A DAEM Algorithm for Mixtures of Gaussian Process Functional Regressions

Di Wu and Jinwen Ma(✉)

Department of Information Science, School of Mathematical Sciences and LMAM, Peking University, Beijing 100871, China
jwma@math.pku.edu.cn

Abstract. The mixture of Gaussian process functional regressions (mix-GPFR) is a powerful tool for curve clustering and prediction. Unfortunately, there generally exist a large number of local maximums for the Q-function of the conventional EM algorithm so that the conventional EM algorithm is often trapped in the local maximum. In order to overcome this problem, we propose a deterministic annealing EM (DAEM) algorithm for mix-GPFR in this paper. The experimental results on the simulated and electrical load datasets demonstrate that the DAEM algorithm outperforms the conventional EM algorithm on parameter estimation, curve clustering and prediction.

Keywords: Gaussian process · Mixture of Gaussian process · EM algorithm · Curve clustering · Deterministic annealing

1 Introduction

Gaussian process (GP) [1, 2] is a powerful tool in the fields of signal and information processing, machine learning and data mining. But the mean function of the GP model is generally assumed to be zero or a linear function of input variables. Moreover, a single GP cannot deal with a multimodality dataset. In fact, curve clustering is a typical multimodality dataset problem. Specifically, each curve can be regarded as one "sample", referred to as a sample curve or functional datum. The aim of curve clustering is to separate these sample curves into different clusters or classes which can be modeled by certain Gaussian processes. So, the actual model of curve clustering is a mixture of Gaussian processes. The mean functions of the Gaussian processes are very important for curve clustering, but they are generally assumed to be zeros or linear functions for easy computation. In literature, there are only a few methods for learning nonlinear mean functions and the Gaussian process functional regression (GPFR) model [3] provides a feasible way. The mean function of the GPFR is assumed to be a linear combination of b-spline basis functions [4]. For solving the curve clustering problem, Shi previously utilized the mixture of GPs (mix-GP) model [5–8] where the sample curves belong to one cluster are subject to a general GP. In order to improve the performance of curve clustering, the GPFR models were introduced and the mixture of GPFRs (mix-GPFR) model [9] was finally utilized.

Although the maximum likelihood estimate (MLE) of the GPFR model can be calculated by the gradient method, the computation of the MLE for the mix-GP model

D.-S. Huang et al. (Eds.): ICIC 2016, Part III, LNAI 9773, pp. 294–303, 2016.
DOI: 10.1007/978-3-319-42297-8_28

is rather difficult. To overcome this problem, the MCMC approach was applied for the mix-GP with zero mean functions. But about 20000 samples were needed and thus it might take one day time on a small dataset. In addition, the MCMC approach is quite difficult to be applied for the mix-GPFR model since the sampling of the parameters in the mean function is not so easy. Alternatively, the conventional EM algorithm [10–13] was adopted for the mix-GPFR model. Although the conventional EM algorithm has some advantages such as low cost per iteration and ease of programming, it is a local search method and cannot get rid of the local maximum problem.

In this paper, we propose a deterministic annealing EM (DAEM) algorithm for the mix-GPFR model to overcome the local maximum problem. The idea of the DAEM algorithm is to transform the Q-function of the conventional EM algorithm into the U-function which can be flexible in a deterministic annealing way. In the early iterations, the U-function is smoother, i.e., has less local maximum, than the Q-function so that the maximum of the U-function is more global than that of the Q-function. During the following iterations, the U-function gradually tends to the Q-function and the DAEM algorithm has more probable to arrive at the global maximum point. We conduct the experiments on both simulated and electrical load datasets. The experimental results demonstrate that the DAEM algorithm for the mix-GPFR model outperforms the conventional EM algorithm on parameter estimation as well as curve clustering.

The remainder of this paper is organized as follows. The GPFR and mix-GPFR models are introduced in Sect. 2. In Sect. 3, we propose the DAEM algorithm for the mix-GPFR model. The experimental results and comparisons are summarized in Sect. 4. Finally, we give a brief conclusion in Sect. 5.

2 The GPFR and Mix-GPFR Models

In this section, we introduce the GPFR model as well as the mix-GPFR model.

