EQ2340 Pattern Recognition Project Assignment 1 Report

Feiyang Liu (liuf@kth.se)
Baoqing She (baoqing@kth.se)

***Abstract*—HMM[1](Hidden Markov Model) is widely used to characterize signals in many fields nowadays, due to its superiority, such as simple mathematical structure, describing complex feature-vector sequences, automatically adapted among others. We want to create and analyze HMM in our project.**

I. Introduction

In this report, we look into basic property of HMM, that is, the way an HMM generates sequences with one kind structured randomness. To be specific, we are supposed to code and verify and an HHM which can generate an output sequence of random real numbers with scalar Gaussian output distributions. For generalization, our model is required to work with vector random variables as well. All of our simulations are based on MATLAB, and all relevant source codes are attached in zip archive.

II. Theoretical Knowledge

Any HHM is defined by a set of λ which are consist of two time-invariant objects, a Markov Chain (MC) and an array of Output Probability Distribution (B). While MC is defined by two parameter, q and matrix A. q is initial state probability distribution and A is a transition probability matrix. B characterizes the distribution of each of the sub-sources. For brevity, we will not include much knowledge about HMM. For more details, refer to [1].

$$\lambda = \{MC, B\} = \{\{q, A\}, B\} \tag{1}$$

III. verify HMM Random Source

For the vilification of our HMM, we are given the following infinite-duration HMM $\lambda = \{q, A, B\}$ as a first test example:

$$q = \begin{bmatrix} 0.75 \\ 0.25 \end{bmatrix}, A = \begin{bmatrix} 0.99 & 0.01 \\ 0.03 & 0.97 \end{bmatrix}, B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{2}$$

where $b_1(x)$ is a scalar Gaussian density function with mean μ_1=0 and standard deviation σ_1 = 1, and $b_2(x)$ is a similar distribution with mean μ_2 = 3 and standard deviation σ_2 = 2.

A. Question 1

In this question, we want to prove state probability $P(S_t = j)$ is constant for all t. that is, $P_{t+1}^T = P_t^T * A$. Assume t=1, we immediately obtain:

$$P_2^T = P_1^T * A = q * A \tag{3}$$

substitute (2) in (3):

$$P_2^T = \begin{bmatrix} 0.75 \\ 0.25 \end{bmatrix} = P_1^T \tag{4}$$

The step from t = 1 to t = 2 is equivalent to a step from any t-1 to t, and the forward step can be immediately generalized: $P_{t+1}^T = P_t^T = ... = P_1^T = q$. Hence, $P(S_t = j)$ is constant for all t.

B. Question 2

In this question, we use the Markov Chain we build to generate 10000 state integer numbers. Then frequency of occurrences of S_t = 1 and S_t = 2 are computed to see if the practical results are identical to the theoretical ones.
The results from MATLAB is demonstrated in Table I. The relative frequency of occurrences of S_t = 1 and S_t = 2 are 0.7399 and 0.2601 respectively, which approximately equals to P(S_t).

TABLE I
Simulation Results

Frequency of Occurrences	Theoretical value	Simulation value
$S_t = 1$	0.75	0.7399
$S_t = 2$	0.25	0.2601

C. Question 3

In this question, we deal with the distribution of output sequence to verify our model. Firstly, we need to obtain the theoretical property of output distribution,i.e. mean and variance. We have the following conditional expectation formulas formulas to help compute the mean and variance of output sequence.

$$\mu_X = E[X] = E_Z[E_X[X|Z]] \tag{5}$$

$$var(X) = E_Z[var_X(X|Z)] + var_Z[E_X[X|Z]] \tag{6}$$

After substitution, the theoretical value can be obtained:

$$\begin{aligned} E[X_t] &= E_{S_t}[E_X[X|S_t]] \\ &= P(S_t = 1) * \mu_1 + P(S_t = 2) * \mu_2 \\ &= 0.75 * 0 + 0.25 * 3 = 0.75 \end{aligned} \tag{7}$$

$$
\begin{aligned}
var(X) &= E_{S_t}[var_X(X|S_t)] + var_{S_t}[E_X[X|S_t]] \\
&= \sum_{j=1}^{2} P(S_t = j) * \sigma_j^2 + \sum_{j=1}^{2} P(S_t = j) * (\mu_j - E[X_t])^2 \\
&= 0.75 * 1 + 0.25 * 4 + 0.75 * (0 - 0.75)^2 \\
&+ 0.25 * (3 - 0.75)^2 = 3.4375
\end{aligned} \quad (8)
$$

Then, the practical mean and variance are computed based the output sequence generated by our HHM.The results are given in Table II. There are difference between theoretical and practical value, because the samples are not large enough. If we increase the size of output samples, the difference will become smaller.

