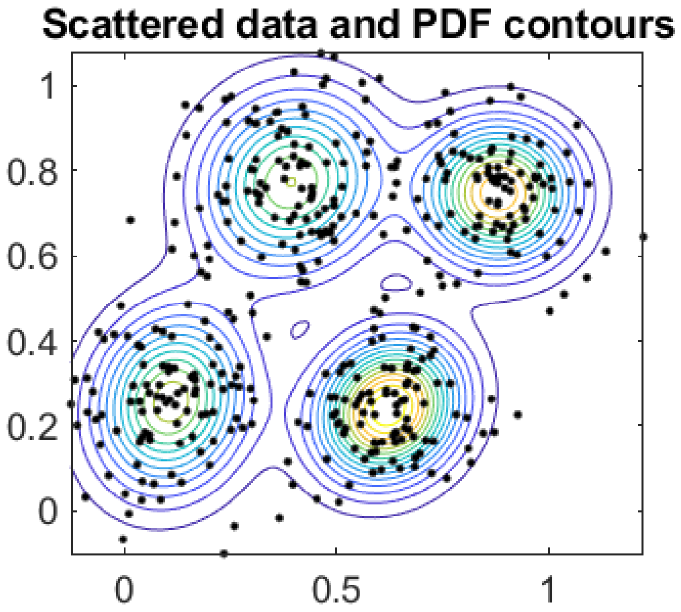
Reinforced EM Algorithm through Clever Initialization for Clustering with Gaussian Mixture Models

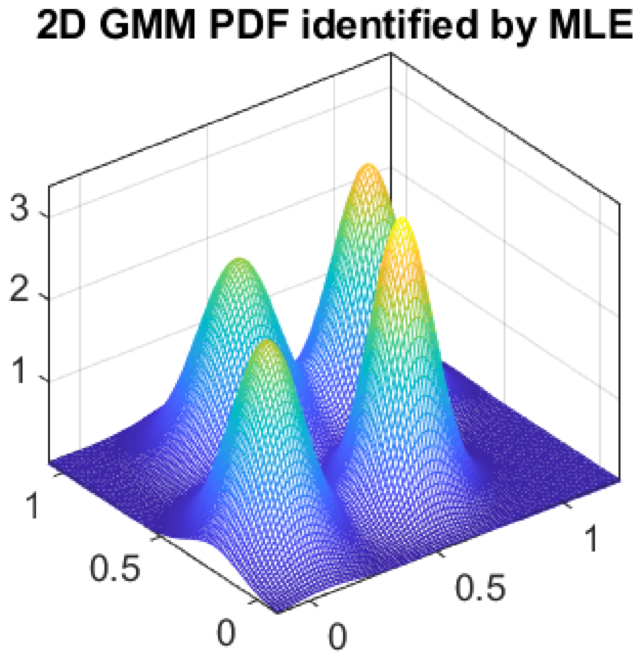
Joshua Tobin
TCD

Chin Pang Ho
CityU HK

Mimi Zhang
TCD

GMM for Clustering





A GMM density has the form

$$f(\boldsymbol{x}) = \sum_{j=1}^{m} \pi_j \phi(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\,.$$

Clustering is done by assigning each $\boldsymbol{x}_i$ to the mixture component (i.e., cluster) to which it is most likely to belong a posteriori.

EM Algorithm

1. Initialize the parameters: $\{\pi_1, \ldots, \pi_m\}$, $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m\}$ and $\{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_m\}$.
2. Compute the responsibilities: for $i = 1, \ldots, n$ and $j = 1, \ldots, m$,

$$r_{ij} = \frac{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{v=1}^{m} \pi_v \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_v, \boldsymbol{\Sigma}_v)}.$$

3. Update the estimates: for $j = 1, \ldots, m$,

$$\pi_j = \frac{\sum_{i=1}^{n} r_{ij}}{n}, \quad \boldsymbol{\mu}_j = \frac{\sum_{i=1}^{n} r_{ij}\boldsymbol{x}_i}{\sum_{i=1}^{n} r_{ij}}, \quad \boldsymbol{\Sigma}_j = \frac{\sum_{i=1}^{n} r_{ij}(\boldsymbol{x}_i - \boldsymbol{\mu}_j)(\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T}{\sum_{i=1}^{n} r_{ij}}.$$

4. Iterate steps 2 and 3 until convergence.

EM Algorithm

With random initialization, converge to bad local maxima with probability $1 - e^{-O(m)}$.

1. $\boxed{\text{Initialize}}$ the parameters: $\{\pi_1, \ldots, \pi_m\}$, $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m\}$ and $\{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_m\}$.
2. Compute the responsibilities: for $i = 1, \ldots, n$ and $j = 1, \ldots, m$,

$$r_{ij} = \frac{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{v=1}^{m} \pi_v \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_v, \boldsymbol{\Sigma}_v)}.$$

3. Update the estimates: for $j = 1, \ldots, m$,

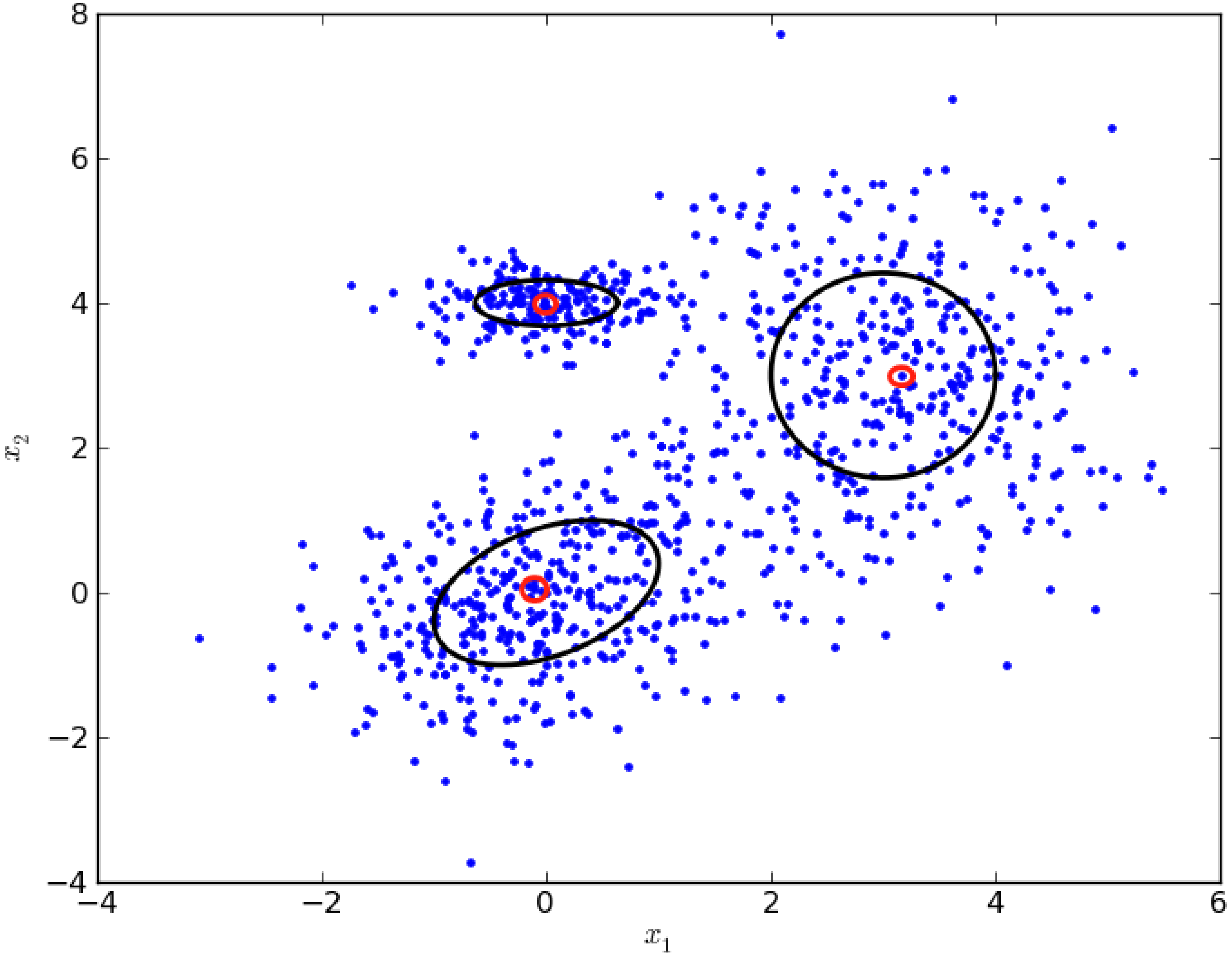
$$\pi_j = \frac{\sum_{i=1}^{n} r_{ij}}{n}, \quad \boldsymbol{\mu}_j = \frac{\sum_{i=1}^{n} r_{ij} \boldsymbol{x}_i}{\sum_{i=1}^{n} r_{ij}}, \quad \boldsymbol{\Sigma}_j = \frac{\sum_{i=1}^{n} r_{ij} (\boldsymbol{x}_i - \boldsymbol{\mu}_j)(\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T}{\sum_{i=1}^{n} r_{ij}}.$$

4. Iterate steps 2 and 3 until $\boxed{\text{convergence.}}$

May converge to a singularity.

