Graphical Models: Homework 3

Charles Darmon

Marin Robert De Beauchamp

Master MASH

I HMM - Implementation

We consider the following HMM model:

The chain (q_t) which has 4 possible states for $t = 1 \dots T$

The observation $u_t = (x_t, y_t) \in \mathbb{R}^2$ for $t = 1 \dots T$

An initial probability distribution $\pi \in \mathbb{R}^4$ such that $\sum_{i=1}^{4} \pi_i = 1$

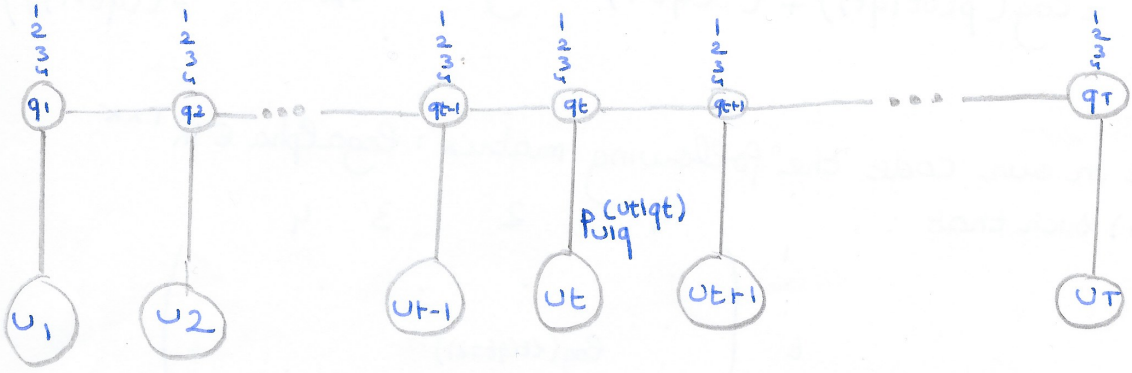
A probability transition matrix $A \in \mathbb{R}^{4 \times 4}$ defined by:

$A_{ij} = P(q_t = i \mid q_{t-1} = j)$ & $\sum_i A_{ij} = 1$ such that

$$A = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & \cdot & \cdot & \cdot & \cdot \\ 2 & \cdot & \cdot & \cdot & \cdot \\ 3 & \cdot & p(q_t=3 \mid q_{t-1}=2) & \cdot & \cdot \\ 4 & \cdot & \cdot & \cdot & \cdot \end{pmatrix}$$

The observations conditionnaly on the current states are obtained from Gaussian emission probabilities $u_t \mid q_t = i \sim \mathcal{N}(\mu_i, \Sigma_i)$

The model can be modelled as following:



(1) We want to implement the α & β - recursions

As we have seen in course we have to implement the α & β recursions using logs in order to prevent numerical errors.

The alpha recursion is given by the following formula:

$$\alpha_t(q_t) = p(u_t \mid q_t) \sum_{q_{t-1}} p(q_t \mid q_{t-1}) \alpha(q_{t-1}) \quad \text{where } t = 1, \dots, T$$

- The first message is initialized as follows:

$\alpha_1(q_1) = \mu_{u_1 \to q_1}(q_1)\, p(q_1) = p(u_1 \mid q_1)\, p(q_1)$ where:

$q_1 \sim \pi$ & $u_1 \mid q_1 = i \sim \mathcal{N}(\mu_i, \Sigma_i)$

- we will denote: $\cdot\ \ell(q_{t-1}) = \log\left(p(q_t \mid q_{t-1})\, \alpha(q_{t-1})\right)$

$\cdot\ i^* = \underset{i \in \{1,\dots,4\}}{\operatorname{argmax}} \log\left(p(q_t \mid q_{t-1} = i)\, \alpha(q_{t-1} = i)\right)$

$\cdot\ \ell^*(q_{t-1}) = \log\left(p(q_t \mid q_{t-1} = i^*)\, \alpha(q_{t-1} = i^*)\right) = \ell(q_{t-1} = i^*)$

Then:

$$\log(\alpha_t(q_t)) = \log(p(u_t \mid q_t)) + \log\left(\sum_{q_{t-1}} p(q_t \mid q_{t-1}) \alpha(q_{t-1})\right)$$

$$= \log(p(u_t \mid q_t)) + \log\left(\sum_{i=1}^{4} \exp(\ell(q_{t-1} = i))\right)$$

$$= \log(p(u_t \mid q_t)) + \ell^*(q_{t-1}) + \log\left(1 + \sum_{i \neq i^*} \exp(\ell(q_{t-1} = i) - \ell^*(q_{t-1}))\right)$$

We create in our code the following matrix: $\text{logalpha} \in \mathbb{R}^{T \times K}$ (where $K = 4$) such that:

$$\begin{array}{c} \\ 1 \\ \vdots \\ t \\ \vdots \\ T \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \log(\alpha_t(q_t = 2)) & & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix} \end{array}$$

where $\log(\alpha_t(q_t = 2)) = \log(p(u_t \mid q_t = 2)) + \ell^*(q_{t-1}) + \log\left(1 + \sum_{i \neq i^*} \exp(\ell(q_{t-1} = i) - \ell^*(q_{t-1}))\right)$

$\downarrow$

$\sim \mathcal{N}(\mu_2, \Sigma_2)$

The Beta recursion is given by :

$$\beta_t(q_t) = \sum_{q_{t+1}} p(v_{t+1}|q_{t+1}) p(q_{t+1}|q_t) \beta_{t+1}(q_{t+1})$$

The last message is given by $\beta_T(q_T) = 1$ and therefore passing on logarithmic scale : $\log(\beta_T(q_t=i)) = 0 \quad \forall i = 1, .., 4$

We will denote : $\cdot e(q_{t+1}) = \log(p(v_{t+1}|q_{t+1}) p(q_{t+1}|q_t) \beta_{t+1}(q_{t+1}))$

$$\cdot j^* = \underset{j \in \{1,..,4\}}{\operatorname{argmax}} \; e(q_{t+1}=j)$$

$$\cdot e^*(q_{t+1}) = e(q_{t+1}=j^*)$$

Then :

$$\log(\beta_t(q_t)) = \log\left(\sum_{q_{t+1}} p(v_{t+1}|q_{t+1}) p(q_{t+1}|q_t) \beta_{t+1}(q_{t+1})\right)$$

$$= \log\left(\sum_{j=1}^{4} \exp(e(q_{t+1}=j))\right)$$

$$= e^*(q_{t+1}) + \log\left(1 + \sum_{j \neq j^*} \exp(e(q_{t+1}=j) - e^*(q_{t+1}))\right)$$

We create the following matrix : $\text{Logbeta} \in \mathbb{R}^{T \times K}$ such that

	1	2	3	4
1	.	.	.	.
t	.	$\log(\beta_t(q_t=2))$	.	.
T	.	.	.	.

where $\log(\beta_t(q_t=2)) = e^*(q_{t+1}) + \log\left(1 + \sum_{j \neq j^*} \exp(e(q_{t+1}=j) - e^*(q_{t+1}))\right)$

and where $e(q_{t+1}=j) = \log(p(v_{t+1}|q_{t+1}=j)) + \log(p(q_{t+1}=j|q_t=2))$

$+ \beta_{t+1}(q_{t+1}=j)$

$\downarrow$ $\mathcal{N}(\mu_j, \Sigma_j)$ $\qquad$ $\downarrow$ Afz

We also need to compute:

$$\rightarrow p(q_t \mid v_1, .., v_T) = \frac{\alpha_t(q_t)\beta_t(q_t)}{\sum_{q_t} \alpha_t(q_t)\beta_t(q_t)}$$

