Feature Saliencies in Asymmetric Hidden Markov Models

Carlos Puerto-Santana, Pedro Larrañaga, *Member, IEEE*, and Concha Bielza, *Member, IEEE*

***Abstract*— Many real-life problems are stated as nonlabeled high-dimensional data. Current strategies to select features are mainly focused on labeled data, which reduces the options to select relevant features for unsupervised problems, such as clustering. Recently, feature saliency models have been introduced and developed as clustering models to select and detect relevant variables/features as the model is learned. Usually, these models assume that all variables are independent, which narrows their applicability. This article introduces asymmetric hidden Markov models with feature saliencies, i.e., models capable of simultaneously determining during their learning phase relevant variables/features and probabilistic relationships between variables. The proposed models are compared with other state-of-the-art approaches using synthetic data and real data related to grammatical face videos and wear in ball bearings. We show that the proposed models have better or equal fitness than other state-of-the-art models and provide further data insights.**

***Index Terms*— Asymmetric information, Bayesian networks, embedded feature selection, feature saliencies, feature subset selection (FSS), hidden Markov models (HMMs).**

I. Introduction

FEATURE subset selection (FSS) [1] has become a relevant tool for data scientists to recognize relevant information within large datasets and reduce data dimensionality. However, little effort has been put into the case where no class variable is present. In this manner, many real unsupervised problems are neglected, and many clustering models are forced to work with undesirable or irrelevant features. Recently, feature saliency (FS) models [2] have been proposed as an option to overcome the FSS problem in unlabeled data. FS models try to determine the level of relevancy of each variable or feature. In FS models, a feature is considered irrelevant if it is independent of the clustering variable [2]. FS models can be considered as embedded FSS methods: While they learn their parameters, simultaneously, they compute the relevancy of each feature. In this sense, FS models are preferred to wrapper FSS strategies [3], where the model must be learned for each considered subset of variables.

Hidden Markov models (HMMs) are a powerful tool to understand and model dynamic stochastic unlabeled data. They were traditionally used for speech recognition and gene segmentation [4]. However, lately, their applicability to other areas, such as tool wear monitoring, weather forecasting, and so on, has been noted [5]. Traditional HMMs assume full independence or full dependence between the observed variables. In the first case, the resulting models are not applicable to several real-life problems, and in the second case, the model learns unnecessary parameters that can cause model overfitting. In this sense, asymmetric HMMs (As-HMMs) [6] can be used to find an intermediate point where the model discovers and estimates the existing probabilistic dependencies between variables in order to explain the data.

HMMs and FS models are found in previous works [7]–[9]. However, the models assume that all the variables are independent, which depletes the explanatory power of the model. Consequently, in this article, we introduce an FS model based on As-HMMs to alleviate this issue, i.e., we propose an HMM that is capable of simultaneously determining feature relevancy and giving a probabilistic dependency graph (context-specific Bayesian network [10]) containing only the relevant features.

In this article, we propose an FS model to select features in unsupervised dynamic data. Our model relaxes the independent variable assumption of traditional FS models, which depletes their applicability, by instead generating context-specific Bayesian networks for the selected variables. In addition, as expected, the proposed model is capable of performing inference in testing data. However, the model assumes some hypotheses, such as: 1) variables must follow a linear Gaussian distribution; 2) the hidden states follow the Markov property; and 3) the noise variables are Gaussian. Any dynamic process that drifts abruptly from these conditions is not suitable to be interpreted with the proposed model since the data insights that the model would provide would be wrong.

The organization of this article is given as follows. Section II will review some related work with FSS in HMM and asymmetric models. Section III will briefly review the relevant mathematical and probabilistic tools for the development of the model. Section IV will develop the proposed FS model.

Manuscript received 26 December 2021; revised 23 May 2022; accepted 23 July 2022. Date of publication 8 August 2022; date of current version 1 March 2024. This work was supported in part by the Spanish Ministry of Science and Innovation through the PID2019-109247GB-I00 and DSTREAMS project and in part by the H2020 IoTwins projects (Distributed digital twins for industrial SMEs: a big-data platform) funded by the EU through the call ICT-11-2018-2019 under Grant Agreement 857191. *(Corresponding author: Carlos Puerto-Santana.)*

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

Carlos Puerto-Santana is with Aingura IIoT, 20009 Donostia-San Sebastian, Spain, and also with the Departamento de Inteligencia Artificial, E.T.S. Ingenieros Informáticos, Universidad Politécnica de Madrid, 28660 Madrid, Spain (e-mail: ce.puerto@alumnos.upm.es).

Pedro Larrañaga and Concha Bielza are with the Departamento de Inteligencia Artificial, E.T.S. Ingenieros Informáticos, Universidad Politécnica de Madrid, 28660 Madrid, Spain (e-mail: pedro.larranaga@fi.upm.es; mcbielza@fi.upm.es).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2022.3194597, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2022.3194597

TABLE I

TABLE OF REVIEWED ARTICLES BY TOPIC

Topic	Name
Filter	**Supervised**
	CFS [11]
	RELIEF [12]
	Markov blanket and HMM [13]
	AdaBoost and HMM [14]
	Ensemble of filters and HMM [15]
	Unsupervised
	RELIEF for clustering [16]
Wrapper	A GB FSS with HMM classifier [17]
	PSO with HMM classifier [18]
Embedded	**Supervised**
	Lasso regression [19]
	Penalized neural networks [20]
	Reinforcement learning in neural networks [21]
	Clustering
	GMM-FS [2], [22]
	GMM-LFS [23]
	Semi-supervised GMM-LFS [24]
	VB-SMM-LFS [25]
	iGMM-LFS [26]
	HMMs
	FS-HMM [7]
	SHMM-LFS [8]
	Discrete FS-HMM [9]
	FS-AsHMM [This]
Asymmetric models	BMM [27]
	HMM with Chow-Liu trees [28]
	MRF-HMMs [29]
	As-HMM [6]
	AR-AsLG-HMM [5]

Section V will validate the proposed model with synthetic and real data. Finally, Section VI will present conclusions and future work.

II. RELATED WORK

Regarding FSS, the state-of-the-art strategies are usually grouped into three categories: filter, wrapper, and embedded. There are also dimensionality reduction strategies as in [30] or [31]. In these articles, the features are projected into lower dimensionality spaces where the information is concentrated, and these new features are used as predictors. Nevertheless, these kinds of strategies are beyond the scope of this article since we are concerned with the interpretation of the model. In Table I, the reviewed articles are briefly summarized and sorted by topic.

A. Filter Techniques

1) Supervised: Among the traditional filter FSS techniques, we find the correlation-based feature selection (CFS) [11]. CFS was designed to search for a subset of features, which maximizes the feature relevancy with respect to a class variable and minimizes the redundancy between them. RELIEF [12] provided a relevancy score for each feature based on distances between classes. Features whose relevancy overpassed a threshold were selected.

As regards filter algorithms with supervised data related to HMMs, the following articles were found: [13] proposed a sequential data feature selection algorithm based on Markov blankets. Their methodology gradually computed the Markov blanket of a target or class variable using the HITON algorithm [32]. The HITON algorithm was fed with an HMM to learn the corresponding Markov blanket. The variables in the Markov blanket of the class variable were applied as features of a classification model. In [14], for each class value and variable, an HMM, which was used as a classifier, was learned. Next, an AdaBoost algorithm was employed to select which HMMs improved the accuracy of the prediction. Momenzadeh *et al.* [15] coupled discrete HMM with different feature selection filtering scores. An HMM was created using the ranking information obtained by the filters. The resultant emission probabilities were used as a relevancy score.

2) Unsupervised: Only one filter strategy for variable selection was found in the case of unsupervised data. Dash and Ong [16] used the RELIEF algorithm [12] to discriminate features. The authors used the K-medoid algorithm to generate artificial class labels and execute the RELIEF algorithm using these. This process was iterated as many times as the user determined.

B. Wrapper Techniques

In wrapper techniques, only a few works were found. All of which look for the best set of variables to improve the score of a clustering model. For instance, Yue *et al.* [17] used a greedy-backward (GB) FSS algorithm to select the features for an adaptive variable duration mixture of Gaussian HMMs. Farag *et al.* [18] applied the particle swarm optimization (PSO) algorithm [33] to maximize the HMM accuracy. In both cases, heuristic or metaheuristic methods were employed to find the best set of features for an HMM classifier.

C. Embedded Techniques

1) Supervised: We reviewed some embedded techniques for supervised problems. For example, the well-known work of [19], where the lasso regression was introduced, or [20], where a regularization term was added in the learning phase of neural networks in order to determine relevant features. More recently, Mnih *et al.* [21] introduced a convolutional neural network capable of selecting image sections to be processed, but it had to be learned using reinforcement learning.

2) Clustering: In [2], the concept of FS was introduced (as will be explained in Section III) and applied to perform model learning and feature selection simultaneously in clustering models. The feature saliencies were utilized to indicate the level of relevancy of each variable in a Gaussian mixture model (GMM-FS). All the parameters were learned using the EM algorithm. In [22], the GMM-FS was revisited, and a variational Bayesian algorithm was used to estimate the model parameters. Later, Li *et al.* [23] proposed a localized FS model for a mixture of Gaussians (GMM-LFS), where, depending on the mixture component, the set of relevant features could change. The learning process was carried out using the variational Bayesian methodology. Next, Guerra *et al.* [24] proposed a semisupervised GMM-LFS. The aim of the authors was to classify partially labeled data. The authors added cluster-dependent feature saliencies to perform the feature selection procedure. Nguyen *et al.* [25] developed a mixture model (VB-SMM-LFS) where, depending on the cluster instance, feature saliencies indicated which variables were relevant for the cluster. Also, the clusters were expressed with piecewise-t-Student distributions, and the learning of parameters was performed using Bayesian variational methods. Finally, Song *et al.* [26] coupled an infinite components piecewise Gaussian mixture model with localized feature saliencies (iGMM-LFS). The learning phase was performed using a Bayesian method through a Markov chain Monte Carlo algorithm.

