Mixture Multi-Step-Ahead Prediction

Ivan Nagy[2,1], Evgenia Suzdaleva[1], and Tereza Mlynářová[1]

[1] Department of Signal Processing, The Institute of Information Theory and Automation of the Czech Academy of Sciences, Pod vodárenskou věží 4, 18208 Prague, Czech Republic
(E-mail: suzdalev@utia.cas.cz)
[2] Faculty of Transportation Sciences, Czech Technical University, Na Florenci 25, 11000 Prague, Czech Republic
(E-mail: nagy@utia.cas.cz)

Abstract. The presented paper deals with a task of the multi-step prediction with mixture models under Bayesian methodology. The main contribution of the paper is a recursive prediction algorithm for mixtures with the dynamic switching model. The proposed algorithm is based on construction of the weighting vector predicting the active component and its combination with data predictions from components. With the help of illustrative examples the paper compares the results with those obtained for the mixture prediction with the static switching model.
Keywords: mixture model; mixture prediction; recursive mixture estimation; dynamic pointer; active component.

1 Introduction

The paper deals with a problem of the recursive prediction with mixture models. A mixture model consists of components describing different modes of the observed system behavior and a model of their switching. Necessity of solving such a task can be explained with the help of the following simple example. Imagine that during driving a vehicle the driver assistance system (DAS) provides a warning for the driver, which says: "Your driving is dangerous". It means that, based on measurements from sensors, DAS classifies the driving style (i.e., the active component of the mixture) as *already* dangerous, and at any instant the vehicle might crash. In reality such a warning is useless for the driver. Much more information would be brought by predicting the driving style evolution and by a warning of the type: "Your driving begins to be dangerous", which means that if the driver continues driving in the detected style, the vehicle will crash in a moment. This was a motivation of the presented research.

Research concerned with mixtures are intensively developed. Most of them focus on the parameter estimation of mixture components and of the switching model. Papers found in this area are mainly based on (i) the EM algorithm [1], see, e.g., [2,3]; (ii) the Variational Bayes (VB) approach [4,5]; (iii)

16^{th} *ASMDA Conference Proceedings, 30 June – 4 July 2015, Piraeus, Greece*

Markov Chain Monte Carlo (MCMC) techniques, e.g., [6–9]. So far a systematic methodology of the mixture prediction is still missing.

The presented approach is based on the recursive Bayesian estimation theory [10,11] and primarily inspired by [12,13], which, unlike the above mentioned sources, represent on-line data-based estimation algorithms avoiding numerical iterative computations as far as possible. This gives a great potential for real-time (on-line) applications in data analysis due to eliminating the need to take care of convergence of algorithms. Such applications are desired, for instance, in the field of fault detection, on-line diagnostic systems, intelligent transportation systems, medicine, etc. Mixture models naturally serve for the data analysis tasks of clustering and classification, which are strictly divided in data-mining methods, see, e.g., [14,15]. Within the bounds of the considered recursive mixture estimation theory, these tasks are closely related, which enables to utilize either the clustering or the classification models depending on the application.

This paper proposes a recursive algorithm of the multi-step prediction for mixtures with the dynamic switching model. Specific tasks the paper deals with are (i) construction of the weighting vector predicting the active component and its combination with data predictions from components; and (ii) the parameter estimation of the mixture. The first of them is the main contribution of the paper, while the second one is solved for the static switching model in [12] and for the dynamic one in [13]. The presented paper uses these experiences. Results obtained for the mixture prediction with dynamic switching are compared with those for the static one.

The layout of the paper is as follows. Section 2 provides notations and introduces models used in the paper. It also recalls the existing algorithms of estimation of individual models entering the considered mixture model and of the mixture estimation. Section 3 is devoted to the mixture prediction both with the static and the dynamic models of switching. Section 4 compares results of experiments with both the models. Conclusions and plans of the future work can be found in Section 5.

