Clustering Analysis of Gene Expression Data without Knowing Cluster Number

PhD Student: Hong Jia
Principal Supervisor: Prof. Yiu-ming Cheung

Department of Computer Science and Institute of Computational and Theoretical Studies
Hong Kong Baptist University, Hong Kong SAR, China

Department of Computer Science

香港浸會大學
HONG KONG BAPTIST UNIVERSITY

Outline

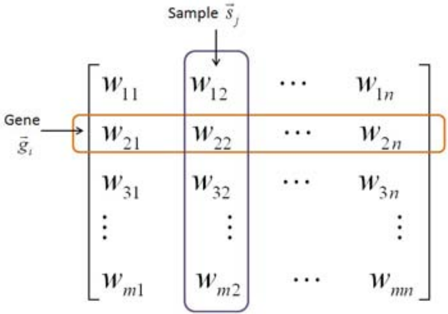
1. Introduction
   - Research Background and Motivations
   - Previous Work
2. Cooperative and Penalized Competitive Learning
   - Cooperation and Penalization Mechanisms
   - CPCL Algorithm
3. Experiments
   - Evaluation Criteria
   - Performance on UCI Data Set
   - Performance on Gene Expression Data
4. Conclusion

Outline

1. Introduction
   - Research Background and Motivations
   - Previous Work
2. Cooperative and Penalized Competitive Learning
   - Cooperation and Penalization Mechanisms
   - CPCL Algorithm
3. Experiments
   - Evaluation Criteria
   - Performance on UCI Data Set
   - Performance on Gene Expression Data
4. Conclusion

Gene Expression Data

Generally, a gene expression data set can be represented by a real-valued expression matrix $M = [w_{ij}]_{m\times n}$.



$$\begin{array}{cccccc}
 & & \text{Sample } \vec{s}_j & & & \\
 & w_{11} & w_{12} & \cdots & w_{1n} & \\
\text{Gene } \vec{g}_i & w_{21} & w_{22} & \cdots & w_{2n} & \\
 & w_{31} & w_{32} & \cdots & w_{3n} & \\
 & \vdots & \vdots & & \vdots & \\
 & w_{m1} & w_{m2} & \cdots & w_{mn} &
\end{array}$$

w_{ij}: measured expression level of gene i in sample j.

Clustering Analysis of Gene Expression Data

Clustering analysis is very helpful to understand *gene function*, *gene regulation*, *cellular processes*, and *subtypes of cells*.

For example:

- Coexpressed genes can be clustered together with similar *cellular functions*;
- Coexpressed genes in the same cluster are likely to be involved in the same *cellular processes*;
- A strong correlation of expression patterns between coexpressed genes indicates *coregulation*;
- Clustering different samples based on the expression profiles may reveal *subcell types*.

Categories of Gene Expression Data Clustering

For gene expression data, it is meaningful to cluster both genes and samples (Jiang et al., TKDE'2004).

- *Gene-based clustering*
  - Genes $\longrightarrow$ Objects; Samples $\longrightarrow$ Features
  - Coexpressed genes can be grouped in clusters
- *Sample-based clustering*
  - Samples $\longrightarrow$ Objects; Genes $\longrightarrow$ Features
  - Each group may correspond to some macroscopic phenotype
- *Subspace clustering*
  - Genes and samples are treated symmetrically
  - Capture clusters formed by a subset of genes across a subset of samples

The three categories of clustering analysis face different challenges and therefore different computational strategies should be adopted.

Gene-based Clustering

Some conventional clustering algorithms can be utilized, such as k-means, SOM, hierarchical clustering, and model-based clustering.

Challenges:

- The clustering algorithm should depend as little as possible on prior knowledge.

  - For example, a clustering algorithm which can accurately estimate the number of clusters will be more favored.
- Gene expression data often contains a huge amount of noise.
- Clusters of gene expression data may be highly intersected.
- Sometimes, graphical representation of the cluster structure is also needed.

Gene-based Clustering

Some conventional clustering algorithms can be utilized, such as k-means, SOM, hierarchical clustering, and model-based clustering.

Challenges:

- The clustering algorithm should depend as little as possible on prior knowledge.

