Robust EM algorithm for model-based curve clustering

Faicel Chamroukhi

Information Sciences and Systems Laboratory, UMR CNRS 7296
Southern University of Toulon-Var, France

email: chamroukhi@univ-tln.fr
web: chamroukhi.univ-tln.fr

IJCNN 2013, Dallas, Texas

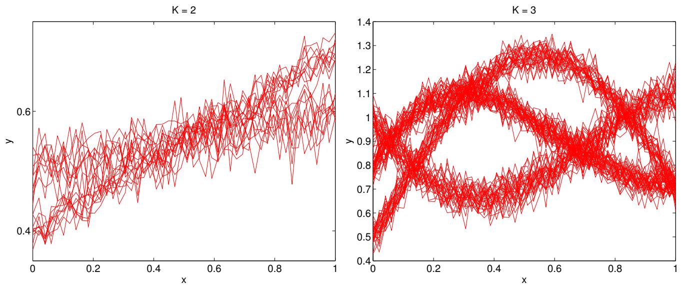
5 August 2013

Outline

1. Context and objective
2. Model-based clustering with GMMs
3. Model-based curve clustering with regression mixtures
4. Proposed robust EM algorithm for model-based curve clustering
5. Simulation study
6. Conclusion

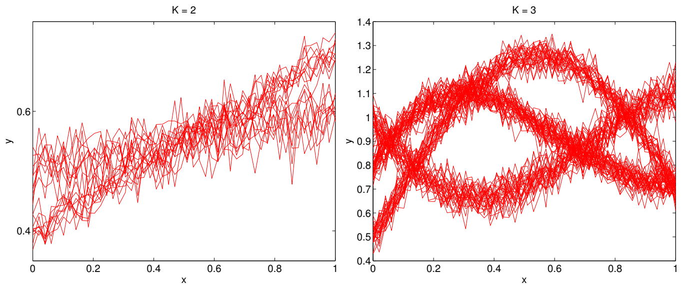
Context

- Curve clustering framework: the data are curves or functions rather than vectors (functional data analysis framework)



Context

- Curve clustering framework: the data are curves or functions rather than vectors (functional data analysis framework)



Objective

- Learn a probabilistic generative model for model-based curve clustering
- Deal with the problems of choosing the number of clusters and algorithm initialization

Model-based clustering

Model-based clustering

- The aim of clustering in general is to find a partition of an unlabeled dataset into clusters (groups)
- the data within the same group tend to be more similar, in the sense of a chosen dissimilarity measure, to one another as compared to the data belonging to different groups
- Model-based clustering techniques: rely on the finite mixture model formulation [7]
- They are one of the most popular and successful approaches in cluster analysis.
- The mixture density estimation is generally performed by maximizing the observed-data log-likelihood by using the expectation-maximization (EM) algorithm.

Model-based clustering of multivariate data using GMMs

Model-based clustering [1, 8, 4], generally used for multidimensional data, is based on a finite mixture model formulation [7]:

- $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ an observed i.i.d dataset, $\mathbf{x}_i$ is represented a multidimensional vector in $\mathbb{R}^d$.
- $\mathbf{z} = (z_1, \ldots, z_n)$ the hidden cluster labels ($z_i \in \{1, \ldots, K\}$, K clusters).
- The finite Gaussian mixture density is defined as:

$$f(\mathbf{x}_i; \mathbf{\Psi}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \tag{1}$$

- ⇒ The problem of clustering therefore becomes the one of estimating the parameters of the Gaussian mixture model $\mathbf{\Psi} = (\pi_1, \ldots, \pi_K, \mathbf{\Psi}_1, \ldots, \mathbf{\Psi}_K)$
- ⇒ This can be performed by maximizing the f observed-data log-likelihood of $\mathbf{\Psi}$ by using the EM algorithm [6, 3]

- However, these presented model-based clustering approaches are concerned with vectorial data where the observations are assumed to be vectors of reduced dimension.
- $\Rightarrow$ When the data are rather curves or functions, one can rely on regression mixtures which are more adapted for curves than standard Gaussian mixture modeling.