2 Preliminaries

2.1 General conventions

The following notations are used throughout the text:

- Both the probability density function and the probability function are replaced by the abbreviation *pdf*.
- For continuous random variables A, B and their realizations a, b the conditional pdf is denoted by

$$f_{A|B}(a|b) \equiv f(a|b).$$

- For categorical random variables C, D and their realizations c, d the conditional pdf is denoted by

$$f_{C|D}(c|d) \equiv f(C = c|D = d).$$

- Mixed distributions of random variables A, B, C, D with realizations a, b, c, d are denoted by

$$f_{A,C|B,D}(a, c|b, d) \equiv f(a, C = c|b, D = d).$$

- t used as a subscript of a random variable denotes discrete time instants, where the variable is measured, $t = \{1, 2, \ldots\}$.
- x_t denotes the random variable x measured at the discrete time instant t.
- x^* is a set of all possible values of the random variable x.
- $x(t)$ is a collection of all available measurements of the random variable x up to the time instant t, i.e., $x(t) = \{x_0, x_1, \ldots, x_t\}$, including prior data x_0.
- $\propto$ denotes equality up to the normalization constant.
- $\equiv$ means equality by definition.
- $y_t \in \mathbb{R}$ is the output measured on the observed system.
- In general, all variables are column vectors.
- k_y is the dimension of the vector y_t.

2.2 Components

In this paper, the observed multi-modal system is described by the mixture model, which consists of m_c static components in the form of the following pdf

$$f\left(y_t|\Theta, c_t = i\right), \ \forall i \in \{1, 2, \ldots, m_c\} \equiv c^*, \tag{1}$$

where Θ are parameters of components, and $c_t \in c^*$ is the random unmeasured categorical variable called the pointer, which indicates the component, active at time t.

Here, each i-th component (1) is specified as the normal regression model

$$y_t = \theta_i + e_{i;t}, \ \forall i \in c^*, \tag{2}$$

where $e_{i;t}$ is the normally distributed noise of the i-th component with the zero mean vector and the covariance matrix r_i, and $\{\theta_i, r_i\} \equiv \Theta_i$ corresponding to the i-th component, and $\Theta \equiv \{\Theta_i\}_{i=1}^{m_c}$ is the collection of parameters of all components.

2.3 Static switching model

Switching the active components can be described by the static model

$$f\left(c_t = i|\alpha\right), \ \forall i \in c^*, \tag{3}$$

represented by the transition table

c_t	$c_t = 1$	$c_t = 2$	$\cdots$	$c_t = m_c$
$f\left(c_t = i|\alpha\right)$	α_1	α_2	$\cdots$	α_{m_c}

where α is the m_c-dimensional vector parameter, which contains stationary probabilities α_i, $\forall i \in c^*$, of activity of individual components (i.e., of individual values of the pointer c_t) and it holds

$$\alpha_i \geq 0, \sum_{i=1}^{m_c} \alpha_i = 1, \ \forall i \in c^*. \tag{4}$$

2.4 Dynamic switching model

Switching the components can be also described by the dynamic pointer model

$$f\left(c_t = i | c_{t-1} = j, \alpha\right),\ i, j \in c^*, \tag{5}$$

which unlike (3) is provided by the transition table

	$c_t = 1$	$c_t = 2$	$\cdots$	$c_t = m_c$
$c_{t-1} = 1$	$\alpha_{1\|1}$	$\alpha_{2\|1}$	$\cdots$	$\alpha_{m_c\|1}$
$c_{t-1} = 2$	$\alpha_{1\|2}$		$\cdots$	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
$c_{t-1} = m_c$	$\alpha_{1\|m_c}$		$\cdots$	$\alpha_{m_c\|m_c}$

where the parameter α is the $(m_c \times m_c)$-dimensional matrix, and its entries $\alpha_{i|j}$ are the probabilities of the pointer $c_t = i$ under condition that the previous pointer $c_{t-1} = j$, for $i, j \in c^*$ and it holds the similar assumption as (4), which means that the sum of entries in a row is equal to 1.

2.5 Individual model estimation

This section provides a theoretical background necessary for understanding the rest of the paper. Bayesian estimation of individual models (2), (3) and (5) is briefly recalled here. The estimation is based on Bayes rule [11]

$$f(\Theta|y(t)) \propto f(y_t|\Theta) f(\Theta|y(t-1)), \tag{6}$$

where $f(\Theta|y(t-1))$ is the prior pdf to be reproduced, see the rest of notations in Section 2.1.