  - For example, a clustering algorithm which can accurately estimate the number of clusters will be more favored.

↓

♣ What we focus on.

♣ *Objective:* exploring a novel learning model which can automatically estimate cluster number during clustering analysis.

Previous Work

Previous work on cluster number estimation can be grouped into two lines:

1. Conduct clustering with traditional algorithms and choose the number of clusters based on some statistic criteria.

   - *e.g., X-means (Pelleg and Moore, ICML'2000) and G-means (Hamerly and Elkan, NIPS'2003)*

2. Explore new clustering algorithms which can conduct clustering analysis without knowing the true number of clusters.

   - Non-center-based algorithms

     - *e.g., Affinity Propagation method (Frey and Dueck, Science'2007), Data Spectroscopic clustering (Shi et al., AS'2009), and CSPV algorithm (Lu and Wan, PR'2012)*

   - Center-based algorithms

     - *e.g., RPCCL (Cheung, TKDE'2005), DSRPCL (Ma and Wang, TSMC-B'2006), and CoRe (Bacciu and Starita, TNN'2008)*

Outline

1 Introduction

- Research Background and Motivations
- Previous Work

2 Cooperative and Penalized Competitive Learning

- Cooperation and Penalization Mechanisms
- CPCL Algorithm

3 Experiments

- Evaluation Criteria
- Performance on UCI Data Set
- Performance on Gene Expression Data

4 Conclusion

Definition of the Winner

- Suppose N inputs, $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N$, come from k^* unknown clusters, and k $(k \geq k^*)$ seed points $\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_k$ are randomly initialized.
- Given an input $\mathbf{x}_t$ each time, the winner among k seed points is determined by

$$I(j|\mathbf{x}_t) = \begin{cases} 1, \text{ if } j = c = \arg\min_{1 \leq i \leq k} \gamma_i ||\mathbf{x}_t - \mathbf{m}_i||^2, \\ 0, \text{ otherwise}, \end{cases} \qquad (1)$$

with the relative winning frequency γ_i of $\mathbf{m}_i$ defined as

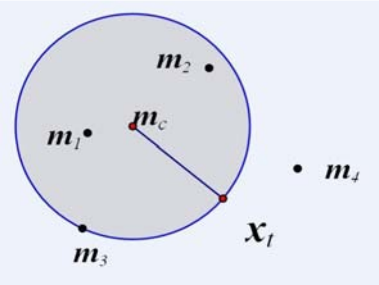
$$\gamma_i = \frac{n_i}{\sum_{j=1}^{k} n_j}, \qquad (2)$$

where n_i is the winning times of $\mathbf{m}_i$ in the past.

Territory of the Winner

Definition 1

The area centered at the winner $\mathbf{m}_c$ with the radius $||\mathbf{m}_c - \mathbf{x}_t||$ is regarded as the territory of $\mathbf{m}_c$.



Any other seed points which have intruded into this territory will either cooperate with the winner or be penalized by it.

Reliability of the Winning Seed Point

- In social life, people always prefer to cooperate with the person who has higher reliability.
- Inspired by this phenomenon, we assign a confidence coefficient, denoted as E_c ($E_c \in [0, 1]$), to the winner $\mathbf{m}_c$ to measure its reliability.
- Since more successful experience usually results in higher reliability, the confidence coefficient E_c of $\mathbf{m}_c$ can be given by

  $$E_c = \min(1, \eta \cdot n_c). \tag{3}$$

  Where η is a pre-specified small positive learning rate and n_c denotes the winning times of $\mathbf{m}_c$ in the past.

Determining the Cooperating Team

- The number of cooperators owned by a winner is determined by its confidence coefficient E_c.
- Suppose there are q seed points which have intruded into the winner's territory, then the number of cooperators q_w can be calculated by

$$q_w = \lfloor q \cdot E_c \rfloor = \lfloor q \cdot \min(1, \eta \cdot n_c) \rfloor , \tag{4}$$

  where $\lfloor \cdot \rfloor$ denotes the floor function.
- In this learning approach, the competitor nearest to the winner has the priority to be a cooperator.

