- 1 繰返し番号を $q = 1$ とし, $\delta_q = d_{\min}$ とする。
 - 2 C_b に対して式 (7) を最小とする $x_a \in C_a$ を求める。
 - 3 C_a, C_b, x_a, δ_q より式 (17) で ρ_q を計算する。
 - 4 以下のいずれかであれば, δ_q を出力し計算を終了する。
 - 式 (18) を満たす
 - $\rho_q = 1$
 - $\delta_q \geq d_{\max}$
 - 5 $\delta_q = \rho_q \delta_q, q = q + 1$ とし, 2 へ戻る。
-

各 2 対のクラスタにより決定された $k(k-1)/2$ 個の $\delta_{i,j} (1 \leq i, j \leq k, i \neq j)$ のうち $\delta_{i,j} \geq d_{\max}$ となる個数を n_δ とし, n_δ の値によって式 (19) の基準で最終的な δ を決定する。

$$\delta = \begin{cases} d_{\max} & (n_\delta \geq k-1) \\ \max_{1 \leq i, j \leq k, i \neq j} \delta_{i,j} & (\text{otherwise}) \end{cases}. \quad (19)$$

以上により定まった初期クラスタ集合 $D_k^r = \{C_1, C_2, \dots, C_k\}$ とガウスカーネルのパラメータ δ を用いて重み付きカーネル k -平均法を実行する。クラスタリングアルゴリズムを Algorithm 4 に示す。

3. 提案手法

重み付きカーネル k -平均法はカーネルパラメータに加え, 初期クラスタへの依存も強いいため, 初期クラスタの段階で適切に分類されていることが望ましい。文献(柳, 2016)の初期クラスタ探索法は, クラスタ間が十分離れ

Algorithm 4 入力: X, k, r

- 1 X, k, r を入力とし, Algorithm 2 により初期クラスタ C_1, C_2, \dots, C_k を求める。
- 2 式 (15) より δ の探索区間 $[d_{\min}, d_{\max}]$ を決定する。
- 3 全ての 2 対の初期クラスタ C_i, C_j , ($1 \leq i, j \leq k, i \neq j$) に対して, $C_i, C_j, [d_{\min}, d_{\max}]$ を入力として, Algorithm 3 により $\delta_{i,j}$ を求める。
- 4 $\delta_{i,j}$ ($1 \leq i, j \leq k, i \neq j$) と $[d_{\min}, d_{\max}]$ を用いて, 式 (19) より δ を決定する。
- 5 $X, k, \delta, C_1, C_2, \dots, C_k$ を入力として Algorithm 1 を実行し, クラスタ C_1, C_2, \dots, C_k を再計算する。
- 6 C_1, C_2, \dots, C_k を出力して計算を終了する。

ており, かつ, クラスタ内のデータ間の距離がほぼ均一であるようなデータにはある程度有効性を示しているものの, クラスタ同士で重なっているものと分離しているものが混在していたり, 内部が密集していないクラスタが混在しているデータに対しては不安定である。その理由を従来手法の初期クラスタ D_k^r が段階的に構成される仕組みより説明する。1 段階目の初期クラスタ集合 $D_{1,k}^r$ をデータ集合 X , つまり, $D_{1,k}^r = \{x_1, x_2, \dots, x_n\}$ とし, t 回目の結合で得られるクラスタ集合を $D_{t,k}^r = \{D_1^{(t)}, D_2^{(t)}, \dots, D_{m_t}^{(t)}\}$ とする。 m_t は t 段階目の初期クラスタ集合のクラスタ数である。 $t-1$ 段階目から t 段階目に遷移するとき, $D_i^{(t)}$ ($1 \leq i \leq m_t$) は式 (12) を満たす全ての $T = D_j^{(t-1)}$ ($1 \leq j \leq m_{t-1}$) を含むため, 各段階におけるクラスタ数 m_t は t によって大きく異なる。ここで, 通常クラスタリングは教師なし学習であり, 正解が存在するわけではないが, 従来手法により適切な初期クラスタが得られる場合と, そうでない場合の考え方を簡単に説明するため, 適切なクラスタリング結果という意味で, 正解クラスタ集合 $A = \{A_i \mid \cup_i A_i = X, A_i \cap A_j = \emptyset, 1 \leq i, j \leq k, i \neq j\}$ が存在すると仮定する。また, X の任意の 2 点間の距離は全て異なるものとする。まず, $\text{disjoint}(X, k) = \{A_1, A_2, \dots, A_k\}$ を満たす場合, 適切な r を選択することで, Algorithm 2 は $A = \{A_1, A_2, \dots, A_k\}$ を初期クラスタとして出力する。すなわち, t 段階目の初期クラスタ数 m_t を $m_t = k$ とすると, $D_{t,k}^r = \{D_1^{(t)}, D_2^{(t)}, \dots, D_{m_t}^{(t)}\} = \{A_1, A_2, \dots, A_k\}$ と

なる。各クラスタ $D_i^{(t)}$ ($1 \leq i \leq k$) が t_i 段階目で完成すると仮定すると, $r \leq \min_i \{t_i\}$ となることから, $D_{t,k}^r = \{D_1^{(t)}, D_2^{(t)}, \dots, D_{m_t}^{(t)}\} = \{A_1, A_2, \dots, A_k\}$ となる十分条件となる。文献(柳, 2016)の実験結果より, $\text{disjoint}(X, k) = \{A_1, A_2, \dots, A_k\}$ であれば, $r = 2$ とすることで, 大体適切な分類ができると考えられる。次に, $\text{disjoint}(X, k) \neq \{A_1, A_2, \dots, A_k\}$ の場合を考える。 $X' = A_2 \cup A_3 \cup \dots \cup A_k$ に対して $\text{disjoint}(X', k) = \{A_2, A_3, \dots, A_k\}$ を満たし, $\arg \min_{A' \in X - \{A_1\}} d(A_1, A') = A_2$ と仮定する。可能な限り正解クラスタに近い初期クラスタを構成するという観点より, この場合の従来手法の目標は, ある $Y \subset A_1 \cup A_2$ を用いて, 初期クラスタ集合が $D_{t,k}^r = \{Y, A_1 \cup A_2 - Y, A_3, \dots, A_k\}$ となることとなる。このとき, $\max_{Y \subset (A_1 \cup A_2)} d((A_1 \cup A_2) - Y, Y) > \max_{3 \leq i \leq k} \{\max_{Y \subset A_i} d(A_i - Y, Y)\}$ を満たす Y が存在し, Y が t_1 段階目, $(A_1 \cup A_2) - Y$ は t_2 段階目, $3 \leq i \leq k$ において A_i が t_i 段階目で得られるとすると, $r \leq \min_i \{t_i\}$ と設定できれば, 初期クラスタ集合として, $\{Y, (A_1 \cup A_2) - Y, A_3, \dots, A_k\}$ が得られる。しかし, 任意のデータ集合 X において上記のような Y が存在するとは限らない。そのような場合, $A_1 \cup A_2$ が t' 段階目で構成されるとすると, $\max_{3 \leq i \leq k} \{t_i\} \leq t'$ を満たすとき, 初期クラスタ集合として, $\{Y, (A_1 \cup A_2) - Y, A_3, \dots, A_k\}$ が得られる。一般的に, $\text{disjoint}(X, k)$ を満たさない正解クラスタが A_1 だけではなく複数存在する場合は, 最近傍集合同士を結合して初期クラスタを構成していく過程において, 各 $A_i \in \text{disjoint}(X, k)$ が構成される段階 t_i が, 任意の 2 つの正解クラスタ $A_j, A_l \notin \text{disjoint}(X, k)$ が結合して初期クラスタ $A_j \cup A_l$ となる段階 t' に対して, $t_i \leq t'$ を満たせばよいことになる。つまり, 全ての A_i が全ての $A_j \cup A_l$ より早い段階で構成されればよい。ただし, 従来手法では t' を自分で調整することはできず, 集合同士の最近傍距離により決定されるため, パラメータ r の調整で対応することはできない。正解クラスタが構成される段階の速度調整の考え方として, 何らかの基準を設け, 最近傍とならない集合同士も結合するか, または, 最近傍となる集合同士でも結合しないかの二つの戦略が考えられるが, 後者の考え方を採択する。すなわち, 距離

が大きい上位 k 個の最近傍データ同士以外は全て無条件で結合するという従来手法の戦略を破棄し、結合の可否に関する選択を設ける。その際、結合可否の選択基準が必要となるが、各正解クラスタの構成段階 t_i が異なる理由として、ある段階の正解クラスタ内の部分集合に関して、最近傍となる部分集合数に差がある、すなわち、クラスタ内のデータの密度の差が大きい事が挙げられる。そのため、十分離れている集合同士は同じ初期クラスタとはならず、かつ、得られる初期クラスタ内のデータの密度に依存せず、同じ段階で同程度の集合数を結合していくことができれば、上記問題点を解決できると考える。そこで、階層的に集合の最近傍同士の結合を繰り返すことで生起するクラスタの確率分布を定義し、その情報を結合の可否の選択に利用することを考える。