Model-based curve clustering

- The aim curve clustering is to cluster n iid unlabeled curves $((\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n))$ into K clusters
- We assume that each curve consists of m observations $\mathbf{y}_i = (y_{i1}, \ldots, y_{im})$ regularly observed at the inputs $\mathbf{x}_i = (x_{i1}, \ldots, x_{im})$
- $\Rightarrow$ find the unknown cluster labels $\mathbf{z} = (z_1, \ldots, z_n)$, with $z_i \in \{1, \ldots, K\}$, K being the number of clusters
- $\Rightarrow$ the curve clustering can be performed based on regression mixture models including polynomial regression mixtures (PRM) and polynomial spline regression mixtures (PSRM) [5, 2].

Regression mixtures for model-based curve clustering

- The regression mixture models assume that each curve is drawn from one of K clusters whose proportions are $(\pi_1, \ldots, \pi_K)$.
- Each cluster is modeled by a polynomial (spline) regression model
- The regression mixture density of the ith curve can be written as:

$$f(\mathbf{y}_i|\mathbf{x}_i;\boldsymbol{\Psi}) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m). \quad (2)$$

- model parameters: $\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_K, \boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_K)$: where $\boldsymbol{\Psi}_k = (\boldsymbol{\beta}_k, \sigma_k^2)$ are respectively the regression coefficients and the noise variance

Regression mixtures for model-based curve clustering

- The regression mixture models assume that each curve is drawn from one of K clusters whose proportions are $(\pi_1,\ldots,\pi_K)$.
- Each cluster is modeled by a polynomial (spline) regression model
- The regression mixture density of the ith curve can be written as:

$$f(\mathbf{y}_i|\mathbf{x}_i;\boldsymbol{\Psi}) = \sum_{k=1}^{K} \pi_k \; \mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m). \tag{2}$$

- model parameters: $\boldsymbol{\Psi} = (\pi_1,\ldots,\pi_K,\boldsymbol{\Psi}_1,\ldots,\boldsymbol{\Psi}_K)$: where $\boldsymbol{\Psi}_k = (\boldsymbol{\beta}_k,\sigma_k^2)$ are respectively the regression coefficients and the noise variance

- The parameter vector $\boldsymbol{\Psi}$ is estimated by maximizing the log-likelihood:

$$\mathcal{L}(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \; \mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m). \tag{3}$$

- maximization can be performed iteratively via the EM algorithm [5, 3, 2].

Limitations

1. However, it can be noticed that, the standard EM algorithm for regression mixture model is sensitive to initialization
2. In addition, it requires the number of clusters to be supplied by the user. While the number of cluster can be chosen by some model selection criteria, this requires performing additional model selection procedure.

Limitations

1. However, it can be noticed that, the standard EM algorithm for regression mixture model is sensitive to initialization
2. In addition, it requires the number of clusters to be supplied by the user. While the number of cluster can be chosen by some model selection criteria, this requires performing additional model selection procedure.

In general, theses two issues have been considered each separately

- Initialization techniques: randomly, K-means, CEM, few runs of EM, etc
- Choosing the number of clusters is an afterward model selection problem: BIC, AIC, ICL, etc

Limitations

1. However, it can be noticed that, the standard EM algorithm for regression mixture model is sensitive to initialization
2. In addition, it requires the number of clusters to be supplied by the user. While the number of cluster can be chosen by some model selection criteria, this requires performing additional model selection procedure.

In general, theses two issues have been considered each separately

- Initialization techniques: randomly, K-means, CEM, few runs of EM, etc
- Choosing the number of clusters is an afterward model selection problem: BIC, AIC, ICL, etc

- ⇒ Here we attempt to overcome these limitations in this case of model-based curve clustering:
- ⇒ We propose an EM algorithm which is robust with regard initialization and automatically estimate the number of clusters as the learning proceeds

Penalized maximum likelihood estimation

- For estimating the regression mixture model ⇒ maximize a penalized log-likelihood function rather than its standard log-likelihood (3)
- penalize the log-likelihood by a term accounting for the model complexity

Penalized maximum likelihood estimation

- For estimating the regression mixture model ⇒ maximize a penalized log-likelihood function rather than its standard log-likelihood (3)
- penalize the log-likelihood by a term accounting for the model complexity

