2H3-4

k 近傍の最大距離に基づくノイズにロバストな引力と斥力を考慮したコードベクトルによるクラスタリング手法

A Clustering Method by Code Vectors Considering Attractive and Repulsive Based on Maximum Distance of k Neighbors

今村 弘樹[*1] 藤村 誠[*1] 黒田 英夫[*2]
Hiroki Imamura Makoto Fujimura Hideo Kuroda

[*1]長崎大学工学部情報システム工学科
Nagasaki University, Department of Computer and Information Sciences

[*2]長崎大学生産科学研究科
Nagasaki University, Graduate School of Science and Technology

When noise data is included, the clustering method based on code vectors considering attractive and repulsive force can not precisely classify data. In this paper, we propose the clustering method based on code vectors considering attractive and repulsive force which can precisely classify data even when noise data is included.

1. はじめに

筆者らは，引力と斥力を考慮したコードベクトルに基づくクラスタリング手法を提案した [1]．ただし，この手法は，ノイズとなるデータに対するロバスト性は考慮されていないため，ノイズとなるデータが存在する場合，クラスタリング精度が著しく低下する．

ここでは，ノイズとなるデータに対してロバストな引力と斥力を考慮したコードベクトルに基づくクラスタリング手法として，k 近傍の最大距離に基づくノイズにロバストな引力と斥力を考慮したコードベクトルに基づくクラスタリング手法を提案する．この手法は，クラスタリングする各データの k 近傍の最大距離の平均と分散に基づく閾値により，ノイズとなるデータを選定し，コードベクトルをフィッティングする際に，それらのデータからの影響を除外する．これにより，ノイズに対するロバスト性が向上することが期待できる．

2. 提案手法のアルゴリズム

ここでは，提案手法のアルゴリズムを示す．図 1 に提案手法の処理の流れを示す．なお，提案手法のアルゴリズムは，ノイズとなるデータを除外するための閾値の決定とコードベクトルのフィッティング処理以外は，従来手法 [1] におけるアルゴリズムと同じであるので，ノイズとなるデータを除外するための閾値の決定，コードベクトルの生成とフィッティング処理の個所のみ，以下に示すこととする．

2.1 ノイズとなるデータを除外するための閾値の決定

まず，ノイズとなるデータを除外するための閾値の決定のアルゴリズムを以下に示す．ただし，クラスタリングするデータは d 次元のベクトル $\boldsymbol{x}$ とし，データ数は M 個とする．また，閾値 Th_{dist} を決定する際に考慮する k 近傍のデータ数を $K\_NEIGHBOR$ とする．

1. h=1 とする．
2. k=1 とする．
3. $$\boldsymbol{x}_h^{(k)} = \arg \min_{1 \le i \le M, i \ne h} ||\boldsymbol{x}_h - \boldsymbol{x}_i||$$
   を満たすデータ $\boldsymbol{x}_h^{(k)}$ を抽出する．ただし，$\boldsymbol{x}_h^{(k)}$ は，h 番目のデータ $\boldsymbol{x}_h$ に対して k 番目に近いデータを表す．
4. もし，k が $K\_NEIGHBOR$ ならば 5. へ，そうでなければ，$k = k + 1$ として，3. へ．
5. $$k\_max_h = \arg \max_{2 \le k \le K\_NEIGHBOR} ||\boldsymbol{x}_h^{(k-1)} - \boldsymbol{x}_h^{(k)}||$$
   を抽出する．
6. もし，h が M なら 7. へ，そうでなければ，$h = h + 1$ として，2. へ．
7. $k\_max_h (1 \le h \le M)$ の平均 E_k と分散 V_k を算出する．
8. $Th_{dist} = E_k + \lambda V_k$ とする．ただし，λ は重み係数とする．

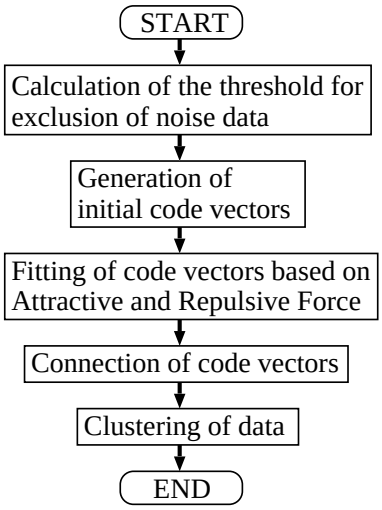

図 1: 提案手法の処理の流れ図.