本論文では、文献(柳, 2016)の初期クラスタ探索法を改良し、集合同士の距離を基に計算目標とするクラスタの分布を定義し、結果として得られるクラスタの分布との差が小さくなるよう集合同士を結合していくことで初期クラスタを構成する手法を提案する。目標とするクラスタの確率分布 P は、集合同士の最近傍距離を基に、以下の2点を満たすように定義する。

- (1) クラスタ内のデータ密度に確率が依存しないようにする。
 - (2) 十分離れている集合同士が同じクラスタ内に生じる確率を小さくするようにする。
- まず、クラスタ D_i 内の最近傍距離の集合 $dist_i$ を式 (20) で定義する。

$$dist_i = \{d \mid d = d(D_i - A, A), \forall A \subset D_i\}. \quad (20)$$

データ集合 X の任意の2点間の距離は、画像データのように離散値をとる場合、同じものが存在する可能性もあるが、本論文では全て異なるものと仮定する。この場合、クラスタ集合 $D = \{D_1, D_2, \dots, D_k\}$ が $disjoint(X, k) = D$ を満たすなら、 D_i の最近傍距離集合 $dist_i$ は式 (13) の各 $D_{j,k} \subseteq D_i$ の要素とその最近傍との距離の j に関する和集合に一致する。次に、式 (21) のように、クラスタ D_i 内の最近傍距離 $d' \in dist_i$ の総和を d_i とする。

$$d_i = \sum_{d' \in dist_i} d'. \quad (21)$$

式 (21) より、クラスタ D_i の内部距離 w_i を式

(22) で定義する。

$$w_i = \frac{d_i}{|dist_i| \cdot \max\{dist_i\}}. \quad (22)$$

式 (22) の $|dist_i|$ は $dist_i$ の要素数である。式 (22) は D_i の最近傍距離を、その最大値 $\max\{dist_i\}$ で割り、平均をとっているため正規化の効果があり、データの密度の大小関係への依存は比較的弱いと考えられ、上記 (1) の効果を満足することを期待できる。さらに、最大値 $\max\{dist_i\}$ で割ることにより、極端に大きな最近傍距離が $dist_i$ に含まれるときは、相対的に小さくなると考えられ、上記 (2) の効果を期待できる。各内部距離を、その総和 $w = \sum_{i=1}^M d_i$ で割ることにより、確率 P_i を式 (23) で定義する。

$$P_i = \frac{w_i}{w}. \quad (23)$$

また、実際の計算結果として得られる確率 Q は、単純に各クラスタが含むデータ数 $|D_i|$ を全データ数 n で割り $Q_i = |D_i|/n$ と定義する。 P と Q の差が小さくなるような初期クラスタを構成することを目指す、その差を測る尺度として、カルバック・ライブラー・ダイバージェンス(以下、KLダイバージェンス)を用いる。KLダイバージェンスは確率分布 $P = \{P_1, P_2, \dots, P_n\}$ と $Q = \{Q_1, Q_2, \dots, Q_n\}$ の違いを表す測度であり、式 (24) で定義される。

$$D_{KL}(P||Q) = \sum_{i=1}^n P_i \log \frac{P_i}{Q_i} \quad (24)$$

データ集合 X とクラスタ数 k が与えられたときの初期クラスタ D_i の構成基準、すなわち、各段階でどの最近傍集合を結合するかを選択基準を式 (25) の最適化問題として定式化する。

$$\begin{aligned} \min & \sum_{i=1}^k P_i \log \frac{P_i}{Q_i}, \\ \text{s.t. } & P_i = \frac{w_i}{\sum_{i=1}^k w_i}, \\ & w_i = \frac{1}{|dist_i| \cdot \max\{dist_i\}} \sum_{d' \in dist_i} d', \\ & Q_i = \frac{|D_i|}{m}. \end{aligned} \quad (25)$$

式 (25) の最適化問題を解くためには、可能な $dist_i$ を全て検査しなければならないため、組み合わせてきな計算量を必要とする。そこで、逐次評価により近似解を求める。

ここで、あらためて、クラスタ集合の初期値 $D^{(1)}$ をデータ集合 X , つまり, $D^{(1)} = \{x_1, x_2, \dots, x_n\}$ とし, t 回目の結合で得られるクラスタ集合を $D^{(t)} = \{D_1^{(t)}, D_2^{(t)}, \dots, D_M^{(t)}\}$ とする。各段階で2つの集合 $D_0^{(t)} \in D^{(t)}$ と $\text{NN}(D_0^{(t)}) \in D^{(t)}$ を結合し, $D_0^{(t+1)} = D_0^{(t)} \cup \text{NN}(D_0^{(t)}) \in D^{(t+1)}$ を構成する。 $D^{(t)}$ と $D^{(t+1)}$ の差異は $(D^{(t)} - D^{(t+1)}) \cup (D^{(t+1)} - D^{(t)}) = \{D_0^{(t)}, \text{NN}(D_0^{(t)}), D_0^{(t+1)}\}$ である。次に t 段階目の KL ダイバージェンス $D_{\text{KL}}(P||Q)^{(t)}$ と $t+1$ 段階目の KL ダイバージェンス $D_{\text{KL}}(P||Q)^{(t+1)}$ を計算し, 各 KL ダイバージェンスをクラスタ数で割った値を比較する。 $D_{\text{KL}}(P||Q)^{(t+1)}/|D^{(t+1)}| < D_{\text{KL}}(P||Q)^{(t)}/|D^{(t)}|$ であれば, 上記の結合を実行し, そうでなければ結合しないこととする。 $t+1$ 回目の KL ダイバージェンスは各確率を更新して計算し直す必要があるが, 計算量のコストを抑えるため, 確率分布の更新に影響する部分のみ計算する。簡単のため, t 段階目の KL ダイバージェンス $D_{\text{KL}}(P||Q)^{(t)}$ を $I^{(t)}$ と表記する。また, t 段階目のクラスタ数を M とし, 一般性を失わず, $D_{M-1}^{(t+1)} = D_{M-1}^{(t)} \cup D_M^{(t)}$ を評価するものとする, $t+1$ 段階目の各確率は式 (26) のように更新される。

$$\begin{aligned} P_{M-1}^{(t+1)} &= \frac{w_{M-1}^{(t+1)}}{w^{(t+1)}}, \\ w_{M-1}^{(t+1)} &= \frac{d_{M-1}^{(t)} + d_M^{(t)} + d(D_{M-1}^{(t)}, D_M^{(t)})}{|\text{dist}_{M-1}^{(t+1)}| \cdot \max\{\text{dist}_{M-1}^{(t+1)}\}}, \\ \text{dist}_{M-1}^{(t+1)} &= \text{dist}_{M-1}^{(t)} \cup \text{dist}_M^{(t)} \cup \\ &\quad \{d(D_{M-1}^{(t)}, D_M^{(t)})\}, \\ w^{(t+1)} &= w^{(t)} - w_{M-1}^{(t)} - w_M^{(t)} + w_{M-1}^{(t+1)}, \\ P_i^{(t+1)} &= \frac{w^{(t)}}{w^{(t+1)}} P_i^{(t)}, 1 \leq i \leq M-2, \\ Q_{M-1}^{(t+1)} &= Q_{M-1}^{(t)} + Q_M^{(t)}, \\ Q_i^{(t+1)} &= Q_i^{(t)}, 1 \leq i \leq M-2. \end{aligned} \quad (26)$$

また, t 回目と $t+1$ 回目の KL ダイバージェンスは式 (27) となる。

$$\begin{aligned} I^{(t)} &= \sum_{i=1}^M P_i^{(t)} \log \frac{P_i^{(t)}}{Q_i^{(t)}}, \\ I^{(t+1)} &= \sum_{i=1}^{M-2} P_i^{(t+1)} \log \frac{P_i^{(t+1)}}{Q_i^{(t+1)}} + P_{M-1}^{(t+1)} \log \frac{P_{M-1}^{(t+1)}}{Q_{M-1}^{(t+1)}}. \end{aligned} \quad (27)$$

式 (26) と式 (27) より, $a^{(t+1)} = 1 - P_{M-1}^{(t)} - P_M^{(t)}$ として, 式 (28) の漸化式を得る。

$$\begin{aligned} I^{(t+1)} &= \frac{w^{(t)}}{w^{(t+1)}} I^{(t)} + \frac{w^{(t)}}{w^{(t+1)}} a^{(t+1)} \log \frac{w^{(t)}}{w^{(t+1)}} \\ &\quad - \frac{w^{(t)}}{w^{(t+1)}} P_{M-1}^{(t)} \log \frac{P_{M-1}^{(t)}}{Q_{M-1}^{(t)}} \\ &\quad - \frac{w^{(t)}}{w^{(t+1)}} P_M^{(t)} \log \frac{P_M^{(t)}}{Q_M^{(t)}} \\ &\quad + P_{M-1}^{(t+1)} \log \frac{P_{M-1}^{(t+1)}}{Q_{M-1}^{(t+1)}}. \end{aligned} \quad (28)$$