Regularization term

- As the model complexity is mainly governed by the number of clusters (the hidden variables z_i) ⇒ use as penalty the entropy of the hidden variable z_i
- The (differential) entropy of one variable ($z_i \in \{1,\ldots,K\}$) is defined by

$$H(z_i) = -\mathbb{E}[\log p(z_i)] = -\sum_{k=1}^{K} p(z_i = k)\log p(z_i = k) = -\sum_{k=1}^{K} \pi_k \log \pi_k. \quad (4)$$

- The variables $\mathbf{z} = (z_1,\ldots,z_n)$ are i.i.d, ⇒ the whole entropy for $\mathbf{z}$ is:

$$H(\mathbf{z}) = -\sum_{i=1}^{n}\sum_{k=1}^{K} \pi_k \log \pi_k. \quad (5)$$

Penalized maximum likelihood estimation

- The proposed objective function we propose to maximize is thus constructed by penalizing the observed data log-likelihood (3) by the entropy term (5), that is:

$$\begin{aligned}\mathcal{J}(\lambda,\boldsymbol{\Psi}) &= \mathcal{L}(\boldsymbol{\Psi}) - \lambda H(\mathbf{z}), \quad \lambda \geq 0 \\ &= \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{y}_i; \mathbf{X}_i \boldsymbol{\beta}_k, \sigma_k^2 \mathbf{I}_m) + \lambda \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_k \log \pi_k \end{aligned} \quad (6)$$

- $\mathcal{L}(\boldsymbol{\Psi})$ is the observed-data log-likelihood maximized by the standard EM algorithm for regression mixtures (see Equation (3))
- When the entropy is large, the fitted model is rougher, and when it is small, the fitted model is smoother.
- $\lambda \geq 0$ is a smoothing parameter for establishing a trade-off between closeness of fit to the data and a smooth fit

- the model parameters $\boldsymbol{\Psi}$ are estimated by maximizing the penalized observed-data log-likelihood (6)

$$\mathscr{J}(\lambda, \boldsymbol{\Psi})$$

given an i.i.d training dataset of n curves $\mathscr{D} = ((\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n))$

- $\mathscr{J}(\lambda, \boldsymbol{\Psi})$ is iteratively maximized by using a dedicated EM algorithm

- the model parameters $\mathbf{\Psi}$ are estimated by maximizing the penalized observed-data log-likelihood (6)

$$\mathscr{J}(\lambda, \mathbf{\Psi})$$

given an i.i.d training dataset of n curves $\mathscr{D} = ((\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n))$

- $\mathscr{J}(\lambda, \mathbf{\Psi})$ is iteratively maximized by using a dedicated EM algorithm

⇒ The complete-data log-likelihood of $\mathbf{\Psi}$ in this penalized case is given y:

$$\mathscr{J}_c(\lambda, \mathbf{\Psi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log \left[\pi_k \mathscr{N}(\mathbf{y}_i; \mathbf{X}_i \boldsymbol{\beta}_k, \sigma_k^2 \mathbf{I}_m)\right] + \lambda \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_k \log \pi_k \cdot (7)$$

- z_{ik} is an indicator binary variable such that $z_{ik} = 1$ iff $z_i = k$ (i.e., if the ith curve $\mathbf{x}_i$ is generated by cluster k)

Robust EM algorithm for model-based curve clustering using regression mixtures

Start with an initial solution (parameter $\mathbf{\Psi}^{(0)}$ and a number of clusters K)

Robust EM algorithm for model-based curve clustering using regression mixtures

Start with an initial solution (parameter $\mathbf{\Psi}^{(0)}$ and a number of clusters K)

1. **E-step** Compute the expected penalized complete-data log-likelihood (7)

$$Q(\lambda,\mathbf{\Psi};\mathbf{\Psi}^{(q)}) = \mathbb{E}\big[\mathscr{J}_c(\lambda,\mathbf{\Psi})|\mathscr{D};\mathbf{\Psi}^{(q)}\big] \quad (8)$$

$$= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(q)}\log\big[\pi_k\mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m)\big]+\lambda\sum_{i=1}^{n}\sum_{k=1}^{K}\pi_k\log\pi_k$$

Robust EM algorithm for model-based curve clustering using regression mixtures

Start with an initial solution (parameter $\boldsymbol{\Psi}^{(0)}$ and a number of clusters K)