We compute it with the log and denoting:

$$e(q_t) = \log(\alpha_t(q_t)\beta_t(q_t))$$

$$e^*(q_t) = \log(\alpha_t(q_t=i^*)\beta_t(q_t=i^*)) \text{ where } i^* = \underset{i\in\{1,..,4\}}{\text{argmax}} \log(\alpha_t(q_t=i)\beta_t(q_t=i))$$

We obtain:

$$\log(p(q_t \mid v_1, .., v_T)) = e(q_t) - e^*(q_t) - \log\left(1 + \sum_{\substack{i=1 \\ i\neq i^*}}^{4} \exp(e(q_t=i) - e^*(q_t))\right)$$

We create a matrix "proba" $\in \mathbb{R}^{T\times 4}$ such that:

	1	2	3	4
t	.	.	.	.
	.	[p(qt=2|v1,..,vT)]	.	.
T	.	.	.	.

(we pass to the exponential when we add the proba in the matrix)

$$\rightarrow p(q_t, q_{t+1} \mid v_1, .., v_T) = \frac{\alpha_t(q_t)\beta_{t+1}(q_{t+1})p(q_{t+1}\mid q_t)p(v_{t+1}\mid q_{t+1})}{p(v_1, .., v_T)}$$

We recall that $p(v_1, .., v_T) = \sum_{q_t} \alpha_t(q_t)\beta_t(q_t)$

We compute the joint probability with the log and we denote as for the computation of $\log(p(q_t \mid v_1, .., v_T))$: $e(q_t) = \log(\alpha_t(q_t)\beta_t(q_t))$

Then we obtain:

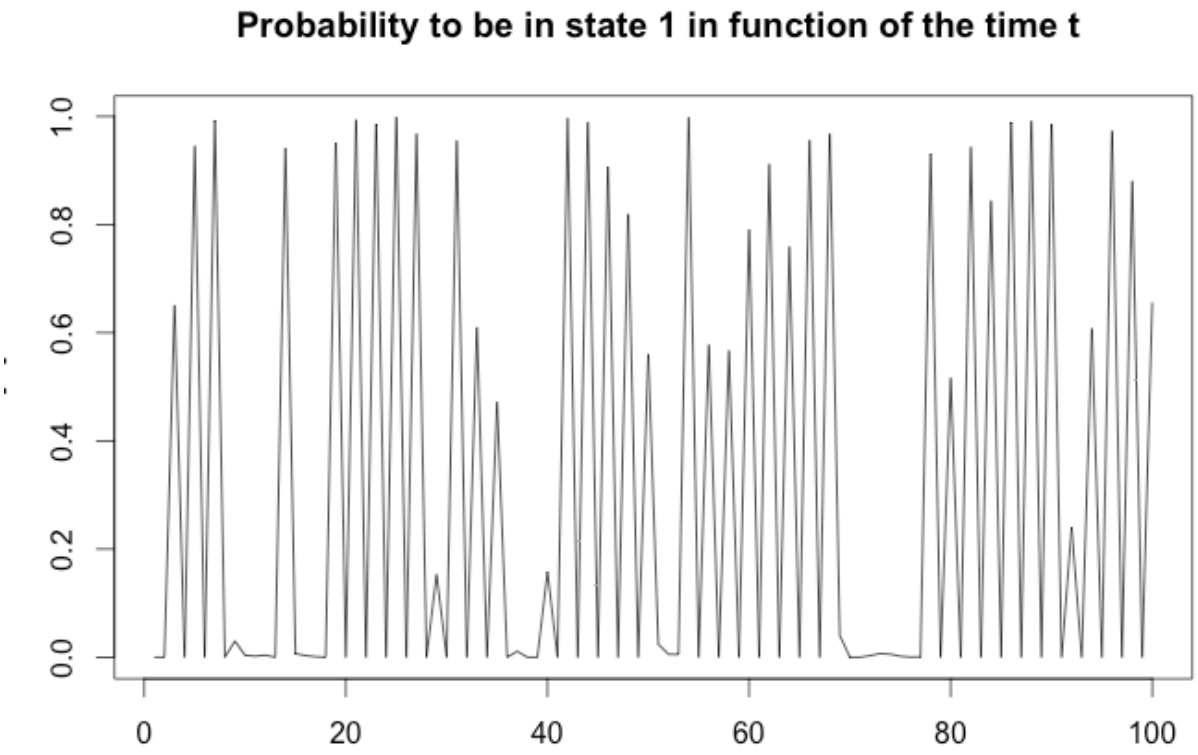
$$\log(p(q_t, q_{t+1} \mid v_1, .., v_T)) = \log(\alpha_t(q_t)) + \log(\beta_{t+1}(q_{t+1})) + \log(p(q_{t+1}\mid q_t)) + \log(p(v_{t+1}\mid q_{t+1}))$$
$$- e^*(q_t) - \log\left(1 + \sum_{\substack{i=1 \\ i\neq i^*}}^{4} \exp(e(q_t=i) - e^*(q_t))\right)$$

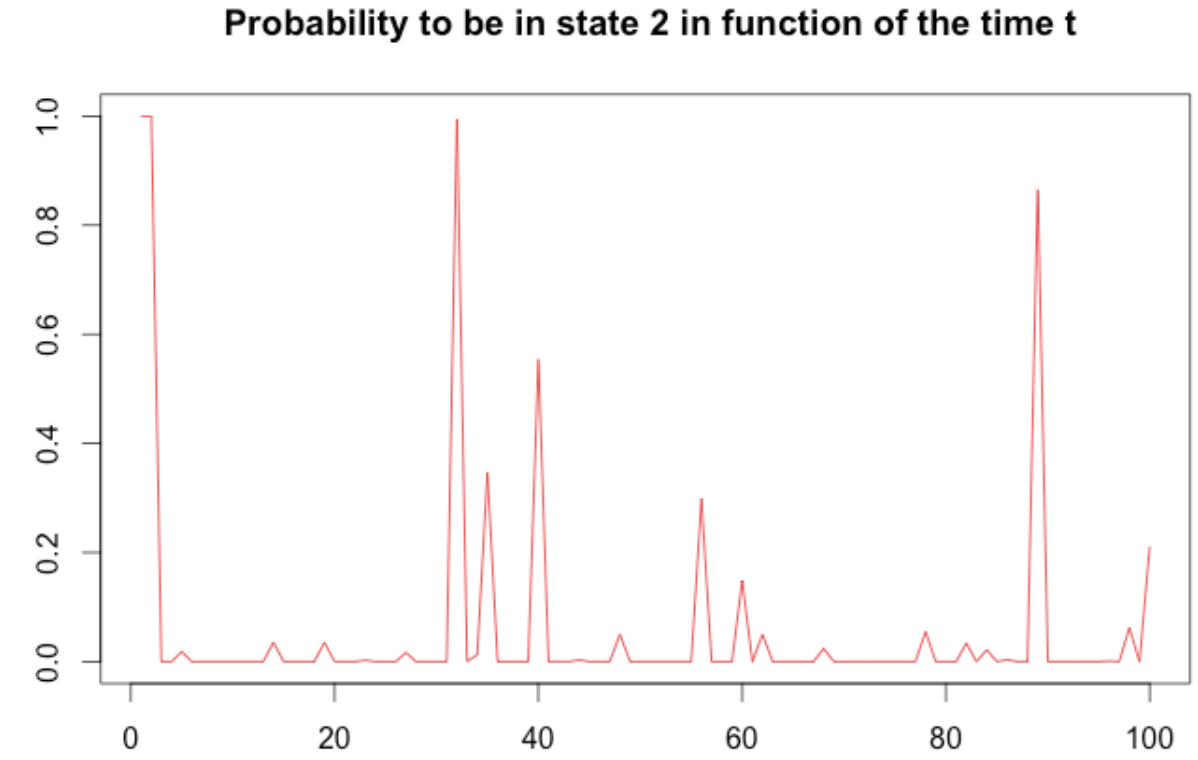
We create a list of matrices where the t^{th} element is $p(q_t, q_{t+1} \mid v_1, .., v_T) \in \mathbb{R}^{4\times 4}$ $(t \in 1, .., T-1)$ defined such that:

