Assessment of various initialization strategies for the Expectation-Maximization algorithm for Hidden Semi-Markov Models with multiple categorical sequences

Brice Olivier [1], Anne Guérin-Dugué [2] & Jean-Baptiste Durand [1]

[1] *Université Grenoble Alpes, Laboratoire Jean Kuntzmann, Inria Mistis, 51 rue des Mathématiques, B.P. 53, F-38041 Grenoble cedex 9, France;* brice.olivier@inria.fr, jean-baptiste.durand@univ-grenoble-alpes.fr

[2] *Université Grenoble Alpes, CNRS, Gipsa-lab, 11 rue des Mathématiques, B.P. 46, F-38040 Grenoble, France;* anne.guerin@gipsa-lab.grenoble-inp.fr

Résumé. Dans cette étude, nous proposons une nouvelle stratégie d'initialisation de l'algorithme d'Espérence-Maximisation afin d'augmenter la vraisemblance en sortie de l'algorithme, appliqué aux modèles semi-Markoviens cachés en fournissant des valeurs initiales des paramètres calculées à partir des observations. Cette stratégie se révèle efficace sur certains jeux de données à séquences multiples et catégorielles.

Mots-clés. Modèles semi-Markoviens cachés, choix des valeurs initiales pour l'algorithme d'Espérance-Maximisation, séquences multiples et à valeurs catégorielles.

Abstract. In this study, we propose a method called *sequence breaking framework* to search high local maximum of the likelihood by providing starting values based on the observations for the Expectation-Maximization algorithm, for Hidden semi-Markov model parameters estimation. The method is shown to be efficient on several datasets with multiple categorical sequences.

Keywords. Hidden semi-Markov models, choosing starting values for the EM algorithm, multiple categorical sequences.

1 Introduction

Since introduced by Ferguson (1980), Hidden semi-Markov models (HSMM) have received a lot of attention in the literature as a natural and more flexible extension of the well-known Hidden Markov model (HMM) by relaxing the geometric state sojourn time assumption. The HSMM framework being very generic, modeling assumptions have been proposed, mainly focused around the dependencies between the state and its sojourn time, see Yu (2010) for a summary, as well as the latent process time censoring Guédon (2003), leading to a wide variety of inference algorithms. Parameter estimation is generally performed using the Expectation-Maximization algorithm, McLachlan and Krishnan (2007). Barbu and Limnios (2009) have proved the asymptotic convergence and normality of the estimators, but did not provide any detail on the convergence speed or on the multiple sequence framework. The Expectation-Maximization algorithm finds a local maximum of the likelihood and is known to be extremely sensitive to starting values.

In this context, we propose a method that we call *sequence breaking framework*, which aims at finding high local maxima of the likelihood by choosing starting values for HSMM's EM, for which the randomness is controlled by the observed sequences in order to restrict the search space. To our knowledge, most of the contributions of this kind have been done around the independent Mixture Models (MMs), see Biernacki et al. (2003) for Gaussian MMs or Juan et al. (2004) for Bernoulli MMs, while for HMM, what seems to work best is a simple jitter of the parameters around their centers as implemented in the python library hmmlearn[1], method which is here compared to ours.

2 Hidden semi-Markov model

An HSMM is composed of two stochastic processes. The former is a finite-state homogeneous semi-Markov chain (SMC) which is latent. It conditions the latter which is the observed process. An SMC is like a Markov Chain except that the within-state sojourn time is not necessarily geometric and can therefore be of any form with support included in $\mathbb{N}$ (categorical or parametric).

In this section, we focus on the simplest assumption on the HSMM also called the Explicit Duration HMM (EDHMM) for which the within-sojourn state duration is only conditioned by the state. Using the formalism proposed by Murphy (2012) which is Dynamic Bayesian Network-oriented (DBN), we describe a SMC by $\{S_t, R_t, F_t\}_{t \in [\![1,T]\!]}, \forall t \in [\![1,T]\!]$ $S_t \in [\![1,M]\!]$, $R_t \in [\![1,D]\!]$ and $F_t \in \{0,1\}$. S_t is the state value while R_t encodes the residual time in the current state S_t at time t. At the beginning of a new state, a new duration is randomly sampled from an arbitrary distribution and then counts down deterministically to 1, and so on. F_t simply acts as a binary switch which is turned on when $R_{t-1} = 1$ and off else. Initialization is performed s.t. $F_0 = 1$ and $R_0 = 1$, and we also have $F_T = 1$ which means that the process starts at time 1 and will end at time T. Follows the parameters associated to the conditional probability distributions associated to a SMC:

$$P(S_1 = k) = \pi_k$$

with $\pi_k \in [\![1,M]\!]$, a vector of size M representing all the initial probabilities, then

$$P(S_t = k | S_{t-1} = k', F_{t-1} = f) = \begin{cases} \mathbb{1}\{k = k'\} & \text{if } f = 0 \\ A_{k'k} & \text{if } f = 1 \end{cases}$$

with $\mathbb{1}$ denoted as the indicator function, and $A \in M \times M$, a matrix representing the transition probabilities from one state to another with zeros on the diagonal to forbid

