INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
Int. J. Adapt. Control Signal Process. 2011; **25**:765–787
Published online 3 May 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/acs.1239

Bayesian estimation of dynamic finite mixtures

I. Nagy[1, 2, *, †], E. Suzdaleva[2], M. Kárný[2] and T. Mlynářová[1]

[1] *Faculty of Transportation Sciences, Czech Technical University, Na Florenci 25, 110 00 Prague, Czech Republic*
[2] *Institute of Information Theory and Automation, Czech Academy of Sciences, Pod vodárenskou věží 4, 182 08 Prague, Czech Republic*

SUMMARY

The paper introduces an algorithm for estimation of dynamic mixture models. A new feature of the proposed algorithm is the ability to consider a dynamic form not only for component models but also for the pointer model, which describes the activities of the mixture components in time. The pointer model is represented by a table of transition probabilities that stochastically control the switching between the active components in dependence on the last active one. This feature brings the mixture model closer to real multi-modal systems. It can also serve for a prediction of the future behavior of the modeled system. Copyright © 2011 John Wiley & Sons, Ltd.

Received 30 April 2010; Revised 5 November 2010; Accepted 4 February 2011

KEY WORDS: mixture model; Bayesian estimation; clustering; classification; working point detection and prediction

1. INTRODUCTION

Non-linear dynamic systems are frequently met in applications. It is well known that their treatment is difficult. Mixtures as universal approximations of non-linear models [1] are flexible and tractable means of modeling real systems [2–7]. They are especially convenient for a description of such systems that switch their behavior among a finite number of working points. Example of such a situation is a process of supervising a system to check if it operates in a failure-free regime or if it approaches a non-optimal or even failure state.

In general, mixtures are widely used and intensively developed [8–12]. Majority of results focuses on an off-line estimation of static mixture models. They are mostly applied to data mining, i.e. analysis of extensive databases. At the same time, on-line analysis of the data measured on dynamic systems is supported weakly. Static analysis can be worthless, for instance, for operators who want to be automatically warned in critical situations and advised how to improve system performance.

Algorithms based on the so-called variational Bayes method, e.g. [13, 14] represent a significant exception from this state [15, 16]. They provide feasible solutions, however, at the price of using a non-optimal variant of the Kullback–Leibler divergence [17]. Consequently, there is a limited space for their future improvement. A possibility of further development of the mixture estimation theory with an accent on its usability in applications is undertaken in the presented paper. Moreover, it offers a theoretical solution that can be furthermore improved.

[*] Correspondence to: I. Nagy, Faculty of Transportation Sciences, Czech Technical University, Na Florenci 25, 110 00 Prague, Czech Republic.
[†] E-mail: nagy@utia.cas.cz

The presented theory is also closely related to that based on hidden Markov models (HMM). A nice presentation of this approach is given in [18]. However, it concerns only discrete variables and the algorithms run in an off-line mode supported by MC computations. An important feature of the proposed theory is that the algorithms used run in an on-line mode and numerical procedures are applied only to those parts that cannot be computed analytically. In this way, the amount of computations as well as the risk of collapsing is minimized. The approach using HMM is also widely used in solution to applications [19–22].

To characterize the contribution of this paper it is necessary to clarify the meaning of the adjectives static/dynamic. A mixture model consists of a collection of components and a process, pointing at each time instant to the active component. The components are typically represented by static regression models. The first step to make a mixture dynamic is to use dynamic regression models as components. This step has been made in [23, 24]. However, the used pointer model remains static, i.e. probabilities of the component activation are constant. The mixture model becomes truly dynamic if the dynamic pointer model is considered, i.e. if the pointer process is modeled by a Markov chain.

The search for an improved estimation of rich truly dynamic models has been motivated by a specific application, namely, modeling of a car–driver behavior. It should be applied to data measured on a driver and his car during driving. The driver can be in a good mode, tired, nervous, worried, etc. The car can go slowly or quickly, the driving can be responsible or hazardous, etc. The combination of such driver and car modes can lead to a rich combination of system modes. Consequently, the system driver–car can be in a safe mode or approaching safety limits of various character. The research aims to classify the working regimes and to warn the driver against bad or emerging dangerous states. However, this application is by far not the only one. There is a wide range of others where the monitored system dynamically switches from one mode to another and the task is to estimate its current or better to predict its future active modes.

A similar task, which has been solved lately, was that of giving advice to operators supervising the 20-high rolling mill producing metal tin with very high precision in its thickness. Here, the operating of individual operators was not of the same quality. The advising system created clusters in the data space and ex post assigned their evaluation. Then, when an operator worked in a cluster with low value of evaluation, it warned him or even recommended him how to get to the nearest cluster which was better evaluated. This task was solved in the framework of the project [23]. Universal tools developed are described in [24].

A bit different situation leading to a mixture description is traffic control. Here, the system itself has a nature of a mixture of different models. Everybody knows traffic situation in big cities is drastically different in the morning, during the day, in the evening and in the course of the night. If the control is to be relevant, it is necessary to react to a proper situation. Control based on a mixture model can not only distinguish individual mentioned traffic states, but it can also combine them to find the best possible picture of actual traffic.

A number of other tasks requiring multiple-mode dynamic modeling can be found in a variety of applications ranging from technology to societal processes.

In summary, this paper presents the first step to estimate fully dynamic mixture models, where the components and the pointer models are both data dependent. Here, the process of pointer switching is modeled by the Markov chain model with unknown transition probabilities. This model can describe situations when the modeled system stays for a while at a specific working point, described by the active static or dynamic component, and jumps to other working point only from time to time. Such a model covers rather a wide range of situations met in various applications.

The relatively straightforward design of the estimator consists of

- constructing the joint probability density function (pdf) of all observed data and unknown objects and decomposing it into the known or recursively developed pdfs,
- data updating the prior pdfs to posterior ones for recursive estimation of unknown parameters,
- approximating the posterior pdfs, which during each updating step lose their prescribed original form. Kerridge inaccuracy [25], a part of Kullback–Leibler divergence [26], is adopted as

a theoretically justified proximity measure. This divergence is known to be an optimal tool within the Bayesian approach [17].

The paper is organized in the following way. Section 2 introduces the basic elements of dynamic mixture—component models and a pointer model, all in dynamic forms. They are put together in Section 3, where the mixture models with known and unknown parameters are described. The used linear normal model forming a mixture of components is introduced in Section 3.2. In Section 4, the forms of conjugate priors for estimation of both the component and pointer model parameters are provided. The main theoretical emphasis of the paper lies in Section 5, which is devoted to estimation of mixture models. Essentially, the joint pdf is considered. As a by-product, the estimate of the pointer variable is obtained, i.e. the on-line classification problem is solved. Section 6 describes the approximation adopted. It ensures recursive feasibility of computations. The approximation is presented for estimation of both the component and the pointer models. The overall proposed algorithm of mixture model estimation is summarized in Section 6.3. Section 7 provides illustrative experiments. They demonstrate the clustering and classification abilities of the proposed algorithm. Both simulated and real traffic data are processed. Appendix A contains derivations of the proposed formulas.

1.1. Notation

In the following text, these notations will be used:

d_t denotes a real vector of measured data on the system at the discrete time

$$t \in \{0, 1, 2, \ldots, n_t\} \equiv t^*.$$

The time index 0 formally denotes the time period, when the prior data as well as expert information are acquired.

$d(t)$ represents all measured data up to time t also including the prior one d_0, i.e.

$$d(t) = \{d_0, d_1, d_2, \ldots, d_t\}.$$

$f(a|b)$ denotes conditional probability (density) function (pf) or (pdf) of the random variable A with realizations a conditioned by the random variable B, with realizations b. The symbol

- is used for both discrete and continuous random variables A,
- omits the subscript distinguishing different random variables to which it refers. It means that instead of $f_A(a)$ we write only $f(a)$ and the subscript is implied by the name of the argument.

In derivations, the basic rules for operations with conditional pdfs are used [24]. The key ones are:

Chain rule

$$f(a, b|c) = f(a|b, c) f(b|c). \tag{1}$$

Bayes rule

$$f(a|b, c) = \frac{f(b|a, c) f(a|c)}{f(b|c)},$$

which, using the sign of proportionality $\propto$, can be also expressed in the form

$$f(a|b, c) \propto f(a, b|c). \tag{2}$$

2. BASIC ELEMENTS OF A DYNAMIC MIXTURE

There are two types of models connected with the mixtures. They model the components and the pointer to them.

DOI: 10.1002/acs

2.1. *Component model*

Let us consider a mixture model describing n_c different working modes. The model of the cth mixture component, $c=1,2,\ldots,n_c$, describes the system behavior in its cth mode. Generally, it is assumed to have the following 'regression' form, which is later on specialized to linear normal regression model:

$$f(d_t|c,d(t-1),\Theta_c)=f(d_t|c,\phi_{t-1},\Theta_c)\equiv m_{c;t}, \tag{3}$$

where $d_t=[d_{1;t},d_{2;t},\ldots,d_{n_d;t}]'$ is the real vector of data of the length n_d, it is measured at the time instants $t\in t^*$, the apostrophe denotes transposition, $c\in\{1,2,\ldots,n_c\}\equiv c^*$ is the index labeling particular components, $d(t-1)$ denotes the collection of data measured up to and including time $t-1$, it also includes prior data d_0, Θ_c is the collection of unknown parameters of the model of the cth component, ϕ_{t-1} is the regression vector of a fixed finite length, made of data used for predicting d_t. Here, the regression vector is common to all components.

The model (3) is expressed as the joint pdf of the vector modeled variable $d_t=[d_{1;t},d_{2;t},\ldots,d_{n_d;t}]'$. This joint pdf can be factorized according to the chain rule (1) as follows

$$f(d_t|c,\phi_{t-1},\Theta_c)=f(d_{1;t}|c,d_{2;t},\ldots,d_{n_d;t},\phi_{t-1},\Theta_c)\cdots f(d_{n_d;t}|c,\phi_{t-1},\Theta_c). \tag{4}$$

The terms on the right-hand side of the previous expression are called *model factors*. More detailed information about the factors can be found in Appendix A.1.

