A New Kernel-Based Fuzzy Clustering Approach: Support Vector Clustering With Cell Growing

Bharat Yadav
MA15w1-M

Hochschule Mittweida

04/01/2017

Outline

Introduction

Clustering?

Grouping a set of objects similar to each other.

Fuzzy Clustering?

- Each data point has a membership grade indicating its belongingness degree to each cluster, rather than assigning the data point to only one of the clusters.
- The fuzzy formalism can provide a framework to model cluster prototypes and to detect noise patterns for further classification.

Introduction

Kernel-based Clustering

- A nonlinear mapping between the input space and feature space through a kernel function.
- Computing the sphere with minimum radius.

SV Clustering Algorithm

- The support vectors are used to define a sphere.
- This sphere corresponds to a set of contours which enclose all the points in original input space.

Introduction

Problem with SV Clustering

The clustering results provide no information about either prototypes (i.e., cluster centers) or actual grade of memberships in partition.

Solution

"One cluster as one sphere" in the feature space so that it becomes a convertible kernel-based mapping between original space and feature space.

SV Clustering

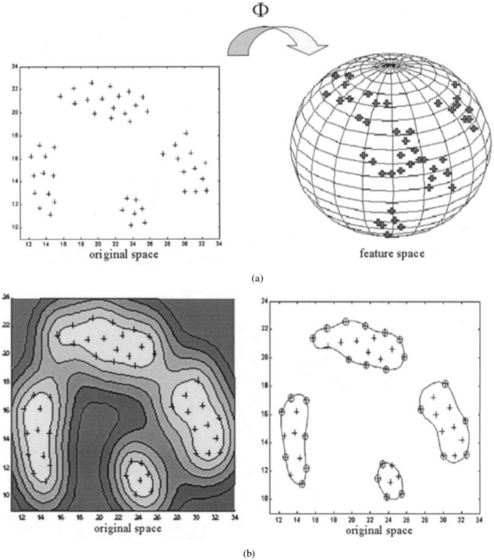

Fig. 1. Schematic diagram of the SV clustering

SV Clustering

- Φ denotes a nonlinear transformation.
- Clustering may be viewed as finding a smallest sphere that encloses all the data points in the feature space, as shown in Fig. 1(a).
- The contour diagram shown in Fig. 1(b) can be obtained by estimating the distance between the spherical center and corresponding input point.

SV Clustering

Mathematical Formulation

Given a set of input patterns, $\{\mathbf{x}_i\} \subseteq \mathcal{X}$

$$\min_{R,a,\xi_i} R^2 + C \sum_i \xi_i$$

$$\text{s.t.} \left\| \Phi(\mathbf{x}_i) - \mathbf{a} \right\|^2 \; ; \; \xi_i \geq 0 \; \forall \; i \tag{1}$$

where, R is radius
a is the center of the enclosing sphere
ξ_i is a slack variable and
C is a constant controlling the penalty of noise

SV Clustering

Mathematical Formulation

The Wolfe Dual optimization problem is

$$\max_{\beta_i} W = \sum_i \Phi(\mathbf{x}_i).\Phi(\mathbf{x}_i)\beta_i - \sum_{i,j} \beta_i\beta_j\Phi(\mathbf{x}_i).\Phi(\mathbf{x}_j) \tag{2}$$

subject to $\sum_i \beta_i = 1$ and $0 \leq \beta_i \leq C$. Where β_i are the Lagrange multipliers.

SV Clustering

Mathematical Formulation

The functional form of mapping $\Phi(x_i)$ does not need to be known since it is implicitly defined by the choice of kernel function,

$$K(\mathbf{x}_i, \mathbf{x}_j) \equiv \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

Throughout this paper, we use the Gaussian kernel,

$$K(\mathbf{x}_i, \mathbf{x}_j) = exp\Bigg(- q\Big\|\mathbf{x}_i - \mathbf{x}_j\Big\|^2 \Bigg) \tag{3}$$

The distance between input pattern and spherical center

$$\begin{aligned} R^2(x) &= \Big\|\Phi(\mathbf{x}) - \mathbf{a}\Big\|^2 \\ &= K(\mathbf{x}, \mathbf{x}) - 2\sum_i \beta_i K(\mathbf{x}_i, \mathbf{x}) + \sum_{i,j} \beta_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j) \end{aligned} \tag{4}$$

SV Clustering

Mathematical Formulation

- $\mathbf{x}_i$ is outside the sphere if $\xi_i > 0$
- From the Karush–Kuhn–Tucker (KKT) conditions we know that $\beta_i = C$. Thus, we refer to those points as outside vectors (OVs). Those points do not exist when $C \geq 1$.
- The point is located in the spherical surface when $0 < \beta_i < C$. We denote these points as border vectors (BVs).
- Thus, the set of support vectors consists of an OV set and a BV set.
- Similarly, the point $\mathbf{x}_i$ is located inside the sphere when $\beta_i = 0$.