連絡先: 長崎大学工学部情報システム工学科，〒 852-8521 長崎市文教町 1 － 14，Tel:095-819-2586

9. アルゴリズム終了.

$f_h(1\leq)h\leq M)$ をノイズデータのフラグとし，全てのフラグを 0 で初期化しておく．ここで，k_max_h が Th_{dist} より大きい値となったデータをノイズのデータとし，そのデータに相当する f_h の値を 1 とする．

2.2 コードベクトルの生成とフィッティング処理

次に，コードベクトルの生成とフィッティング処理のアルゴリズムを以下に示す.

2.2.1 コードベクトルの生成

m_j を $j(1\leq j\leq N)$ 番目における k 次元のコードベクトルとする．まず，データを含むベクトル空間において N 個のコードベクトルをランダムに生成する.

2.2.2 コードベクトルのフィッティング

まず，t を繰り返し計算回数とし，t における m_j に対する引力を

$$\mathrm{a}_j^{(t)} = \sum_{h=1,f_i=0}^{M} \frac{\alpha}{Exp_a} \frac{(\mathrm{p}_h - \mathrm{m}_j^{(t)})}{||\mathrm{p}_h - \mathrm{m}_j^{(t)}||}, \quad (1)$$

と定義する．ただし，Exp_a は，$\exp\{\gamma(\mathrm{p}_h-\mathrm{m}_j^{(t)})\cdot(\mathrm{p}_h-\mathrm{m}_j^{(t)})^T\}$ とし，α は，パラメータ係数，γ は γ=$0.7/(1+(t/7))$ とし，$\mathrm{m}_j^{(t)}$ は，t における m_j を表す．$f_i=0$ となるデータのみを用いて，この引力を考慮することにより，コードベクトルがノイズデータ以外のデータの方向へ移動するように促す．また，t における m_j に対する斥力を

$$\mathrm{r}_j^{(t)} = \sum_{i=1,i\neq j}^{M} \frac{\beta}{Exp_r} \frac{(\mathrm{m}_i^{(t)} - \mathrm{m}_j^{(t)})}{||\mathrm{m}_i^{(t)} - \mathrm{m}_j^{(t)}||}, \quad (2)$$

と定義する．ただし，Exp_r は，$\exp\{\gamma(\mathrm{m}_i^{(t)} - \mathrm{m}_j^{(t)})\cdot(\mathrm{m}_i^{(t)} - \mathrm{m}_j^{(t)})^T\}$) とし，$\beta$ は，パラメータ係数を表す．この斥力を考慮することにより，コードベクトルがデータに対して，分散してフィッティングするように促す．これらコードベクトルの引力と斥力に基づいて，以下の繰り返し計算により，コードベクトルをデータにフィッティングさせていく．まず，m_j における引力と斥力の合力を

$$\mathrm{F}_j^{(t)} = \mathrm{a}_j^{(t)} - \mathrm{r}_j^{(t)}. \quad (3)$$

と定義し，繰り返し回数に従って，m_j を以下のように更新していく.

$$\mathrm{m}_j^{(t+1)} = \mathrm{m}_j^{(t)} + \gamma \mathrm{F}_j^{(t)}, \quad (4)$$

コードベクトルが十分にデータにフィッティングした時点で，上記の繰り返し計算を終了する.

3. 実験

提案手法の有効性を評価するために，コンピュータにより生成した人工データに対して，クラスタリングを行った．ここでは，文献 [1] の手法を従来手法とし，提案手法との比較を行った．また，ここで用いる人工データは，文献 [1] で用いられているデータと同等のものにノイズを付加したものとした．なお，従来手法のパラメータは，コードベクトル数は 40，α=2.0,β=2.0，γ=1.0,Th_θ=0.8 とし，提案手法のパラメータは，コードベクトル数は 40，α=2.0,β=2.0,γ=1.0,λ=0.6,Th_θ=0.8,$k\_NEIGHBOR$=2 とした.