Exemplar Means

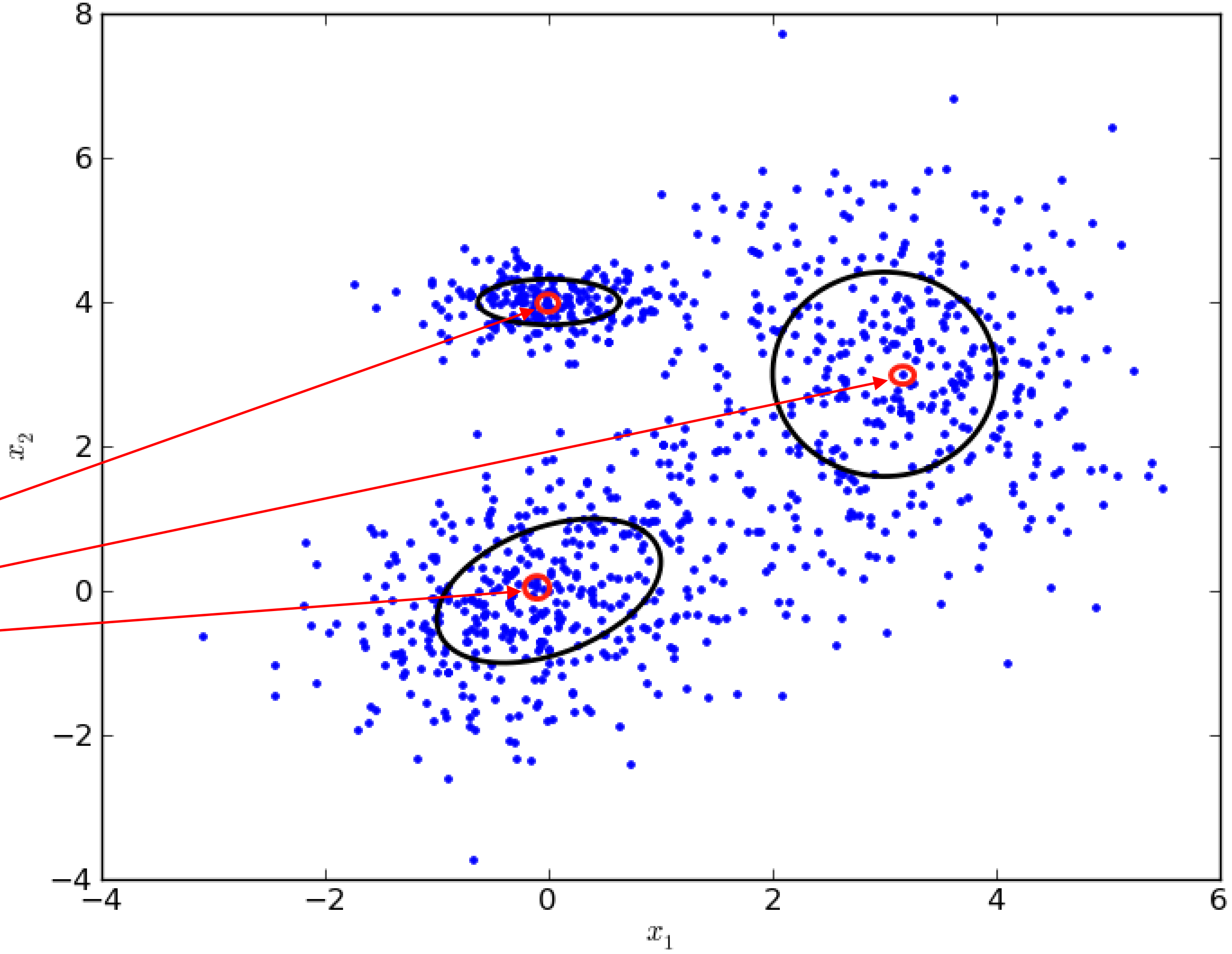
Assume that the clusters are dense enough, such that there is always a data point very close to the real cluster centre.



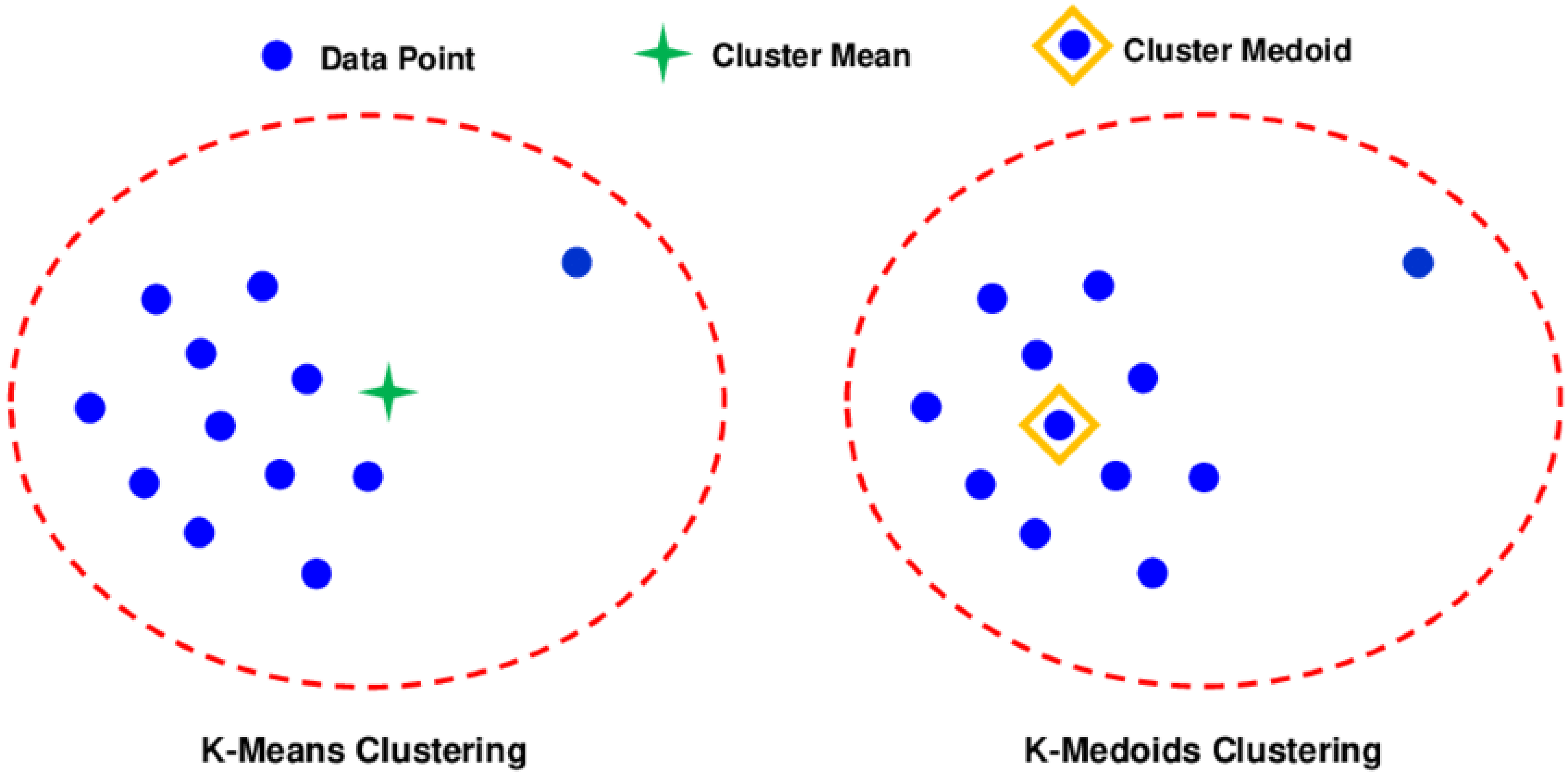
Exemplar Means

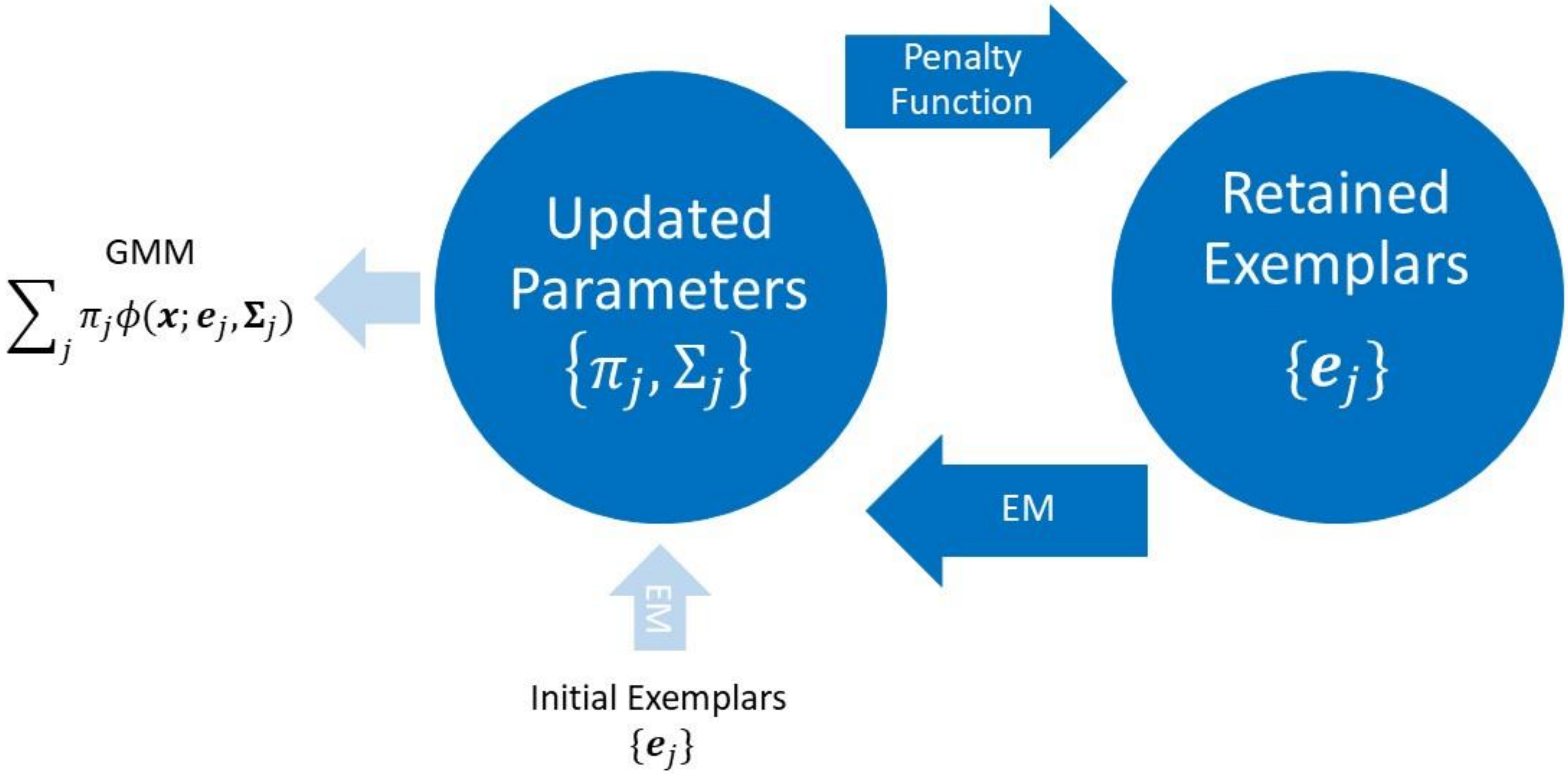
Assume that the clusters are dense enough, such that there is always a data point very close to the real cluster centre.