2.1 The GPFR Model

The GP is a common and important stochastic process in which any group of states (as random variables) are subject to a Gaussian distribution. $y(x) \in \mathbb{R}$ (the real number field) is a stochastic process, where $x \in \mathbb{R}$. With any natural number N and any vector $\mathbf{x} = (x_1, \ldots, x_N)^T$, the definition of the Gaussian process can be given as follows. If $\mathbf{y} = [y_1, \ldots, y_N]^T$, where $y_n = y(x_n)$, is subject to an N-dimensional Gaussian distribution $N(\boldsymbol{\mu}, \mathbf{C})$, then $y(x)$ is said to follow a Gaussian process, where $\boldsymbol{\mu} = [\mu(x_1), \ldots, \mu(x_N)]^T$ and $\mathbf{C} = \left[C\left(x_n, x_{n'}\right)\right]_{N \times N}$ represents an $N \times N$ kernel matrix in which $C(x_n, x_{n'})$ is a kernel function. The GP model is written as

$$y(x) \sim GP[\mu(x), C(x, x')].$$

Here, we utilize the kernel function

$$C(x_n, x_{n'}) = (\theta_1)^2 \exp\left[-\frac{1}{2}(\theta_2)^2(x_n - x_{n'})^2\right] + (\theta_3)^2 \delta_{nn'},$$

where $\delta_{nn'}$ is the Kronecker delta function and $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$.

However, the mean function of GP is generally assumed to be zero, or a linear function or quite simple nonlinear function. To learn the mean function of the GP model better, Shi proposed the model of GPFR [3]. In this model, the mean function is approximated by a linear combination of b-spline basis functions and we illustrate a set of b-basis functions in Fig. 2(a). Denote a set of b-spline basis functions, $\varphi = [\varphi_1(x), \ldots, \varphi_D(x)]^T$. Then the mean function is approximate by

$$\mu(x) = \mathbf{b}^T \varphi = \sum\nolimits_{j=1}^{D} b_j \varphi_j(x),$$

where $\mathbf{b} = (b_1, \ldots, b_D)^T$ is a D-dimensional coefficient vector. Thus, the GPFR can be described by

$$y(x) \sim \mathrm{GPFR}(x; \mathbf{b}, \boldsymbol{\theta}).$$

2.2 The Mix-GPFR Model

There is heterogeneity among the sample curves sometimes and this kind of dataset cannot be learned by a single GPFR. To overcome this problem, Shi [9] proposed the mix-GPFR model. The M curves generated by mix-GPFR could be separated into K components or classes and the curves belong to each component is subject to a same GPFR model. The mix-GPFR is a powerful model for curve clustering and the detail of mix-GPFR model is given as follows.

We introduce an indicator variable z_{mk}, where $m = 1, \ldots, M$ and $k = 1, \ldots, K$. If the m-th batch belongs to the k-th component, $z_{mk} = 1$; otherwise, $z_{mk} = 0$. All the indicator variables are assumed to share the same prior and the prior is given by

$$P(z_{mk} = 1) = \pi_k,$$

where $\sum_{k=1}^{K} \pi_k = 1$. After these curves are separated into the components by the indicator variables, the output of the k-th component $y(x)$ is subject to a GPFR model.

$$y(x) \sim \mathrm{GPFR}(x; \mathbf{b}_k, \boldsymbol{\theta}_k).$$

The total log likelihood of mix-GPFR model is

$$L(\Theta) = \sum\nolimits_{m=1}^{M} \sum\nolimits_{k=1}^{K} z_{mk}[\log \pi_k + \log p(\mathbf{y}_m | \mathbf{x}_m, \mathbf{b}_k, \boldsymbol{\theta}_k)],$$

where $\Theta = \{\pi_k, \mathbf{b}_k, \boldsymbol{\theta}_k\}_{k=1}^{K}$.

3 The DAEM Algorithm

The conventional EM algorithm is widely used in machine learning, but it has the local maximum problem. Thus, if the initialization of parameters is not good enough, the performance of the EM algorithm may be very poor. However, the initialization of mix-GPFR is very difficult. So we construct a DAEM algorithm to solve the local maximum problem of the conventional EM algorithm. As these M curves are independent, the Q-function of the conventional EM algorithm for mix-GPFR can be given as follows.