According to [11,12], the individual normal model (2) is estimated via (6) using the conjugate prior Gauss-inverse-Wishart pdf with the statistics V_{t-1} and k_{t-1} of the appropriate dimensions. Substitution of this conjugate prior pdf with the initially chosen statistics and the model (2) into (6) allows to recompute the statistics as follows [11,12]:

$$V_t = V_{t-1} + \begin{bmatrix} y_t \\ 1 \end{bmatrix} [y_t,\, 1]\,, \tag{7}$$

$$\kappa_t = \kappa_{t-1} + 1, \tag{8}$$

The point estimates of parameters are then recomputed for the time instant t as follows [11]:

$$\hat{\theta}_t = V_1^{-1} V_y, \quad \hat{r}_t = \frac{V_{yy} - V_y^{'} V_1^{-1} V_y}{\kappa_t} \tag{9}$$

with the help of partition

$$V_t = \begin{bmatrix} V_{yy} & V_y^{'} \\ V_y & V_1 \end{bmatrix}, \tag{10}$$

where V_{yy} is the square matrix of the dimension k_y of the vector y_t, V_y' is the k_y-dimensional column vector and V_1 is scalar.

As regards the individual categorical models (3) and (5), their parameter estimation is the same with the difference in the dimension of their statistics. According to [10], in the case of the measured values of c_t and c_{t-1} (the last is necessary only for (5)), the models (3) and (5) are estimated via (6) using the conjugate prior Dirichlet pdf with the recomputable statistics ϑ_{t-1}, which is the m_c-dimensional vector for (3) and the square m_c-dimensional matrix for (5). Their entries for $c_t = i$ are updated for (3) in the following way:

$$\vartheta_{i;t} = \vartheta_{i;t-1} + 1 \tag{11}$$

and for $c_t = i$ and $c_{t-1} = j$ for (5) as follows:

$$\vartheta_{i|j;t} = \vartheta_{i|j;t-1} + 1. \tag{12}$$

The point estimate of the parameter α is then obtained by the corresponding normalization, i.e., for (3)

$$\hat{\alpha}_{i;t} = \frac{\vartheta_{i;t}}{\sum_{k=1}^{m_c} \vartheta_{k;t}}, \ \ \forall i \in c^*. \tag{13}$$

and for (5)

$$\hat{\alpha}_{i|j;t} = \frac{\vartheta_{i|j;t}}{\sum_{k=1}^{m_c} \vartheta_{k|j;t}}, \ \ \forall i, j \in c^*, \tag{14}$$

see details in [10].

Using the recalled estimation of individual models (2), (3) and (5), the recursive mixture estimation with switching described by (3) is proposed in [12] and with (5) – in [13]. The basis for both the algorithms is the construction of the weighting vector, whose entries are probabilities of activity of components. These probabilities enter updates of corresponding statistics. A solution to the problem formulated below utilizes these mixture estimation algorithms.

2.6 Problem formulation

The task of the recursive mixture prediction with components (2) and the dynamic switching model (5) is specified as follows: *based on the available data collection $y(t-1)$, predict the output of the system for n steps ahead.*

3 Recursive mixture multi-step-ahead prediction

3.1 Mixture prediction with the static switching model

To derive the prediction solution with the model (5), it is useful to demonstrate on a relatively trivial case with the static switching model (3). Derivations are based on construction of the joint pdf of all variables to be predicted and estimated and application of the Bayes rule and the chain rule [11]. For simplicity, the prediction is shown here for 2 steps ahead, which means from time $t-1$ to $t+1$. Its generalization does not cause any computational complexity. The

mixture predictive pdf with the switching model (3) takes the following form, for $i \in c^*$,

$$f\left(y_{t+1}|y\left(t-1\right)\right) = \sum_{i\in c^*} \int\int \underbrace{f\left(y_{t+1}, \Theta, c_{t+1}=i, \alpha|y\left(t-1\right)\right)}_{joint\ pdf} d\Theta d\alpha =$$

$$= \sum_{i\in c^*} \int\int f\left(y_{t+1}|\Theta, c_{t+1}=i\right) f\left(c_{t+1}=i|\alpha\right) f\left(\Theta|y\left(t-1\right)\right)$$

