확률 분포의 2차 거리 측정을 이용한 클러스터링

이용진○ 최승진
포항공과 대학교 컴퓨터 공학과
{solarone○, seungjin}@postech.ac.kr

Clustering Using Quadratic Distance Measure Between Densities

Yongjin Lee○ Seungjin Choi
Dept. of Computer Science, Pohang University of Science and Technology

요 약

We derive a simple clustering algorithm which partitions the given data by minimizing overlap between clusters. For simple implementation and less complexity, Parzen window density estimation and quadratic distance measure between densities are adopted.

1. Introduction

Stephen et al. have suggested a clustering algorithm based on information theory and shown good clustering results by minimizing the entropy of cluster posterior [1, 2]. They partitioned the data by minimizing the overlap between clusters after density estimation using mixture of gaussian, which is required a lot of careful attention.

In this paper, we start from the idea suggested in [1], and derive a similar clustering algorithm but much simpler and easier to implement. We estimate the density of data using Parzen windows and calculate divergence by quadratic distance measure between probability density instead of using mixture of gaussian and Kullback-Liebler divergence. After reviewing some related methods in Sec. 2. and 3., we derive our algorithm in Sec 4 and 5. And some experiment results are given in Sect 6.

2. Minimum Entropy Data Partitioning

We begin by briefly reviewing the method of minimum entropy data partitioning[1,2] since this idea is a starting point for our method. In maximum certainty data partitioning, one constructs candidate partition models for data sets in such a way that overlap between partitions is minimal.

Let us consider a partitioning of the data into a set of K clusters. The probability density function of a single datum $\mathbf{x}$, conditioned on a set of K partitions, is given by

$$p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x}\,|\,i)p(i) \tag{1}$$

The overlap between the unconditional density $p(\mathbf{x})$ and the contribution to this density function of the ith partition, $p(\mathbf{x}\,|\,i)$, is measured by Kullback-Liebler(KL) divergence between these two distributions:

$$V_i = -KL\left[p(\mathbf{x}\,|\,i)\,\|\,p(\mathbf{x})\right] \tag{2}$$

which is upper-bounded by 0(Since KL divergence is always nonnegative). When the ith class is well-separated from all others, V_i, is minimized.

The total overlap over a set of K partitions, V, is defined by

$$\begin{aligned} V &\sqcup -\sum_{i=1}^{K} p(i)V_i \\ &= -\sum_{i=1}^{K} p(i)\int p(\mathbf{x}\,|\,i)\log\left(\frac{p(\mathbf{x}\,|\,i)}{p(\mathbf{x})}\right)d\mathbf{x} \end{aligned} \tag{3}$$

It follows from Bayes' theorem that Eq.(3) can be rewritten as

$$\begin{aligned} V &= -\int p(\mathbf{x})\left(\sum_{i=1}^{K} p(i\,|\,\mathbf{x})\log p(i\,|\,\mathbf{x})\right)d\mathbf{x} + \sum_{i=1}^{K} p(i)\log p(i) \\ &= \int p(\mathbf{x})H(i\,|\,\mathbf{x})d\mathbf{x} - H(i) \end{aligned} \tag{4}$$

The total overlap measure V consists of the expected (Shannon' s) entropy of the class posteriors and the negative entropy of the priors. Therefore minimizing V is equivalent to minimizing the expected entropy of the partitions given a set of observed variables [1,2].

Alternatively, we can rewrite the total overlap measure V in (3) as

$$V = \sum_{i=1}^{K} p(i) \int p(\mathbf{x}|i) \log p(\mathbf{x}|i) d\mathbf{x} + \int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x} = -\left[H(x) - \sum_{i=1}^{K} p(i) H(\mathbf{x}|i) \right] \quad (5)$$

Minimizing the total overlap measure is equivalent to minimizing the expected entropy of class-conditional density.

3. Quadratic Distance Measure Between Densities

Principe et all derive quadratic distance measures for probability density functions somewhat heuristically [3,4]. In case of density estimation using Parzen window it gives simple calculation for divergence between two densities. The difference of vectors inequality

$$(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}) \geq 0 \Leftrightarrow \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - 2\mathbf{x}^T\mathbf{y} \geq 0 \quad (6)$$

gives the expression

$$K_T(f,g) = \int f(\mathbf{x})^2 d\mathbf{x} + \int g(\mathbf{x})^2 d\mathbf{x} - 2\int f(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \quad (7)$$

It is easy to see that the measures are always positive, and when $f(\mathbf{x}) = g(\mathbf{x})$ the measure evaluate to zeros.