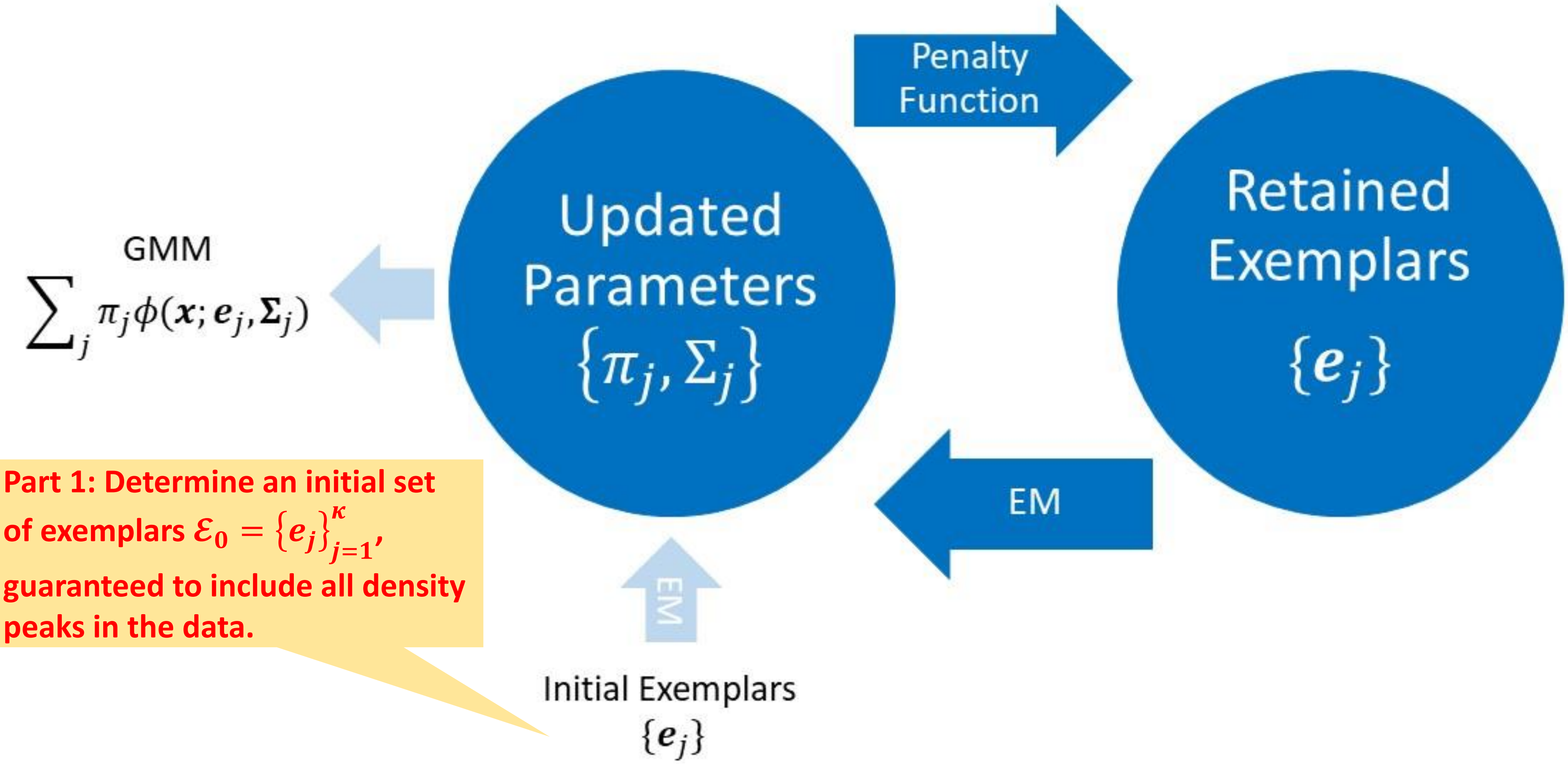
Fix Gaussian means at "exemplars" in the dataset.



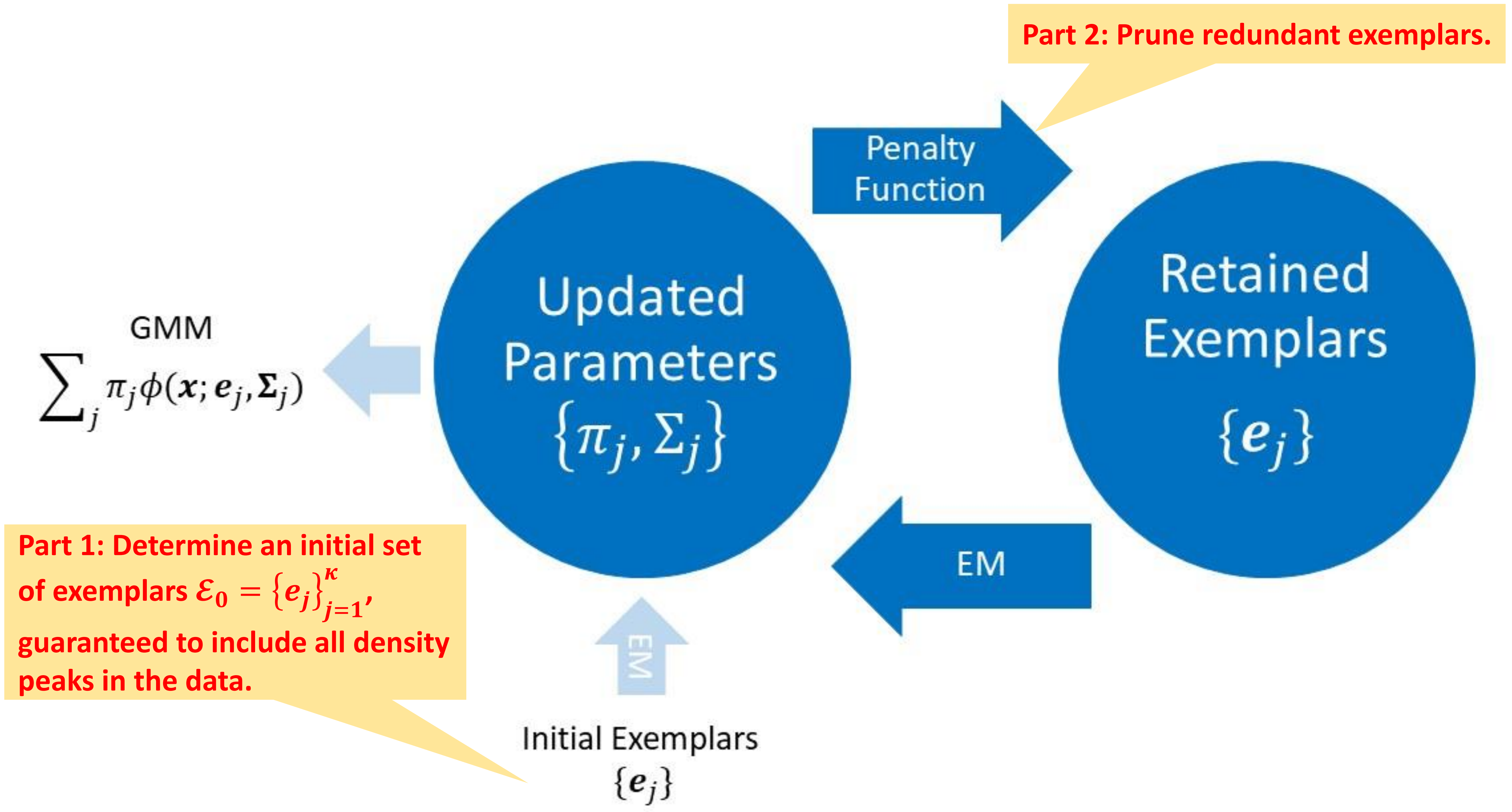
Exemplar Means




GMM
$\sum_{j} \pi_j \phi(\boldsymbol{x}; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)$
Updated Parameters
$\{\pi_j, \Sigma_j\}$
Penalty Function
Retained Exemplars
$\{\boldsymbol{e}_j\}$
EM
EM
Initial Exemplars
$\{\boldsymbol{e}_j\}$



Part 1: Determine an initial set of exemplars $\mathcal{E}_0 = \{e_j\}_{j=1}^{\kappa}$, guaranteed to include all density peaks in the data.