Penalized Seed Points

- All of the other non-cooperating intruders in the winner's territory will be penalized.

- The number of penalized seed points, denoted as q_p, is calculated by

$$\begin{aligned} q_p =& q - q_u \\ =& q - \lfloor q \cdot \min(1, \eta \cdot n_c) \rfloor \\ =& \lceil q \cdot \max(0, 1 - \eta \cdot n_c) \rceil \,, \end{aligned} \tag{5}$$

  where $\lceil \cdot \rceil$ means the ceiling function.

- At the initial stage, the winning times of each seed point are very few, then we have $q_u = 0$ and $q_p = q$.

Updating Formula

- After determining the cooperating team and penalized team at time t, each cooperator, denoted as $\mathbf{m}_u$, will be updated by

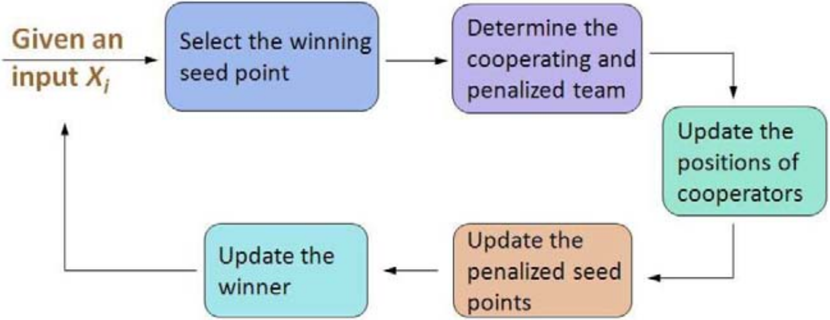
$$\mathbf{m}_u^{(t)} = \mathbf{m}_u^{(t-1)} + \eta \frac{\left\|\mathbf{m}_c^{(t-1)} - \mathbf{x}_t\right\|}{\max\left(\left\|\mathbf{m}_c^{(t-1)} - \mathbf{x}_t\right\|, \left\|\mathbf{m}_u^{(t-1)} - \mathbf{x}_t\right\|\right)} (\mathbf{x}_t - \mathbf{m}_u^{(t-1)}). \qquad (6)$$

- The other penalized seed points in the winner's territory, denoted as $\mathbf{m}_p$, will be penalized by

$$\mathbf{m}_p^{(t)} = \mathbf{m}_p^{(t-1)} - \eta \frac{\left\|\mathbf{m}_c^{(t-1)} - \mathbf{x}_t\right\|}{\left\|\mathbf{m}_p^{(t-1)} - \mathbf{x}_t\right\|} (\mathbf{x}_t - \mathbf{m}_p^{(t-1)}). \qquad (7)$$

General Process of the Competitive Learning

During each learning epoch:


Given an input X_i
Select the winning seed point
Determine the cooperating and penalized team
Update the positions of cooperators
Update the penalized seed points
Update the winner

CPCL Algorithm

Step1: Initialize k seed points. Set $n_j^{(0)} = 1$ with $j = 1, 2, \ldots, k$, and $t = 1$.

Step2: Determine the winner unit $\mathbf{m}_c^{(t-1)}$. Let S_c be the set of seed points fallen into the territory of $\mathbf{m}_c^{(t-1)}$. That is, let $S_c = \emptyset$, and then we span S_c by

$$S_c = S_c \cup \left\{ \mathbf{m}_j^{(t-1)} | \left\| \mathbf{m}_c^{(t-1)} - \mathbf{m}_j^{(t-1)} \right\| \le \left\| \mathbf{m}_c^{(t-1)} - \mathbf{x}_t \right\| \right\}, j \neq c. \qquad (8)$$

Step4: Sort the units in S_c based on the distance between each unit to the winner $\mathbf{m}_c^{(t-1)}$. We denote the sorted units as: ${\mathbf{m}_1}'^{(t-1)}$, ${\mathbf{m}_2}'^{(t-1)}$, $\ldots$, ${\mathbf{m}_q}'^{(t-1)}$, with

$$\left\| {\mathbf{m}_1}'^{(t-1)} - \mathbf{m}_c^{(t-1)} \right\| \le \left\| {\mathbf{m}_2}'^{(t-1)} - \mathbf{m}_c^{(t-1)} \right\| \le \cdots \le \left\| {\mathbf{m}_q}'^{(t-1)} - \mathbf{m}_c^{(t-1)} \right\|. \qquad (9)$$