3) HMMs: Concerning HMMs, inspired by the FS-GMMs, Adams *et al.* [7] developed an HMM (FS-HMM) where a set of feature saliencies were added to the emission probabilities to determine which variables were relevant for the model. The emission distribution assumed full independence between variables, and a maximum *a posteriori* approach was used to learn the parameter models. In addition, the model was extended to hidden semi-Markov models, where the sojourn times could be modeled in order not to strictly follow a geometric distribution, as in any traditional HMM. Later, Adams and Beling [9] proposed an FS-HMM for discrete features in HMMs. It was assumed that the relevant features followed a state-dependent Poisson distribution, whereas irrelevant features followed a state-independent Poisson distribution. The author provided an EM algorithm and the Bayes algorithm to learn a discrete model. Zheng *et al.* [8] introduced an HMM where the emission probabilities were modeled as mixtures of t-Student distributions (SHMM-LFS). Feature saliencies were added to the model at the component level such that the model was capable of determining, depending on the hidden state and mixture component, in which features were noise or relevant. The learning procedure was performed with variational Bayesian methods.

D. Asymmetric Models

Asymmetric models can be defined as probabilistic models where, depending on the instance of a hidden context variable, the probabilistic relationships between observable variables may change. In the case of HMMs, the context is given by the hidden state, and the emission probabilities change their explanatory graphical model (context-specific Bayesian networks). As will be seen, none of the reviewed articles related to As-HMMs used feature saliencies, which is a gap that this article tries to fill.

Bilmes [27] introduced an autoregressive (AR) HMM called buried Markov models (BMMs), which selected the AR order dependencies for each hidden state using mutual information. Kirshner *et al.* [28] developed an HMM where, depending on the hidden state, a different Chow-Liu tree was expressed to encode the probabilistic relationship between variables. Stadler and Mukherjee [29] proposed a learning algorithm based on the EM to generate sparse precision matrices, i.e., each hidden state had its own sparse precision matrix, which could be interpreted as a Markov random field (MRF).[1] Later, Bueno *et al.* [6] introduced the As-HMMs, created for discrete data. The model expressed the emission probabilities as context-specific Bayesian networks that were learned using a taboo search algorithm. Finally, Puerto-Santana *et al.* [5] developed an As-HMM for continuous variables, which was capable of determining the AR order for each variable depending on the hidden state (AR-AsLG-HMMs).

III. Theoretical Framework

A. Hidden Markov Models

An HMM can be seen as a double chain stochastic model, where a chain is observed, namely, $\boldsymbol{X}^{0:T} = (\boldsymbol{X}^0, \dots, \boldsymbol{X}^T)$, where $\boldsymbol{X}^t = (X_1^t, \dots, X_M^t) \in \mathcal{R}^M$ continuous variables and the other chain is hidden, namely, $\boldsymbol{Q}^{0:T} = (Q^0, \dots, Q^T)$. Here, $T+1$ is the length of the data. Let N be the order of the range of the hidden variables Q^t or the number of hidden states. An HMM can be summarized with the parameter $\boldsymbol{\lambda} = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi}) \in \boldsymbol{\Omega}$, where $\boldsymbol{\Omega}$ denotes the space of all possible parameters [4], $\mathbf{A} = [a_{ij}]_{i,j=1}^N$ is a matrix representing the transition probabilities between hidden states $i, j \in R(Q^t)$ over time, i.e., $a_{ij} = P(Q^{t+1} = j | Q^t = i, \boldsymbol{\lambda})$; $\mathbf{B}$ is a vector representing the emission probability of the observations given the hidden state, $\mathbf{B} = [b_i(\mathbf{x}^t)]_{i=1}^N$, where $b_i(\boldsymbol{x}^t) = P(\boldsymbol{X}^t = \boldsymbol{x}^t | Q^t = i, \boldsymbol{\lambda})$ is a probability density function (pdf); and $\boldsymbol{\pi}$ is the initial probability distribution of the hidden states, $\boldsymbol{\pi} = [\pi_j]_{j=1}^N$, where $\pi_j = P(Q^0 = j | \boldsymbol{\lambda})$.

An HMM can be used to solve the three following problems. First, compute the likelihood of an observation $\boldsymbol{x}^{0:T}$ given a model $\boldsymbol{\lambda}$, i.e., $P(\mathbf{x}^{0:T} | \boldsymbol{\lambda})$, which can be performed using the forward–backward algorithm. Second, compute the most likely sequence of hidden states and observations, i.e., find the value of $\delta^t(i) = \max_{\boldsymbol{q}^{0:t-1}} \{P(\boldsymbol{x}^{0:t}, \boldsymbol{q}^{0:t-1}, Q^t = i | \boldsymbol{\lambda})\}$, $t = 1, \dots, T$, $i = 1, \dots, N$, which can be solved using the Viterbi algorithm. Third, learn the parameter $\boldsymbol{\lambda}$, which is usually estimated with the EM algorithm [34]. A theoretical tutorial can be found in [4].

B. EM and SEM Algorithms

The expectation–maximization (EM) algorithm is a gradient-based learning strategy used for models where hidden or unknown variables appear [34]. In the case of HMM, it can be seen as a generalization of the Baum–Welch algorithm [4]. The algorithm consists of two parts: the expectation step or E-step, where the *a posteriori* probabilities of the hidden or unknown variables are estimated; the maximization step or M-step, where an auxiliary $\mathcal{Q}$ function (which lower bounds the log-likelihood (LL) of the model) is maximized. Dempster *et al.* [34] showed that the iteration of these steps converges to a local optimum of the LL function.

Some models not only require their parameters to be estimated but also their topology, such as in the case of Bayesian networks. In such cases, the structural EM (SEM) algorithm [35] can be used to simultaneously estimate the model parameters and topology. The algorithm uses an extension of the $\mathcal{Q}$ function of the EM algorithm, where a penalization (usually Bayesian information criterion (BIC) style [36]) in the complexity structure is imposed in order to penalize topologies

[1] An MRF is a probabilistic graphical model, which represents a set of variables that follow the Markov property. The graph must be undirected and may have cycles.

that require a large number of parameters. As in the case of the EM, the SEM algorithm must be iterated until convergence to a local optimum in BIC is reached.

C. Feature Saliency Models

The idea behind any FS model is to declare a set of binary variables, say $\{Z_m\}_{m=1}^M$, which will indicate the feature relevancy. Each Z_m variable follows a Bernoulli distribution with a parameter ρ_m, which is called the FS of the X_m variable. If $\rho_m = 1$, it implies that the feature is relevant. If $\rho_m = 0$, it indicates that the variable is irrelevant. If $\rho_m \in (0, 1)$, a threshold $\overline{\rho}$ can be imposed as a decision boundary to determine whether or not a variable is relevant. For example, in the case of FS-HMM, the FS parameters are added to the emission probabilities [7]

$$b_i(\boldsymbol{X}^t) = \prod_{m=1}^{M} \rho_m \mathcal{N}\left(X_m^t \middle| \mu_{im}, \sigma_{im}^2\right) + (1-\rho_m)\mathcal{N}\left(X_m^t \middle| \epsilon_m, \tau_m^2\right) \tag{1}$$

where $\mathcal{N}(\mu, \sigma^2)$ is the pdf of a normal distribution with mean μ and variance σ^2. If $\rho_m = 1$, the pdf used for the variable X_m depends on the hidden state i, and the variable is affected by changes in the context. This pdf will be referred to as the relevant component. Alternatively, if $\rho_m = 0$, the pdf does not depend on the hidden state i, and the variable X_m is considered as noise.

In the case of mixture models, in [25], assuming that the discrete cluster variable is C, the saliency variables are added to the mixture component densities as

$$\begin{aligned} P(\boldsymbol{X}^t|C_k) = \prod_{m=1}^{M} \rho_{km} \sum_{j=1}^{M_{k_j}} \eta_{kjm} S\left(X_m^t \middle| \mu_{kjm}, \tau_{kjm}, \nu_{kjm}\right) \\ + (1-\rho_{km}) S\left(X_m^t \middle| \epsilon_{km}, \gamma_{km}, \zeta_{km}\right). \end{aligned} \tag{2}$$

C_k is $C = k$, and S is the pdf of a t-Student with mean μ, precision τ, and degrees of freedom ν. In this case, the set of binary variables is $\boldsymbol{Z} = \{Z_{km}\}_{k,m=1}^{K,M}$. Therefore, it is possible to assume that the FS ρ_{km} not only depends on the variable X_m but on the mixture component $C = k$. This means that, depending on the mixture component k and variable X_m, different features may be considered relevant ($Z_{km} = 1$) or noise ($Z_{km} = 0$). From (2), it is clear that the relevant features are assumed to follow a mixture of M_{kj} t-Student pdf (a mixture inside a mixture), whereas irrelevant features only change with the mixture component.