Proposed Multisphere SV Clustering

Three major difficulties with SV clustering for real data

1. SV clustering algorithm identifies variously shaped clusters by finding connected components among data points. This is a relatively time-consuming and unreliable procedure.
2. The results of SV clustering do not provide cluster centers (or prototypes), which makes a geometrical picture of clusters impossible.
3. The SV clustering lacks the ability to represent the nature of the grade of membership in partition.

Proposed Multisphere SV Clustering

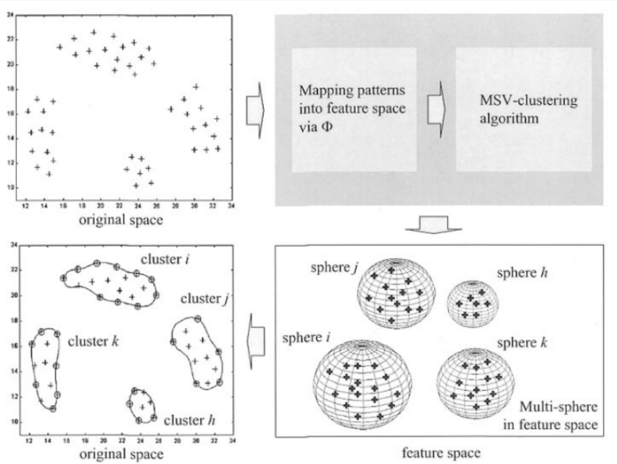

Fig. 2. Schematic diagram of the multisphere SV clustering.

Proposed Multisphere SV Clustering

Five processes for multisphere SV (MSV) clustering

1. **Competition process:** find a winning cluster to enclose new point.
2. **Validation process:** test the vigilance degree of the winning cluster.
3. **Reset process:** disable the winning cluster.
4. **Growing process:** create a new cluster to enclose new point.
5. **Learning process:** update cluster parameters.

MSV Clustering Algorithm

Variables and Parameters

P_k	partition of data points
S_k	support vectors set in sphere k for cluster P_k, where $S_k \subseteq P_k$
nc:	numbers of clusters
$\mathbf{V}_k$	center of sphere k in feature space
R_k	radius of sphere k in feature space
$D_k(\mathbf{x})$	distance between $\mathbf{x}$ and $\mathbf{V}_k$ in feature space
$d(\mathbf{x}, P_k)$	distance between $\mathbf{x}$ and P_k in original space
$g_k(\mathbf{x})$	vigilance degree of $\mathbf{x}$ belonging to P_k
β_i	the Lagrange multiplier corresponding to data point $\mathbf{x}_i$
J	index for winning cluster
q, C	the cluster parameters
ε	vigilance threshold

MSV Clustering Algorithm

1. **Initialization.**

   Initial first cluster. Set $P_1 = S_1 = \{\mathbf{x}^{(1)}\}$, $\beta_1 = 1$, and $R_1 = 0$.

2. **Input Points Presentation**

   Perform the following computation for $\mathbf{x}^{(i)}$, $i = 2,\ldots,n$.

3. **Winning Cluster Competition.**

   Find winning cluster *J*, such that $d(\mathbf{x}^{(i)}, P_J) = \min_{k=1,\ldots,nc} \left\{ d(\mathbf{x}^{(i)}, P_k) \right\}$

4. **Validity Test.**

   IF $g_J(\mathbf{x}^{(i)}) < \varepsilon$,

   THEN $\mathbf{x}^{(i)}$ belongs to cluster *J*, goto Step 5.

   ELSE

   Disable cluster *J* by setting $d(\mathbf{x}^{(i)}, P_J) = \infty$

   IF there exists available cluster *k*,

   THEN return to Step 3.

   ELSE

   goto Step 6.

5. **Parameter Learning**

   Solve the MSV optimization problem in the winning cluster J, and update the cluster parameters.

   Return to Step 2 (First enable all disabled clusters.)