2.2. *Pointer models*

2.2.1. Predictive pointer model. The mixture model is more than only a collection of components. Its specification needs one more object—a random process $\mathscr{C}=\{c_t\}_{t\in t^*}$, whose items $c_t\in c^*$ point to the active component at each time instant $t\in t^*$. Evolution of this process is assumed

$$f(c_t|c_{t-1},d(t-1),\alpha,\Theta)=f(c_t|c_{t-1},\alpha)=\alpha_{c_t|c_{t-1}}, \tag{5}$$

where the transition probability $\alpha_{c_t|c_{t-1}}$ is a probability that the system will be in mode c_t at time t when its mode at time $t-1$ was c_{t-1}. It holds

$$\alpha\in\alpha^*\equiv\left\{\alpha_{i|j}\geqslant 0,\ \sum_{k\in c^*}\alpha_{k|j}=1,\ \forall i,j\in c^*\right\}.$$

It can be seen in (5) that

- The random variable c_t is assumed independent on the data $d(t-1)$ and the component parameters Θ_c.
- The item c_t of the random process $\mathscr{C}$ is supposed to depend only on its previous item c_{t-1}, i.e. the random process is a dynamic Markov one.
- The pointer c_t has a finite number of possible values so that the transition probabilities $\alpha_{c_t|c_{t-1}}$ form a finite-dimensional matrix, which is used as an unknown parameter of the model for description of the process $\mathscr{C}$.

2.2.2. Posterior probability of the pointer. The model (5) describes time evolution of the pointer. To be able to use it, one must have information about the value of the last pointer c_{t-1}. This is what the following model yields:

$$f(c_{t-1}|d(t-1))\equiv f^c_{c_{t-1};t-1} \tag{6}$$

with the condition $f^c_{c_{t-1};t-1}\geqslant 0$, $\forall c_{t-1}\in c^*$ and $\sum_{c_{t-1}\in c^*}f^c_{c_{t-1};t-1}=1$. Relation (6) introduces a vector of probabilities of component activity at time $t-1$. This vector, at time $t-1$, is a prior pdf for estimation of the pointer model, and it is recursively updated to a posterior pdf, using the currently measured data.

3. MODELS OF DYNAMIC MIXTURE

Here, basic elements describing dynamic mixtures are put together.

3.1. *Model of dynamic mixture with known parameters*

For the given parameters $\Theta=(\Theta_1, \Theta_2, \ldots, \Theta_{n_c})$ and α the mixture model describes the data d_t regardless the knowledge of the active component. The mixture has the form

$$\begin{aligned} f(d_t|d(t-1), \alpha, \Theta) &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*} f(d_t, c_t, c_{t-1}|d(t-1), \alpha, \Theta) \\ &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*} f(d_t|c_t, d(t-1), \Theta) f(c_t|c_{t-1}, \alpha) f(c_{t-1}|d(t-1)), \qquad (7) \end{aligned}$$

where the chain rule (1) has been used. To be able to exploit this mixture, one should have all three models entering (7). If the pointer parameter α and the component parameters $\Theta=[\Theta_1, \Theta_2, \ldots, \Theta_{n_c}]$ are known, the formula (7) can be used directly, as all the pdfs in the decomposition are known. However, if the parameters α and Θ are unknown, they must be included into the set of unknown and thus estimated objects. This case, which is of a practical use, is treated further.

3.2. *Model of dynamic mixture with unknown parameters*

For the unknown mixture parameters α and Θ the predictive pdf (7) is

$$\begin{aligned} f(d_t|d(t-1)) &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\alpha^*}\int_{\Theta^*} f(d_t, c_t, c_{t-1}, \alpha, \Theta|d(t-1))\,\mathrm{d}\alpha\,\mathrm{d}\Theta \\ &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\alpha^*}\int_{\Theta^*} f(d_t|c_t, c_{t-1}, d(t-1), \Theta) f(c_t|c_{t-1}, \alpha) f(c_{t-1}|d(t-1)) \\ &\quad \times f(\alpha, \Theta|d(t-1))\,\mathrm{d}\alpha\,\mathrm{d}\Theta. \qquad (8) \end{aligned}$$

The form of decomposition of the joint pdf $f(d_t, c_t, c_{t-1}, \alpha, \Theta|d(t-1))$ into the pdfs which are known or can be recursively evolved is essential for derivation of the mixture estimation algorithm. The right-hand side of the expression (8) includes the following terms. The first three pdfs correspond to the component model (3), evolution pointer model (5) and pointer posterior pdf (6), respectively. The last pdf is a description of the unknown parameters α and Θ. These parameters are assumed to be conditionally independent

$$f(\alpha, \Theta|d(t-1)) = f(\alpha|d(t-1)) f(\Theta|d(t-1)). \qquad (9)$$

4. CONJUGATE PRIORS

In the previous section, it has been shown that the mixture model is composed of two types of models. The components have the form of normal regression models, the pointer model is a discrete one with multinomial distribution. The conjugate prior for linear normal regression model is the Gauss-inverse-Wishart (GiW) pdf, for the discrete one it is the Dirichlet pdf [24].

4.1. *Conjugate prior pdf for estimation of* Θ

The conjugate prior $f(\Theta|d(t-1))$ to normally distributed components is the GiW pdf

$$f(\Theta|d(t-1)) = \prod_{c\in c^*}\mathcal{G}_\Theta(V_{c;t-1}, \kappa_{c;t-1}) = \prod_{c\in c^*}\frac{g_\Theta(V_{c;t-1}, \kappa_{c;t-1})}{I(V_{c;t-1}, \kappa_{c;t-1})}, \qquad (10)$$

where $g_\Theta = g_{\theta,r}$ denotes the un-normalized GiW pdf

$$g_{\theta,r}(V,\kappa) = r^{-0.5(\kappa+n_\phi+2)} \exp\left\{-\frac{1}{2r}[-1,\theta']V[-1,\theta']'\right\}. \tag{11}$$

The statistic $V_{c;t-1}$ is called the extended information matrix. It is a symmetric and positive-definite matrix having the size of the extended regression vector Φ. The statistics κ is a positive scalar. $I(V,\kappa)$ is the normalization integral

$$I(V,\kappa) = \int_{\Theta^*} g_\Theta(V,\kappa)\,\mathrm{d}\Theta = \Gamma(0.5\kappa) D_d^{-0.5\kappa} |D_\phi|^{-0.5} 2^{0.5\kappa} (2\pi)^{0.5 n_\phi}. \tag{12}$$

The integral is expressed in terms of computationally advantageous decomposition of the extended information matrix $V = L'DL$ with

$$L = \begin{bmatrix} 1 & 0 \\ L_{d\phi} & L_\phi \end{bmatrix}, \quad D = \begin{bmatrix} D_d & 0 \\ 0 & D_\phi \end{bmatrix}.$$

Here, L is a lower triangular matrix with unit diagonal and D is a diagonal matrix with nonnegative items.

The component model (3) has its form as a product of factors (4) which are mutually independent. Similarly, the conjugate prior of individual components can be expressed either in the joint form or as a product of conjugate priors corresponding to individual factors. For normal component models the conjugate priors for factors are also of scalar Gauss-inverse-Wishart (GiW) distribution.

More detailed information about the factors or the decomposition of information matrix and its usage in estimation can be found in Appendix A.2 or in the book [24].

4.2. Conjugate prior pdf for estimation of α

For estimation of parameters of the pointer model (5), the Dirichlet distribution is chosen for the conjugate prior pdf $f(\alpha|d(t-1))$, see [24]. It has the form

$$f(\alpha|d(t-1)) = \mathscr{D}_\alpha(v_{t-1}) = \frac{b_\alpha(v_{t-1})}{B(v_{t-1})}, \tag{13}$$

where

v_{t-1} is an $(n_c \times n_c)$-matrix statistics of the distribution.
$b_\alpha(v_{t-1})$ is the un-normalized Dirichlet pdf

$$b_\alpha = \prod_{c_t \in c^*} \prod_{c_{t-1} \in c^*} \alpha_{c_t|c_{t-1}}^{v_{c_t|c_{t-1};t-1}}.$$

$B(v_{t-1})$ is a normalization constant, which has the form of multivariate beta function [24]

$$B = \prod_{j \in c^*} \frac{\prod_{i \in c^*} \Gamma(v_{i|j;t-1})}{\Gamma\left(\sum_{i \in c^*} v_{i|j;t-1}\right)}. \tag{14}$$

5. ESTIMATION OF THE DYNAMIC MIXTURE

According to the Bayes rule (2), estimation consists in an evolution of the pdfs of unknown variables from (8), using information from the currently measured data. Expressing the unknown variables as vector $U = [c_t, c_{t-1}, \alpha, \Theta]$, one can write the following form of the Bayes rule:

$$f(U|d(t)) \propto f(d_t, U|d(t-1)) = f(d_t|d(t-1), U) f(U|d(t-1)).$$

The basic step in the use of the above formula is the construction of the joint pdf $f(U|d(t))$.

5.1. Construction of the joint pdf

According to (8), the joint pdf can be built up with the help of the conditional pdfs in the following way:

$$
\begin{aligned}
f(c_t, c_{t-1}, \alpha, \Theta|d(t)) &\propto f(d_t, c_t, c_{t-1}, \alpha, \Theta|d(t-1))\\
&= f(d_t|c_t, \phi_{t-1}, \Theta_c) f(c_t|c_{t-1}, \alpha) f(c_{t-1}|d(t-1)) f(\alpha|d(t-1)) f(\Theta|d(t-1))\\
&= m_{c_t;t}\alpha_{c_t|c_{t-1}} f^c_{c_{t-1};t-1} \mathscr{D}_\alpha(v_{t-1}) \mathscr{G}_\Theta(V_{t-1}, \kappa_{t-1}), \qquad (15)
\end{aligned}
$$

where the following denotations have been used: $m_{c_t;t} = f(d_t|c_t, \phi_{t-1}, \Theta_c)$ is the component model (3), $\alpha_{c_t|c_{t-1}} = f(c_t|c_{t-1}, \alpha)$ is the predictive pointer model (5), $f^c_{c_{t-1};t-1} = f(c_{t-1}|d(t-1))$ is a denotation for the prior pdf of the pointer (6), $\mathscr{D}_\alpha(v_{t-1})$ is the conjugate prior pdf for α (13), $\mathscr{G}_\Theta(V_{t-1}, \kappa_{t-1})$ is the conjugate prior pdf for Θ estimation (A3).

The assumptions accepted in (3), (5) and (6) are respected in the decomposition (15). The following arrangements can be made for the formula (15).

