A NEW DIMENSION REDUCTION METHOD?
—— FEATURE SELECTION IN GAUSSIAN MIXTURE CLUSTERING

YIU-MING CHEUNG （张晓明）

DEPARTMENT OF COMPUTER SCIENCE
HONG KONG BAPTIST UNIVERSITY, HONG KONG

香港浸會大學
HONG KONG BAPTIST UNIVERSITY

ABOUT HKBU

ABOUT HK BAPTIST UNIVERSITY
关于香港浸会大学

- Established in **1956** – the 2nd longest history in HK
  于1956年成立-是香港第二间历史悠久的大学。
- Funded by the Government
  由香港政府资助
- 35 undergraduate and 58 postgraduate programmes
  共有35个本科专业和58个研究生专业
- About **8,500 students**
  约有8,500个学生

ABOUT HK BAPTIST UNIVERSITY
关于香港浸会大学

University 大学	2010 World Ranking 2010年世界大学排名
Peking University 北京大学	37
University of Science and Technology of China 中国科技大学	49
Tsinghua University 清华大学	58
Hong Kong Baptist University **香港浸会大学**	**111**
Nanjing University 南京大学	120
Sun Yat-Sen University 中山大学	171
Zhejiang University 浙江大学	197

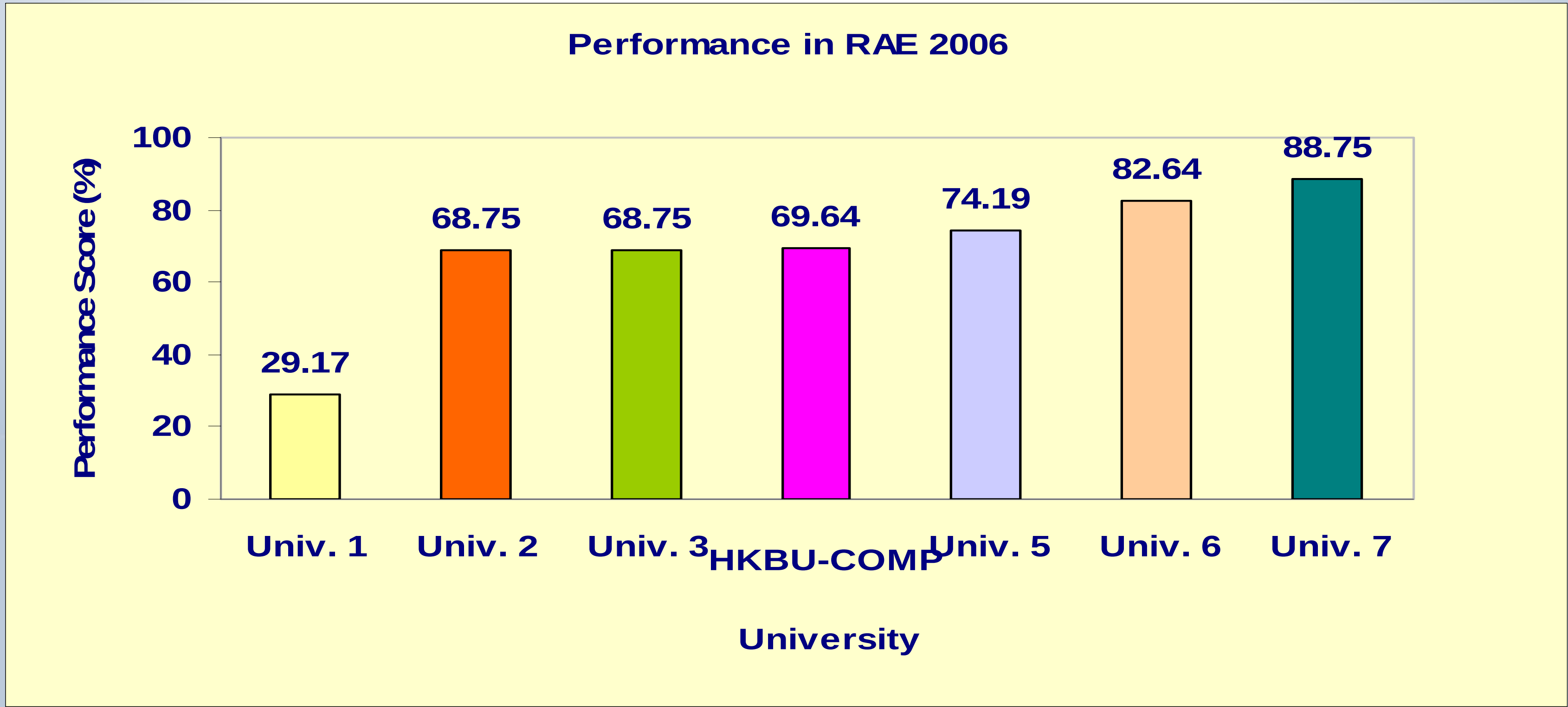
Reference: Times Higher Education – 2010 World University Ranking
参考资料：〈英国泰晤士报高等报育〉２０１０年世界大学排名榜

University 大学	2011 Asia Ranking 2011年亚洲大学排名
Peking University 北京大学	3
Tsinghua University 清华大学	8
University of Science and Technology of China 中国科技大学	19
Fudan University 复旦大学	25
Nanjing University 南京大学	32
Hong Kong Baptist University **香港浸会大学**	**36**
Sun Yat-Sen University 中山大学	39
Shanghai Jiaotong University 上海交通大学	46
Zhejiang University 浙江大学	49

Reference: Times Higher Education – 2011 World University Ranking
参考资料:〈英国泰晤士报高等报育〉２０１１年世界大学排名榜

ABOUT COMPUTER SCIENCE DEPARTMENT, HKBU

- In Research Assessment Exercise conducted by UGC of the HKSAR Government in 2006, ranking of the local CS/IT Departments:



根据香港政府2006年对香港本地大学所做的研究评估报告

MOTIVATION

High-dimensional input data is common, e.g.

a feature

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Input vector
with **n** features

n >> m

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

m features only

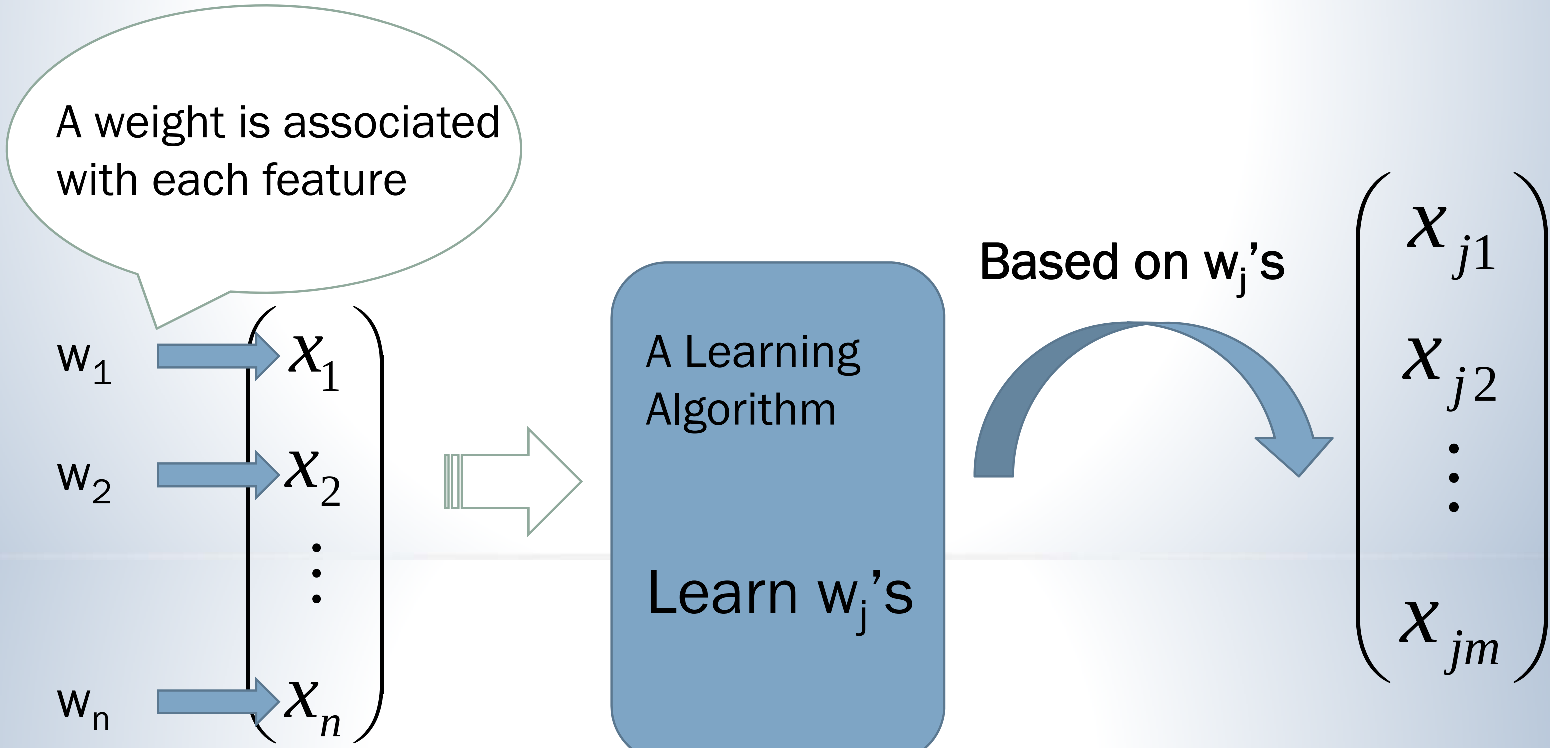
Two problems in existing dimension reduction methods:

1.How many dimensions will the input vectors be reduced, i.e. how to select the value of m?