1. **E-step** Compute the expected penalized complete-data log-likelihood (7)

$$\begin{aligned} Q(\lambda,\boldsymbol{\Psi};\boldsymbol{\Psi}^{(q)}) &= \mathbb{E}\big[\mathcal{J}_c(\lambda,\boldsymbol{\Psi})|\mathcal{D};\boldsymbol{\Psi}^{(q)}\big] \qquad (8)\\ &= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(q)}\log\big[\pi_k\mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m)\big]+\lambda\sum_{i=1}^{n}\sum_{k=1}^{K}\pi_k\log\pi_k \end{aligned}$$

⇒ simply consists in computing the posterior cluster probabilities:

$$\tau_{ik}^{(q)}=\frac{\pi_k^{(q)}\mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k^{(q)},\sigma_k^{2(q)}\mathbf{I}_m)}{\sum_{h=1}^{K}\pi_h^{(q)}\mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_h^{(q)},\sigma_h^{2(q)}\mathbf{I}_m)}. \qquad (9)$$

Robust EM algorithm for model-based curve clustering using regression mixtures

Start with an initial solution (parameter $\mathbf{\Psi}^{(0)}$ and a number of clusters K)

1. **E-step** Compute the expected penalized complete-data log-likelihood (7)

$$\begin{aligned} Q(\lambda, \mathbf{\Psi}; \mathbf{\Psi}^{(q)}) &= \mathbb{E}[\mathcal{J}_c(\lambda, \mathbf{\Psi})|\mathcal{D}; \mathbf{\Psi}^{(q)}] \qquad (8)\\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} \tau_{ik}^{(q)} \log\left[\pi_k \mathcal{N}(\mathbf{y}_i; \mathbf{X}_i\boldsymbol{\beta}_k, \sigma_k^2 \mathbf{I}_m)\right] + \lambda \sum_{i=1}^{n}\sum_{k=1}^{K} \pi_k \log \pi_k \end{aligned}$$

⇒ simply consists in computing the posterior cluster probabilities:

$$\tau_{ik}^{(q)} = \frac{\pi_k^{(q)} \mathcal{N}(\mathbf{y}_i; \mathbf{X}_i\boldsymbol{\beta}_k^{(q)}, \sigma_k^{2(q)} \mathbf{I}_m)}{\sum_{h=1}^{K} \pi_h^{(q)} \mathcal{N}(\mathbf{y}_i; \mathbf{X}_i\boldsymbol{\beta}_h^{(q)}, \sigma_h^{2(q)} \mathbf{I}_m)}. \qquad (9)$$

2. **M-step** Updating step: $\mathbf{\Psi}^{(q+1)} = \arg\max_{\mathbf{\Psi}} Q(\lambda, \mathbf{\Psi}; \mathbf{\Psi}^{(q)})$.

1. The mixing proportions updates are obtained by maximizing the function

$$Q_\pi(\lambda;\boldsymbol{\Psi}^{(q)}) = \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(q)}\log\pi_k + \lambda\sum_{i=1}^{n}\sum_{k=1}^{K}\pi_k\log\pi_k$$

⇒ This can be solved using Lagrange multipliers :

$$\pi_k^{(q+1)} = \frac{1}{n}\sum_{i=1}^{n}\tau_{ik}^{(q)} + \lambda\pi_k^{(q)}\left(\log\pi_k^{(q)} - \sum_{h=1}^{K}\pi_h^{(q)}\log\pi_h^{(q)}\right) \qquad (10)$$

1. The mixing proportions updates are obtained by maximizing the function

$$Q_\pi(\lambda;\mathbf{\Psi}^{(q)}) = \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(q)}\log\pi_k + \lambda\sum_{i=1}^{n}\sum_{k=1}^{K}\pi_k\log\pi_k$$

⇒ This can be solved using Lagrange multipliers :

$$\pi_k^{(q+1)} = \frac{1}{n}\sum_{i=1}^{n}\tau_{ik}^{(q)} + \lambda\pi_k^{(q)}\left(\log\pi_k^{(q)} - \sum_{h=1}^{K}\pi_h^{(q)}\log\pi_h^{(q)}\right) \quad (10)$$