Update of the statistics for estimation of α. The product $\alpha_{c_t|c_{t-1}}\mathscr{D}_\alpha(v_{t-1})$ in (15) can be written in the form

$$\alpha_{c_t|c_{t-1}}\mathscr{D}_\alpha(v_{t-1}) = \hat{\alpha}_{c_t|c_{t-1}}\mathscr{D}_\alpha(v^{[c_t,c_{t-1}]}_{t-1}) \qquad (16)$$

with

$$v^{[c_t,c_{t-1}]}_{i|j;t-1} \equiv v_{i|j;t-1} + \delta(c_t, i)\delta(c_{t-1,j}) \quad \forall i, j \in c^* \qquad (17)$$

and

$$\hat{\alpha}_{c_t|c_{t-1}} \equiv v_{c_t|c_{t-1};t-1} \Bigg/ \sum_{k\in c^*} v_{k|c_{t-1};t-1}, \qquad (18)$$

where $\delta(i, j)$ is the Kronecker delta function (i.e. $\delta(i, j)$ is one for $i = j$ and zero otherwise), $v_{c_t|c_{t-1};t-1}$ is an entry of the statistics v_{t-1} and the time index ${}_{t-1}$ at $\hat{\alpha}_{c_t|c_{t-1}}$ has been omitted for lucidity reasons.

The derivation is available in Appendix A.3.

Update of the statistics for estimation of Θ. Similar to (16), the product $m_{c_t;t}\mathscr{G}_\Theta(V_{t-1}, \kappa_{t-1})$ in (15) can be expressed as

$$m_{c_t;t}\mathscr{G}_\Theta(V_{t-1}, \kappa_{t-1}) = f^d_{c_t;t}\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1}) \qquad (19)$$

with

$$f^d_{c_t;t} = f(d_t|d(t-1))$$

and

$$V^{[c_t]}_{c\iota;t-1} = V_{c\iota;t-1} + \delta(c_t, c)\Phi_{c\iota;t}\Phi_{c\iota;t}{}' \quad \text{and} \quad \kappa^{[c_t]}_{c\iota;t-1} = \kappa_{c\iota;t-1} + \delta(c_t, c) \quad \forall c\iota \in \mathscr{F}, \qquad (20)$$

where $\Phi_{c\iota;t}$ is the extended regression vector of the factor $c\iota$, see (A2), and $\delta(i, j)$ is the Kronecker delta function, i.e. $\delta(i, j)$ is one for $i = j$ and zero otherwise.

Remark

The meaning of the operation producing $V^{[c_t]}_{t-1}$ or $\kappa^{[c_t]}_{t-1}$ is that the statistics is updated as if known, that at the present time instant precisely the c_tth component is active. In this way, only the parts of the statistics corresponding to this component are updated. The rest of the statistics remains unchanged. Similar explanation also holds for the relation (17) and that is why these statistics can be called partially updated.

This derivation can be found in Appendix A.4.

After substitution (16) and (19) into (15) the required joint pdf takes the form

$$f(c_t, c_{t-1}, \alpha, \Theta|d(t)) \propto f^d_{c_t;t}\hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1}\mathscr{D}_\alpha(\nu^{[c_t,c_{t-1}]}_{t-1})\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1})$$
$$= w_{c_t|c_{t-1}}\mathscr{D}_\alpha(\nu^{[c_t,c_{t-1}]}_{t-1})\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1}), \tag{21}$$

where

$$w_{c_t|c_{t-1}} = f^d_{c_t;t}\hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1} \tag{22}$$

is a probability that the system at time t is in the mode c_t, while at time $t-1$ it was in the mode c_{t-1} computed for all data up to time instant t.

5.2. Estimation of the parameter α

The construction of the joint pdf (21) gives the instructions for estimation of the parameters of both the models (3) and (5). The posterior pdf of the parameter α can be derived from the joint pdf (21) in the following way:

$$f(\alpha|d(t)) \propto \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\Theta*} f(c_t, c_{t-1}, \alpha, \Theta|d(t))\,\mathrm{d}\Theta$$
$$= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\Theta*} f^d_{c_t;t}\hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1}\mathscr{D}_\alpha(\nu^{[c_t,c_{t-1}]}_{t-1})\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1})\,\mathrm{d}\Theta$$
$$= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*} w_{c_t|c_{t-1}}\mathscr{D}_\alpha(\nu^{[c_t,c_{t-1}]}_{t-1}). \tag{23}$$

It can be seen that the computation is not feasible because the involved summation destroys the prescribed Dirichlet form of the posterior pdf $f(\alpha|d(t))$. Thus, an approximation restoring this form is necessary. The approximation is discussed in Section 6.

5.3. Estimation of the parameter Θ

The posterior pdf for estimation of the parameter Θ can be similarly to (23) evolved with the help of the joint pdf (21). It reads as

$$f(\Theta|d(t)) \propto \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\alpha*} f(c_t, c_{t-1}, \alpha, \Theta|d(t))\,\mathrm{d}\alpha$$
$$= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}\int_{\alpha*} f^d_{c_t;t}\hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1}\mathscr{D}_\alpha(\nu^{[c_t,c_{t-1}]}_{t-1})\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1})\,\mathrm{d}\alpha$$
$$= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*} w_{c_t|c_{t-1}}\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1}) = \sum_{c_t\in c^*} w_{c_t}\mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1}) \tag{24}$$

with $w_{c_t|c_{t-1}}$ defined in (22) and w_{c_t} defined by the relation

$$w_{c_t} = \sum_{c_{t-1}\in c^*} w_{c_t|c_{t-1}}. \tag{25}$$

Again, the prescribed GiW form of the posterior pdf $f(\Theta|d(t))$ is destroyed due to the summation. Similar to the parameter α case, an approximation restoring the pdf form is necessary. The approximation is described in Section 6.

The choice of the prior pdf strongly depends on the amount and quality of the information that is at disposal before the estimation algorithm starts and the data are measured. If no information is at disposal, the priors are constructed as flat distributions reflecting ignorance.

5.4. Estimation of the active component

At each step of the mixture estimation, the currently active component must be estimated. It means that the pdf $f(c_t|d(t))$ is determined as follows.

$$
\begin{aligned}
f(c_t|d(t)) \equiv f^c_{c_t;t} &\propto \sum_{c_{t-1}\in c^*} \int_{\alpha^*} \int_{\Theta^*} f(c_t, c_{t-1}, \alpha, \Theta|\, \mathrm{d}(t))\, \mathrm{d}\alpha\, \mathrm{d}\Theta \\
&= \sum_{c_{t-1}\in c^*} \int_{\alpha^*} \int_{\Theta^*} f^d_{c_t;t} \hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1} \mathscr{D}_\alpha(v^{[c_t,c_{t-1}]}_{t-1}) \mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1})\, \mathrm{d}\alpha\, \mathrm{d}\Theta \\
&= f^d_{c_t;t} \sum_{c_{t-1}\in c^*} \hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1} = w_{c_t} \qquad (26)
\end{aligned}
$$

as the integrals over α^* and Θ^* are equal to one. The definition of the weights $w_{c_t|c_{t-1}}$ is given in (25) and (22).

This part of the algorithm coincides with the task of classification. In the adopted approach it appears naturally as a part of mixture estimation.

Remark
The weights (25) have the meaning of component predictive pdf. It means, they show probabilities of activities of individual components. These probabilities arise from two sources. One source is represented by component prediction produced by the last two members of the definition in (25). The second source is data predictive pdf, given by the first term in (25). The data prediction takes into account the currently measured data item and computes the probability, that the current data might have been generated from individual components. The component prediction is based only on old data, up to time $t-1$ and the prediction is computed from the currently estimated dynamic Markov model giving the stochastic link between two successive values of active components.

Generally, it can be said that the proper choice of the prior pdf for both components and the pointer is an important and not an easy task. That is why, for a choice of the number components and their initial setting, an initialization procedure, based on a prior data sample, has been constructed. Its detailed description can be found in [24]. The estimation algorithms discussed here rely on its exploitation.

5.5. Structural algorithm of the dynamic mixture estimation

The structure of one step of the mixture estimation algorithm can be presented as follows. In the (present) time instant t one has to do

```
for all factors $c\imath$ do
  for each $c_t$ and $c_{t-1}$ do
    1. Update the statistics $v_{t-1}$, $V_{t-1}$, $\kappa_{t-1}$ by the measured data on condition that the label of the active component is $c_t$, and the last active component was $c_{t-1}$. Thus, we obtain the partially updated statistics $v^{[c_t,c_{t-1}]}_{t-1}$, $V^{[c_t]}_{t-1}$, $\kappa^{[c_t]}_{t-1}$ according to (17) and (20).
    2. Construct weights $w_{c_t}$ according to (25) and (22).
    3. Compute updated probabilities of the actual component $f^c_{c_t;t} = w_{c_t}$ according to (26).
    4. Update the pdf for the parameter $\alpha$ according to (23) and approximate it.
    5. Update the pdf describing the parameter $\Theta$ according to (24) and approximate it.
  end of $c_t$, $c_{t-1}$
end of $c\imath$
```

Remark
The previous paragraphs were devoted to the task of estimation of the unknown parameters α and Θ of the mixture model (8). This task in terms of 'data analysis' can be called clustering (learning). It is the phase when the clusters are allocated in the data space. The second phase of data analysis

DOI: 10.1002/acs

is classification when the incoming data items are assigned to or distributed among the fixed positioned clusters. This task is performed by estimating the current value of the pointer variable. In the area of data mining, both the phases clustering and classification are usually separated. Here, they are rather mixed. The parameter estimation (clustering) is based on the pointer estimation and vice versa; during the pointer estimation (classification) the parameter estimates can still be corrected by the coming data.

