The 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
15-17 September 2011, Prague, Czech Republic

Comparison of State Estimation Using Finite Mixtures and Hidden Markov Models

Ivan Nagy,[1] Evgenia Suzdaleva,[2] Tereza Mlynářová[2]
[1] Faculty of Transportation Sciences, Czech Technical University,
Na Florenci 25, 11000 Prague, Czech Republic, nagy@utia.cas.cz
[2] Department of Adaptive Systems, Institute of Information Theory and Automation of the ASCR,
Pod vodárenskou věží 4, 18208 Prague, Czech Republic, suzdalev@utia.cas.cz, t.mlynarova@seznam.cz

***Abstract* – Many various algorithms are developed for state estimation of dynamic switching systems. It is not a straightforward task to choose the most suitable one. This paper deals with testing of state estimation via two well-known approaches: recursive estimation with finite mixtures and iterative technique with hidden Markov models. A discussion of comparison of these online and offline counterparts is of true interest. The paper describes experiments providing a comparison of these methods.**

***Keywords* – mixture models; hidden Markov models; state estimation; online estimation**

I. Introduction

Systems switching dynamically among different modes are met in many application fields. Modeling of behavior of such systems is a difficult task. Finite mixture models [1] and hidden Markov models (HMM) [2] both are widely-spread and powerful tools for description and identification of such systems. Variety of estimation approaches concerned with some of them (and not only) is developed [3]–[9].

In some areas (supervising a rolling mill, traffic flow control, optimization of fuel consumption, medicine) effective online (recursive) estimators based on currently measured data are strongly desired. In other applications offline iterative methods demonstrate needed quality of estimation. However, it is not a straightforward task to see instantly advantages and drawbacks of existing algorithms. In other words, it is not always clear if pluses of online estimation does not lose in the background of necessary approximations against offline methods.

An inspiration for this paper has arisen during the work on developing the novel algorithms [10] for fully dynamic mixture estimation. The question was to compare the effectiveness of the online mixture estimation [1] and offline HMM estimation [2] with the help of experiments with simulated data. This is the main task addressed in the paper.

The paper is organized as follows. Section II provides basic facts about mixtures and hidden Markov models. Section III describes experiments of comparison of the mentioned estimators under different conditions and demonstrates the results. Section IV provides a conclusion and summarizes the work.

The research was partially supported by projects MŠMT 1M0572 and TAČR TA01030123.

II. Preliminaries

A general state-space model used in the paper has the following form

$$\textit{state evolution (transition) model} \qquad f(x_t|x_{t-1},\alpha), \quad (1)$$

$$\textit{output (emission) model} \qquad f(y_t|x_t,\theta), \quad (2)$$

where x_t is an unobserved state to be estimated, α and θ are the unknown model parameters, y_t is a measured output, and $t = 1, 2, 3, \ldots, T$ denotes discrete time moments for all involved processes. In general, the used variables can be as of a discrete as of a continuous nature at time t.

A. Mixture Model

A mixture model represents probability (density) function (p(d)f) in the form of a sum of weighted distributions.

For this paper the following mixture is considered

$$\sum_{i=1}^{C} \alpha_{i|x_{t-1}} f_i(y_t|x_t = i, \theta), \qquad (3)$$

where state x_t is a discrete variable described by the transition table

$$f\left(x_t|x_{t-1},\alpha\right) = P\left(x_t = i|x_{t-1} = j\right) = \alpha_{x_t|x_{t-1}} \qquad (4)$$

with probabilities $\alpha_{x_t=i|x_{t-1}=j}$, $i = 1, \ldots, C$, $j = 1, \ldots, C$ of transition to state i from state j, where $\sum_i^C \alpha_{i|j} = 1, \alpha_{i|j} \geq 0 \;\forall\; i, j$. They represent weights in (3). For example, for the model with two states the transition table can be as follows.

	$x_t = 1$	$x_t = 2$
$x_{t-1} = 1$	$\alpha_{1\|1}$	$\alpha_{2\|1}$
$x_{t-1} = 2$	$\alpha_{1\|2}$	$\alpha_{2\|2}$

Transition probabilities reflect a switching of the system to a possible mode, i.e., indicate an active component at the current time moment t conditioned by the past active component. The i-th component in (3) indexed by state $x_t = i$ is represented by normal model $f_i(y_t|x_t = i, \theta) \sim \mathcal{N}_i(\mu_i, r_i)$, where

$$y_t = \mu_i + e_t, \;\; e_t \sim \mathcal{N}(0, r_i)$$

with mean value μ_i and variance r_i and where e_t is a white normal noise.