$$Q(\Theta) = \sum_{m=1}^{M} \sum_{k=1}^{k} \tilde{\alpha}_{mk}[\log \pi_k + \log p(\mathbf{y}_m|\mathbf{x}_m, \mathbf{b}_k, \boldsymbol{\theta}_k)],$$

where

$$\tilde{\alpha}_{mk} = \hat{\pi}_k p(\mathbf{y}_m|\mathbf{x}_m, \hat{\mathbf{b}}_k, \hat{\boldsymbol{\theta}}_k) / \sum_{j=1}^{K} \hat{\pi}_j p\left(\mathbf{y}_m|\mathbf{x}_m, \hat{\mathbf{b}}_j, \hat{\boldsymbol{\theta}}_j\right).$$

We introduce an annealing parameter β to the Q-function and then construct the U-function of the DAEM algorithm for mix-GPFR as follows.

$$U(\Theta, \beta) = \sum_{m=1}^{M} \sum_{k=1}^{K} \alpha_{mk}[\log \pi_k + \log p(\mathbf{y}_m|\mathbf{x}_m, \mathbf{b}_k, \boldsymbol{\theta}_k)],$$

where

$$\alpha_{mk} = \left[\hat{\pi}_k p\left(\mathbf{y}_m|\mathbf{x}_m, \hat{\mathbf{b}}_k, \hat{\boldsymbol{\theta}}_k\right)\right]^{\beta} / \sum_{j=1}^{K} \left[\hat{\pi}_j p\left(\mathbf{y}_m|\mathbf{x}_m, \hat{\mathbf{b}}_j, \hat{\boldsymbol{\theta}}_j\right)\right]^{\beta}.$$

Obviously, $U(\Theta, 1)$ is equal to $Q(\Theta)$ so the conventional EM algorithm could be regarded as a special case of the DAEM algorithm with $\beta = 1$. With the U-function, we can show the details of the DAEM algorithm in five steps.

Step 1. Initialize α_{mk} by a simple curve clustering method (such as the k-means algorithm) and set the initial value $\beta = \beta_{min}$, where $\beta_{min} < 1$
Step 2. M-step: calculate Θ by maximizing $U(\Theta, \beta)$
Step 3. E-step: update α_{mk}
Step 4. $\beta = \min(\beta \times \text{const}, 1)$
Step 5. When $\beta = 1$ and the increase of $U(\Theta, 1)$ is small enough, stop; otherwise, return to Step 2.

For the DAEM algorithm, the initialization is not so important and infact most of curve clustering methods can be used. The only distinction between the DAEM algorithm and the conventional EM algorithm is just β. The experiments demonstrate that $\beta_{min} = 0.2$ is small enough to avoid the local maximum problem. But we sometimes use a bigger β_{min} because some useful initialization can be utilized in certain practical applications. Another advantage of the DAEM algorithm is that the time consumer of the DAEM algorithm is about two times of the conventional EM

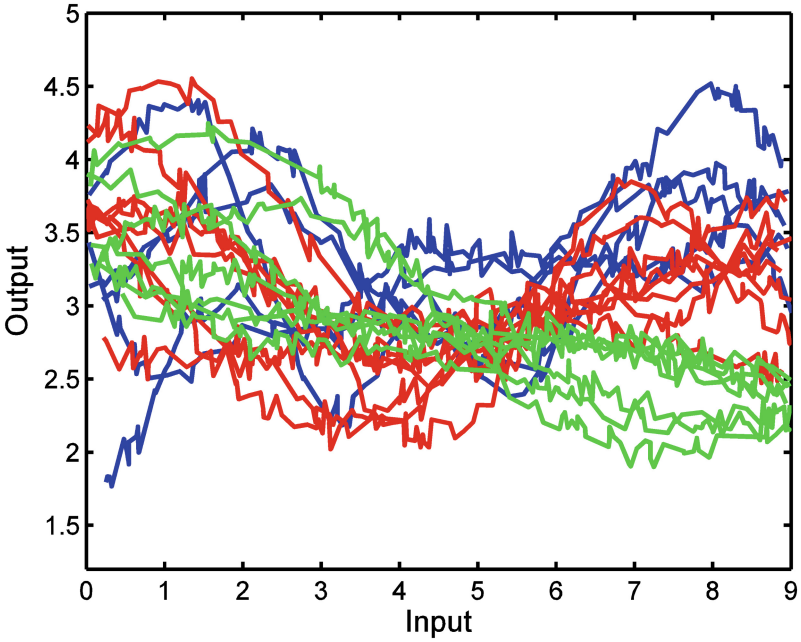

Fig. 1. The curves are the training samples of the simulated dataset for the mix-GPFR model and there are only 20 curves of 3 components in 3 colors. (Color figure online)

algorithm. The M-step of the DAEM algorithm is quite the same as the M-step of the conventional EM algorithm and the only difference is the new parameter β. In [9], there are two types of prediction, but the second type is not very useful. So we just consider the first type prediction for mix-GPFR and the details of the M-step and prediction method can be referred to [9]. In addition, the theory of the DAEM algorithm was described in [11, 12].

