A Cooperative and Penalized Competitive Learning Approach to Gaussian Mixture Clustering

Yiu-ming Cheung and Hong Jia

Department of Computer Science, Hong Kong Baptist University,
Hong Kong SAR, China
{ymc,hjia}@comp.hkbu.edu.hk

Abstract. Competitive learning approaches with penalization or cooperation mechanism have been applied to unsupervised data clustering due to their attractive ability of automatic cluster number selection. In this paper, we further investigate the properties of different competitive strategies and propose a novel learning algorithm called Cooperative and Penalized Competitive Learning (CPCL), which implements the cooperation and penalization mechanisms simultaneously in a single competitive learning process. The integration of these two different kinds of competition mechanisms enables the CPCL to have good convergence speed, precision and robustness. Experiments on Gaussian mixture clustering are performed to investigate the proposed algorithm. The promising results demonstrate its superiority.

Keywords: Competitive Learning, Cooperation, Penalization, Gaussian Mixture Clustering, Number of Clusters.

1 Introduction

As a typical competitive learning algorithm, k-means [1] has a variety of applications in the different scientific areas such as image processing, clustering analysis, and so on. However, it suffers from a selection problem of cluster number. That is, k-means needs to pre-assign the number of clusters exactly; otherwise, it will almost always give out an incorrect clustering result.

In the literature, one main kind of technique to solve this selection problem is to introduce some competitive learning mechanisms into an algorithm so that it can perform automatic cluster number selection during the learning process. For example, the Rival Penalized Competitive Learning (RPCL) [2] can automatically select the cluster number by gradually driving extra seed points, i.e. those data points that are learnable towards the cluster centers in the input space, far away from the input data set. Nevertheless, the empirical studies have also found that the performance of RPCL may completely break down without an appropriate delearning rate. Under the circumstances, paper [3] has proposed an improved version, namely Rival Penalization Controlled Competitive Learning (RPCCL),

which determines the rival-penalized strength based on the distance between the winner and the rival relative to the current input. However, both of RPCL and RPCCL always penalized the extra seed points even if they are much far away from the input data set. Consequently, the seed points as a whole will not tend to convergence. By contrast, another variant of RPCL called Stochastic RPCL (S-RPCL) [4], developed from the Rival Penalized Expectation-Maximization (RPEM) algorithm, can lead to a convergent learning process by penalizing the nearest rival stochastically based on its posterior probability. Nevertheless, when the data clusters are overlapped, the convergence speed of S-RPCL, as well as the RPCL, may become slow and the final locations of seed points may have a bias from the cluster centers.

Alternatively, Competitive and Cooperative Learning (CCL) [5] implements a cooperative learning process, in which the winner will dynamically select several nearest competitors to form a cooperative team to adapt to the input together. The CCL can make all the seed points converge to the corresponding cluster centers and the number of those seed points stayed at different positions is exactly the cluster number. Nevertheless, the performance of CCL is somewhat sensitive to the initial positions of seed points. To overcome this difficulty, Li et al. [6] have proposed an improved variant, namely Cooperation Controlled Competitive Learning (CCCL) method, in which the learning rate of each seed point within the same cooperative team is adjusted adaptively. Unfortunately, the CCCL may still not work well if the initial seed points are all gathered in one cluster.

This paper will present a new competitive learning algorithm, namely Cooperative and Penalized Competitive Learning (CPCL), which performs the two different kinds of learning mechanisms simultaneously: cooperation and penalization, during the single competitive learning process. That is, given an input, the winner generated from the competition of all seed points will not only dynamically select several nearest competitors to form a cooperative team to adapt to the input together, but also penalize some other seed points which compete intensively with it. The cooperation mechanism here enables the closest seed points to update together and gradually converge to the corresponding cluster centers while the penalization mechanism supplies the other seed points with the opportunity to wander in the clustering space and search for more appropriate cluster centers. Consequently, this algorithm features the fast convergence speed and the robust performance against the initialization of the seed points. The experiments have demonstrated its outstanding performance and robustness.

