Parameter Estimation and State Inference in Hidden Drifting Markov Models

Aguêmon Léon TEHUNGUE
Phd student
Supervisors: Vlad Stefan BARBU, Nicolas VERGNE

University of Rouen Normandy
Laboratory LMRS

ANR HSMM-INCA
Paris, July 1 – 4, 2025

Outline

Outline

Introduction – Why Non-Homogeneous Chains? Motivations and Applications for Non-Homogeneity

- Homogeneity of Markov models
  - Markov chains model dynamic random phenomena.
  - The homogeneity assumption is often unrealistic.
  - Non-homogeneous chains allow modeling time-varying transitions and/or emissions.
- Many phenomena exhibit time variability.
- More realistic for:
  - Biological data
  - Speech recognition: variations in rhythm/intonation
  - Writing recognition: context-sensitive lexical transitions
  - Finance: markets with evolving dynamics
  - Survival and reliability analysis
- Better tracks natural dynamics.
- Our context : Study of Hidden Markov Models with controlled non-homogeneity, usually Hidden Drifting Markov Models.

Outline

Drifting Markov Models

Definition (Linear Drifting Markov Models of Order 1)

A sequence $X_0, X_1, \ldots, X_n$ with state space $E = \{1, 2, \ldots, s\}$ is said to be a *linear drifting Markov chain of order 1* of length n between transition matrices $\mathbf{A}_0$ and $\mathbf{A}_1$ if, for $t = 0, \ldots, n$, the law of X_t satisfies:

$$\mathbb{P}(X_t = v \mid X_{t-1} = u, X_{t-2}, \ldots) = \mathbb{P}(X_t = v \mid X_{t-1} = u) = A_{\frac{t}{n}}(u,v), \quad u,v \in E$$

and initial distribution

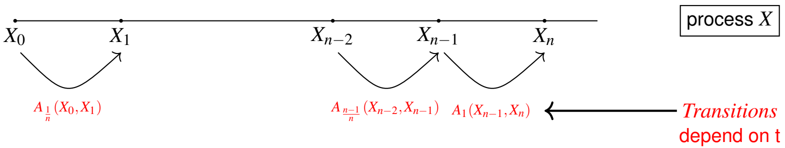
$$a = \left(a(u) = P(X_0 = u), \quad 1 \leq u \leq s\right),$$

where

$$A_{\frac{t}{n}}(u,v) = \left(1 - \frac{t}{n}\right) A_0(u,v) + \frac{t}{n} A_1(u,v), \quad u,v \in E.$$

The full set of parameters is therefore: $\theta = (a, A_0, A_1)$

$\Rightarrow$ Polynomial drifting and higher-order Markov chains are natural extensions.



Hidden Drifting Markov Models

Definition (Linear Hidden Drifting Markov Model)

A linear Hidden Drifting Markov Model (HDMM) of type $DM1$-$M0$ is characterized by two nested processes:
the hidden process $S = (S_t)_{t=0}^{n}$, taking values in the finite set $\mathcal{S} = \{1,\ldots,q\}$, and the observed process $Y = (Y_t)_{t=0}^{n}$, taking values in the set $\mathcal{Y} = \{1,\ldots,r\}$;

- The hidden process S is a linear Drifting Markov order 1 chain with transition matrix

  $$A_{\frac{t}{n}} = \left(A_{\frac{t}{n}}(u,v) = \mathbb{P}(S_t = v \mid S_{t-1} = u), \quad 1 \leq u,v \leq q\right)$$

  and initial distribution

  $$a = (a(u) = P(S_0 = u), \quad 1 \leq u \leq q).$$

  i.e.

  $$A_{\frac{t}{n}} = (1 - \frac{t}{n})A_0 + \frac{t}{n}A_1$$

  where

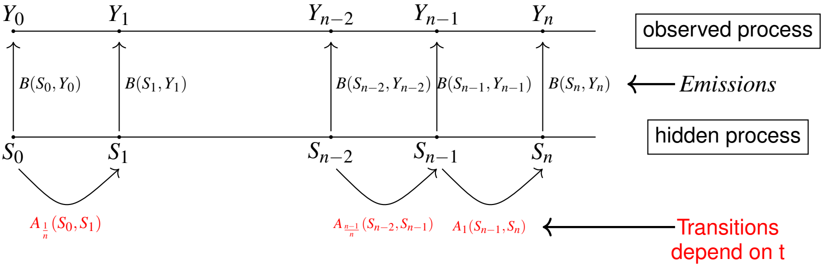
  $$A_0 = (A_0(u,v) \quad 1 \leq u,v \leq q) \quad \text{and} \quad A_1 = (A_1(u,v) \quad 1 \leq u,v \leq q)$$

- The emission law of each observation Y_t along the sequence depends on the associated hidden state S_t, and the observations Y_t are independently given the hidden states S_t, with emission probabilities:

  $$B = (B(u,i) = P(Y_t = i \mid S_t = u), \quad 1 \leq u \leq q,\ 1 \leq i \leq r).$$

The full set of parameters is therefore: $\theta = (a, A_0, A_1, B)$.