2. The regression parameters for each class k are updated by maximizing

$$Q_{\mathbf{\Psi}_k}(\lambda,\boldsymbol{\beta}_k,\sigma_k^2;\mathbf{\Psi}^{(q)}) = \sum_{i=1}^{n}\tau_{ik}^{(q)}\log\mathcal{N}(\mathbf{y}_i;\mathbf{X}_i\boldsymbol{\beta}_k,\sigma_k^2\mathbf{I}_m)$$

⇒ consists in analytic solutions of K weighted least-squares problems:

$$\boldsymbol{\beta}_k^{(q+1)} = \Big[\sum_{i=1}^{n}\tau_{ik}^{(q)}\mathbf{X}_i^T\mathbf{X}_i\Big]^{-1}\sum_{i=1}^{n}\tau_{ik}^{(q)}\mathbf{X}_i^T\mathbf{y}_i \quad \sigma_k^{2(q+1)} = \frac{\sum_{i=1}^{n}\tau_{ik}^{(q)}\|\mathbf{y}_i-\mathbf{X}_i\boldsymbol{\beta}_k\|^2}{m\sum_{i=1}^{n}\tau_{ik}^{(q)}} \quad (11)$$

- for very small value of λ: the update of the mixing proportions is close to the one in the standard EM update
- however for a large value of λ : the penalization term will play its role in order to make clusters competitive ⇒ allows for discarding illegitimate clusters and enhancing actual clusters This depends on

$$\left(\log \pi_k^{(q)} - \sum_{h=1}^{K} \pi_h^{(q)} \log \pi_h^{(q)}\right) > or < 0$$

- a cluster k can be discarded if its proportion is less than $\frac{1}{n}$, $(\pi_k^{(q)} < \frac{1}{n})$.
- Finally, the penalization coefficient λ is set in an adaptive way to be large for enhancing competition

$$\lambda^{(q+1)} = \left\{ \frac{\sum_{k=1}^{K} \exp\left(\eta n |\pi_k^{(q+1)} - \pi_k^{(q)}|\right)}{K}, \frac{1 - \max_{k=1}^{K}\left(\frac{\sum_{i=1}^{n} \tau_{ik}^{(q)}}{n}\right)}{-\pi_k^{(q)} \sum_{k=1}^{K} \pi_k^{(q)} \log \pi_k^{(q)}} \right\} \quad (12)$$

Initialization strategy and stopping rule

Initialization

- initialization of the number of clusters : $K^{(0)} = n$, with n the number of curves,
- initialization of the mixing proportions : $\pi_k^{(0)} = \frac{1}{K^{(0)}}$, $(k = 1, \ldots, K^{(0)})$,
- to initialize the regression parameters $\boldsymbol{\beta}_k$ and the noise variances $\sigma_k^{2(0)}$, fit a polynomial regression model to each curve $k \Rightarrow \boldsymbol{\Psi}_k^{(0)} = (\boldsymbol{\beta}_k^{(0)}, \sigma_k^{2(0)})$.

Stopping rule

The proposed EM algorithm is stopped when the maximum variation of the estimated regression parameters between two iterations $\max_{1 \leq k \leq K^{(q)}} \|\boldsymbol{\beta}_k^{(q+1)} - \boldsymbol{\beta}_k^{(q)}\|$ is less than a threshold ϵ (e.g., 10^{-6}).

Evaluation on simulated curves

- Evaluation of the proposed approach on simulated curves.
- Simulated linear and non-linear arbitrary curves (not simulated according to the model)
- See how the performance of the proposed robust EM algorithm with regard to finding the correct number of clusters and the actual partition

Situation 1: Linear curves

- dataset consists of $n = 20$ curves, each curve consists of $m = 50$ observationas generated as a linear function corrupted by a Gaussian noise as follows. For the ith curve $(i = 1, \ldots, n)$, the jth observation $(j = 1, \ldots, m)$ is generated as follows:
  - $y_{ij} = 0.3\, x_{ij} + 0.4 + \sigma_1 \epsilon_{ij}$;
  - $y_{ij} = 0.1\, x_{ij} + 0.5 + \sigma_2 \epsilon_{ij}$.

  respectively for class 1 and class 2, where x are linearly equally spaced points in the range $[0,1]$, with $\sigma_1 = 0.02$ and $\sigma_2 = 0.03$ are the corresponding noise standard deviations, and $\epsilon_{ij} \sim \mathcal{N}(0,1)$ are standard zero-mean unit variance Gaussian variables.
- The two classes have the same proportion.