$$\times f\left(\alpha|y\left(t-1\right)\right) d\Theta d\alpha =$$

$$= \sum_{i\in c^*} \int f\left(y_{t+1}|\Theta, c_{t+1}=i\right) f\left(\Theta|y\left(t-1\right)\right) d\Theta$$

$$\times \int f\left(c_{t+1}=i|\alpha\right) f\left(\alpha|y\left(t-1\right)\right) d\alpha, \tag{15}$$

where the integral over the parameter α provides the point estimate $\hat{\alpha}_{i;t-1}$ obtained according to (13), see [10]. The integral over the parameter Θ is evaluated by substituting its point estimates $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ (obtained according to (9) for the i-th component according to [12]) into each i-th model (2). Marginalization over the values of unavailable c_t is performed by summation. It can be seen that the prediction in this case reduces to the following steps: take the point estimates from the previous time instant, substitute them into all components and make the weighted average of predictions obtained from all components.

It is seen in (15) that in the static case the mixture prediction for n steps ahead will have the same form with the substituted last available point estimates. The mixture predictive pdf takes the form:

$$f\left(y_{t+(n-1)}|y\left(t-1\right)\right) = \sum_{i\in c^*} \hat{\alpha}_{i;t-1} \underbrace{f\left(y_{t+(n-1)}|y\left(t-1\right), \hat{\Theta}_{i;t-1}\right)}_{\tilde{y}_{i;t+(n-1)}}, \tag{16}$$

where $\tilde{y}_{i;t+(n-1)}$ is the output prediction obtained from the i-th component by substituting $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ into (2) for $i \in c^*$, and where $\hat{\alpha}_{i;t-1}$ plays a role of the weight of the the i-th component activity. The mixture predictive pdf could be further used as the distribution. However, in practice it is often useful to take the point estimate of the predicted value, which can be expressed as the expectation. The mixture prediction algorithm can be summarized as the following straightforward algorithm.

Algorithm 1

Initial part (for t=1)

- Specify m_c components (2) and the switching model (3).
- $\forall i \in c^*$ set initial statistics of all components $V_{i;0}$, $\kappa_{i;0}$ and ϑ_0 of appropriate dimensions. The statistics can be chosen as small-valued matrices (respectively vectors) for experiments with simulated data and are constructed from the prior data for the real data application.

- Using these initial statistics, compute the initial point estimates of all parameters and for all components according to (9) and (13).

On-line part (for t=2,...)
Prediction

1. Substitute the previous point estimates $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ into each i-th component (2) and obtain the n-step-ahead data prediction $\tilde{y}_{i;t+(n-1)}$ from the i-th component.
2. Compute the weighted average of the data predictions from all components using $\hat{\alpha}_{i;t-1}$ and $\tilde{y}_{i;t+(n-1)}$ and obtain the mixture prediction $\hat{y}_{t+(n-1)}$ (as its expectation). The prediction of the active component is defined according to the maximal entry of the vector $\hat{\alpha}_{t-1}$.

Estimation

3. Measure the new data y_t.
4. $\forall i \in c^*$, substitute y_t and the point estimates $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ into each i-th component. Construct the m_c-dimensional vector of proximities from results from all components.
5. Multiply entry-wise the vector of proximities from the previous step and the point estimate vector $\hat{\alpha}_{t-1}$ and normalize the result. The result is the weighting vector w_t with entries $w_{i;t}$, which express the updated probabilities of activity of the i-th component.
6. $\forall i \in c^*$ update all statistics, using $w_{i;t}$ according to [12] as follows:

$$V_{i;t} = V_{i;t-1} + w_{i;t} \begin{bmatrix} y_t \\ 1 \end{bmatrix} [y_t,\, 1]\,, \tag{17}$$

$$\kappa_{i;t} = \kappa_{i;t-1} + w_{i;t}, \tag{18}$$

$$\vartheta_{i;t} = \vartheta_{i;t-1} + w_{i;t}. \tag{19}$$

7. $\forall i \in c^*$ recompute the point estimates of all parameters according to (9) and (13) and use them for step 1 of the on-line part of the algorithm, i.e., for the next prediction.

