2010 International Conference on Pattern Recognition

Maximum Likelihood Estimation of Gaussian Mixture Models Using Particle Swarm Optimization

Çağlar Arı
Department of Electrical and Electronics Engineering
Bilkent University
Bilkent, 06800, Ankara, Turkey
cari@ee.bilkent.edu.tr

Selim Aksoy
Department of Computer Engineering
Bilkent University
Bilkent, 06800, Ankara, Turkey
saksoy@cs.bilkent.edu.tr

***Abstract*—We present solutions to two problems that prevent the effective use of population-based algorithms in clustering problems. The first solution presents a new representation for arbitrary covariance matrices that allows independent updating of individual parameters while retaining the validity of the matrix. The second solution involves an optimization formulation for finding correspondences between different parameter orderings of candidate solutions. The effectiveness of the proposed solutions are demonstrated on a novel clustering algorithm based on particle swarm optimization for the estimation of Gaussian mixture models.**

I. Introduction

Clustering is an unsupervised classification technique where unlabeled data are partitioned into groups of similar objects. Among many, iterative partitioning methods such as k-means and its extensions have been widely used. Like most other clustering algorithms, these methods share common problems. For instance, cluster modeling capability of the k-means algorithm is limited to spherical clusters with similar number of data points. Fitting parametric density models such as Gaussian mixture models (GMM) by using the Expectation-Maximization (EM) algorithm can be interpreted as model-based clustering methods where each mixture component is viewed as a cluster. Due to its capability of discovering clusters of arbitrary ellipsoidal shapes, the GMM-EM algorithm is a superior version of k-means. However, as the number of dimensions increases, significant difficulties arise in the estimation of covariance matrices for GMMs. Furthermore, due to their objective of interest being a non-convex optimization problem, k-means and GMM-EM easily get trapped in local minima, and are very sensitive to initializations. The common practice is to run these algorithms many times from different initial values and to employ several local search heuristics.

With the advent of inexpensive high-speed computers, many researchers are increasingly turning to population-based stochastic search algorithms to solve complicated problems. Similarly, there is a growing interest in the use of these methods to solve clustering problems. For example, Chang et al. [1] designed a genetic algorithm for improving k-means, Maulik and Saha [2] proposed a modified differential evolution algorithm for fuzzy c-means clustering, and Schroeter et al. [3] used a genetic algorithm for the estimation of GMMs. In the past decade, the applications of GMMs have widened substantially. In addition to their applications in mainstream statistical analyses, they are widely used in unsupervised pattern recognition, medical imaging, and speech recognition [4]. Hence, it is of great interest to improve the effectiveness and broaden the use of population-based search algorithms to the estimation of GMMs.

In this paper, we propose solutions to problems that prevent the effective use of population-based algorithms in clustering problems, and present a novel clustering algorithm based on particle swarm optimization (PSO) for maximum likelihood estimation of GMMs. Solutions to two vital problems are presented and their uses are demonstrated in this paper. First of all, in clustering problems with K clusters, there exist $K!$ ways to represent parameters of different clusters as a candidate solution. A correspondence identification problem arises when different candidates are required to interact with each other. Furthermore, there is no suitable parametrization and a corresponding algorithm to represent and estimate arbitrary covariance matrices from data. Section II gives general description of the PSO algorithm. Section III discusses the limitations of existing methods and presents the proposed clustering algorithm. Section IV illustrates its effectiveness in the clustering of various data sets.

II. Particle swarm optimization

PSO is a population-based stochastic search algorithm based on the social interaction among different swarm animals. In PSO, each member of the population is called a particle. Each particle Z consists of a position vector Z_X and velocity vector Z_V. Position of each particle $Z_X \in \mathbb{R}^m$ corresponds to a candidate solution for an m-dimensional optimization problem. A fitness function defined for the optimization problem of interest is used to assign a goodness value to a particle based on its position. The particle having the best fitness value is called the global best (Z_{GB}). In addi-

