Pattern recognition system
Exercise Project

Report Assignment n°1

Student
Antoine Honoré
honore@kth.se

Student
Audrey Brouard
brouard@kth.se

2015 Period 1

I

Let us calculate $P(S_t = j)$ $\forall j \in \{1, 2\}$ and for t = 1,2,3,...

We will do it $\forall t \in \{1..3\}$ and notice that this is constant $\forall t$.

— For t = 1, the probabilities are given by the matrix q_j. Thus, $P(S_1 = 1) = 0.75$ and $P(S_1 = 2) = 0.25$.

$\forall t \geq 2$, we will use the following formula :

— $P(S_t = j) = \sum_{i=1}^{2} P(S_t = j, S_{t-1} = i) = \sum_{i=1}^{2} P(S_t = j|S_{t-1} = i)P(S_{t-1} = i)$

For the case t = 2, we have :

— $P(S_t = j) = \sum_{i=1}^{2} a_{ij}q_i$

We thus obtain $P(S_2 = 1) = 0.75$ and $P(S_2 = 2) = 0.25$. We immediately notice that

$$\forall i \; P(S_2 = i) = P(S_1 = i)$$

. By recurrency, we have $\forall t \; \Pr(S_t = j)$ constant.

II

After we generated 10 000 state integers, we found the following probabilities :

$$\Pr(S_t = 1) = 0.7525 \; and \; \Pr(S_t = 2) = 0.2475$$

III @HMM/rand

III.1 Theorical calculation

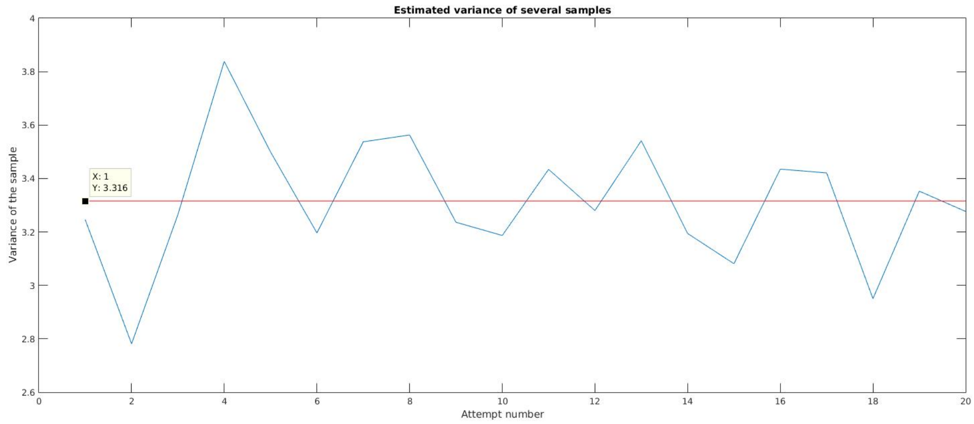
Let us now calculate $E[X_t]$ and $Var[X_t]$.

— $E[X_t]$: the book gives the formula $E[X] = E_S[E_X[X|S]]$, and according to the two different possible values of S, X density probability function is either b_1 or b_2.
Then, for j = 1, $E[X|S] = \mu_1$; and for j = 2, $E[X|S] = \mu_2$.
$E[X] = P(S = 1) * \mu_1 + P(S = 2) * \mu_2 = 0 + 3 * 0.25 = 0.75$.

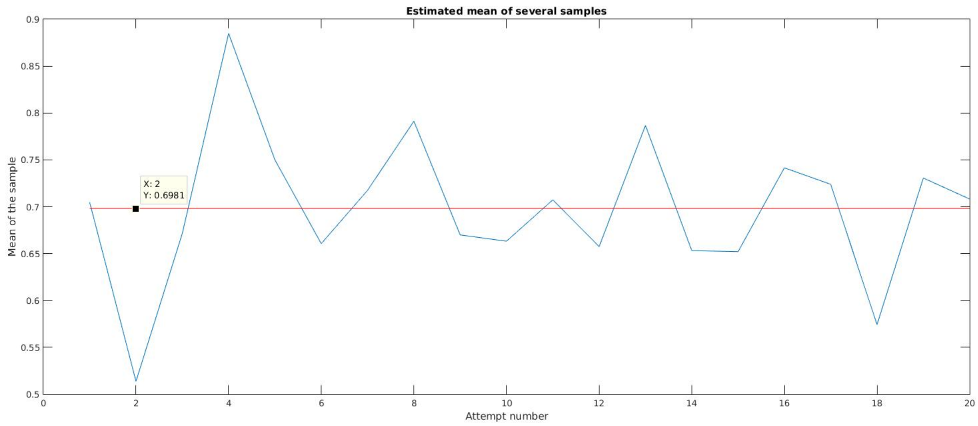
— $Var[X_t]$: according to the book, $Var[X_t] = E_S[Var_X[X|S]] + Var_S[E_X[X|S]]$.
Thanks to the same observation as before, the expression becomes :
$Var[X_t] = [0.75 * \sigma_1 + 0.25 * \sigma_2] + [0.75 * \mu_1^2 + 0.25 * \mu_2^2 - 0.75^2] = 1 + 0.25 * 9 - 0.75^2 = 2.6875$.