Table 1. The parameter estimation of the DAEM algorithm on the simulated dataset for the mix-GPFR: we show the true value (TV), estimated value (EV) and relative error (RE) of the parameters

		π_k	θ_{k1}	θ_{k2}	θ_{k3}
k = 1	TV	0.3333	0.6325	1.0000	0.0632
	EV	0.3256	0.6449	0.9982	0.0631
	RE	2.3 %	2.0 %	0.2 %	0.2 %
k = 2	TV	0.3333	0.4472	0.7071	0.0632
	EV	0.3274	0.4409	0.6999	0.0627
	RE	1.8 %	1.4 %	1.0 %	0.8 %
k = 3	TV	0.3333	0.3162	0.4472	0.0632
	EV	0.3470	0.3124	0.4523	0.0633
	RE	4.1 %	1.2 %	1.1 %	0.2 %

4 Experimental Results

In this section, we demonstrate the experimental results of the DAEM algorithm for the mix-GPFR model on both the simulated and electrical load datasets, being compared with the conventional EM algorithm and related approaches. Note that the mix-GP just means the mixture of the GPs with zero mean functions.

4.1 On the Simulated Dataset

We conduct various experiments on datasets generated by different mix-GPFR models and the DAEM algorithm always performs very well. Typically, we show the results on a simulated dataset generated by the mix-GPFR model with 3 components. The mean functions of GPFR models are $\mu_1(x) = 0.5\sin\left[0.125(x-4)^2\right] + 3$, $\mu_2(x) = -3(2\pi)^{-0.5}\exp\left[-0.125(x-4)^2\right] + 3.7$ and $\mu_3(x) = 0.5\arctan(0.5x-2) + 3$, respectively. The parameters π_k and $\boldsymbol{\theta}_k$ of components are shown in Table 1. In Fig. 1,

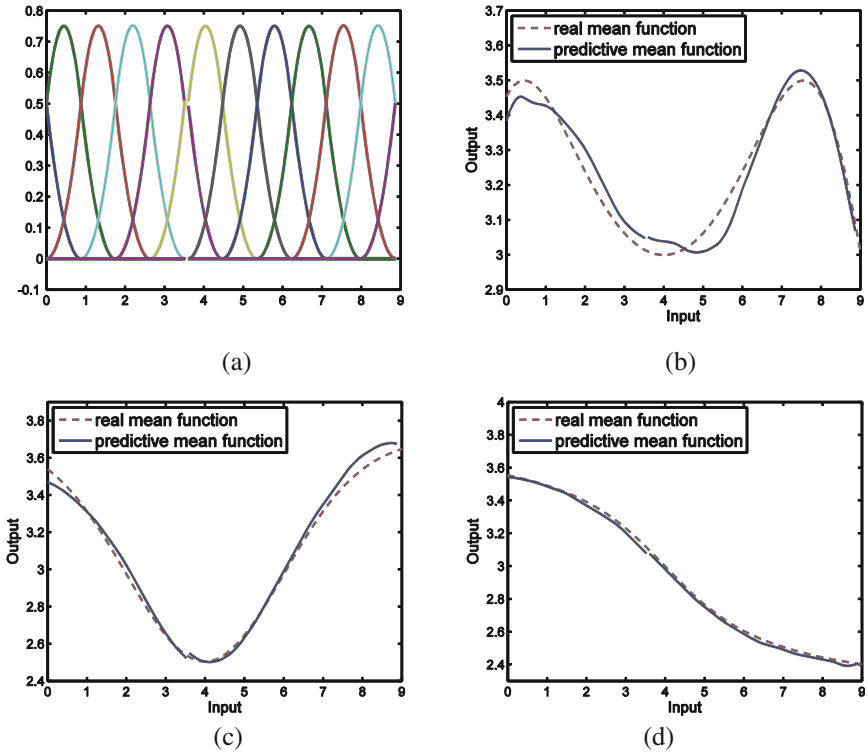

Fig. 2. (a) A set of b-spline basis functions; (b–d) The components' real and predictive mean functions of the simulated dataset for the mix-GPFR trained by the DAEM algorithm, respectively

20 sample curves of training dataset are already illustrated and the curves are obviously difficult to be clustered. Actually, we generate 300 training curves and each of them consists of 50 points. On the other hand, we generate 600 test curves and each of them have 40 known points and 110 test points.