2.It is hard or even impossible to interpret the physical meaning of y_j's.

MOTIVATION (CONT'D 1)

- In our method:



where $w_i \in [0, 1]$

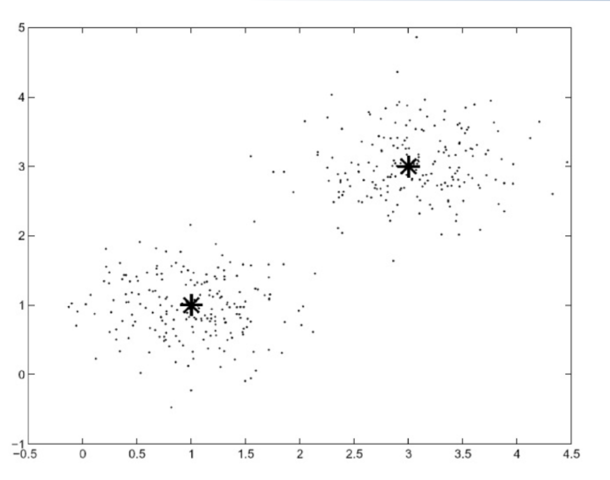
IN THIS TALK

- Focus on Density Mixture Clustering (Gaussian Mixture in particlar)
  - Model:

$$p(x \mid \Theta^*) = \sum_{j=1}^{k^*} \alpha_j^* p(x \mid \theta_j^*)$$

with

$$\sum_{j=1}^{k^*} \alpha_j^* = 1, \forall 1 \leq j \leq k^*, \alpha_j^* > 0.$$



  - Data Classification:

$$h(j \mid x_t, \Theta^*) = \frac{\alpha_j^* p(x_t \mid \theta_j^*)}{\sum_{r=1}^{k^*} \alpha_r^* p(x_t \mid \theta_j^*)}, 1 \leq j \leq k^*.$$

- Two Learning Problems:
  - Problem 1: Estimate the model parameters $\Theta^* = \{\alpha_j^*, \theta_j^*\}_{j=1}^{k^*}$
  - Problem 2: Determine the number of mixture components, i.e. the number of clusters

IN THIS TALK (CONT'D 1)

- Problem 1:
  - Expectation-Maximization (EM) Algorithm provides a general solution of model parameter estimation;
  - An adaptive EM Algorithm (given an estimate k of k*):
    - E-Step:

      Fixing $\Theta^{(old)}$ and calculate

$$h(j \mid x_t, \Theta^{(old)}) = \frac{\alpha_j^{(old)} p(x_t \mid \theta_j^{(old)})}{\sum_{r=1}^{k} \alpha_r^{(old)} p(x_t \mid \theta_j^{(old)})} \qquad j = 1, 2, \ldots, k$$

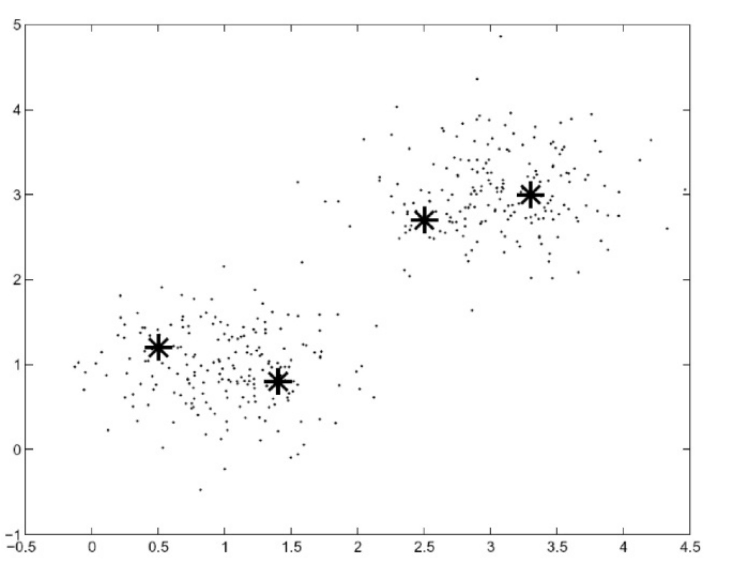
    - M-Step Fixing $h(j \mid x_t, \Theta^{(old)})s$, we update Θ using gradient ascent method:

$$\Theta^{new} = \Theta^{(old)} + \eta \frac{\partial \ell(\Theta; x_t)}{\partial \Theta} \Big|_{\Theta^{(old)}}$$

DRAWBACK OF THE EM ALGORITHM

- Scenario: Traditional Expectation-Maximization (EM) algorithm leads to a poor parameter estimation when the number k of densities in a mixture is mis-specified;

Drawback: The EM algorithm cannot determine the number of components automatically.



IN THIS TALK (CONT'D 2)

Scenario:

- Common to cluster high-dimensional data, e.g. in Microarray data analysis, image processing, pattern recognition.
- Irrelevant features could hinder the detection of cluster structures. click
- Among the relevant features, some may be redundant. click

- Problem 3: To find the minimal feature subset that best represents the partition of interest via learning the associated weights of the features
- Difficulties
  - Absence of the ground-truth class labels of the training data to guide the feature selection;
  - True number of clusters is unknown a priori;
  - Feature subset and clusters are inter-related. click

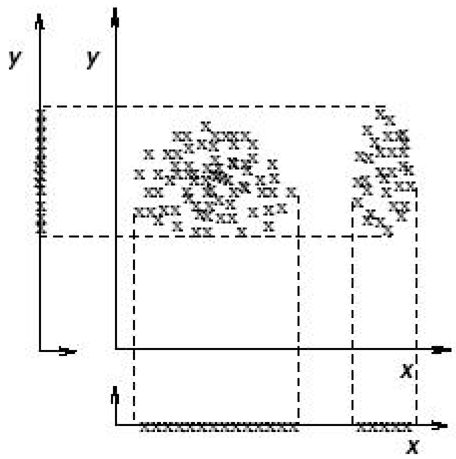
THE PROPOSED APPROACH

- Develop Rival Penalized EM (RPEM) Algorithm within the learning framework of Maximum Weighted Likelihood Approach
  - To solve Problem 1 and Problem 2
- Present an unsupervised feature selection scheme
  - To solve Problem 3
- Develop an Iterative Feature Selection and Clustering Algorithm
  - which is an integration of RPEM and Unsupervised Feature Selection Scheme

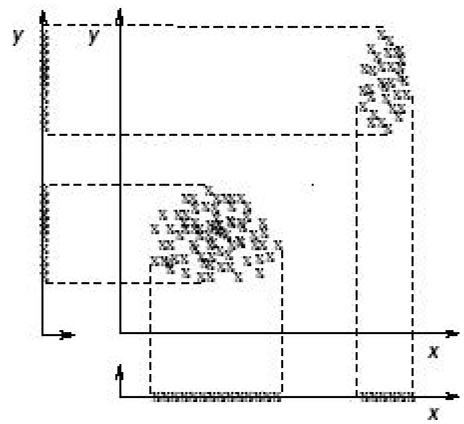
  - Highlights:

Simultaneous learning of the three tasks:

- Problem 1: Model parameter estimation;
- Problem 2: Select the number of components (i.e. the number of clusters);
- Problem 3: The learning of the associated feature weights w_j's.



y
y
x
x



y
y
x
x

OUTLINE

- Introduction
  - The existing unsupervised feature selection methods
- The RPEM Algorithm
- Unsupervised Feature Selection Schemes
- The Iterative Feature Selection and Clustering Algorithm
- Experimental Results
- Conclusion

INTRODUCTION: THREE KINDS OF FEATURE SELECTION APPROACHES

- Filter Approach (e.g. see [Dash et al. 2002, Miltra et al.2002])
  - Perform feature selection prior to the clustering algorithm.
- Wrapper Approach (e.g. see [Dy and Brodley 2000 &2005])
  - For each feature subset candidate, evaluate it by wrapping around the clustering algorithm.
- Embedded Approach (e.g. see [Law et al. 2002, Constantinopoulos et al. 2006])
  - Optimize the two tasks in a single optimization paradigm;
  - Assume that the pdf of the irrelevant features is Gaussian (Sensitive).
- Our approach
  - Iterate between clustering and feature selection;
  - Robust against the pdf of the irrelevant features;
  - Perform not only the relevance analysis, but also the redundancy analysis to gradually shrink the search space.