B. Hidden Markov Models (HMM)

In the HMM case both models (1)-(2) are discrete and given by tables. The transition model (1) has the same form as for the considered mixture model, i.e., (4). The emission model (2) is considered as

$$f(y_t|x_t, \theta) = P(y_t = k|x_t = i) = \theta_{y_t|x_t} \qquad (5)$$

with probabilities $\theta_{k|i}$ of $y_t = k$, $k = 1, \dots, N$, when the current state is i, and it holds $\sum_k^N \theta_{k|i} = 1, \theta_{k|i} \geq 0 \;\forall\; k, i$. For example, for sets of possible values of the state $x^* = \{1, 2\}$ and of the output $y^* = \{1, 2, 3\}$ respectively this model can take the form

	$y_t = 1$	$y_t = 2$	$y_t = 3$
$x_t = 1$	$\theta_{1\|1}$	$\theta_{2\|1}$	$\theta_{3\|1}$
$x_t = 2$	$\theta_{1\|2}$	$\theta_{2\|2}$	$\theta_{3\|2}$

C. Problem Formulation

For both the mixture models and HMM a problem is to estimate the unobservable state x_t that means a mode to which the system is going to switch currently. The task is to compare the quality of estimation under various conditions of experiments. It is not the aim of the present paper to describe the algorithms exploited in details; they can be found in literature. For mixtures the recursive Quasi-Bayes mixture estimation algorithm proposed in [1] and implemented in MATLAB toolbox [11] developed in authors' department under project ProDaCTool [12] has been used. For HMM estimation the standard iterative procedures available in MATLAB was exploited.

III. Comparison of Mixture and HMM Estimators

In order to compare these two methods, a special form of the HMM emission model with many (greater than 15) possible values of the output variable y_t was considered. Number of data was 300 for all the experiments.

The experiments consist in the following steps.

1) The HMM system was simulated with the output having a certain great number of possible values. The table of probabilities for the output model was constructed so that the probabilities were distributed along a normal-distribution curve with some chosen mean value of the output. The dimension of the state was 2 for all the experiments.
2) The estimation has been performed with the HMM estimator with some initial parameters, to obtain the transition and emission tables.
3) The state was estimated by the HMM method and compared with the simulated one.
4) The same simulated data was used for the mixture estimation with a discrete 2-dimensional state and 2 normal components. The state was estimated and compared with the simulated one.
5) The number of correctly point-estimated states was evaluated for both the methods.
6) Computational time was evaluated for both the algorithms.

During the experimental work it was realized that, in general, three basic factors influence the quality of estimation for both the algorithms. They are: i) distance between mean values in the output probabilities table; ii) number of possible values of the output ($15 - 100$); iii) transition probabilities (more deterministic or more uncertain).

A. Experiments with Distance between Mean Values

These experiments deal with the impact of different distances between mean values in the output probabilities table. The number of possible values for y_t for the HMM simulator was set 15, state transition probabilities were set more deterministic $[0.97 \;\; 0.03; \;\; 0.05 \;\; 0.95]$ in MATLAB notation, so that the system remains some time at one of the states and only then jumps to another one. The experiment was run with various distances between mean values and great variances: from strongly overlapping up to almost not overlapping ones. It was realized, that the mixture estimator is sensitive to strongly overlapping mean values: the quality of estimation was worse, see Fig.1 (left). The HMM shows perfect results, see Fig.1 (right). The number of correctly point-estimated states is 286 for mixtures and 299 for HMM.

However, for the almost not overlapping mean values from the emission table the state estimation was error-free for both the methods.

Comparison of computational time (i.e., elapsed time in seconds) is provided in Table I. It can be seen that the mixture estimator was faster.