6. APPROXIMATION IN THE MIXTURE ESTIMATION

As it has been mentioned, the estimation of the parameters α and Θ is not straightforward and to obtain feasible recursion it needs approximation. The approximation has to restore the original forms of prior pdfs, i.e. Dirichlet pdf for estimation of α and GiW pdf for estimation of Θ. It means that one has to find the Dirichlet pdf $\hat{f}(\alpha|d(t))$ and the GiW pdf $\hat{f}(\Theta|d(t))$ which are as close as possible to the posterior pdfs $f(\alpha|d(t))$ in (23) and $f(\Theta|d(t))$ in (24), i.e.

$$f(\alpha|d(t)) \rightarrow \hat{f}(\alpha|d(t)) \sim \mathscr{D}_\alpha(v_t), \tag{27}$$

$$f(\Theta|d(t)) \rightarrow \hat{f}(\Theta|d(t)) \sim \mathscr{G}_\Theta(V_t, \kappa_t). \tag{28}$$

To measure the proximity of the pdfs, the Kerridge inaccuracy is used

$$K_P(f \| \hat{f}) = \int_{P^*} f(P) \ln \frac{1}{\hat{f}(P)} \mathrm{d}P,$$

where P stands either for $P=\alpha$ or $P=\Theta$.

In the next estimation step, the approximations are taken as the prior pdfs.

6.1. *Approximation of the pdf of* α

For approximation of the pdf of α we require that the approximant $\hat{f}(\alpha|d(t)) = \mathscr{D}_\alpha(v_t)$ has the Dirichlet pdf and that it minimizes the Kerridge inaccuracy with substituted pdf (23)

$$\begin{aligned} K_\alpha &= \int_{\alpha^*} f(\alpha|d(t)) \ln \frac{1}{\hat{f}(\alpha|d(t))} \mathrm{d}\alpha \\ &= \int_{\alpha^*} \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}} \mathscr{D}_\alpha(v_{t-1}^{[c_t, c_{t-1}]}) \ln \frac{1}{\mathscr{D}_\alpha(v_t)} \mathrm{d}\alpha, \end{aligned} \tag{29}$$

with respect to the statistics v_t, with entries $v_{i|j;t}$, $i \in c^*$, $j \in c^*$. This task leads to a solution of the non-linear system of algebraic equations

$$G \Xi(v_{i|j;t}) - H_{i|j} = 0, \quad i,\ j \in c^*,$$

where

$$G \equiv \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}},$$

$$H_{i|j} \equiv \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}} \Xi(v_{i|j;t-1}^{[c_t, c_{t-1}]}), \quad i, j \in c^* \tag{30}$$

with

$$\Xi(v_{i|j}) = \Psi(v_{i|j}) - \Psi\left(\sum_{k \in c^*} v_{k|j}\right), \quad i, j \in c^*,$$

where the Ψ function is

$$\Psi(z) = \frac{\mathrm{d}}{\mathrm{d}z} \ln \Gamma(z).$$

DOI: 10.1002/acs

The above-mentioned problem of the solution to the non-linear system of algebraic equations leads to a numerical minimization. However, the function to be minimized is convex (see [27]), so the solution, using e.g. the Newton method, is quick and stable.

The derivation of the result presented above is sketched in Appendix A.5.

6.2. Approximation of the pdf of Θ

The approximation of the pdf $f(\Theta|d(t))$ from (24) searches for a pdf $\hat{f}(\Theta|d(t)) = \mathcal{G}_{\Theta}(V_t, \kappa_t)$ with GiW distribution that minimizes the Kerridge inaccuracy

$$\begin{aligned} K_{\Theta} &= \int_{\Theta^*} f(\Theta|d(t)) \ln \frac{1}{\hat{f}(\Theta|d(t))} \mathrm{d}\Theta \\ &= \int_{\Theta^*} \sum_{c_t \in c^*} f^d_{c_t;t} \sum_{c_{t-1} \in c^*} \hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1} \mathcal{G}_{\Theta}(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1}) \ln \frac{1}{\mathcal{G}_{\Theta}(V_t, \kappa_t)} \mathrm{d}\Theta \end{aligned} \tag{31}$$

with respect to the statistics V_t and κ_t for $t \in t^*$.

The previous formula can be written for factors (see (4) and (A3))

$$K_{\Theta} = \sum_{c \in c^*} \sum_{\imath \in \imath^*} \sum_{c_t \in c^*} w_{c_t} K(\mathcal{G}_{\Theta}(V^{[c_t]}_{c\imath;t-1}, \kappa^{[c_t]}_{c\imath;t-1}) \| \mathcal{G}_{\Theta}(V_{c\imath;t}, \kappa_{c\imath;t})), \tag{32}$$

where $w_{c_t} = f^d_{c_t;t} \sum_{c_{t-1} \in c^*} \hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1}$ is the probability that c_t is the active component at time t, see (26), $\mathcal{G}_{\Theta}(V^{[c_t]}_{c\imath;t-1}, \kappa^{[c_t]}_{c\imath;t-1})$ is the $\imath$th factor of the cth component partially updated by the data d_t, $\mathcal{G}_{\Theta}(V_{c\imath;t}, \kappa_{c\imath;t})$ is the $\imath$th factor of the cth component of the approximating pdf.

The solution to the minimization of (32) can be written explicitly as follows:

Let us denote for each factor $c\imath$, $V_{c\imath} = V$ and let the statistics V be factorized $V = L'DL$ so that L is a lower triangular matrix with unit main diagonal and D is a diagonal one. In addition, let the matrices L and D be partitioned in the following way:

$$L = \begin{bmatrix} 1 & 0 \\ L_{d\phi} & L_{\phi} \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} D_d & 0 \\ 0 & D_{\phi} \end{bmatrix},$$

where $L_{d\phi}$ is a column vector with the same dimension as Θ, L_{ϕ} is a lower triangular matrix with ones on the main diagonal, D_d is a positive number and D_{ϕ} is a diagonal matrix with positive entries on its diagonal.

Then the GiW pdf is equivalently described by the statistics $\{V, \kappa\}$ as well as the statistics $\{\vartheta, D_d, C, \kappa\}$, $\vartheta = L^{-1}_{\phi} L_{\phi}$ are the point estimates of the factor parameters ϑ, D_d are the estimates of factor noise variances, $C = D^{-1}_{\phi}$ are the parts of parameter variances and κ are the numbers of steps of freedom of the factors (counters of the data items).

The statistics of the approximate factors $\mathcal{G}_{\Theta}(V_{c\imath;t}, \kappa_{c\imath;t})$ are denoted by $\hat{\kappa}$, $\hat{D}_d$, $\hat{\vartheta}$, $\hat{C}$, the statistics of the approximated pdf factors $\mathcal{G}_{\Theta}(V^{[c_t]}_{c\imath;t-1}, \kappa^{[c_t]}_{c\imath;t-1})$ are denoted by κ_c, $(D_d)_c$, ϑ_c, C_c for $c_t = c \in c^*$. The auxiliary computations are

$$A = \sum_{c \in c^*} w_c \frac{\kappa_c}{(D_d)_c}, \qquad B = \ln(A/2) + \sum_{c \in c^*} w_c [\ln((D_d)_c) - \Psi(0.5\kappa_c)].$$

The statistics of the approximate pdf are calculated as

- $\hat{\kappa} = \frac{1 + \sqrt{1 + \frac{2}{3} B}}{2B}$,
- $\hat{D}_d = \hat{\kappa}/A$,
- $\hat{\vartheta} = \left(\sum_{c \in c^*} w_c (\kappa_c / (D_d)_c) \vartheta_c \right) / A$,
- $\hat{C} = \sum_{c \in c^*} w_c \{ C_c + (\kappa_c / (D_d)_c) [(\vartheta_c - \hat{\vartheta})(\vartheta_c - \hat{\vartheta})'] \}$.

A detailed explanation of the approximation result is available in [28].

6.3. Detailed algorithm of the mixture estimation

The detailed algorithm of the mixture estimation can be presented now. With the known prior pdfs (or those obtained in the last step of the estimation) $f^c_{c_{t-1};t-1}$, ν_{t-1}, V_{t-1} and κ_{t-1} the algorithm for the step at time t includes the following computations:

1. Partially update the statistics

$$V^{[c_t]}_{c\iota;t-1} = V_{c\iota;t-1} + \delta(c_t, c)\Phi_{c\iota;t}\Phi_{c\iota;t}' \quad \text{and} \quad \kappa^{[c_t]}_{c\iota;t-1} = \kappa_{c\iota;t-1} + \delta(c_t, c) \quad \forall c\iota \in \mathscr{F}$$

   according to (20) and (17).
2. Construct data prediction

$$f^d_{c_t;t} = I(V^{[c_t]}_{t-1}; \kappa^{[c_t]}_{t-1}) / I(V_{t-1}; \kappa_{t-1}),$$

   where the integral I is defined in (12) and the updated statistics are according to (20).
3. Determine the point estimate of α according to (18)

$$\hat{\alpha}_{c_t|c_{t-1}} = \frac{\nu_{c_t|c_{t-1};t-1}}{\sum_{k \in c^*} \nu_{k|c_{t-1};t-1}} \tag{33}$$

   with the ν statistics known from the last time $t-1$.
4. Compute pointer predictive pdf $f^c_{c_t;t-1}$

$$f^c_{c_t;t-1} = \sum_{c_{t-1} \in c^*} \hat{\alpha}_{c_t|c_{t-1}} f^c_{c_{t-1};t-1}$$

   as given in (26).
5. Construct the transition weights $w_{c_t|c_{t-1}}$ according to (22)

$$w_{c_t|c_{t-1}} = f^d_{c_t;t} f^c_{c_t;t-1}.$$

6. Compute the updated pointer estimate $f^c_{c_t;t}$ that is equal to the weights w_{c_t} using (26)

$$f^c_{c_t;t} \equiv w_{c_t} = \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}}$$

   substituting the results of the second and the fourth steps of this algorithm.
7. Update the pdf describing α

$$f(\alpha|d(t)) = \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}} \mathscr{D}_\alpha(\nu^{[c_t|c_{t-1}]}_{t-1})$$

   with computed weights $w_{c_t|c_{t-1}}$ and the updated statistics according to (17).
8. Approximate the updated pdf of α by the Dirichlet pdf, see (27)

$$f(\alpha|d(t)) \rightarrow \hat{f}(\alpha|d(t)) \sim \mathscr{D}_\alpha(\nu_t)$$

   according to Section 6.1.
9. Update the pdf describing Θ

$$f(\Theta|d(t)) = \sum_{c_t \in c^*} w_{c_t} \mathscr{G}_\Theta(V^{[c_t]}_{t-1}, \kappa^{[c_t]}_{t-1})$$

   with computed weights $w_{c_t|c_{t-1}}$ and the updated statistics according to (20).