IV. Proposed Model: Feature Saliency Asymmetric HMM

In this contribution, we assume that the emission probabilities are a mixture of Gaussian noise and AR asymmetric linear Gaussian Bayesian networks. Thus, depending on the hidden state, the Bayesian network that describes the relevant distribution may change. This model will be referred to as FS asymmetric HMM (FS-AsHMM). As notation, if $\boldsymbol{Q}^{0:T}$ or $\boldsymbol{Z}^{0:T}$ is found as a summation index, it refers to $\boldsymbol{q^{0:T}} \in R(\boldsymbol{Q^{0:T}})$ or $\boldsymbol{z^{0:T}} \in R(\boldsymbol{Z^{0:T}})$, respectively, where $R(\boldsymbol{F}^{0:T})$ denotes the range of an arbitrary stochastic vector $\boldsymbol{F}^{0:T}$.

The embedded FSS process assumes that irrelevant features are not affected by changes in hidden states; therefore, a Bernoulli vector $\boldsymbol{Z}^t = (Z_1^t, \ldots, Z_M^t)$ is introduced in the model, and the irrelevant behavior is modeled for each variable with a Gaussian distribution with parameters ϵ_m and τ_m^2. The dependency of $\boldsymbol{X}^t$ given $\boldsymbol{Z}^t$ and its at most p^* past values is modeled as

$$\begin{aligned} b_i(\boldsymbol{x}^t|\boldsymbol{z}^t) &:= P\left(\boldsymbol{x}^t \middle| \boldsymbol{x}^{t-p^*:t-1}, \boldsymbol{z}^t, Q^t = i, \boldsymbol{\lambda}\right) \\ &= \prod_{m=1}^{M} f_{im}\left(x_m^t\right)^{z_m^t} g_m\left(x_m^t\right)^{\left(1-z_m^t\right)} \end{aligned} \tag{3}$$

where $f_{im}(x_m^t) = \mathcal{N}(x_m^t|\boldsymbol{\beta}_{im} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im} \cdot \boldsymbol{d}_{im}^t, \sigma_{im}^2)$ is the relevant component, $g_m(x_m^t) = \mathcal{N}(x_m^t|\epsilon_m, \tau_m^2)$ is the noise term, and $\mathbf{pa}_{im}^t = [1, u_{im1}^t, \ldots, u_{imk_{im}}^t]$ and $\boldsymbol{d}_{im}^t = [x_m^{t-1}, \ldots, x_m^{t-p_{im}}]$ are vectors with the values of the k_{im} parents of X_m^t in the Bayesian network graph and its $p_{im} \leq p^*$ past values, with p^* an AR order fixed upper bound. To be clear, the mean of the relevant term is the linear combination of the parameters $\boldsymbol{\beta_{im}} = [\beta_{im0}, \beta_{im1}, \ldots, \beta_{imk_{im}}]$ and $\boldsymbol{\eta_{im}} = [\eta_{im1}, \ldots, \eta_{imp_{im}}]$ with $\mathbf{pa}_{im}^t$ and $\boldsymbol{d}_{im}^t$, respectively, and its variance is σ_{im}^2. The noise term for each variable X_m is a Gaussian distribution with mean ϵ_m and variance τ_m^2, which does not depend on the hidden state.

Observe in Fig. 1 an example of the new model topology. In this example, a network with two variables/features is presented. When $Q^t = 1$, no probabilistic relationships appear between X_1^t and X_2^t; also, X_2^t depends on one AR value or X_2^{t-1}. When $Q^t = 2$, there is a probabilistic dependency of X_2^t from X_1^t. In addition, X_1^t depends on one AR value, that is, X_1^{t-1} and X_2^t depend on two AR values or X_1^{t-1} and X_1^{t-2}. Finally, $\boldsymbol{X}^t$ on both contexts, $Q^t = 1$ and $Q^t = 2$, depends on the binary vector $\boldsymbol{Z}^t$.

The probability of $\boldsymbol{Z}^t$ can be expressed as

$$\zeta(\boldsymbol{z}^t) := P(\boldsymbol{z}^t|\boldsymbol{\lambda}) = \prod_{m=1}^{M} \rho_m^{z_m^t} (1-\rho_m)^{(1-z_m^t)}. \tag{4}$$

$\rho_m := P(Z_m^t = 1|\boldsymbol{\lambda})$ for $m = 1, \ldots, M$. Note that we are assuming that the Z_m^t Bernoulli variables are independent between them and that the ρ_m parameters do not change with time. From (3) and (4), the emission probabilities can be derived

$$\begin{aligned} b_i(\boldsymbol{x}^t) &:= P\left(\boldsymbol{x}^t \middle| \boldsymbol{x}^{t-p^*:t-1}, Q^t = i, \boldsymbol{\lambda}\right) \\ &= \sum_{\boldsymbol{R}(\boldsymbol{Z}^t)} P\left(\boldsymbol{x}^t, \boldsymbol{z}^t \middle| \boldsymbol{x}^{t-p^*:t-1}, Q^t = i, \boldsymbol{\lambda}\right) \\ &= \sum_{\boldsymbol{R}(\boldsymbol{Z}^t)} b_i(\boldsymbol{x}^t|\boldsymbol{z}^t)\zeta\left(\boldsymbol{z}^t\right) \\ &= \prod_{m=1}^{M} \rho_m f_{im}\left(x_m^t\right) + (1-\rho_m) g_m\left(x_m^t\right) \end{aligned} \tag{5}$$

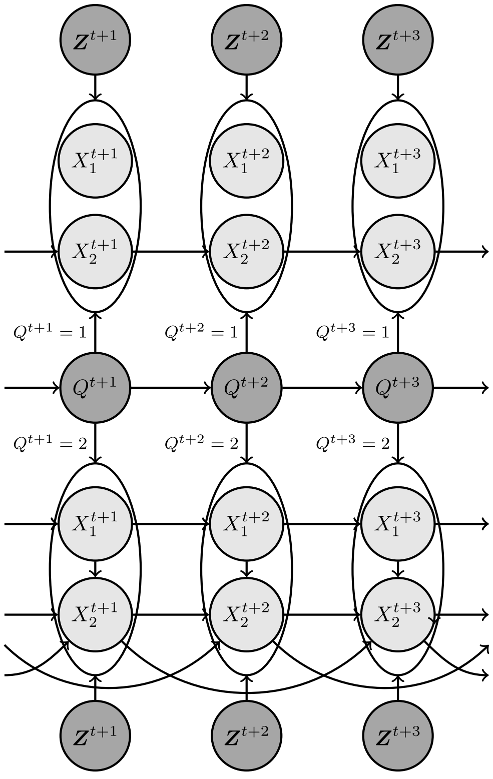


Fig. 1. Example of a structure of a global FS-AsHMM.

and the full information probability can be written as follows:

$$P\left(\boldsymbol{q}^{p^*:T}, \boldsymbol{z}^{p^*:T}, \boldsymbol{x}^{p^*:T} \middle| \boldsymbol{x}^{0:p^*-1}, \boldsymbol{\lambda}\right) = \pi_{q^{p^*}} \prod_{t=p^*}^{T-1} a_{q^t q^{t+1}} \prod_{t=p^*}^{T} \zeta(\boldsymbol{z}^t) b_{q^t}(\boldsymbol{x}^t|\boldsymbol{z}^t). \quad (6)$$

A. E-Step

We define the auxiliary function

$$\mathcal{Q}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') := \sum_{\boldsymbol{Q}^{p^*:T}} \sum_{\boldsymbol{Z}^{p^*:T}} P\left(\boldsymbol{q}^{p^*:T}, \boldsymbol{z}^{p^*:T} \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \times \ln P\left(\boldsymbol{q}^{p^*:T}, \boldsymbol{z}^{p^*:T}, \boldsymbol{x}^{p^*:T} \middle| \boldsymbol{x}^{0:p^*-1}, \boldsymbol{\lambda}\right). \quad (7)$$

From (7), we can obtain the LL of the $\boldsymbol{\lambda}$ model

$$\begin{aligned}\mathcal{Q}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') &= \mathcal{H}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') + \ln P\left(\boldsymbol{x}^{p^*:T} \middle| \boldsymbol{x}^{0:p^*-1}, \boldsymbol{\lambda}\right) \\ &= \mathcal{H}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') + LL(\boldsymbol{\lambda}) \quad (8)\end{aligned}$$

where

$$\mathcal{H}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') = \sum_{\boldsymbol{Q}^{p^*:T}} \sum_{\boldsymbol{Z}^{p^*:T}} P\left(\boldsymbol{q}^{p^*:T}, \boldsymbol{z}^{p^*:T} \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \times \ln P\left(\boldsymbol{q}^{p^*:T}, \boldsymbol{z}^{p^*:T} \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}\right). \quad (9)$$

By (8) and (9), and [5], it is known that each iteration of the EM algorithm with $\mathcal{Q}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}')$ implies improvements in the likelihood function. Introducing (6) in (7), we obtain a tractable expression of $\mathcal{Q}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}')$, which will be useful to find the updating formulas of the model parameters

$$\begin{aligned}\mathcal{Q}_1(\boldsymbol{\lambda}|\boldsymbol{\lambda}') &= \sum_{i=1}^{N} \gamma^{p^*}(i) \ln\left(\pi_i^{p^*}\right) \\ &+ \sum_{t=p^*}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} \xi^t(i,j) \ln\left(a_{ij}\right) \\ &+ \sum_{t=p^*}^{T} \sum_{i=1}^{N} \sum_{m=1}^{M} \psi_m^t(i) \ln\left(\rho_m f_{im}\left(x_m^t\right)\right) \\ &+ \sum_{t=p^*}^{T} \sum_{i=1}^{N} \sum_{m=1}^{M} \phi_m^t(i) \ln\left((1-\rho_m) g_m\left(x_m^t\right)\right). \quad (10)\end{aligned}$$