2 Cooperative and Penalized Competitive Learning (CPCL) Algorithm

Suppose N inputs, $X_1, X_2, ..., X_N$, come from k^* unknown clusters, and k ($k \geq k^*$) seed points $m_1, m_2, ..., m_k$ are randomly initialized in the input space. Subsequently, given an input X_i each time, as described in [7], the winner among k seed points is determined by

$$I(j|X_i) = \begin{cases} 1, & if \ j = \arg\min_{1 \leq r \leq k} \gamma_r ||X_i - m_r||^2 \\ 0, & otherwise \end{cases}, \tag{1}$$

with the relative winning frequency γ_r of m_r defined as

$$\gamma_r = \frac{n_r}{\sum_{j=1}^{k} n_j}, \tag{2}$$

where n_j is the winning frequency of m_j in the past. That means the winning chances of frequent winning seed points are gradually reduced by an implicit penalty. After selecting out the winner m_w, the circle centered at m_w with the radius $||m_w - X_i||$ is regarded as the territory of m_w. Any other seed points intruded into this territory will either cooperate with the winner or be penalized by it.

The winner m_w in this learning approach always chooses the seed points nearest to it as its cooperators and the number of cooperators needed by the winner is gradually increased as the learning process repeats. Consequently, we define that in the ith learning epoch of the algorithm, the winner can at most choose C_i cooperators, where $C_i = \min\{i, k-1\}$. This kind of cooperating scheme ensures that the seed points have enough opportunities to drift in the whole input space and converge smoothly. Each member in the cooperating team, denoted as m_o, will be updated by

$$m_o^{new} = m_o + \eta \frac{||m_w - X_i||}{\max(||m_w - X_i||, ||m_o - X_i||)} (X_i - m_o), \tag{3}$$

where η is a specified positive learning rate. It can be seen that a cooperator will have a full learning rate η as $||m_o - X_i|| \leq ||m_w - X_i||$. Otherwise, the learning strength is gradually attenuated when the distance between the cooperator and the current input increases.

The other non-cooperating seed points in the winner's territory, denoted as m_p, will be penalized with a dynamical penalizing rate:

$$m_p^{new} = m_p - \eta \frac{||m_w - X_i||}{||m_p - X_i||} (X_i - m_p). \tag{4}$$

That is, the closer the seed point is to the input, the more penalization it will suffer from. Consequently, the CPCL algorithm can be given as follows:

Step 1: Pre-specify the number k of clusters $(k \geq k^*)$, and initialize the k seed points $\{m_1, m_2, ..., m_k\}$. Set $t = 1$, $i = 1$ and $n_j = 1$ with $j = 1, 2, ..., k$, where t and i are the number of epochs and input data, respectively.

Step 2: Given an input X_i, calculate $I(j|X_i)$ by Eq. (1).

Step 3: Determine the winner unit m_w. Let S_w be the set of seed points fallen into the territory of m_w. That is, let $S_w = \{m_w\}$, and span S_w by

$$S_w = S_w \cup \{m_j| \ ||m_w - m_j|| \leq ||m_w - X_i||\}. \tag{5}$$

Step 4: Sort the units in S_w based on the distance between each unit to the winner m_w. We denote the sorted units as: $m_1', m_2', ..., m_s'$, with

$$||m_1' - m_w|| \leq ||m_2' - m_w|| \leq \cdots \leq ||m_s' - m_w|| \tag{6}$$

where $s = |S_w|$.

Step 5: Select a subset S_c of S_w to form a cooperating team of m_w, where $S_c = \{m'_1, m'_2, ..., m'_q\}$ with $q = |S_c| = min\{s, t\}$. Then update all members in S_c by Eq. (3).

Step 6: Let $S_p = S_w - S_c$, then, penalize all seed points in S_p by Eq. (4).

Step 7: Update n_w by $n_w^{new} = n_w^{old} + 1$. Let $i = i + 1$, $t = 1 + \lfloor i/N \rfloor$.

The above Step 2 to Step 7 are iterated for each input until all m_js converge.