3.2 Mixture prediction with the dynamic switching model

The multi-step-ahead prediction of the mixture with the dynamic switching model is a more complicated task because of the c_{t-1} in (5). Derivations are similarly based on construction of the joint pdf of all variables to be predicted and estimated with application of the Bayes and the chain rule [11] as in Section 3.1. Here, the mixture predictive pdf takes for $i \in c^*$ the following form (for simplicity, shown again for 2 prediction steps):

$$f\left(y_{t+1}|y\left(t-1\right)\right)$$

$$= \sum_{k,i,j \in c^*} \int\int \underbrace{f\left(y_{t+1}, c_{t+1}=k, c_t=i, c_{t-1}=j, \Theta, \alpha | y\left(t-1\right)\right)}_{joint\ pdf} d\Theta d\alpha =$$

$$= \sum_{k,i,j \in c^*} \int\int f\left(y_{t+1}|\Theta, c_{t+1}=k\right) f\left(c_{t+1}=k|c_t=i,\alpha\right) f\left(c_t=i|c_{t-1}=j,\alpha\right) \times$$

$$\times f\left(c_{t-1}=j|y\left(t-1\right)\right) f\left(\Theta|y\left(t-1\right)\right) f\left(\alpha|y\left(t-1\right)\right) d\Theta d\alpha =$$

$$= \sum_{k,i,j \in c^*} \int f\left(y_{t+1}|\Theta, c_{t+1}=k\right) f\left(\Theta|y\left(t-1\right)\right) d\Theta \times$$

$$\times \int f\left(c_{t+1}=k|c_t=i,\alpha\right) f\left(c_t=i|c_{t-1}=j,\alpha\right) f\left(\alpha|y\left(t-1\right)\right) d\alpha$$

$$\times f\left(c_{t-1}=j|y\left(t-1\right)\right) =$$

$$= \sum_{k,i,j \in c^*} \tilde{y}_{k;t+1} \hat{\alpha}_{k|i;t-1} \hat{\alpha}_{i|j;t-1} w_{j;t-1} = diag\left(\tilde{y}_{t+1}\right) \underbrace{\hat{\alpha}_{t-1}^2 diag\left(w_{t-1}\right)}_{\hat{w}_{t+1}}, \qquad (20)$$

where *diag* denotes a diagonal matrix created from a vector; $\tilde{y}_{k;t+1}$ denotes the prediction obtained from the k-th component (2) with the substituted point estimates $\hat{\theta}_{k;t-1}$ and $\hat{r}_{k;t-1}$, $k \in c^*$; and $\tilde{y}_{t+1}$ is a vector $[\tilde{y}_{1;t+1}, \ldots, \tilde{y}_{m_c;t+1}]$. Again, the previous point estimates are used, but the weighting vector is predicted for 2 steps ahead as $\hat{w}_{t+1} = \hat{\alpha}_{t-1}^2 diag\left(w_{t-1}\right)$. The weighted average of individual predictions is taken, however, now the weights are entries of the predicted weight $\hat{w}_{t+1}$.

The algorithm for n prediction steps can be summarized as follows. The initial part of the algorithm is the same as for the mixture estimation task, see [13]. Similarly as in Section 3.1 it is based on the estimation solution from [12]. Here, it can be seen that within the considered theory, the used models can be applied either for the clustering (estimation) or the classification (prediction) task.

Algorithm 2
Initial part (for t=1)

- Specify m_c components (2) and the switching model (5).
- $\forall i \in c^*$ set the initial statistics of all components $V_{i;0}$, $\kappa_{i;0}$ and ϑ_0.
- Using these initial statistics, compute the initial point estimates of all parameters and for all components according to (9) and (14).
- Set the initial m_c-dimensional vector w_0.

On-line part (for t=2,...)
Prediction

1. Substitute the previous point estimates $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ into each i-th component (2) and obtain the data predictions $\tilde{y}_{i;t+(n-1)}$ from the i-th component.
2. Multiply entry-wise the prior vector w_{t-1} and the previous point estimate matrix $\hat{\alpha}_{t-1}$ and obtain the prediction of the weighting vector, i.e.,

$$\hat{w}_{t+(n-1)} = \hat{\alpha}_{t-1}^n diag\left(w_{t-1}\right). \qquad (21)$$

3. Compute the weighted average of the data predictions from all components using $\hat{w}_{i;t+(n-1)}$ and $\tilde{y}_{i;t+(n-1)}$ and obtain the mixture data prediction $\hat{y}_{t+(n-1)}$.

