Advances in Self-Organising Maps, pp.95-101, Springer, 2001

VQ-Based Clustering Algorithm of Piecewise-Dependent-Data

Itshak Lapidot (Voitovetsky)[1] and Hugo Guterman[2]

[1]Department of Software Engineering
Negev Academic College of Engineering
P.O.B. 45 Beer-Sheva, 84100, Israel

[2]Department of Electrical and Computer Engineering
Ben-Gurion University of the Negev
P.O.B. 653, Beer-Sheva, 84105, Israel

itsik@nace.ac.il, hugo@ee.bgu.ac.il

Abstract

In this paper a piecewise-dependent-data (PDD) clustering algorithm is presented, and a proof of its convergence to a local minimum is given. A distortion measure-based model represents each cluster. The proposed algorithm is iterative. At the end of each iteration, a competition between the models is performed. Then the data is regrouped between the models. The "movement" of the data between the models and the retraining allows the minimization of the overall system distortion. The Kohonen Self-Organizing Map (SOM) was used as the VQ model for clustering. The clustering algorithm was tested using data generated from four generators of Continuous Density HMM (CDHMM). It was demonstrated that the overall distortion is a decreasing function.

1 Introduction

Many time signals can be viewed as time-dependent-data, e.g. speech signals, bio-signals, seismic signals, etc. In these, a signals dependence exists between consecutive samples or frames (e.g., the same speaker or the same sleep stage). Most of the clustering algorithms described in the literature are static [1]-[3]. This means that dependence between consecutive vectors is not taken into account. Although, several algorithms exist for clustering of piecewise-dependent-data [4]-[8], no theoretical proofs of convergence have been provided. The objective of this research is to present a piecewise-dependent-data algorithm. This paper is divided as follows: a multi-VQ-based iterative algorithm is presented in section 2. In section 3 a convergence of the iterative algorithm is proven for the minimal distortion sense. The results of an experiment using Kohonen SOM [9] as a VQ algorithm for the cluster models is presented in section 4. The overlapping synthesized database simulates four CDHMM [8] generators. Conclusions are given in section 5.

2 The VQ-Based Clustering Algorithm

For given piecewise-dependent-data, i.e., a database consisting of distinct segments while the vectors in each segment are dependent, the goal is to cluster the input data into R clusters (Fig. 1). We assume that the switching points between non-dependent segments are known (the switching points refer to the boundary between two adjacent segments). The piecewise-dependent-data consists of N vectors, $\mathbf{V} = \{v_n\}_{n=1,\ldots,N}$. These vectors are partitioned into M segments, $\mathbf{V} = \{\mathbf{V_m}\}_{m=1,\ldots,M}$ (equation 1) according to the switching points. The segments have to be clustered into R clusters ($R \le M$), i.e., two vectors that belong to the same segment must be clustered to the same cluster. In static algorithms each cluster is usually represented by one centroid. In the proposed algorithm each cluster is represented by a VQ-based model. For each model a CodeBook (CB) is created. Every CB_r is of size L_r and presents the r-th cluster (equation 2).

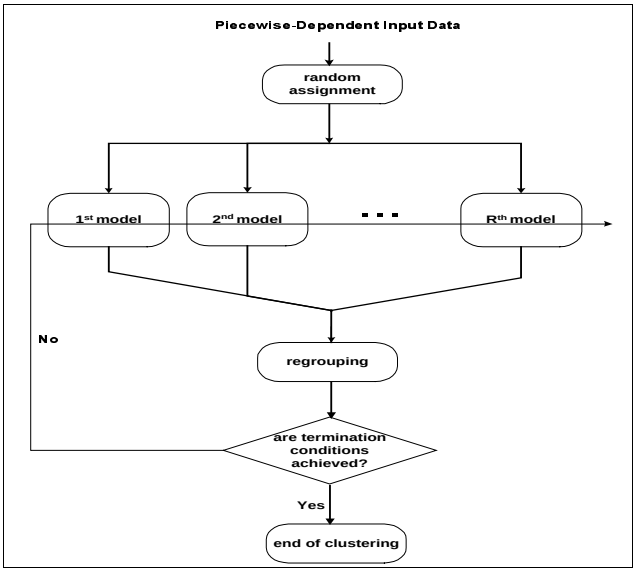

Figure 1. The piecewise-dependent-data clustering algorithm.

$$\begin{cases} \mathbf{V} = \left\{ \underbrace{v_1^1, \ldots, v_{n_1}^1}_{\mathbf{v_1}}, \ldots, \underbrace{v_1^m, \ldots, v_{n_m}^m}_{\mathbf{v_m}}, \ldots, \underbrace{v_1^M, \ldots, v_{n_M}^M}_{\mathbf{v_M}} \right\} = \\ \quad = \{\mathbf{v_m}\}_{m=1,\ldots,M} \\ \sum_{m=1}^{M} n_m = N \end{cases} \tag{1}$$

$$CB_r = \{c_r^l\}_{r=1,\ldots,R\,;\; l=1,\ldots,L_r} . \tag{2}$$

$\left\{c_r^l\right\}_{r=1,\ldots,R\ ;\ l=1,\ldots,L_r}$ is the union of all the Code-Words (CW) that belongs to CB_r.