式 (28) と式 (26) より, $I^{(t)}$ と M 個の要素を持つ3つのリスト $d_i^{(t)}, |\text{dist}_i^{(t)}|, \max\{\text{dist}_i^{(t)}\} (1 \leq i \leq M)$ をキャッシュに保存しておき, 式 (28) に関わる $t+1$ 段階目の値を逐次更新していくことで, $I^{(t+1)}$ は定数オーダで計算できる。また, 逐次更新での比較となるため, t 段階目と $t+1$ 段階目の計算過程における確率分布の確率の数が1つ異なる。そのため, t 段階目の状態から i 番目と j 番目のクラスタ $D_i^{(t)}$ と $D_j^{(t)}$ を結合して得られるクラスタ集合を $D^{(t+1)}(i, j)$, その KL ダイバージェンスを $I_{ij}^{(t+1)}$ とし, $t+1$ 段階目のクラスタ集合 $D^{(t+1)}$ を式 (29) により決定する。

$$D^{(t+1)} = \begin{cases} D^{(t+1)}(i, j) & \left(\frac{I_{ij}^{(t+1)}}{|D^{(t)}| - 1} \leq \frac{I^{(t)}}{|D^{(t)}|} \right) \\ D^{(t)} & (\text{otherwise}) \end{cases} \quad (29)$$

さらに, ある段階で式 (29) を満たす $D_i^{(t)}, D_j^{(t)}$ の対が1組も存在しない可能性があるため, そのときは, 式 (30) を満たす $D_i^{(t)}, D_j^{(t)}$ を強制的に結合する。

$$\begin{aligned} \min_{i, j} & \frac{I_{ij}^{(t+1)}}{|D^{(t)}| - 1} - \frac{I^{(t)}}{|D^{(t)}|}, \\ \text{s.t. } & D_i^{(t)} = \text{NN}(D_j^{(t)}), \\ & D_j^{(t)} = \text{NN}(D_i^{(t)}). \end{aligned} \quad (30)$$

式 (30) は, $t+1$ 段階目における1クラスタ当たりの KL ダイバージェンスの増加が最小で, かつ, 互いに最近傍要素となっている2対のクラスタを結合することを意味する。

ここで, 式 (29) の評価順は任意であるため, 式 (26) の dist_i は探索順に依存することになり, 厳密に式 (20) を満たすとは限らない。すなわ

ち, $D_a = \{x_1, x_2, x_3\}$ と $D_b = \{x_2, x_1, x_3\}$ において, $dist_a = dist_b$ となるとは限らない。生起確率に着目して考えると, $t+1$ 段階目のクラスタの生起確率 $P^{(t+1)}$ は t 段階目のクラスタの生起確率 $P^{(t)}$ に対して独立ではなく, $p(D_i^{(t+1)} | D_j^{(t)})$ を i 行 j 列目の要素とする M 行 M 列の遷移確率行列により定められる。したがって, 各段階におけるクラスタ集合の確率ベクトルは, クラスタ集合の確率ベクトルの初期値と各段階における遷移確率行列との積により確定することになる。厳密な $p(D_i^{(t+1)} | D_j^{(t)})$ の検討は今後の課題とし, 本論文では式 (26) により確率分布を計算する。

また, クラスタ集合の初期値 $D^{(1)}$ はデータ集合とするが, このとき内部距離を定義することができないため, 1 段階目の計算では KL ダイバージェンスによる結合基準を使用せず, 最近傍となるデータ同士を無条件で結合する。初期クラスタを求めるアルゴリズムを Algorithm 5 に示す。Algorithm 5 では, 表記を簡単にするため, クラスタ D_i は, その最近傍距離の総和 d_i , 最近傍距離集合の要素数 $|dist_i|$, 最近傍距離の最大値 $\max\{dist_i\}$ の 3 つの値を保持しているものとし, KL ダイバージェンス I は, その内部距離の総和 w を保持しているものとする。上記の設定により, 関数 $I(D_i^{(t)}, D_j^{(t)}, I^{(t)})$ により, 式 (28) と式 (26) から得られる $t+1$ 番目の KL ダイバージェンス $I^{(t+1)}$ を計算できるものとする。

最終的な提案手法のアルゴリズムを Algorithm 6 に示す。

3-1 計算量

Algorithm 1 の重み付きカーネル k -平均法の計算量のオーダーは Algorithm 1 の繰返し数 t , クラスタ数 k , データ数 n , データの次元数 d を用いて $\mathcal{O}(tkn^3d)$ と表すことができる。Algorithm 5 の計算量のオーダーは $\mathcal{O}(n^2(n+d))$, Algorithm 3 の計算量のオーダーは Algorithm 3 の繰返し数 q を用いて $\mathcal{O}(qn^2d)$ となる。したがって, Algorithm 6 の計算量のオーダーは $\mathcal{O}(n^2(n+d) + qn^2d + tkn^3d)$ となり, $q < tkn$ ならば Algorithm 1, そうでなければ Algorithm 3 に依存する。一方で, 従来法の計算量のオーダーは, $\mathcal{O}(n^2(\log n + d) + qn^2d + tkn^3d)$ であり, やはり Algorithm 1 と Algorithm 3 に依存している。Algorithm 2 と Algorithm 5 の影響はないように見えるが, 実際には初期ク

Algorithm 5 入力: $k, X = \{x_1, x_2, \dots, x_n\}$

- 1 $j \leftarrow 1$ とする。
 - 2 一つのデータを初期クラスタとしてクラスタ集合 $D_k = \{C_i \mid C_i = \{x_i\}, i = 1, 2, \dots, n\}$ を作成する。
 - 3 $|D_k| > k$ なら $DSAVE \leftarrow D_k$ とする。
 - 4 $|D_k| = k$ なら D_k を出力して終了する。
 - 5 $T \leftarrow \emptyset$ とする。
 - 6 $|D_k| < k$ なら, $\forall C \in DSAVE$ に対して $NN(C)$ を求め, $d(C, NN(C))$ が大きい上位 k 個の C を T の要素とし, $D_k \leftarrow DSAVE - T, j \leftarrow 1$ とする。そうでなければ, $\forall C \in D_k$ に対して $NN(C)$ を求める。
 - 7 $j \geq 2$ なら, D_k より $I^{(j-1)}$ を計算する。
 - 8 $D \leftarrow \emptyset$ とする。
 - 9 $\forall C \in D_k$ を一つ選択する。
 - 10 以下の更新を行う。

$$D \leftarrow D \cup \{C\}$$

$$D_k \leftarrow D_k - \{C\}$$

$$C \leftarrow NN(C)$$
 - 11 $j \geq 2$ なら, 以下の処理を行う。
 - i $C \in D_k$ なら $I^{(j)} \leftarrow I(C, D, I^{(j-1)})$ とし, $I^{(j-1)}$ と $I^{(j)}$ が式 (29) を満たすなら 12 へ進む。
 - ii $C \in S, \exists S \in T$ なら $I^{(j)} \leftarrow I(S, D, I^{(j-1)})$ とし, $I^{(j-1)}$ と $I^{(j)}$ が式 (29) を満たすなら 13 へ進む。
 - iii $C \in D$ ならば, $T \leftarrow T \cup \{D\}$ とし 8 へ戻る。
 - iv $T \leftarrow T \cup \{D\}, D \leftarrow \emptyset$ とし 10 へ戻る。
 - 12 $C \in D_k$ なら 10 へ戻る。
 - 13 $C \notin S, \forall S \in T$ なら $T \leftarrow T \cup \{D\}$ とし, そうでなければ $C \in S$ を満たす S に以下の処理を行う。

$$T \leftarrow T - \{S\}$$

$$T \leftarrow T \cup \{S \cup \{D\}\}$$
 - 14 $D_k \neq \emptyset$ なら 8 へ戻る。
 - 15 $|DSAVE| = |T|$ なら式 (30) を満たす $D, D' \in T$ より, $D_k \leftarrow (T - \{D\} - \{D'\}) \cup \{D \cup D'\}$ とし, そうでなければ, $D_k \leftarrow T$ とする。
 - 16 $j \leftarrow j+1$ とし 3 へ戻る。
-

ラスタが, Algorithm 1 と Algorithm 3 に影響を与えるため, 提案手法の計算コストが高くなる可能性もある。

4. 実験

実験は, 教師ラベル付きの人工データと実データに対して行う。実験環境は, Intel Xeon

Algorithm 6 入力: X, k

- 1 X, k を入力とし, Algorithm 5 により初期クラスタ C_1, C_2, \dots, C_k を求める。
- 2 式 (15) より δ の探索区間 $[d_{\min}, d_{\max}]$ を決定する。
- 3 全ての 2 対の初期クラスタ $C_i, C_j, (1 \leq i, j \leq k, i \neq j)$ に対して, $C_i, C_j, [d_{\min}, d_{\max}]$ を入力として, Algorithm 3 により $\delta_{i,j}$ を求める。
- 4 $\delta_{i,j} (1 \leq i, j \leq k, i \neq j)$ と $[d_{\min}, d_{\max}]$ を用いて, 式 (19) より δ を決定する。
- 5 $X, k, \delta, C_1, C_2, \dots, C_k$ を入力として Algorithm 1 を実行し, クラスタ C_1, C_2, \dots, C_k を再計算する。
- 6 C_1, C_2, \dots, C_k を出力して計算を終了する。