Hidden Drifting Markov Models



	$(S_t)_t$ MC order 1 (M_1-)	$(S_t)_t$ DMC order 1 (DM_1-)
$(Y_t \mid S_t)_t$ independent $(-M0)$	$M1-M0$	$DM1-M0$
$(Y_t \mid S_t)_t$ independent and derived $(-DM0)$	$M1-DM0$	$DM1-DM0$

Table: HDMMs with conditionally independent observations

	$(S_t)_t$ MC order 1 (M_1-)	$(S_t)_t$ DMC order 1 (DM_1-)
$(Y_t \mid S_t)_t$ MC of order m $(-Mm)$	$M1-Mm$	$DM1-Mm$
$(Y_t \mid S_t)_t$ DMC of order m $(-DMm)$	$M1-DMm$	$DM1-DMm$

Table: HDMMs with conditionally Markov observations

Outline

EM Algorithm for $DM1 - M0$ linear model

- function $Q(\theta \mid \theta^{(p)})$

$$Q(\theta \mid \theta^{(p)}) = \sum_{v=1}^{q} P(s_0 = v \mid y, \theta^{(p)}) \log a(v) + \sum_{t=1}^{n} \sum_{u=1}^{q} \sum_{v=1}^{q} P(s_{t-1} = u, s_t = v \mid y, \theta^{(p)})$$

$$\times \log \left((1 - \frac{t}{n}) A_0(u,v) + \frac{t}{n} A_1(u,v) \right) + \sum_{t=0}^{n} \sum_{v=1}^{q} \sum_{j=1}^{r} P(s_t = v \mid y, \theta^{(p)}) 1_{\{y_t = j\}} \log B(v,j)$$

- E-step of the EM algorithm proceeds successfully.
- M-step, which consists of maximizing the quantity $Q(\theta \mid \theta^{(p)})$ with respect to θ, requires maximizing in $A_0(u,v)$ and $A_1(u,v)$:

$$\sum_{t=1}^{n} P(s_{t-1} = u, s_t = v \mid y, \theta^{(p)}) \times \log \left((1 - \frac{t}{n}) A_0(u,v) + \frac{t}{n} A_1(u,v) \right)$$

We cannot solve explicitly solution for this problem

We introduce another algorithm that we call **Em**, as it incorporates the E-step of the EM algorithm, but instead of maximizing the function $Q(\theta \mid \theta^{(p)})$, it minimizes an alternative function.

- Distance:

$$d(X,Y) = \sqrt{\mathbb{E}[(X-Y)^2 \mid Z]}.$$

- Using this distance while assuming knowing the complete data (y,s), the quadratic risk of the complete data for the models $DM1-M0$ is defined by:

$$SP_1(y,s \mid \theta) = \sum_{t=1}^{n}\sum_{u=1}^{q}\left[1_{\{s_{t-1}=u\}}\sum_{v=1}^{n}\left(a_{\frac{t}{n}}(u,v) - 1_{\{s_t=v\}}\right)^2\right]$$

$$+\sum_{t=0}^{n}\sum_{v=1}^{q}\left[1_{\{s_t=v\}}\sum_{i=1}^{r}\left(b(v,i) - 1_{\{y_t=i\}}\right)^2\right]$$

If we assume to know only the observation y, the quadratic risk of incomplete data is defined as:

$$SPI(y \mid \theta) = \mathbb{E}[SP_1(y, S \mid \theta)] \tag{1}$$

where :

- (y, S) represents the complete data (observed + missing),
- y represents observed data,
- $SP_1(y, S \mid \theta)$ is the quadratic risk of the complete data of (y, S) based on parameters θ.