The initialization of the algorithm is performed by randomly assigning each of the M segment to the R codebooks; $\mathbf{V^{r,0}}$ notates the segments that are partitioned to CB_r at the beginning of the algorithm, and each model is trained with its segments. After the training, the regrouping process is applied and the models are retrained again. After i iterations the partition will be:

$$\begin{cases} \mathbf{V^{r,i}} = \left\{\mathbf{v_m^{r,i}}\right\}_{m=1,\ldots M_{r,i}\ ;\ r=1,\ldots,R} \\ \mathbf{v_m^{r,i}} = \left\{v_{m,n}^{r,i}\right\}_{n=1,\ldots,n_m\ ;\ m=1,\ldots M_{r,i}\ ;\ r=1,\ldots,R} \\ \sum_{r=1}^{R} M_{r,i} = M \end{cases} \tag{3}$$

and the code-books are:

$$CB_r^i = \left\{c_r^{l,i}\right\}_{r=1,\ldots,R\ ;\ l=1,\ldots,L_r} \tag{4}$$

Different algorithms such as Kohonen SOM [9], LBG [10], fuzzy C-means [11], can be used for VQ training.

After the retraining the data is regrouped. The reordering of the data is attained by finding which CB_k^i best fits every $\mathbf{v_m} \in \mathbf{V}$ according to a given distance measure. Thus a new partition, $\mathbf{V^{r,i+1}}$ is produced.

The system has to be retrained according to the new partition. The convergence condition is met when:

$$\mathbf{V^{r,i}} = \mathbf{V^{r,i+1}} \tag{5}$$

3 Proof of System Convergence

The distance measure can be any measure that meets the conditions that satisfy:

$$\begin{cases} d(x,y) \geq 0 \text{ where equality holds if and only if } x = y \\ d(x,y) = d(y,x) \\ d(x,y) \leq d(x,z) + d(z,y) \end{cases}, \tag{6}$$

and the VQ algorithm must converge to at least a local minimum [9], [10].

After the i-th iteration the partition of the data between the models will be according to equation (3), where $\mathbf{V^{r,i}}$ is the data set associated with the r-th model at the i-th iteration, and the CB_r at the i-th iteration is $CB_r^i = \left\{c_r^{l,i}\right\}_{r=1,\ldots,R\ ;\ l=1,\ldots,L_r}$.

Let the distance between the m-th vector of the n-th segment that belongs to CB_q^{i-1} ($v_{m,n}^{q,i-1}$) and $c_r^{l,i}$ be $\mathrm{d}_{\mathrm{m,n}}^{l,\mathrm{i}}(r,q)$. Then the distance between $v_{m,n}^{q,i-1}$ and CB_r^i is:

$$\mathrm{d}_{\mathrm{m,n}}^{\mathrm{i}}(r,q) = \min_{l=1,\ldots,L_r} \left\{ \mathrm{d}_{\mathrm{m,n}}^{l,\mathrm{i}}(r,q) \right\}. \tag{7}$$

The distance between $\mathbf{v_m^{q,i-1}}$ and CB_r^i is:

$$\mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r,q) = \sum_{n=1}^{n_m} \mathrm{d}_{\mathrm{m,n}}^{\mathrm{i}}(r,q), \tag{8}$$

and the minimal distance between the segment that belongs to CB_q at iteration $(i-1)$ from all the CBs is:

$$\begin{aligned} &\mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(j) = \min_{r=1,\ldots,R} \left\{ \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r,q) \right\} \\ &j = \arg\min_{r=0,\ldots,R} \left\{ \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r,q) \right\} \Rightarrow \mathbf{v_m^{q,i-1}} = \mathbf{v_m^{j,i}} \end{aligned}. \tag{9}$$

If after the i-th iteration the overall distance is calculated, the partition before regrouping is:

$$\mathrm{D}^{\mathrm{i}} = \sum_{r=1}^{R} \sum_{m=1}^{M_r} \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r,r) \tag{10}$$

and the distance according to a new partition (after regrouping) is:

$$\tilde{\mathrm{D}}^{\mathrm{i}} = \sum_{r=1}^{R} \sum_{m=1}^{M_r} \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r). \tag{11}$$

Because $\tilde{\mathrm{D}}^{\mathrm{i-1}}$ is the distance before the i-th retraining and D^{i} is the minimum according to the previous partition after retraining, the next inequality holds:

$$\tilde{\mathrm{D}}^{\mathrm{i}} \le \mathrm{D}^{\mathrm{i}} \le \tilde{\mathrm{D}}^{\mathrm{i-1}} \tag{12}$$

if

$$\tilde{\mathrm{D}}^{\mathrm{i}} < \mathrm{D}^{\mathrm{i}}. \tag{13}$$

This means that there exists at least one segment, $\mathbf{v_m^{r,i}} = \mathbf{v_m^{q,i-1}}$, whose distances are:

$$\mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r) < \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(q,q) \tag{14}$$

and there exists a CB_r so that:

$$\mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(r,q) < \mathrm{D}_{\mathrm{m}}^{\mathrm{i}}(q,q). \tag{15}$$

In other words, there exists a better partition of $\mathbf{V}$ that gives a lower distance $\tilde{\mathrm{D}}^{\mathrm{i}}$. If the new partition is chosen, then the previous VQ is not optimal because it was

designed with an other partition. It can be seen that from the i-th to $(i+1)$-th iteration the overall distance did not increase. The iterative process will stop when

$$\begin{cases} \tilde{D}^{i+1} = D^{i+1} \\ \mathbf{V}^{r,i+1} = \mathbf{V}^{r,i} \end{cases}. \tag{16}$$

In this case there is no change in the partition between the two consecutive iterations.