TABLE II
SIMULATION RESULTS

Characteristic Parameter	Theoretical value	Simulation value
mean	0.75	0.7772
variance	3.4375	3.5144

D. Question 4

In this subsection, we are required to plot a series of 500 contiguous samples X_t from the HMM. The result is shown in Fig.3. In this case, We can classify X_t by its amplitude. We know output sequence $b_1(x)$ is Gaussian distribution with mean μ_1=0 and standard deviation $\sigma_1 = 1$, and $b_2(x)$ is a similar distribution with mean $\mu_2 = 3$ and standard deviation σ_2 = 2. The Standard Deviation is a measure of how spread out numbers are. Therefore, signal whose amplitude approximately lie in range$[-1, 1]$ and $[1, 5]$ are classified to b_1 and b_2 respectively. Moreover, it is also obvious that points from the interval$[-1, 1]$ have higher probability than points from the other one, around 0.75.

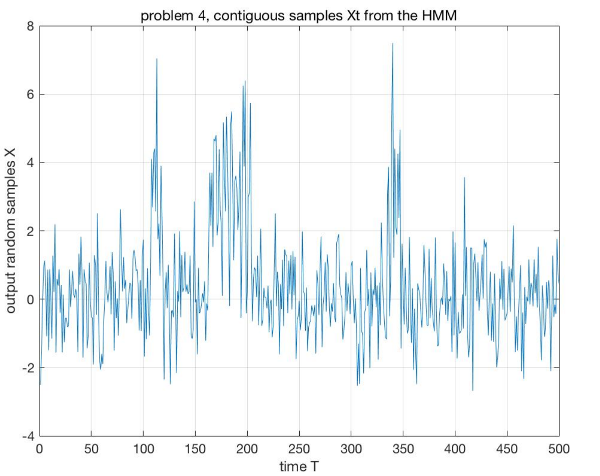

Fig. 1. Simulation result of 500 contiguous samples X_t from the HMM,.

E. Question 5

After changing μ_2 to 0, others remain unchanged, we obtain new simulation Fig.4 of 500 contiguous samples sequence X_t. As we can see from the Fig.1 and Fig.2, the shape that output points spreads out remain the same, because the identical variance keeps the spread-out unchanged. However, it is impossible to estimate the state sequence S_t of the underlying Markov chain from the observed output variables X_t now, since the mean of two Gaussian distribution are the same. As a consequence, for output point in the amplitude range $[-1, 1]$, it is hard to tell which state it belongs to.

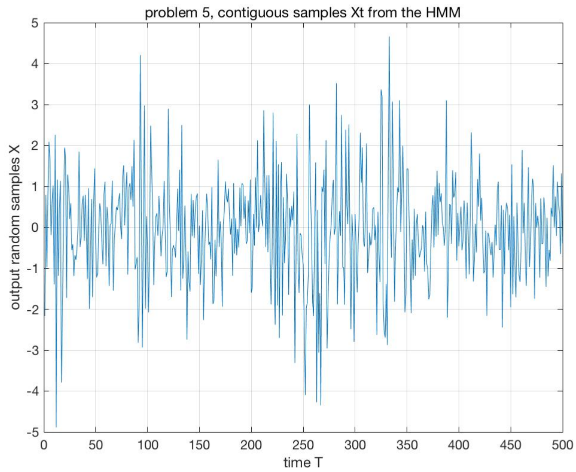

Fig. 2. Simulation result of 500 contiguous samples X_t from the HMM after modifying (μ_1=μ_2=0).