OUTLINE

- Introduction √
  - The existing unsupervised feature selection methods
- The RPEM Algorithm
- Unsupervised Feature Selection Schemes
- The Iterative Feature Selection and Clustering Algorithm
- Experimental Results
- Conclusion

MAXIMUM WEIGHTED LIKELIHOOD AND RPEM ALGORITHM

- A general MWL learning framework
- The ML estimate of Θ^* can be obtained via maximizing the cost function:

$$l(\Theta) = \int \ln p(x \mid \Theta) dF(x)$$

with

$$p(x \mid \Theta) = \sum_{j=1}^{k} \alpha_j p(x \mid \theta_j), \sum_{j=1}^{k} \alpha_j, \forall 1 \le j \le k, \alpha_j > 0$$

where $k \ge k^*$. The above equation can be further represented as

$$l(\Theta) = \int \sum_{j=1}^{k} g(j \mid x, \Theta) \ln p(x \mid \Theta) dF(x)$$

where $g(j \mid x, \Theta)$ is the designable weight that satisfying

$$\sum_{j=1}^{k} g(j \mid x, \Theta) = 1$$

By Baye's formula

$$h(j \mid x, \Theta) = \frac{\alpha_j p(x \mid \theta_j)}{p(x \mid \Theta)}$$

Subsequently, we have

$$p(x \mid \Theta) = \frac{\alpha_j p(x \mid \theta_j)}{h(j \mid x, \Theta)}$$

Consequently, we have

$$l(\Theta) = \int \sum_{j=1}^{k} g(j \mid x, \Theta) \ln\left[\alpha_j p(x \mid \theta_j)\right] dF(x) - \int \sum_{j=1}^{k} g(j \mid x, \Theta) \ln h(j \mid x, \Theta) dF(x) \qquad (1)$$

Theorem 1: Suppose $p(x \mid \Theta)$ is an identifiable model with respect to Θ.
Eq.(1) reaches the global maximum if and only if $\Theta = \Theta^*$.
Particularly, as N is large enough, the empirical MWL cost function is then:

$$Q(X_N; \Theta) = \frac{1}{N} \sum_{t=1}^{N} \sum_{j=1}^{k} \mathrm{g}(j \mid x_t, \Theta) \ln[\alpha_j p(x_t \mid \Theta_j)] - \frac{1}{N} \sum_{t=1}^{N} \sum_{j=1}^{k} \mathrm{g}(j \mid x_t, \Theta) \ln h(j \mid x_t, \Theta)$$

where

$$\forall j, \mathrm{g}(j \mid x_t, \Theta) = 0 \quad if \quad h(j \mid x_t, \Theta) = 0$$

Some choices of $g(j \mid x_t, \Theta)$:

- If $g(j \mid x_t, \Theta) = h(j \mid x_t, \Theta)$
  - **Equal to the Kullback-Leibler divergence function derived from Ying-Yang Machine with the backward architecture.**

- If $g(j \mid x_t, \Theta) = I(j \mid x_t, \Theta)$

$$I(j \mid x_t, \Theta) = \begin{cases} 1 \quad if \quad j = c = \arg\max_{1 \le r \le k} h(j \mid x_t, \Theta) \\ \qquad 0, \quad otherwise \end{cases}$$

  - Equal to the cost function of hard-cut EM.

- A specific design of $g(j \mid x_t, \Theta)$ herein:

$$g(j \mid x_t, \Theta) = 2\varphi(j \mid x_t, \Theta) - h(j \mid x_t, \Theta) \qquad (2)$$

where $\varphi(j \mid x_t, \Theta)$ is a special probability function named indicator function.

RIVAL PENALIZED EM ALGORITHM

By considering the specific weights defined above, the cost function becomes

$$Q(\Theta; X_N) = \frac{1}{N}\sum_{t=1}^{N} q_t(\Theta; x_t)$$

with

$$q_t(\Theta; x_t) = R_{t(\Theta; x_t)} + H_t(\Theta; x_t)$$

and

$$R_t(\Theta; x_t) = \sum_{j=1}^{k} [2\varphi(j \mid x_t, \Theta) - h(j \mid x_t, \Theta)] \ln[\alpha_j p(x_t \mid \theta_j)]$$

$$H_t(\Theta; x_t) = -\sum_{j=1}^{k} [2\varphi(j \mid x_t, \Theta) - h(j \mid x_t, \Theta)] \ln h(j \mid x_t, \Theta)$$

One choice of $\varphi(j \mid x_t, \Theta)$

$$\varphi(j \mid x_t, \Theta) = I(j \mid x_t, \Theta) = \begin{cases} 1 \quad if \quad j = c = \arg\max_{1 \le r \le k} h(j \mid x_t, \Theta) \\ \quad 0, \quad otherwise \end{cases}$$

Learn Θ via maximizing the cost function $Q(\Theta; X_N)$ adaptively:

- Step A.1

  Fixing $\Theta^{(old)}$, and calculate $h(j \mid x_t, \Theta^{(old)})$ and $\varphi(j \mid x_t, \Theta)$, as given an input x_t

- Step A.2

  Fixing $h(j \mid x_t, \Theta^{(old)})$ s, we update Θ using gradient ascent method.

$$\alpha_j = \frac{\exp(\beta_j)}{\sum_{r=1}^{k} \exp(\beta_r)} \qquad for \quad 1 \le j \le k$$

and update $\beta_j s$ directly instead of $\alpha_j s$. As a result,

$$\beta_c^{(new)} = \beta_c^{(old)} + \eta \frac{\partial q_t(\Theta; x_t)}{\partial \beta_c} \Big|_{\Theta^{(old)}}$$

$$\Theta_c^{(new)} = \Theta_c^{(old)} + \eta \frac{\partial q_t(\Theta; x_t)}{\partial \Theta_c} \Big|_{\Theta_c^{(old)}}$$

meanwhile

$$\beta_r^{(new)} = \beta_r^{(old)} + \eta \frac{\partial q_t(\Theta; x_t)}{\partial \beta_r} |_{\Theta^{(old)}}$$

$$\Theta_r^{(new)} = \Theta_r^{(old)} + \eta \frac{\partial q_t(\Theta; x_t)}{\partial \Theta_r} |_{\Theta_r^{(old)}}, (r \neq c)$$

where $c = \arg\max_{1 \le r \le k} h(j \mid x_t, \Theta).$

The above two steps are iteratively implemented for each input until Θ converges.

Remarks:

We have proved that the convergence of Θ is guaranteed.

DETAILED RPEM IN GAUSSIAN DENSITY MIXTURE MODEL

- Suppose the N inputs $\{x_t\}_{t=1}^{N}$ all iid distribution, and come from a Gaussian density mixture, i.e.,

$$p(x \mid \Theta) = \sum_{j=1}^{k} \alpha_j G(x_t m_j, \sum_j)$$

- Initialization

  Given a specific $k(k \geq k^*)$, we initialize Θ. Then, at each time step t, we implement the following two steps:

  Step B.1:

  Fixing $\Theta^{(old)}$, and calculate

$$h(j \mid x_t, \Theta^{(old)}) = \frac{\alpha_j^{(old)} G(x_t \mid m_j^{(old)}, \sum_j^{(old)})}{p(x_t \mid \Theta^{(old)})}$$

$$g(j \mid x_t, \Theta) = 2\varphi(j \mid x_t, \Theta) - h(j \mid x_{t,} \Theta), 1 \leq j \leq k$$

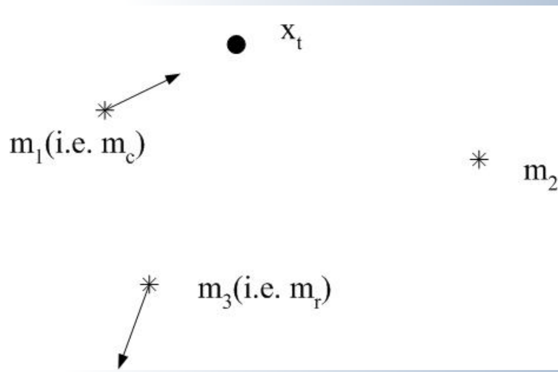
Step B.2:

Fixing $h(j \mid x_t, \Theta^{(old)})s$,we update Θ using gradient ascent method.

$$\beta_j^{(new)} = \beta_j^{(old)} + \eta[\mathrm{g}(j \mid x_t, \Theta^{(old)}) - \alpha_j^{old}]$$

$$m_j^{(new)} = m_j^{(old)} + \eta\,\mathrm{g}(j \mid x_t, \Theta^{(old)}) \sum_j^{-1(old)} (x_t - m_j^{(old)})$$

$$\sum_j^{-1(new)} = [1 + \eta\,\mathrm{g}(j \mid x_t, \Theta^{(old)})] \sum_j^{-1(old)} - \eta\,\mathrm{g}(j \mid x_t, \Theta^{(old)}) \mathrm{U}_{t,j}$$



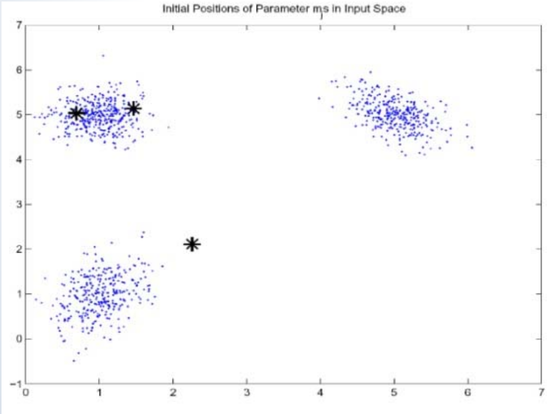
where

$$U_{t,j} = [\sum_j^{-1(old)} (x_t - m_j^{(old)})(x_t - m_j^{(old)})^T \sum_j^{-1(old)}].$$

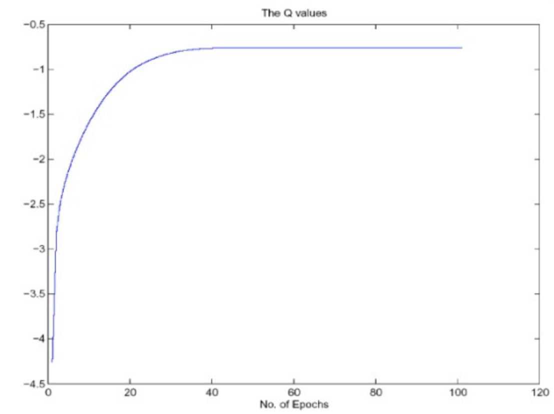
Note that, to simplify the computation $\sum_j^{-1}s$ update, we have updated $\sum_j^{-1}$ along the direction of $\sum_j^{-1} \frac{\partial q_t(\Theta; x_t)}{\partial \sum_j^{-1}} \sum_j^{-1}$