Situation 1: Linear curves

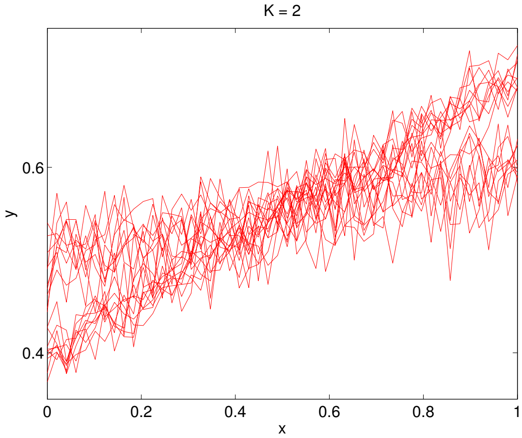


Figure : Simulated linear curves (a two-class problem)

Obtained results for situation 1

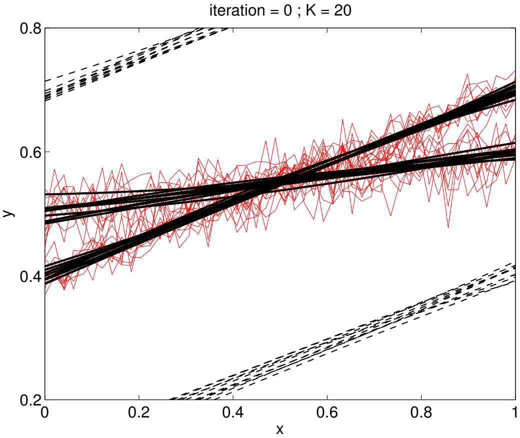


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 1$)

Obtained results for situation 1

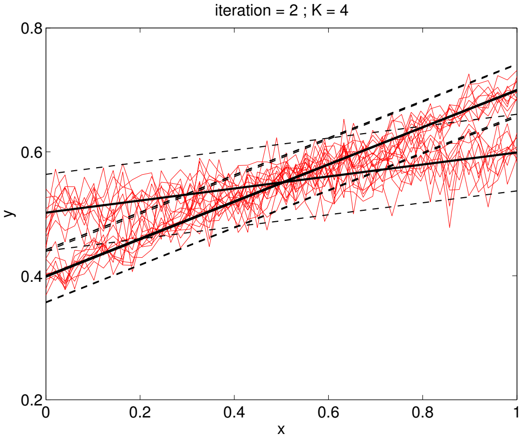


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 1$)

Obtained results for situation 1

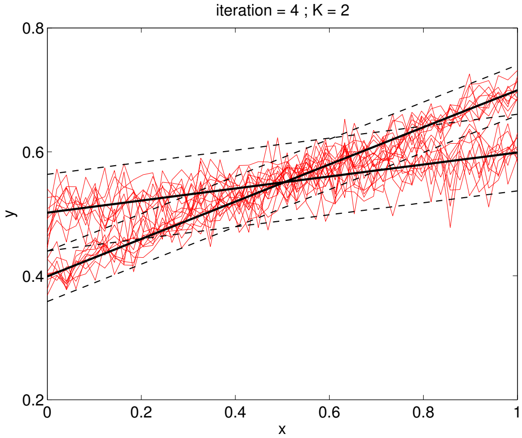


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 1$)

Situation 2: Arbitrary curves

- a three-class problem
- dataset of $n = 100$ arbitrary non-linear curves, each curve consists of $m = 50$ observations generated as follows. For the ith curve $(i = 1, \ldots, n)$, the jth observation $(j = 1, \ldots, m)$ is generated as follows:
  - $y_{ij} = 0.8 + 0.5\exp(-1.5x)\sin(1.3\pi x) + \sigma_1 \epsilon_{ij}$;
  - $y_{ij} = 0.5 + 0.8\exp(-x)\sin(0.9\pi x) + \sigma_2 \epsilon_{ij}$;
  - $y_{ij} = 1 + 0.5\exp(-x)\sin(1.2\pi x) + \sigma_3 \epsilon_{ij}$.

  with $\sigma_1 = 0.04$, $\sigma_2 = 0.04$ and $\sigma_3 = 0.05$.
- The classes have respectively the proportions: $\pi_1 = 0.4, \pi_2 = 0.3, \pi_3 = 0.3$.