2.00 GHz 4 コア \times 2 のデュアルプロセッサ, および, 24GB RAM のワークステーションで, OS は 64 ビット版 Linux を使用した。gcc 4.8.4 を使用して C++ により Algorithm 1 ~ 6 を実装した。

4-1 クラスタリング結果の評価方法

教師ラベル付きのデータのクラスタリング結果に対して評価を行う代表的な指標として, Adjusted Rand Index (ARI) (Hubert and Arabie, 1985) や Normalized Mutual Information (NMI) (Strehl and Ghosh, 2003) がある。本論文では NMI を評価指標として用いる。あらかじめ教師ラベルにより決定されている k 個のクラスタ集合を $C^t = \{C_1^t, C_2^t, \dots, C_k^t\}$, 実際にクラスタリングすることにより得られる k 個のクラスタ集合を $C = \{C_1, C_2, \dots, C_k\}$, そして, $n = |C_1^t \cup C_2^t \cup \dots \cup C_k^t| = |C_1 \cup C_2 \cup \dots \cup C_k|$ をデータ数とするとき, C^t と C のエントロピーは式 (31) で定義される。

$$H(C^t) = - \sum_{j=1}^k \frac{|C_j^t|}{n} \log \frac{|C_j^t|}{n},$$

$$H(C) = - \sum_{i=1}^k \frac{|C_i|}{n} \log \frac{|C_i|}{n}. \quad (31)$$

以降, 本論文では C^t を正解クラスタ集合と呼ぶことにする。また, C^t と C の結合エントロピーは式 (32) で定義される。

$$H(C^t, C) = - \sum_{i=1}^k \sum_{j=1}^k \frac{|C_i \cap C_j^t|}{n} \log \frac{|C_i \cap C_j^t|}{n}. \quad (32)$$

NMI の計算式は式 (31) と式 (32) を用いて, 式 (33) により定義される。

$$\text{NMI} = \frac{H(C^t) + H(C) - H(C^t, C)}{\sqrt{H(C^t)H(C)}}. \quad (33)$$

NMI は 0 と 1 の間の値をとり, 正解クラスタと実際に得られたクラスタが完全に一致するとき, $H(C^t) = H(C) = H(C^t, C)$ より, 最大値の 1 をとる。

4-2 実験条件

まず, Algorithm 3 のパラメータ推定における最大繰返し数を 1000 とし 1000 を越えても計算を終了しない場合は強制的に計算を終了するようにした。また, ガウスパラメータの探索区間の左端点は文献 (柳, 2016) 同様, ガウス関数の値が計算可能な有効桁以上となるように, 倍精度浮動小数型変数の正の最小値である 2.25074×10^{-308} と, 計算対象データのデータ間の最大距離 D_{\max} を用いて式 (34) とした。

$$d_{\min} = \frac{D_{\max}}{\sqrt{308 \log_e 10 - 1}}. \quad (34)$$

Algorithm 1 の重み付きカーネル k -平均法の最大繰返し数を 100 回とし, 100 回の繰返しを越えても計算を終了しない場合は強制的に計算を終了するようにした。従来手法の Algorithm 2 の D_k^r のパラメータは $r = 2, 3, 4$ と変化させた。

提案手法の Algorithm 5 におけるデータの結合評価順をランダムに変化させ, 提案手法と従来手法共に 30 回の試行を行った。

4-3 使用データの詳細

人工データは全て 2 次元とし, 分類精度の評価に適していると思われるものを 19 種類用意した。表 1 に各データの詳細を示し, 図 1~15 に各データの分布状態を示す。

実データに関しては UCI machine learning repository (Frank and Asuncion, 2010) より, iris, wine, ionosphere, glass, soybeans-small, satimage, pen digit の 7 つのデータを使用した。satimage と pen digit に関しては, 元のデータのデータ数が非常に多いため, 各クラスタからランダムに 80 個を選択した。各データの詳細を表 2 に示す。

4-4 クラスタリング精度の比較

表 3 に各データに対する重み付きカーネル

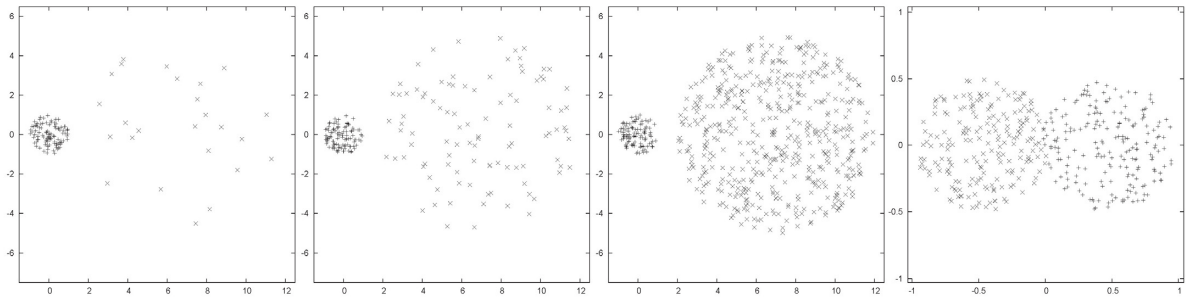


図1 データ1

図2 データ2

図3 データ3

図4 データ4

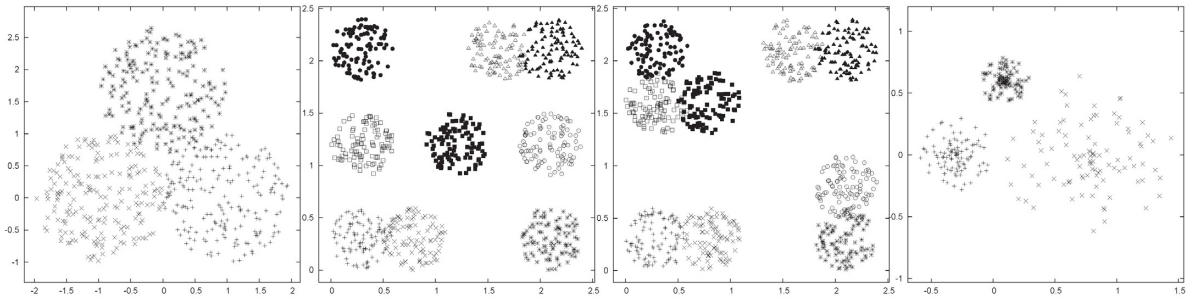


図5 データ5

図6 データ6

図7 データ7

図8 データ8

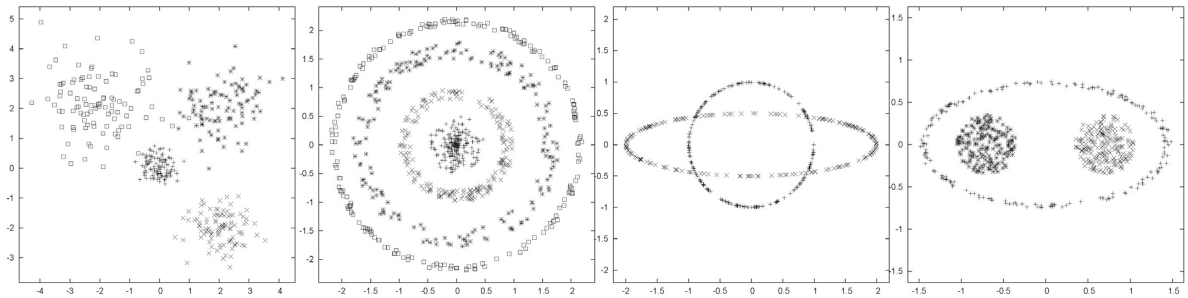


図9 データ9

図10 データ10

図11 データ11

図12 データ12

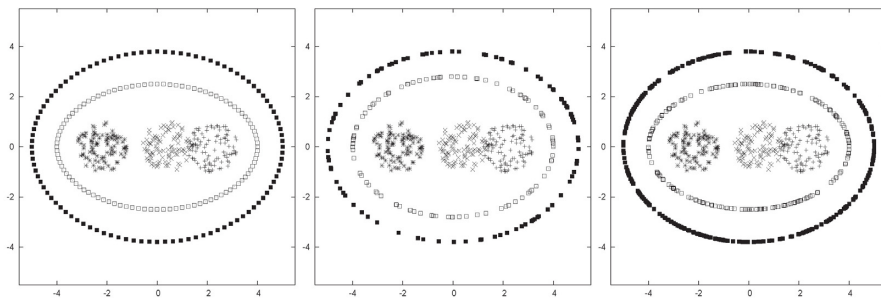


図13 データ13

図14 データ14

図15 データ15

k-平均法による最終的な NMI 値と Algorithm 5 による初期クラスタの計算が終了した時点での NMI 値 (A1.5) および Algorithm 2 による初期クラスタの計算が終了した時点での NMI 値 (A1.2) を示す。各手法に対して 30 回の試行を行ったが、従来手法に関しては、iris と wine の $r = 4$ の NMI 値を除いて、文献 (柳, 2016) 同様全て同じ計算結果であったため、値を 1 つだけ示してある。iris と wine の $r = 4$ の NMI 値は、それぞれ 2 種類の結果が得られたため、その平均を示してある。提案手法に関しては、初期クラスタの計算が終了した時点での KL