EXPERIMENTAL SIMULATION
EXPERIMENT I

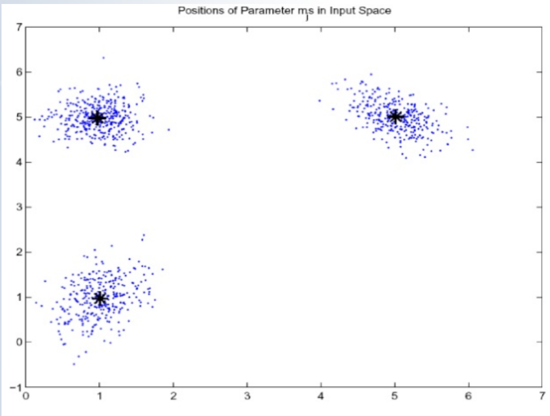
The number k of seed points is 3



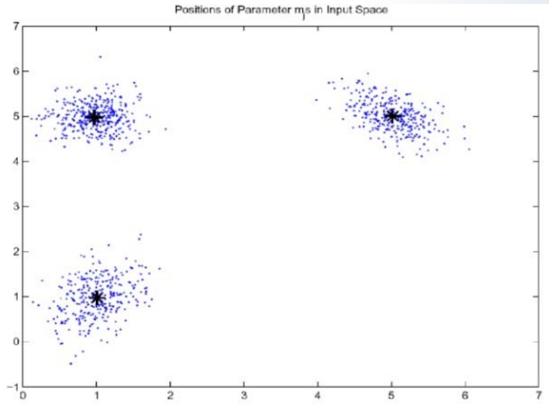
Initial random seed points



Change of Q

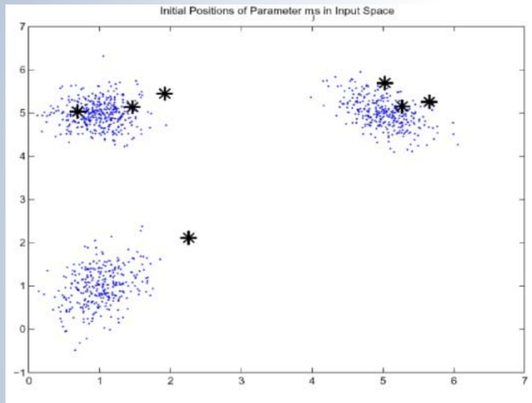


Results of RPEM

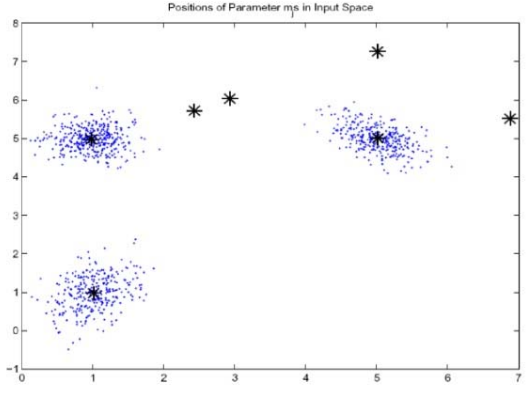


Results of EM

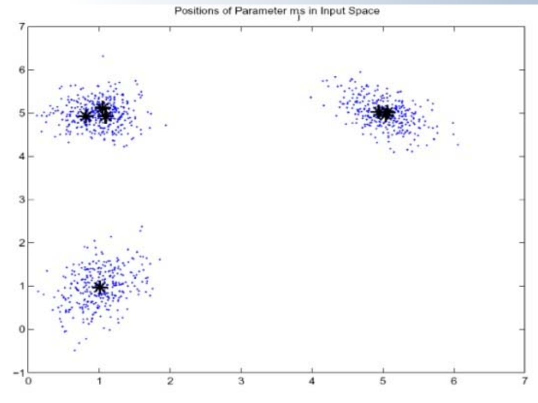
Suppose the number k of seed points is 7 rather than 3