Quadratic Risk of Incomplete Data

The point-by-point method of estimation, which consists in minimizing the function $SPI(y \mid \theta)$ is not easy because the hidden regime S is unknown. indeed $SPI(y \mid \theta)$ is not computable knowing θ:

$$SPI(y \mid \theta) = \mathbb{E}[SP_1(y, S \mid \theta)]$$

For the models $DM1 - M0$ linear:

$$SPI(y \mid \theta) = \sum_{t=1}^{n} \sum_{u=1}^{q} \sum_{v=1}^{q} \Big\{ P(S_{t-1} = u \mid y)((1 - \frac{t}{n})A_0(u,v) + \frac{t}{n}A_1(u,v))^2$$

$$-2((1 - \frac{t}{n})A_0(u,v) + \frac{t}{n}A_1(u,v))P(S_{t-1} = u, S_t = v \mid y) \Big\}$$

$$+P(S_{t-1} = u, S_t = v \mid y) + \sum_{t=0}^{n} \sum_{v=1}^{q} \left[P(S_t = v \mid y) \sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_t = i\}} \right)^2 \right]$$

Thus we use the Em algorithm in order to have an ideal candidate of this minimization problem.

Core Idea of the Em Algorithm

Let θ be the parameter defining the model, and $\theta^{(p)}$ an iterative estimate of the parameter.
Instead of minimizing the incomplete-data quadratic risk $SPI(y \mid \theta)$ directly, the Em algorithm minimizes the expected complete-data quadratic risk:

$$R(\theta|\theta^{(p)}) = \mathbb{E}[SP_1(y, S \mid \theta) \mid y, \theta^{(p)}].$$

At iteration p of the algorithm, starting from an initial value $\theta^{(1)}$ and given the current estimate $\theta^{(p)}$, proceed as follows:

- **E-step (Expectation):** Compute $R(\theta \mid \theta^{(p)})$
  - Using a **Forward** step
  - Followed by a **Backward** step
- **m-step (minimization):** Set

$$\theta^{(p+1)} = \arg\min_{\theta} R(\theta|\theta^{(p)}).$$

Consequently, the Em algorithm decreases the quadratic risk of the incomplete data at each iteration:

$$SPI(y \mid \theta^{(p+1)}) \leq SPI(y \mid \theta^{(p)}).$$

The $E-$Step, which is to calculate the function

$$R(\theta|\theta^{(p)}) = \mathbb{E}[SP_1(y, S \mid \theta) \mid y, \theta^{(p)}] \tag{2}$$

$$\begin{aligned}
R(\theta|\theta^{(p)}) = \sum_{t=1}^{n}\sum_{u=1}^{q}\sum_{v=1}^{q} \Big\{ & P(S_{t-1} = u \mid y, \theta^{(p)})((1-\frac{t}{n})A_0(u,v) + \frac{t}{n}A_1(u,v))^2 \\
& -2((1-\frac{t}{n})A_0(u,v) + \frac{t}{n}A_1(u,v))P(S_{t-1} = u, S_t = v \mid y, \theta^{(p)}) \Big\} \\
& + P(S_{t-1} = u, S_t = v \mid y, \theta^{(p)}) \\
& + \sum_{t=0}^{n}\sum_{v=1}^{q}\left[P(S_t = v \mid y, \theta^{(p)}) \sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_t = i\}} \right)^2 \right]
\end{aligned}$$

$$y_{t_1}^{t_2} = y_{t_1}, y_{t_1+1}, \ldots, y_{t_2-1}, y_{t_2} \quad \text{or} \quad s_{t_1}^{t_2} = s_{t_1}, s_{t_1+1}, \ldots, s_{t_2-1}, s_{t_2}$$

refers to the following $t_2 - t_1 + 1$ consecutive bases or regimes.

Predictive Equation

$$P(s_t = v \mid y_0^{t-1}, \theta^{(p)}) = \sum_{u=1}^{q} A_{\frac{t}{n}}^{(p)}(u,v) P(s_{t-1} = u \mid y_0^{t-1}, \theta^{(p)}). \tag{3}$$

Filtering

$$P(s_t = v \mid y_0^{t}, \theta^{(p)}) = \frac{B^{(p)}(v, y_t) P(s_t = v \mid y_0^{t-1}, \theta^{(p)})}{\sum_{u=1}^{q} B^{(p)}(u, y_t) P(s_t = u \mid y_0^{t-1}, \theta^{(p)})}. \tag{4}$$