Step5: Select a subset S_u of S_c to form a cooperating team of $\mathbf{m}_c^{(t-1)}$, where

$$S_u = \left\{ {\mathbf{m}_1}'^{(t-1)}, {\mathbf{m}_2}'^{(t-1)}, \ldots, {\mathbf{m}_{q_u}}'^{(t-1)} \right\}$$

and q_u is calculated by Eq. (4). Then update all members in S_u by Eq. (6).

Step6: Let $S_p = S_c - S_u$, then we penalize all seed points in S_p by Eq. (7).

Step7: Update the winner $\mathbf{m}_c$ by

$$\mathbf{m}_c^{(t)} = \mathbf{m}_c^{(t-1)} + \eta \cdot (\mathbf{x}_t - \mathbf{m}_c^{(t-1)}). \qquad (10)$$

Step8: Update n_c by $n_c^{(t)} = n_c^{(t-1)} + 1$, and increase t by 1.

Outline

1 Introduction
- Research Background and Motivations
- Previous Work

2 Cooperative and Penalized Competitive Learning
- Cooperation and Penalization Mechanisms
- CPCL Algorithm

3 Experiments
- Evaluation Criteria
- Performance on UCI Data Set
- Performance on Gene Expression Data

4 Conclusion

Evaluation Criteria

- *Partition Quality (PQ)*:

$$PQ = \begin{cases} \frac{\sum_{i=1}^{k^*}\sum_{j=1}^{k'}[p(i,j)^2 \cdot (p(i,j)/p(j))]}{\sum_{i=1}^{k^*} p(i)^2}, & \text{if } k' > 1, \\ 0, & \text{otherwise}, \end{cases}$$

where k^* is the true number of classes and k' is the cluster number learned by the algorithm. The term $p(i,j)$ calculates the frequency-based probability that a data point is labeled i by the true label and labeled j by the obtained label.

- *Rand Index (RI)*:

$$RI = \frac{TP + TN}{TP + FP + FN + TN}.$$

Utilized Data Sets

- *Seeds*
  This data set has 210 instances with 7 attributes. All the instances are distributed into three different varieties of wheat: Kama, Rosa and Canadian.

- *Wisconsin Diagnostic Breast Cancer (WDBC)*
  This data set contains 569 instances described by 30 features. 357 instances of them have the diagnosis of benign while the other 212 samples are regarded as malignant.

Clustering Results on Seeds Data

Table 1: Clustering Results on the Seeds Data Set ($k^* = 3$)

k	Methods	#Clusters	PQ	RI	Time (#Epochs)
$-$	DaSpec	2	0.5968	0.7375	0.6271 (1)
$-$	CSPV	2	0.5456	0.7149	0.11 (1)
4	DSRPCL1	4±0.0	0.6623	0.8628	0.36 (94.85)
	DSRPCL2	3.95±0.22	0.6377	0.8565	0.20 (47.7)
	RPCCL	**2.95**±0.22	0.6849	0.8499	0.91 (100)
	CoRe	2.1±0.31	0.5794	0.7593	1.04 (19.5)
	CCCL	2.85±0.81	0.6273	0.8095	0.71 (77.35)
	CPCL	3.25±0.55	**0.6922**	**0.8635**	0.56 (49.5)
10	DSRPCL1	8.25±1.12	0.3146	0.7673	1.61 (187.2)
	DSRPCL2	10±0.0	0.2748	0.7546	0.65 (84.45)
	RPCCL	8.85±1.18	0.3718	0.7763	9.06 (500)
	CoRe	2.45±0.51	0.6385	0.8028	2.13 (28.75)
	CCCL	3.5±0.82	0.6536	0.8442	3.68 (189.5)
	CPCL	**3.25**±0.58	**0.7302**	**0.8840**	2.55 (110.9)
20	DSRPCL1	17.05±1.57	0.2020	0.7311	4.87 (296.15)
	DSRPCL2	19.95±0.22	0.1620	0.7182	2.07 (163.1)
	RPCCL	18.3±1.03	0.1783	0.7200	33.72 (1000)
	CoRe	2.7±0.47	0.6738	0.8342	3.69 (39.7)
	CCCL	3.7±0.92	0.6437	0.8329	13.71 (368.5)
	CPCL	**3.1**±0.45	**0.7332**	**0.8771**	7.33 (168.3)