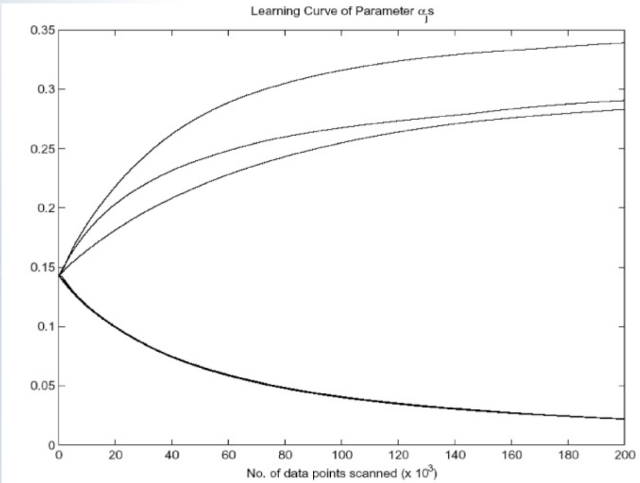
Initial seed points



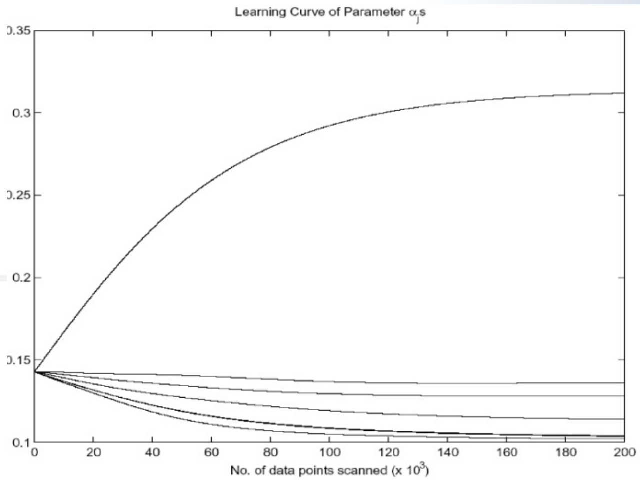
Results for RPEM



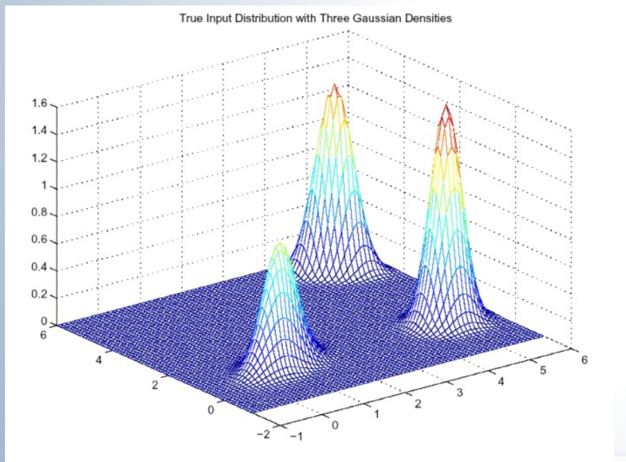
Results for EM

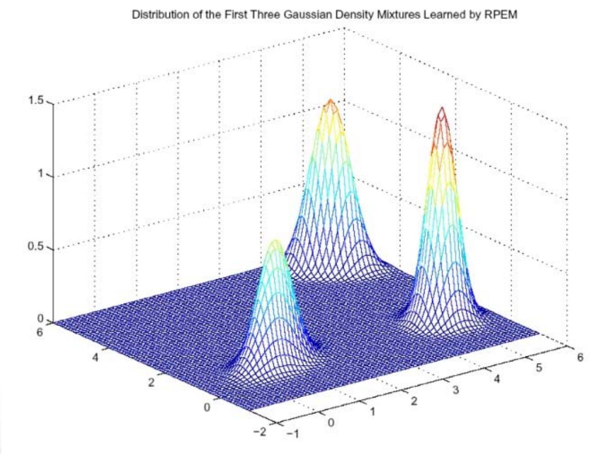


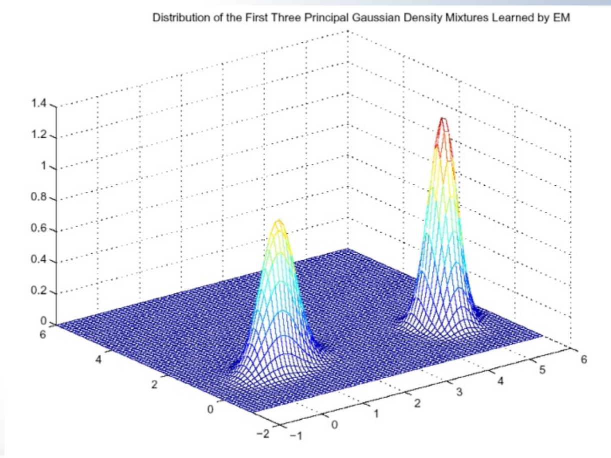
Learning curve for RPEM



Learning curve for EM







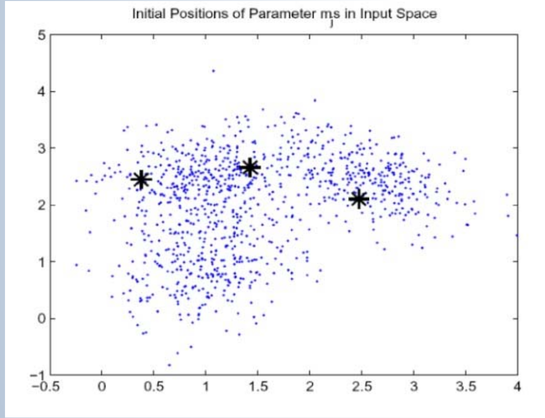
True distribution of components

Results for RPEM

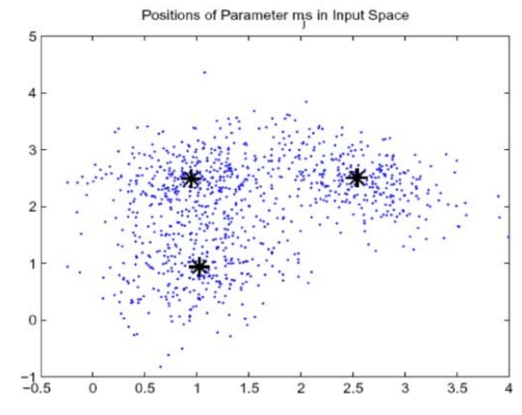
Results for EM

EXPERIMENT II

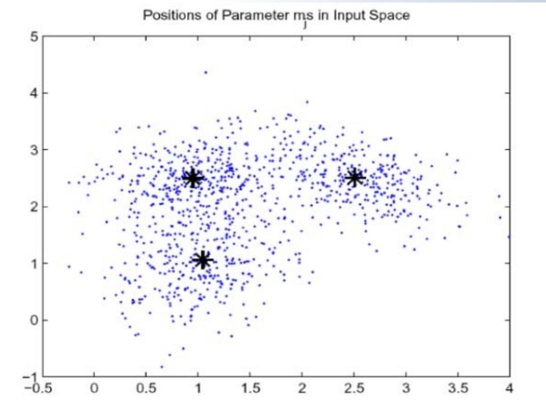
The data points are generated from the mixture Gaussian models, where the three clusters are overlapped.



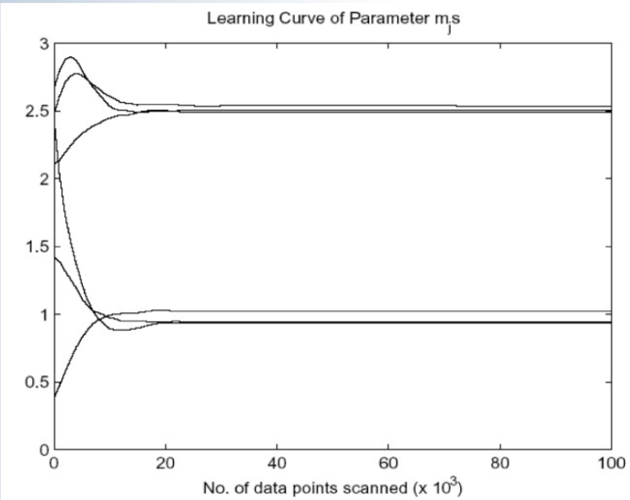
Initial seed points



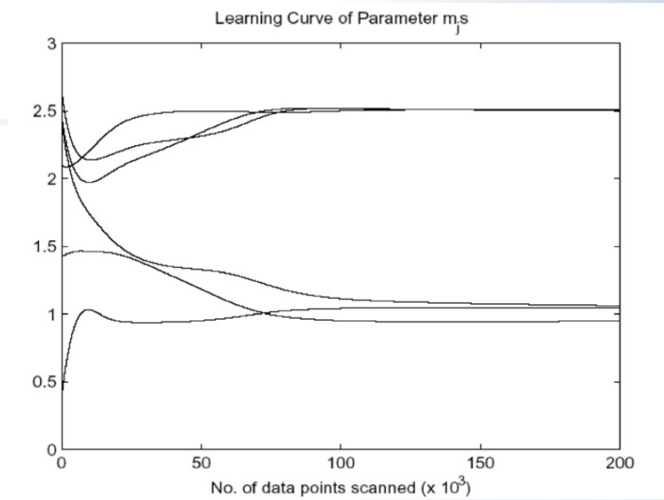
Results for RPEM



Results for EM

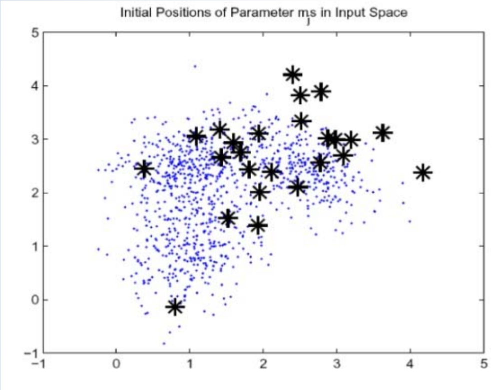


Learning curve for RPEM

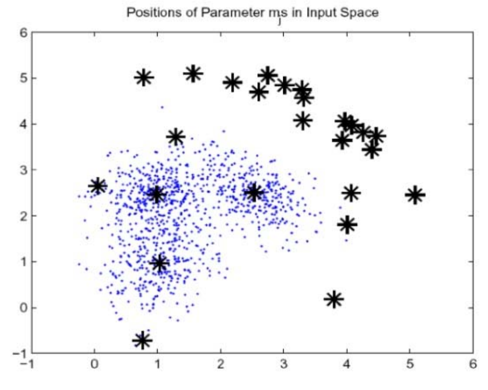


Learning curve for EM

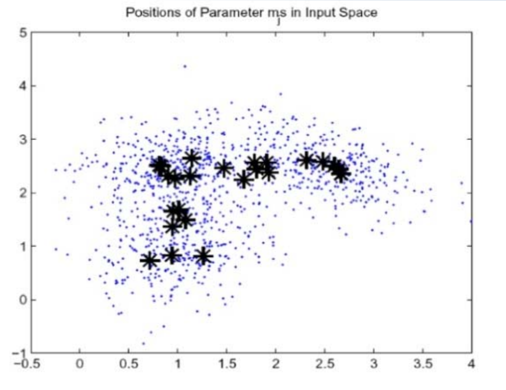
For k = 25, the distribution of the convergent seed points