Estimation

4. Measure the new data y_t.
5. For all components, substitute y_t and the point estimates $\hat{\theta}_{i;t-1}$ and $\hat{r}_{i;t-1}$ into each i-th component. Construct the m_c-dimensional vector of proximities from results from all components.
6. Multiply entry-wise the resulted vector from the previous step, the prior weighting vector w_{t-1} and the point estimate matrix $\hat{\alpha}_{t-1}$.
7. The result of this entry-wise multiplication is the matrix with entries $W_{i,j;t}$ with $i, j \in c^*$. Normalize this matrix.
8. Perform the summation of the normalized matrix over rows and obtain the updated vector w_t with entries $w_{i;t}$.
9. $\forall i \in c^*$ update the statistics $V_{i;t}$ and $\kappa_{i;t}$ using $w_{i;t}$ according to (17) and (18) and the statistics $\vartheta_{i|j;t}$ using $W_{i,j;t}$ as follows:

$$\vartheta_{i|j;t} = \vartheta_{i|j;t-1} + W_{i,j;t},\ i, j \in c^*, \tag{22}$$

see details in [13], where the estimation solution is introduced with the approximation based on the Kerridge inaccuracy [16]. However, here for simplicity, the weighting vector is used in the corresponding updates similarly to [12], which computationally coincides with updates in [13].

10. $\forall i \in c^*$ recompute the point estimates of all parameters according to (9) and (14) and use them for step 1 of the on-line part of the algorithm, i.e., for the next prediction.

4 Results

The quality of prediction with both the above algorithms was compared on a series of experiments in Scilab (see www.scilab.org). Here the most illustrative of them are demonstrated.

Three components (i.e, $m_c = 3$) with the two-dimensional output vector are simulated using the regression coefficients $\theta_1 = [1\ \ 1]'$, $\theta_2 = [5\ \ 4]'$, $\theta_3 = [6\ \ 0]'$ and the noise covariance matrices $r_1 = r_2 = r_3$, which are chosen the same for all components as the two-dimensional unit matrices. The models of switching α used for the simulation should not be too uncertain. 1000 data samples are generated. The initial statistics are chosen as follows: $V_{i;0}$ is the three-dimensional zero matrix for each $i \in \{1, 2, 3\}$; $\kappa_{i;0}$ is zero for each component; and ϑ_0 is the zero three-dimensional vector for the static case and the zero square (3×3) matrix for the dynamic switching. The initial point estimates are computed using these statistics according to Algorithms 1 and 2. The initial weighting vector chosen for Algorithm 2 is $w_0 = [0.33\ \ 0.33\ \ 0.34]$. The on-line parts of both the algorithms are performed as the time cycles for 1000 time instants.

To show initial advantages of the algorithms, Figure 1 shows the 0-step-ahead prediction with both of them, which is a simple estimation of the output

y_t before its measuring obtained by substituting the point estimates into the components. In this case, the prediction detects three clusters the same for both algorithms.

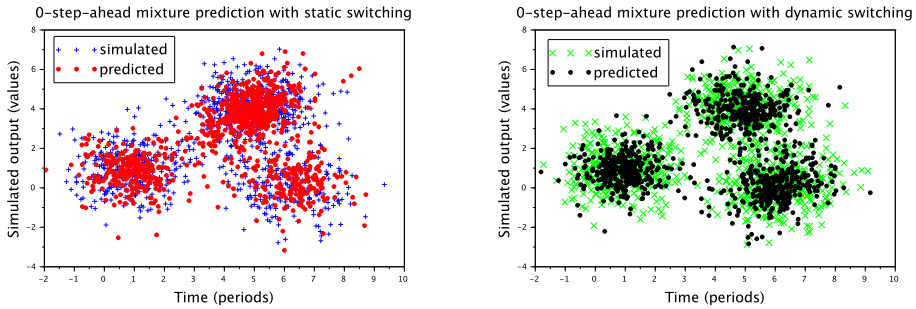

Fig. 1. The 0-step-ahead mixture prediction with static (left) and dynamic (right) switching. Notice that the results provide three predicted clusters similarly for both the algorithms.

