Asymmetric Hidden Markov Models with Continuous Variables

Carlos Puerto-Santana(✉), Concha Bielza, and Pedro Larrañaga

Technical University of Madrid, Madrid, Spain
ce.puerto@alumnos.upm.es, {mcbielza,pedro.larranaga}@fi.upm.es

Abstract. Hidden Markov models have been successfully applied to model signals and dynamic data. However, when dealing with many variables, traditional hidden Markov models do not take into account asymmetric dependencies, leading to models with overfitting and poor problem insight. To deal with the previous problem, asymmetric hidden Markov models were recently proposed, whose emission probabilities are modified to follow a state-dependent graphical model. However, only discrete models have been developed. In this paper we introduce asymmetric hidden Markov models with continuous variables using state-dependent linear Gaussian Bayesian networks. We propose a parameter and structure learning algorithm for this new model. We run experiments with real data from bearing vibration. Since vibrational data is continuous, with the proposed model we can avoid any variable discretization step and perform learning and inference in an asymmetric information frame.

1 Introduction

Hidden Markov Models (HMMs) have been used to predict and analyse dynamic and sequential data, e.g., in speech recognition or gene prediction. These models assume a hidden variable which explains an observed variable. However, they rely on the assumption of an equal emission probability function for every state (except for changes in parameters) of the hidden variable, which for the case of multiple observable variables may lead to a huge unnecessary amount of parameters to be learned and produce models with data overfitting and poor problem insights, specially when few data is available.

Many attempts have tried to capture asymmetries within probabilistic graphical models. For example, Bayesian multinets [5] describe different local graphical models depending on the values of certain observed variables; similarity networks [7] allow to build independent influence diagrams for subsets of a given domain. Context-specific independence in Bayesian networks [1] have tree structured conditional probability distributions with a d-separation-based algorithm

to determine statistical dependencies between variables according to contexts. It has been shown that the use of these asymmetries within the model can improve the inference and learning procedures [2].

In HMMs, which can be seen as probabilistic graphical models, asymmetries could be emulated to be as those in Bayesian multinets, similarity networks or context-specific independence in Bayesian networks. A Chow-Liu tree or a conditional forest is used in [9] to model the observed variables given the hidden state. More recently asymmetric hidden Markov models (As-HMMs) are proposed in [2], where a local graphical model (not necessarily a tree or a forest) is associated to each state of the hidden variable. However, only models with discrete observable variables were discussed, leaving continuous observable variables forced to be discretized. The number of parameters depends upon the discretization method which can affect the inference and learning phases. In this paper we extend As-HMMs to deal with continuous variables, which permits avoiding discretization steps and the errors that may come from this process.

The structure of this document is the following. Section 2 recalls HMMs in a general way. Section 3 covers As-HMMs and introduce the asymmetric linear Gaussian hidden Markov models (AsLG-HMMs) which are capable of modelling As-HMMs with continuous variables. We also discuss the parameter and structure learning of AsLG-HMMs. Section 4 presents experiments with real vibrational data from bearings. The results obtained using AsLG-HMM are compared against the ones from using mixtures of Gaussian hidden Markov models (HMM-MoG). The paper is sounded off in Sect. 5 with conclusions and comments on possible future research lines.

2 Hidden Markov Models

Let $\mathbf{X}^T = (\mathbf{x}^0, ..., \mathbf{x}^T)$ be the observed variables $\mathbf{X} = \{X_1, ..., X_M\}$ over time, where $\mathbf{x}^t = (x_1^t, ..., x_M^t)$, $t = 0, 1, ..., T$. We assume that the observed variable $\mathbf{X}$ is influenced by a discrete variable Q which is hidden and has N possible values, i.e. $\text{dom}(Q) = \{1, 2, ..., N\}$. A HMM is a double chain stochastic process, where the evolution of hidden states $\mathbf{Q}^T = (q^0, ..., q^T)$, of Q is assumed to fulfill the Markov property. Moreover $\mathbf{x}^t$ is assumed to be independent of itself over time given q^t.