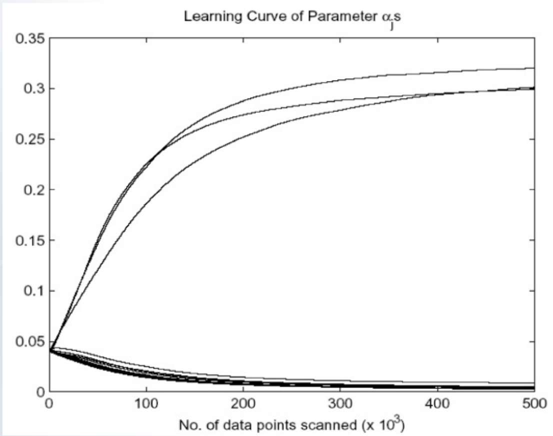
Initial seed points



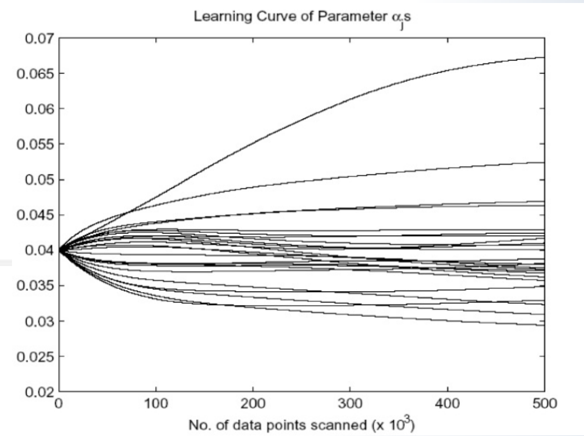
Results for RPEM



Results for EM

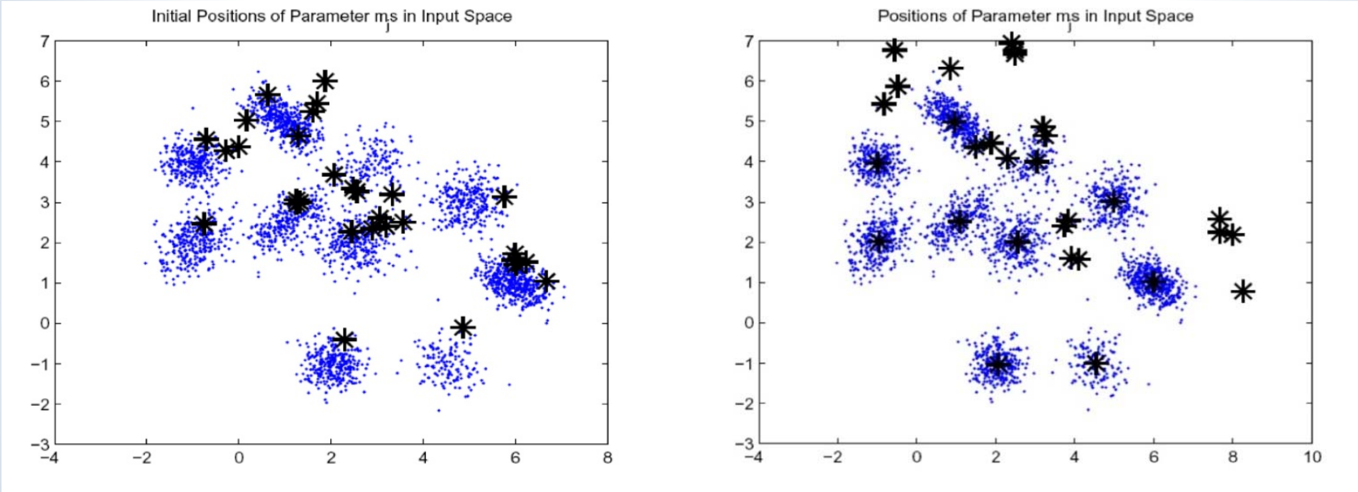


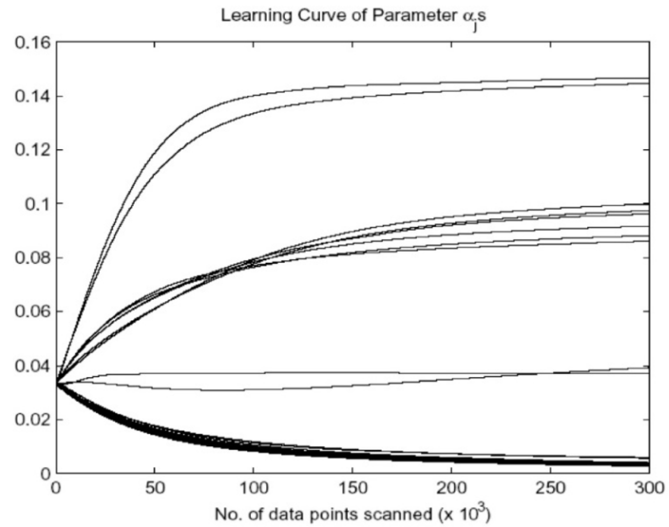
Learning curve for RPEM



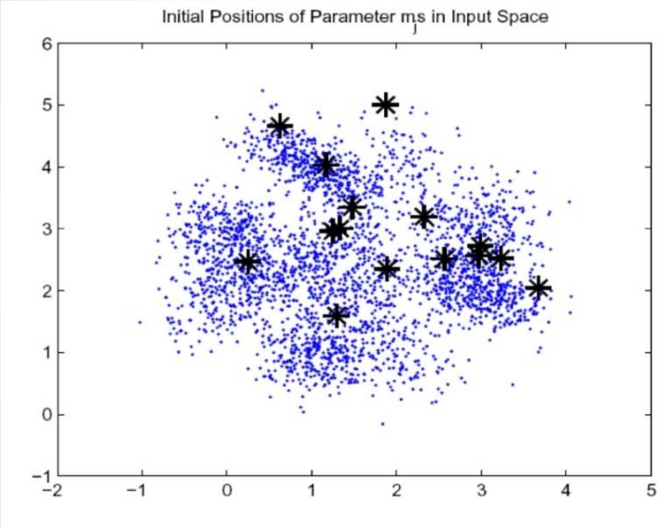
Learning curve for EM

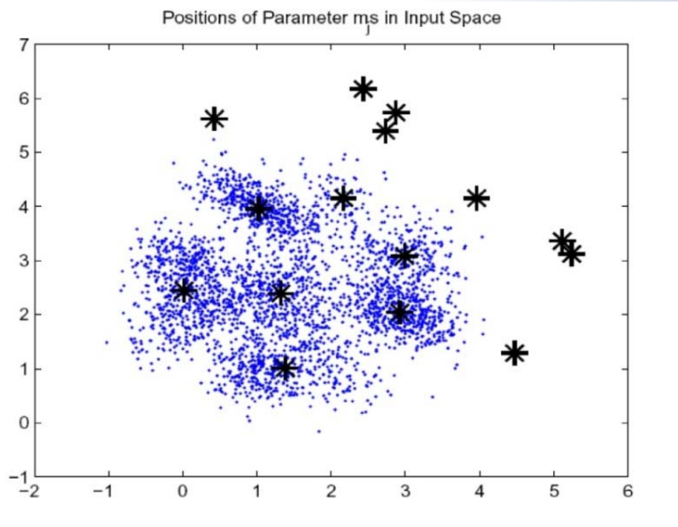
The performance of RPEM in more clusters





Empirical investigation of robustness of RPEM

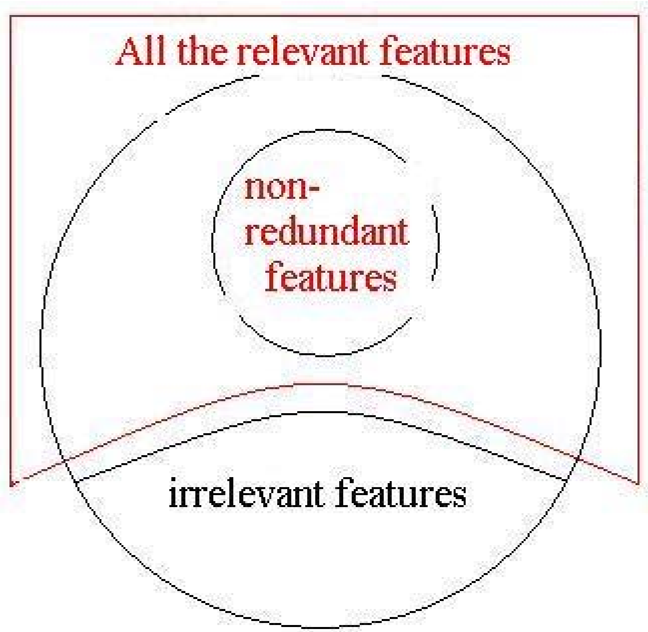




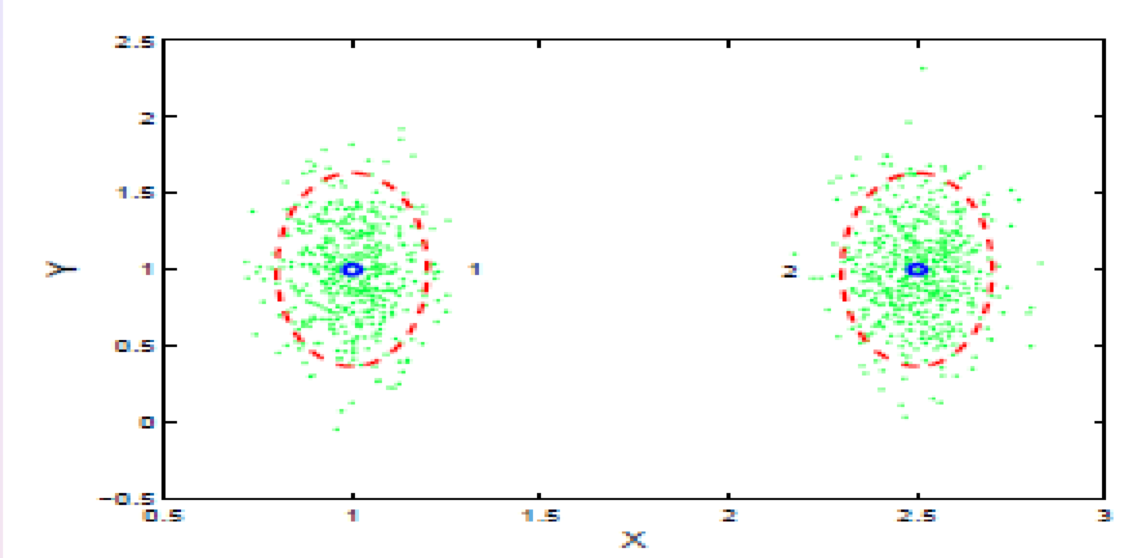
OUTLINE

- Introduction √
  - The existing unsupervised feature selection methods
- The RPEM Algorithm √
- Unsupervised Feature Selection Schemes
- The Iterative Feature Selection and Clustering Algorithm
- Experimental Results
- Conclusion

UNSUPERVISED FEATURE SELECTION



- Selecting the relevant features



  - The feature X is relevant to the partitioning, while the feature Y is irrelevant.
  - Our Claim: A feature is less relevant if, along this feature, the variance of observations in a cluster is closer to the global variance of observations in all clusters.

We propose a quantitative index to measure the relevance of each feature:

$$SCORE_l = \frac{1}{k}\sum_{j=1}^{k} Score_{l,j} = \frac{1}{k}\sum_{j=1}^{k}(1-\frac{\delta_{l,j}^2}{\delta_l^2}), l = 1,\ldots,d$$

$\delta_{l,j}^2$: the variance of the j^{th} cluster projected on the l^{th} dimension (local):

$$\delta_{l,j}^2 = \frac{1}{N_j - 1}\sum_{t=1}^{N_j}(x_{l,t} - \mu_{l,j})^2, \mathbf{x}_t \in j^{th} cluster,$$

δ_l^2: the variance of the whole data on the lth dimension (global):

$$\delta_l^2 = \frac{1}{N-1}\sum_{t=1}^{N}(x_{l,t} - \bar{\mu}_l)^2, \bar{\mu} = \frac{1}{N}\sum_{t=1}^{N} x_{l,t}.$$

- the optimal case: $SCORE_l = 1$; the worst case: $SCORE_l = 0$.
- the refined relevant feature subset:

$$R' = F - \{F_l \mid SCORE_l < \beta, F_l \in F\}$$

- Selecting the non-redundant features

- Markov Blanket (Pearl): Given a feature F_l, let $M_l \subset F \; (F_l \notin M_l)$
  M_l is said to be the Markov Blanket for Fl if:

$$P(F - M_l - F_l, C \mid F_l, M_l) = P(F - M_l - F_l, C \mid M_l).$$

- If a Markov Blanket M_l for F_l can be found in the feature set F, i.e. M_l subsumes the information that F_l has about C, we are able to eliminate the feature F_l from F without affecting the class prediction accuracy.
- The closeness of candidate M_l to being a true Markov Blanket for F_l is measured by (Koller&Sahami):

$$\Delta(F_l \mid M_l) \sum_{f_{M_l}, f_l} P(M_l = f_{M_l}, F_l = f_l) \cdot KL(P(C \mid M_l = f_{M_l}, F_l = f_l) \,\|\, P(C \mid M_l = f_{M_l}))$$

- where $KL(.\|.)$ denotes the Kullback-Leibler divergence:

$$KL(P \,\|\, Q) = \sum_z P(z) \log(P(z) / Q(z)).$$

- Exact Markov Blanket for $F_l : \Delta(F_l \mid M_l) = 0;$
- Approximate Markov Blanket for $F_l : \Delta(F_l \mid M_l)$ being small.