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.188

tion, each particle keeps track of its own best position since the first iteration and it is called the personal best (Z_{PB}). In the first iteration, particles are initialized with random positions and small random velocities. In the subsequent iterations, each of the m velocity components in Z_V is computed independently using its previous value, the global best, and the particle's own personal best in a stochastic manner as

$$Z_{V_{t+1}} = w\, Z_{V_t} + c_1\, U_{1_t}\, (Z_{\text{PB}_t} - Z_{X_t}) + c_2\, U_{2_t}\, (Z_{\text{GB}_t} - Z_{X_t}) \quad (1)$$

where w is called the inertia weight, U_{1_t} and U_{2_t} stand for random numbers sampled from Uniform$[0, 1]$, c_1 and c_2 are called acceleration weights, and t is the iteration. The new position of the particle is computed using its old position and its current velocity as

$$Z_{X_{t+1}} = Z_{X_t} + Z_{V_{t+1}}, \quad (2)$$

and its personal best is updated based on its new fitness value. In addition, the global best of the population is updated after each iteration using particles' new fitness values.

The most important property of PSO is its use of the global best to coordinate the movement of all particles and the use of personal bests to remember the history of each particle where the global best serves as the current state of the problem and the personal bests serve as the current states of the particles.

III. Proposed clustering algorithm

We consider a family of mixtures of K multivariate Gaussian distributions on $\mathbb{R}^d$ indexed by the set of parameters $\Theta = \{\pi_1, \mu_1, \Sigma_1, \ldots, \pi_K, \mu_K, \Sigma_K\}$ such that $\mu_k \in \mathbb{R}^d$ are the means, $\Sigma_k \in \mathbb{S}^d_{++}$ are the covariance matrices, and $\pi_k \in [0, 1]$ are the prior probabilities for clusters $k = 1, \ldots, K$, where $\sum_{k=1}^K \pi_k = 1$. All data points $\{x_j\}_{j=1}^N$ are assumed to be i.i.d. according to the mixture probability density function $p_\Theta(x_j) = \sum_{k=1}^K \pi_k p_k(x_j|\mu_k, \Sigma_k)$. The objective is to find the parameters $\hat{\Theta}$ by maximizing the likelihood of the data points. We solve this estimation problem by minimizing the negative log-likelihood, i.e.,

$$\underset{\mu_1,\ldots,\mu_K,\Sigma_1,\ldots,\Sigma_K}{\text{minimize}} \; -\sum_{j=1}^N \log\Big(\sum_{k=1}^K \pi_k p_k(x_j|\mu_k, \Sigma_k)\Big) \quad (3)$$

where only the means $\{\mu_k\}_{k=1}^K$ and covariances $\{\Sigma_k\}_{k=1}^K$ are to be estimated, and the prior probabilities are calculated based on the probabilistic assignments of the data points.

A. Particle definition

Each mean vector is parametrized with d real numbers. Parametrization of arbitrary covariance matrices requires $d(d+1)/2$ parameters. An important problem is the lack of a suitable parametrization for covariance matrices. It is not possible to directly use upper (or lower) triangular part of a covariance matrix because each component in the particle position vector Z_X is independently updated using the corresponding component in the velocity vector Z_V in (2), and independent updates of the covariance components will very often violate the requirement for the matrix being positive definite. Hence, existing population-based stochastic search algorithms limit their covariance matrices to be diagonal [5] or do not use any covariance structure at all [1], [2].

We propose a new parametrization where the parameters are unique, are independently modifiable, and have certain upper and lower bounds. The proposed parametrization is based on eigenvalue decomposition ($\Sigma = V \Lambda V^T$ where Λ is a diagonal and V is an orthogonal matrix). Let $\{\lambda_i\}_{i=1}^d$, $\lambda_i \in \mathbb{R}_{++}$ denote the eigenvalues and $\{v_i\}_{i=1}^d$, $v_i \in \mathbb{R}^d$ denote the eigenvectors of the d-dimensional covariance matrix $\Sigma \in \mathbb{S}^d_{++}$. The covariance matrix can be written in terms of its eigenvalues and eigenvectors as $\Sigma = \sum_{i=1}^d \lambda_i v_i v_i^T$. However, there is no order relation among the eigenvalues in this summation, and the multiplication of any eigenvector by -1 does not change the representation of covariance matrices. Therefore, there exist $2^d d!$ different representations.