Definition 1. *A HMM is a triplet* $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\pi})$ *where* $\boldsymbol{A} = [a_{i,j}]_{i,j=1}^N$ *is a matrix representing the transition probabilities between the hidden states* i, j *over time* t, *i.e.* $a_{i,j} = P(q^{t+1} = j | q^t = i)$; $\boldsymbol{B}$ *is a vector representing the emission probability of the observations given the hidden state,* $\boldsymbol{B} = [b_j(\mathbf{x}^t)]_{j=1}^N$, *where* $b_j(\mathbf{x}^t) = P(\boldsymbol{X} = \mathbf{x}^t | q^t = j)$ *is a probability density function; and* $\boldsymbol{\pi}$ *is the initial distribution of the hidden states* $\boldsymbol{\pi} = [\pi_j]_{j=1}^N$ *where* $\pi_j = P(q^0 = j)$.

Given a model $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$, it is possible to compute the probability of the complete information i.e., $P(\mathbf{Q}^T, \mathbf{X}^T | \lambda)$ as:

$$P(\mathbf{Q}^T, \mathbf{X}^T | \lambda) = \pi_{q^0} \prod_{t=0}^{T-1} a_{q^t, q^{t+1}} \prod_{t=0}^{T} b_{q^t}(\mathbf{x}^t). \tag{1}$$

Three main tasks can be performed in the context of HMMs: first, compute the likelihood of $\mathbf{X}^T$, i.e. $P(\mathbf{X}^T|\lambda)$, which can be done using the forward-backward algorithm [12]. Second, compute the most probable sequence of states i.e., find the value of $\delta_t(i) = \max_{Q^{t-1}} P(\mathbf{X}^t, \mathbf{Q}^{t-1}, q^t = i|\lambda)$, $t = 0, ...T$, $i = 1, ..., N$, which can be solved using the Viterbi algorithm [12]. Third, learn the parameters λ, which can be done with the expectation maximization (EM) algorithm [12].

To execute the forward-backward algorithm, the forward and backward variables are needed, which are defined respectively as $\alpha_t(i) = P(q^t = i, \mathbf{x}^0, ..., \mathbf{x}^t|\lambda)$, $i = 1, .., N$, $t = 0, ..., T$, and $\beta_t(i) = P(\mathbf{x}^{t+1}, ..., \mathbf{x}^T|q^t = i, \lambda)$, $i = 1, ..., N$, $t = 0, ..., T$. Observe in particular that the forward variable can help us to compute the likelihood of $\mathbf{X}^T$, since: $P(\mathbf{X}^T|\lambda) = \sum_{i=1}^N P(\mathbf{X}^T, q^T = i|\lambda) = \sum_{i=1}^N \alpha_T(i)$.

For the second problem, it is noticeable that the variable $\delta_t(i)$ can be seen as: $\delta_t(j) = \max_{i=1,..N}\{\delta_{t-1}(i)a_{i,j}\}b_j(\mathbf{x}^{t+1}), j = 1, .., N, t = 0, ..., T$, which can be solved recursively. For more details about this algorithm, see [12].

For the third problem, we will recall how to learn the parameters λ^* using the Baum-Welch method or equivalently, the EM algorithm [12]. The algorithm consists of two steps, the expectation (E) step and the maximization (M) step. We assume that a prior $\lambda^0 = (\mathbf{A}^0, \mathbf{B}^0, \boldsymbol{\pi}^0)$ is known. In the E step, we compute the distribution of the latent variable Q, i.e., $P(\mathbf{Q}^T|\mathbf{X}^T, \lambda^0)$. In the M step, we find λ^* as solution of the problem: $\lambda^* = \arg\max_\lambda H(\lambda|\lambda^0)$, where $H(\lambda|\lambda^0)$ is defined as

$$H(\lambda|\lambda^0) = \sum_{\mathcal{Q}} P(\mathbf{Q}^T|\mathbf{X}^T, \lambda^0) \ln P(\mathbf{Q}^T, \mathbf{X}^T|\lambda), \tag{2}$$