10. Approximate the updated pdf of Θ by a GiW pdf, see (28)

$$f(\Theta|d(t)) \rightarrow \hat{f}(\Theta|d(t)) \sim \mathcal{G}_{\Theta}(V_t, \kappa_t)$$

according to Section 6.2.

7. EXPERIMENTS

The practical goal the authors aim at is a classification of the behavior of a driver and his car during driving. The aim is to warn the driver if his driving is bad (e.g. from the viewpoint of safety, ecology or economy, etc.). It is clear that in some situations the information can come late. If the driver lies in a ditch with the car turned upside down, it is too late to give him information that his driving can be dangerous. For this reason, a good prediction of the classified driver state is highly desirable.

One of the most valuable features of the proposed estimation algorithm is that it is dynamic. It means a reasonable data prediction can be computed. There are two types of data prediction connected with fully dynamic mixtures. The first is a prediction of the output for individual dynamic components. The second is a prediction of the pointer, i.e. prediction of the component that will be active at the specified time instant. It is clear that if the components are mutually far enough, the pointer prediction is much more important. On the other side, if the probabilities of staying in individual components are uniform, the profit from predicting component activities will be minimal. However, if these probabilities are different, the pointer prediction can be relatively accurate and the contribution of the dynamic pointer model could be decisive. The aim of the examples is to demonstrate these facts.

Theoretically it is clear that the dynamic mixtures cannot be worse than the static ones at anytime. However, the algorithms for dynamic mixture estimation are based on more numeric computations and the question if the advantage of carrying more information does not get lost in more approximative computations. The experiments demonstrate that the dynamics of the mixture model prevails the inaccuracy following from the approximation used.

The experiments have the following general scheme:

1. *Initialization*: As it has been mentioned, an initialization algorithm for estimation of a proper number of components and their initial set-up was previously constructed. Its detailed description is given in [24]. It is based on sequential dividing and merging of the existing components. The actions are evaluated by the likelihood function. Initialization can but need not precede the estimation. Here, the initialization was used for the example with real data.
2. *Estimation*: A defined portion of the data sample (mostly the first half of the data) is used for estimation (training) of the mixture model. In this phase, the pointer prediction (classification) is mixed with estimation (learning). Here, the parameters of both component and pointer models are estimated and fixed.
3. *Evaluation*: In this phase the second portion of data is used. The future output values are estimated on the basis of combined component and pointer prediction.
4. *Illustration*: The results obtained in the experiment are illustrated. Graphs demonstrating the reality and its prediction as well as tables with the numerical results are shown. For numerical evaluation, the prediction error PE is computed

$$\mathrm{PE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \hat{e}_t{}'\hat{e}_t}, \tag{34}$$

where N is the data length and $\hat{e}_t$ is the point prediction error at time t.

7.1. Simulated data

The aim of this experiment is to document the function of the proposed estimation algorithm in details, i.e. to visualize the results and to compare them with the simulated system. The data

DOI: 10.1002/acs

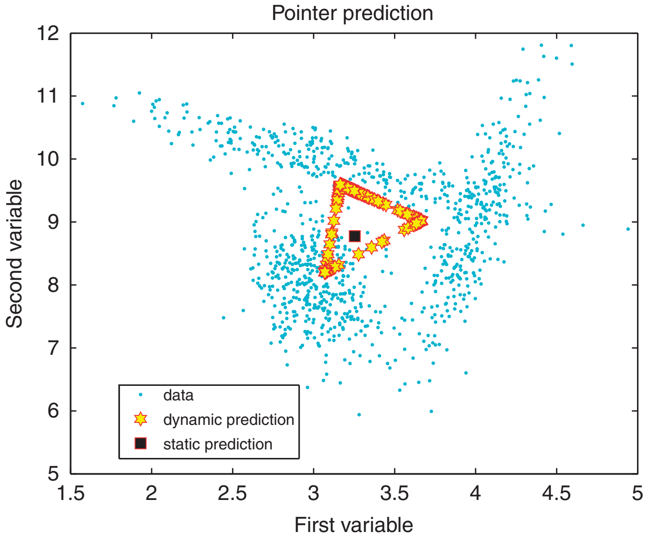

Figure 1. Pointer prediction for static and dynamic pointer model. The data, denoted as small dots create three-shaped Gaussian clusters. The static algorithm estimates the centers and the vector of stationary probabilities assigned to component activities. As the point prediction from a single static component is its center and the activity probabilities are stationary (fixed), the prediction is the same for all data. It is denoted by the dark square. The dynamic algorithm estimates the centers as well, and instead of stationary probabilities of components activities it estimates the dynamic pointer model in the form of transition table. Thus the current pointer estimate depends on its last value and changes in time. The pointer prediction is denoted by the light stars.

sample was simulated using a mixture model with three static normal components with centers (expectations) given in the left-hand side table. The components were shaped by properly chosen covariances (the shapes are visible in Figure 1). The table of simulated transition probabilities was chosen in the form presented in the right-hand side table.

Simulated parameters of the mixture

Component centers	
label	center
1	[3; 8]
2	[3; 10]
3	[4; 9]

Simulated transition probabilities α

0.9	0.05	0.05
0.1	0.1	0.8
0.1	0.8	0.1

These two tables show the most important parameters of the simulated mixture. The left-hand table contains coordinates of centers of the three simulated components. The right-hand table shows the simulated transition probabilities α. *The item* $\alpha_{i,j}$ *is the probability of transition from the component* i *to* j.

These transition probabilities have the highest values: (i) for staying in the first component or (ii) switching between the second and third components. Such a regime is suitable for dynamic mixture and it can best show its properties.

For estimation the same model structure as for simulation has been selected. Thus, a correct estimation should produce practically the same parameters as those used for simulation. The simulated data set of the length 1500 was used. Estimation used 500 data items, the prediction and evaluation were based on 1000 data items.

Table I. Prediction errors for experiment with simulated data.

Static pointer	1.243
Dynamic pointer	1.025

The results of estimation are as follows. Point estimates of the component regression coefficients and the estimated transition probabilities α for both the static and dynamic pointer were equal and they are shown in the following tables:

Estimated parameters of the mixture

Component centers	
label	center
1	$[2.995;\ 8.986]$
2	$[3.021;\ 9.991]$
3	$[3.998;\ 8.968]$

Estimated transition probabilities α

0.856	0.071	0.073
0.145	0.191	0.664
0.122	0.728	0.150

These tables correspond to the above tables with the simulated parameters. Here, the estimates of the simulated parameters are demonstrated. It can be seen, that the estimates are practically the same as the parameters in the simulation.

The tables with simulated parameters and their estimates show a good agreement. It testifies that the problem of current estimation of the component activities will be solved approximately equally well by both the static and the dynamic algorithms. However, in the prediction ability, the situation with the static and the dynamic pointer differs. Figure 1 shows the difference.

Numerical comparison of the overall prediction error (34) is shown in Table I. Owing to the precise estimation and the relatively small number of components, the difference in the prediction quality is not big.

7.2. Real data

7.2.1. Two-dimensional data sample. The simulated data sample can only verify correctness of programming and it cannot testify much to the validity and robustness of the ideas behind the program. To do this, it is necessary to test a real data sample. For that reason, a two-dimensional data sample containing intensity (number of cars per time unit) and density (number of cars per length unit) of the traffic flow in a chosen point of traffic communications in Prague has been considered. These data, plotted in the density–intensity plane, form a concave parabolic noise curve which is well interpretable. Its ideal course starts in the point $[0, 0]$ which corresponds to no traffic. Then it continues with increasing density of traffic flow but still sufficiently low so that there are no interactions between cars in the traffic flow. The interactions begin around the apex of the curve, where the density still grows but the intensity stagnates. Its downfall represents the over-saturation ending in the traffic jam near the density axis. Thus the individual parts of the data curve correspond to the traffic demand and can be used for traffic load classification. However, the data sample used was measured in a tunnel, where standing queues are not allowed. Thus it forms the parabola only in its first half.

The aim of the experiment is to perform a clustering of the data sample that could be potentially used e.g. for estimation of the level of service.

The structure of the experiment is the same as for the case with simulated data. However, for the real data where the optimal number of components is not known, the initialization procedure (see [24, 29]) was used with the first 1000 data. Then, 5000 data items (including those used for initialization) were used for estimation of the mixture model with four components. The following data sample with the length of 1000 items was used for prediction and verification.

Figure 2 shows the data sample in a 3D-view and the estimated centers of components.

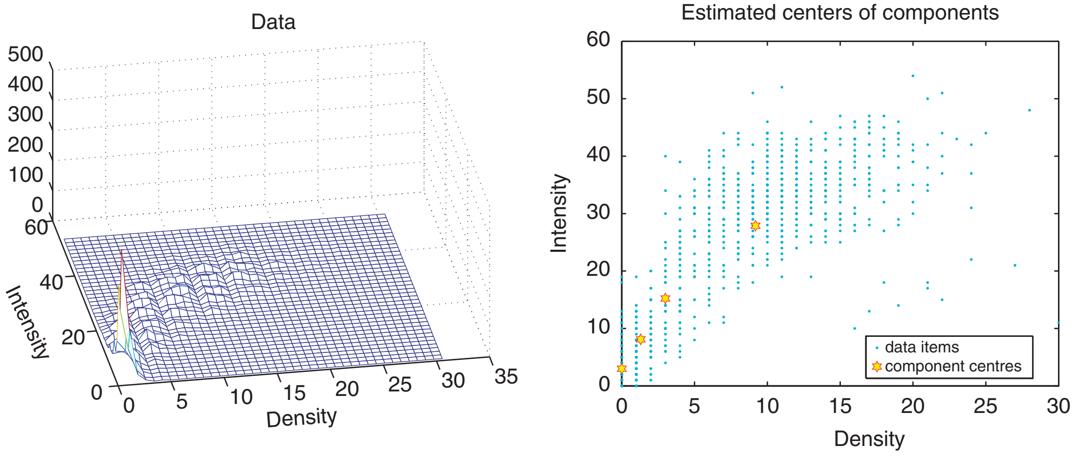

Figure 2. Data (left) and estimated component centers (right). The picture on the left-hand side gives an approximate 3D-view of the data sample used for the example. The graph plots the frequencies of the data, interpreted as points in the horizontal plane. It is evident that there are big differences between the frequencies. The area of the highest data density is the left lower part of the graph. It corresponds to the fact that the tunnel where the data have been measured is mostly empty. The shape of the concave parabola is noticeable, however, it goes only to its apex where it disappears. It is given by the fact that in the tunnel the cars are not allowed to stay for a long time. The right-hand side of the figure shows the data from above. Here, also the centers of the estimated components are visible.