We propose to parametrize the eigenvalues with d positive real numbers in the order determined by the cyclic Jacobi eigenvalue decomposition algorithm. In the cyclic Jacobi algorithm, for a given symmetric matrix $\Sigma \in S^d$ and a minimum error value $\epsilon > 0$, Σ is overwritten with $V^T \Sigma V$ where V is an orthogonal matrix until the absolute sum of the off-diagonal entries of $V^T \Sigma V$ is less than ϵ. The cyclic Jacobi algorithm starts with V initialized to the identity matrix. Then, while the absolute sum of off-diagonal entries of $V^T \Sigma V$ are greater than ϵ, it computes cosine-sine pairs $(\cos\phi^{pq}, \sin\phi^{pq})$ such that if $\hat{\Sigma} = G(p, q, \phi^{pq})^T \Sigma G(p, q, \phi^{pq})$ then $\hat{\Sigma}_{pq} = \hat{\Sigma}_{qp} = 0$ and $V = VG(p, q, \phi^{pq})$ for $p = 1, \ldots, d-1$, $q = p+1, \ldots, d$. $G(p, q, \phi^{pq})$ stands for a Givens rotation matrix [6] with 3 input parameters, 2 indices p and q, and an angle ϕ^{pq}. A Givens rotation matrix $G(p, q, \phi^{pq})$ has the form

$$G(p, q, \phi^{pq}) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & cos(\phi^{pq}) & \cdots & sin(\phi^{pq}) & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & & -sin(\phi^{pq}) & \cdots & cos(\phi^{pq}) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}. \quad (4)$$

By avoiding the aforementioned problems, the eigenvector matrix is parametrized with $d(d-1)/2$ Givens rotation angles. These angles can be computed using QR factorization. QR factorization of an orthogonal matrix $V = QR$ can be done via Givens rotation matrices where the Q matrix can be written as a multiplication of $L = d(d-1)/2$ Givens rotation matrices (G_i's), i.e., $Q = G_1 G_2 \ldots G_L$ [6]. In the QR algorithm, for the given indices p and q, the angle ϕ^{pq}

is calculated using the $V(p,p)$ and $V(q,p)$ values, and then, V is premultiplied with the transpose of the Givens rotation matrix as $V = G(p,q,\phi^{pq})^T V$ which zeros the $V(p,q)$. This process is performed for $p = 1,\ldots,d-1$, $q = p+1,\ldots,d$, and the resulting matrix R is a diagonal matrix with entries being either $+1$ or -1, and the angles $\phi^{pq} \in [\frac{-\pi}{2}, \frac{\pi}{2}]$.

B. Correspondence identification

Another important problem in parameter updates in stochastic search algorithms as in Section II is the unknown correspondence between different components of two particles. In clustering problems with K clusters, there exist $K!$ different particle representations due to different parameter orderings for the same candidate solution. For instance, for $K = 2$, means can be written as either $[\mu_1, \mu_2]$ or $[\mu_2, \mu_1]$. Suppose that one particle is in the first form and the global best is in the second. When that particle is updated, it will use μ_2 to update μ_1 erroneously. This problem is often ignored in the literature but it causes major problems for population-based algorithms because due to random movement of particles, the correspondences between cluster parameters of different particles are never known, and particle updates using (1) become based on wrong interactions.

We propose a matching algorithm to find the right correspondence relation between the components of a particle and the global best for correct updates. The correspondence identification problem is formulated as a minimum cost network flow optimization problem. The objective is to find the correspondence relation that minimizes the sum of weighted cluster mean distances where $\{\mu_{X_{PB}}^{(i)}\}_{i=1}^K$ represent the set of personal best means of a particle of interest and $\{\mu_{X_{GB}}^{(j)}\}_{j=1}^K$ represent the set of means for the global best particle. In addition, the global best particle's covariance matrices $\{\Sigma_{X_{GB}}^{(j)}\}_{j=1}^K$ are used for weighting purposes. The cost of matching the former particle's i'th cluster parameters to the global best particle's j'th cluster parameters is computed as

$$c_{ij} = (\mu_{X_{PB}}^{(i)} - \mu_{X_{GB}}^{(j)})^T (\Sigma_{X_{GB}}^{(j)})^{-1} (\mu_{X_{PB}}^{(i)} - \mu_{X_{GB}}^{(j)}), \quad (5)$$

and the correspondences are found by solving the following optimization problem:

$$\begin{aligned} \underset{y_{11},\ldots,y_{KK}}{\text{minimize}} \quad & \textstyle\sum_{i=1}^K \sum_{j=1}^K c_{ij} y_{ij} \\ \text{subject to} \quad & \textstyle\sum_{i=1}^K y_{ij} = 1, \quad \forall j \in \{1,\ldots,K\} \\ & \textstyle\sum_{j=1}^K y_{ij} = 1, \quad \forall i \in \{1,\ldots,K\} \\ & y_{ij} = \begin{cases} 1, & \text{correspondence between} \\ & \quad i\text{'th and } j\text{'th clusters} \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (6)$$