Situation 2: Arbitrary curves

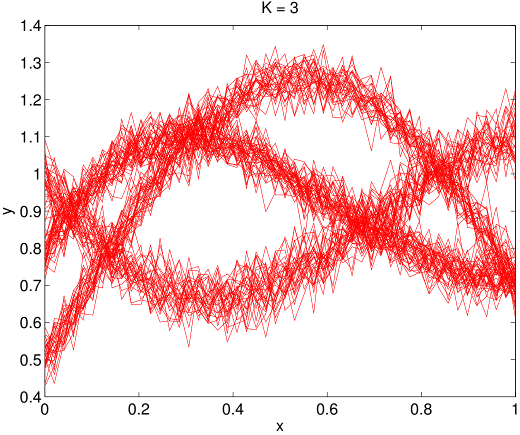


Figure : Simulated arbitrary curves for the second situation (a three-class problem)

Obtained results for situation 2

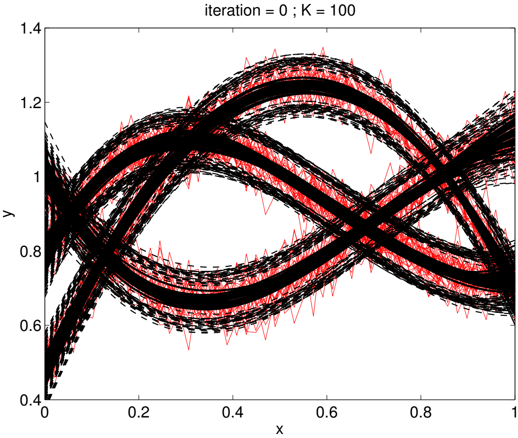


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

Obtained results for situation 2

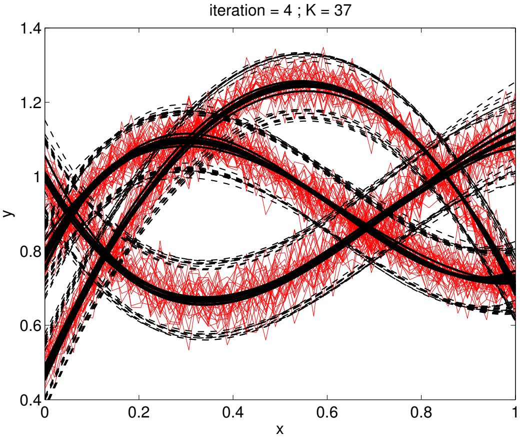


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

Obtained results for situation 2

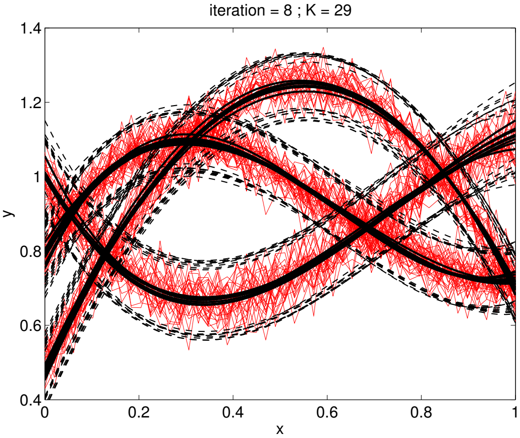


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

Obtained results for situation 2

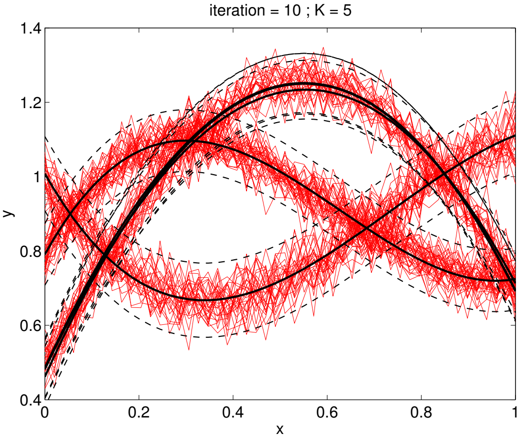


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

Obtained results for situation 2

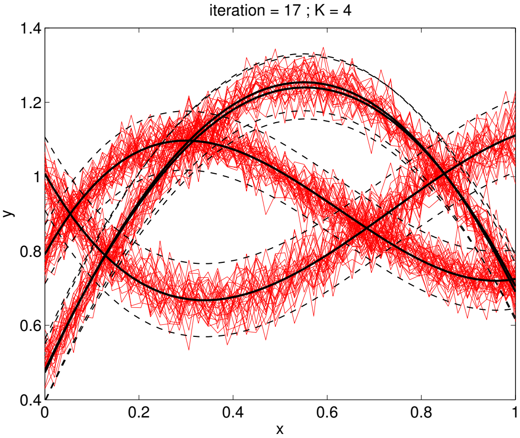


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

Obtained results for situation 2

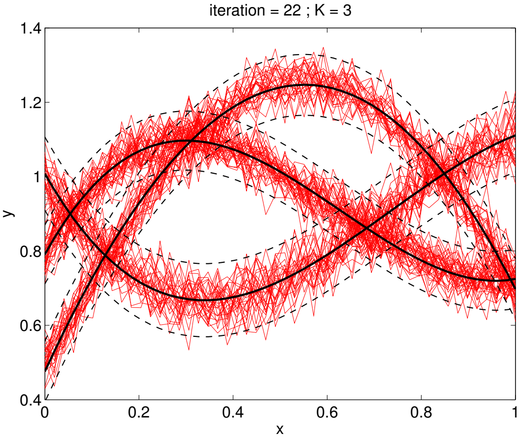


Figure : Clustering results obtained with the proposed robust EM-MixReg algorithm (with polynomial degree $p = 3$)

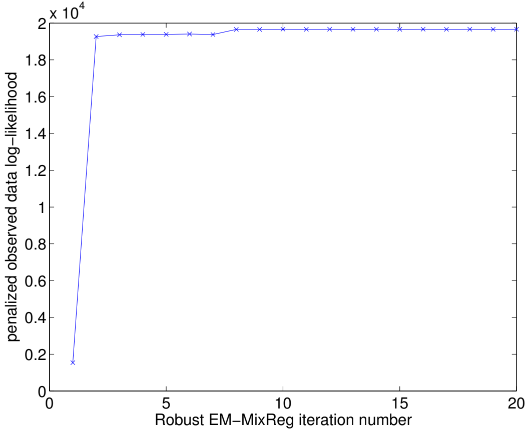


Figure : Penalized log-likelihood at each iteration for a simulated set of curves

Conclusions

- In this paper, we presented a new robust EM algorithm for model-based curve clustering.
- It optimizes a penalized observed-data log-likelihood using the entropy of the hidden structure.
- The proposed algorithm is robust with regard to both the initialization and determining the optimal number of clusters for regression mixtures.