Table 2. The mean RMSEs of the models and algorithms on the simulated and electrical load datasets for the mix-GPFR.

Model	Algorithm	Simulated		Electrical load	
		K	RMSE	K	RMSE
Mix-GPFR	DAEM	2	0.0735	4	0.6394
Mix-GPFR	Conventional EM	2	0.0741	3	0.6647
Mix-GP	Conventional EM	2	0.0793	4	0.9741
GPFR	MLE	–	0.0772	–	0.8608
GP	Gradient method	–	0.0838	–	1.0394

After trial and error, $\beta_{min} = 0.2$ is used at last and the const in Step 4 of the DAEM algorithm is 1.1576. The number of b-spline basis functions $D = 22$. We show the estimation results of parameters π_k and $\boldsymbol{\theta}_k$ in Table 1 and the predicted as well as real mean functions are shown in Fig. 2(b–d). The parameter estimation of the DAEM algorithm is generally good except π_k on this dataset. It does not matter, because it is caused by not only the algorithm but also the stochasticity. What is more, the prediction of the mean function of the 1st component is not very good because $\theta_{2,1}$, which controls the amplitude, is the biggest.

We show the values of α_{mk} of the DAEM algorithm in Fig. 4(a–b) and compare it with $\tilde{\alpha}_{mk}$ of the conventional EM algorithm in Fig. 4(c–d). We find out that α_{mk} of the DAEM algorithm are more similar in the early iterations and it leads the effect of the initialization more little. The U-function with $\beta_{min} = 0.2$ during iterations of the

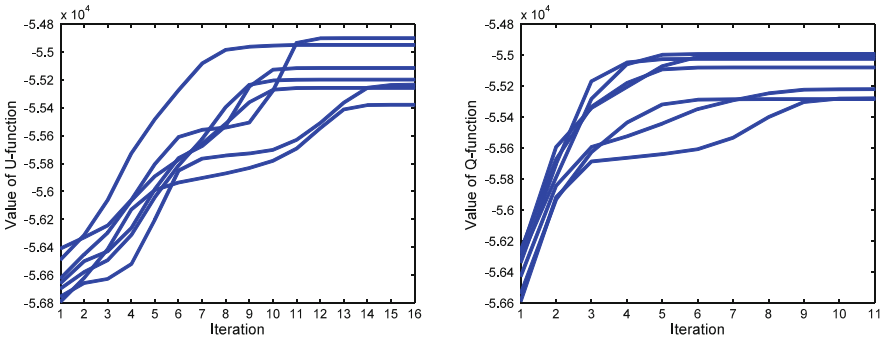

Fig. 3. (a) The value of U-function of the DAEM algorithm with $\beta_{min} = 0.2$ during the iterations; (b) The value of Q-function of the conventional EM algorithm during the iterations

DAEM algorithm and the Q-function during iterations of the conventional EM algorithm are illustrated in Fig. 3(a–b), respectively. The two EM algorithms are effective but the DAEM algorithm is better because the value of U-function is a bit bigger after the convergence.

The classification accuracy rate (CAR) of the DAEM algorithm on the curves of the test dataset is 98.17 %, which is bigger than the CAR of the conventional EM algorithm, 97.17 %. The root mean square errors (RMSEs) of the DAEM and conventional EM algorithms for the mix-GPFR are shown in Table 2. In addition, we show the RMSEs of other three models, which are GPFR, mix-GP and GP models. The RMSEs of the mix-GPFR are smaller and the RMSE of the mix-GPFR trained by the DAEM algorithm is the smallest. The GPFR and mix-GP models are not the best but better than the GP.