Starting with the 1-step-ahead prediction, the difference becomes noticeable in favor of the dynamic case. Figure 2 provides the 1- and the 3-step-ahead predictions with the static switching. It is seen that the predicted values are compressed around the center of all components. With an increasing number of prediction steps the results with the static switching remain the same, thus they are not shown here to save space.

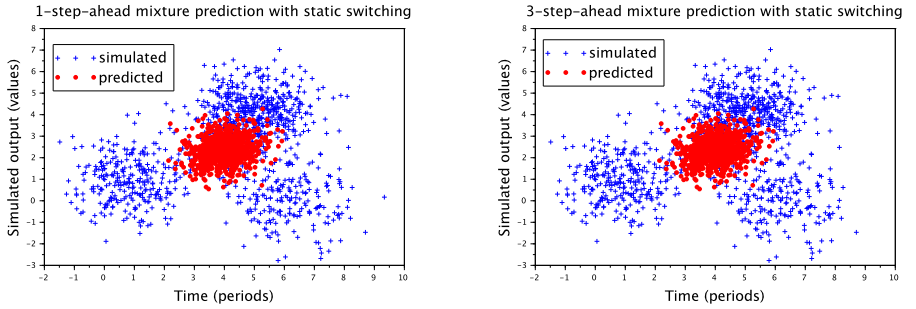

Fig. 2. The 1- and 3-step-ahead mixture predictions with static switching. Notice that the predicted values are compressed around the center of all components.

Figure 3 demonstrates results obtained with the dynamic switching for 1, 3, 5 and 15 steps of prediction respectively. Here, unlike the static prediction, the dynamic one still allows to detect the simulated clusters for 1, 3 and 5 steps. It worsens around the 15th step, where predicted values start to be located around the center of the figure. After that it does not change with an increasing number of steps.

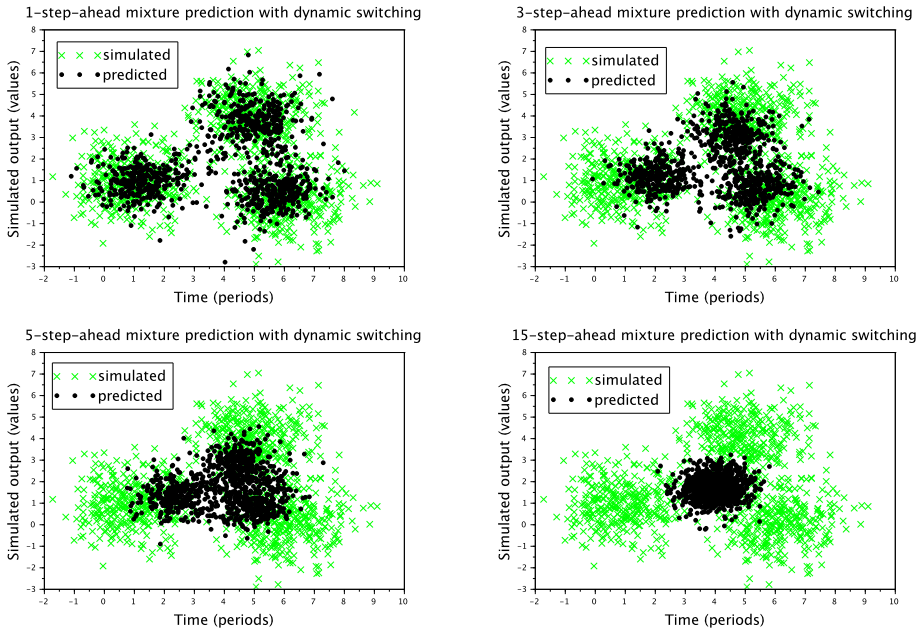

Fig. 3. The 1, 3, 5, 15-step-ahead mixture predictions with dynamic switching. Notice the predicted values detect three clusters for 1, 3, 5 steps. They start to be compressed around their center for the 15-step-ahead prediction.