ダイバージェンスの値を保持しておき、KL ダイバージェンスが最小となる値の初期クラスタで計算した結果を NMI 値の代表とした。また、NMI 値と NMI 値 (A1.5) の括弧の中は最小と最大である。表 3 よりデータ 1~3 に関しては、データ 1 の結果のみ提案手法の評価結果が良く、データ 2 とデータ 3 に関しては同等の評価結果であった。図 1~3 より、大小 2 つの円の半径は同じだが、内部の密度を変化させており、データ 1 は大きな円のデータの密度が極端に小さくなっている。大きな円のクラスタ内の部分集合に対する最近傍集合との距離に 2

表 3 各データに対する NMI 値の比較

データ	$D_{KL}(P Q)$	提案手法		従来手法					
		NMI	NMI(A15.)	$r = 2$		$r = 3$		$r = 4$	
				NMI	NMI(A12.)	NMI	NMI(A12.)	NMI	NMI(A12.)
1	0.247	1 (1, 1)	1 (1, 1)	0.167	0.167	0.576	0.576	0.766	0.237
2	0.007	1 (1, 1)	1 (1, 1)	1	1	1	1	1	1
3	0.007	1 (1, 1)	1 (1, 1)	1	1	1	1	1	1
4	0.248	0.915 (0.202, 0.915)	0.382 (0.202, 0.382)	0.915	0.024	0.915	0.058	0.915	0.202
5	0.2	0.514 (0.473, 0.95)	0.493 (0.481, 0.493)	0.964	0.053	0.23	0.239	0.23	0.239
6	0.198	0.971 (0.925, 0.971)	0.934 (0.914, 0.938)	0.93	0.919	0.93	0.916	0.892	0.898
7	0.069	0.942 (0.883, 0.942)	0.858 (0.805, 0.858)	0.876	0.764	0.889	0.779	0.892	0.77
8	4.5×10^{-4}	0.907 (0.907, 1)	0.886 (0.886, 1)	1	1	1	1	1	1
9	0.018	0.958 (0.951, 0.958)	0.94 (0.94, 0.94)	0.825	0.827	0.951	0.94	0.951	0.94
10	0.005	1 (1, 1)	1 (1, 1)	1	1	1	1	1	1
11	0.003	0.002 (0.002, 0.085)	0.002 (0.002, 0.085)	0.006	0.041	0.08	0.054	0.085	0.085
12	0.072	1 (1, 1)	1 (1, 1)	1	1	1	1	1	1
13	0.096	0.959 (0.794, 0.959)	0.917 (0.891, 0.917)	0.866	0.903	0.897	0.891	0.824	0.677
14	0.014	0.962 (0.721, 0.962)	0.882 (0.882, 0.917)	0.689	0.866	0.689	0.866	0.766	0.866
15	0.027	0.861 (0.606, 0.875)	0.861 (0.831, 0.894)	0.898	0.898	0.898	0.898	0.898	0.898
iris	0.009	0.798 (0.798, 0.901)	0.871 (0.871, 0.871)	0.71	0.736	0.798	0.871	0.856	0.871
wine	0.058	0.432 (0.418, 0.432)	0.42 (0.42, 0.42)	0.442	0.416	0.442	0.416	0.424	0.42
iono.	0.055	0.135 (0.041, 0.135)	0.121 (0.036, 0.121)	0.058	0.058	0.069	0.069	0.024	0.018
glass	0.03	0.361 (0.335, 0.427)	0.308 (0.304, 0.407)	0.394	0.388	0.369	0.37	0.369	0.37
soy.	0.023	0.718 (0.718, 0.753)	0.718 (0.718, 0.753)	0.793	0.793	0.848	0.848	0.848	0.848
sat.	0.019	0.568 (0.557, 0.652)	0.644 (0.602, 0.644)	0.369	0.369	0.467	0.397	0.603	0.602
pen.	0.051	0.752 (0.752, 0.806)	0.733 (0.733, 0.807)	0.215	0.215	0.6	0.6	0.811	0.81

表 1 人工データの詳細

データ名	図番号	サンプル数	クラス数
データ1	図 1	125	2
データ2	図 2	200	2
データ3	図 3	600	2
データ4	図 4	400	2
データ5	図 5	600	3
データ6	図 6	900	9
データ7	図 7	900	9
データ8	図 8	300	3
データ9	図 9	400	4
データ10	図 10	800	4
データ11	図 11	400	2
データ12	図 12	600	3
データ13	図 13	500	5
データ14	図 14	500	5
データ15	図 15	800	5

つのクラスターの距離より大きいものが含まれているため、従来手法の r の調整では正解クラスターを得ることは出来ないと考えられる。また、提案手法に関しても、データ 1 は、クラスター内のデータ数に大きく偏りがあるため、ある程度計算が進むと、設計したクラスターの生起確率 P

表 2 実データの詳細

データ名	次元数	サンプル数	クラス数
iris	4	150	3
wine	13	178	3
ionosphere	35	351	2
glass	9	214	6
soybean-small	35	47	4
satimage	36	480	6
pen digit	16	800	10

とクラスター内データ数と全データ数の比である Q との KL ダイバージェンスは低下しなくなると考えられるが、全ての結合において KL ダイバージェンスの減少が見られない場合は、最も増加が小さく、かつ、互いに最近傍となっている集合同士を 1 組だけ結合するという条件がうまく働いたと考えられる。

次にデータ 4~7 に関しては、凸形状のデータ分布が完全に重なっている場合と部分的に重なっている場合である。データ 4 に関してはほぼ同じ評価結果となった。データ 5 に関しては、提案手法の NMI 値が従来手法の $r = 2$ に比してかなり小さくなっているが、Algorithm

5が出力した初期クラスタのNMI値とAlgorithm 2が出力した初期クラスタのNMI値を比較すると提案手法が勝っている。提案手法が、ある程度適切な初期クラスタを得たため、Algorithm 3でパラメータ探索に成功し、最終的な重み付きカーネルk-平均法が、ある程度初期クラスタを反映した分類を行ったのに対して、従来手法は適切な初期クラスタを得ることが出来ず、Algorithm 3でカーネルパラメータの最大値を得たことにより、最終的な分類結果が初期クラスタに依存しなかったためと考えられる。従来手法の $r = 2$ におけるAlgorithm 2によるクラスタリング結果を図16に示す。データ6とデータ7に関しては、いずれも提案

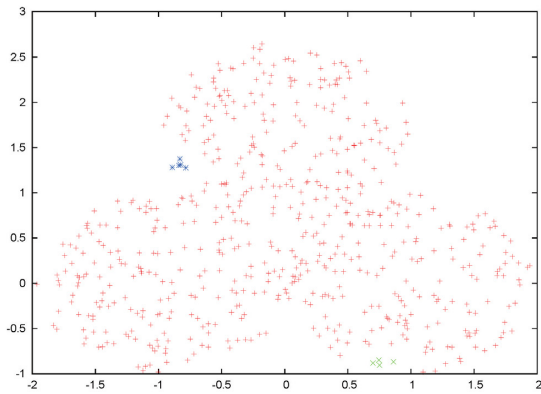


図16 Al. 2によるデータ5の初期クラスタ ($r = 2$)

手法の評価が高かった。いくつかのクラスタが重なっているものの、クラスタ内のデータ数にそれほど偏りがないため、設計したクラスタの生起確率が、このデータのクラスタをうまく表現できていたと考えられる。提案手法のAlgorithm 5、および、従来手法の $r = 4$ におけるAlgorithm 2とAlgorithm 4によるデータ7のクラスタリング結果を、それぞれ図17、図18、図19に示す。図18より、従来手法の