- The experimental results on simulated data demonstrates the potential benefit of the proposed approach for curve clustering.
- Future work will concern additional experiments on real data including temporal curves.

References

[1] J. D. Banfield and A. E. Raftery. Model-based gaussian and non-gaussian clustering. Biometrics, 49(3):803–821, 1993.

[2] F. Chamroukhi. Hidden process regression for curve modeling, classification and tracking. Ph.D. thesis, Université de Technologie de Compiègne, Compiègne, France, 2010.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of The Royal Statistical Society, B, 39(1):1–38, 1977.

[4] C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis, and density estimation. Journal of the American Statistical Association, 97:611–631, 2002.

[5] S. J. Gaffney. Probabilistic Curve-Aligned Clustering and Prediction with Regression Mixture Models. PhD thesis, Department of Computer Science, University of California, Irvine, 2004.

[6] G. J. McLachlan and T. Krishnan. The EM algorithm and extensions. Wiley, 1997.

[7] G. J. McLachlan and D. Peel. Finite mixture models. Wiley, 2000.

[8] G.J. McLachlan and K.E. Basford. Mixture Models: Inference and Applications to Clustering. Marcel Dekker, New York, 1988.

[9] Miin-Shen Yang, Chien-Yo Lai, and Chih-Ying Lin. A robust em clustering algorithm for gaussian mixture models. Pattern Recognition, 45(11):3950–3961, November 2012.

Thank you!