Table II. Prediction errors for experiment with real data.

Static pointer	1.697
Dynamic pointer	1.056

Table III. Variables measured on a driven car.

Variable	Meaning	Variable	Meaning
1	Car position X	7	Speed of car rotation
2	Car position Y	8	Cross acceleration
3	Car position Z	9	Speed of the left front wheel
4	Angle of driving wheel	10	Speed of the right front wheel
5	Position of accelerator	11	Engine revs
6	Power of braking	12	Engine moment

Comparison of the prediction errors for both the static and the dynamic pointer models is shown in Table II. The static error is by 60% higher than the dynamic one.

7.2.2. Multi-dimensional data sample. To be able to make a good classification e.g. of a monitored car behavior, naturally, it is necessary to include as many of the relevant variables as possible into the data sample. To verify whether the proposed estimation algorithm would manage such a situation, an experiment with 12-dimensional data sample of variables, measured on a moving car, was considered. Specifically the variables are listed in Table III.

For estimation, a mixture model with dynamic components of the first order was used. The data length for estimation was 9000 samples from which the first 1000 data was used for the model initialization. Five-hundred successive samples were used for the prediction and results verification.

Figure 3 shows the pointer predictions for the static and the dynamic pointer models.

The resulting comparison of the prediction error is shown in Table IV. Here, the difference is much more significant.

DOI: 10.1002/acs

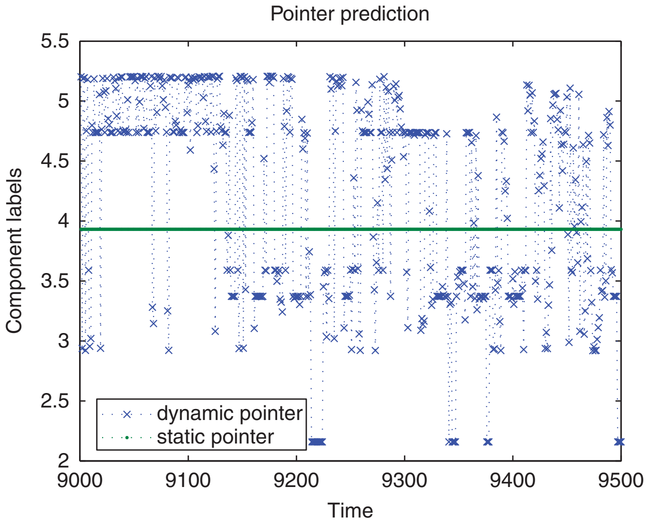

Figure 3. Dynamic and static pointer prediction for 12 variables measured on a moving car. Similar to the example with densities and intensities of the traffic flow, the static prediction is independent on the current data and thus it is constant during the whole time interval in which future data item is compared to its prediction. The dynamic prediction is, due to the estimated transition table, data dependent and thus more close to the predicted data item.

Table IV. Prediction errors for experiment with multi-dimensional data.

Static pointer	5.848
Dynamic pointer	1.458

8. CONCLUSION

The paper proposes a new method of estimation of a mixture model. The main contribution of the proposed method is the use of the dynamic pointer model. In the previously published works [30, 31], the description of the pointer was assumed to be static. Practically, the use of a static pointer model means that each mixture component is assigned a constant probability of its activity. No model of the pointer evolution exists.

In this paper, the pointer is described dynamically via a table of transition probabilities (conditional probability function) in dependence on the last active component. This dynamic pointer description makes possible much more realistic modeling of multi-modal system, where the switching among activities of individual components is realized only from time to time. For such systems, a meaningful prediction of both future data and future active components is possible.

APPENDIX A

A.1. System model in a factorized form

The component model (3) forms the joint pdf for the data vector $d_t = [d_{1;t}, d_{2;t}, \ldots, d_{n_d;t}]'$. It can be expressed as a product of factors according to the chain rule (1)

$$f(d_t|c, d(t-1), \Theta_c) = \prod_{\iota=1}^{n_d} f(d_{\iota;t}|c, d_{\iota+1;t}, \ldots, d_{n_d;t}, \phi_{t-1}, \Theta_{c\iota}), \tag{A1}$$

DOI: 10.1002/acs

where for $\iota = n_d$ the symbol $d_{n_d+1;t} \ldots d_{n_d;t}$ denotes an empty set and

$d_{\iota;t}$ is the ιth entry of the data sample d_t,
$\iota \in \{1, 2, \ldots, n_d\} = \iota^*$ is the index, denoting factors within a specified component,
$\Theta_{c\iota}$ are the parameters of the ιth factor within the cth component of the model (3).

The factors are labeled by double indices $c\iota \in \mathscr{F}$, where $\mathscr{F} = c^* \times \iota^*$ is a set of all double indices of the mixture model, where c denotes a component, ι is a factor within the component c.

Formal factorization into the factors helps in designing the resulting algorithms as all the factors are scalar pdfs.

Generally, the mixture estimation requires no specific form of factor models. Further development relies on the existence of conjugate prior pdfs specified by fixed-dimensional sufficient statistics. Consequently, they have to be chosen from exponential family [32], otherwise, an extra approximation would be needed.

For specificity and wide applicability, factors (4) in the form of linear normal regression models are considered further on

$$f(d_{\iota;t}|c, d_{\iota+1;t}, \ldots, d_{n_d;t}, d(t-1), \Theta_{c\iota}) = (2\pi r_{c\iota})^{-0.5} \exp\left\{-\frac{1}{2r_{c\iota}}[-1, \theta'_{c\iota}]\Phi_{c\iota;t}\Phi'_{c\iota;t}[-1, \theta'_{c\iota}]'\right\}.$$

The model is parameterized by $\Theta_{c\iota} = (\theta_{c\iota}, r_{c\iota})$=(vector of regression coefficients, noise variance). The involved data are collected into the extended regression vector

$$\Phi_{c\iota;t} = [d_{\iota;t}, \ldots, d_{n_d;t}, \phi'_{t-1}]', \tag{A2}$$

with ϕ being the regression vector, see (3).

A.2. Conjugate prior to normal regression model

For estimation of the component models (3) and even the factors within components, the parameters are assumed to be *a priori* independent. This property is preserved during estimation of respective component models even when made within mixture estimation [24]. Thus, the used conjugate prior pdf $f(\Theta|d(t-1))$ of the collection of all component parameters—$\Theta = (\Theta_1, \ldots, \Theta_{n_c})$, where $\Theta_c = (\Theta_{c1}, \ldots, \Theta_{cn_d})$, where $\Theta_{c\iota} = (\theta_{c\iota}, r_{c\iota})$—is the product of GiW pdfs for individual factors.

The decomposition up to the factor level (4)

- increases flexibility of the component model as regression vectors of individual factors even in a single component are allowed to differ,
- simplifies evaluation, as for factors the GiW pdf is scalar and thus it reduces to simpler Gauss-inverse-Gamma pdf,
- simplifies presentation as it can be done without a loss of generality for scalar d_t $(n_d = 1)$: the general case is obtained by using double indices $_{c\iota}$ pointing to the ιth factor in the cth component.

From now on, we use the formal simplification implied by the last item. Thus, the conjugate prior has the form

$$f(\Theta|d(t-1)) = \mathscr{G}_\Theta(V_{t-1}, \kappa_{t-1}) = \prod_{c \in c^*} \mathscr{G}_{\Theta_c}(V_{c;t-1}, \kappa_{c;t-1}) = \prod_{c\iota \in \mathscr{F}} \mathscr{G}_{\Theta_{c\iota}}(V_{c\iota;t-1}, \kappa_{c\iota;t-1}) \tag{A3}$$

where $\mathscr{G}_\Theta$, $\mathscr{G}_{\Theta_c}$ or $\mathscr{G}_{\Theta_{c\iota}}$ denote the Gauss-inverse-Wishart distribution of the whole posterior pdf that of the cth component or that of the factor $c\iota$; and (V_{t-1}, κ_{t-1}), $(V_{c;t-1}, \kappa_{c;t-1})$ or $(V_{c\iota;t-1}, \kappa_{c\iota;t-1})$ are corresponding statistics of the distributions. That is why the GiW distribution can be uniquely denoted by $\mathscr{G}_\Theta$ and its meaning is distinguished only by the argument.

The components for the case $n_d = 1$ coincide with factors.

DOI: 10.1002/acs

A.3. Estimate of the parameter α

The product $\alpha_{c_t|c_{t-1}}\mathscr{D}_\alpha(\nu_{t-1})$ entering the formula for the joint pdf (15) is

$$\alpha_{c_t|c_{t-1}}\mathscr{D}_\alpha(\nu_{t-1}) = \alpha_{c_t|c_{t-1}}\frac{\prod_{i\in c^*}\prod_{j\in c^*}\alpha_{i|j}^{\nu_{i|j;t-1}}}{B(\nu_{t-1})} = \frac{\prod_{i\in c^*}\prod_{j\in c^*}\alpha_{i|j}^{\nu_{i|j;t-1}+\delta(c_t,i)\delta(c_{t-1,j})}}{B(\nu_{t-1})}$$

$$= \frac{B(\nu_{t-1}^{[c_t,c_{t-1}]})}{B(\nu_{t-1})}\mathscr{D}_\alpha(\nu_{t-1}^{[c_t,c_{t-1}]}) = \hat{\alpha}_{c_t|c_{t-1}}\mathscr{D}_\alpha(\nu_{t-1}^{[c_t,c_{t-1}]}),$$

where

$$\nu_{i|j;t-1}^{[c_t,c_{t-1}]} \equiv \nu_{i|j;t-1} + \delta(c_t,i)\delta(c_{t-1,j}) \quad \forall i,j\in c^*$$

and

$$B(\nu_{t-1}^{[c_t,c_{t-1}]})/B(\nu_{t-1}) = \frac{\Gamma(\nu_{c_t|c_{t-1};t-1}+1)}{\Gamma\left(\sum_{k\in c^*}\nu_{k|c_{t-1};t-1}+1\right)}\prod_{j\in\{c^*\backslash c_{t-1}\}}\left[\frac{\prod_{i\in\{c^*\backslash c_t\}}\Gamma(\nu_{i|j;t-1})}{\Gamma\left(\sum_{k\in c^*}\nu_{k|j;t-1}\right)}\right]$$

$$= \left[\Gamma(\nu_{c_t|c_{t-1};t-1}+1)/\,\Gamma\left(\sum_{k\in c^*}\nu_{k|c_{t-1};t-1}+1\right)\right]\Big/ B(\nu_{t-1})$$

$$= \left[\nu_{c_t|c_{t-1};t-1}\Big/\sum_{k\in c^*}\nu_{k|c_{t-1};t-1}\right]B(\nu_{t-1})\Big/ B(\nu_{t-1})$$

$$= \left[\nu_{c_t|c_{t-1};t-1}\Big/\sum_{k\in c^*}\nu_{k|c_{t-1};t-1}\right] \equiv \hat{\alpha}_{c_t|c_{t-1}},$$

where $\{c^*\backslash c_t\}$ denotes the set of all components but the c_tth one.