where $\mathcal{Q} := \text{dom}(Q^T)$. In [3] it is shown that $P(\mathbf{X}^T|\lambda^*) \geq P(\mathbf{X}^T|\lambda^0)$ with equality if and only if $H(\lambda^*|\lambda^0) = H(\lambda^0|\lambda^0)$ and $P(\mathbf{Q}^T|\mathbf{X}^T, \lambda^0) = P(\mathbf{Q}^T|\mathbf{X}^T, \lambda^*)$. Therefore, iterating the E and the M steps produce improvements in the model likelihood.

The E step for HMMs, for a prior λ^0, is done by calculating the quantities $\gamma_t(i) = P(q^t = i|\mathbf{X}^T, \lambda^0)$ and $\xi_t(i, j) = P(q^t = i, q^{t+1} = j|\mathbf{X}^T, \lambda^0)$, $i, j = 1, ..., N$, $t = 0, ..., T$ which can be computed using the forward and backward variables in the following way:

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{u=1}^N \alpha_t(u)\beta_t(u)}, \tag{3}$$

$$\xi_t(i, j) = \frac{\alpha_t(i)a_{ij}b_j(\mathbf{x}^{t+1})\beta_{t+1}(j)}{\sum_{u=1}^N \sum_{v=1}^N \alpha_t(u)a_{uv}b_v(\mathbf{x}^{t+1})\beta_{t+1}(v)}. \tag{4}$$

The M step for HMM requires to maximize the $H(\lambda|\lambda^0)$ function, which is done by using Eqs. (1) and (2):

$$\begin{aligned} H(\lambda|\lambda^0) = \sum_{\mathcal{Q}} P(\mathbf{Q}^T|\mathbf{X}^T, \lambda) \ln(\pi_{q^0}) + \sum_{\mathcal{Q}} \sum_{t=0}^{T-1} P(\mathbf{Q}^T|\mathbf{X}^T, \lambda) \ln(a_{q^t, q^{t+1}}) \\ + \sum_{\mathcal{Q}} \sum_{t=0}^{T} P(\mathbf{Q}^T|\mathbf{X}^T, \lambda) \ln(b_{q^t}(\mathbf{x}^t)), \end{aligned} \tag{5}$$

with restrictions $\sum_{i=1}^{N} \pi_i = 1$ and $\sum_{j=1}^{N} a_{i,j} = 1$, $i = 1, ..., N$. The updating formulas for parameters $\mathbf{A}$ and $\boldsymbol{\pi}$ can be computed using Lagrange multipliers. The resulting formulas are:

$$\pi_i^*(i) = \gamma_0(i), \quad i = 1, ..., N, \qquad a_{i,j}^* = \frac{\sum_{t=0}^{T-1} \xi_t(i,j)}{\sum_{t=0}^{T-1} \gamma_t(i)}, \quad i, j = 1, ..., N. \tag{6}$$

The updating formula for parameter $\mathbf{B}$ relies on the assumptions made over the observable variables and the emission probabilities. In the next section we develop the formulas to update parameter $\mathbf{B}$ in the context of an As-HMM with Gaussian variables.

3 Asymmetric Linear Gaussian Hidden Markov Models

In this section we recall definitions and relevant aspects from As-HMMs mentioned in [2]. Then we model the asymmetric emission probabilities using linear Gaussian Bayesian networks and deduce the update algorithm for parameter $\mathbf{B}$ and discuss its complexity. Finally, we describe the structure learning procedure.

3.1 Definitions

First of all, we recall the definition of linear Gaussian Bayesian network (LGBN), see [10]. This model will be used to describe the asymmetries in HMMs with continuous variables.