4. Clustering using Parzen window and Quadratic Distance

For a continuous random variable $\mathbf{x} \in \mathbb{R}^d$ whose realization is given by $\{\mathbf{x}_n\}_{n=1}^N$ where N is the number of data points, the probability density of $\mathbf{x}$ estimated by the Parzen window using a Gaussian Kernel is given by

$$p(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^{N} G(\mathbf{x}, \mathbf{x}_n, \sigma^2) \quad (8)$$

where

$$G(\mathbf{x}, \mathbf{x}_n, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_n\|^2}{2\sigma^2} \right).$$

And we can write i th class(denoted by C_i) conditional density as

$$p(\mathbf{x}|i) = \frac{1}{N_i} \sum_{n=1}^{N_i} G(\mathbf{x}, \mathbf{x}_n^i, \sigma^2) \quad (9)$$

where $\mathbf{x}_n^i \in C_i$ and N_i is the number of data points belonging to C_i. We can rewrite Eq. (10) using indicator variables $\{z_{in}\}$ as

$$p(\mathbf{x}|i) = \frac{1}{N_i} \sum_{n=1}^{N} z_{in} G(\mathbf{x}, \mathbf{x}_n, \sigma^2) \quad (10)$$

where

$$z_{in} = \begin{cases} 1 & if \quad \mathbf{x}_n \in C_i \\ 0 & otherwise \end{cases}$$

and $N_i = \sum_{n=1}^{N} z_{in}$.

Institutively, the indicator variable can be considered as posterior, i.e. $z_{in} = p(i|\mathbf{x}_n)$.

Incorporating the density estimated by Parzen window into quadratic distance measure, Eq. (7), leads to

$$\begin{aligned} V_i &= -K_T[p(\mathbf{x}|i) \| p(\mathbf{x})] \\ &= -\int p(\mathbf{x}|i)^2 d\mathbf{x} - \int p(\mathbf{x})^2 d\mathbf{x} + 2\int p(\mathbf{x}|i) p(\mathbf{x}) d\mathbf{x} \\ &= -\frac{1}{N_i^2} \mathbf{z}_i^T \mathbf{G} \mathbf{z}_i - \frac{1}{N^2} \mathbf{1}^T \mathbf{G} \mathbf{1} + \frac{2}{N N_i} \mathbf{z}_i^T \mathbf{G} \mathbf{1} \end{aligned} \quad (11)$$

where $\mathbf{G} \in \mathbb{R}^{N \times N}$ by $[\mathbf{G}]_{nm} = G(x_n, x_m, 2\sigma^2)$ and $[\mathbf{z}_i]_n = z_{in}$.

In a similar way of the previous section, the total overlap can be written as

$$\begin{aligned} V &= \sum_{i=1}^{K} p(i) V_i \\ &= -\sum_{i=1}^{K} p(i) K_T[p(\mathbf{x}|i) \| p(\mathbf{x})] \\ &= \frac{1}{N} \left[\frac{\mathbf{1}^T \mathbf{G} \mathbf{1}}{N} - \sum_{i=1}^{K} \frac{\mathbf{z}_i^T \mathbf{G} \mathbf{z}_i}{N_i} \right] \end{aligned} \quad (12)$$

which is reminiscent of the Eq. (5). The second term of last equality in Eq. (12) can be considered within-cluster association. Therefore minimization of overlap between partitions is equivalent to maximization of within-cluster association

$$L = \sum_{i=1}^{K} \frac{1}{N_i} \sum_{n=1}^{N} \sum_{m=1}^{N} z_{in} z_{im} G(\mathbf{x}_n, \mathbf{x}_m, 2\sigma^2) \quad (13)$$

5. Maximization of Within-Cluster Association

We adjust the indicator variables, $\{z_{in}\}$, to maximize the within-cluster association (13). $\{z_{in}\}$ have to be bounded [0,1], so we parameterize the indicator variables using a generalized logistic function of the form

$$z_{in} = \frac{\exp[\theta_{in}]}{\sum_{c=1}^{K} \exp[\theta_{cn}]} \quad (14)$$