6. **Create a New Cluster**

   Create a new cluster $P_{nc} = S_{nc} = \{\mathbf{x}^{(i)}\}$, $R_{nc} = 0$,

   Set the corresponding Lagrange multiplier of $\mathbf{x}^{(i)}$ to 1.

   Return to Step 2 (First enable all disabled clusters.)

Fuzzy Memberships Computation and Heuristic Analysis

Computation of Cluster Memberships

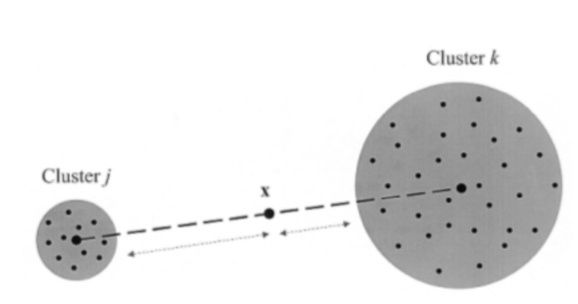

Fig. 3. Two different sizes of clusters in the feature space.

Fuzzy Memberships Computation and Heuristic Analysis

Computation of Cluster Memberships

$$\mu_j(\mathbf{x}) = \begin{cases} 0.5 \times \left(\dfrac{1 - \left(\frac{1}{R_j}\right) D_j(\mathbf{x})}{1 + \lambda_1 \left(\frac{1}{R_j}\right) D_j(\mathbf{x})} \right) + 0.5, \\ \qquad \text{if } D_j(\mathbf{x}) \leq R_j \\ 0.5 \times \left(\dfrac{1}{1 + \lambda_2 (D_j(\mathbf{x}) - R_j)} \right), \\ \qquad \text{otherwise} \end{cases}$$

- Where R_j is the spherical radius in feature space that corresponding to cluster j
- $D_j(x)$ is the distance between x and spherical center in feature space
- The parameters $\lambda_1 and \lambda_2$ satisfy the relation below which makes $\mu_j(x)$ differentiable as $D_j(x) = R_j$

$$\lambda_2 = \frac{1}{R_j(1 + \lambda_1)}$$

Fuzzy Memberships Computation and Heuristic Analysis

Estimation of Winning Cluster Validity

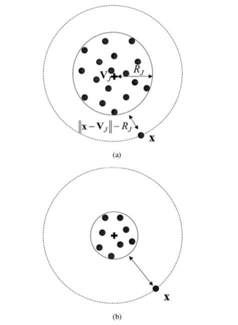

Fig. 4. Difference between x and spherical surface for two examples. (a) Large R_j. (b) Small R_j

Fuzzy Memberships Computation and Heuristic Analysis

Heuristic 1 Method to Reduce the Computation Time

If the input point x_i is located inside the cluster sphere, the algorithm does not need to update cluster parameters by setting the Lagrange multiplier $\beta_i = 0$. And then check whether the following condition satisfies:

$$D_j(x_i) \leq R_j$$

If yes, no need to update the cluster parameters. Otherwise, update cluster parameter using Heuristic 2.

Fuzzy Memberships Computation and Heuristic Analysis

Heuristic 2 Method to Reduce the Computation Time

By using only the support vectors, one would obtain exactly the same final results as if one had the full training set. Therefore, one can directly solve the reduced problem, saving significantly computation time.

Experimental Results

Unequal Populations Data Set

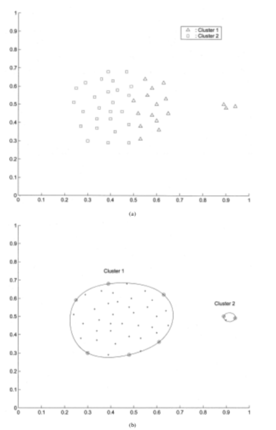

Fig. 5. Crisp partition results obtained from (a) classical FCM algorithm with c = 2 and m = 2 , and (b) MSV algorithm. The support vectors are marked with small circles.