Table I. Comparison of Computational Time

	Computational Time
Mixture estimator	1.927 seconds
HMM estimator	2.122 seconds

B. Experiments with Number of Output Possible Values and with Random Start

The experiments were done with increasing number of the output values: 30, 50, 100. The mean values were set as almost not overlapping in order to justify the conditions for both the estimators. The following results have been obtained. Starting with almost precise initial parameters for the HMM estimation, both the methods demonstrate

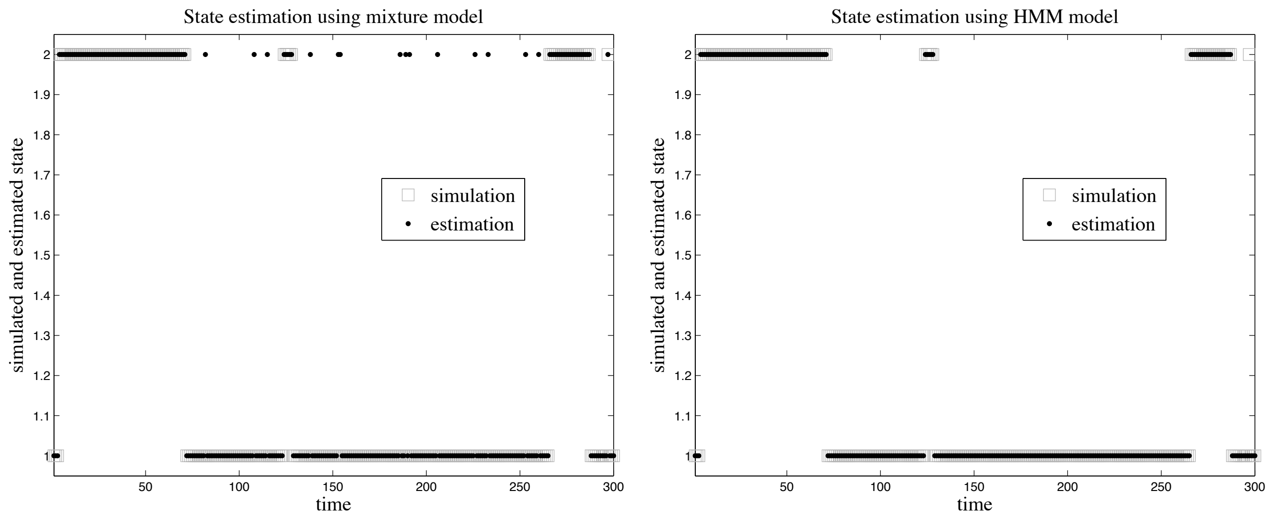

Figure 1. State estimation with strongly overlapping mean values with the mixture estimator (left) and HMM estimator (right). Note that the HMM estimator shows more stability: the number of correctly point-estimated states is 286 for mixtures and 299 for HMM.

very similar nice results with some advantage of the HMM. It shows the error-free state estimation from 30 to 100 possible output values, while the mixture estimator provides insignificant improvement of the results: for 30 values it has 296 correctly point-estimated states, for 100 values – 298. Several errors at the beginning are caused by the insufficient knowledge of the online algorithm at the start of learning.

The same experiments were also done with random parameter start for the HMM estimator. It was realized, that with a random start, the estimation with HMM mostly failed. For 30 output values the HMM method has 181 correctly point-estimated states, for 50 values – 173 , for 100 values – 152. The mixture estimator always started randomly and shows 298 correctly point-estimated states for all these experiments. The results for 100 possible output values with random start can be seen in Fig.2.

Computational time for the experiments with 100 output values and the random parameter start can be seen in Table II for both the methods. Here, the HMM estimator demonstrates a significantly longer computational time than the mixture one.

Table II. COMPARISON OF COMPUTATIONAL TIME

	Computational Time
Mixture estimator	1.929 seconds
HMM estimator	5.123 seconds

C. *Experiments with Transition Probabilities*

These experiments deal with various settings of state transition probabilities. The mean values are chosen almost not overlapping, and the number of possible output values is 15. The initial parameters for the HMM estimation are almost precise. The state transition probabilities are chosen rather uncertain: $[0.6 \quad 0.4; \quad 0.4 \quad 0.6]$, then $[0.55 \quad 0.45; \quad 0.5 \quad 0.5]$ and further $[0.5 \quad 0.5; \quad 0.5 \quad 0.5]$, so that the system jumps from one state to another. The HMM estimator is more sensitive to the uncertain transition model: it provides 296, 299 and 174 correctly point-estimated states respectively. The mixture estimation proves more stability and has 300 correct estimates for all these experiments. The results for the last attempt are shown in Fig.3.