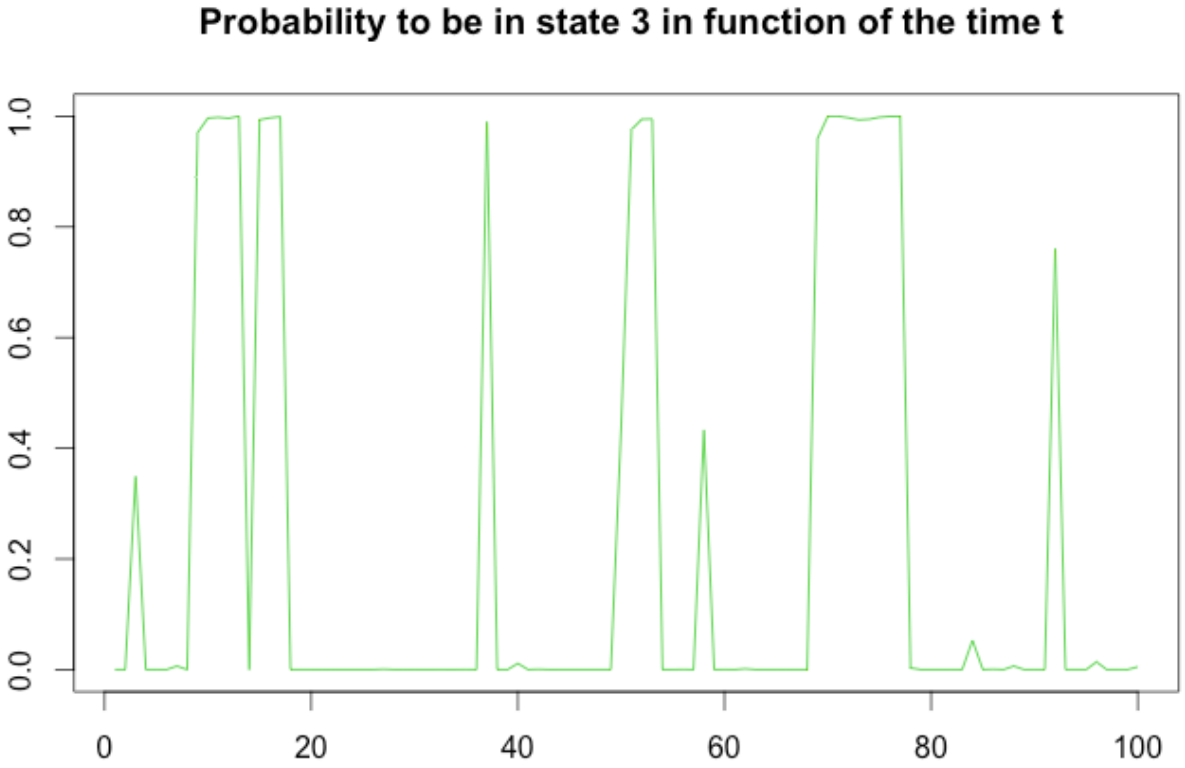
	1	2	3	4
1	.	.	.	.
2	.	.	.	.
3	.	p(q2=3, q1=2 | v1, .., vT)	.	.
4	.	.	.	.

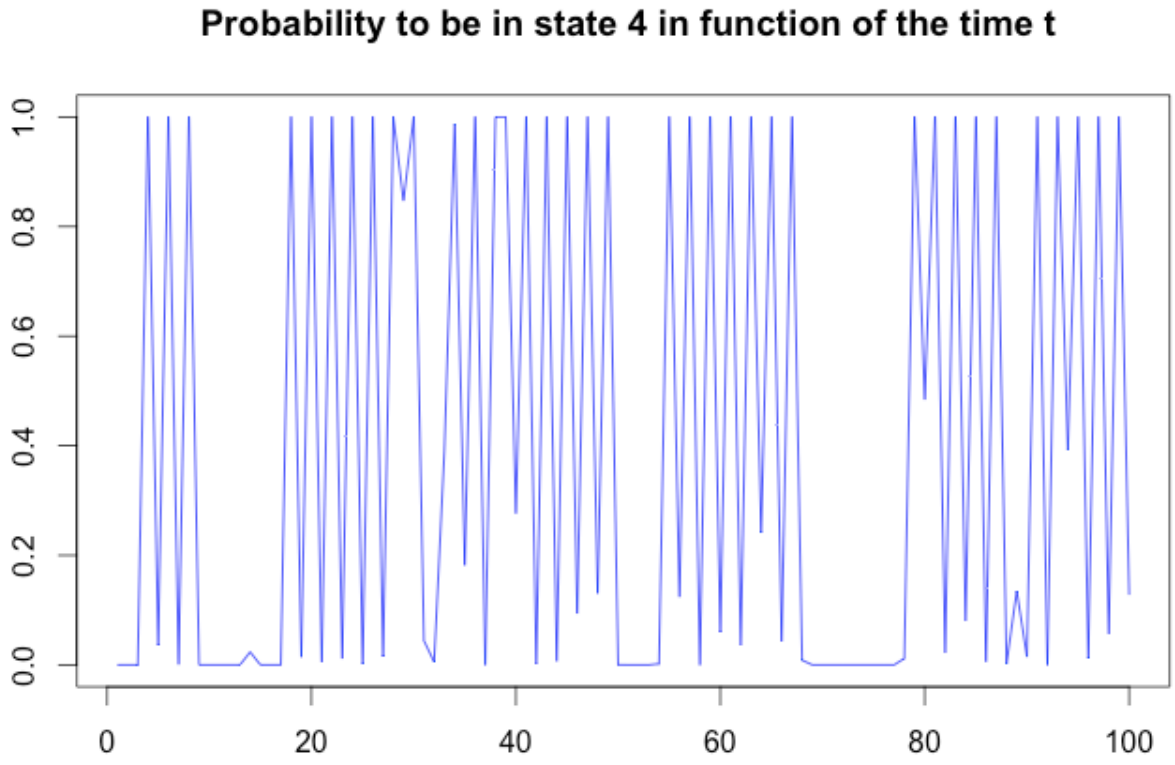
← matrix [[1]] of the list

Question 2 :


Probability to be in state 1 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Probability to be in state 2 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Probability to be in state 3 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Probability to be in state 4 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100

(3) We are going to implement the E.M algorithm.