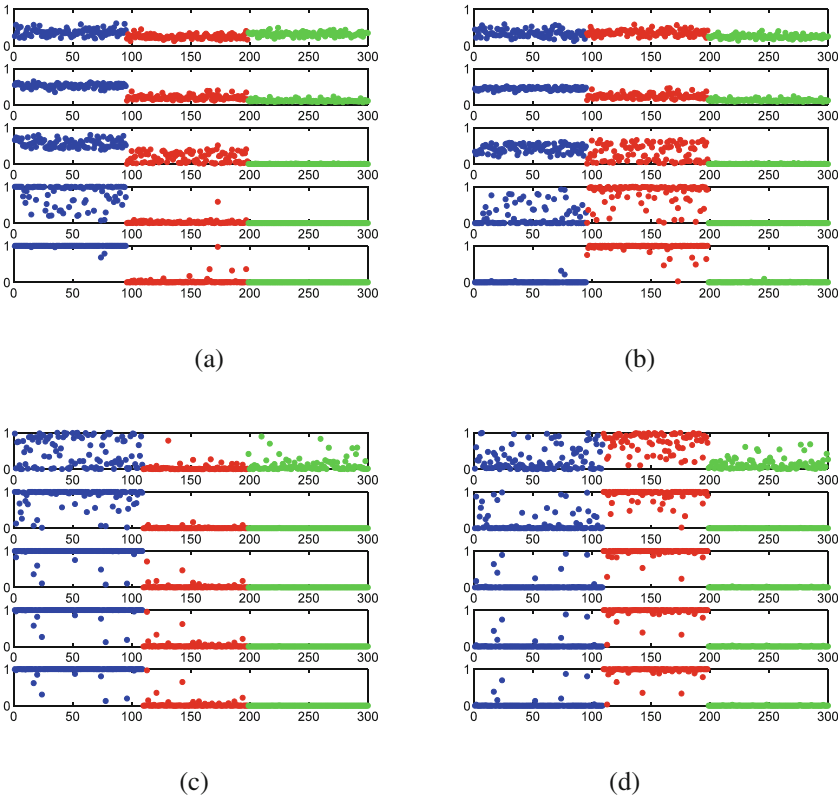

Fig. 4. α_{mk} or $\tilde{\alpha}_{mk}$ of batches belonging to the 1st, 2nd and 3rd components are illustrated by blue, red and green points, respectively. (a–d) α_{m1} and α_{m2} of 300 training batches with the iterations, which are the 1st, 4th, 7th, 10th and 13th iterations, of the DAEM algorithm with $\beta_{min} = 0.1$ for mix-GPFR, respectively; (c–d) $\tilde{\alpha}_{m1}$ and $\tilde{\alpha}_{m1}$ of training batches with iterations, which are the 1st, 3th, 5th, 7th and 9th iterations, of the conventional EM algorithm for mix-GPFR, respectively (Color figure online)

4.2 On the Electrical Load Dataset

Electrical load prediction plays a vital role in optimal unit commitment, start up and shut down of thermal plants, control of reserve and exchanging electric power in interconnected systems [14]. The electrical load dataset is from the Northwest China Grid Company. There are 100 sample curves in electrical load dataset and 96 points of each curve are observations of one day. We separate this dataset into 2 groups and each group has 50 sample curves. One group of sample curves is the training dataset and the other is the test dataset. We also separate the points of each curve in test dataset into two groups with 48 points in each. We use one group for training and the other for testing.

We make model selection by the cross validation method on the training dataset with various numbers of components K. In Fig. 5, the 50 training sample curves are separated into 4 components by the DAEM algorithm and the curves belonging to the same component are illustrated in the same color. 48 curves belong to two components, which are blue and green, respectively, and there are only 2 sample curves in the other 2 components. Obviously, the clustering of the blue and green curves is good. From Table 2, the mix-GPFR is the best model and the DAEM algorithm is much better than the conventional EM algorithm on prediction. The performance of the GPFR model is also good, so the mean function is important for the electrical load dataset.