The gradient of (13) with respect to θ_{in} is given by

$$\frac{\partial L}{\partial \theta_{in}} = \sum_{j=1}^{K} \frac{\partial L_j}{\partial z_{jn}} \cdot \frac{\partial z_{jn}}{\partial \theta_{in}} \quad (15)$$

where, $L_j = \mathbf{z}_j^T \mathbf{G} \mathbf{z}_j / N_j$

The $\partial L_j / \partial z_{jn}$ and $\partial z_{jn} / \partial \theta_{in}$ are given by

$$\frac{\partial L_j}{\partial z_{jn}} = \frac{2}{N_j} \sum_{m=1}^{N} z_{jm} G(\mathbf{x}_n, \mathbf{x}_m, 2\sigma^2) - \frac{1}{N_j} L_j \quad (16)$$

$$\frac{\partial z_{jn}}{\partial \theta_{in}} = z_{jn}\delta_{i,j} - z_{in} z_{jn} \quad (17)$$

where $\delta_{i,j} = \begin{cases} 1 & if\ i = j \\ 0 & otherwise \end{cases}$.

Therefore the gradient is written as

$$\frac{\partial L}{\partial \theta_{in}} = \sum_{j=1}^{K} \frac{1}{N_j} \left(2 \sum_{m=1}^{N} z_{jm} G(\mathbf{x}_n, \mathbf{x}_m, 2\sigma^2) - L_j \right) \left(z_{jn}\delta_{i,j} - z_{in} z_{jn} \right) \quad (18)$$

The updating rule with Eq.18 can be implemented in a couple of lines using Matlab.

6. Numerical Experiments

We carried out two experiments. The data sets and clustering results are shown in Fig 1 and 2 respectively. Each one is consisted of five and three clusters. The boundary of the first data set is linear and can be easily partitioned. But the second data set has a highly non-linear boundary. Thus it cannot be separated easily. K-means algorithm definitely fails to partition the data. The suggested algorithm separated both of the data sets perfectly.

7. Discussion

Starting from ' Minimum Entropy Data Partitioning' [1,2], we derived a simpler clustering algorithm. The density is estimated by Parzen window and the complexity and calculation, however, is simplified by using quadratic distance measure for densities [3,4]. The updating rule can be implemented in a couple of lines of codes using Matlab. Lastly, the experiments showed the promising results.

Reference

[1] S. J. Roberts, R. Everson, and I. Rezek. Maximum certainty data partitioning. *Pattern Recognition*, 33:833-839, 2000.

[2] S. J. Roberts, C. Holmes, and D. Denison. Minimum entropy data partitioning using reversible jump Markov chain monte carlo. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(8):909-914, Aug.2001.

[3] J. C. Principe, D. Xu, and J. W. Fisher III. Information-theoretic learning. In S. Haykin, editor, *Unsupervised Adaptive Filtering: Blind Source Separation*. John Wiley & Sons, Inc., 2000.

[4] K. Torkkola and W. M. Campbell. Mutual information in learning feature transformations. In *Proc. Int. Conf. Machine Learning*, pages1015-1022, 2000.

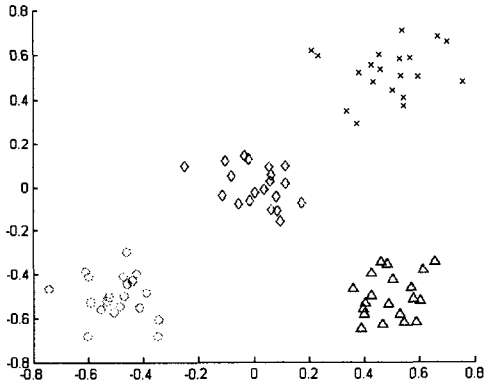

Fig 1. Experiment 1

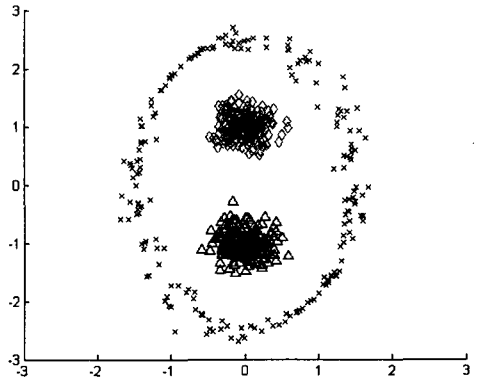

Fig 2. Experiment 2