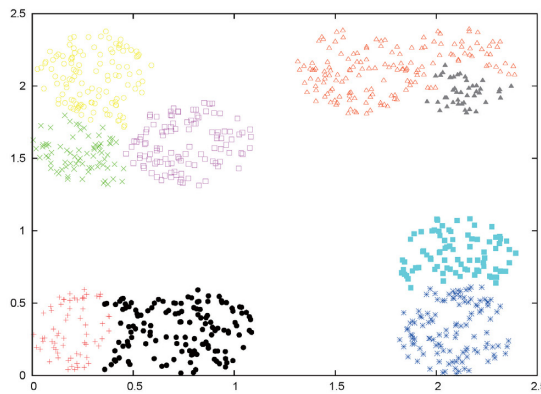


図17 Al. 5によるデータ7の初期クラスタ

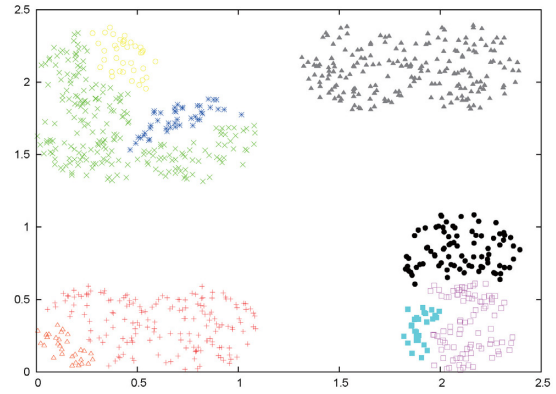


図18 Al. 2によるデータ7の初期クラスタ ($r = 4$)

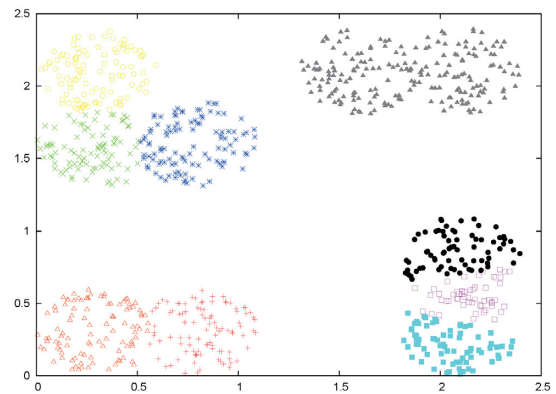


図19 Al. 4によるデータ7の結果 ($r = 4$)

Algorithm 2は、右上のクラスタを完全に分離することができていない。データ6とデータ7は、いくつかのクラスタが重なっており、かつ、その重なったクラスタ同士が分離している分布形状なので、初期クラスタの段階で重なっているクラスタが一つに結合されていると、図19のように、最終的に重み付きカーネルk-平均法で正解クラスタを得るのは難しいと考えられる。

次にデータ8とデータ9は、分散が異なるガウス分布であり、部分的にクラスタ同士が重なっている。データ8に関しては、提案手法の評価結果が低かった。KLダイバージェンスが最小となる初期クラスタを採用しているが、30回のうち、この初期クラスタが得られたのは2回だけであり、他の結果は全て正解クラスタに一致していた。また、2回の不正解のうち1回のKLダイバージェンスの値は0.122であり、正解となった28回の試行におけるKLダイバージェンスの値の中には、この値未満のものもいくつか見られた。逐次的にKLダイバージェンスを計算するため、KLダイバージェンスの計算結果が探索順に依存して不安定であり、今後の検討課題である。データ9に関して

は、提案手法の評価結果が僅かに高かった。

データ 10 とデータ 12 は非線形で初期クラスタへの依存度が強い分布形状であるが、クラスタ同士が完全に分離しているため、従来手法と提案手法共に良好な結果を得た。

データ 11 は、クラスタ同士が部分的に重なってしまっている土星型の形状であり、重み付きカーネル k -平均法やスペクトラルクラスタリングでの分類は難しいと考えられる(柳, 2016)。従来手法の NMI 値が僅かに大きかったが、決して良好な結果とはいえない。

データ 13~15 は、5つのクラスタからなる同じ形状のデータで、内側のクラスタは3つのデータ共に同じ分布となっており、凸状の3つのクラスタのうち2つが重なっている。外側の環状の2つのクラスタはデータの密度、または、データ数が異なっており、データ 13 は一定の間隔で分布しているのに対して、データ 14 はランダムに分布している。また、データ 15 は内側の凸状の3つのクラスタのデータ数が各 100 であるのに対して、内側の環状のクラスタのデータ数が 200、外側の環状のクラスタのデータ数が 300 となっている。クラスタが部分的に重なっており、かつ、非線形で分離しているデータ構造は従来手法が苦手とする形状である(柳, 2016)。データ 13 と 14 に関しては提案手法の評価結果が高かった。データ 13 に関して、従来手法の各 r における Algorithm 2 が出力した初期クラスタの NMI(AI.2) をみると、 $r=2$ のときが最も大きい、重み付きカーネル k -平均法の初期クラスタとして妥当なのは $r=3$ の場合である。 $r=2$ のときの、従来手法における Algorithm 2 の出力するクラスタを図 20 に示す。図 20 より $r=2$ のとき

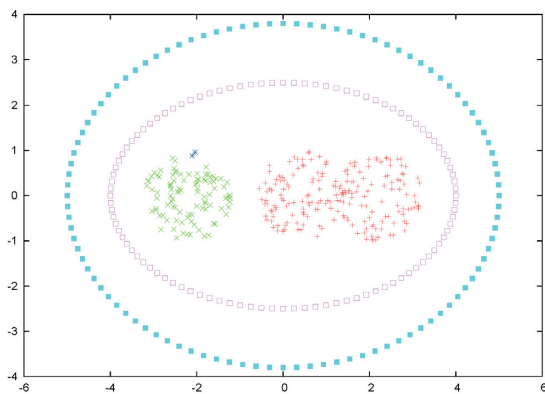


図 20 AI.2 によるデータ 13 の初期クラスタ ($r=2$)

は、内側の重なっているクラスタが結合してし

まっているため、図 20 の初期クラスタから重み付きカーネル k -平均法により良好な分類結果を得ることは、図 21 のように困難と考えられる。また、 $r=3$ のときの従来手法における

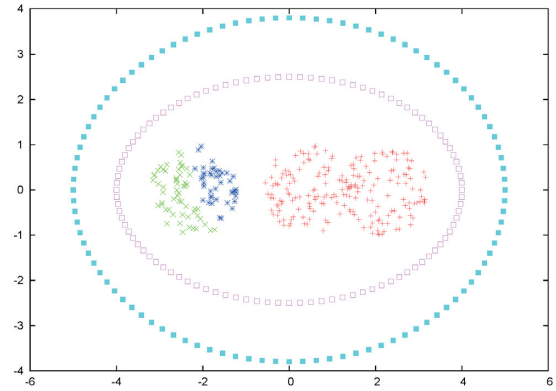


図 21 AI.4 によるデータ 13 の結果 ($r=2$)

Algorithm 2 の出力するクラスタを図 22 に示す。図 22 より、 $r=3$ のときは内側の重なっ

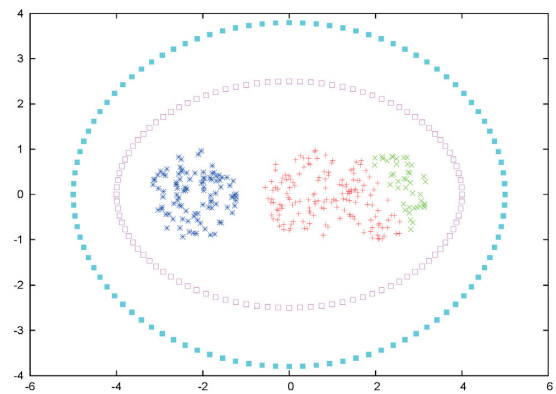


図 22 AI.2 によるデータ 13 の初期クラスタ ($r=3$)

ているクラスタが2つのクラスタとして分類されているため、図 23 のように最終的な分類結果では $r=2$ の場合よりよい。また、 $r=4$

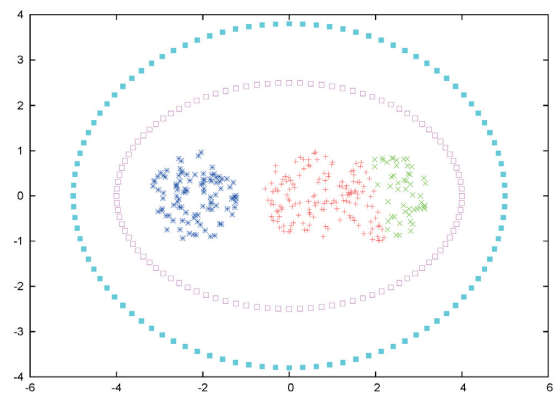


図 23 AI.4 によるデータ 13 の結果 ($r=3$)

のときの Algorithm 2 が出力する初期クラスタ

タは、図 24 のように外側の 2 つのクラスタと内側の一部のデータが結合してしまっているため、NMI 値が小さくなっている。外側の 2

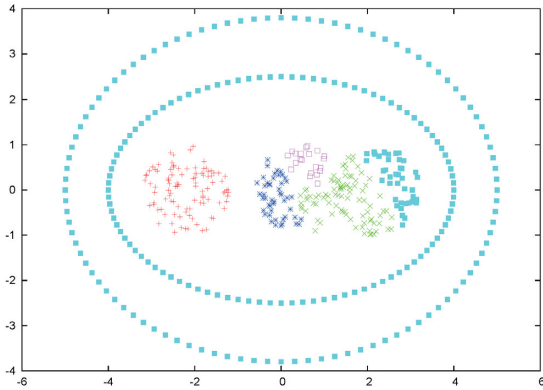


図 24 Al. 2 によるデータ 13 の初期クラスタ ($r = 4$)