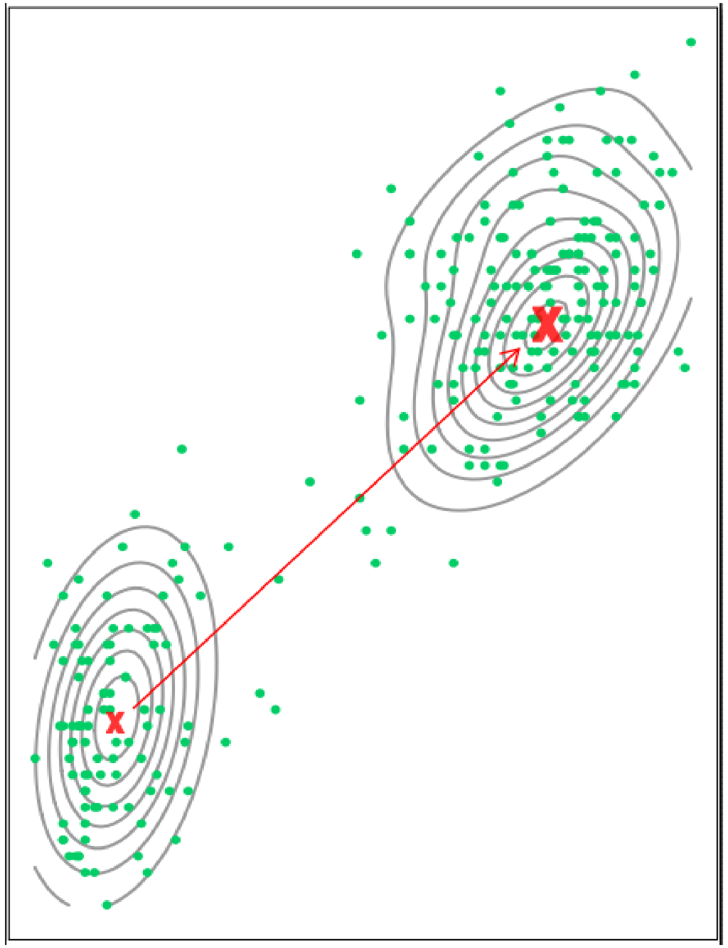

Part 2: Prune redundant exemplars.
Penalty Function
GMM
$\sum_j \pi_j \phi(\boldsymbol{x}; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)$
Updated Parameters $\{\pi_j, \Sigma_j\}$
Retained Exemplars $\{\boldsymbol{e}_j\}$
EM
EM
Part 1: Determine an initial set of exemplars $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, guaranteed to include all density peaks in the data.
Initial Exemplars $\{\boldsymbol{e}_j\}$

Part 1 – Density-Peak Finding

Density peaks are characterized by:

(1) a higher density than their neighbours

(2) a relatively large distance from points with higher densities

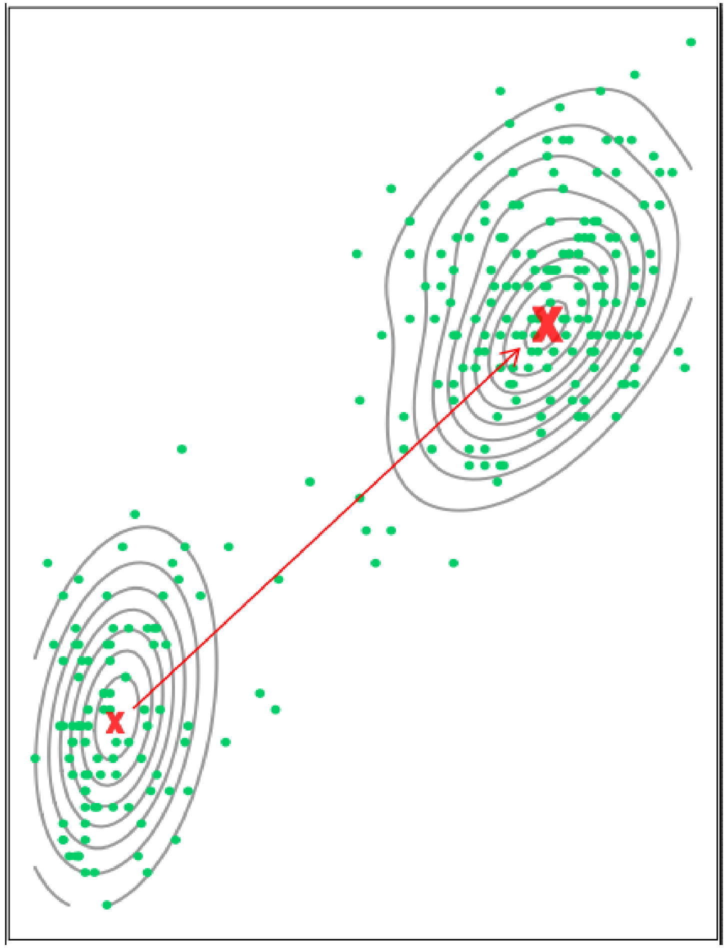


Part 1 – Density-Peak Finding

Density peaks are characterized by:

(1) a higher density than their neighbours

(2) a relatively large distance from points with higher densities



Part 1 – Density-Peak Finding

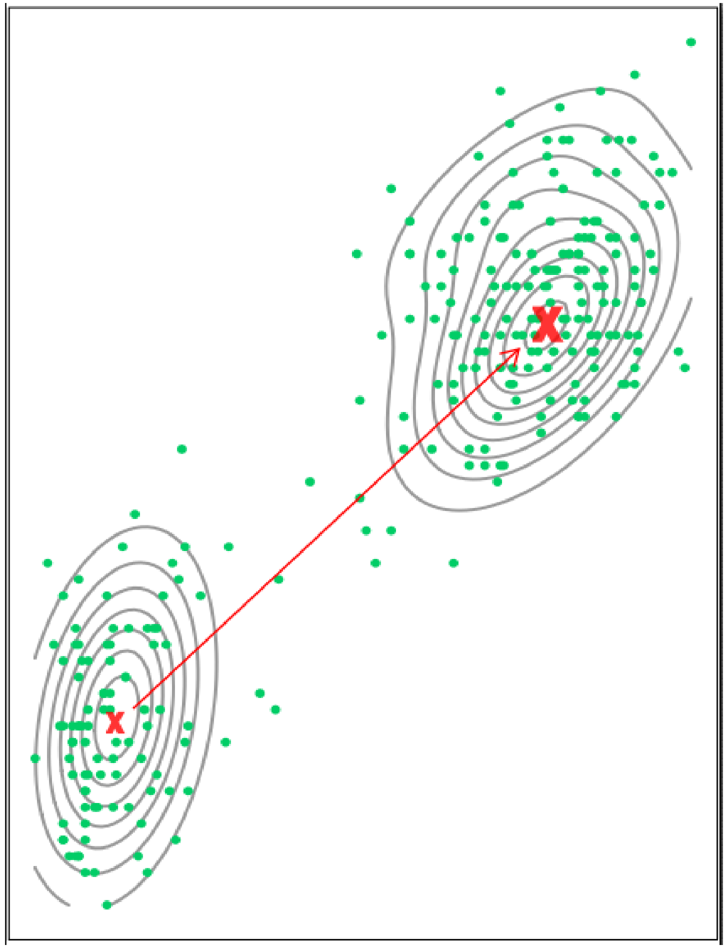
Density peaks are characterized by:

(1) a higher density than their neighbours

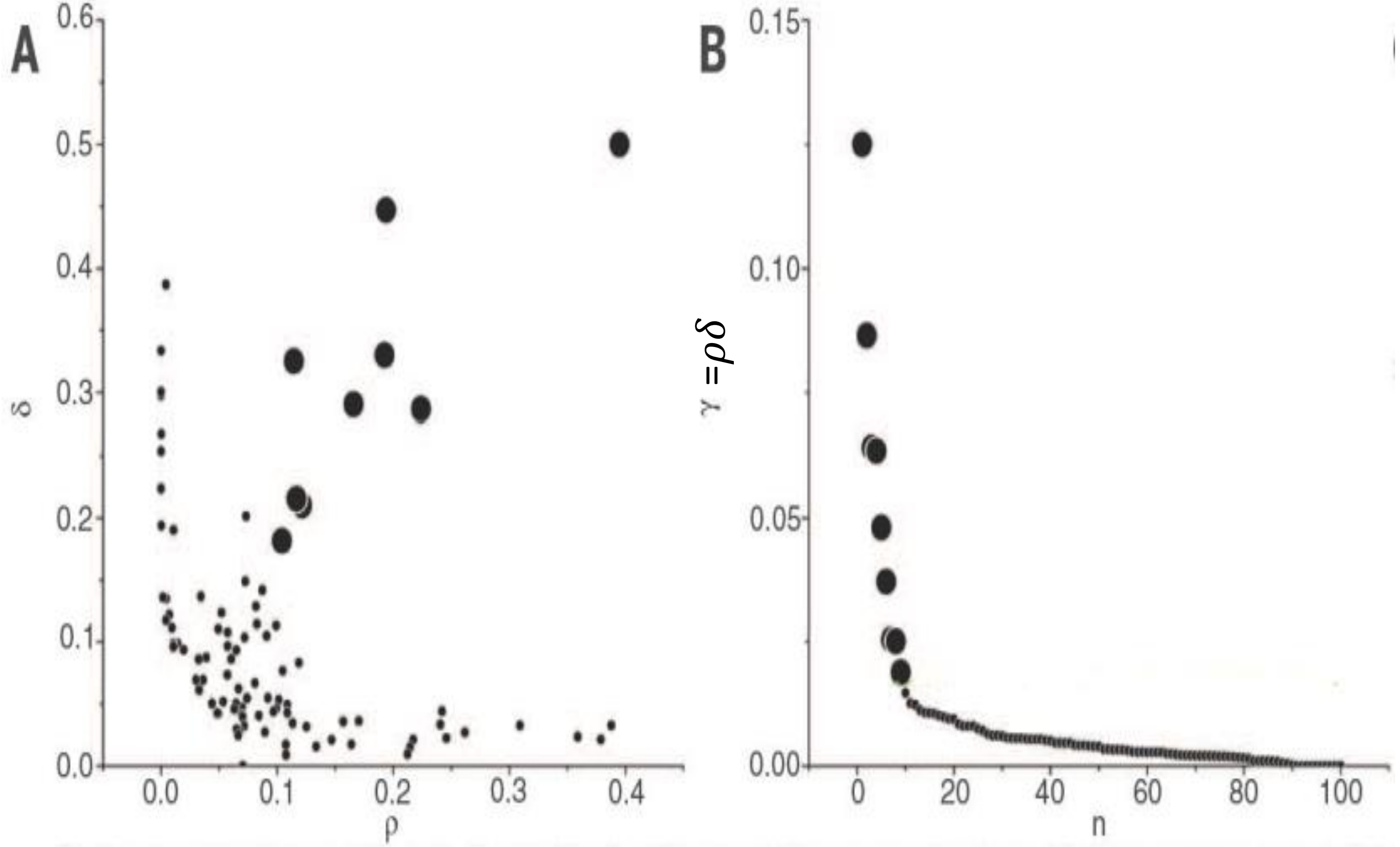
$\rho(\boldsymbol{x})$ – Gaussian kernel density estimate