[1] https://hmmlearn.readthedocs.io

self-transitions. We also have

$$P(R_t = d | R_{t-1} = d', S_t = k, F_{t-1} = 1) = p_k(d)$$

with $p_k(d)$ being an arbitrary, upper-bounded by D, probability distribution representing the sojourn distributions for each state k while entering a new state at time t and then sampling a new value $d \geq 1$ for R_t. Finally,

$$P(R_t = d | R_{t-1} = d', S_t = k, F_{t-1} = 0) = \begin{cases} \mathbb{1}\{d = d' - 1\} & \text{if } d > 1 \\ \text{undefined} & \text{if } d = 1 \end{cases}$$

and

$$P(F_t = f | R_t = d) = \begin{cases} \mathbb{1}\{d = 1\} & \text{if } f = 1 \\ \mathbb{1}\{d > 1\} & \text{if } f = 0 \end{cases}$$

define the countdown process, i.e. the residual time in the current state.

To sum up, the process can be described as the transition from a latent state to another at time t triggering the following changes: the finishing node switches on $F_{j-1} = 1$, requiring a transition to a new state, $S_t = k$, from the previous one, $S_{t-1} = k'$, with $k' \neq k$, along with its new duration sampled s.t. $R_t \sim p_k \geq 1$.

The discrete observed process $\{X_t\}_{t \in [\![1,T]\!]}$ takes values in $\{v_1, ..., v_G\}$ and is conditioned by $\{S_t\}_{t \in [\![1,T]\!]}$ s.t.

$$P(O_t = v_g | S_t = k) = b_k(v_g)$$

where $b_k(v_g)$ represent either a categorical distribution, i.e. a matrix of size $M \times G$, or M parametric distributions. Finally, we denote θ the set of all model parameters. Note that here, we treat the discrete observation case but it is generalizable when observations are continuous.

3 Experimental strategy

Multiple sequence framework. So far, we have written down the EDHMM considering only one sequence of observation $\{X_t\}_{t \in [\![1,T]\!]}$ for notation convenience. We now consider that we have multiple observed sequences $\mathbf{X} = \{\{X_t^{(1)}\}_{t \in [\![1,T_1]\!]}, ..., \{X_t^{(K)}\}_{t \in [\![1,T_K]\!]}\}$.

Proposition 1. *For an observed semi-Markov chain $\{S_t, R_t, F_t\}_{t \in [\![1,T]\!]}$ with a corresponding output process $\{X_t\}_{t \in [\![1,T]\!]}$, the MLE of each set of parameters is given by the conditional empirical frequencies.*

Proof. Given that a EDHMM can be expressed as a DBN, see section 2, and since the likelihood of a DBN can be expressed as a global decomposition of the local-likelihood of each node given its parents, see Ghahramani (2001), then the parameters of the EDHMM can be locally optimized by MLE using the conditional empirical frequencies. □

Choosing starting values with the sequence breaking framework. The proposed strategy relies on two intertwined algorithms:

- Algorithm 1 **HighLikelihodSearch**: describes the global framework, it randomly chooses α observed sequences $\mathbf{X}^{(Q_\alpha)}$ from $\mathbf{X}$, generates the corresponding state sequences $\mathbf{S}^{(Q_\alpha)}$ using **SequenceBreaking**, computes the parameters θ^{init} by MLE using Proposition 1 and injects it as a starting value for the EM algorithm which finds the a local maximum of the likelihood for all data $\mathbf{X}$. The goal of sampling sequences randomly from $\mathbf{X}$ is to generate starting values related to the observation process while keeping only a subset to maintain the randomness of the starting values. Note that **Sample**(.) is a function which samples uniformly on the given set.