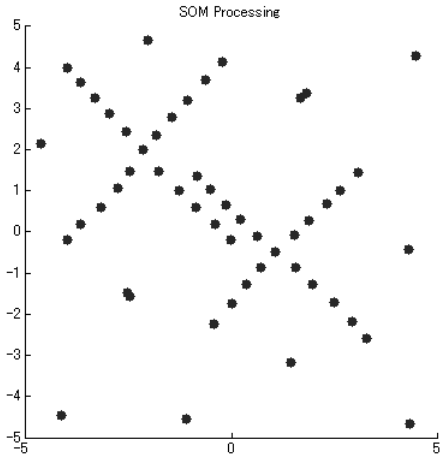

図 2: 生成した人工データ#1.

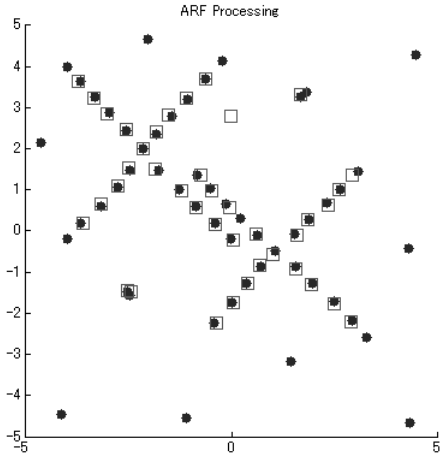

図 3: 繰り返し計算終了後におけるコードベクトルの状態 (従来手法).

まず，図 2 に示すデータに対して，実験を行った．図 3 は，従来手法における繰り返し計算終了後におけるコードベクトルの状態を示している．図 5 は，従来手法におけるクラスタリング結果を示している．なお，クラスタリング結果では，同じクラスタには同じ記号を，異なるクラスタには異なる記号で表している．また，図 6 は，提案手法においてノイズデータとして処理されたデータを $\times$ 印で表している．図 8 は，提案手法における繰り返し計算終了後におけるコードベクトルの状態を示している．図 9 は，提案手法におけるクラスタリング結果を示している．従来手法では，ノイズデータの影響により，コードベクトルがノイズデータ以外のデータに対し良好にフィットできなかった．これにより，良好にクラスタリングすることできなかった．これに対し，提案手法は，ノイズデータの影響を除外できたことにより，コードベクトルがノイズデータ以外のデータに対し良好にフィットできていることが分かる.

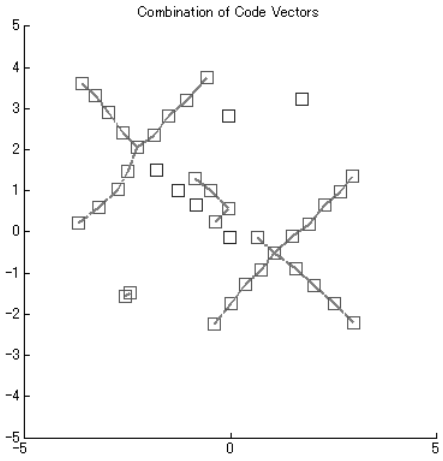

図 4: コードベクトルの連結状態 (従来手法).

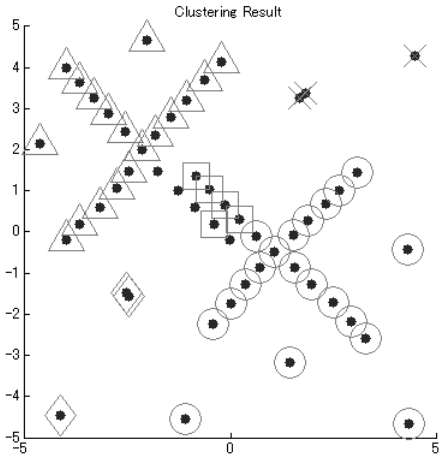

図 5: クラスタリング結果 (従来手法).