Algorithm 1: The Markov Blanket filtering algorithm.

Initialize

- $G^{(1)} = F;$

Iterate

- For each feature $F_l \in G^{(m)}$ let M_l be the set of T features $F_i \in G^{(m)} - F_l$ for which the correlation between F_l and F_i are the highest;
- Compute $\Delta(F_l|M_l)$ for each feature l;
- Choose the F_{l_m} that minimizes $\Delta(F_l|M_l)$, and define $G^{(m+1)} = G^{(m)} - F_{l_m}$;

Until $|G^{(m+1)}| = T$.

- non-redundant features (classes are replaced by clusters):

$$R'' = \{F_{l_m} \mid \min_{F_l \in G^{(m)}} \Delta(F_l \mid M_l) > \gamma \cdot \min_{F_l \in G^{(1)}} \Delta(F_l \mid M_l)\} \bigcup \{R' - \{F_{l_1}, F_{l_2}, \ldots, F_{l_{|R'|-T}}\}\}$$

where $m = 1, 2, \ldots, |R'| - T, F_{l_m} \in R', G^{(1)} = R'.$

OUTLINE

- Introduction √
  - The existing unsupervised feature selection methods
- The RPEM Algorithm √
- Unsupervised Feature Selection Schemes √
- The Iterative Feature Selection and Clustering Algorithm
- Experimental Results
- Conclusion

THE ITERATIVE FEATURE SELECTION AND CLUSTERING ALGORITHM

Algorithm 2: Iterative Feature Selection in RPEM clustering algorithm.

input : $\mathbf{X}_N, k_{max}, \eta, epoch_{max}, \beta, \gamma, T$
output: *the most relevant and non-redundant feature subset* $\hat{R}$

$\hat{R} \leftarrow \{F\}$;
$epoch\_count \leftarrow 0$;
while $epoch\_count \leq epoch_{max}$ **do**
  for $t \leftarrow 1$ **to** N **do**
    Step 1: *Calculate* $h(j|\mathbf{x}_t, \hat{\Theta})$*'s to obtain* $g(j|\mathbf{x}_t, \hat{\Theta})$*'s on* $\hat{R}$*;*
    Step 2: *Update parameters* $\hat{\Theta}$ *on* F*;*
    $\hat{\Theta}^{(new)} = \hat{\Theta}^{(old)} + \eta \left.\frac{\partial \mathcal{M}(\mathbf{x}_t;\hat{\Theta})}{\partial \Theta}\right|_{\hat{\Theta}^{(old)}}$;
  end
  $\hat{R} \leftarrow$ `FeatureSelection`(F, β, γ, T);
  $epoch\_count \leftarrow epoch\_count + 1$;
end

Procedure `FeatureSelection` (F, β, γ, T)

input : F, β, γ, T
output: $\hat{R}$

`// Selecting the relevant features`
Calculate $SCORE_l, F_l \in F$;
$R' \leftarrow F - \{F_l | SCORE_l < \beta, F_l \in F\}$;
`// Selecting the non-redundant features`
Perform Markov Blanket filtering;
$R'' = \{F_{l_m} | \min_{F_l \in G^{(m)}} \Delta(F_l|M_l) > \gamma \cdot \min_{F_l \in G^{(1)}} \Delta(F_l|M_l)\} \cup \{R' - \{F_{l_1}, F_{l_2}, \ldots, F_{l_{|R'|-T}}\}\}$;
$\hat{R} \leftarrow R''$;

OUTLINE

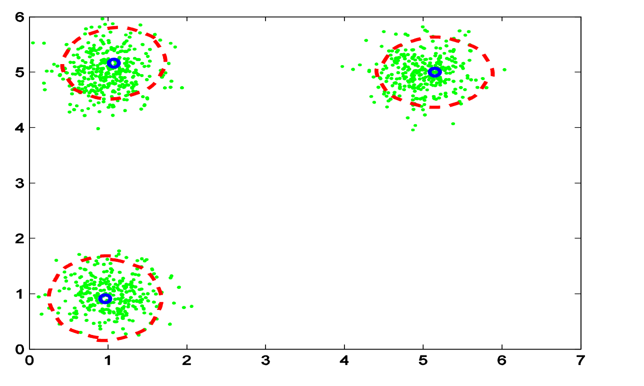
- Introduction √
    - The existing unsupervised feature selection methods
- The RPEM Algorithm √
- Unsupervised Feature Selection Schemes √
- The Iterative Feature Selection and Clustering Algorithm √
- Experimental Results
- Conclusion

EXPERIMENTAL RESULTS

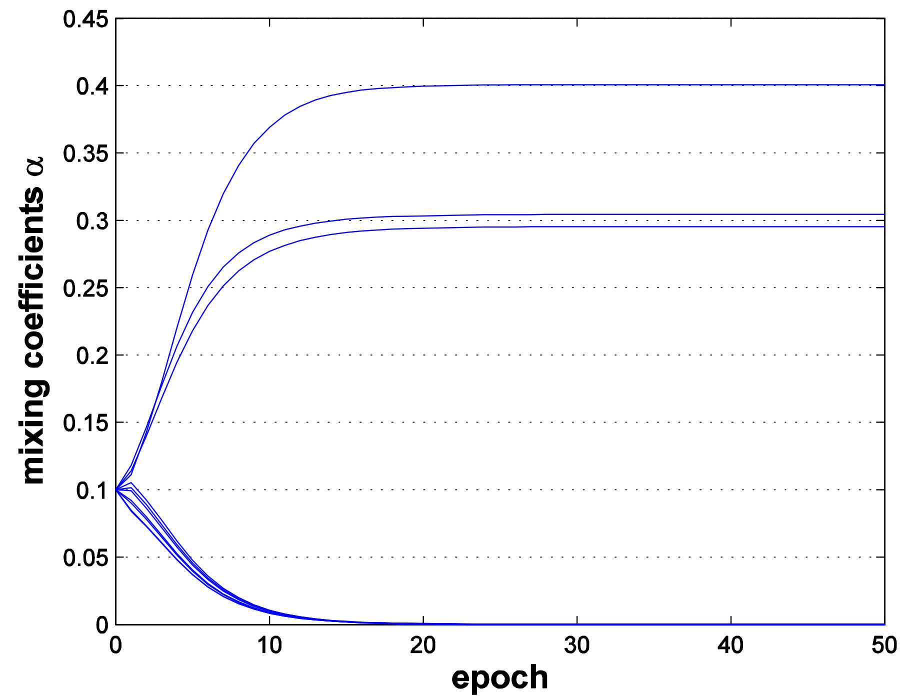
- The system parameters

parameter	k_{max}	β	γ	T
value	10	0.4	2	2

- Synthetic data 1

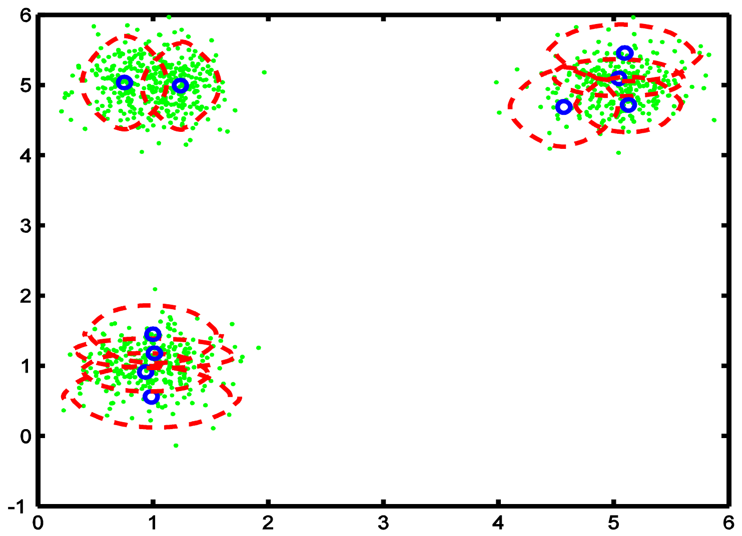


- F_1 and F_2 are relevant features;
- F_3; F_4 are obtained by duplicating F_1 and F_2; (thus either {F_3; F_4} or {F_1; F_2} are redundant.)
- F_5 -F_{10} were sampled from standard Gaussian, thus being unimodal (irrelevant to the clustering);