つのクラスタのデータ間隔は一定であるため、繰り返しの 1 回目で各クラスタ内のデータの半分が結合し、2 回目で外側の 2 つの環状クラスタが完成してしまっていた。そのため、 $3 < r$ となる 3 回目で本来分離すべきクラスタが結合してしまっていた。図 24 のように外側の環状の 2 つのクラスタが結合してしまっている初期クラスタから、重み付きカーネル k-平均法により良好な分類結果を得ることは、図 25 のように困難と考えられる。一方で、提案手法

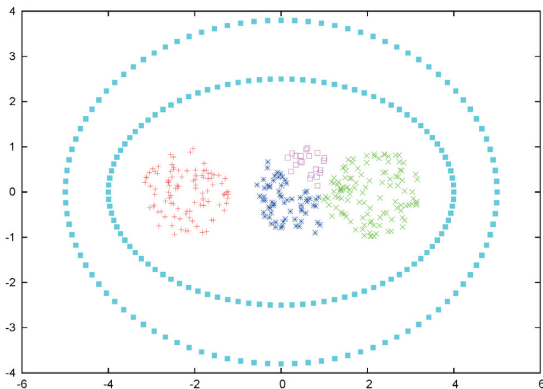


図 25 Al.4 によるデータ 13 の結果 ($r = 4$)

の Algorithm 5 では、やはり、外側の 2 つの環状クラスタの結合速度は速いものの、 $r = 3$ のときの Algorithm 2 が出力した図 22 とほぼ同じ初期クラスタが得られ、最終的なクラスタリング結果も良好であった。KL ダイバージェンス基準がうまく作用したと考えられる。データ 14 では、従来手法の Algorithm 2 が出力する初期クラスタの NMI 値は $r = 2, 3, 4$ で全て 0.866 と等しく、提案手法の NMI 値 0.882 と

大差はないにも関わらず、最終的な結果では $r = 4$ の 0.766 が従来手法の最大値であり、提案手法の 0.962 と差がついている。Algorithm 2 の $r = 2, 3, 4$ における初期クラスタを図 26 に示し、Algorithm 5 の初期クラスタを図 27 に示す。データ 14 は、外側の 2 つの環状クラ

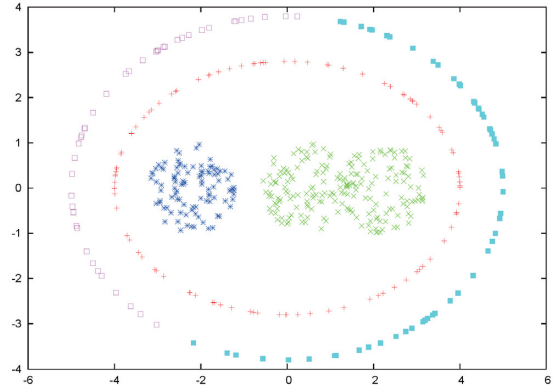


図 26 Al. 2 によるデータ 14 の初期クラスタ

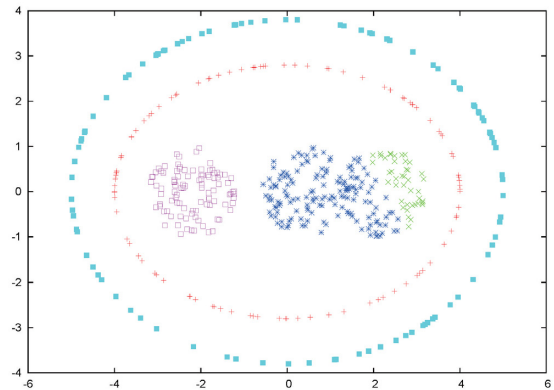


図 27 Al. 5 によるデータ 14 の初期クラスタ

スタのデータをランダムに発生させているため、データ間の距離にばらつきがあり、かつ、内側の凸状のクラスタと比較して密度が小さい。そのため、Algorithm 2 では内側のクラスタの結合と外側のクラスタの結合に関して、パラメータ r だけではうまく調整できなかったと考えられる。一方で、Algorithm 5 では、クラスタの生起確率がデータ間の距離になるべく依存しないよう設計しているため、5 つのクラスタの結合速度をうまく調整できたと考えられる。データ 15 は、従来手法の NMI 値は Algorithm 2 と Algorithm 4 共に $r = 2, 3, 4$ で全て 0.898 と等しく、提案手法の NMI 値は Algorithm 5 と Algorithm 6 共に 0.861 であった。実際のクラスタリング結果は図 28, 図 29 に示すように、共に適切な分類ができていな

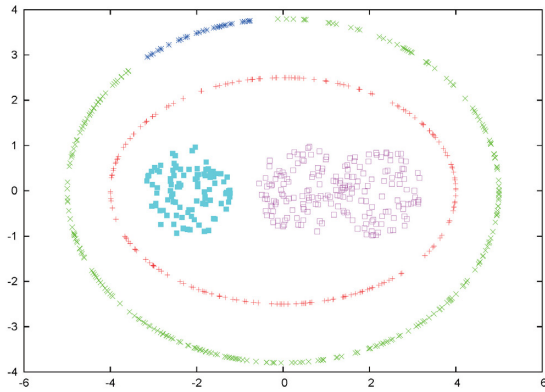


図 28 Al. 2 によるデータ 15 の初期クラスタ

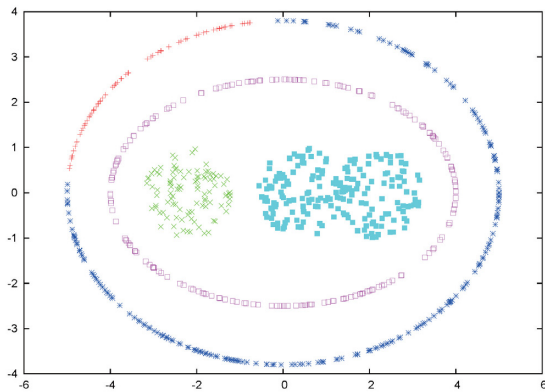


図 29 Al. 5 によるデータ 15 の初期クラスタ

い。内側と外側のクラスタでデータ数が偏っているため、従来手法と提案手法共に、結合速度の調整ができなかったと考えられる。データ 13 やデータ 14 のような複雑な分布形状のクラスタからなるデータに対しては、従来手法のパラメータ r の設定は難しく、一方で提案手法の KL ダイバージェンス基準による結合は良好に動作する可能性が高いと考えられる。しかし、データ 15 のように、複雑な分布形状に加えてクラスタのデータ数が大きく偏っている場合は提案手法も不安定となった。クラスタの分布 P の定義が、データ 15 の形状をうまく表現できていないことが理由と考えられる。

実データに関しては、ionosphere 以外では提案手法の評価が低く、ionosphere の結果も決して良好ではない。ただし、従来手法のパラメータ r に関して個別に比較すると、特定の r に対する評価が高いのではなく、データに依存している。iris では $r = 4$, wine では $r = 2, 3$, glass では $r = 2, 3, 4$, soybeans では $r = 2, 3, 4$, sat image では $r = 4$, そして、pen digit では $r = 4$ のとき従来手法の評価が高かった。密度が小さいデータに関しては r を小さくし、密度が大きいデータに関しては r

を大きくすることで、従来手法の分類性能をある程度向上させることができ、一般的にはデータ数に比して次元数が大きくなれば、密度は小さくなる傾向にある。次元数が大きい soybeans, satimage, pen digit では r の増加に対して NMI 値も増加しているため、その傾向が見られるが、次元数が小さい iris でも r の増加に対して NMI 値が増加しているため、次元数とデータ数に対する r との決定的な相関は確認できず、また、データ 13, 14, 15 のように、様々な分布形状が混合する場合もあるため、 r の調整は難しい。一方で、クラスタ分布の適切な設計が必要ではあるものの、パラメータを必要とせず、初期クラスタの採用基準として目的関数を保持する提案手法は、従来手法に対して、ある程度の優位性を持つと考える。

4-5 計算時間の比較

表 4 に各データに対する提案手法と従来手法の計算時間の平均、および、各 Algorithm の繰り返し回数の平均を示す。表 4 より、全体的に提案手法の Algorithm 5 の繰り返し回数が、従来手法の Algorithm 2 より大きく上回っている。各データに対する Algorithm 5 の繰り返し回数の平均とデータ数との比を図 30 に示す。図 30 の横軸はデータ名であり、縦軸は比率である。図 30 より各データにおいて、デー