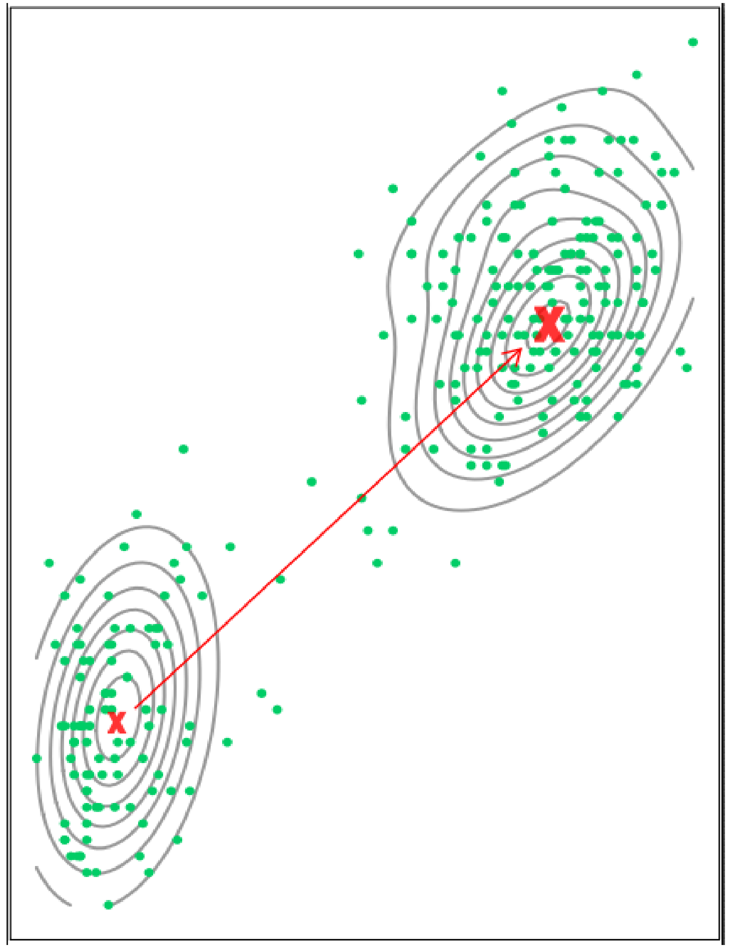
(2) a relatively large distance from points with higher densities

$\delta(\boldsymbol{x})$ -- distance to the nearest neighbour of higher local density