Definition 2. *Let $\boldsymbol{X} = (X_1, ..., X_M)$ be a continuous random variable. A linear Gaussian Bayesian network over $\boldsymbol{X}$ is a tuple $\mathcal{B}(R, G)$, where G is a directed acyclic graph (DAG). The parents of X_m are given by G and are represented by an ordered vector of length k_m denoted as $\boldsymbol{Pa}(X_m) = (U_{m,1}, ..., U_{m,k_m})$, $m = 1, ..., M$, with $U_{m,l} \in \boldsymbol{X}$, $l = 1, ..., k_m$. R is formed by the local distribution of each X_m conditioned on $\boldsymbol{Pa}(X_m)$. The joint density function satisfies:*

$$P(\boldsymbol{X}) = \prod_{m=1}^{M} \mathcal{N}(X_m | \beta_{m,0} + \beta_{m,1} U_{m,1} + \cdots + \beta_{m,k_m} U_{m,k_m}, \sigma_m^2), \tag{7}$$

where $\mathcal{N}$ denotes the one-dimensional Gaussian probability density function and β_i, k, $i = 1, ..., N$, $k = 0, ..., k_m$ are real numbers.

Now, state-specific Bayesian network, As-HMMs and AsLG-HMMs are defined. The idea is to give a distinct LGBN to every state of the hidden variable. As a consequence, in every state the parents of each variable are different. This representation captures asymmetries in the data.

Definition 3. *Let $\boldsymbol{X} = (X_1, ..., X_M)$ and Q be random variables. For each $q \in dom(Q)$, we associate a Bayesian network over $\boldsymbol{X}$ called state-specific Bayesian network for q, $\mathcal{B}_q(R_q, G_q)$. We define the following conditional distribution:*

$$P_q(\boldsymbol{X}) := P(\boldsymbol{X}|q) = \prod_{m=1}^{M} P_q(X_m | \boldsymbol{Pa}_q(X_m)). \tag{8}$$

Definition 4. *An asymmetric hidden Markov model over the random variables* $(\boldsymbol{X}, Q)$*, being* Q *the hidden variable, is a model* $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\pi})$ *with initial distribution* $\boldsymbol{\pi} = [\pi_j]_{j=1}^N$ *where* $\pi_j = P(q^0 = j)$*, transition probabilities between the hidden states* $\boldsymbol{A} = [a_{i,j}]_{i,j=1}^N$ *where* $a_{i,j} = P(q^{t+1} = j | q^t = i)$ *and the emission density function vector* $\boldsymbol{B} = [b_j(\mathbf{x}^t)]_{j=1}^N$ *where* $b_j(\mathbf{x}^t) = P_j(\mathbf{x}^t)$ *i.e. a state-specific Bayesian network.*

Definition 5. *An asymmetric linear Gaussian hidden Markov model over* $\boldsymbol{X} = (X_1, ..., X_M)$ *the continuous random variables and* Q *the hidden discrete random variable, is an As-HMM* $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\pi})$ *with the property that for each* $q \in dom(Q)$ *a state-specific linear Gaussian Bayesian network* $\mathcal{B}_q(R_q, G_q)$ *is associated. If the parents of variable* X_m *for the state* q *are an ordered column vector of length* k_m^q *denoted as* $\boldsymbol{Pa}_q(X_m) = (U_{m,1}^q, ..., U_{m,k_m^q}^q)$*,* $m = 1, ..., M$*, with* $U_{m,l}^q \in \boldsymbol{X}$*,* $l = 1, ..., k_m^q$*, then the emission probabilities* $\boldsymbol{B} = [b_j(\mathbf{x}^t)]_{j=1}^N$ *have the form:*

$$b_j(\mathbf{x}^t) = P_j(\mathbf{x}^t) = \prod_{m=1}^{M} \mathcal{N}\big(x_m^t | \beta_{m,0}^j + \beta_{m,1}^j U_{m,1}^j + \cdots + \beta_{m,k_m^j}^j U_{m,k_m^j}^j, (\sigma_m^j)^2\big), \quad (9)$$

Observe that each linear Gaussian model for each state $q \in \mathrm{dom}(Q)$ is determined by the set of coefficients $\mathcal{T}_q := \bigcup_{m=1}^M \{\beta_{m,0}^q, ..., \beta_{m,k_m^q}^q\}$, since the mean of each variable is a function of these coefficients, see [10].