To summarize the experimental part of the work, it should be said the multi-step-ahead mixture prediction with the static switching remains the same, while with the dynamic switching it first provides noticeably better results, and then it naturally worsens. Thus, the static prediction is similar to the dynamic one for a significant number of steps ahead.

5 Conclusions

The paper focuses on recursive Bayesian algorithms of the multi-step-ahead prediction for mixtures with the dynamic model of switching the components. The static case is also considered for comparison with a dynamic one. The proposed algorithm enables to predict the weighting vector containing probabilities of activity of the mixture components, which is the task highly desired in many areas of data analysis (fault detection, on-line diagnostic systems, intelligent transportation systems, etc.). The provided examples demonstrate advantages of the dynamic mixture prediction in comparison with the static one.

The paper presents the solution only for normal components. However, the proposed algorithm is not restricted in this sense and can be applied for mixtures of other components with recomputable statistics (e.g., categorical, exponential, etc.). This will be an issue of the future work.

Acknowledgment

The research was supported by projects GAČR GA15-03564S, MŠMT 7H14004 and MŠMT 7H14005.

References

1. M. R. Gupta, Y. Chen, *Theory and Use of the EM Method. (Foundations and Trends(r) in Signal Processing).* Now Publishers Inc, 2011.
2. O. Boldea, J. Magnus, "Maximum likelihood estimation of the multivariate normal mixture model," *Journal Of The American Statistical Association*, vol. 104, no. 488, pp. 1539–1549, 2009.
3. H. Wang, B. Luo, Q. Zhang, and S.Wei, "Estimation for the number of components in a mixture model using stepwise split-and-merge em algorithm," *Pattern Recognition Letters*, vol. 25, no. 16, 2004.
4. C. A. McGrory, D. M. Titterington, "Variational Bayesian analysis for hidden Markov models," *Australian & New Zealand Journal of Statistics*, vol. 51, p. 227244, 2009.
5. V. Šmídl, A. Quinn, *The Variational Bayes Method in Signal Processing.* Springer, 2005.
6. S. Frühwirth-Schnatter, *Finite Mixture and Markov Switching Models. 2nd Edition.* Springer New York, 2006.
7. S. Früwirth-Schnatter, "Fully bayesian analysis of switching gaussian state space models," *Annals of the Institute of Statistical Mathematics*, vol. 53, no. 1, pp. 31–49, 2001.
8. A. Doucet and C. Andrieu, "Iterative algorithms for state estimation of jump markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 6, pp. 1216–1227, 2001.
9. R. Chen, J. S. Liu, "Mixture kalman filters," *J. R. Statist. Soc.B*, vol. 62, pp. 493–508, 2000.
10. M. Kárný, J. Böhm, T. V. Guy, L. Jirsa, I. Nagy, P. Nedoma and L. Tesař, *Optimized Bayesian Dynamic Advising: Theory and Algorithms.* London, Springer, 2006.
11. V. Peterka, *Trends and Progress in System Identification.* Oxford: Pergamon Press, 1981, ch. Bayesian system identification, pp. 239–304.
12. M. Kárný, J. Kadlec, and E. Sutanto, "Quasi-bayes estimation applied to normal mixture," in *Preprints of the 3rd European IEEE Workshop on Computer-Intensive Methods in Control and Data Processing*, Eds: J. Rojíček, M. Valečková, M. Kárný, and K. Warwick, CMP '98 /3./, Prague, CZ, September 1998, pp. 77–82.
13. I. Nagy, E. Suzdaleva, M. Kárný, and T. Mlynářová, "Bayesian estimation of dynamic finite mixtures," *Int. Journal of Adaptive Control and Signal Processing*, vol. 25, no. 9, pp. 765–787, 2011.
14. S.K. Ng, G.J. McLachlan. Mixture models for clustering multilevel growth trajectories. *Computational Statistics & Data Analysis.* 71(0), 2014, pages 43–51.
15. I. Vrbik, P. D. McNicholas. Parsimonious skew mixture models for model-based clustering and classification. *Computational Statistics & Data Analysis.* 71(0), 2014, pages 196–210.
16. D. Kerridge, "Inaccuracy and inference," *Journal of Royal Statistical Society B*, vol. 23, pp. 284–294, 1961.