Smoothing Equation

$$P(s_{t-1} = u, s_t = v \mid y, \theta^{(p)}) = P(s_{t-1} = u \mid s_t = v, y, \theta^{(p)}) P(s_t = v \mid y, \theta^{(p)})$$

$$= \frac{A_{\frac{t}{n}}^{(p)}(u,v) P(s_{t-1} = u \mid y_0^{t-1}, \theta^{(p)}) P(s_t = v \mid y, \theta^{(p)})}{P(s_t = v \mid y_0^{t-1}, \theta^{(p)})}. \tag{5}$$

The Em Algorithm of the DM1-M0 Model
$E-$Step: forward-back

Prediction

Y_0 —— Y_{t-1}

S_t

$P(s_t = v \mid y_0^{t-1}, \theta^{(p)})$

Filtering

Y_0 —— Y_t

S_t

$P(s_t = v \mid y_0^{t}, \theta^{(p)})$

total probability

Y_0 —— Y_n

S_t

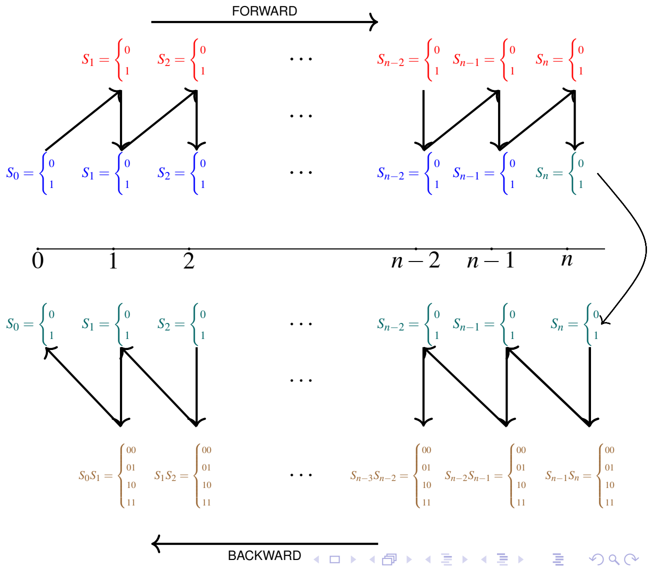
$P(s_t = v \mid y, \theta^{(p)})$

Smoothing

Y_0 —— Y_n

S_{t-1} —— S_t

$P(s_{t-1}s_t = uv \mid y, \theta^{(p)})$



The Em algorithm of the DM1-M0 model
$m-$Step

$$P_{uv}^{(p)}(t) = P(s_{t-1} = u, s_t = v \mid y, \theta^{(p)}), \quad \text{and} \quad P_u^{(p)}(t) = P(s_t = u \mid y, \theta^{(p)})$$

$$A_0^{(p+1)}(u,v) = \frac{\left(\begin{array}{c} \left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(\frac{t}{n}\right)^2\right)\left(\sum_{t=1}^n P_{uv}^{(p)}(t)\left(1-\frac{t}{n}\right)\right) \\ -\left(\sum_{t=1}^n P_u^{(p)}(t)\left(1-\frac{t}{n}\right)\frac{t}{n}\right)\left(\sum_{t=1}^n P_{uv}^{(p)}(t)\frac{t}{n}\right) \end{array} \right)}{\left(\begin{array}{c} \left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(\frac{t}{n}\right)^2\right)\left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)^2\right) \\ -\left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)\frac{t}{n}\right)^2 \end{array} \right)}$$

$$A_1^{(p+1)}(u,v) = \frac{\left(\begin{array}{c} \left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)^2\right)\left(\sum_{t=1}^n P_{uv}^{(p)}(t)\frac{t}{n}\right) \\ -\left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)\frac{t}{n}\right)\left(\sum_{t=1}^n P_{uv}^{(p)}(t)\left(1-\frac{t}{n}\right)\right) \end{array} \right)}{\left(\begin{array}{c} \left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(\frac{t}{n}\right)^2\right)\left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)^2\right) \\ -\left(\sum_{t=1}^n P_u^{(p)}(t-1)\left(1-\frac{t}{n}\right)\frac{t}{n}\right)^2 \end{array} \right)}$$

$$a^{(p+1)}(v) = \frac{\sum_{t=0}^n P(s_t = v \mid y, \theta^{(p)})}{n+1};$$

$$B^{(p+1)}(v,j) = \frac{\sum_{t=0}^n 1_{\{y_t=j\}} P(s_t = v \mid y, \theta^{(p)})}{\sum_{t=0}^n P(s_t = v \mid y, \theta^{(p)})};$$

Numerical Results: DM1-M0 model
Estimation with the initial parameter close to the true parameter