III.2 Measures

For these measures, we use the section "Test @HMM/rand" in the file main.m As we can see on the figures 1a and 1b the experimental values are close from the theoretical ones.



(a) $Var(X) \simeq 3.316$



(b) $E(X) \simeq 0.6981$

FIGURE 1 – Validation of @HMM/rand
Blue : Plot of the 20 attempts
Red : Mean over the 20 attempts

IV HMM behavior 1/2

For this part, we changed the behavior of the @HMM/rand function, in order to get a vector from b_1 (not a scalar like the previous question). So here, each sample is a vector $x_t \sim N(\mu_j, \sigma_j^2) of size 500$ (where j=1 or 2).

The following figure shows the vector ploted for different values of t.

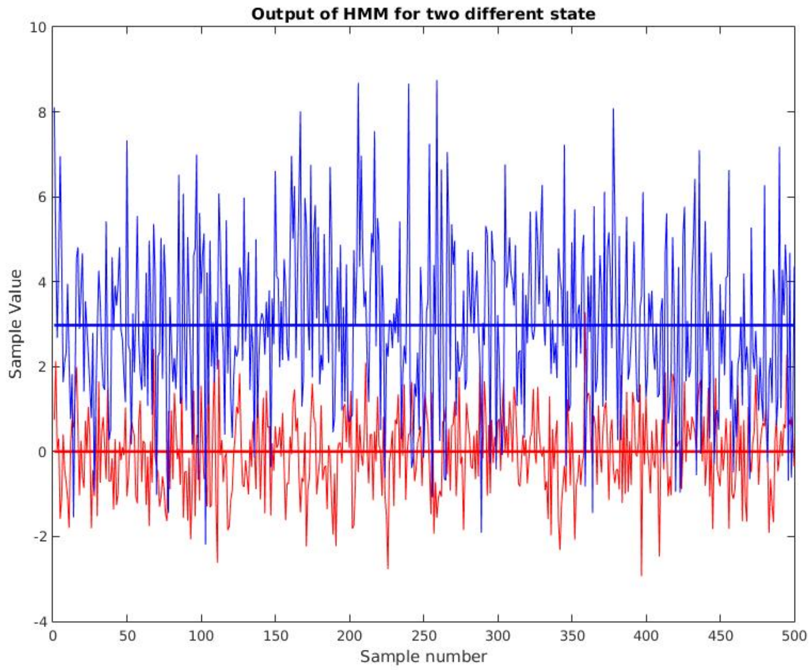


FIGURE 2 – Blue : Sample at a time t_1. $\mu = -0.0065$ and $\sigma^2 = 0.985$
Red : Sample at a time t_2. $\mu = 2.9759$ and $\sigma^2 = 3.8953$
The width lines represent the mean of the samples

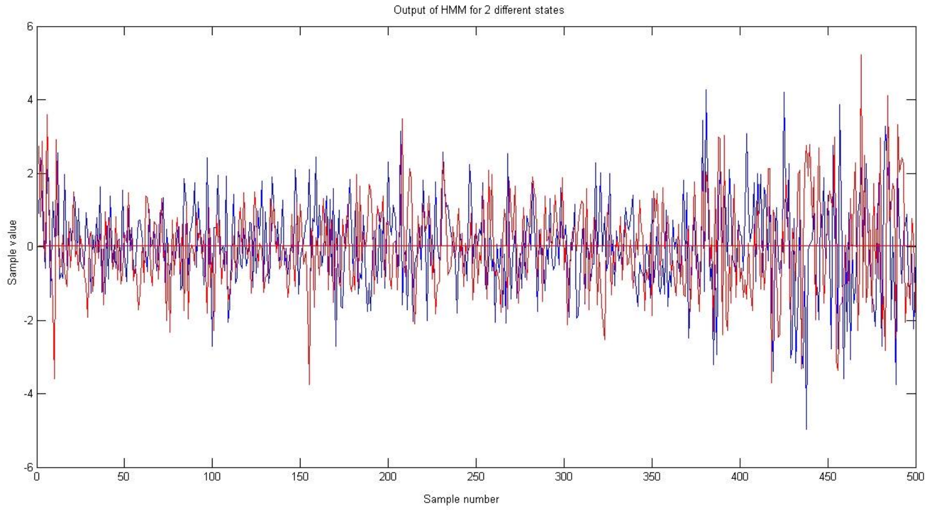
Observation

It is easy to find out in which state the system was when the sample was produced. On the figure 2 the samples have very different means and variances.

V HMM behavior 2/2

We now modify the mean of b_2, μ_2 so that it is equal to μ_1 : $\mu_1 = \mu_2 = 0$.

We can easily notice a change with the previous HMM. On the figure below are represented a sample produced while the state was 1 in blue, and when the state was 2 in red :



Here, we can see that both means are mingled so it seems impossible to determine whether the observation sample was produced under the state 1 or 2. Even if the standard deviations are different, it is not sufficient to spot the state at the moment the sample was produced.

VI Finite-duration HMMs

The implementation works with finite duration Markov chain.
We defined a transition matrix $A = \begin{pmatrix} 0.6 & 0.3 & 0.1 \\ 0.5 & 0.4 & 0.1 \end{pmatrix}$ for a Markov chain, then a HMM based on this Markov chain, and the call to @HMM/rand returns the correct information. The difference is that we don't know how long the sequence will be since we don't know precisely when the system will enter the exit state.