C. Update equations

The correspondence relation computed in (6) is denoted with a function $f(k)$ that maps the current particle's cluster index k to the corresponding global best particle's cluster index $f(k)$. Mean and covariance parameters of particles are updated as in (1) and (2) but by using correct correspondence relations as follows:

- Mean update equations

$$\mu_{V_{t+1}}^{(k)} = w\mu_{V_t}^{(k)} + c_1(\mu_{PB_t}^{(k)} - \mu_{X_t}^{(k)}) + c_2(\mu_{GB_t}^{(f(k))} - \mu_{X_t}^{(k)}) \quad (7)$$

$$\mu_{X_{t+1}}^{(k)} = \mu_{X_t}^{(k)} + \mu_{V_{t+1}}^{(k)} \quad (8)$$

- Covariance update equations — Angle updates

$$\begin{aligned} \phi_{V_{t+1}}^{pq,(k)} = {} & w\phi_{V_t}^{pq,(k)} + c_1(\phi_{PB_t}^{pq,(k)} - \phi_{X_t}^{pq,(k)}) \\ & + c_2(\phi_{GB_t}^{pq,(f(k))} - \phi_{X_t}^{pq,(k)}) \end{aligned} \quad (9)$$

$$\phi_{X_{t+1}}^{pq,(k)} = \phi_{X_t}^{pq,(k)} + \phi_{V_{t+1}}^{pq,(k)} \quad (10)$$

- Covariance update equations — Eigenvalue updates

$$\begin{aligned} \lambda_{V_{t+1}}^{i,(k)} = {} & w\lambda_{V_t}^{i,(k)} + c_1(\lambda_{PB_t}^{i,(k)} - \lambda_{X_t}^{i,(k)}) \\ & + c_2(\lambda_{GB_t}^{i,(f(k))} - \lambda_{X_t}^{i,(k)}) \end{aligned} \quad (11)$$

$$\lambda_{X_{t+1}}^{i,(k)} = \lambda_{X_t}^{i,(k)} + \lambda_{V_{t+1}}^{i,(k)} \quad (12)$$

IV. Experiments

We evaluated the performance of the proposed algorithm using four data sets from the UCI Machine Learning Repository. The *wine* data set consists of 178 points having 13 features and 3 classes. The *glass* data set has 214 points with 9 features and 6 classes. The *Statlog image segmentation* data set contains 2310 points with 19 features and 7 classes. The *Statlog Landsat satellite* data set has 4435 points with 36 features and 7 classes. Comparative experiments were performed using GMM-EM as well.

In each experiment, the proposed PSO algorithm was run using 50 particles with different random initializations. To be comparable, the GMM-EM procedure was run using 50 different initializations where one of the GMM-EM runs and one of the PSO particles used the same initialization. For each initialization, first, K mean vectors were randomly selected from the data points. Then, initial clusters were formed by assigning each data point to the closest mean. Finally, the covariance matrix of each cluster was computed and the angles and eigenvalues were estimated using the cyclic Jacobi algorithm and QR factorization. After the initialization, both the PSO algorithm and each GMM-EM procedure were run for 500 iterations. At the end of the experiment, the parameters corresponding to the global best particle constituted the result of the PSO algorithm, and the parameters of the best GMM-EM run (among the 50 runs with different initializations) with the highest likelihood value were used as the competing GMM-EM result. These experiments were repeated 50 times, corresponding to a total of 50 PSO runs with 50 particles for each run and a total of 2500 GMM-EM runs.