Estimation with the initial parameter close to the true parameter

1) True	2) init Very Close	3) init Slightly Close	4) estim with Very Close	5) estim with Slightly Close
pi: 0.5, 0.5	pi: 0.48, 0.52	pi: 0.6, 0.4	pi: 0.503, 0.497	pi: 0.521, 0.479
A0:	A0:	A0:	A0:	A0:
0.6, 0.4	0.61, 0.39	0.7, 0.3	0.605, 0.395	0.648, 0.352
0.5, 0.5	0.51, 0.49	0.55, 0.45	0.502, 0.498	0.503, 0.497
A1:	A1:	A1:	A1:	A1:
0.5, 0.5	0.49, 0.51	0.45, 0.55	0.5, 0.5	0.527, 0.473
0.4, 0.6	0.41, 0.59	0.35, 0.65	0.403, 0.597	0.393, 0.607
B:	B:	B:	B:	B:
0.4, 0.1, 0.1, 0.4	0.39, 0.11, 0.09, 0.41	0.35, 0.15, 0.12, 0.38	0.399, 0.101, 0.101, 0.399	0.36, 0.142, 0.124, 0.374
0.05, 0.5, 0.4, 0.05	0.06, 0.49, 0.41, 0.04	0.1, 0.4, 0.45, 0.05	0.049, 0.501, 0.4, 0.049	0.079, 0.471, 0.386, 0.064
Distance to true: 0	Distance to true: 0.0966	Distance to true: 0.5438	Distance to true: 0.0185	Distance to true: 0.2184

Numerical Results: DM1-M0 model
Estimation with the initial parameter far to the true parameter

Estimation with the initial parameter far to the true parameter

1) True	2) init Far	3) init Very Far	4) estim With Far	5) estim With Very Far
pi: 0.5, 0.5	pi: 0.3, 0.7	pi: 0.9, 0.1	pi: 0.496, 0.504	pi: 0.5, 0.5
A0:	A0:	A0:	A0:	A0:
0.6, 0.4	0.3, 0.7	0.9, 0.1	0.403, 0.597	0.9, 0.1
0.5, 0.5	0.2, 0.8	0.1, 0.9	0.199, 0.801	0.1, 0.9
A1:	A1:	A1:	A1:	A1:
0.5, 0.5	0.7, 0.3	0.1, 0.9	0.708, 0.292	0.1, 0.9
0.4, 0.6	0.8, 0.2	0.9, 0.1	0.704, 0.296	0.9, 0.1
B:	B:	B:	B:	B:
0.4, 0.1, 0.1, 0.4	0.2, 0.3, 0.2, 0.3	0.05, 0.45, 0.45, 0.05	0.18, 0.356, 0.287, 0.177	0.225, 0.3, 0.25, 0.225
0.05, 0.5, 0.4, 0.05	0.25, 0.25, 0.3, 0.2	0.05, 0.45, 0.45, 0.05	0.269, 0.244, 0.213, 0.273	0.225, 0.3, 0.25, 0.225
Distance to true: 0	Distance to true: 2.0001	Distance to true: 2.8819	Distance to true: 1.6659	Distance to true: 2.1101

Outline

Viterbi Algorithm: Steps for the $DM1 - M0$ Model

1. Initialization ($t = 0$):

$$\nu_0(i) = a(i) \cdot B(i, y_0), \quad \text{for } i \in E$$

2. Recursion (for $t = 1$ to n and for $i \in E$):

$$\nu_t(i) = B(i, y_t) \cdot \max_{k \in E} (\nu_{t-1}(k) \cdot A_{\frac{t}{n}}(k, i))$$

$$\tau_t(i) = \arg\max_{k \in E} (\nu_{t-1}(k) \cdot A_{\frac{t}{n}}(k, i))$$

3. Termination:

$$P(y, s^*) = \max_{k \in E} \nu_n(k)$$

$$s_n^* = \arg\max_{k \in E} \nu_n(k)$$

4. Backtracking (for $t = n, \ldots, 1$):

$$s_{t-1}^* = \tau_t(s_t^*)$$

As in the Viterbi algorithm, we will describe another algorithm. Instead of seeking the most probable path, this algorithm looks for the least risky path, given the model parameters.