To do this, we are going to derive the estimation equations of the E.M algorithm.

Our goal is to maximise the uncomplete likelihood defined by

$$\log(p(u_1,\dots,u_T)) = \log\Big(\sum_{q_t} \alpha_t(q_t)\beta_t(q_t)\Big)$$

To do this we are going to compute the complete likelihood

we denote $\theta = (\pi, A, \Sigma, \mu)$.

Then : $$l_c(\theta) = \log\Big(p(q_1)\prod_{t=1}^{T-1} p(q_{t+1}|q_t)\prod_{t=1}^{T} p(u_t|q_t)\Big)$$

$$= \sum_{i=1}^{K} \delta(q_1=i)\log(\pi_i) + \sum_{t=1}^{T-1}\sum_{i,j=1}^{K} \delta(q_{t+1}=i, q_t=j)\log(A_{i,j})$$

$$+ \sum_{t=1}^{T}\sum_{i=1}^{K} \delta(q_t=i)\log(p(u_t|q_t))$$

E-Step : $$\mathbb{E}_{q|u}[l_c(\theta)] = \sum_{i=1}^{K} p(q_1=i|u_1,\dots,u_T)\log(\pi_i)$$

$$+ \sum_{t=1}^{T-1}\sum_{i,j=1}^{K} p(q_{t+1}=i, q_t=j|u_1,\dots,u_T)\log(A_{ij})$$

$$+ \sum_{t=1}^{T}\sum_{i=1}^{K} p(q_t=i|u_1,\dots,u_T)\log(p(u_t|q_t=i))$$

M-Step : We are going to maximise according each parameter one by one since everything decouple.

We first care about π: the associated problem is given by :

$$\max_{\pi_i}\Big(\sum_{i=1}^{4} p(q_1=i|u_1,\dots,u_T)\log(\pi_i)\Big) \quad \text{for } i=1,\dots,4$$

$$\sum_{i=1}^{4}\pi_i = 1$$

We introduce the lagrangian :

$$\mathcal{L}(\pi,\lambda) = \sum_{i=1}^{4} p(q_1=i|u_1,\dots,u_T)\log(\pi_i) + \lambda\Big(\sum_{i=1}^{4}\pi_i - 1\Big)$$

Then $\frac{\partial \mathcal{L}}{\partial \pi_i} = \frac{p(q_1 = i \mid u_1, \dots u_T)}{\pi_i} + \lambda$

$\Rightarrow \frac{\partial \mathcal{L}}{\partial \pi_i} = 0 \Rightarrow \pi_i = -\frac{p(q_1 = i \mid u_1, \dots, u_T)}{\lambda}$

Since $\sum_{i=1}^{4} \pi_i = 1 \Rightarrow \lambda = -\sum_{i=1}^{4} p(q_1 = i \mid u_1, \dots, u_T)$

Therefore $\pi_i^* = \frac{p(q_1 = i \mid u_1, \dots, u_T)}{\underbrace{\sum_{i=1}^{4} p(q_1 = i \mid u_1, \dots, u_T)}_{=1}} = p(q_1 = i \mid u_1, \dots, u_T)$

$= \frac{\alpha_1(q_1 = i)\beta_1(q_1 = i)}{\sum_{i=1}^{K} \alpha_1(q_1 = i)\beta_1(q_1 = i)}$

We know care about A : the associated problem is given by

$$\max_{A_{ij}} \sum_{t=1}^{T-1} \sum_{i,j=1}^{4} p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T) \log(A_{ij})$$

$$\sum_{i=1}^{4} A_{ij}$$

The associated Lagrangian is given by :

$$\mathcal{L}(A, \lambda) = \sum_{t=1}^{T-1} \sum_{i,j=1}^{4} p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T) \log(A_{ij}) + \lambda \left(\sum_{i=1}^{4} A_{ij} - 1 \right)$$

Then $\frac{\partial \mathcal{L}}{\partial A_{ij}} = \sum_{t=1}^{T-1} \frac{p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T)}{A_{ij}} + \lambda$

$\Rightarrow \frac{\partial \mathcal{L}}{\partial A_{ij}} = 0 \Rightarrow A_{ij} = -\frac{\sum_{t=1}^{T-1} p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T)}{\lambda}$

Since $\sum_{i=1}^{4} A_{ij} = 1 \Rightarrow \lambda = -\sum_{i=1}^{4} \sum_{t=1}^{T-1} p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T)$

$= -\sum_{t=1}^{T-1} p(q_t = j \mid u_1, \dots, u_T)$

$$\Rightarrow A_{ij}^* = \frac{\sum_{t=1}^{T-1} p(q_{t+1} = i, q_t = j \mid u_1, \dots, u_T)}{\sum_{t=1}^{T-1} p(q_t = j \mid u_1, \dots, u_T)}$$

We now care about (Σ, μ).

We recall that : $u_t | q_t = i \sim \mathcal{N}(\mu_i, \Sigma_i)$

We have :

$$\sum_{t=1}^{T} \sum_{i=1}^{k} p(q_t = i | u_1, \dots, u_T) \log(p(u_t | q_t = i))$$

$$= \sum_{t=1}^{T} \sum_{i=1}^{k} p(q_t = i | u_1, \dots, u_T) \left\{ \log\left(\frac{1}{(2\pi)^{k/2}} \right) + \log\left(\frac{1}{|\Sigma_i|^{1/2}} \right) - \frac{1}{2} (u_t - \mu_i)^t (\Sigma_i)^{-1} (u_t - \mu_i) \right\}$$

we have :

$$\frac{\partial \dots}{\partial \mu_i} = \sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T) \left((\Sigma_i)^{-1} u_t - (\Sigma_i)^{-1} \mu_i \right)$$

$$\frac{\partial \dots}{\partial \mu_i} = 0 \Rightarrow \sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T) u_t = \sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T) \mu_i$$