3.2 Learning Parameters

Now that we know how to represent the emission probabilities $\mathbf{B}$ for the case of AsLG-HMMs, we build the parameter update formulas. Assume the prior λ^0 is known and we execute the E step as in Sect. 2, therefore $\gamma_t(i)$, $i = 1, ..., N$, $t = 0, 1, ..., T$ quantities are defined. Let $D_q[F] := \sum_{t=0}^T f^t \gamma_t(q)$, with F any variable and $q \in \mathrm{dom}(Q)$. First, we need the value of $\beta_{m,0}^q$ that maximizes the H function, hence we derive Eq. (5) with respect to $\beta_{m,0}^q$ and equate to zero. Observe that $\beta_{m,0}^q$ appears in the H function inside $b_{q^t}(\mathbf{x}^t) = P_{q^t}(\mathbf{x}^t)$, hence:

$$\frac{\partial H(\lambda|\lambda^0)}{\partial \beta_{m,0}^q} = \frac{\partial \sum_{t=0}^T \sum_{i=1}^N \gamma_t(i) \ln P_i(\mathbf{x}^t)}{\partial \beta_{m,0}^q} = \sum_{t=0}^T \gamma_t(q) \frac{\partial \ln P_q(\mathbf{x}^t)}{\partial \beta_{m,0}^q} = 0, \quad (10)$$

making the derivation assuming that $P_q(\mathbf{x}^t)$ is defined by an AsLG-HMMs and $u_{m,l}^{t,q}$ is the value of the l-parent of X_m for state q at time t, $l = 1, .., k_m^q$. Then we have:

$$\frac{\partial H(\lambda|\lambda^0)}{\partial \beta_{m,0}^q} = \sum_{t=0}^T -2 \frac{\gamma_t(q)}{(\sigma_m^q)^2} (\beta_{m,0}^q + \beta_{m,1}^q u_{m,1}^{t,q} + \cdots + \beta_{m,k_m^q}^q u_{m,k_m^q}^{t,q} - x_m^t) = 0. \quad (11)$$

This leads to the following expression:

$$D_q[X_m] = \beta_{m,0}^q D_q[1] + \beta_{m,1}^q D_q[U_{m,1}^q] + \cdots + \beta_{m,k_m^q}^q D_q[U_{m,k_m^q}^q]. \quad (12)$$

If Eq. (5) is derived with respect to the coefficients $\beta^n_{m,k}$ with $k = 1, ..., k^q_m$ as in Eq. (10) we obtain the following equations:

$$\begin{cases} D_q[X_m U^q_{m,1}] = \beta^q_{m,0} D_q[U^q_{m,1}] & + \cdots + \beta^q_{m,k^q_m} D_q[U^q_{m,k^q_m} U^q_{m,1}] \\ \vdots & \vdots \\ D_q[X_m U^q_{m,k^q_m}] = \beta^q_{m,0} D_q[U^q_{m,k^q_m}] & + \cdots + \beta^q_{m,k^q_m} D_q[(U^q_{m,k^q_m})^2]. \end{cases} \tag{13}$$

Equations (12) and (13) form a linear system of $k^q_m + 1$ unknowns with $k^q_m + 1$ equations. The solution of this system gives the coefficients $\{\beta^q_{m,0}, \beta^q_{m,1}, ..., \beta^q_{m,k^q_m}\}$, for each variable $m = 1, 2, ..., M$ and state $q \in \text{dom}(Q)$. Once, these coefficients are known, the mean $\mu^{t,q}_m = \beta^q_{m,0} + \beta^q_{m,1} u^{t,q}_{m,1} + \cdots + \beta^q_{m,k^q_m} u^{t,q}_{m,k^q_m}$ is estimated. To obtain the updating formula of $(\sigma^q_m)^2$, we must derive Eq. (5) with respect $(\sigma^q_m)^2$ and equate to zero:

$$\frac{\partial H(\lambda|\lambda^0)}{\partial (\sigma^q_m)^2} = \sum_{t=0}^{T} \gamma_t(q) \frac{\partial \ln P_q(\mathbf{x}^t)}{\partial (\sigma^q_m)^2} = 0. \tag{14}$$