Algorithm 1: HighLikelihoodSearch: High local maximum of the likelihood search by sequence breaking

```
Input: $\alpha \in [\![1, K]\!]$ the number of sequence to sample,
       $N$, the number of initialization
$\hat{\theta} \leftarrow \emptyset$;
for $n \leftarrow 0$ to $N$ do
    Sample $\mathbf{X}^{(Q_\alpha)} \subset \mathbf{X}$ observed sequences s.t. $Q_\alpha \subset [\![1, K]\!]$;
    $\mathbf{S}^{(Q_\alpha)} \leftarrow$ SequenceBreaking($\mathbf{X}^{(Q_\alpha)}$);
    $\theta^{init} \leftarrow \arg\max_\theta \mathcal{L}(\theta; \{\mathbf{X}^{(Q_\alpha)}, \mathbf{S}^{(Q_\alpha)}\})$; # MLE provided by Proposition 1
    $\hat{\theta} \leftarrow \hat{\theta} \cup$ ExpectationMaximization($\theta^{init}, \mathbf{X}$);
end
$\hat{\theta}^* \leftarrow \arg\max_{\hat{\theta}} \mathcal{L}(\hat{\theta}; \mathbf{X})$
Output: $\hat{\theta}^*$, a high local maximum of the log-likelihood.
```

- Algorithm 2 **SequenceBreaking**: generates randomly a hidden state sequence. Given each observed sequence $\mathbf{X}^{(q)} \in \mathbf{X}^{(Q_\alpha)}$ with its length, it randomly chooses a number of transition J as well as transition instants I which "breaks" the sequences into pieces and then affects a state randomly to each piece of sequence with the constraints that two consecutive states should be different due to the EDHMM assumption on the transition matrix.

Choosing starting values with a jittered-center parameters. We compare the proposed methodology with the default strategy used for HMM which consists in selecting slightly perturbed parameters around their centers, i.e. s.t. each event has equal probability, similarly to Juan et al. (2004).

Example 1. *With $M = 2$, we wish to randomly sample π s.t. $\pi_1 =$ **Sample**$([\epsilon, 1-\epsilon])$ and $\pi_2 = 1 - \pi_1$, with $\epsilon \in]0, 0.5]$.*

Algorithm 2: SequenceBreaking

```
Input: $\mathbf{X}^{(Q_\alpha)}$, an observed sequence subset of size $\alpha$
for $q \in Q^{(\alpha)}$ do
    $J \leftarrow$ Sample($[\![1, T_q - 1]\!]$); # number of transitions
    $I \leftarrow \emptyset$;
    for $j \leftarrow 0$ to $J$ do
        $i \leftarrow$ Sample($[\![1, T_q]\!]$) s.t. $i \notin I$; # transition instant
        $I \leftarrow I \cup i$
        $\{\mathbf{S}_t^{(q)}\}_{t \in [\![I_{j-1}, i]\!]} \leftarrow$ Sample($[\![1, M]\!]$) s.t. $S^{(q)}_{I_{j-1}} \neq S^{(q)}_{I_{j-1}-1}$; # choose state
    end
    $\{\mathbf{S}_t^{(q)}\}_{t \in [\![i, T_q]\!]} \leftarrow$ Sample($[\![1, M]\!]$) s.t. $S^{(q)}_{i} \neq S^{(q)}_{I_{j-1}-1}$; # choose final state
end
Output: $\mathbf{S}^{(Q_\alpha)}$, a randomly sampled state sequences
```

Example 1 is a specific instance with a Bernoulli distribution. In order to generalize with $M \geq 2$, i.e. to the Multinomial distribution, one solution is to sample from its conjugate, the Dirichlet distribution. Therefore, we apply a Dirichlet sample for each set of parameters except the sojourn distribution which we initialize with a Geometric distribution of parameter $p = 0.1$.

4 Results

The experiments consists in comparing numerically both methods in finding the highest likelihood, but also to compare the ability to find a maximum as well as the convergence speed of EM, for three different datasets, artificial, artificial with noise and real.

Datasets. The first dataset $\mathcal{D}^{(a)}$ is artificial, composed of 100 sequences of length 100 each, with $K = 5$ clusters and $G = 5$ observed parameters, and parameters $\pi = (0.2, 0.2, 0.2, 0.2, 0.2)$, $p(.) = (\mathcal{G}(0.2), \mathcal{G}(0.05), \mathcal{NB}(8, 0.5), \mathcal{P}(4), \mathcal{NB}(5, 0.1))$ where $\mathcal{G}$ stands for the Geometric distribution, $\mathcal{NB}$ Negative Binomial and $\mathcal{P}$ Poisson,

$$A = \begin{pmatrix} 0 & 0.5 & 0.3 & 0.1 & 0.1 \\ 0.5 & 0 & 0.1 & 0.3 & 0.1 \\ 0.25 & 0.25 & 0 & 0.25 & 0.25 \\ 0.2 & 0.2 & 0.2 & 0 & 0.4 \\ 0.1 & 0.1 & 0.4 & 0.4 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0.1 & 0.2 & 0.4 & 0.2 & 0.1 \\ 0.25 & 0.2 & 0.1 & 0.2 & 0.25 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.35 & 0.3 & 0.2 & 0.1 & 0.05 \\ 0.05 & 0.1 & 0.2 & 0.3 & 0.35 \end{pmatrix}$$