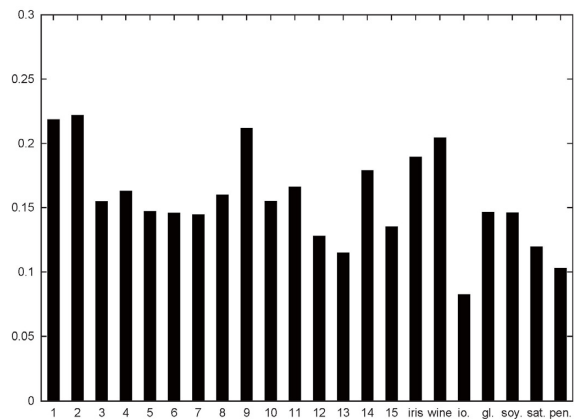


図 30 各データの Al. 4 の平均回数とデータ数の比

タ数の 1~2 割りの繰り返し回数を必要としており、式 (30) による非効率な探索が多かったと考えられる。一方で計算時間に関しては、Algorithm 5 の影響は、それほど顕著に表れてはいない。表 4 の計算時間の平均のグラフを図 31 に示す。図 31 の横軸はデータ名であり、縦軸は計算時間の平均である。図 31 より、提

表4 各データに対する計算時間(秒)と繰り返し回数の比較

データ	提案手法				従来手法											
	計算時間				$r = 2$			$r = 3$			$r = 4$					
		Al.1	Al.5	Al.3	計算時間	Al.1	Al.2	Al.3	計算時間	Al.1	Al.2	Al.3	計算時間	Al.1	Al.2	Al.3
1	0.003	1	27.333	1	0.002	1	5	1	0.002	1	5	1	0.013	4	3	37.6
2	0.007	1	44.367	1	0.005	1	6	1	0.006	1	6	1	0.006	1	5	1
3	0.062	1	92.833	1	0.045	1	6	1	0.045	1	6	1	0.046	1	6	1
4	0.093	3.667	65.233	22.733	0.061	7	6	14	0.07	7	6	17	0.084	6	6	21
5	0.902	6.533	88.267	102.567	0.431	8	6	39	1.005	3	6	112	1.006	3	6	112
6	5.169	9	131.267	271.5	1.349	12	7	36	4.482	11	7	206	6.79	8	7	337
7	15.686	6.733	130.133	815.333	7.003	8	7	341	4.425	11	7	199	12.832	18	7	663
8	0.022	1.133	48	3.433	0.019	1	6	3	0.019	1	6	3	0.019	1	5	3
9	0.33	2.7	84.8	98	0.118	2	6	27	0.323	2	6	98	0.323	2	6	98
10	0.227	1	124.133	6	0.152	1	7	6	0.153	1	6	6	0.152	1	7	6
11	0.023	1	66.5	1	0.075	6	6	19	0.066	6	6	16	0.016	1	6	1
12	0.08	1	76.833	3	0.079	1	7	3	0.08	1	7	3	0.079	1	7	3
13	0.452	8.5	57.533	84.433	0.098	11	6	10	0.522	5	6	104	0.209	12	4	37
14	0.238	8.167	89.567	34.667	0.125	8	7	10	0.124	8	7	10	0.122	14	6	10
15	0.48	6.1	108.233	20.4	0.518	1	7	30	0.519	1	7	30	0.549	1	7	30
iris	0.006	5.267	28.433	3	0.006	7	5	3	0.006	7	5	3	0.005	4.733	3	3
wine	0.063	5	36.4	93	0.008	2	5	3	0.008	2	5	3	0.09	5	3	141
iono.	0.04	1.967	29	6.1	0.027	1	5	1	0.026	1	5	1	0.026	2	3	1
glass	0.58	12.567	31.4	674.133	0.043	2	5	16	0.03	3	4	15	0.029	3	4	15
soy.	0.002	1	6.867	6	0.002	1	3.233	6	0.002	1	2.267	6	0.002	1	2.5	6
sat.	0.665	9.233	57.433	125.433	0.2	1	5	15	0.722	15	6	137	0.547	16	5	99
pen.	1.589	2.767	82.433	47.1	1.607	1	6	45	1.72	1	6	45	1.59	2	5	45

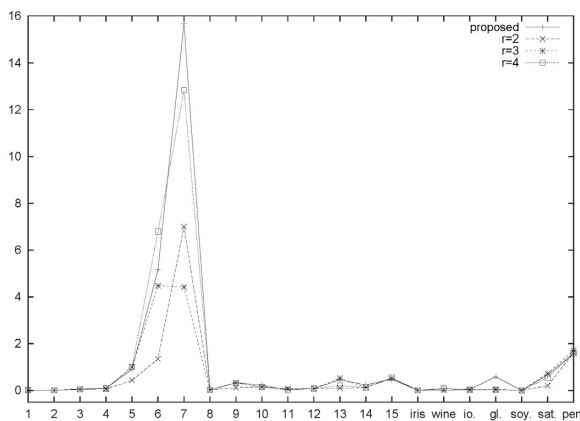


図31 各手法における各データの計算時間(秒)の平均

提案手法の計算時間の平均が小さいデータも見られる。データ6とデータ7における提案手法と従来手法の $r = 3$ の場合の計算時間の平均が突出して大きくなっているが、図30の比率がある程度一定で他のデータの計算時間は大きくないことから、Algorithm 5の影響ではなく、むしろAlgorithm 3の計算回数の影響が大きいと考えられる。実際に、表4より、提案手法のAlgorithm 3における繰り返し回数の平均は815.333、従来手法の $r = 4$ のAlgorithm

3における繰り返し回数の平均は663とかなり大きい。提案手法と従来手法の計算量に支配的なのは、むしろAlgorithm 1とAlgorithm 3であり、計算量の意味でも提案手法の実用性は従来手法に比較して劣っていないと考える。

5. むすび

本論文では、文献(柳, 2016)の手法を基に、データ集合間の距離に基づくクラスタの分布を用いた重み付きカーネルk-平均法の初期クラスタを探索するアルゴリズムを提案した。いくつかの人工データと実データを用いて、従来手法との比較実験を行い、クラスタ内のデータ数の偏りが小さいデータに関しては提案手法が有効であることを示した。一方で、クラスタ内のデータ数に大きな偏りがあるデータに関しては有効性を確認できなかった。今後の課題として、上記の分布形状を持つデータに対するクラスタの生起確率の検討がある。

参考文献

- [1] Chan, Pak K, Martine DF Schlag, and Jason Y Zien (1994) "Spectral k-way ratio-cut partitioning and clustering," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, Vol. 13, No. 9, pp. 1088–1096.
- [2] Dhillon, Inderjit S, Yuqiang Guan, and Jacob Kogan (2002) "Iterative clustering of high dimensional text data augmented by local search," in *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pp. 131–138, IEEE.
- [3] Dhillon, I.S., Y. Guan, and B. Kulis (2004) "Kernel k-means: spectral clustering and normalized cuts," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 551–556, ACM.
- [4] Dhillon, I.S., Y. Guan, and B. Kulis (2007) "Weighted graph cuts without eigenvectors a multilevel approach," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 29, No. 11, pp. 1944–1957.
- [5] Frank, A. and A. Asuncion (2010) "UCI Machine Learning Repository," <http://archive.ics.uci.edu/ml>.
- [6] Hubert, L. and P. Arabie (1985) "Comparing partitions," *Journal of classification*, Vol. 2, No. 1, pp. 193–218.
- [7] Karypis, George and Vipin Kumar (1998) "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on scientific Computing*, Vol. 20, No. 1, pp. 359–392.
- [8] MacQueen, J. (1967) "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297.
- [9] Máté, L. (1989) *Hilbert space methods in science and engineering*: Hilger.
- [10] Muller, K.R., S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf (2001) "An introduction to kernel-based learning algorithms," *Neural Networks, IEEE Transactions on*, Vol. 12, No. 2, pp. 181–201.
- [11] Ng, A.Y., M.I. Jordan, and Y. Weiss (2002) "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, Vol. 2, pp. 849–856.
- [12] Shi, J. and J. Malik (2000) "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 22, No. 8, pp. 888–905.
- [13] Strehl, A. and J. Ghosh (2003) "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *The Journal of Machine Learning Research*, Vol. 3, pp. 583–617.
- [14] Yu, S., L.C. Tranchevent, B. De Moor, and Y. Moreau (2011) "Optimized data fusion for kernel k-means clustering," *Kernel-based Data Fusion for Machine Learning*, pp. 89–107.
- [15] Yu, Stella X and Jianbo Shi (2003) "Multiclass spectral clustering," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp. 313–319, IEEE.
- [16] Zelnik-Manor, Lihi and Pietro Perona (2004) "Self-tuning spectral clustering," in *Advances in neural information processing systems*, pp. 1601–1608.
- [17] 柳信一 (2016) 「集合間の距離に基づく重み付きカーネルk-平均法のパラメータ探索法」, 『北海道情報大学紀要』, 第27巻, 第2号, pp.51–71, 3月。