Computational time for the experiments with the state transition probabilities $[0.55 \quad 0.45; \quad 0.5 \quad 0.5]$ is shown in Table III. For probabilities $[0.5 \quad 0.5; \quad 0.5 \quad 0.5]$ the computational time for mixtures was 1.883 seconds and for HMM – 4.918 seconds. It can be seen the mixture estimator was much faster for these experiments.

Table III. COMPARISON OF COMPUTATIONAL TIME

	Computational Time
Mixture estimator	1.395 seconds
HMM estimator	6.163 seconds

IV. CONCLUSION

To summarize the experiments, it can be said, that both the algorithms have their own strong and weak points. The HMM offline estimation proves better results for either strongly or almost not overlapping mean values in the emission table, but it is more sensitive to increasing number of the output possible values and “jumping” uncertain states. The HMM estimation shows the worsened stability with the random initial parameters. The mixture online estimator worsens with more overlapping mean values,

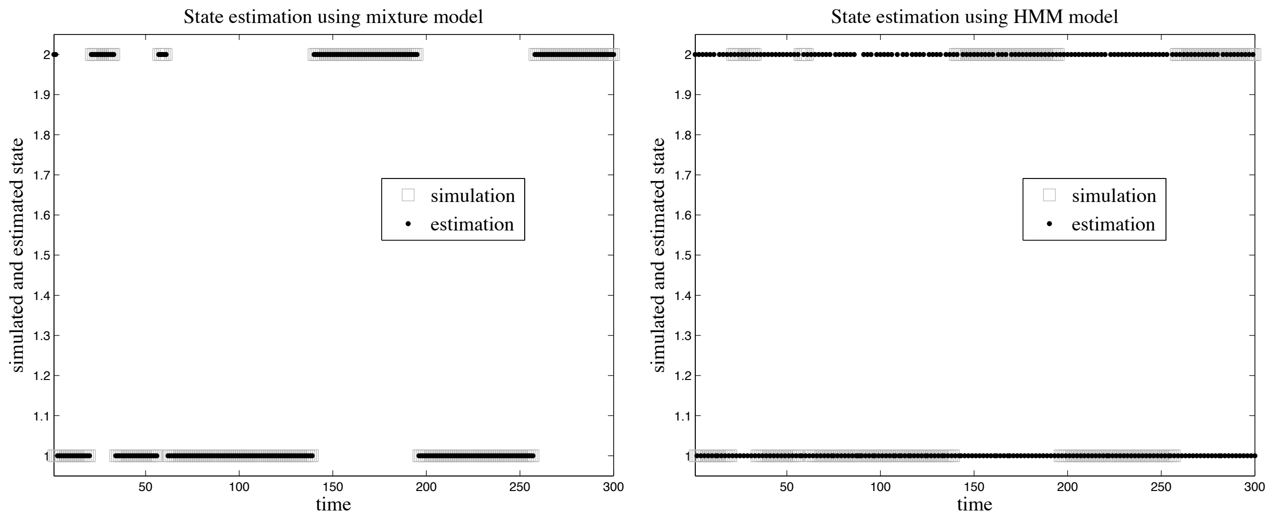

Figure 2. State estimation for 100 possible output values with the mixture estimator (left) and HMM estimator (right). Note that for a random start of the HMM estimator, the stability of the estimation could be violated. The number of correctly point-estimated states is 298 for the mixtures and 152 for HMM.