3   Experimental Results

3.1   Experiment 1

To demonstrate the performance of the CPCL algorithm in comparison with the CCCL and S-RPCL, we generated 1,000 data points from a mixture of three 2-dimension Gaussian densities:

$$p(X|\Theta) = 0.3G(X|\mu_1, 0.15I) + 0.4G(X|\mu_2, 0.15I) + 0.3G(X|\mu_3, 0.15I), \quad (7)$$

with $\mu_1 = [1.0, 1.0]^T$, $\mu_2 = [1.0, 2.5]^T$ and $\mu_3 = [2.5, 2.5]^T$. Six seed points were randomly initialized in the input space. For each algorithm, the learning rate η was set at 0.001 and φ in CCCL algorithm was set to 0.5 according to [6].

After 200 learning epochs, the positions of seed points obtained by the three algorithms are shown in Fig. 1(a) to Fig. 1(c), respectively. It can be seen that all the three algorithms have identified the true number of clusters successfully. However, the S-RPCL had not located the cluster centers accurately yet. Moreover, Fig. 2 shows the learning curve of m_js via each method. It can be seen that the CPCL approach is faster than the CCCL and the S-RPCL had not converged after 200 epochs. This scenario shows the good performance of CPCL in terms of convergence rate and precision.
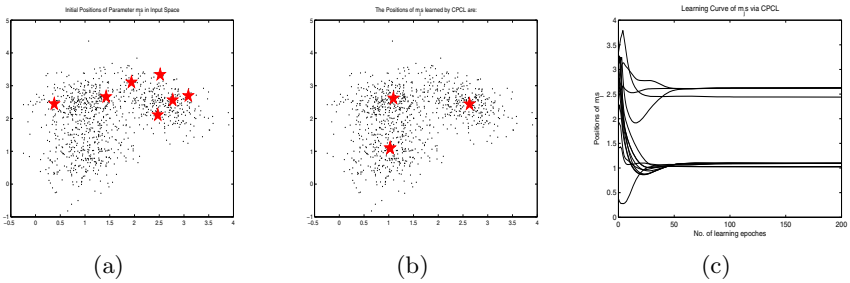
3.2   Experiment 2

In this experiment, we further investigated the performance of CPCL on the mixture clusters that were seriously overlapped and a cluster was in elliptical

(a)   (b)   (c)
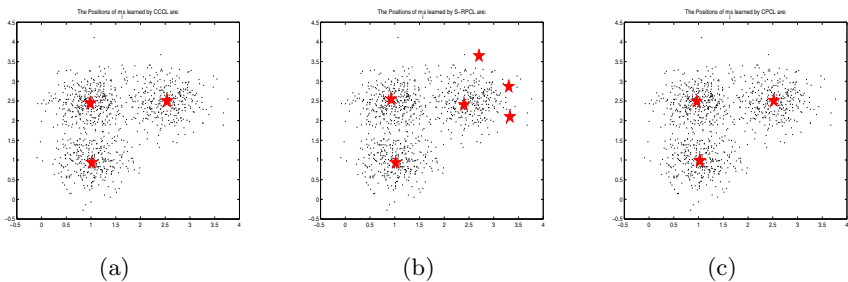
Fig. 1. The positions of six seed points marked by "★" in the input space in Experiment 1 learned by (a) CCCL, (b) S-RPCL and (c) CPCL respectively
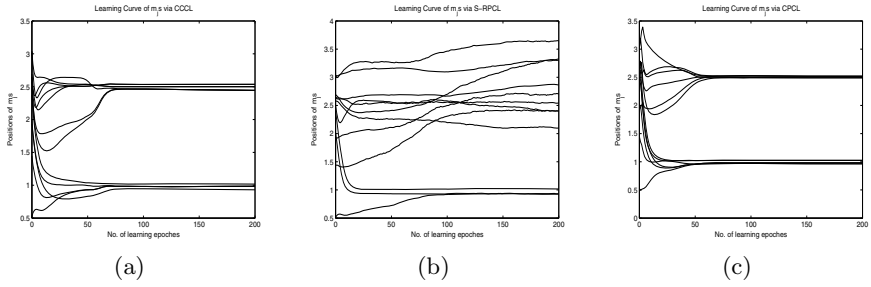
Fig. 2. The learning curves of six seed points in Experiment 1 obtained by (a) CCCL, (b) S-RPCL, and (c) CPCL, respectively