Assuming that $P_q(\mathbf{x}^t)$ is defined by an AsLG-HMM, we have:

$$\frac{\partial H(\lambda|\lambda^0)}{\partial (\sigma^q_m)^2} = \sum_{t=0}^{T} \gamma_t(q) \Big(\frac{(x^t_m - \mu^{t,q}_m)^2}{(\sigma^q_m)^4} - \frac{1}{(\sigma^q_m)^2} \Big) = 0, \tag{15}$$

which leads to the following expression:

$$((\sigma^q_m)^2)^* = \frac{\sum_{t=0}^{T} \gamma_t(q)(x^t_m - \mu^{t,q}_m)^2}{\sum_{t=0}^{T} \gamma_t(q)}. \tag{16}$$

We discuss now the complexity of computing the $\mathcal{T} := \bigcup_{i=1}^{N} \mathcal{T}_i$ coefficients. Assume that we have N states, M variables and that the factorization for each state is the most complex i.e., every variable is dependent of the others. This implies that $|\mathcal{T}_i| = 1 + 2 + 3 + \cdots + M = \frac{M(M+1)}{2}$, $i = 1, ..., N$, therefore $|\bigcup_{i=1}^{N} \mathcal{T}_i| = \frac{NM(M+1)}{2}$. It is known that the complexity of solving a linear system of k variables is at most $O(k^3)$ (using for example Gauss-Jordan algorithm). Hence for the worst case scenario the complexity of determining the coefficients for a single state is $O(1^3) + O(2^3) + \cdots + O(M^3) = O(\frac{M^2(M+1)^2}{4})$. Therefore to compute the coefficients for every state, the complexity is $O(NM^2(M+1)^2)$. On the other hand, for the simplest factorization i.e., every variable is independent of the others given the state, the complexity of determining the coefficients is $O(NM)$, since we must solve M linear systems of one variable for N states.

3.3 Learning Structure

For the structure learning task the SEM algorithm, proposed in [4] is used. Assume the prior model $\lambda^{0,0} = (\mathbf{A}^{0,0}, \mathbf{B}^{0,0}, \boldsymbol{\pi}^{0,0})$ for the initial model M^0. One

SEM iteration consists of using the EM algorithm to get the parameters $\lambda^{0,*} = (\mathbf{A}^{0,*}, \mathbf{B}^{0,*}, \boldsymbol{\pi}^{0,*})$. Next, using the estimation $\lambda^{0,*}$ and $P(\mathbf{Q}^T|\mathbf{X}^T, \lambda^{0,*})$ got in the E step, we look for a model M^1 such that maximizes a given score function, usually the Bayesian information criterion (BIC). Once the model M^1 has been found, we set $\lambda^{1,0} := \lambda^{0,*}$ i.e., the found parameters are used as prior parameters for the next iteration of the SEM algorithm. As noticed in [2] the BIC score can be deduced and reduced from Eq. (5) as follows:

$$\text{Score} = \sum_{q=1}^{N} \sum_{t=0}^{T} \gamma_t(q) \ln P_q(\mathbf{x}^t) - \frac{1}{2} \ln(T) \#(\mathcal{B}_q(R_q, G_q)), \tag{17}$$

where $\#(\mathcal{B}_q(R_q, G_q))$ is the number of parameters used for the state-specific linear Gaussian Bayesian network for state q. We must also mention that any algorithm can be used to optimize the score function. In particular, the simulated annealing introduced in [8] was used for this study. Recall that [6] proved the convergence of this method which gives us an ending guarantee of the optimization process.