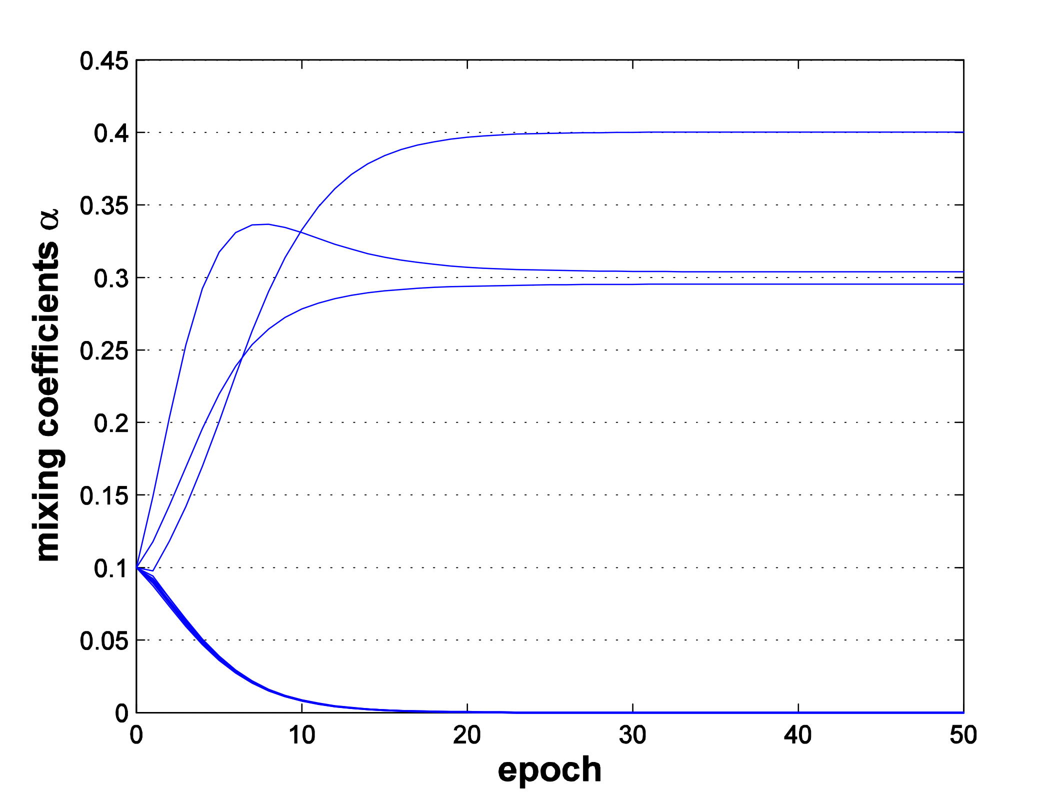


epoch	ranking	selected features
1	0.9(F_1) 0.9(F_4) 0.9(F_3) 0.9(F_2) 0.3(F_6) 0.3(F_7) 0.2(F_{10}) 0.1(F_8) 0.1(F_5) 0.1(F_9)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}
15	0.8(F_1) 0.8(F_2) 0.8(F_4) 0.8(F_3) 0.2(F_7) 0.2(F_8) 0.2(F_6) 0.2(F_{10}) 0.2(F_5) 0.1(F_9)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}
50	0.9(F_2) 0.9(F1) 0.9(F_4) 0.9(F_3) 0.0(F_7) 0.0(F_5) 0.0(F_8) 0.0(F_9) 0.0(F_{10}) 0.0(F_6)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}

- The algorithm in (Law et al. 2004) assumes that the pdf of the irrelevant features is Gaussian;
- Let be uniformly distributed (irrelevant to the clustering); The distribution of the irrelevant features is bias from the pre-specified one in (Law et al. 2004).



- Remark: The algorithm in (Law et al. 2004) is sensitive to the assumed pdf for the irrelevant features;



epoch	ranking	selected features
1	0.9(F_1) 0.9(F_2) 0.9(F_3) 0.9(F_4) 0.3(F_6) 0.3(F_5) 0.2(F_9) 0.2(F_8) 0.1(F_{10}) 0.1(F_7)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}
7	0.6(F_1) 0.6(F_4) 0.6(F_2) 0.6(F_3) 0.2(F_8) 0.1(F_6) 0.1(F_{10}) 0.1(F_7) 0.0(F_5) 0.0(F_9)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}
50	0.9(F_1) 0.9(F_2) 0.9(F_3) 0.9(F_4) 0.0(F_{10}) 0.0(F_8) 0.0(F_9) 0.0(F_7) 0.0(F_5) 0.0(F_6)	{F_1; F_2; F_3; F_4}
	0(F_1) 0(F_2)	{F_3; F_4}

IRRFS-RPEM: the proposed algorithm;

IRFS-RPEM: a variant without redundancy analysis.

Data Set	Method	Model Order mean ± std	Error Rate mean ± std
Wdbc d=30 N=569 k^* =2	RPEM GMClusFW IRFS-RPEM IRRFS-RPEM	1.7 ± 0.4 5.7 ± 0.3 2.3 ± 0.4 Fixed at 2	0.2610 ± 0.0781 0.1005 ± 0.0349 0.1021 ± 0.0546 0.0897 ± 0.0308
Sonar d=30 N=569 k^* =2	RPEM GMClusFW IRFS-RPEM IRRFS-RPEM	2.3 ± 0.8 1.0 ± 0.0 2.8 ± 0.6 2.7 ± 0.7	0.4651 ± 0.0532 0.5000 ± 0.0000 0.3625 ± 0.0394 0.3221 ± 0.0333
Wine d=30 N=569 k^* =2	RPEM GMClusFW IRFS-RPEM IRRFS-RPEM	2.5 ± 0.7 3.3 ± 1.4 4.7 ± 1.7 3.1 ± 0.5	0.0843 ± 0.0261 0.0673 ± 0.0286 0.0492 ± 0.0182 0.0509 ± 0.0248
Ionosphere d=30 N=569 k^* =2	RPEM GMClusFW IRFS-RPEM IRRFS-RPEM	1.8 ± 0.5 3.2 ± 0.6 2.6 ± 0.8 2.5 ± 0.5	0.4056 ± 0.0121 0.2268 ± 0.0386 0.2921 ± 0.0453 0.2121 ± 0.0273

Data Set	Method	Model Order mean ±std	Error Rate mean ±std
wdbc	IRFS-RPEM IRRFS-RPEM	2.3 ± 0.4 Fixed at 2	0.1021 ± 0.0546 **0.0897 ± 0.0308**
sonar	IRFS-RPEM IRRFS-RPEM	2.8 ± 0.6 2.7 ± 0.7	0.3625 ± 0.0394 **0.3221 ± 0.0333**

Table: The proportions of the average selected features

Data	IRFS-RPEM	IRRFS-RPEM
wdbc	51.16%	50.33%
sonar	57%	55.83%

Data Set	Method	Model Order mean ±std	Error Rate mean ±std
wine	IRFS-RPEM IRRFS-RPEM	4.7 ± 1.7 3.1 ± 0.5	0.0492 ± 0.0182 0.0509 ± 0.0248
ionosphere	IRFS-RPEM IRRFS-RPEM	2.6 ± 0.8 2.5 ± 0.5	0.2921 ± 0.0453 0.2121 ± 0.0273

Table: The proportions of the average selected features

Data	IRFS-RPEM	IRRFS-RPEM
wine	83.65%	62.31%
ionosphere	68.13%	34.38%

Data Set	Method	Model Order mean $\pm$ std	Error Rate mean std
wdbc	GMClusFW IRRFS-RPEM	5.7 $\pm$ 0.3 **Fixed at 2**	0.1005 $\pm$ 0.0349 **0.0897 $\pm$ 0.0308**
sonar	GMClusFW IRRFS-RPEM	1.0 $\pm$ 0.0 2.7 $\pm$ 0.7	0.5000 $\pm$ 0.0000 **0.3221 $\pm$ 0.0333**
wine	GMClusFW IRRFS-RPEM	3.3 $\pm$ 1.4 3.1 $\pm$ 0.5	0.0673 $\pm$ 0.0286 **0.0509 $\pm$ 0.0248**
ionosphere	GMClusFW IRRFS-RPEM	3.2 $\pm$ 0.6 2.5 $\pm$ 0.5	0.2268 $\pm$ 0.0386 **0.2121 $\pm$ 0.0273**

CONCLUSION

- Develop RPEM algorithm from the MWL learning framework;
- A new feature relevance measurement index is proposed;
- The algorithm iterates between the clustering and feature selection, featuring that:
  - It does not particularly assume the pdf for the irrelevant features;
  - Effective in eliminating both irrelevant and redundant features;

Thanks!

Q&A

REFERENCES:

- [Dash et al. 2002] M. Dash, K. Scheuermann, P. Liu, ”Feature Selection for Clustering – A Filter Solution”, Proceedings of IEEE International Conferenceon Data Mining, pp. 115-122, 2002.
- [Miltra et al. 2002] P. Miltra, C. Murthy, S. Pal, ”Unsupervised Feature Selection Using Feature Similarity”, IEEE Transactions on Pattern Analysis and Machinary Intelligence, 24(2), pp. 301-312, 2002.
- [Law et al. 2004] M. Law, M. Figueiredo, A. Jain, ”Simultaneous Fature Selection and Clustering Using Mixture Models, IEEE Transactions on Pattern Analysis and Machinary Intelligence, 26(9), pp. 1154-1166, 2004.
- [Dy and Brodley 2000] J. Dy, C. Brodley, ”Visualization and Interactive Feature Selection for Unsupervised Data”, Proceedings of ACM Special Interest Group on Knowledge Discovery in Data, pp. 360-364, 2000.

- [Dy and Brodley 2005] J. Dy, C. Brodley, ”Feature Selection for Unsupervised Learning”, J. Machine Learning Res., 5, pp. 845-889, 2005.

- [Constantinopoulos et al. 2006] C .Constantinopoulos, M. Titsias, A. Likas, ”Bayesian Feature and Model Selection for Gaussian Mixture Models”,IEEE Transactions on Pattern Analysis and Machinary Intelligence, 28(6), pp. 1013-1018, 2006.