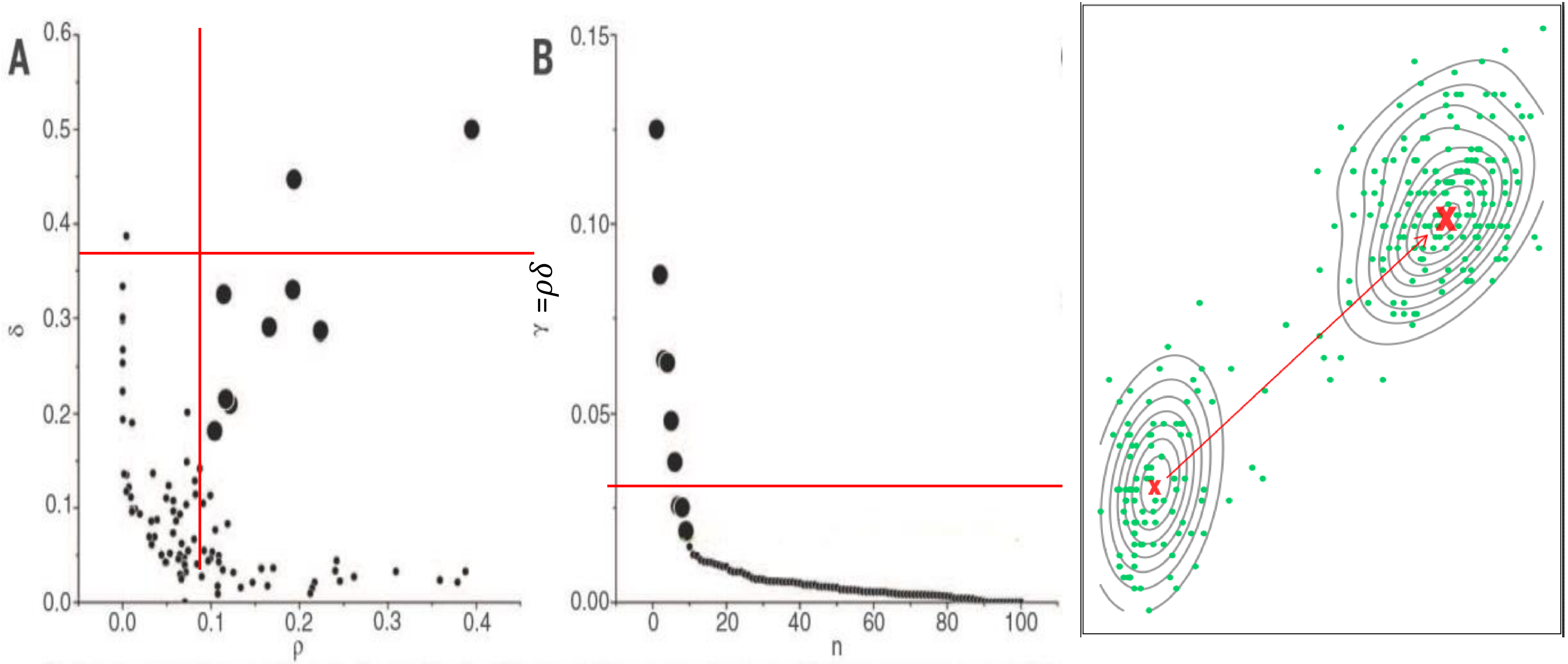


Part 1 – Density-Peak Finding

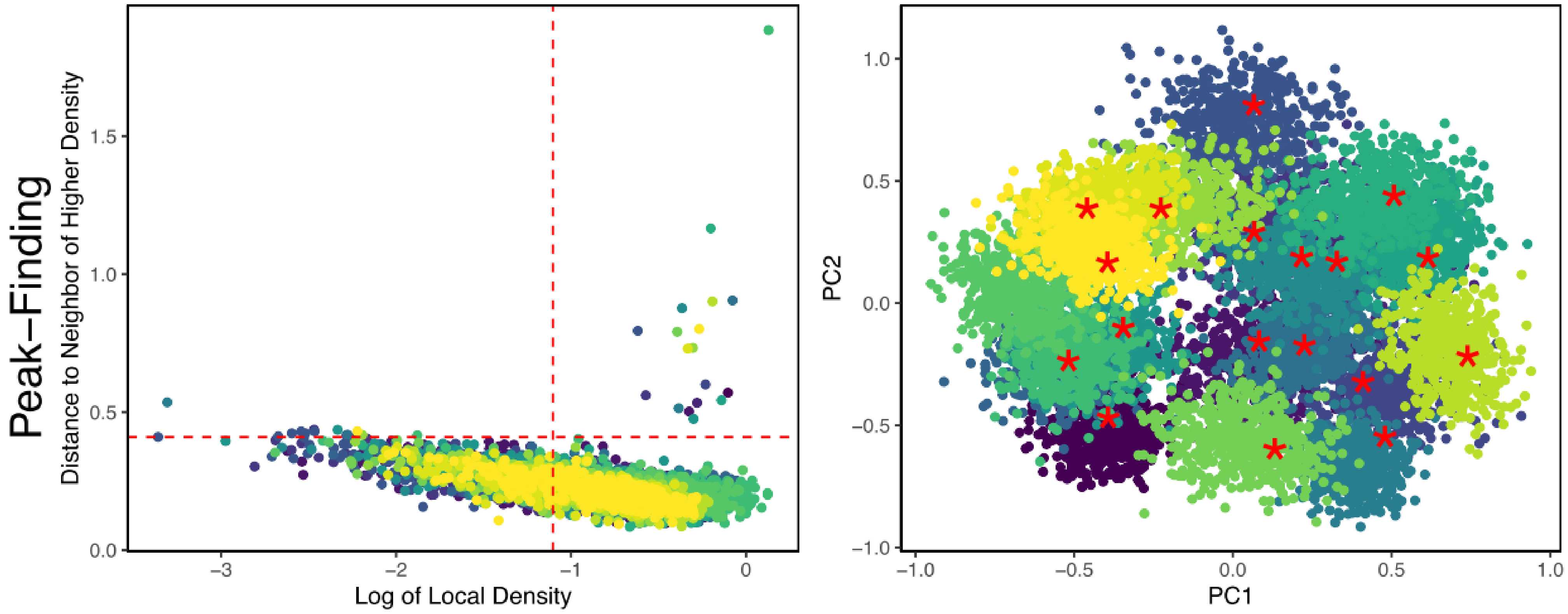




Part 1 – Density-Peak Finding

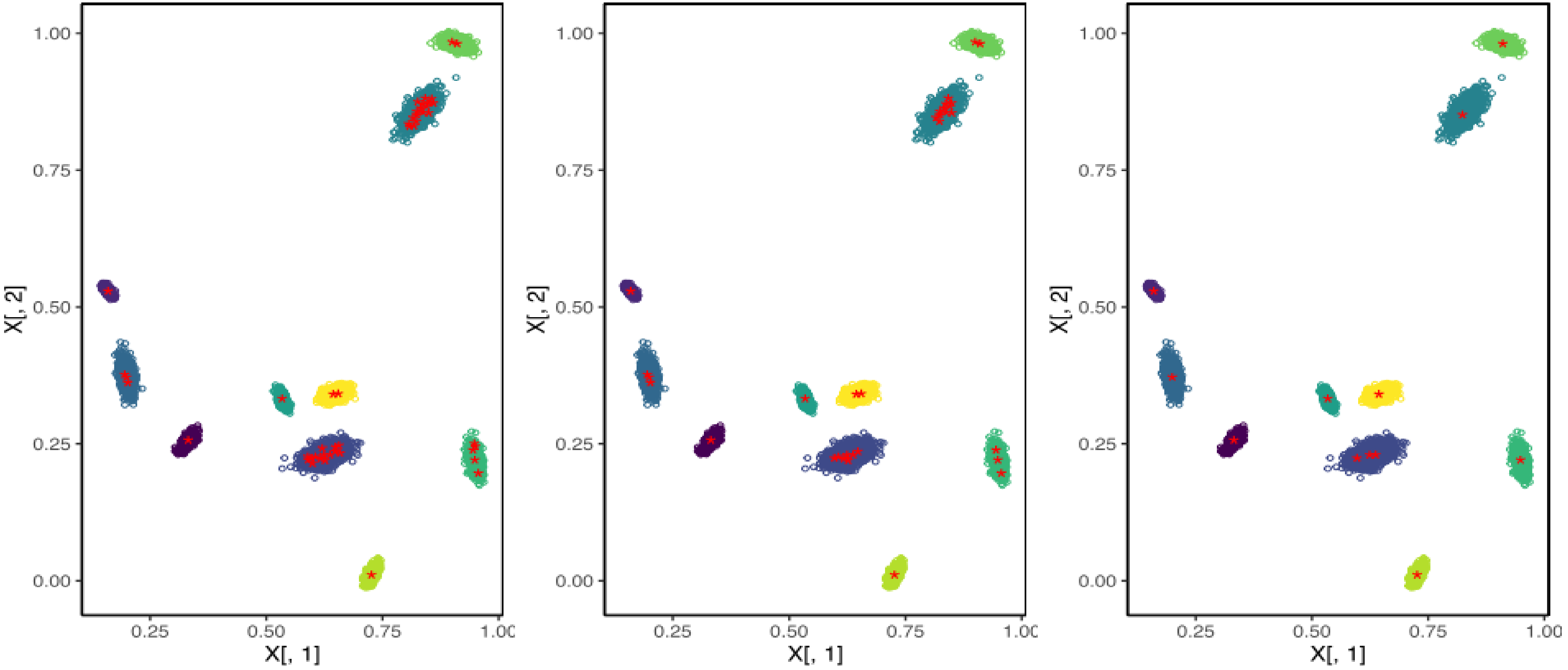


Part 1 – Density-Peak Finding



A 10-dimensional dataset contains 20 Gaussian components. Clusters are indicated by different colours. The right figure shows the locations of the selected peaks, projected onto the first two principal components.

Part 1 – Density-Peak Finding



Exemplars selected for different cut-off levels on the density $\rho(\boldsymbol{x})$ and distance $\delta(\boldsymbol{x})$. Left: 10^{th}-percentile of $\rho(\boldsymbol{x})$ & 97.5^{th}-p of $\delta(\boldsymbol{x})$. Center: 20^{th}-p of $\rho(\boldsymbol{x})$ & 97.5^{th}-p of $\delta(\boldsymbol{x})$. Right: 20^{th}-p of $\rho(\boldsymbol{x})$ & 99.5^{th}-p of $\delta(\boldsymbol{x})$.

Part 1 – Density-Peak Finding

Theorem: For n large enough, with high probability, $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{k}$ contains unique estimates for all the true modes of the GMM.

Part 1 – Density-Peak Finding

Theorem: For n large enough, with high probability, $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{k}$ contains unique estimates for all the true modes of the GMM.

Relaxing the cut-off levels on the density $\rho(\boldsymbol{x})$ and distance $\delta(\boldsymbol{x})$, $\mathcal{E}_0$ is guaranteed to include all density peaks in the data.

Part 1 – Density-Peak Finding

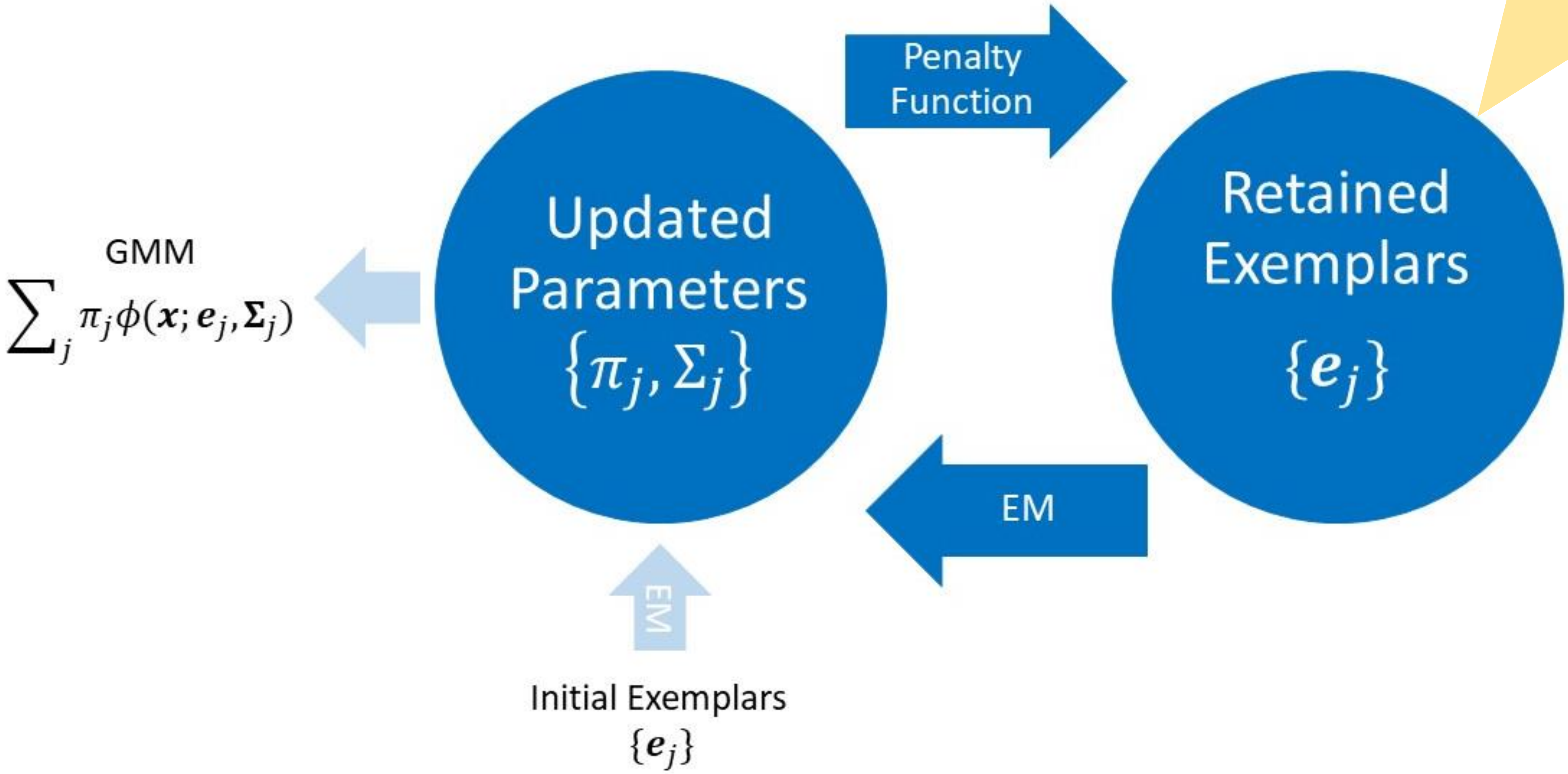
Theorem: For n large enough, with high probability, $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{k}$ contains unique estimates for all the true modes of the GMM.

Relaxing the cut-off levels on the density $\rho(\boldsymbol{x})$ and distance $\delta(\boldsymbol{x})$, $\mathcal{E}_0$ is guaranteed to include all density peaks in the data.

Apply a pruning strategy to retain only instances that well represent their associated Gaussian components.

Part 2 – Exemplar Pruning

When prune the exemplars $\{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the covariance estimates $\{\boldsymbol{\Sigma}_j\}_{j=1}^{\kappa}$ are fixed.



Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\sum_{i=1}^{n} \log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right).$$

Introduce sparsity into $\boldsymbol{\pi} = (\pi_1, \dots, \pi_\kappa)$; if $\pi_j = 0$, then the exemplar $\boldsymbol{e}_j$ is dismissed as cluster centre.

Objective function:

simplified log-likelihood + cardinality penalty of $\boldsymbol{\pi}$

Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\sum_{i=1}^{n} \log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right).$$

Introduce sparsity into $\boldsymbol{\pi} = (\pi_1, \dots, \pi_\kappa)$; if $\pi_j = 0$, then the exemplar $\boldsymbol{e}_j$ is dismissed as cluster centre.