In (10), we have the latent *a posteriori* probabilities

$$\begin{aligned}\gamma^t(i) &:= P\left(Q^t = i \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \\ \xi^t(i,j) &:= P\left(Q^{t+1} = j, Q^t = i \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \\ \psi_m^t(i) &:= P\left(Q^t = i, Z_m^t = 1 \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \\ \phi_m^t(i) &:= P\left(Q^t = i, Z_m^t = 0 \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \quad (11)\end{aligned}$$

for $t = p^*, \ldots, T$, $i = 1, \ldots, N$, and $m = 1, \ldots, M$. The E-step consists of estimating these quantities. In the case of $\psi_m^t(i)$, we have

$$\begin{aligned}\psi_m^t(i) &= P\left(Q^t = i, Z_m^t = 1 \middle| \boldsymbol{x}^{0:T}, \boldsymbol{\lambda}'\right) \\ &= P\left(Z_m^t = 1 \middle| Q^t = i, \boldsymbol{x}_m^{t-p^*:t}, \boldsymbol{\lambda}'\right) \gamma^t(i) \\ &= \frac{\rho_m f_{im}\left(x_m^t\right) \gamma^t(i)}{\rho_m f_{im}\left(x_m^t\right) + (1-\rho_m) g_m\left(x_m^t\right)}. \quad (12)\end{aligned}$$

It is not hard to note that $\gamma^t(i) = \phi_m^t(i) + \psi_m^t(i)$ for $m = 1, \ldots, M$ and $i = 1, \ldots, N$. Therefore, $\phi_m^t(i) = \gamma^t(i) - \psi_m^t(i)$ and

$$\phi_m^t(i) = \frac{(1-\rho_m) g_m\left(x_m^t\right) \gamma^t(i)}{\rho_m f_{im}\left(x_m^t\right) + (1-\rho_m) g_m\left(x_m^t\right)}. \quad (13)$$

Now, we indicate how to estimate $\gamma^t(i)$

$$\gamma^t(i) = \frac{\alpha_{p^*}^t(i) \beta_{p^*}^t(i)}{LL(\boldsymbol{\lambda}')}. \quad (14)$$

In the previous equation, the forward variable is $\alpha_{p^*}^t(i) := P(Q^t = i, \boldsymbol{x}^{p^*:t} | \boldsymbol{x}^{0:p^*-1}, \boldsymbol{\lambda})$, and the backward variable is $\beta_{p^*}^t(i) := P(\boldsymbol{x}^{t+1:T} | Q^t = i, \boldsymbol{x}^{0:t}, \boldsymbol{\lambda})$. The forward–backward algorithm stated in [5] must be applied to estimate $\alpha_{p^*}^t(i)$ and $\beta_{p^*}^t(i)$. Finally, $\xi^t(i,j)$ can be computed as

$$\xi^t(i,j) = \frac{\alpha_{p^*}^t(i) a_{ij} b_j(\boldsymbol{x}^{t+1}) \beta_{p^*}^{t+1}(j)}{LL(\boldsymbol{\lambda}')}. \quad (15)$$

B. M-Step

The M-step corresponds to optimizing (10) with respect to the model parameters. The following theorem gives the updating formulas that result from the optimization.

Theorem 1: Assume that there is a current model $\boldsymbol{\lambda}^{(s)}$ such that the E-step has been computed with it. From optimizing (10), the resulting parameter $\boldsymbol{\lambda}^{(s+1)}$ can be obtained with the following updating formulas.

The feature saliencies $\{\rho_m^{(s+1)}\}_{m=1}^M$ are updated as

$$\rho_m^{(s+1)} = \frac{\sum_{i=1}^N \sum_{t=p^*}^T \psi_m^t(i)}{T+1-p^*}. \tag{16}$$

The initial distribution $\boldsymbol{\pi}^{(s+1)} = \{\pi_i^{(s+1)}\}_{i=0}^N$ is updated as

$$\pi_i^{(s+1)} = \gamma^{\,p^*}(i). \tag{17}$$

The transition matrix $\mathbf{A}^{(s+1)} = \{a_{ij}^{(s+1)}\}_{i,j=1}^N$ is updated as

$$a_{ij}^{(s+1)} = \frac{\sum_{t=p^*}^{T-1} \xi^t(i,j)}{\sum_{t=p^*}^{T-1} \gamma^{\,t}(i)}. \tag{18}$$

The mean and variance, $\{\epsilon_m^{(s+1)}\}_{m=1}^M$ and $\{(\tau_m^2)^{(s+1)}\}_{m=1}^M$, from the noise component, are updated as

$$\begin{aligned} \epsilon_m^{(s+1)} &= \frac{\sum_{t=p^*}^T \sum_{i=1}^N \phi_m^t(i) x_m^t}{\sum_{t=p^*}^T \sum_{i=1}^N \phi_m^t(i)} \\ \left(\tau_m^2\right)^{(s+1)} &= \frac{\sum_{t=p^*}^T \sum_{i=1}^N \phi_m^t(i)(x_m^t - \epsilon_m)^2}{\sum_{t=p^*}^T \sum_{i=1}^N \phi_m^t(i)}. \end{aligned} \tag{19}$$

Setting $\nu_{im}^t := \boldsymbol{\beta}_{im}^{(s)} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im}^{(s)} \cdot \boldsymbol{d}_{im}^t$ for $m = 1, \ldots, M$, $t = p^*, \ldots, T$, and hidden state $i = 1, \ldots, N$, the parameters $\{\eta_{imr}^{(s+1)}\}_{r=1}^{p_{im}}$ and $\{\beta_{imk}^{(s+1)}\}_{k=0}^{k_{im}}$ can be updated jointly, solving the following linear system:

$$\begin{cases} \sum_{t=p^*}^T \psi_m^t(i) x_m^t = \sum_{t=p^*}^T \psi_m^t(i) \nu_{im}^t \\ \sum_{t=p^*}^T \psi_m^t(i) x_m^t u_{im1}^t = \sum_{t=p^*}^T \psi_m^t(i) u_{im1}^t \nu_{im}^t \\ \quad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ \sum_{t=p^*}^T \psi_m^t(i) x_m^t u_{imk_{im}}^t = \sum_{t=p^*}^T \psi_m^t(i) u_{imk_{im}}^t \nu_{im}^t \\ \sum_{t=p^*}^T \psi_m^t(i) x_m^t x_m^{t-1} = \sum_{t=p^*}^T \psi_m^t(i) x_m^{t-1} \nu_{im}^t \\ \quad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ \sum_{t=p^*}^T \psi_m^t(i) x_m^t x_m^{t-p_{im}} = \sum_{t=p^*}^T \psi_m^t(i) x_m^{t-p_{im}} \nu_{im}^t. \end{cases} \tag{20}$$

Setting $\hat{\nu}_{im}^t := \boldsymbol{\beta}_{im}^{(s+1)} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im}^{(s+1)} \cdot \boldsymbol{d}_{im}^t$, then $\{(\sigma_{im}^2)^{(s+1)}\}_{i,m=1}^{N,M}$ can be updated as

$$\left(\sigma_{im}^2\right)^{(s+1)} = \frac{\sum_{t=p^*}^T \psi_m^t(i)\left(x_m^t - \hat{\nu}_{im}^t\right)^2}{\sum_{t=p^*}^T \psi_m^t(i)}. \tag{21}$$

The proof of this theorem is provided in the Supplementary Material. It is worth noting that, from (20), for each variable $m = 1, \ldots, M$ and hidden state $i = 1, \ldots, N$, the size of the linear system will depend on the number of parents and AR values; the longer the list of dependencies, the larger the linear system.

TABLE II

COMPUTATIONAL COMPLEXITY OF DIFFERENT ROUTINES OF THE LEARNING AND INFERENCE ALGORITHMS

Routine	Complexity
Means	$O(NMT(M+p^*))$
Probabilities	$O(TMN)$
Forward-Backward	$O(TN^3)$
Viterbi	$O(NT(M(M+p^*)+N))$
E-step	$O(TN(N^2+M))$
M-step	$O(NM(M+p^*)^2(M+p^*+T))$
Graph scoring	$O(NM(M+p^*)((M+p^*)^2+T))$

C. Structural EM

In this article, we use the greedy-forward algorithm proposed in [5] to search the space of possible graphical models. However, in this model, it is plausible to think that, if a variable is a noise, it should not be considered in any explanatory graphical model. Therefore, we impose a restriction during the search of structures such that no noise variable is added to any context-specific Bayesian network. The restriction consists of omitting any possible arc coming to or from variables X_m, which fulfills the following condition: $\rho_m \leq \overline{\rho}$, where $\overline{\rho} \in [0, 1)$ is a threshold that determines which variables are relevant. Observe that the opposite of this assumption is not true, i.e., if a variable does not have any relationship with any other variable in a context-specific Bayesian network, it does not mean that it is irrelevant or noise under our relevance definition.