Fig. 3. The results of Experiment 2: (a) the initial positions of seven seed points marked by "★" in the input space, (b) the positions of seed points learned by CPCL, and (c)the learning curves of seven seed points

shape. Similar to Experiment 1, $1,000$ data points were generated from a mixture of three 2-dimension Gaussian densities:

$$p(X|\Theta) = 0.3G(X|\mu_1, \Sigma_1) + 0.4G(X|\mu_2, \Sigma_2) + 0.3G(X|\mu_3, \Sigma_3), \qquad (8)$$

with $\mu_1 = [1.0, 1.0]^T$, $\mu_2 = [1.0, 2.5]^T$, $\mu_3 = [2.5, 2.5]^T$, $\Sigma_1 = \begin{pmatrix} 0.2, 0.05 \\ 0.05, 0.3 \end{pmatrix}$, $\Sigma_2 = \begin{pmatrix} 0.2, 0.0 \\ 0.0, 0.2 \end{pmatrix}$ and $\Sigma_3 = \begin{pmatrix} 0.2, -0.1 \\ -0.1, 0.2 \end{pmatrix}$. We randomly initialized 7 seed points in the input space as shown in Fig. 3(a). From Fig. 3(b), we can see that the seed points learned by CPCL had been converged accurately to the corresponding cluster centers after 200 learning epochs. Furthermore, Fig. 3(c) shows that the learning curves of m_js had converged during the first 50 epochs. Also, it can be seen from this experiment that, although the concept of winner's territory in CPCL is based on Euclidean distance only, this new algorithm can work well on not only ball-shaped data clusters, but also elliptical ones.

4   Concluding Remarks

In this paper, we have presented a novel CPCL algorithm, which performs the competition with the two different kinds of mechanisms simultaneously: cooperation and penalization. On the one hand, the cooperation mechanism enables the closest seed points to update together and gradually converge to the corresponding cluster centers, which gives the algorithm good convergence speed and high precision. On the other hand, the penalization mechanism provides the other seed points with the opportunity to wander in the clustering space, which enables it to perform the clustering problem with the robustness against the initialization of the seed points and the overlap of the data clusters. The experimental results have shown its outstanding performance. Essentially, the underlying learning mechanism of the proposed approach is applicable to the general density mixture clustering, although this paper has presented the details of this approach on Gaussian mixture clustering only.

Acknowledgment

References

1. MacQueen, J.B.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, Calif., USA, pp. 281–297 (1967)
2. Xu, L., Krzyzak, A., Oja, E.: Rival Penalized Competitive Learning for Clustering Analysis. RBF Net, and Curve Detection. IEEE Transactions on Neural Networks 4, 636–648 (1993)
3. Cheung, Y.M.: Rival penalization controlled competitive learning for data clustering with unknown cluster number. In: Proceedings of 9th International Conference on Neural Information Processing, pp. 18–22 (2002)
4. Cheung, Y.M.: Maximum weighted likelihood via rival penalized EM for density mixture clustering with automatic model selection. IEEE Transactions on Knowledge and Data Engineering 17, 750–761 (2005)
5. Cheung, Y.M.: A competitive and cooperative learning approach to robust data clustering. In: Proceedings of the IASTED International Conference of Neural Networks and Computational Intelligence (NCI 2004), pp. 131–136 (2004)
6. Li, T., Pei, W.J., Wang, S.P., Cheung, Y.M.: Cooperation Controlled Competitive Learning Approach for Data Clustering. In: Proceedings of 2008 International Conference on Computational Intelligence and Security (CIS 2008), pp. 24–29 (2008)
7. Ahalt, S.C., Krishnamurty, A.K., Chen, P., Melton, D.E.: Competitive learning algorithms for vector quantization. Neural Networks 3, 277–291 (1990)