Clustering Results on WDBC Data

Table 2: Clustering Results on the WDBC Data Set ($k^* = 2$)

k	Methods	#Clusters	PQ	RI	Time (#Epochs)
$-$	DaSpec	1	0	0.5316	5.18 (1)
$-$	CSPV	2	0.5602	0.5335	0.7493 (1)
3	DSRPCL1	3±0.0	0.6248	0.7553	0.47 (55.5)
	DSRPCL2	3±0.0	0.6194	0.7521	0.15 (14.25)
	RPCCL	1.85±0.36	0.4781	0.5553	2.11 (100)
	CoRe	2.15±0.93	0.2664	0.5964	8.03 (26.2)
	CCCL	2.15±0.36	0.7573	0.8321	0.72 (23.5)
	CPCL	**2±0.0**	**0.7725**	**0.8415**	0.69 (20.4)
10	DSRPCL1	9.7±0.47	0.2111	0.5774	5.46 (225.8)
	DSRPCL2	9.9±0.31	0.2013	0.5723	1.35 (62.95)
	RPCCL	5.9±2.05	0.5136	0.6984	26.29 (500)
	CoRe	2.6±1.31	0.2931	0.5719	23.51 (61.20)
	CCCL	1.95±0.22	0.7215	0.8177	3.02 (47.15)
	CPCL	**2±0.0**	**0.7551**	**0.8298**	2.63 (39.35)
20	DSRPCL1	19.95±0.22	0.1228	0.5311	20.99 (457.3)
	DSRPCL2	20±0.0	0.1098	0.5243	3.3992 (95.6)
	RPCCL	15.25±1.86	0.1925	0.5629	96.67 (1000)
	CoRe	3.1±0.91	0.3290	0.6126	49.51 (107.15)
	CCCL	1.85±0.36	0.7267	0.8211	9.62 (82.05)
	CPCL	**2.05**±0.22	**0.7582**	**0.8306**	7.97 (63.7)

Gene Expression Data Set

This data was published by Cho et al. in 1998. The data set we used was comprised of 384 genes which had expression levels peaking at different time points corresponding to the five phases of the cell cycle.

Number of clusters:

We arbitrarily initialized 20 seed points in the running of the CPCL. After 10 trials, the average and most frequent number of clusters obtained by CPCL are **4.9** and **5**, respectively. That is, the true cluster number has been identified.

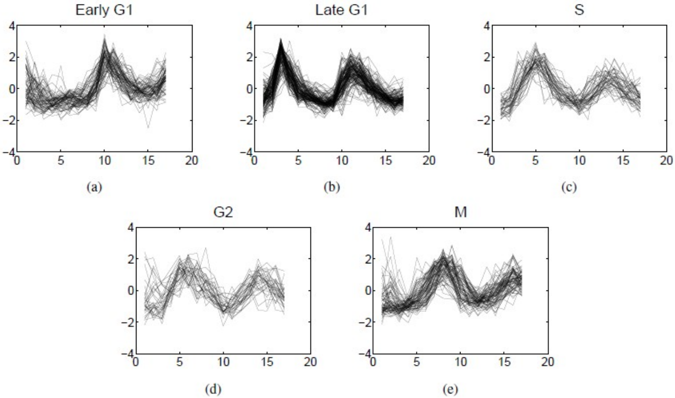
Clustering Errors on the Gene Expression Data

Table 3: Clustering Errors of Different Methods

	Methods							
Division phase	CPCL		M1		M2		M3	
	FP	FN	FP	FN	FP	FN	FP	FN
Early G1 (67 genes)	25	16	50	12	21	21	38	10
Late G1 (135 genes)	37	23	28	40	24	35	43	10
S (75 genes)	18	47	33	49	37	36	72	18
G2 (52 genes)	11	30	28	41	18	29	46	5
M (55 genes)	28	3	38	42	19	8	47	2
Summation	119	119	177	184	119	129	246	45
Total Error (FP + FN)	**238**		361		248		291	

M1: EM algorithm based on BIC (Yeung et al., Bioinformatics'2001);
M2: supervised clustering method (Qu and Xu, Bioinformatics'2004);
M3: support vector machines algorithm (Brown et al., NAS'2000).