D. Computational Complexity

In Table II, the computational complexity in big O notation of the different routines of the proposed algorithm is shown. We assume that the learned networks are dense or several arcs appear in the context-specific Bayesian networks. We also assume that $p^* \ll T$ or the maximum lag of the AR processes is small compared to the length of the data.

Given a prior or current model $\boldsymbol{\lambda}$, the *means* and *probabilities* routines refer to computing and storing the temporal means ν_{im}^t and probabilities $\{b_i(\boldsymbol{x}^t)\}_{i=1}^N$, $\{f_{im}(x_m^t)\}_{i=1,m=1}^{N,M}$, and $\{g_m(x_m^t)\}_{m=1}^M$ for $t = p^*, \ldots, T$, which are required to perform the *forward–backward*, *Viterbi*, and *E-step* routines. The *forward–backward* routine refers to the computation of the forward variable $\{\alpha_{p^*}^t(i)\}_{i=1}^N$ and backward variable $\{\beta_{p^*}^t(i)\}_{i=1}^N$, $t = p^*, \ldots, T$ in (14). These can be used to compute LLs or perform the *E-step* in the EM algorithm. The *E-step* consists of computing the latent probabilities in (11), and the *M-step* is to update the parameters of $\boldsymbol{\lambda}$ using Theorem 1. Then, the *means* and *probabilities* routines are again executed if another EM iteration is needed. Finally, the *graph-scoring* routine refers to the evaluation of a new set of graphs or context-specific Bayesian networks during the SEM algorithm. It means using the *means*, *probabilities* routines, (20), and (21).

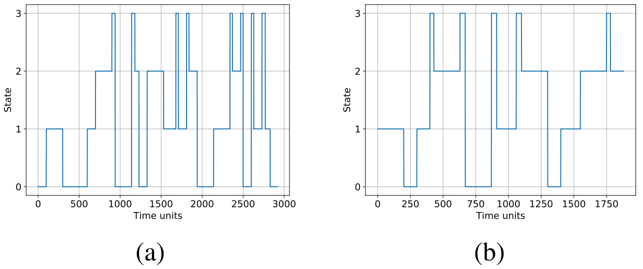
V. EXPERIMENTS

In this section, we will compare our model with the models proposed in [7] (FS-HMM) and [8] (SHMM-LFS). Since these models have been previously compared in favor to clustering

TABLE III

SYNTHETIC DATA GLOBAL DESCRIPTION. S DENOTES THE SCENARIO CASE. # DEP STANDS FOR THE NUMBER OF DEPENDENCIES BETWEEN VARIABLES. # AR REPRESENTS THE NUMBER OF AR DEPENDENCIES. NOISE DENOTES THE NOISE VARIABLES. P. NOISE STANDS FOR THE PARTIAL NOISE VARIABLES

S	# Dep	# AR	Noise	P. noise
1	0	0	X_3, X_5, X_{10}	-
2	28	0	X_3, X_5, X_{10}	-
3	17	17	X_3, X_{10}	X_5, X_7



(a)

(b)

Fig. 2. Signals used for (a) training and (b) testing.

TABLE IV

RESULTS FOR THE TEST SEQUENCE FOR THE DIFFERENT COMPARED MODELS

S	Model	$\overline{\text{LL}}$	$\overline{\text{BIC}}$	σ_{LL}	#
1	AsHMM	**-30527.66**	**61824.36**	9.95	**102**
	FS-HMM	-30600.85	62181.84	9.93	130
	FS-AsHMM	-30537.76	62063.21	**9.91**	131
	SHMM-LFS	-48217.39	98696.64	18.02	300
2	AsHMM	**-30526.87**	**61815.24**	9.57	**101**
	FS-HMM	-31816.89	64613.93	9.68	130
	FS-AsHMM	-31706.78	64431.40	**9.52**	135
	SHMM-LFS	-46062.65	94387.16	16.24	300
3	AsHMM	**-33855.63**	**68955.54**	**0.52**	165
	FS-HMM	-43263.58	87507.51	17.57	**130**
	FS-AsHMM	-34531.54	70239.50	18.57	156
	SHMM-LFS	-702438.3	1407138.94	73.56	300

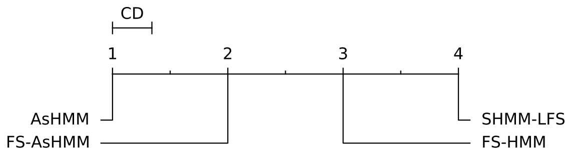


Fig. 3. Critical difference diagram with the Nemenyi hypothesis test for the ranking of BICs.

FS models, such as [22], [23], and [25], we omit these in the experiments. In addition, the model in [5] (AR-AsLG-HMM) will be compared to observe the advantages and disadvantages of using FS models. We will use synthetic data and real data from face grammatical videos and degradation datasets of ball bearings. In the case of [8], the number of mixture components is fixed to 1 for the synthetic data since the data are structured to behave like that. In the case of real data, two components were only used since additional components drastically increased the number of parameters to be estimated, as will be seen later. Finally, for all the experiments, $\overline{\rho} = 0.9$. Recall that this value determines which variables cannot be in the context-specific Bayesian networks. Further details are given in the following. For the sake of space, simply, AsHMM will mean AR-AsLG-HMM.

A. Synthetic Data

1) Data Description: In this study, a synthetic dataset with no physical interpretation is built. It contains noise variables, partial noise variables, and relevant variables. The noise variables are those which follow a normal distribution with fixed mean and variance. Partial noise variables are those whose parameters do not change for every drift in the hidden state in the full data. Relevant variables follow a normal distribution whose mean and variance change with every drift in the hidden state in the dataset. In addition, we assume that noise variables have no probabilistic relationship with other variables. The data is built with probabilistic relationships between variables and different AR values, and is assumed to have four hidden states and ten variables.

Three possible scenarios are analyzed, i.e., three sets of parameters are used. The set of parameters is described in the Supplementary Material. In all the scenarios, the variables with indices 3 and 10 are considered as noise. In scenarios 1 and 2, the variable with index 5 is also noise. In scenario 3, the variables with indices 5 and 7 are considered as partial noise variables. Now, with respect to the dependency maps of each scenario, the first scenario assumes that all the variables are independent. The second scenario assumes that probabilistic relationships between relevant variables may appear. The third scenario is the most complex due to the presence of probabilistic relationships between variables and AR values, and some variables are partial noise. This information is summarized in Table III.

The training data are generated following the sequence of hidden states exposed in Fig. 2(a). For the testing phase, a testing sequence of hidden states is used, which is pictured in Fig. 2(b). The testing sequence is generated fifty times for each scenario for population evaluation purposes. To evaluate and compare the models, the mean LL ($\overline{\text{LL}}$), BIC ($\overline{\text{BIC}}$) [36], and the standard deviation of LL (σ_{LL}) are used.

2) Synthetic Data Results: In Table IV, the obtained results for the different models in terms of LL and BIC are shown. In boldface, the best scores are marked and, in the case of the number of parameters, the model with the lowest amount of parameters. In addition, Fig. 3 shows the critical difference diagrams [37] with confidence of 90% for the obtained rankings in the testings datasets for LL and BIC scores, respectively. In this case, we have used the Nemenyi test, which evaluates for all pairs of models the hypothesis of no difference in the ranking position. The *CD* value indicates the minimum distance in the rank to give evidence of statistical difference. In the graphs, models grouped by the same bold line are not statistically different in their rank. In our critical diagrams, 1 is the best rank, and 4 is the worst rank.

From Fig. 3, it can be observed that the model with the best BIC in mean was AsHMM, followed in order by FS-AsHMM,

TABLE V

ρ RESULTS IN SCENARIO 3 FOR THE DIFFERENT MODELS IN THE SYNTHETIC DATA

Model	Q	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5
		ρ_6	ρ_7	ρ_8	ρ_9	ρ_{10}
FS-HMM		0.98	1.0	0.07	0.81	0.07
		1.0	0.98	1.0	0.93	0.06
FS-AsHMM		0.98	0.99	0.08	0.82	0.1
		1.0	0.98	1.0	0.93	0.11
SHMM-LFS	1	1.0	1.0	1.0	1.0	1.0
		1.0	1.0	1.0	1.0	1.0
	2	1.0	1.0	1.0	1.0	1.0
		1.0	1.0	1.0	1.0	1.0
	3	1.0	1.0	1.0	1.0	1.0
		1.0	1.0	1.0	1.0	1.0
	4	1.0	1.0	1.0	1.0	1.0
		1.0	1.0	1.0	1.0	1.0

TABLE VI

SECONDS TO LEARN A MODEL BY SCENARIO IN THE SYNTHETIC DATA

Model	t-S1 (s)	t-S2 (s)	t-S3 (s)
AsHMM	25.97	20.05	26.23
FS-HMM	53.99	144.30	22.79
FS-AsHMM	39.08	169.53	61.54
SHMM-LFS	**14.61**	**14.64**	**14.21**

FS-HMM, and SHMM-LFS. By the Nemenyi test, it can be noted that all models were statistically different since no bold line paired pairs of models. The same results were obtained for the rankings with LL.

In terms of standard deviations of the prediction scores, SHMM-LFS obtained the worst results in all the scenarios. Next, FS-AsHMM obtained the best standard deviation in scenarios 1 and 2, but it increased in scenario 3 where AsHMM obtained the best result.