4. おわりに

ノイズとなるデータに対してロバストな引力と斥力を考慮したコードベクトルに基づくクラスタリング手法として，k 近傍の最大距離に基づくノイズにロバストな引力と斥力を考慮したコードベクトルに基づくクラスタリング手法を提案した．

提案手法の有効性を評価するために，コンピュータにより生成した人工データに対して，従来手法と提案手法を用いてクラスタリングを行った．実験の結果，提案手法は，従来手法に比べ，良好にクラスタリングできたと考える．

今後は，従来手法と提案手法を実データのクラスタリングに適用することにより，提案手法の有効性を評価する予定である．

参考文献

[1] 今村 弘樹，藤村 誠，黒田 英夫，“ 引力と斥力を考慮したコードベクトルに基づくクラスタリング手法，”信学技法，NC2007-101，Vol.107, No.413, pp.85-89, 2007.

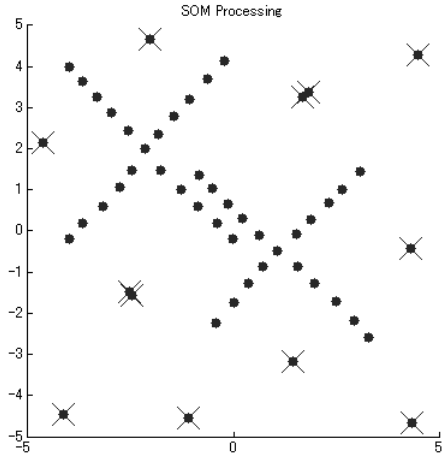

図 6: ノイズデータとして処理されたデータ.

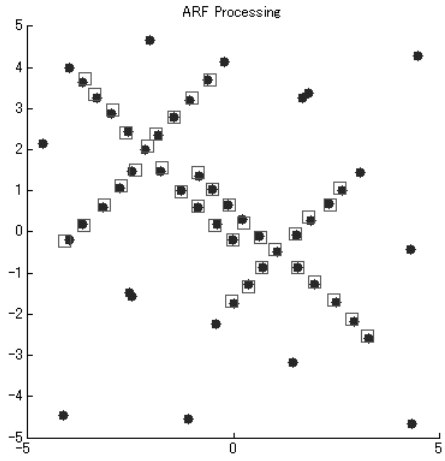

図 7: 繰り返し計算終了後におけるコードベクトルの状態 (提案手法).

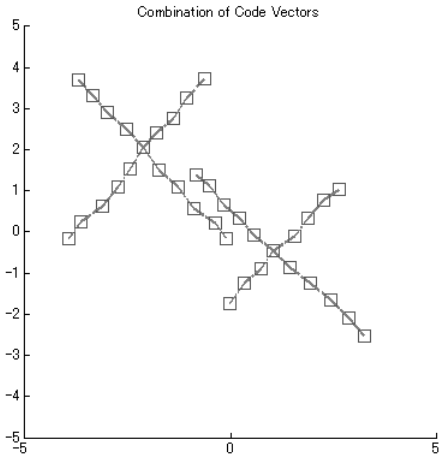

図 8: コードベクトルの連結状態 (提案手法).

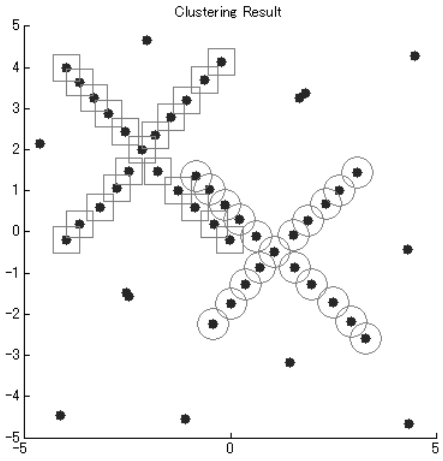

図 9: クラスタリング結果 (提案手法).