4 Experiments

For the experiments we use a real dataset. The data comes from bearing vibrational information, see [11]. The data is filtered using spectral kurtosis algorithms and envelope techniques as in [14]; next, the bearing fundamental frequencies and its harmonics are extracted: ball pass frequency outer (BPFO) related to the bearings outer race, ball pass frequency inner (BPFI) related to the bearings inner race, ball spin frequency (BSF) related to the bearings rollers and the fundamental train frequency (FTF) related to the bearings cage. The mechanical set-up is shown in Fig. 1. In real life applications, bearings are fundamental components inside of tool machines. Is desirable to surveillance the bearing health state. However, the health state is a hidden variable; hence, HMMs can be applied to estimate the bearing health. In the literature, the health estimation is usually done with mixtures of Gaussian hidden Markov model (MoG-HMM) as in [13].

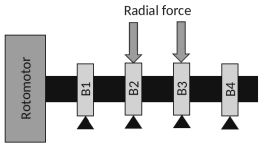
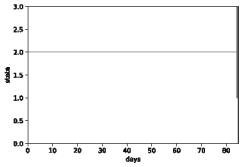


Fig. 1. Graphical representation of the mechanical set-up. A rotomotor spins a shaft at a rotational speed of 2000RPM coupled with four Rexnord ZA-2115 double row bearings with labels B1, B2, B3 and B4. A constant radial load of 2721.554 kg is applied to bearings 2 and 3. Vibrational data is recorded until one of the bearings fails. A signal record of 0.1s is taken every twenty minutes. The sampling rate is 20 kHz.

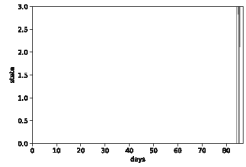
There is a training and a testing signal. The learning signal consist of 2156 records and the testing signal of 6324 records. We have information of the fundamental frequencies and three harmonics of each frequency, hence 16 variables are used in both signals. We will assume that there are four possible health states. In the training dataset, B3 fails due to its inner race and B4 due to its rollers. In the testing dataset B3 fails due to its outer race. The results of using AsLG-HMMs and MoG-HMMs with three mixtures are shown in Table 1. We show results for log-likelihood (LL), BIC score, and number of parameters (#). Notice that the number of parameters needed by a MoG-HMM is $NK((M^2+M)/2+1)$, where K is the number of mixtures components.

Table 1. Likelihood and BIC results for test signal.

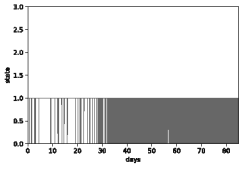
	MoG-HMM			AsLG-HMM		
B	#	LL	BIC	#	LL	BIC
1	1644.0	−170045.02	−177232.72	560.0	−162654.42	**−165028.46**
2	1644.0	−204349.46	−211537.16	560.0	−178040.19	**−180414.23**
3	1644.0	−349099.49	−356287.2	137.0	−270698.52	**−271226.57**
4	1644.0	−84479.32	−91667.03	133.0	−74495.96	**−75006.56**



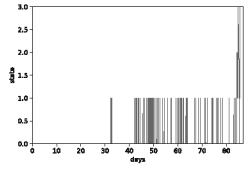
(a) State sequence MoG-HMM B3.



(b) State sequence AsLG-HMM B3.



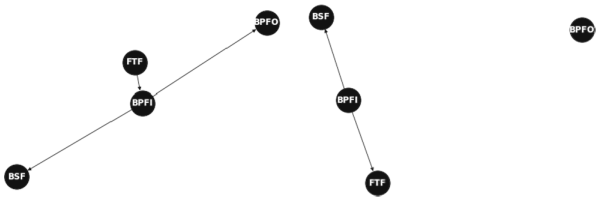
(c) State sequence MoG-HMM B4.



(d) State sequence AsLG-HMM B4.

Fig. 2. Sequences of states predicted by Viterbi algorithm for B3 and B4 due to B3 is the failure bearing and B4 has the lowest BIC score. (a) and (c) are state sequences predicted with MoG-HMMS. (b) and (d) are state sequences predicted with AsLG-HMMs.