The previous derivation is based on the definition of the Dirichlet pdf (13) and its normalization constant (14), and on the basic formula for the gamma function (for z scalar)

$$\Gamma(z+1) = z\Gamma(z).$$

A.4. Data prediction

The product $m_{c_t;t}\mathscr{G}_\Theta(V_{t-1},\kappa_{t-1})$ can be found in the formula (15). It can be further modified

$$m_{c_t;t}\mathscr{G}_\Theta(V_{t-1},\kappa_{t-1}) = m_{c_t;t}\prod_{c\in c^*}\mathscr{G}_\Theta(V_{c;t-1},\kappa_{c;t-1})$$

$$= m_{c_t;t}\prod_{c\in c^*}\frac{g_\Theta(V_{c;t-1},\kappa_{c;t-1})}{I(V_{c;t-1},\kappa_{c;t-1})} \tag{A4}$$

according to (A3) and (10). The function g_Θ is a product of component (factor) model pdfs for time instants $\tau = 1,2,\ldots,t-1$. It is the likelihood multiplied by the prior pdf. With respect to the Gaussian pdf of the component model and the conjugate form of the prior pdf for time $t-1$, the update of the statistics $V_{c;t-1}$ and $\kappa_{c;t-1}$, $c = 1,2,\ldots,n_c$ in the function g_Θ when multiplied by the model $m_{c_t;t}$ is as follows:

$$V_{c\imath;t-1}^{[c_t]} = V_{c\imath;t-1} + \delta(c_t,c)\Phi_{c\imath;t}\Phi'_{c\imath;t} \quad \text{and} \quad \kappa_{c\imath;t-1}^{[c_t]} = \kappa_{c\imath;t-1} + \delta(c_t,c) \quad \forall c\in c^*,\ \imath\in\imath^*$$

where $\delta(i,j)$ is the Kronecker delta function (i.e. $\delta(i,j)$ is one for $i=j$ and zero otherwise).

DOI: 10.1002/acs

After substitution into the numerator of (A4), we obtain

$$\begin{aligned}\prod_{c\in c^*}\frac{g_\Theta(V^{[c_t]}_{c;t-1},\kappa^{[c_t]}_{c;t-1})}{I(V_{c;t-1},\kappa_{c;t-1})} &= \prod_{c\in c^*}\frac{I(V^{[c_t]}_{c;t-1},\kappa^{[c_t]}_{c;t-1})}{I(V_{c;t-1},\kappa_{c;t-1})}\mathscr{G}_\Theta(V^{[c_t]}_{c;t-1},\kappa^{[c_t]}_{c;t-1})\\ &= \frac{I(V_{c_t;t-1}+\Phi_{c;t}\Phi_{c;t},\kappa_{c_t;t-1}+1)}{I(V_{c_t;t-1},\kappa_{c_t;t-1})}\prod_{c\in c^*}\mathscr{G}_\Theta(V^{[c_t]}_{c;t-1},\kappa^{[c_t]}_{c;t-1})\\ &= f^d_{c_t;t}\prod_{c\in c^*}\mathscr{G}_\Theta(V^{[c_t]}_{c;t-1},\kappa^{[c_t]}_{c;t-1}),\end{aligned}$$

where $f^d_{c_t;t}=I(V^{[c_t]}_{c_t;t-1},\kappa^{[c_t]}_{c_t;t-1})/I(V_{c_t;t-1},\kappa_{c_t;t-1})$ is the data prediction from the component c_t with the integral defined in (12), and $\Phi_{c;t}$ the extended regression vector.

A.5. Approximation of posterior pdf for α

After updating the Dirichlet prior pdf $f(\alpha|d(t-1))\rightarrow f(\alpha|d(t))$, its form is destroyed in (23), and must be approximately restored. According to (29) the task is to minimize the Kerridge inaccuracy K_α with respect to the statistic v_t

$$K_\alpha=\int_{\alpha^*}\sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}w_{c_t|c_{t-1}}\mathscr{D}_\alpha(v^{[c_t,c_{t-1}]}_{t-1})\ln\frac{1}{\mathscr{D}_\alpha(v_t)}\,\mathrm{d}\alpha.$$

Substituting for $\mathscr{D}_\alpha(v_t)$ from the definition of the Dirichlet pdf (13) and changing summation and integration, we obtain

$$\begin{aligned}K_\alpha &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}w_{c_t|c_{t-1}}\int_{\alpha^*}\mathscr{D}_\alpha(v^{[c_t,c_{t-1}]}_{t-1})\ln\frac{B(v_t)}{\prod_{i\in c^*}\prod_{j\in c^*}\alpha_{i|j}^{v_{i|j;t}-1}}\,\mathrm{d}\alpha\\ &= \sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}w_{c_t|c_{t-1}}\int_{\alpha^*}\mathscr{D}_\alpha(v^{[c_t,c_{t-1}]}_{t-1})\left[\ln B(v_t)-\sum_{i\in c^*}\sum_{j\in c^*}(v_{i|j;t}-1)\ln\alpha_{i|j}\right]\mathrm{d}\alpha.\end{aligned}$$

Let us define the function $\Xi(v_{i|j})$ as

$$\Xi(v_{i|j})=\Psi(v_{i|j})-\Psi\left(\sum_{k\in c^*}v_{k|j}\right),\quad i,j\in c^*,$$

where $\Psi(v_{i|j})=d/dv_{i|j}\ln\Gamma(v_{i|j})$ and $\Gamma(\cdot)$ is the gamma function. Then it holds

$$\int_{\alpha^*}\ln(\alpha_{i|j})\mathscr{D}_\alpha(v)d\alpha=\Xi(v_{i|j}),\quad i,j\in c^* \tag{A5}$$

as well as

$$\frac{\partial}{\partial v_{i|j}}\ln B(v)=\Xi(v_{i|j}),\quad i,j\in c^*. \tag{A6}$$

Taking into account (A5) the expression K_α takes the form

$$K_\alpha=\ln B(v_t)\sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}w_{c_t|c_{t-1}}-\sum_{c_t\in c^*}\sum_{c_{t-1}\in c^*}w_{c_t|c_{t-1}}\sum_{i,j\in c^*}(v_{i|j;t}-1)\Xi(v^{[c_t,c_{t-1}]}_{i|j;t-1}).$$

Minimum of K_α can be found as zero point of its derivative. As the minimized function is convex (see [27]) the result, even using numerical minimization, is unambiguous and easy to find. It is given by the solution of the system of equations

$$G\Xi(v_{i|j;t})-H_{i|j}=0,\quad i,j\in c^*,$$

where

$$G \equiv \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}},$$

$$H_{i|j} \equiv \sum_{c_t \in c^*} \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}} \Xi(v_{i|j;t-1}^{[c_t,c_{t-1}]}), \quad i, j \in c^*$$

and the second property (A6) has been used.

A.6. Approximation of posterior pdf for Θ

Similar to the parameter α, the update of the pdf describing Θ (24) destroys its GiW form and must be approximately restored. The task is to minimize the Kerridge inaccuracy

$$K_\Theta = \int_{\Theta^*} \sum_{c_t \in c^*} w_{c_t} \mathscr{G}_\Theta(V_{t-1}^{[c_t]}, \kappa_{t-1}^{[c_t]}) \ln \frac{1}{\mathscr{G}_\Theta(V_t, \kappa_t)} \mathrm{d}\Theta$$

with respect to the statistics V_t and κ_t. The expression for the updated pdf for Θ, i.e. the expression under the integral and before the logarithm, is taken from (24) with the denotation for the weights $w_{c_t} = \sum_{c_{t-1} \in c^*} w_{c_t|c_{t-1}}$ from (22) and (25).

It is advantageous to express the components into the factors (4) and their conjugate priors (A3) in the correspondence to these factors (10), (11). Then the formula for K_Θ is

$$\begin{aligned}
K_\Theta &= \int_{\Theta^*} \sum_{c_t \in c^*} w_{c_t} \prod_{k \in c^*} \prod_{l \in \iota^*} \mathscr{G}_\Theta(V_{kl;t-1}^{[c_t]}, \kappa_{kl;t-1}^{[c_t]}) \ln \frac{1}{\prod_{c \in c^*} \prod_{\iota \in \iota^*} \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t})} \mathrm{d}\Theta \\
&= -\int_{\Theta^*} \sum_{c_t \in c^*} w_{c_t} \prod_{k \in c^*} \prod_{l \in \iota^*} \mathscr{G}_\Theta(V_{kl;t-1}^{[c_t]}, \kappa_{kl;t-1}^{[c_t]}) \ln \prod_{c \in c^*} \prod_{\iota \in \iota^*} \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t}) \mathrm{d}\Theta \\
&= -\int_{\Theta^*} \sum_{c_t \in c^*} w_{c_t} \prod_{k \in c^*} \prod_{l \in \iota^*} \mathscr{G}_\Theta(V_{kl;t-1}^{[c_t]}, \kappa_{kl;t-1}^{[c_t]}) \sum_{c \in c^*} \sum_{\iota \in \iota^*} \ln \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t}) \mathrm{d}\Theta \\
&= -\sum_{c \in c^*} \sum_{\iota \in \iota^*} \sum_{c_t \in c^*} w_{c_t} \int_{\Theta^*} \prod_{k \in c^*} \prod_{l \in \iota^*} \mathscr{G}_\Theta(V_{kl;t-1}^{[c_t]}, \kappa_{kl;t-1}^{[c_t]}) \ln \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t}) \mathrm{d}\Theta \\
&= -\sum_{c \in c^*} \sum_{\iota \in \iota^*} \sum_{c_t \in c^*} w_{c_t} \int_{\Theta_{c\iota}^*} \mathscr{G}_\Theta(V_{c\iota;t-1}^{[c_t]}, \kappa_{c\iota;t-1}^{[c_t]}) \ln \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t}) \mathrm{d}\Theta \\
&= \sum_{c \in c^*} \sum_{\iota \in \iota^*} \sum_{c_t \in c^*} w_{c_t} K(\mathscr{G}_\Theta(V_{c\iota;t-1}^{[c_t]}, \kappa_{c\iota;t-1}^{[c_t]}) \| \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t})).
\end{aligned}$$