For the number of parameters (last column in Table IV), AsHMM obtained the least amount in scenarios 1 and 2 in spite of the fact that FS-HMM assumed full independence between variables. However, in scenario 3, FS-HMM used the least number of parameters since AsHMM increased drastically its number of parameters. It must be noted that, in all the scenarios, the number of parameters of the asymmetric models changed; this is due to the different context-specific Bayesian networks that were found during the learning phase. It is also noticeable that the SHMM-LFS model obtained the largest amount of parameters in spite of the fact that this model also assumed full independence between variables. This model, in particular, in contradiction with the definition of irrelevant features [2], assumes that the noise distribution also changes its parameters with the hidden state and mixture component, which increases its number of parameters drastically.

It has been shown that AsHMM obtained good fitness and does not require a critical amount of parameters. Nonetheless, the proposed models introduce feature saliencies that are capable of estimating feature relevancy and provide more data insights. Following this idea, in Table V, we can observe the estimated relevancies of each model for scenario 3. Recall from the synthetic data description that variables 3 and 10 are represented as noise, i.e., their parameters do not change with the hidden states; variables 5 and 7 are represented as partial noise variables, i.e., their parameters do not change for all the hidden states. Observe that FS-HMM and FS-AsHMM were capable of detecting the noise or irrelevant variables since the relevancies of ρ_3 and ρ_{10} are close to zero, while relevancies of partial noise variables, ρ_5 and ρ_7, obtained mixed results. The remaining values have higher relevancy but with contrasts, e.g., the relevancies ρ_2, ρ_6, and ρ_8 are close to one ($\rho > 0.9$), but, for relevancies ρ_4, it is not clear if they are totally relevant or not ($0.5 < \rho < 0.9$).

In the case of SHMM-LFS, the relevancies change with the hidden states. In Table V, the column Q refers to each (numbered) hidden state. Regarding SHMM-LFS, the feature saliencies are estimated at the component level in a mixture of Gaussians; however, as previously mentioned, for comparative purposes, mixture models with only one component are considered in the synthetic data. In this case, observe that, for SHMM-LFS, all the features are predicted as relevant, for all the hidden states and features.

Concerning the execution times of the tested algorithms, in Table VI, it is reported the learning times of all the algorithms for each scenario. Note that FS-AsHMM was the largest time consumer in two out of the three scenarios. In scenario 1, FS-HMM was the slowest model to learn. In this manner, it is observed that the additional information that was provided by FS-AsHMM and FS-HMM (feature saliencies) had the cost of longer training times. SHMM-LFS was the quickest in all the scenarios. However, as seen before, the data insights obtained by this algorithm were poor. In an intermediate point, AsHMM can be found. The algorithm is capable of giving context-specific Bayesian networks, but no feature selection is performed.

B. Grammatical Facial Expression Data

1) Data Description: In this experiment, grammatical facial expression data are used. Facial gestures are relevant in any sign language: Given a hand signal sequence, facial gestures can change their grammatical sense. The data by Freitas *et al.* [38] were collected using a Kinect camera that recorded fluent Brazilian sign language signalers. From the videos, spacial facial points were collected, processed, and used for classification and segmentation tasks. In each video, there was a person repeating, with pauses, sentences with grammar content. The aim of the problem is to determine in which frames the grammatical content is being performed. HMMs have been used before to tackle this kind of problem (see [39] and [40]). The possible grammar contents are given as follows.

1) *Affirmative (Affir):* Used to make positive sentences.
2) *Conditional (Cond):* Used to create subjunctive clauses.
3) *Doubt Question (DQ):* Used to indicate that new information is being added.
4) *Emphasis (Emp):* Used to highlight information.
5) *Negative (Neg):* Used to make negative sentences.
6) *Relative Sentence (Rela):* Used to provide more information.
7) *Topic (Topic):* Used to change subject.

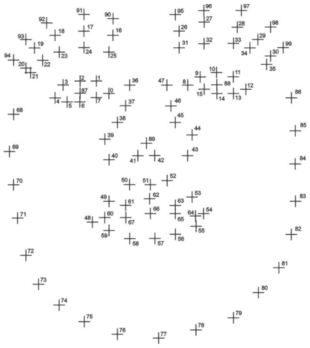


Fig. 4. Raw face point locations.

TABLE VII

Raw Selected Points

Point	Face section	Selected points
$\mathbf{0-7}$	Left eye	2
$\mathbf{8-15}$	Right eye	10
$\mathbf{16-25}$	Left eyebrow	17
$\mathbf{26-35}$	Right eyebrow	27
$\mathbf{36-47}$	Nose	39, 44
$\mathbf{48-67}$	Mouth	48, 51, 54, 57
$\mathbf{68-86}$	Face contour	-
$\mathbf{87}$	Left iris	-
$\mathbf{88}$	Right iris	-
$\mathbf{89}$	Nose tip	89
$\mathbf{90-94}$	Line above left eyebrow	-
$\mathbf{95-99}$	Line above right eyebrow	-

TABLE VIII

Features Construction

Expert knowledge reduced dataset			
d_1	d_2	d_3	d_4
$\|\|\mathbf{2-17}\|\|$	$\|\|\mathbf{17-27}\|\|$	$\|\|\mathbf{27-10}\|\|$	$\|\|\mathbf{10-89}\|\|$
d_5	d_6	d_7	d_8
$\|\|\mathbf{89-2}\|\|$	$\|\|\mathbf{39-89}\|\|$	$\|\|\mathbf{89-44}\|\|$	$\|\|\mathbf{44-57}\|\|$
d_9	d_{10}	d_{11}	a_1
$\|\|\mathbf{57-39}\|\|$	$\|\|\mathbf{57-51}\|\|$	$\|\|\mathbf{48-54}\|\|$	$\angle(\mathbf{10,17,2})$
a_2	a_3	a_4	a_5
$\angle(\mathbf{2,27,10})$	$\angle(\mathbf{89,48,54})$	$\angle(\mathbf{48,89,51})$	$\angle(\mathbf{54,81,51})$
a_6	a_7		
$\angle(\mathbf{48,51,57})$	$\angle(\mathbf{54,51,57})$		

8) *WH-Questions (WH):* Used to create who, what, where, ... questions.
9) *Y/N-Questions (YN):* Used to create yes/no questions.

Later, another expert in Brazilian sign language labeled the video frames indicating when the signaler is performing the sentence. Each video has a unique grammar content, where five repetitions of five different sentences in which the face is not overshadowed by the hand signals are recorded. Each dataset has spatial (x, y, z) coordinate information from 100 facial points (300 raw features). However, by expert knowledge, the dataset can be reduced down to 18 features that contain information about distances and angles between relevant face sections in the (x, y) plane [38]. More details about the raw features can be found in Fig. 4 and Table VII.

The idea is to select relevant points from relevant face parts. In Table VIII, the 18 extracted features are described, namely, $d_1, \ldots, d_{11}$, which are distances between face points, and $a_1, \ldots, a_7$, which are angles between face points, denoted as $||\boldsymbol{a} - \boldsymbol{b}|| = ((a[x] - b[x])^2 + (a[y] - b[y])^2)^{1/2}$ and $\angle(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \arccos((\boldsymbol{b} - \boldsymbol{a}) \cdot (\boldsymbol{c} - \boldsymbol{a}))/(||\boldsymbol{b} - \boldsymbol{a}||||\boldsymbol{c} - \boldsymbol{a}||)$. Two people perform the same sentences for each grammatical context. We take three of the five repetitions of the two signalers as input for a model for each grammar content. Later, we evaluate models with the remaining two repetitions of both signalers for each grammar content (18 sequence tests). The mean $\overline{\text{LL}}$ and $\overline{\text{BIC}}$ scores are reported (standard deviation is not reported since only two test sequences are available for each grammar content). From the supervised binary problem (grammar content or not), the accuracy, recall, and F-score from the classification task are provided.

TABLE IX

Likelihood, BIC Score, and the Number of Parameters Obtained by the Models for the Testing Face Grammar Videos

Grammar	Model	$\overline{\text{LL}}$	$\overline{\text{BIC}}$	#
Afir	AsHMM	**-42403.96**	**85935.22**	183
	FS-HMM	-51617.50	104048.14	**132**
	FS-AsHMM	-51182.32	103220.91	139
	SHMM-LFS	-351868.77	706879.23	510
Cond	AsHMM	**-76062.93**	**154090.58**	289
	FS-HMM	-97076.42	195050.22	**132**
	FS-AsHMM	-93579.59	188178.92	150
	SHMM-LFS	-670067.66	1343602.47	510
DQ	AsHMM	-66560.91	133622.67	**78**
	FS-HMM	-65544.13	131935.84	132
	FS-AsHMM	**-63677.90**	**128267.58**	142
	SHMM-LFS	-458428.4	920131.53	510
Emp	AsHMM	**-40022.70**	**81753.47**	275
	FS-HMM	-53869.89	108559.64	**132**
	FS-AsHMM	-53864.29	108548.44	**132**
	SHMM-LFS	-368888.71	740945.10	510
Neg	AsHMM	**-59059.31**	**120344.83**	349
	FS-HMM	-74477.48	149796.97	**132**
	FS-AsHMM	-75139.11	151196.76	144
	SHMM-LFS	-453618.42	910490.05	510
Rela	AsHMM	**-77468.23**	**157457.23**	363
	FS-HMM	-109543.42	220003.48	**132**
	FS-AsHMM	-107576.03	216110.36	138
	SHMM-LFS	-774777.30	1553096.18	510
Topic	AsHMM	**-69567.09**	**140984.06**	277
	FS-HMM	-85695.22	172271.97	**132**
	FS-AsHMM	-82010.55	165036.20	152
	SHMM-LFS	-588667.95	1180741.82	510
WH	AsHMM	**-35070.70**	**72076.86**	318
	FS-HMM	-47792.39	96388.17	**132**
	FS-AsHMM	-47804.90	96413.2	**132**
	SHMM-LFS	-327078.86	657261.75	510
YN	AsHMM	**-52210.02**	**106155.85**	270
	FS-HMM	-68438.54	137725.68	**132**
	FS-AsHMM	-67808.19	136477.85	134
	SHMM-LFS	-465458.38	934195.49	510

2) Grammatical Face Data Results: From the previous information, the models BICs and LLs for each grammar case are reported. As the aim of this problem is to obtain a binary segmentation of the recorded videos, two hidden states will be used. The accuracy, recall, and F-score obtained by each model will also be reported. Next, the model corresponding to the *Topic* grammar case is explored. From it, its feature saliencies and learned Bayesian networks are presented and analyzed. In this manner, the additional data insights that the proposed models can provide are highlighted.