From the results obtained, it can be seen that the BIC scores from AsLG-HMMs are better than the ones obtained from MoG-HMMs. Also, if we observe Fig. 2, we see that the health evolution of the B3 and B4 predicted by the MoG-HMMs are not easy to read. In (a) at the end of the bearings life the sequence



(a) LGBN for B3, state 0. (b) LGBN for B3, state 3.

Fig. 3. Different state specific Gaussian Bayesian network structures obtained for different states. Here, an illustrative model structure is built using only fundamental frequencies.

jumps between all the states and in (c) any relevant information is shown. On the other hand, the state sequence predicted by AsLG-HMMs reveals a change in the bearings health in its last days of life in (c) and in (d) shows an evolutionary sequence.

On the other hand, to illustrate the better problem insight that As-HMMs provide, we train an AsLG-HMMs with only the fundamental frequencies (four variables) and observe the obtained state-specific LGBN for B3, see Fig. 3. As we see for state 0 (healthy state), FTF frequency determines the others, this was expected since FTF is close to the shaft frequency. Meanwhile in state 3 (failure state), the BPFI determine the BSF and FTF frequencies, which may indicate a failure in the bearings inner race.

5 Conclusions and Future Work

In this paper the AsLG-HMM has been introduced in order to deal with continuous variables in asymmetric hidden Markov models. This model is proposed to overcome overfitting models and discretization steps. Also AsLG-HMM provides useful interpretation of the problem domain, since state-specific LGBN are used. Also As-HMMs open a wide range of research lines, there are many possibilities and variations of As-HMMs that can be explored.

Acknowledgements. This work has been partially supported by the Spanish Ministry of Economy and Competitiveness through TIN2016-79684-P projects, by the Regional Government of Madrid through the S2013/ICE-2845-CASI-CAM-CM project, and by Fundación BBVA grants to Scientific Research Teams in Big Data 2016. Additionally, we give thanks to Etxe-Tar S.A. to provide the filtered datasets to perform the corresponding experiments.

References

1. Boutilier, C., Friedman, N., Goldszmidt, M., Koller, D.: Context-specific independence in Bayesian networks. In: Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence, pp. 115–123. Morgan Kaufmann Publishers Inc., Burlington (1996)

2. Bueno, M.L., Hommersom, A., Lucas, P.J., Linard, A.: Asymmetric hidden Markov models. Int. J. Approx. Reason. **88**, 169–191 (2017)
3. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B (Methodological) **39**(1), 1–38 (1977)
4. Friedman, N.: The Bayesian structural EM algorithm. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, pp. 129–138. Morgan Kaufmann Publishers Inc., San Francisco (1998)
5. Geiger, D., Heckerman, D.: Knowledge representation and inference in similarity networks and Bayesian multinets. Artif. Intell. **82**(1), 45–74 (1996)
6. Granville, V., Krivanek, M., Rasson, J.P.: Simulated annealing: a proof of convergence. IEEE Trans. Pattern Anal. Mach. Intell. **16**(6), 652–656 (1994)
7. Heckerman, D.: Probabilistic similarity networks. Networks **20**(5), 607–636 (1990)
8. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
9. Kirshner, S., Padhraic, S., Andrew, R.: Conditional Chow-Liu tree structures for modeling discrete-valued vector time series. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, pp. 317–324. AUAI Press (2004)
10. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. The MIT Press, Cambridge (2009)
11. Qian, Y., Yan, R., Gao, R.X.: A multi-time scale approach to remaining useful life prediction in rolling bearing. Mech. Syst. Signal Process. **83**, 549–567 (2017)
12. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. In: Readings in Speech Recognition, pp. 267–296. Morgan Kaufmann, San Francisco (1990)
13. Tobon, D., Medjaher, K., Zerhouni, N., Tripot, G.: A data-driven failure prognostics method based on mixture of Gaussians hidden Markov models. IEEE Trans. Reliab. **61**(2), 491–503 (2012)
14. Wang, Y., Liang, M.: An adaptive SK technique and its application for fault detection of rolling element bearings. Mech. Syst. Signal Process. **25**, 1750–1764 (2010)