The second dataset $\mathcal{D}^{(an)}$ is generated from the first one by replacing 20% of the observations at random. The third dataset $\mathcal{D}^{(r)}$ consists of the bounded number of words skipped during an ocular saccade by different subjects for a reading tasks for which $G = 5$ and

	$\mathcal{D}^{(a)}$	$\mathcal{D}^{(an)}$	$\mathcal{D}^{(r)}$
Sequence Breaking	$\mathbf{-15448 \pm 2.3}$	$\mathbf{-15785 \pm 4.5}$	-50567 ± 236
Jittered-centers	-15452 ± 2.9	-15782 ± 2.6	-50592 ± 274

Table 1: Means and standard deviations of maximum likelihood. Significant ($<$5%) mean differences are boldfaced.

we assume $M = 5$. There are 2390 sequences of different lengths, on average 17 with a standard deviation of 8, see Durand et al. (2016) for more information about this dataset.

Global results. For a global analysis, we compute the mean and standard deviation of all the optimal likelihood for each method and each dataset over 100 initializations and 1000 iterations of EM each. Results are presented in table 1.

Local results. For a local analysis, we split the 100 initializations into 10 blocks of 10 and compute the max per block. For $\mathcal{D}^{(a)}$, sequence breaking performed better than the jittered-centers 9 times out of 10. For $\mathcal{D}^{(an)}$, $\mathbf{9/10}$, and for $\mathcal{D}^{(r)}$, 6/10.

Convergence speed. For $\mathcal{D}^{(a)}$, on average, it took the sequence breaking initialization **133** less iterations to convergence than the jittered-centers. For $\mathcal{D}^{(an)}$, it took **305** less, and for $\mathcal{D}^{(r)}$, 44 less iterations on average.

5 Discussion and perspectives

We proposed a new strategy to search for improved maximum likelihood of HSMM with multiple sequences categorical data which is a significant improvement considering that the current Python implementation of HSMM under the virtual plants library *sequence analysis*[2], as well as the R package mhsmm by O'Connell et al. (2011) do not provide a random start option.

We can address a few remarks on this work. First, α, the number of sequences to sample can be seen as a control of the randomness of the initial parameters since the higher it is, the more the observed process will be close to the real one. Secondly, we would like to note the similarity of the procedure with the stick-breaking process view of the Dirichlet process. Thirdly, as shown by Meilă and Heckerman (2001), initialization techniques are data dependent and we still need to inspect more datasets but also more techniques such as Stochastic EM or Classification EM for initialization, to find out the relevance of the sequence breaking framework. Another interesting perspective is to propose a heuristic for early detection of bad candidates in order to save processing time. Finally, we plan on testing the strategy on more datasets and on continuous data as well.

[2] https://github.com/openalea/StructureAnalysis

References

Barbu, V. S. and Limnios, N. (2009). *Semi-Markov chains and hidden semi-Markov models toward applications: their use in reliability and DNA analysis*, volume 191. Springer Science & Business Media.

Biernacki, C., Celeux, G., and Govaert, G. (2003). Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics & Data Analysis*, 41(3-4):561–575.

Durand, J.-B., Guérin-Dugué, A., and Achard, S. (2016). Analyse de séquences oculométriques et d'électroencéphalogrammes par modèles markoviens cachés. In *48èmes Journées de Statistique.*

Ferguson, J. (1980). pp. 143–179, variable duration models for speech. In *Proc. of the Symposium on the applications of hidden Markov models to text and speech, JD Ferguson, Ed. Princeton: IDA-CRD.*

Ghahramani, Z. (2001). An introduction to hidden markov models and bayesian networks. *International journal of pattern recognition and artificial intelligence*, 15(01):9–42.

Guédon, Y. (2003). Estimating hidden semi-markov chains from discrete sequences. *Journal of Computational and Graphical Statistics*, 12(3):604–639.

Juan, A., García-Hernández, J., and Vidal, E. (2004). Em initialisation for bernoulli mixture learning. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 635–643. Springer.

McLachlan, G. and Krishnan, T. (2007). *The EM algorithm and extensions*, volume 382. John Wiley & Sons.

Meilă, M. and Heckerman, D. (2001). An experimental comparison of model-based clustering methods. *Machine learning*, 42(1-2):9–29.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective.* MIT press.

O'Connell, J., Højsgaard, S., et al. (2011). Hidden semi markov models for multiple observation sequences: The mhsmm package for r. *Journal of Statistical Software*, 39(4):1–22.

Yu, S.-Z. (2010). Hidden semi-markov models. *Artificial intelligence*, 174(2):215–243.