4 Experiments and Results

In order to test the proposed algorithm a dataset consisting of four clusters was created. Each cluster was produced by a CDHMM generator, and consisted of three states with two two-dimensional Gausian's per state. The dataset consisted of 100 segments, the length of each segment being a random variable that was uniformly distributed, $U(5,100)$. Each model was a Kohonen SOM [9] of size 5×5. Clustering results and the distorsion are shown in Fig. 2 and 3, respectively. Although the data overlaps, the models fit the cluster properly. The error was only one segment out of a hundred. The segment was short (seven vectors out of 4726). An error percentage of only about 0.15% was found. Only five iterations were needed for the system to converge.

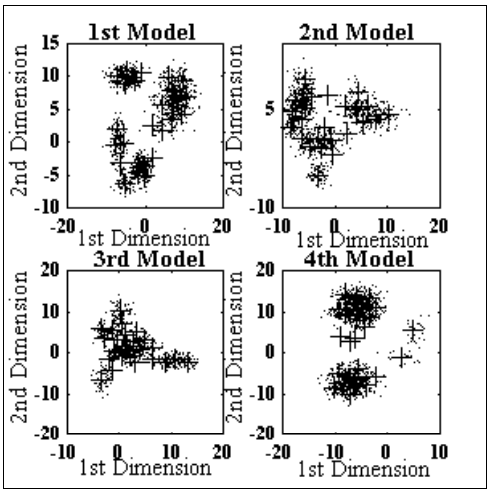

Figure 2. Results of four CHMM clusters. Each subplot presents each model's data (dots) and SOM convergence (+).

5 Conclusions

This paper presented an iterative algorithm for VQ-based PDD clustering. A general proof of the convergence of the algorithm was given as well. The effectiveness of the proposed algorithm was demonstrated on data generated from

four CDHMMs. The models converged correctly to the clusters and the overall distortion function decreased.

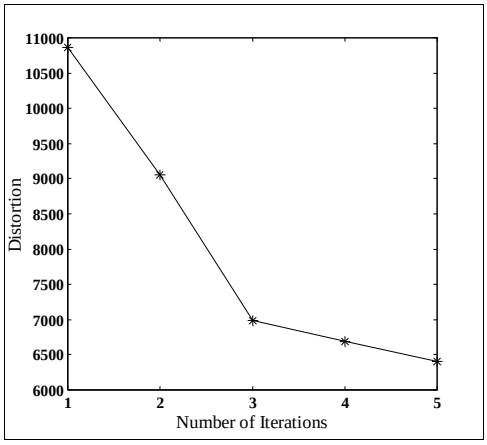

Figure 3. Distortion as a function of the number of iterations for data of 4 CHMMs.

Reference

1. Man Y. and Gath I.Detection and separation of ring-shaped clusters using fuzzy clustering. IEEE Trans. Patt. Anal. Machine Intell. 1994; 16:855-861
2. Rose K., Gurewitz E., and Fox G. C. Statistical mechanics and phase transitions in clustering. Physical Review Letters 1990; 65:945-948
3. Gath I. and Geva A. B. Unsupervised optimal fuzzy clustering. IEEE Trans. Patt. Anal. Machine Intell. 1989; 11:773-781
4. Voitovetsky I., Guterman H., and Cohen A. Unsupervised speaker classification using self-organizing maps (SOM). Proc. Neural Networks for Signal Processing VII IEEE Workshop 1997; 578-587
5. Cohen A. and Lapidus V. Unsupervised, text independent, speaker classification. Proc. of the Int. Conf. on Signal Processing Application and Technology 1996; 1745-1749
6. Kohlmorgen J., Muller K.-R., and Pawelzik K. Segmentation and identification of drifting dynamical systems. Proc. Neural Networks for Signal Processing VII IEEE Workshop 1997; 326-335
7. Moon T. K. Temporal pattern recognition using fuzzy clustering. Proc. of Third IEEE International Conference on Fuzzy Systems 1994; 1:432-435
8. Rabiner L. and Juang B.-H. Fundamentals of Speech Recognition. Prentice Hall, 1993
9. Kohonen T. The self-organizing map. Proc. IEEE 1990; 78:1464-1480
10. Linde Y., Buzo A., and Gray R. M. An algorithm for vector quantizer design. IEEE Trans. Commun. 1980; 28:84-95
11. Pedrycz W. Fuzzy sets in pattern recognition: methodology and methods. Pattern Recognition 1990; 23121-146