Objective function:

simplified log-likelihood + cardinality penalty of $\boldsymbol{\pi}$

Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\sum_{i=1}^{n} \boxed{\log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right)}.$$

By Jensen's inequality we have

$$-\boxed{\log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right)} \leq \sum_{j=1}^{\kappa} r_{ij} \log\left(\frac{r_{ij}}{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)}\right).$$

r_{ij}'s are the responsibilities in the EM algorithm: $\pi_j = \frac{1}{n}\sum_{i=1}^{n} r_{ij}$ and $\sum_{j=1}^{\kappa} r_{ij}$=1.

Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\boxed{\sum_{i=1}^{n}} \log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right).$$

Therefore, the negative log-likelihood is

$$\boxed{\sum_{i=1}^{n}} - \log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right) \le \sum_{i=1}^{n}\sum_{j=1}^{\kappa} r_{ij} \log\left(\frac{r_{ij}}{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)}\right)$$

$$= \min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n}\sum_{j=1}^{\kappa} r_{ij} \log\left(\frac{r_{ij}}{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)}\right).$$

Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\sum_{i=1}^{n} \log\left(\sum_{j=1}^{\kappa} \pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)\right).$$

Minimizing the negative log-likelihood is equivalent to

$$\min_{\{\boldsymbol{\Sigma}_j \succ 0\}_{j=1}^{\kappa}} \min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \sum_{j=1}^{\kappa} r_{ij} \log\left(\frac{r_{ij}}{\pi_j \phi(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j)}\right),$$

which is

$$\min_{\{\boldsymbol{\Sigma}_j \succ 0\}_{j=1}^{\kappa}} \min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \sum_{j=1}^{\kappa} r_{ij} \left[\log\left(\frac{r_{ij}}{\pi_j}\right) + \frac{1}{2}\log(|\Sigma_j|) + \frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{e}_j)^T \Sigma_j^{-1} (\boldsymbol{x}_i - \boldsymbol{e}_j)\right]$$

Part 2 – Exemplar Pruning

$$\cancel{\min_{\{\boldsymbol{\Sigma}_j \succ 0\}_{j=1}^{\kappa}}} \min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \sum_{j=1}^{\kappa} r_{ij} \left[\cancel{\log} \left(\frac{\cancel{r}_{ij}}{\cancel{\pi}_j} \right) + \frac{1}{2} \log(|\Sigma_j|) + \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{e}_j)^T \Sigma_j^{-1} (\boldsymbol{x}_i - \boldsymbol{e}_j) \right]$$

- $\cancel{\min_{\{\boldsymbol{\Sigma}_j \succ 0\}_{j=1}^{\kappa}}}$: the covariance estimates $\{\boldsymbol{\Sigma}_j\}_{j=1}^{\kappa}$ are fixed when pruning.
- $\cancel{\log}\left(\frac{\cancel{r}_{ij}}{\cancel{\pi}_j}\right)$: numerical algorithms will behave erratically for any $\pi_j \to 0$.

Our simplified log-likelihood is

$$\min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \sum_{j=1}^{\kappa} r_{ij} \left[\frac{1}{2} \log(|\Sigma_j|) + \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{e}_j)^T \Sigma_j^{-1} (\boldsymbol{x}_i - \boldsymbol{e}_j) \right].$$

Part 2 – Exemplar Pruning

Given $\mathcal{E}_0 = \{\boldsymbol{e}_j\}_{j=1}^{\kappa}$, the log-likelihood function is

$$\sum_{i=1}^{n} \log\left(\sum_{j=1}^{\kappa} \pi_j \phi\left(\boldsymbol{x}_i; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j\right)\right).$$

Introduce sparsity into $\boldsymbol{\pi} = (\pi_1, \dots, \pi_\kappa)$; if $\pi_j = 0$, then the exemplar $\boldsymbol{e}_j$ is dismissed as cluster centre.

Objective function:

simplified log-likelihood + cardinality penalty of $\boldsymbol{\pi}$

Part 2 – Exemplar Pruning

The classical ℓ_1-norm penalty is not suitable here: $\| \boldsymbol{\pi} \|_1 = 1$ is constant on the simplex.

Our penalty is in the form of $\| \boldsymbol{\omega} \circ \boldsymbol{\pi} \|_1$, where $\circ$ is the element-wise multiplication operator.

The weight vector $\boldsymbol{\omega}$ should be data-driven and has the desirable property that gives more penalty to closer exemplars.

Part 2 – Exemplar Pruning

The weight vector $\boldsymbol{\omega} = (\omega_1, \dots, \omega_\kappa)$ is computed as

$$\omega_i = \max_{j=1,\dots,\kappa} \left\{ \Pr\left(\pi_i \phi(X; \boldsymbol{e}_i, \boldsymbol{\Sigma}_i) < \pi_j \phi\left(X; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j\right) \middle| X \sim N(\boldsymbol{e}_i, \boldsymbol{\Sigma}_i) \right) \right\}.$$

Interpretation: the weight ω_i

(1) measures the likelihood that an instance from the ith mixture component is misclassified (into another mixture component),

(2) reflects the overlapping degree between the component distribution of $\boldsymbol{e}_i$ and the other component distributions.

Part 2 – Exemplar Pruning

The weight vector $\boldsymbol{\omega} = (\omega_1, \dots, \omega_\kappa)$ is computed as

$$\omega_i = \max_{j=1,\dots,\kappa} \left\{ \Pr\left(\pi_i \phi(X; \boldsymbol{e}_i, \boldsymbol{\Sigma}_i) < \pi_j \phi\left(X; \boldsymbol{e}_j, \boldsymbol{\Sigma}_j\right) \middle| X \sim N(\boldsymbol{e}_i, \boldsymbol{\Sigma}_i) \right) \right\}.$$

Analytic calculation of ω_i is impractical, but numerical computation is readily done (Maitra and Melnykov, 2010).

Part 2 – Exemplar Pruning

Objective function:

simplified log-likelihood + cardinality penalty of $\boldsymbol{\pi}$

$$\min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \sum_{j=1}^{\kappa} r_{ij} \left[\frac{1}{2} \log(|\Sigma_j|) + \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{e}_j)^T \Sigma_j^{-1} (\boldsymbol{x}_i - \boldsymbol{e}_j) \right] + \theta \sum_{i=1}^{n} \boldsymbol{r}_i^T \boldsymbol{\omega}$$

Equivalent to

$$\min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \boldsymbol{r}_i^T \left(\frac{1}{2} \boldsymbol{\xi} + \frac{1}{2} \boldsymbol{d}_i + \theta \boldsymbol{\omega} \right),$$

extremely simple (linear and separable).

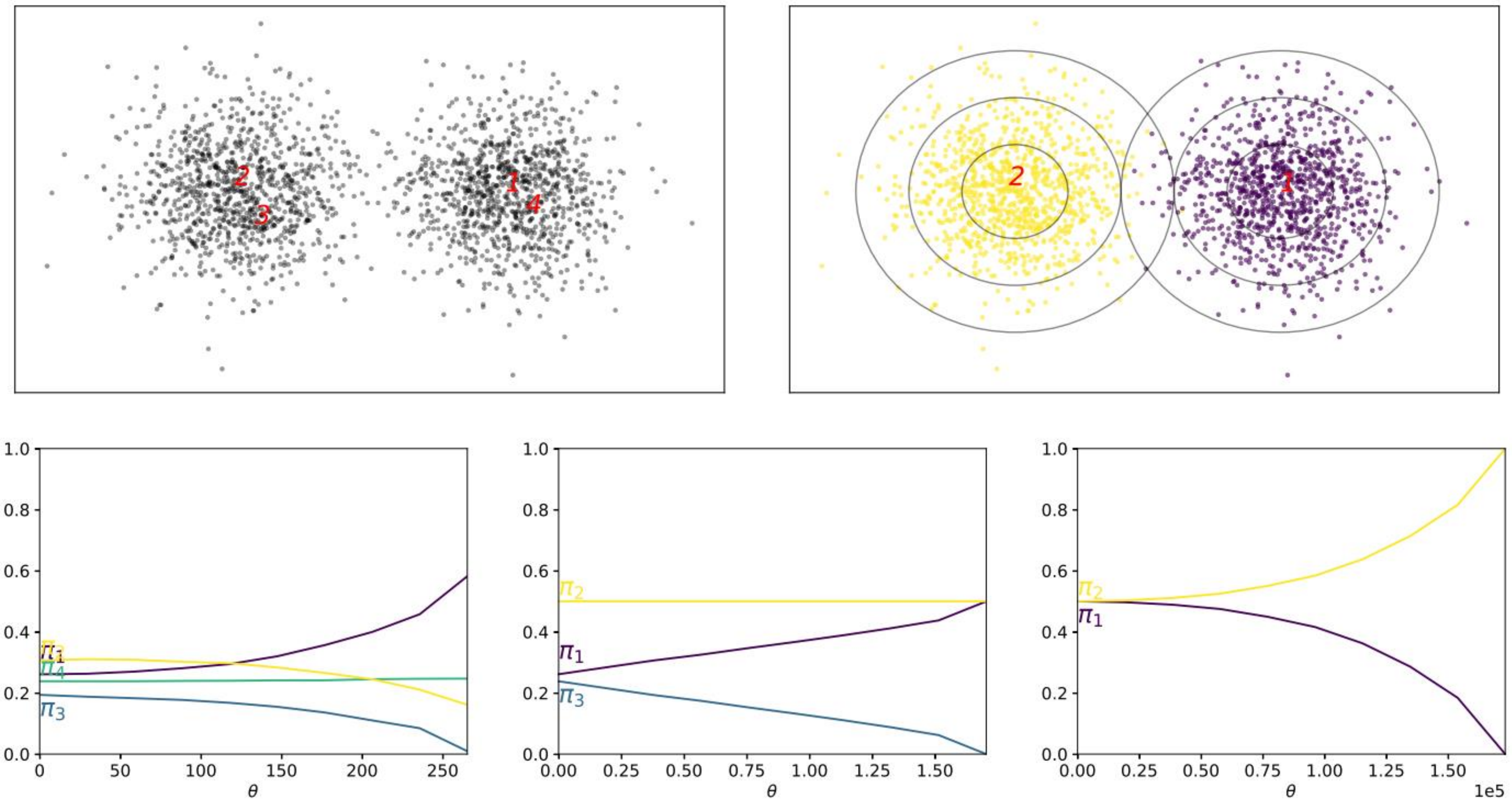
Part 2 – Exemplar Pruning

Objective function:

$$\min_{\{\boldsymbol{r}_i \in \triangle\}_{i=1}^{n}} \sum_{i=1}^{n} \boldsymbol{r}_i^T \left(\frac{1}{2}\boldsymbol{\xi} + \frac{1}{2}\boldsymbol{d}_i + \theta\boldsymbol{\omega}\right).$$