The Five Groups of Genes Formed by CPCL Algorithm



Outline

1. Introduction
   - Research Background and Motivations
   - Previous Work
2. Cooperative and Penalized Competitive Learning
   - Cooperation and Penalization Mechanisms
   - CPCL Algorithm
3. Experiments
   - Evaluation Criteria
   - Performance on UCI Data Set
   - Performance on Gene Expression Data
4. Conclusion

Conclusion

- To conduct clustering without knowing cluster number, a novel competitive learning method has been studied.
- The presented algorithm performs cooperation and penalization mechanisms simultaneously in a single competitive learning process.
- This new algorithm features the good estimate of cluster centers and the robust performance against the initialization of seed points.

Acknowledgement

We would like to thank the organizers of The First Inter-university Symposium on Field Based Design for their great support.

References

1. D. Bacciu and A. Starita, “Competitive repetition suppression (core) clustering: A biologically inspired learning model with application to robust clustering,” *IEEE Transactions on Neural Networks*, vol. 19, no. 11, pp. 1922–1941, 2008.
2. M. P. S. Brown, W. N. Grundy, N. C. D. Lin, C. W. Sugnet, T. S. Furey, M. Ares, and D. Haussler, “Knowledge-based analysis of micro-array gene expression data by using support vector machines,” in *Proceedings of National Academy of Sciences of the USA*, vol. 97, 2000, pp. 262–267.
3. Y. M. Cheung, “On rival penalization controlled competitive learning for clustering with automatic cluster number selection,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1583–1588, 2005.
4. B. J. Frey and D. Dueck, “Clustering by passing messages between data points,” *Science*, vol. 315, no. 5814, pp. 972–976, 2007.
5. G. Hamerly and C. Elkan, “Learning the k in k-means,” in *Proceedings of the 17th Annual Conference on Neural Information Processing Systems (NIPS)*, 2003, pp. 281–288.
6. D. Jiang, C. Tang, and A. Zhang, “Cluster analysis for gene expression data: a survey,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 11, pp. 1370–1386, 2004.
7. T. Li, W. J. Pei, S. P. Wang, and Y. M. Cheung, “Cooperation controlled competitive learning approach for data clustering,” in *Proceedings of International Conference on Computational Intelligence and Security*, 2008, pp. 24–29.
8. Y. Lu and Y. Wan, “Clustering by sorting potential values (cspv): a novel potential-based clustering method,” *Pattern Recognition*, vol. 45, no. 9, pp. 3512–3522, 2012.
9. J. Ma and T. Wang, “A cost-function approach to rival penalized competitive learning (rpcl),” *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 36, no. 4, pp. 722–737, 2006.
10. D. Pelleg and A. Moore, “X-means: extending k-means with efficient estimation of the number of clusters,” in *Proceedings of the 17th International Conference on Machine Learning*, 2000, pp. 727–734.
11. Y. Qu and S. Xu, “Supervised cluster analysis for microarray data based on multivariate gaussian mixture,” *Bioinformatics*, vol. 20, pp. 1905–1913, 2004.
12. T. Shi, M. Belkin, and B. Yu, “Data spectroscopy: Eigenspaces of convolution operators and clustering,” *Annals of Statistics*, vol. 37, no. 6B, pp. 3960–3984, 2009.
13. K. Y. Yeung, C. Fraley, A. Murua, A. E. Raftery, and W. L. Ruzzo, “Model-based clustering and data transformations for gene expression data,” *Bioinformatics*, vol. 17, no. 10, pp. 977–987, 2001.

Thank You!