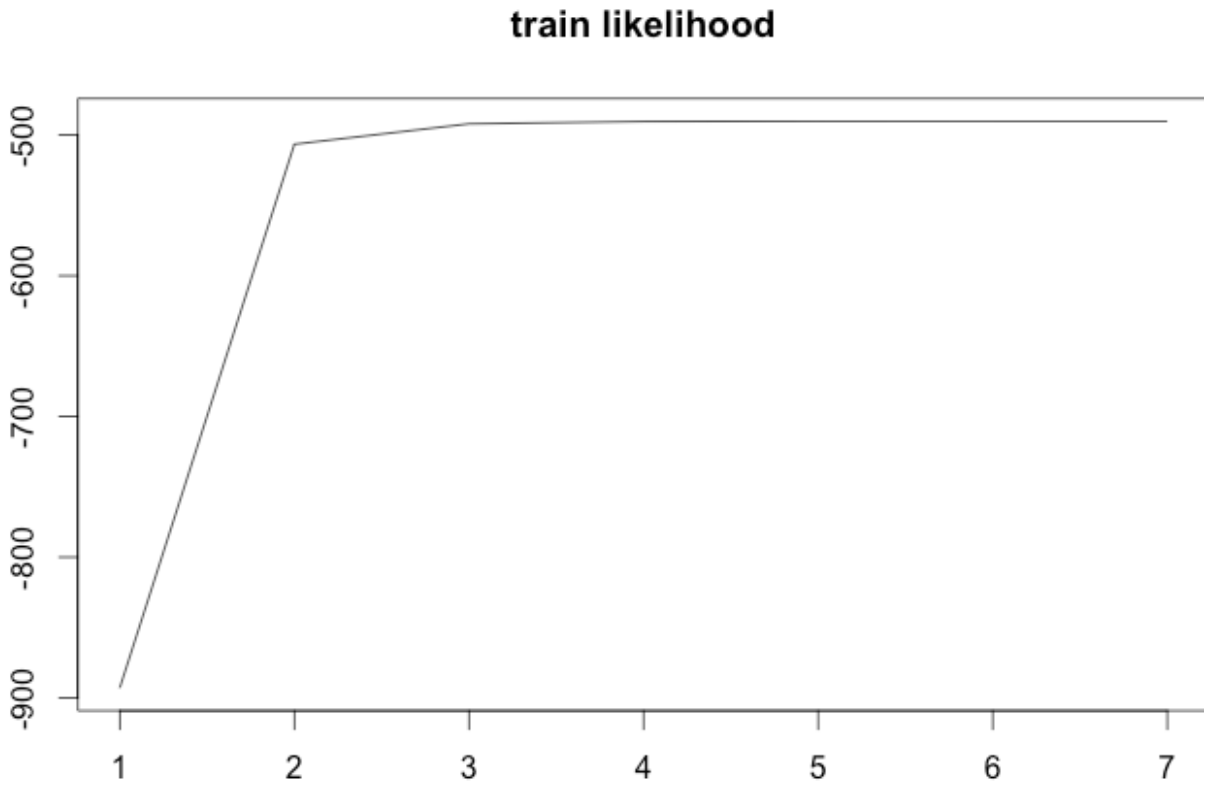
$$\Rightarrow \mu_i^* = \frac{\sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T) u_t}{\sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T)}$$

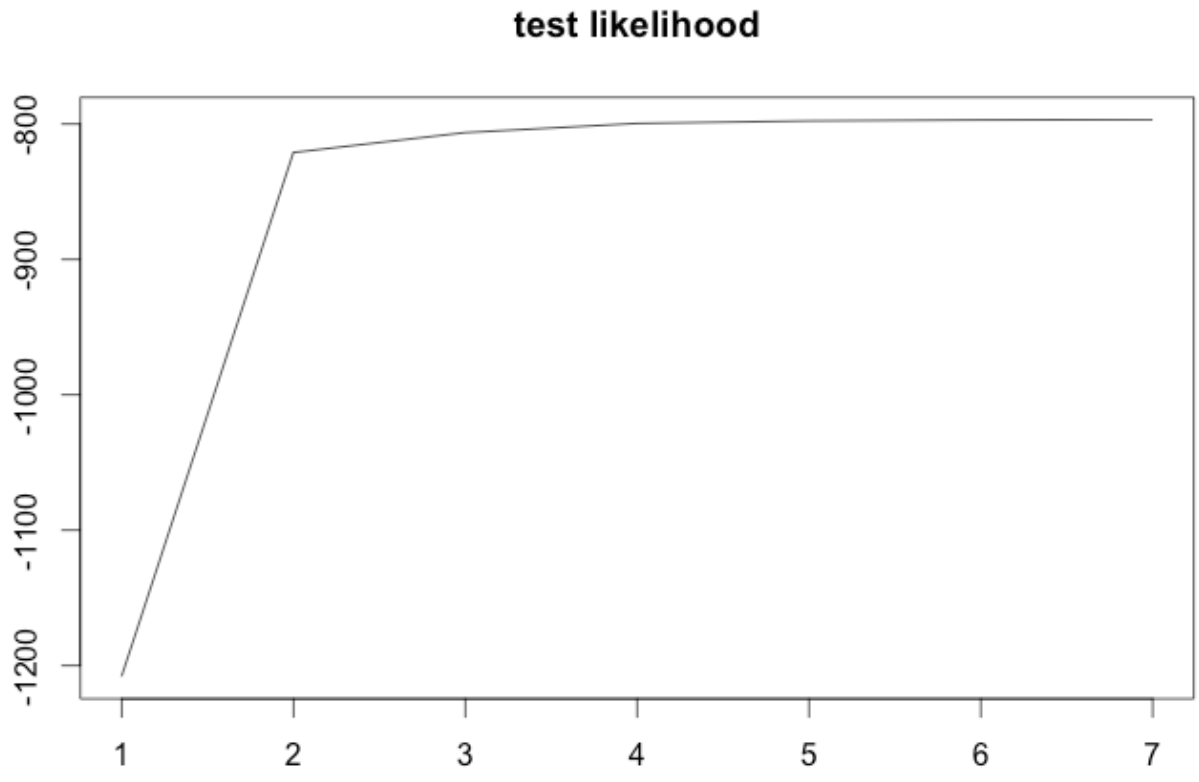
For the maximization over Σ, we use the result that we find in homework 2 adapted here :

$$\Sigma_j^* = \frac{\sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T) (u_t - \mu_j^*)(u_t - \mu_j^*)^T}{\sum_{t=1}^{T} p(q_t = i | u_1, \dots, u_T)}$$

Question 5:

The likelihood of the train data is higher than the one of the test data because we learned the parameters on the train data.





Question 6:

Comparison of the Log-Likelihood

	Gaussian Mixture	HMM
Train	−921.0602	-490.1315
Test	−1214.876	-796.6534

➔ **We recall that we have scaled the data train & the data test:**
➔ **We can replace Train<-scale(Data) by Train<-as.matrix(Data) to modify it.**

In the HMM: We consider the the conditional probability of being in stake i at position t given the entire observe sequence: $p(q_t = i|u_1, \dots, u_T)$.

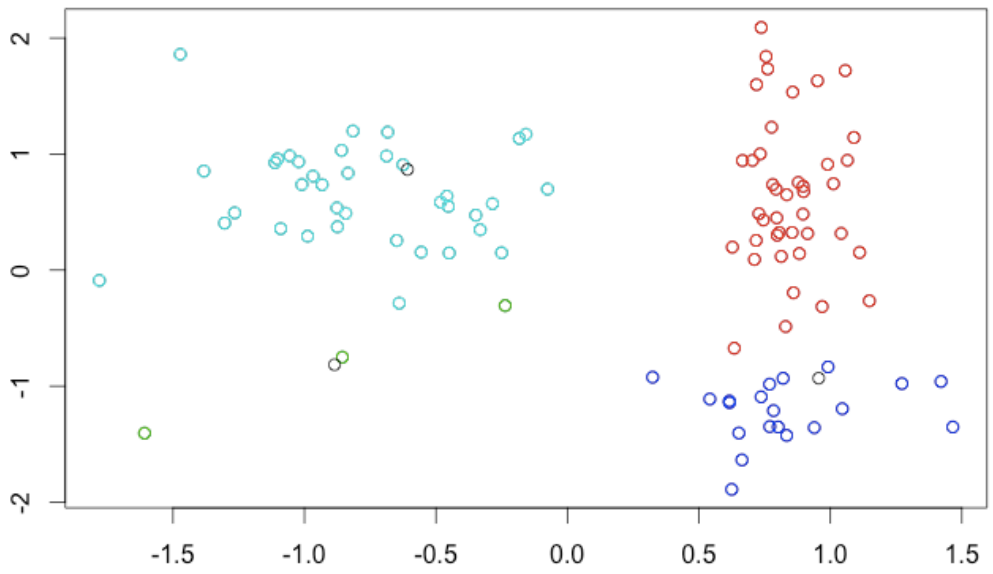
In the Gaussian Mixture Model: We consider the posterior probability of a state given the observation : $p(q_t = i|u_t)$.