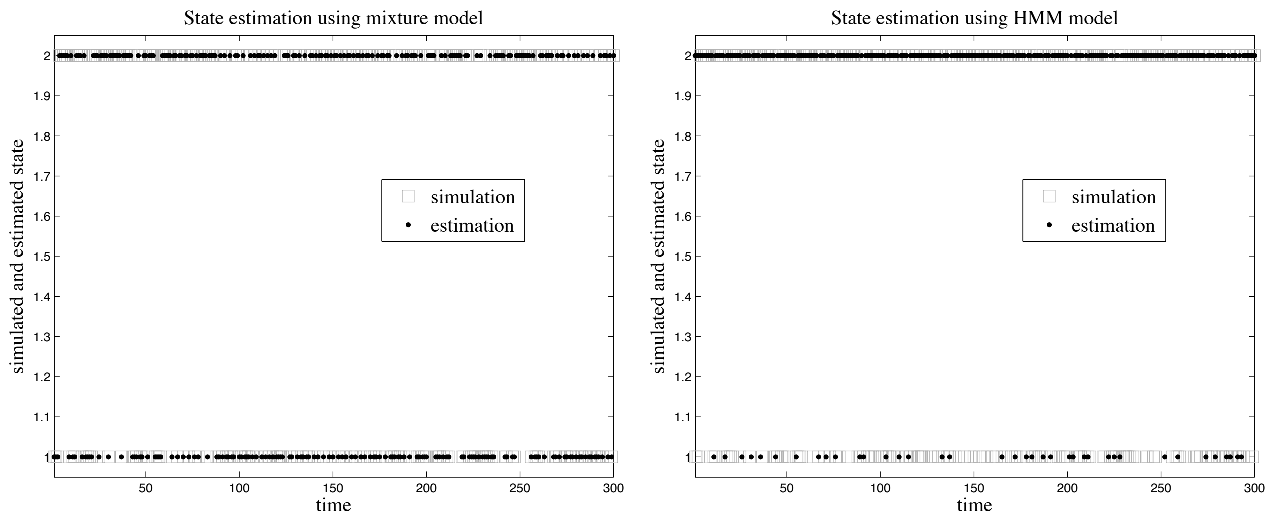

Figure 3. State estimation for uncertain "jumping" transition probabilities for the mixture estimator (left) and HMM (right). The HMM estimator is more sensitive to the uncertain transition model and provides 174 correctly point-estimated states with the transition table $[0.5 \;\; 0.5; \;\; 0.5 \;\; 0.5]$. The mixture estimation proves more stability and has 300 correct estimates.

however it starts randomly successfully and it is stable with great number of the output possible values and with uncertain transition probabilities.

Comparison of computational time proves that the mixture estimator has a shorter (sometimes significantly) elapsed time for running of all the experiments.

It was noticed that offline algorithms proved worse stability in the sense that they either converge with perfect results or they do not converge and the estimation fails. For online algorithms a difference between good and bad results is much more smooth.

REFERENCES

[1] 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, 2005.

[2] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen, "The infinite hidden markov model," in *Advances in Neural Information Processing Systems*, vol. 14, 2002.

[3] S. Lee and R. Lathrop, "Sub-pixel estimation of urban land cover components with linear mixture model analysis and Landsat Thematic Mapper imagery," *International Journal Of Remote Sensing*, vol. 26, no. 22, pp. 4885–4905, NOV 20 2005.

[4] J. Schoenberg and M. Campbell, "Distributed Terrain Estimation Using a Mixture-Model Based Algorithm," in *FUSION: 2009 12th International Conference On Information Fusion, Vols 1-4*, 2009, pp. 960–967, 12th International Conference on Information Fusion, Seattle, WA, JUL 06-09, 2009.

[5] M. Fujisaki and D. Zhang, "Bayesian analysis of compound poisson mixture model and its application to financial data," *Int. Journal of Innovative Computing, Information and Control*, vol. 5, no. 1, pp. 109–118, 2009.

[6] H. C. Cho, K. S. Lee, and M. Fadali, "Online learning algorithm of dynamic bayesian networks for nonstationary signal processing," *Int. Journal of Innovative Computing, Information and Control*, vol. 5, no. 4, pp. 1027–1042, 2009.

[7] 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.

[8] L. Shue, B. Anderson, and S. Dey, "Exponential stability of filters and smoothers for hidden markov models," *IEEE Transactions on Signal Processing*, vol. 46, no. 8, pp. 2180 – 2194, 1998.

[9] V. Šmídl and A. Quinn, "Variational bayesian filtering," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5020–5030, 2008.

[10] I. Nagy, E. Suzdaleva, M. Kárný, and T. Mlynářová, "Bayesian estimation of dynamic finite mixtures," *Int. Journal of Adaptive Control and Signal Processing*, 2011, doi: 10.1002/acs.1239.

[11] P. Nedoma, M. Kárný, T. V. Guy, I. Nagy, and J. Böhm, *Mixtools. (Program)*. Praha: ÚTIA AV ČR, 2003.

[12] "FET, IST12088 – ProDaCTool: Decision Support for Complex Industrial Processes Based on Probabilistic Data Clustering," *http://www.prodactool.rdg.ac.uk/*, 2000 - 2002.