Experimental Results

Random Noise Data Set

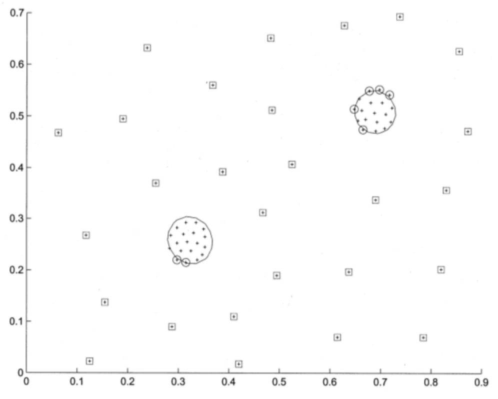

Fig. 6. Clustering results for “random noise data set” with $1/(CN) = 0.5$. The border vectors are marked with small circles. The dense regions are enclosed as clusters whereas the noise points are identified with square symbols.

Experimental Results

“XO” Data Set

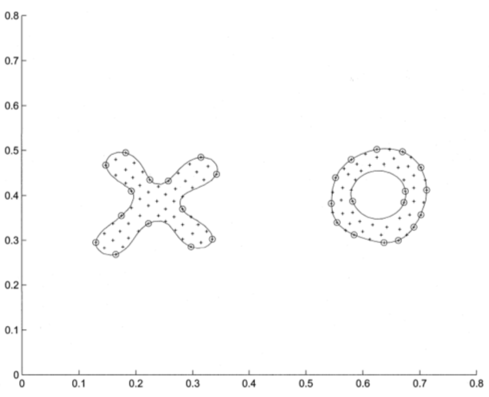

Fig. 7. Clustering results for the “XO” data set. The support vectors are marked with small circles.

Experimental Results

Handwritten Digits Data Set

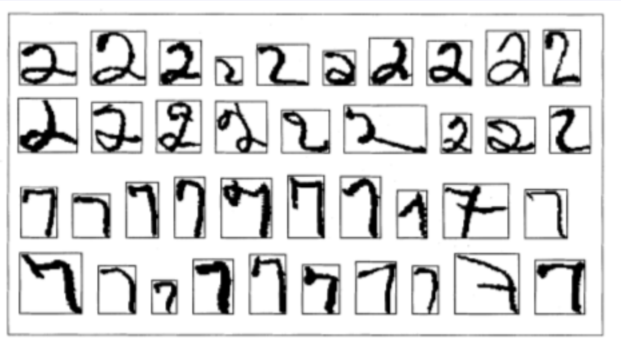

Fig. 8. Examples of some of the handwritten digits from classes “2” and “7”(The Confusion Group) in data set.Each digit has different size.

Experimental Results

Handwritten Digits Data Set

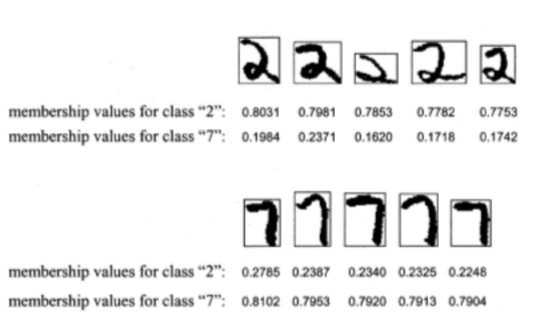

Fig. 9. Examples of typical prototype digits obtained from the MSV clustering algorithm for both classes and their corresponding fuzzy membership values.

Experimental Results

Handwritten Digits Data Set

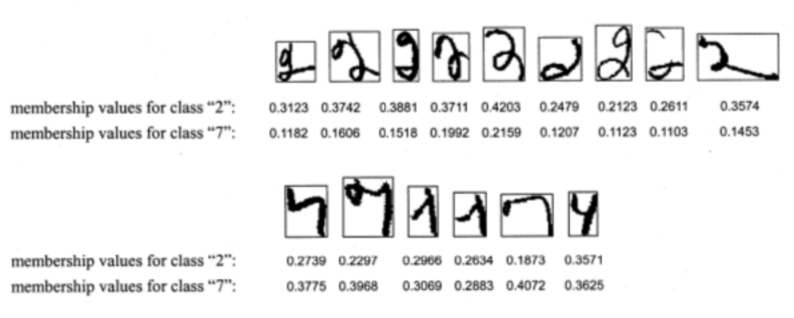

Fig. 10. Examples of outlier digits obtained from the MSV clustering algorithm for both classes and their corresponding fuzzy membership values.

We then decide "good patterns" or "bad patterns" according to the membership values higher or lower than 0.5.

References

1. A New Kernel-Based Fuzzy Clustering Approach: Support Vector Clustering With Cell Growing; Jung-Hsien Chiang, Member, IEEE, and Pei-Yi Hao

Thank you for your attention.