The HMM depends on the entire sequence because of the underlying Markov Process whereas the Gaussian Mixture Model can be seen as a special Hidden Markov Model with the underlying state process that are IID.

We will therefore get a higher likelihood in the HMM than in the Gaussian Mixture Model.

Question 8:

The data in 2D with the cluster black centers



7) Viterbi algorithm.

The goal of Viterbi algorithm is to compute

$$\arg\max_{(q_t)_{1\le t\le T}} p\left((q_t)_{1\le t\le T} \mid (u_t)_{1\le t\le T} \right) \qquad (*),$$

given HMM parameters $p(q_1)$, $p(q_{t+1}|q_t)$, $p(u_t|q_t)$,
and $(u_t)_{1\le t\le T}$ is the sequence of observation.

$$(*) = \arg\max p\left(\left((q_t)_{1\le t\le T} \right), \left((u)_{1\le t\le T} \right) \right)$$

$$= \arg\max_{k} \max_{(q_t)_{1\le t\le T-1}} p\left(q_T = k, (q_t)_{1\le t\le T-1}, (u_t)_{1\le t\le T} \right)$$

V_T^k to compute recursively

V_T^k is the probability of the most likely sequence of states ending at state $q_T = k$.

By induction we have:

$$V_t^k = \max_{q_1,\dots,q_{t-1}} p\left(q_t = k, q_1, \dots, q_{t-1}, u_1, \dots, u_t \right)$$

$$= p(u_t | q_t = k) \max_i p(q_t = k \mid q_{t-1} = i)\, V_{t-1}^i$$

Hence we can compute the V_t^k $\forall\ 1\le t\le T$, $\forall\ 1\le k\le 4$,
the probabilities of the most likely sequence of $(t-1)$ states ending at state $q_{t-1} = k$.

Finally we just have to take the higher probability at each time and the corresponding state:

$$q_T^* = \arg\max_k \; V_T^k$$

$$q_{t-1}^* = \arg\max_k \left\{ p(q_t^* \mid q_{t-1} = i) \; V_{t-1}^i \right\}$$

And recursively we find the optimal sequence

$$(q_1^*, \dots, q_T^*).$$

Algorithm:

- Initialize: $V_1^k = p(u_1 \mid q_1 = k)\, p(q_1 = k) \quad \forall k$
- Iterate: for $t = 2, \dots, T$: $\forall k$:

$$V_t^k = p(u_t \mid q_t = k) \max_i \left\{ p(q_t = k \mid q_{t-1} = i) V_{t-1}^i \right\}$$

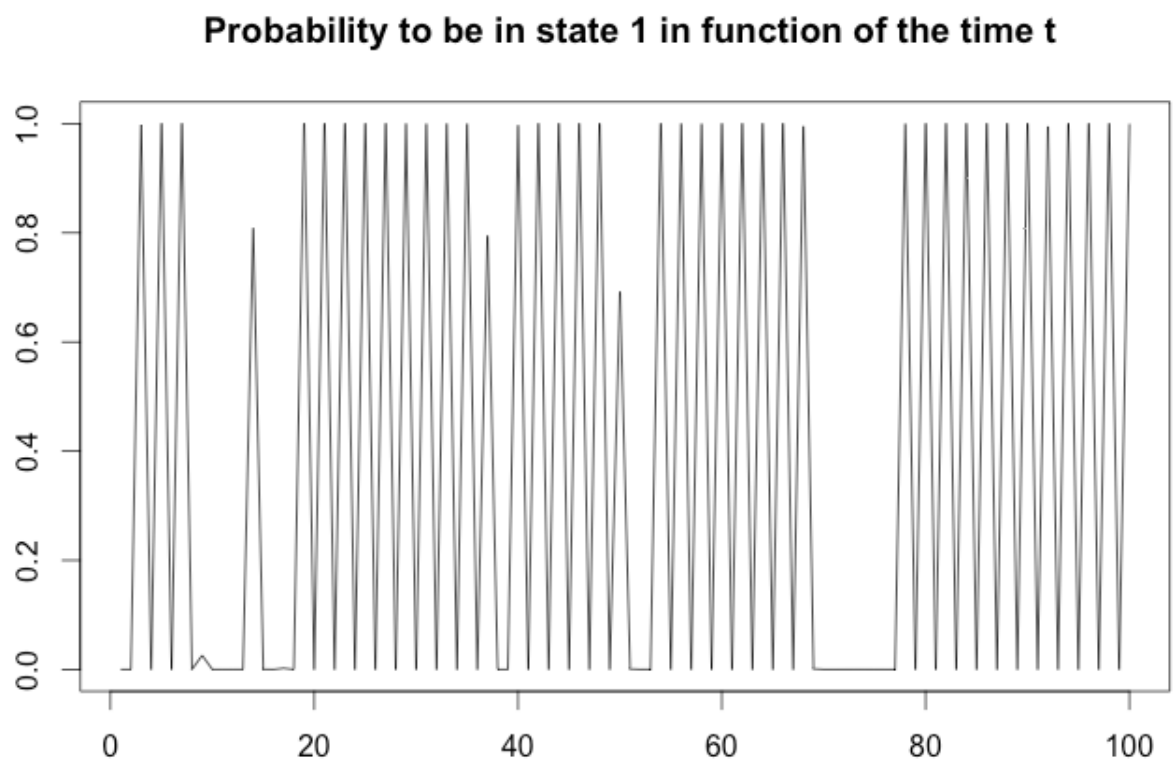
- Termination: $\max_{(q_t)_{1 \le t \le T}} p\left((q_t)_{1 \le t \le T}, (u_t)_{1 \le t \le T}\right) = \max_k V_T^k$
- Traceback: $q_T^* = \arg\max_k V_t^k$