Quantitative performance of unsupervised clustering was measured using the average of cluster entropies computed

from the distribution of the true class labels within individual clusters and class entropies computed from the distribution of individual classes to multiple clusters. A smaller overall entropy value indicates better performance. Figure 1 shows the plots of overall entropy versus the number of clusters for all four data sets. The results for 50 experiments are summarized using average values with error bars at one standard deviation. In addition to obtaining a smaller negative log-likelihood value in all experiments for all data sets, the proposed clustering algorithm also resulted in better (i.e., smaller) entropy values than the GMM-EM algorithm in all cases. We can argue that the reason behind this performance is that the proposed algorithm does not need data for explicit estimation of the cluster parameters because it generates the parameters via update equations, whereas EM is highly data dependent in the calculation of the parameters. In the absence of sufficient data at certain places in the feature space, PSO can still generate different means and covariances, and can find a direction which decreases the negative log-likelihood function. However, EM needs data and even when there is data, GMM-EM is highly affected by the noise and irregularities on its path.

V. Conclusions

We presented solutions to two important problems that prevent the effective use of population-based algorithms for clustering. We demonstrated their effectiveness with a clustering algorithm based on PSO on various data sets. As future work, we are planning to integrate fast local search algorithms like EM with PSO to increase the PSO's capability of fast detection of local minima. Mechanisms that lead to more efficient and robust coverage of the search space become vital as the dimensionality increases and the feature space gets sparser.

Acknowledgment

This work was supported in part by the TUBITAK CAREER Grant 104E074.

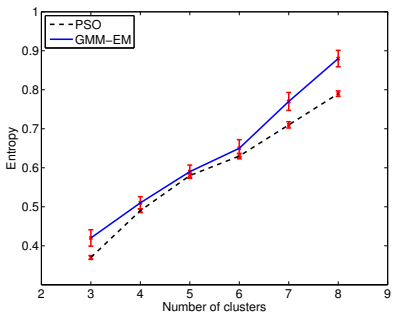
References

[1] D.-X. Chang, X.-D. Zhang, and C.-W. Zheng, "A genetic algorithm with gene rearrangement for k-means clustering," *Pattern Recognition*, vol. 42, no. 7, pp. 1210–1222, July 2009.

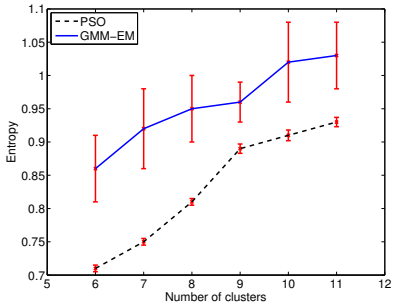
[2] U. Maulik and I. Saha, "Modified differential evolution based fuzzy clustering for pixel classification in remote sensing imagery," *Pattern Recognition*, vol. 42, no. 9, pp. 2135–2149, September 2009.

[3] P. Schroeter, J.-M. Vesin, T. Langenberger, and R. Meuli, "Robust parameter estimation of intensity distributions for brain magnetic resonance images," *IEEE Trans. on Medical Imaging*, vol. 17, no. 2, pp. 172–186, April 1998.

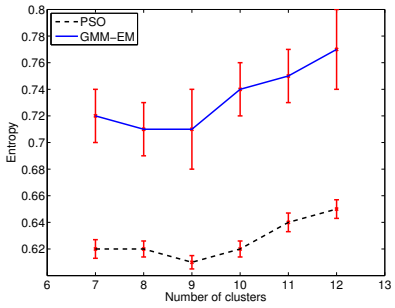
[4] G. McLachlan and D. Peel, *Finite mixture models*. Wiley-Interscience, 2004.



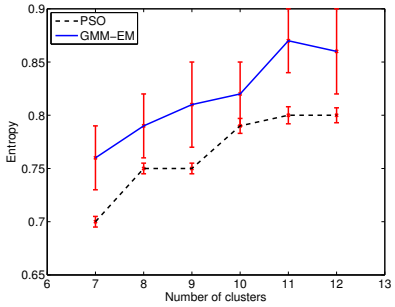
(a) Wine



(b) Glass



(c) Statlog segmentation



(d) Statlog Landsat

Figure 1. Clustering results for four data sets.

[5] A. Paoli, F. Melgani, and E. Pasolli, "Clustering of hyperspectral images based on multiobjective particle swarm optimization," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 47, no. 12, pp. 4175–4188, December 2009.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Johns Hopkins University Press, 1996.