F. Question 6

To model sequences with finite duration, we must introduce a special exit state. In order to test our code also works for finite-durations HMM, the probability transition matrix needs to be modified while the initial state keeps the same. We change the original, square transition probability matrix A in (2) into the following matrix.

$$
\hat{A} = \begin{bmatrix} 0.97 & 0.02 & 0.01 \\ 0.03 & 0.95 & 0.02 \end{bmatrix} \quad (9)
$$

Then, we get the result:

$$
'reach\ the\ exit\ state\ at\ time\ 43,\ the\ HMM\ is\ finite,
$$

$$
the\ length\ of\ output\ sequences\ is\ 42'
$$

which means our rand-function is also feasible for finite-duration HMM. This length is reasonable, since the length is smaller than our set data length.

G. Question 7

Finally, we are supposed to verify our rand-function should work also when the state-conditional output distributions generate random vectors. Here we use 2-dimension GAUSSIAN distribution instead of 1-dimension. We still use the given

GaussD function but the input parameters, i.e. mean and covariance, are matrix. we define three different 2-dimension Gaussian distribution. The mean vectors μ_1,μ_2,μ_3 and the covariance matrix C_1,C_2,C_3 are shown in equation (10)(11)(12).

$$\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, C_1 = \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} \tag{10}$$

$$\mu_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, C_2 = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \tag{11}$$

$$\mu_3 = \begin{bmatrix} 5 \\ 5 \end{bmatrix}, C_3 = \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} \tag{12}$$

Simulation result is demonstrated in Fig.3 and Fig.4, which verify our HHM also works for vector output distribution. In the two figures below, the red points are corresponding mean points.

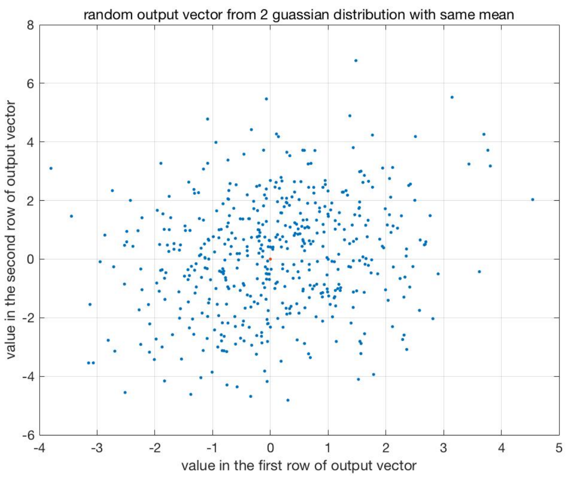

Fig. 3. Random output vector of two Gaussian distribution with same mean and different covariance N(μ_1,C_1), N(μ_2,C_2)

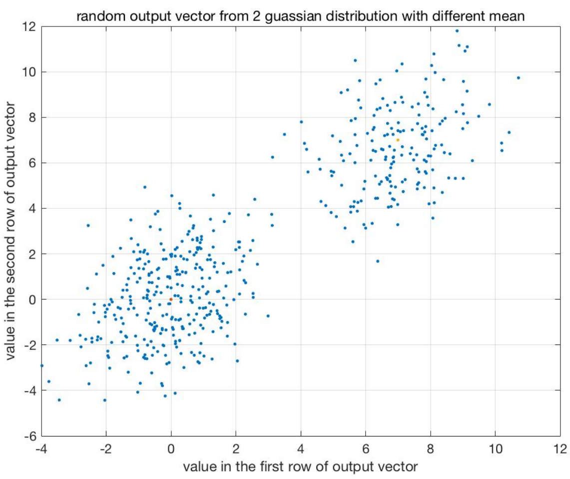

Fig. 4. Random output vector of two Gaussian distribution with different mean and same covariance N(μ_1,C_1), N(μ_3,C_3)

IV. CONCLUSION

To sum up, we verify MATLAB methods of generating an output sequence X_t from an HMM with Gaussian output distributions through seven questions above.

REFERENCES

[1] Arne Leigon and Gustav Eje Henter. Pattern Recognition Fundamental Theory and Exercise Problems