In Table IX, the LLs, BIC scores, and the number of parameters obtained by the different models for each grammatical

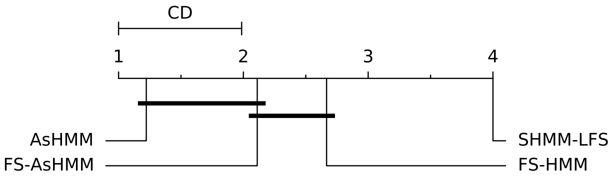


Fig. 5. Critical difference diagram with the Nemenyi hypothesis test for the ranking of BIC scores.

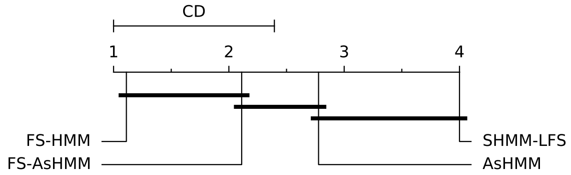


Fig. 6. Critical difference diagram with the Nemenyi hypothesis test for the ranking of the number of variables.

scenario are shown. In addition, Fig. 5 shows the critical difference diagrams for the BIC score. A confidence level of 90% was used. The critical difference diagram from the LL score was omitted since it was the same as in the case of BIC.

Note that AsHMM obtained the best BIC score and LL, followed in order by LFS-AsHMM, FS-HMM, and SHMM-LFS. From the hypothesis test, it can be observed that FS-AsHMM was statistically better than SHMM-LFS, but not enough evidence is available to confirm that it performed better than FS-HMM or worse than AsHMM.

In Fig. 6, we can observe the critical difference diagram for the number of parameters. FS-AsHMM was better ranked (fewer parameters) than SHMM-LFS but not significantly different from FS-HMM and AsHMM. As a final comment, it is remarkable that SHMM-LFS obtained the highest amount of parameters again, as in the synthetic data.

The proposed problem is to learn a model capable of predicting from a video whether or not a signaler is performing a certain grammatical face expression. Therefore, we learn a model for each training set, and then, we predict the testing video. However, as disclaimed, all the models are unsupervised, and they do not take into account the class variable. It follows that the generated models may not be segmenting or clustering the actions corresponding to the class variable.

The prediction phase is performed using the Viterbi algorithm. The Viterbi algorithm in this case, as only two hidden states are considered, returns sequences of zeros and ones. However, it is not clear what zero or a one implies in this sequence; therefore, we compute the confusion matrix for the two following possible assignments.

1) 1 is a grammar expression, and 0 is not.
2) 0 is a grammar expression, and 1 is not.

From the confusion matrix of each assignment, the accuracy is computed, and the assignment with the greater accuracy is chosen as the model segmentation. Nevertheless, a better decision rule could have been made if an expert in Brazilian sign language had reviewed the learned parameters.

TABLE X

PREDICTION SCORES OBTAINED BY THE MODELS FOR THE TESTING FACE GRAMMAR VIDEOS

Grammar	Model	Acuracy	Recall	F-score
Afir	AsHMM	0.61	0.0	0.0
	FS-HMM	**0.70**	**0.79**	**0.67**
	FS-AsHMM	0.68	0.67	0.62
	SHMM-LFS	0.62	0.0	0.0
Cond	AsHMM	0.68	**0.85**	0.61
	FS-HMM	**0.81**	0.77	**0.70**
	FS-AsHMM	**0.81**	0.68	0.67
	SHMM-LFS	0.72	0.0	0.0
DQ	AsHMM	0.57	0.0	0.0
	FS-HMM	0.56	0.52	0.50
	FS-AsHMM	**0.59**	**0.53**	**0.53**
	SHMM-LFS	0.58	0.0	0.0
Emp	AsHMM	0.63	0.59	0.51
	FS-HMM	**0.89**	**0.91**	**0.84**
	FS-AsHMM	0.85	0.85	0.78
	SHMM-LFS	0.67	0.0	0.0
Neg	AsHMM	0.56	0.37	0.43
	FS-HMM	**0.58**	0.41	0.47
	FS-AsHMM	0.56	**0.49**	**0.50**
	SHMM-LFS	0.55	0.0	0.0
Rela	AsHMM	0.55	0.54	0.40
	FS-HMM	**0.86**	**0.95**	**0.79**
	FS-AsHMM	**0.86**	0.90	**0.79**
	SHMM-LFS	0.72	0.0	0.0
Topic	AsHMM	0.72	0.71	0.54
	FS-HMM	**0.86**	**0.91**	**0.75**
	FS-AsHMM	0.85	0.87	0.73
	SHMM-LFS	0.77	0.0	0.0
WH	AsHMM	0.56	0.36	0.42
	FS-HMM	0.75	0.54	0.65
	FS-AsHMM	**0.77**	**0.74**	**0.74**
	SHMM-LFS	0.56	0.0	0.0
YN	AsHMM	0.72	**0.77**	0.69
	FS-HMM	0.73	0.66	0.66
	FS-AsHMM	**0.80**	0.71	**0.74**
	SHMM-LFS	0.60	0.0	0.0

In Table X, the total accuracy, recall, and F-score obtained from the testing videos from signalers A and B are shown. In addition, in Figs. 7–9, the corresponding critical difference diagrams are provided. For accuracy (see Fig. 7), it can be observed that the best ranked models were FS-AsHMM and FS-HMM, followed in order by SHMM-LFS and AsHMM. Statistically, two equivalence groups were found: the first one consisted of FS-HMM and FS-AsHMM, and the second one contained AsHMM and SHMM-LFS.

In terms of recall or true positive rate, Fig. 8 shows that the best ranked model was FS-HMM, followed in order by FS-AsHMM, AsHMM, and SHMM-LFS. In Table X, it can be observed that there are models with high recall scores and low accuracy, and models with zero recall but with relevant accuracy. The main reason for this issue is that the output of the Viterbi algorithms for some models and grammar cases is a constant line at 0 or 1. Thus, this measure is not helpful to differentiate between the tested models. Therefore, the F-score can be used to solve this issue and give another perspective of the model performance for detecting grammar facial expressions.

Then, for the F-score, Fig. 9 shows that the best rank was obtained by FS-AsHMM, followed in order by FS-HMM,

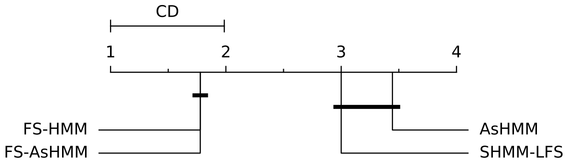


Fig. 7. Critical difference diagram with the Nemenyi hypothesis test for the ranking of accuracy.

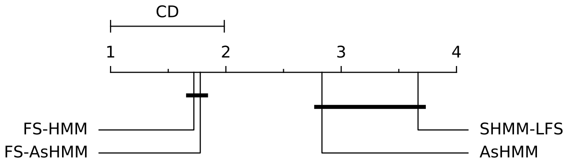


Fig. 8. Critical difference diagram with the Nemenyi hypothesis test for the ranking of recall score.

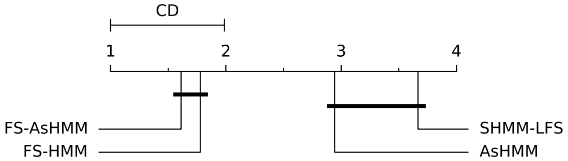


Fig. 9. Critical difference diagram with the Nemenyi hypothesis test for the ranking of F-scores.

AsHMM, and SHMM-LFS. FS-AsHMM is statistically better than AsHMM and SHMM-LFS but statistically equivalent to FS-HMM.

In conclusion, from the accuracy level and F-score, the most accurate and reliable models were FS-AsHMM and FS-HMM for the classification of this kind of data. AsHMM obtained intermediate or poor accuracy and F-score results.

It has been shown from the previous results that FS-AsHMM and FS-HMM can be good unsupervised models to predict grammar facial expressions. However, it is also relevant to check their feature saliencies and observe which variables were relevant for the learning process. In Table XI, the relevancies for each model in the grammar case of *Topic* are presented. Note that FS-HMM and FS-AsHMM (models 1 and 2) selected the same variables under the rule $\rho_i > \overline{\rho}$, i.e., the variables d_3, d_4, d_{10}, and d_{11}. The remaining variables are not relevant or lie at an intermediate level. In any case, only the previously mentioned variables are considered in the context-specific Bayesian network inside FS-AsHMM and are able to have probabilistic relationships with other variables. Regarding SHMM-LFS (model 3), all the variables are relevant for both hidden states, which is undesirable since no FSS procedure is performed.