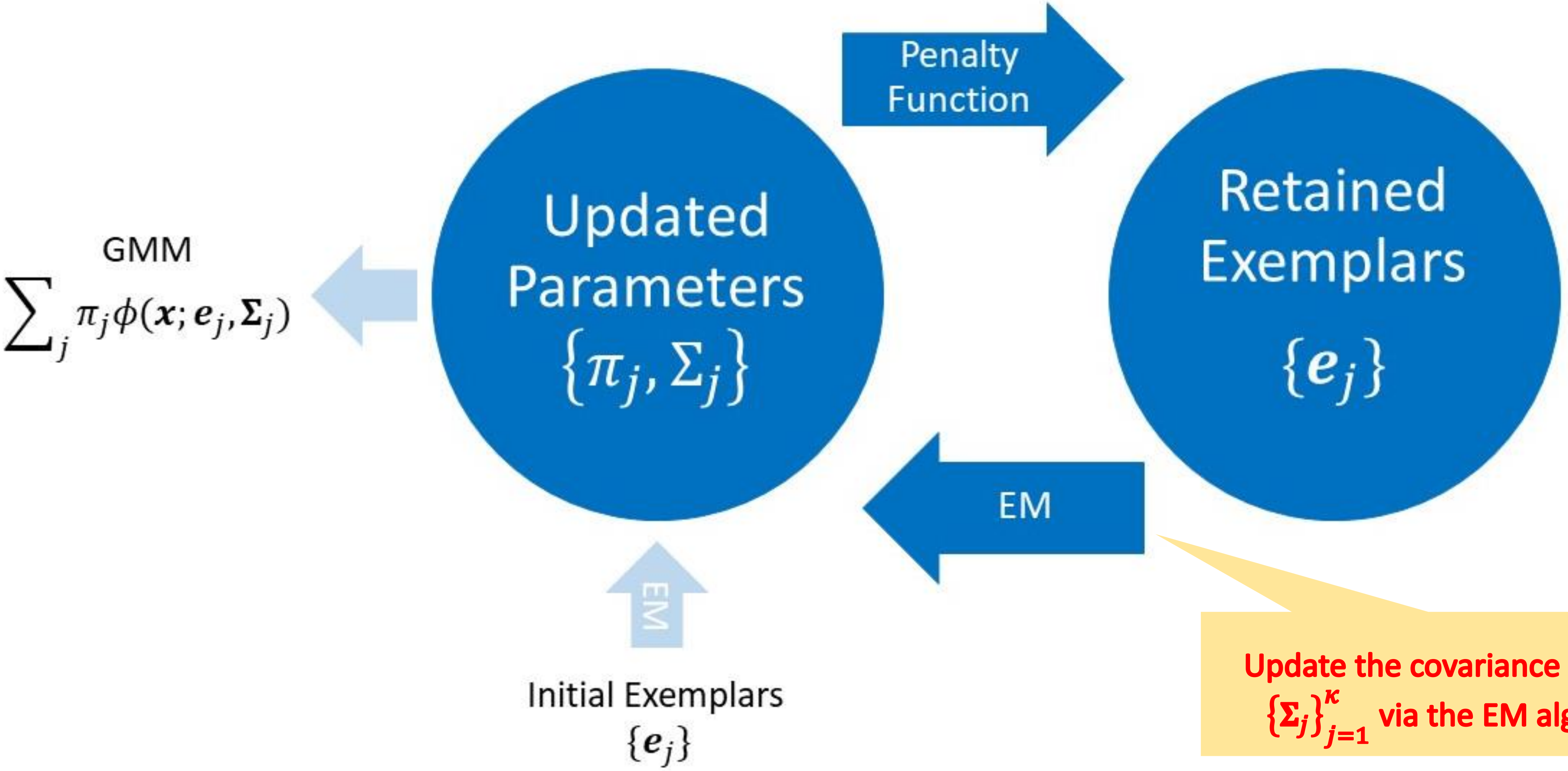
(1) The parameter θ controls the amount of shrinkage on $\boldsymbol{\pi}$ ($= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{r}_i$).

(2) The trajectory of $\boldsymbol{\pi}$, as a function of θ, can be easily computed by piecewise-linear homotopy methods.



Top Left: The data and the four selected exemplars, labelled in decreasing order of $\rho\delta$. **Top Right**: The final clustering obtained by REM. **Bottom**: The whole trajectory of $\boldsymbol{\pi}$, as a function of θ, in each REM iteration. After the first iteration, exemplar $\boldsymbol{e}_3$ is pruned; after the second iteration, exemplar $\boldsymbol{e}_4$ is pruned. The bottom right panel shows that θ needs to be very large to merge two true cluster centres.

Part 2 – Exemplar Pruning



Update the covariance estimates $\{\boldsymbol{\Sigma}_j\}_{j=1}^{\kappa}$ via the EM algorithm.

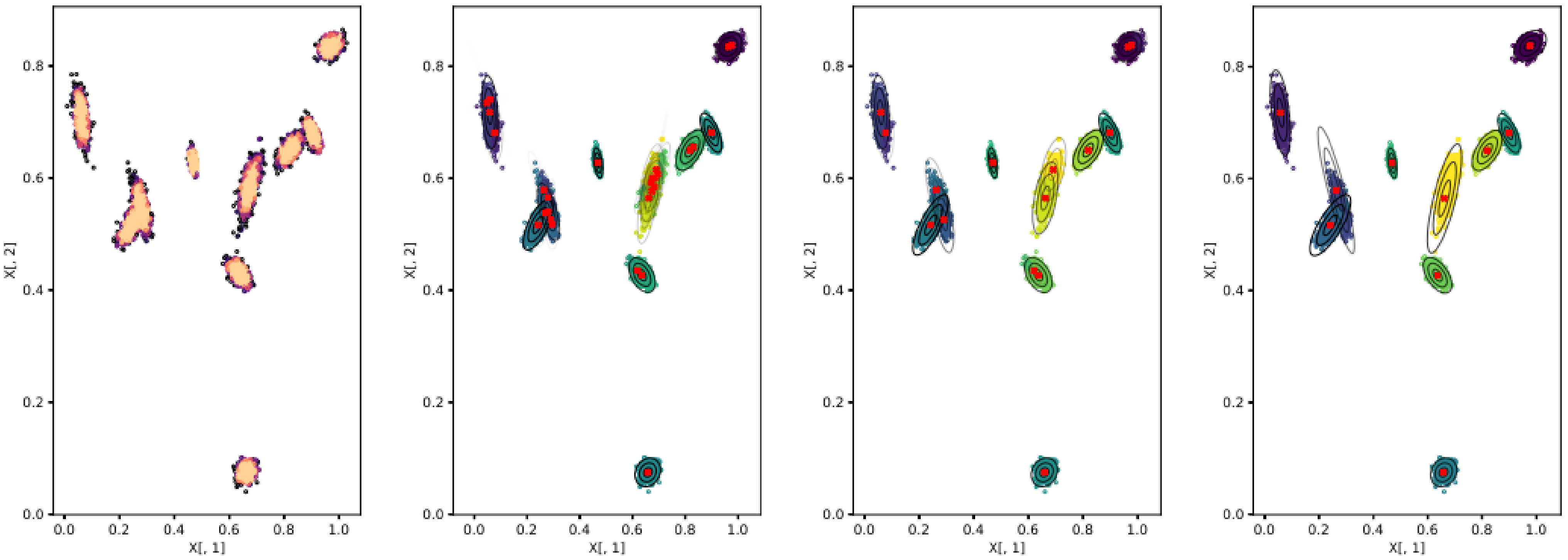


Fig. 4: A worked example of the clustering process for a 2-dimensional dataset with 10 components. (1) The leftmost figure shows the kernel density estimate for each instance, with lighter colors representing instances of higher density. (2) The second figure shows the initial exemplars (in red) with confidence ellipses representing the initial covariance matrix for each exemplar. (3) The third figure shows an intermediate clustering step, when multiple exemplars have been pruned from the initial set. (4) The rightmost figure shows the optimal clustering selected from the sequence using the ICL criterion.

Evaluation

Extensive experimental studies on both synthetic and real datasets can be found in our SDM 2023 publication:

- Tobin, J., Ho, C.P., Zhang, M.: Reinforced EM algorithm for clustering with Gaussian mixture models, Proceedings of the 2023 SIAM International Conference on Data Mining (SDM), 2023.

Reference

- [**Exemplar**] Lashkari, D., Golland, P.: Convex clustering with exemplar-based models. In: Advances in Neural Information Processing Systems 20 (NIPS 2007), pp. 825–832
- [**Exemplar**] Pilanci, M., Ghaoui, L.E., Chandrasekaran, V.: Recovery of sparse probability measures via convex programming. In: Advances in Neural Information Processing Systems 25 (NIPS 2012), pp. 2420–2428
- [**Peak Finding**] Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science (New York, N.Y.) 344(6191), 1492–1496 (2014)
- [**Overlapping Degree**] Maitra, R., Melnykov, V.: Simulating data to study performance of finite mixture modeling and clustering algorithms. Journal of Computational and Graphical Statistics, vol. 19, no. 2, 2010, pp. 354–76.