It means that the minimization of K_Θ can be done by minimization of weighted sums of Kerridge inaccuracies of individual factors, i.e. by minimization of

$$K_{c\iota} = \sum_{c_t \in c^*} w_{c_t} K(\mathscr{G}_\Theta(V_{c\iota;t-1}^{[c_t]}, \kappa_{c\iota;t-1}^{[c_t]}) \| \mathscr{G}_\Theta(V_{c\iota;t}, \kappa_{c\iota;t})) \quad \forall c \in c^*, \iota \in \iota^*.$$

Now the task is to find the minimum of each $K_{c\iota}$ for $c\iota \in \mathscr{F}$. From [24] we have the Kullback–Leibler divergence of two GiW 'factors', which is

$$\begin{aligned}
&D(\mathscr{G}(\vartheta, D_d, C, \kappa) \| \mathscr{G}(\hat{\vartheta}, \hat{D}_d, \hat{C}, \hat{\kappa})) \\
&\quad = \ln\left(\frac{\Gamma(0.5\hat{\kappa})}{\Gamma(0.5\kappa)}\right) - 0.5\ln(C\hat{C}^{-1}) + 0.5\hat{\kappa}\ln\left(\frac{D_d}{\hat{D}_d}\right) + 0.5(\kappa - \hat{\kappa})\Psi(0.5\kappa) \\
&\qquad - 0.5n_\phi - 0.5\kappa + 0.5\mathrm{tr}(C\hat{C}^{-1}) + 0.5\frac{\kappa}{D_d}[(\vartheta - \hat{\vartheta})'\hat{C}^{-1}(\vartheta - \hat{\vartheta}) + \hat{D}_d],
\end{aligned}$$

where $\mathscr{G}$ denotes the same GiW distributions but one is with the statistics ϑ, D_d, C, κ and the other with the statistics $\hat{\vartheta}$, $\hat{D}_d$, $\hat{C}$, $\hat{\kappa}$. The GiW distribution is defined in Section 6.2. As the Kerridge inaccuracy and Kullback–Leibler divergence differ only in a constant, their points of minimum are equal. Thus the Kullback–Leibler divergence, for which the necessary results are already derived in [24], can be used for our purpose.

After differentiation according to individual factor statistics $\hat{\vartheta}_{c\iota}$, $(\hat{D}_d)_{c\iota}$, $\hat{C}_{c\iota}$, $\hat{\kappa}_{c\iota}$ and using an approximative expression for the function Γ we obtain the results listed in the end of Section 6.2.

ACKNOWLEDGEMENTS

This work has been partially supported by the MŠMT project 1M0572 DAR, the treaty between ÚTIA and Škoda ENS/2010/UTIA, projects GAČR 102/08/0567 and TAČR TA01030123.

REFERENCES

1. Haykin S. *Neural Networks*: *A Comprehensive Foundation*. Macmillan: New York, 1994.
2. Schoenberg JR, Campbell M. Distributed Terrain estimation using a mixture-model based algorithm. *FUSION*: *2009 12th International Conference on Information Fusion*, Seattle, WA, vols. 1–4, 6–9 July 2009; 960–967.
3. Morita S, Thall PF, Bekele BN, Mathew P. A Bayesian hierarchical mixture model for platelet-derived growth factor receptor phosphorylation to improve estimation of progression-free survival in prostate cancer. *Journal of the Royal Statistical Society Series C—Applied Statistics* 2010; **59**(Part 1):19–34.
4. Ayres FJ, Campbell GM, Boyd SK. Automatic estimation of body composition using Micro-CT and a Gaussian mixture model. *Journal of Bone and Mineral Research* 2008; **23**(Suppl. S):S408. *30th Annual Meeting of the American-Society-for-Bone-and-Mineral-Research*, Montreal, Canada, 12–16 September 2008.
5. Lee S, Lathrop R. Sub-pixel estimation of urban land cover components with linear mixture model analysis and Landsat Thematic Mapper imagery. *International Journal of Remote Sensing* 2005; **26**(22):4885–4905. DOI: 10.1080/01431160500300222.
6. Nilsson M, Gustafsson H, Andersen S, Kleijn W. Gaussian mixture model based mutual information estimation between frequency bands in speech. *2002 IEEE International Conference on Acoustics*, *Speech*, *and Signal Processing*, vols. I–IV, Orlando, FL, 13–17 May 2002; 525–528.
7. Peng Y, Dear K. A nonparametric mixture model for cure rate estimation. *Biometrics* 2000; **56**(1):237–243.
8. Boldea O, Magnus JR. Maximum likelihood estimation of the multivariate normal mixture model. *Journal of the American Statistical Association* 2009; **104**(488):1539–1549. DOI: 10.1198/jasa.2009.tm08273.
9. Cuesta-Albertos JA, Matran C, Mayo-Iscar A. Robust estimation in the normal mixture model based on robust clustering. *Journal of the Royal Statistical Society Series B—Statistical Methodology* 2008; **70**(Part 4):779–802.
10. Wang H, Luo B, Zhang Q, Wei S. Estimation for the number of components in a mixture model using stepwise split-and-merge EM algorithm. *Pattern Recognition Letters* 2004; **25**(16):1799–1809. DOI: 10.1016/j.patrec.2004.07.007.
11. Cho D, Zhang B. Evolutionary continuous optimization by distribution estimation with variational Bayesian independent component analyzers mixture model. In *Eighth International Conference on Parallel Problem Solving from Nature* (*PPSN VIII*), Yao X, Burke E, Lozano JA, Smith J, MereloGuervos JJ, Bullinaria JA, Rowe J, Tino P, Kaban A, Schwefel HP (eds), Birmingham, England, Lecture Notes in Computer Science, vol. 3242, 18–22 September 2004; 212–221.
12. Dacunha Castelle D, Gassiat E. The estimation of the order of a mixture model. *Bernoulli* 1997; **3**(3):279–299.
13. Bishop CM. Variational principal components. *Proceedings of the Ninth International Conference on Artificial Neural Networks*, (*ICANN99*), University Edinburgh, Edinburgh, Scotland, vols. 1, 2(470). IEEE Conference Publications, 7–10 September 1999; 509–514.
14. Šmídl V, Quinn A. *The Variational Bayes Method in Signal Processing*. Springer: Berlin, 2005.
15. Daunizeau J, Kiebel S, Friston K. Dynamic causal modelling of distributed electromagnetic responses. *Neuroimage* 2009; **47**(2):590–601.
16. Qu H, Hu B. Variational learning for generalized associative functional networks in modeling dynamic process of plant growth. *Ecological Informatics* 2009; **4**(3):163–176.
17. Bernardo JM. Expected information as expected utility. *The Annals of Statistics* 1979; **7**(3):686–690.
18. Beal M, Ghahramani Z, Rasmussen C. The infinite hidden Markov model. In *Proceedings of the 15th Annual Conference on Neural Information Processing Systems* (*NIPS*), Vancouver, Canada, 3–8 December 2001; *Advances in Neural Information Processing Systems 14*, Dietterich TG, Becker S, Ghahramani Z (eds), vols. 1, 2. 2002; 577–584.
19. Fujisaki M, Zhang D. Bayesian analysis of compound poisson mixture model and its application to financial data. *International Journal of Innovative Computing*, *Information and Control* 2002; **5**(1):109–118.
20. Hyun Cheol Cho KSL, Fadali MS. Online learning algorithm of dynamic Bayesian networks for nonstationary signal processing. *International Journal of Innovative Computing*, *Information and Control* 2009; **5**(4):1027–1042.

21. Zhen Tian ZY, Yuan K. Automatic detection of abnormal regions using Guassian mixture model. *ICIC Express Letters* 2009; **3**(4):921–926.
22. Chi-I Hsu CC, Ho MSH. The prediction of PKI security performance using PSO and Bayesian classifier. *ICIC Express Letters* 2009; **3**(4):1031–1036.
23. FET, IST12088—ProDaCTool: Decision Support for Complex Industrial Processes Based on Probabilistic Data Clustering, 2000–2002. Available from: http://www.prodactool.rdg.ac.uk/.
24. Kárný M, Bohm J, Guy TV, Jirsa L, Nagy I, Nedoma P, Tesař L. *Optimized Bayesian Dynamic Advising*: *Theory and Algorithms*. Springer: London, 2005.
25. Kerridge D. Inaccuracy and inference. *Journal of Royal Statistical Society* 1961; **23**:284–294.
26. Kullback S, Leibler R. On information and sufficiency. *Annals of Mathematical Statistics* 1951; **22**:79–87.
27. Algoet P, Cover T. A sandwich proof of the Shannon–McMillan–Breiman theorem. *The Annals of Probability* 1988; **16**:899–909.
28. Dedecius K. Partial forgetting in autoregressive models. *Ph.D. Thesis*, 2010.
29. Nedoma P, Kárný M, Guy TV, Nagy I, Böhm J. *Mixtools Program*. ÚTIA AV ČR: Praha, 2003.
30. Kárný M, Kadlec J, Sutanto EL. Quasi-Bayes estimation applied to normal mixture. In *Preprints of the Third European IEEE Workshop on Computer-intensive Methods in Control and Data Processing*, Rojíček J, Valečková M, Kárný M, Warwick K (eds). ÚTIA AV ČR: Prague, 1998; 77–82.
31. Kárný M, Nagy I, Novovičová J. Mixed-data multi-modelling for fault detection and isolation. *International Journal of Adaptive Control and Signal Processing* 2002; **16**(1):61–83.
32. Barndorff-Nielsen O. *Information and Exponential Families in Statistical Theory*. Wiley: New York, 1978.

DOI: 10.1002/acs