Recall that the quadratic prediction risk of the complete data, SP_1, is typically defined in the case of the $DM1-M0$ models by:

$$SP_1(y,s \mid \theta) = \sum_{t=1}^{n}\sum_{u=1}^{q}\left[1_{\{s_{t-1}=u\}}\sum_{v=1}^{n}\left(A_{\frac{t}{n}}(u,v)-1_{\{s_t=v\}}\right)^2\right]$$
$$+\sum_{t=0}^{n}\sum_{v=1}^{q}\left[1_{\{s_t=v\}}\sum_{i=1}^{r}\left(B(v,i)-1_{\{y_t=i\}}\right)^2\right]$$

- **Decision Strategy**
  In general, among all possible segmentations s (the paths s) of the observed sequence y, the decision strategy will be to choose the least risky one, and thus will follow the procedure outlined below:

  1. Evaluate the risk $SP_1(y,s)$ for each solution s.
  2. Select as the solution s_* the one for which the risk is minimal:

$$s_* = \arg\min_{s \in \mathcal{S}} SP_1(y,s).$$

- **Recurrence formula**

$$R_t(v) = SP_1(\text{the optimal solution ending at site } t \text{ in state } v)$$

$$R_t(v) = \min_{u \in \{1,\ldots,q\}} \left[R_{t-1}(u) + \left(\sum_{w=1}^{q} \left(A_{\frac{t}{n}}(u,w) \right)^2 - 2A_{\frac{t}{n}}(u,v) + 1 \right) + \sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_t=i\}} \right)^2 \right]$$

Indeed,

$$SP_1(s_1,\ldots,s_{t-1},s_t,y_1,\ldots,y_{t-1},y_t) = SP_1(s_1,\ldots,s_{t-1},y_1,\ldots,y_{t-1})$$
$$+ \sum_{u=1}^{q} \left[1_{\{s_{t-1}=u\}} \sum_{w=1}^{q} \left(A_{\frac{t}{n}}(u,w) - 1_{\{s_t=w\}} \right)^2 \right] + \sum_{w=1}^{q} \left[1_{\{s_t=w\}} \sum_{i=1}^{r} \left(B(w,i) - 1_{\{y_t=i\}} \right)^2 \right]$$

Modified Viterbi Algorithm: Steps for the $DM1 - M0$ Model

The steps of the algorithm in the case of the $DM1 - M0$ model are as follows:

1. Initialization: $(t = 0)$:

$$R_0(v) = \left[\sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_0 = i\}} \right)^2 \right].$$

2. Recursion: (for $t = 1$ à n and for $v = 1, \ldots, q$) :

$$R_t(v) = \min_{u \in \{1,\ldots,q\}} \left[R_{t-1}(u) + \left(\sum_{w=1}^{q} \left(A_{\frac{t}{n}}(u,w) \right)^2 - 2A_{\frac{t}{n}}(u,v) + 1 \right) + \sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_t = i\}} \right)^2 \right]$$

$$\tau_t(v) = \arg\min_{u \in \{1,\ldots,q\}} \left[R_{t-1}(u) + \left(\sum_{w=1}^{q} \left(A_{\frac{t}{n}}(u,w) \right)^2 - 2A_{\frac{t}{n}}(u,v) + 1 \right) + \sum_{i=1}^{r} \left(B(v,i) - 1_{\{y_t = i\}} \right)^2 \right]$$

3. Termination:

$$SP_1(y, s_*) = \min_{k \in \{1,\ldots,q\}} R_n(k)$$

$$s_{*_n} = \arg\min_{k \in \{1,\ldots,q\}} R_n(k)$$

4. Backtracking: (for $t = n, \ldots, 1$)

$$s_{*_{t-1}} = \tau_t(s_{*_t})$$

Outline

Conclusion

- Conclusion:
  - HDMMs are powerful tools for modeling time-dependent systems with hidden structures.
  - Em estimates the parameters effectively.
  - Viterbi algorithm to deduce the most likely hidden state sequence.
  - Modified Viterbi algorithm to deduce the least risky hidden state sequence.
- Work also carried out (but not presented here):
  - Parameter estimation in global drifting of type $M1\text{-}DMm$ and of type $DM1\text{-}DMm$
  - Viterbi algorithm for global drifting of type $M1\text{-}DMm$ and of type $DM1\text{-}DMm$
  - Modified Viterbi algorithm for global drifting of type $M1\text{-}DMm$ and of type $DM1\text{-}DMm$
- Future work:
  - Piecewise drifting of type $M1\text{-}DMm$ and of type $DM1\text{-}DMm$
  - Polynomial drifting
  - Development of an R package
  - Apply these methods to real-world datasets.

Thank you!

Questions?