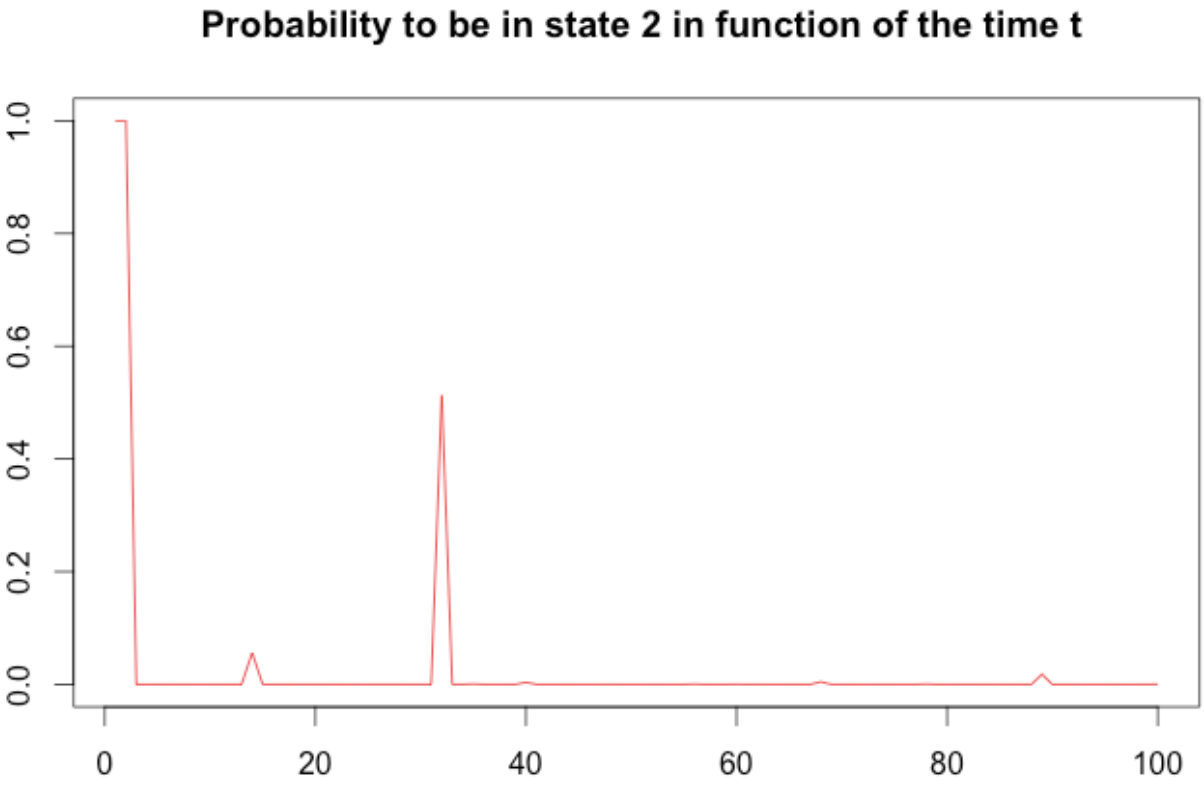
$$q_{t-1}^* = \arg\max_k \left\{ p(q_t^* \mid q_{t-1} = k) \; V_{t-1}^k \right\}$$

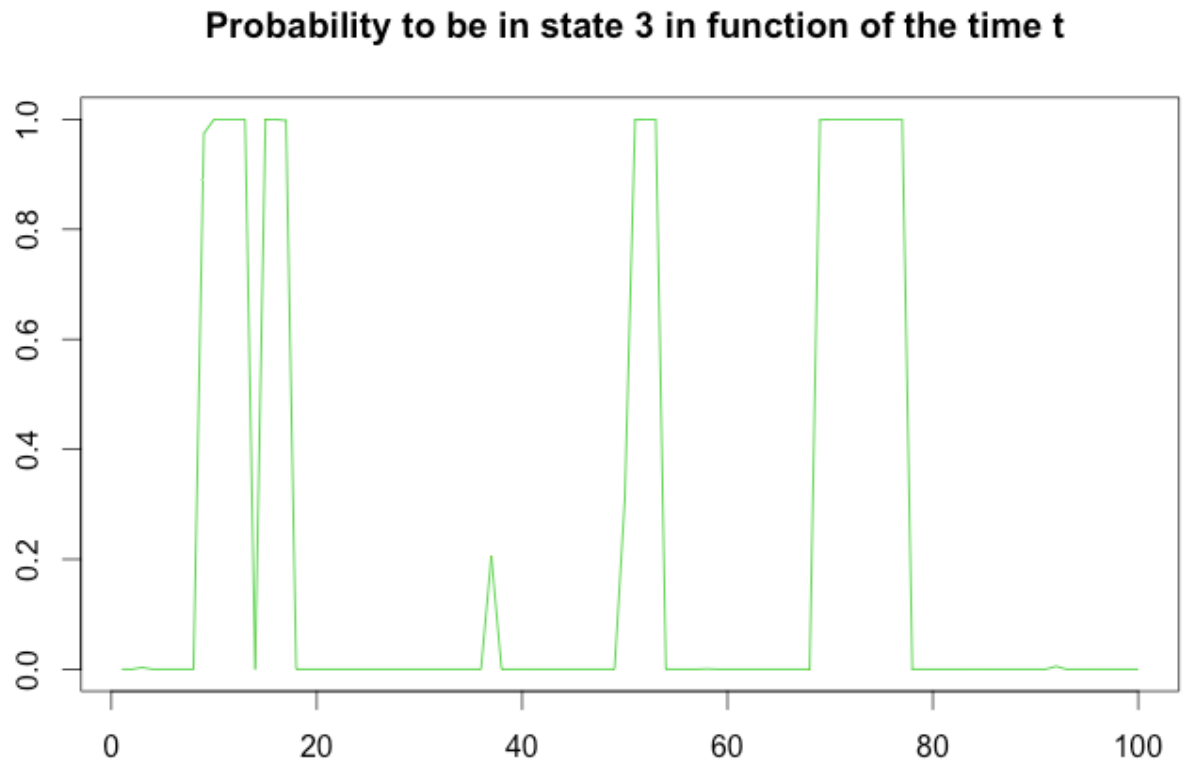
Return $(q_1^*, \dots, q_T^*)$

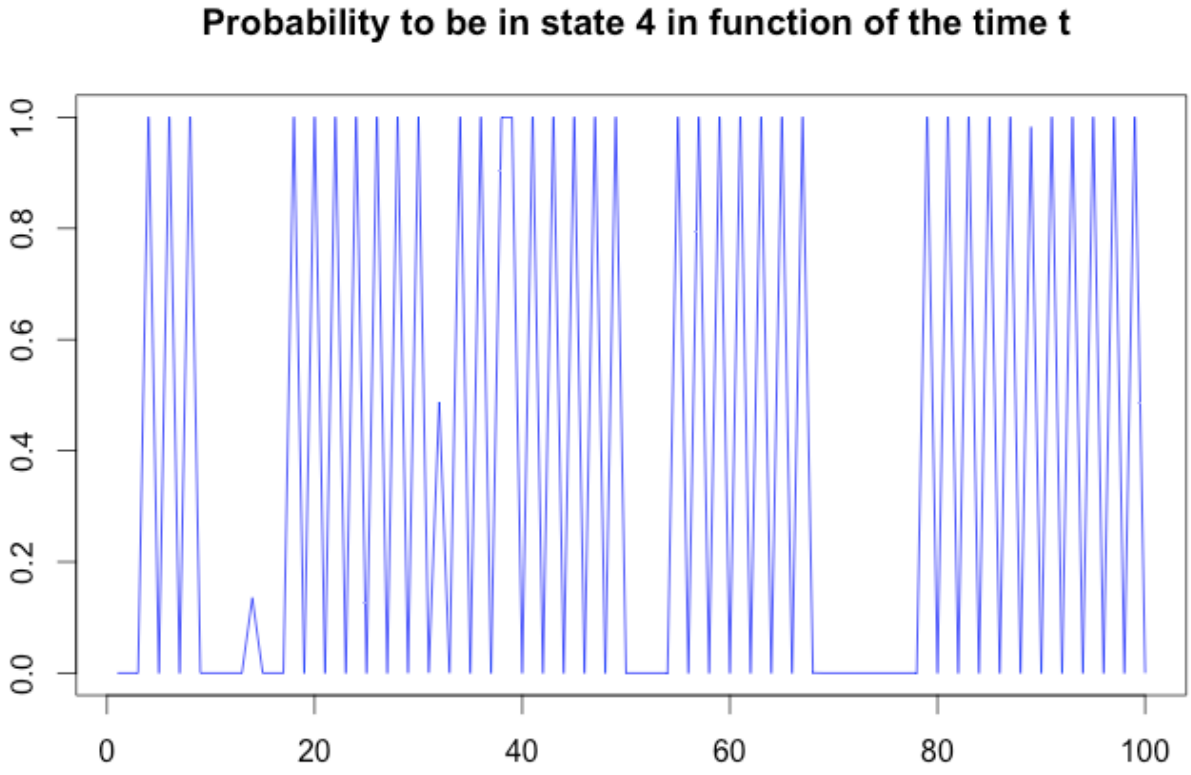
Question 9:

Plot with the parameters learned with the training set.