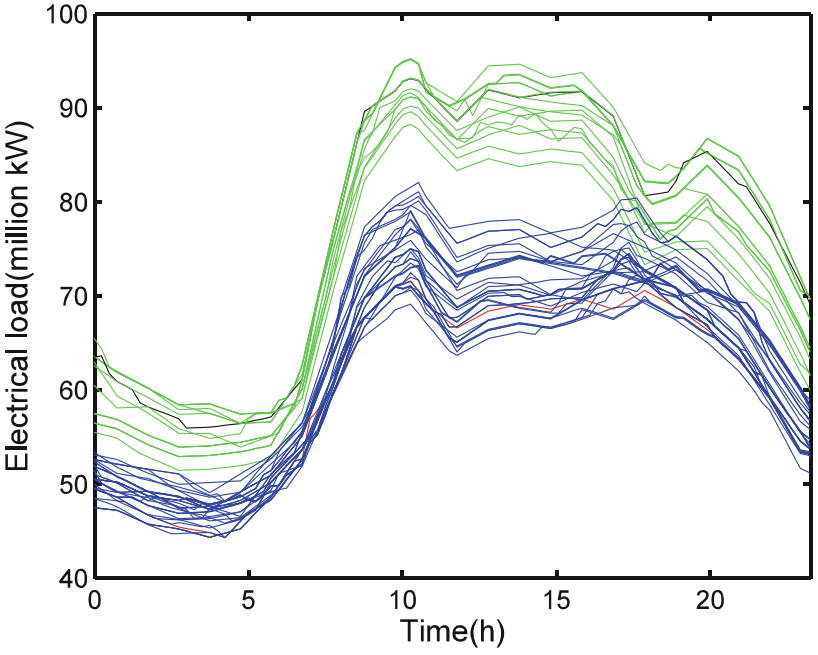

Fig. 5. The training curves of electrical dataset are clustered into 4 components (Color figure online)

5 Conclusions

We have established the DAEM algorithm for the mix-GPFR model to solve the problem of local maximum associated with the conventional EM algorithm. As the key difference between the DAEM algorithm and the conventional EM algorithm, a flexible variable β is introduced in the DAEM algorithm to make the parameter learning process in a deterministic way. On simulated dataset, $\beta_{min} = 0.2$ may be small enough and the DAEM algorithm performs well on parameters estimation. In addition, the DAEM algorithm is better than the conventional EM algorithm on curve clustering and prediction. Moreover, the experimental results on a real-world dataset, i.e., the electrical load dataset, demonstrate that the DAEM algorithm is also good on curve clustering and better than the conventional EM algorithm on prediction.

Acknowledgement. This work is supported by the National Science Foundation of China under Grant 61171138.

References

1. Rasmussen, C.E.: Evaluation of Gaussian Processes and Other Methods for Non-linear Regression. Ph.D. dissertation, Department of Computer Science, University of Toronto (1996)
2. Rasmussen, C.E., Williams, C.K.I.: Regression Gaussian Process for Machine Learning, ch. 2. MIT Press, Cambridge (2006)
3. Shi, J.Q., et al.: Gaussian process functional regression modeling for batch data. Biometrics **63**, 714–723 (2007)
4. Boor, D.E.: On calculating with B-splines. J. Approximation Theory **6**, 50–62 (1972)
5. Shi, J.Q., Murray-Smith, R., Titterington, D.M.: Bayesian regression and classification using mixtures of Gaussian process. Int. J. Adapt. Control Signal Process. **17**, 149–161 (2003)
6. Shi, J.Q., Murray-Smith, R., Titterington, D.M.: Hierarchical Gaussian process mixtures for regression. Stat. Comput. **15**, 31–41 (2005)
7. Kamnik, R., et al.: Nonlinear modeling of FES-supported standing-up in paraplegia for selection of feedback sensors. IEEE Trans. Neural Syst. Rehabil. Eng. **13**(1), 40–52 (2005)
8. Qiang, Zhe, Ma, Jinwen: Automatic model selection of the mixtures of Gaussian processes for regression. In: Hu, X., Xia, Y., Zhang, Y., Zhao, D. (eds.) ISNN 2015. LNCS, vol. 9377, pp. 335–344. Springer, Heidelberg (2015). doi:10.1007/978-3-319-25393-0_37
9. Shi, J.Q., Wang, B.: Curve prediction and clustering with mixtures of Gaussian process functional regression models. Stat. Comput. **18**, 267–283 (2008)
10. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Statist. Soc. (B) **39**, 1–38 (1977)
11. Ueda, N., Nakano, R.: Mixture density estimation via EM algorithm with deterministic annealing. In: Proceedings of the IEEE Neural Networks for Signal Processing, pp. 66–77 (1994)
12. Ueda, N., Nakano, R.: Deterministic annealing EM algorithm. Neural Netw. **11**, 271–282 (1998)
13. Ma, J., Xu, L., Jordan, M.I.: Asymptotic convergence rate of the EM algorithm for Gaussian mixtures. Neural Comput. **12**(12), 2881–2907 (2000)
14. Mohandes, M.: Support vector machines for short-term electrical load forecasting. Int. J. Energy Res. **26**, 335–345 (2002)