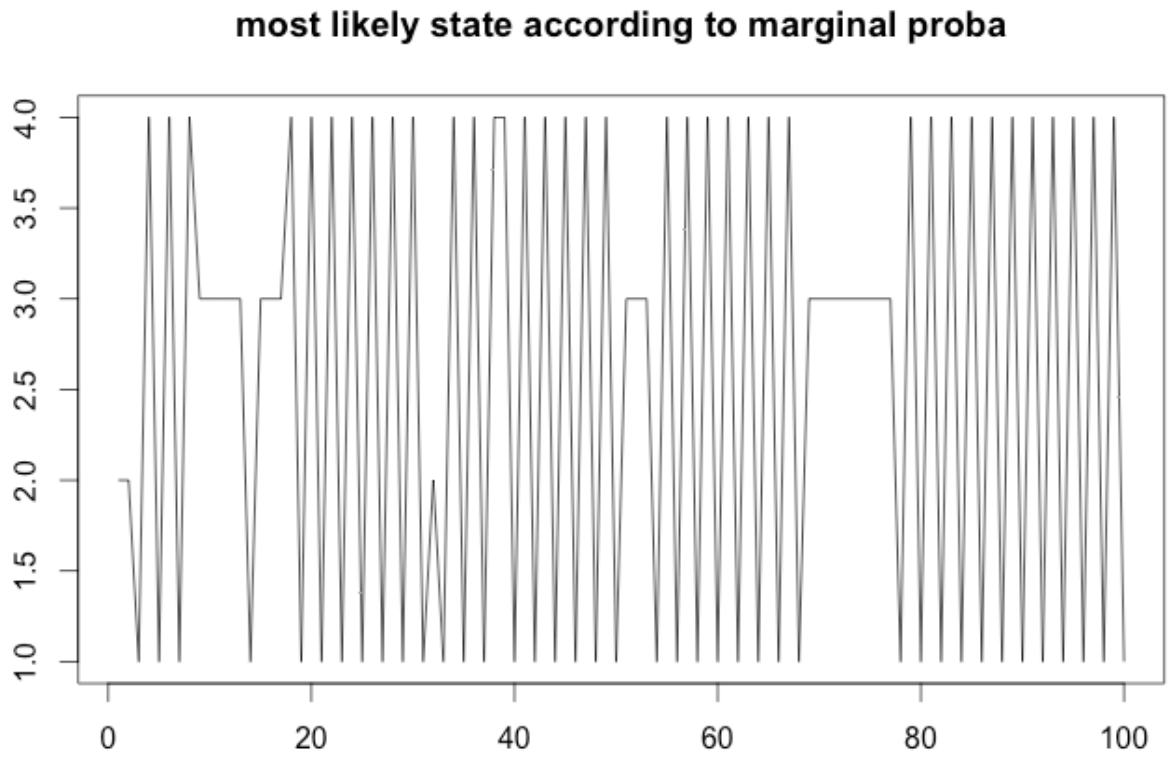




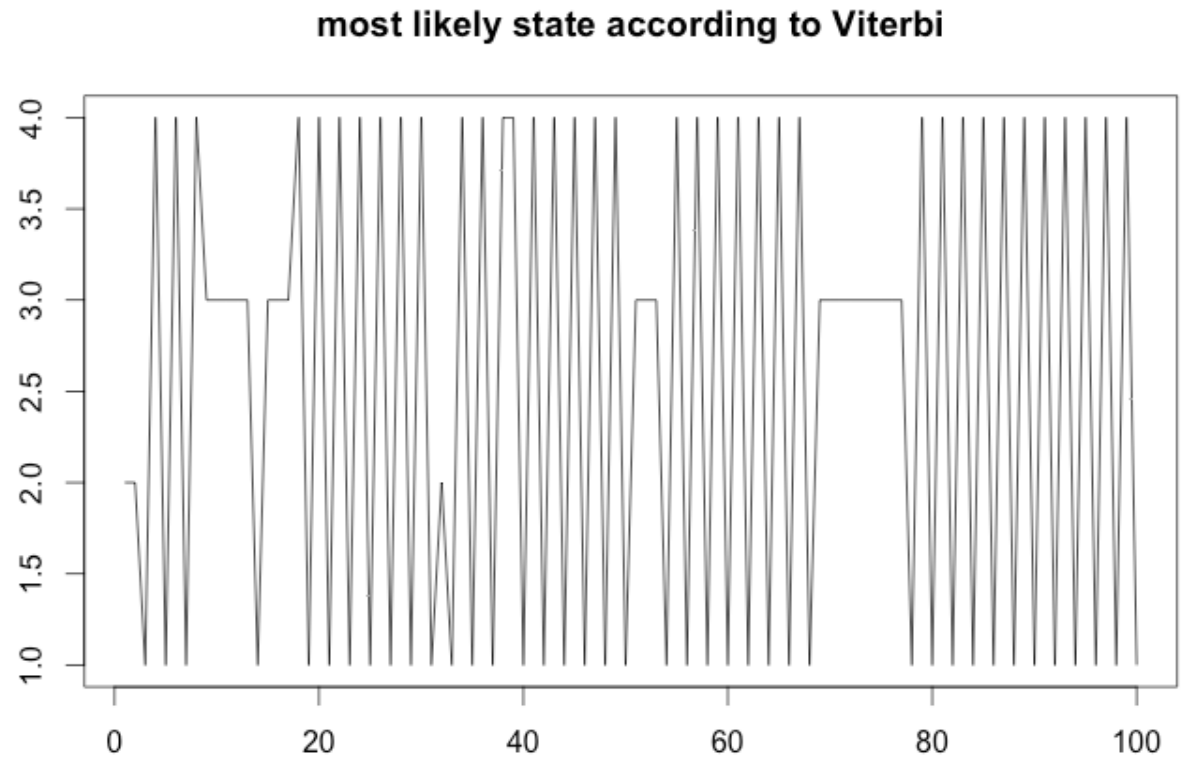

Probability to be in state 3 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Probability to be in state 4 in function of the time t
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100

10)



Question 11:



As we can observe, we find the same states according to the Viterbi Algorithm and the marginal probability which seems logical since the most probable states sequence is near to the sequence of more probable states.

Question 12:

If the number of states is not known, we can't not only refer on the likelihood anymore.
If we increase the number of states:

- ➔ We will get a higher likelihood.
- ➔ We will increase the number of parameters such as transition probabilities and initial state probabilities which lead to slow down our program.
- ➔ We even can reach a distortion of 0 when we take the numbers of states equals to the size of the data. In that case, each data point is the center of its own center which doesn't have interest.

We have to make a compromise between the numbers of states and the number of parameters that we will have to compute.
We have to use some criteria as the Bayesian Information Criterion or the Akaike Information Criterion which take into account the likelihood, the number of parameters and also the size of the sample for the (BIC).