Observe that from FS-AsHMM and FS-HMM, the set of relevant features is small; only four of the eighteen variables are relevant. In this sense, it can be argued that AsHMM learned noise and predicted noise during the testing phase. In comparison, FS-AsHMM and FS-HMM also learned noise but are capable of detecting the level of noise in the variables and use it during the determination of context-specific Bayesian networks and the prediction phase. This may explain why AsHMM obtained good scores in BIC and LL but performed poorly when the class variable was considered.

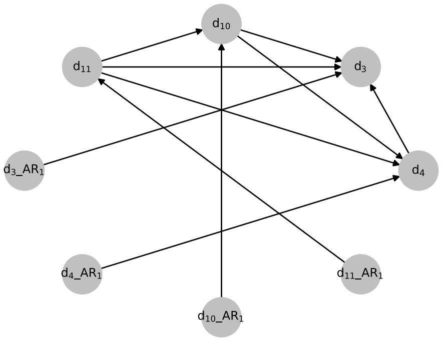


Fig. 10. Learned context-specific Bayesian networks from the FS-AsHMM model for the facial grammar *Topic*.

TABLE XI

ρ RELEVANCIES FOR THE CASE OF THE GRAMMATICAL CASE OF *Topic*. COLUMN M REFERS TO THE MODEL: 1 IS FS-HMM MODEL, 2 IS FS-ASHMM AND 3 IS SHMM-LFS. COLUMN Q REFERS TO THE HIDDEN STATES; IT IS ONLY USED WHEN A MODEL HAS RELEVANCIES THAT DEPEND ON THE HIDDEN STATE (MODEL 3)

		Distance						
M	Q	ρ_{d_1}	ρ_{d_2}	ρ_{d_3}	ρ_{d_4}	ρ_{d_5}	ρ_{d_6}	ρ_{d_7}
1		0.47	0.44	0.9	0.93	0.09	0.5	0.56
2		0.29	0.31	1.0	0.96	0.02	0.44	0.66
3	1	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	2	0.93	0.93	0.93	0.93	0.93	0.93	0.93
M	Q	ρ_{d_8}	ρ_{d_9}	$\rho_{d_{10}}$	$\rho_{d_{11}}$			
1		0.26	0.5	0.91	0.98			
2		0.11	0.22	0.99	0.96			
3	1	0.93	0.93	0.93	0.93			
	2	0.93	0.93	0.93	0.93			
		Angle						
M	Q	ρ_{a_1}	ρ_{a_2}	ρ_{a_3}	ρ_{a_4}	ρ_{a_5}	ρ_{a_6}	ρ_{a_7}
1		0.27	0.33	0.42	0.81	0.85	0.27	0.23
2		0.07	0.12	0.09	0.84	0.79	0.04	0.03
3	1	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	2	0.93	0.93	0.93	0.93	0.93	0.93	0.93

Finally, we look at the learned context-specific Bayesian networks. In Fig. 10, we can observe one of the networks learned from the FS-AsHMM model in the *Topic* grammar case when the grammar expression is being performed. In the graph, the nodes of the context-specific Bayesian network are labeled as *variable_AR_#order*, where *AR_#order* is the AR order. If it is 0, then this suffix is ignored for the label since it is the original variable. In this case, the four relevant distances interact between them. In particular, d_{11} is the distance that governs the network since the remaining distances depend on this one. Also, it is remarkable that one AR value is relevant for all the selected distances, which indicates that the previous (in time) distances have an impact on the current ones. This kind of information can be useful for domain experts to determine or validate how the different face sections interact between them for the different grammar contents. Finally, we note that these networks are only available for asymmetric models. Models such as FS-HMM and SHMM-LFS are not capable of giving this kind of insight.

Regarding the computational cost of the tested algorithms in the grammatical facial expression dataset, in Table XII,

TABLE XII

MEAN AND STANDARD DEVIATION LEARNING TIMES PER UNIT DATA FOR THE GRAMMATICAL FACIAL EXPRESSION DATASET

Model	$\bar{t}/T$ (s)	σ_t/T (s)
AsHMM	0.0287	0.0073
FS-HMM	0.0126	0.0053
FS-AsHMM	0.0305	0.0277
SHMM-LFS	**0.0058**	**0.0004**

a brief summary of learning times is shown. As the dataset for learning each model has a different length, we use the time per unit of data to estimate the time performance of the four algorithms. Observe that, in mean and variance, the most expensive algorithm was FS-AsHMM, followed by AsHMM, FS-HMM, and SHMM-LFS. In this dataset, the proposed algorithm was 2.42 times slower compared to its version without context-specific Bayesian networks (FS-HMM) and 1.06 times slower compared with its version without feature saliencies (AsHMM). Finally, in spite of the fast learning times of SHMM-LFS, the learned models were not capable of extracting relevant information from data.

C. Ball-Bearings' Degradation Data

1) Data Description: Ball bearings are fundamental components inside large tool machines in industrial production lines. Thanks to the newest technologies in embedded systems, sensors, Internet, and cloud service, it is possible to monitor these components continuously and apply artificial intelligence algorithms to determine the ball-bearing health status [41]. However, the raw signals must be significant and useful to represent the ball-bearing health. Since the measurements come from real machines, it is normal that the raw signal contains undesirable noise that must be filtered. Therefore, the raw signal must be passed through different signal processing algorithms, such as filtering, demodulation, and decimation. In this manner, relevant features must be extracted to be used in tool condition monitoring.

In this work, we filter the raw vibrational signals using spectral Kurtosis algorithms as in [42] and extract relevant bearing fundamental frequency amplitudes, such as ball pass frequency outer (BPFO) related to the ball-bearing outer race, ball pass frequency inner (BPFI) related to the ball-bearing inner race, ball spin frequency (BSF) related to the ball-bearing rollers, and the fundamental train frequency (FTF) related to the ball-bearing cage. From these four fundamental frequencies, three harmonics are taken into account (a total of 16 variables). It is known that harmonic frequency magnitudes become more relevant when a failure is present [43]. However, it is expected that some harmonic components can be more relevant than others. The idea of this study is to determine the level of relevancy of the different harmonics when a failure is present.

To validate the proposed models in the case of real ball bearings, the benchmark and data provided by Qiu *et al.* [44] are used. The benchmark consists of four ball bearings mounted on the same shaft under a known mechanical load, as pictured in Fig. 11. In the presence of a failure in any part of the system, vibrations will be generated that will be transmitted across the whole system. Vibrational sensors are used to record the mechanical setup. The training signal has 2156 records, whereas the testing signal has 6324 records. Models with two to five hidden states were tested, and with four hidden states, all the models obtained overall acceptable results.

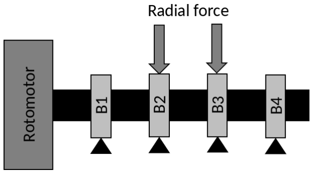


Fig. 11. Mechanical setup. Four ball bearings ZA-2115 with labels B1, B2, B3, and B4 are mounted in a shaft at a rotational speed of 2000 rpm. Bearings with labels B2 and B3 are under an external force of 2721.554 kg. The sensors have a sample rate of 20 kHz.

TABLE XIII

LLS, BICS, AND THE NUMBER OF PARAMETERS OF THE MODELS IN THE TESTING SIGNALS FOR ALL THE BEARINGS B1, B2, B3, AND B4

B	Model	LL	BIC	#
B1	AsHMM	**-607437.06**	**1217153.25**	261
	FS-HMM	-619005.39	1239722.32	**196**
	FS-AsHMM	-618342.31	1238518.40	210
	SHMM-LFS	-4587984.01	9183966.81	916
B2	AsHMM	**-616773.52**	**1236088.15**	291
	FS-HMM	-618441.27	1238594.07	**196**
	FS-AsHMM	-618198.49	1238274.43	215
	SHMM-LFS	-4587984.01	9183966.81	916
B3	AsHMM	**-616748.69**	**1236710.86**	368
	FS-HMM	-880412.04	1762535.61	**196**
	FS-AsHMM	-680386.46	1363811.76	348
	SHMM-LFS	-4587984.01	9183966.81	916
B4	AsHMM	**-470522.75**	**945918.13**	558
	FS-HMM	-550028.31	1101768.14	**196**
	FS-AsHMM	-489337.82	983129.11	510
	SHMM-LFS	-4587984.01	9183966.81	916

In the training dataset, Bearing 3 fails due to its inner race and Bearing 4 due to its rollers. In the testing dataset, Bearing 3 fails due to its outer race. Bearing 3 is the most relevant mechanical component for this case studio since it fails in both the training and testing datasets.

2) Ball-Bearing Data Results: In Table XIII, the results obtained in the test by each model are shown. Observe that, for all the cases, the model with the best fitness was AsHMM followed always in order by FS-AsHMM, FS-HMM, and SHMM-LFS. For B1 and B2, it can be observed that, for all the models, omitting SHMM-LFS, the maximum relative difference with respect to the best BIC score was 1.85%. Meanwhile, in B3, the differences are larger; in the case of FS-AsHMM, the relative difference with respect to the best BIC score model was 10.28%, whereas FS-HMM obtained a much higher difference of 42.52%. This can be explained because B3 is the failing bearing. Due to its exponential behavior in the failing phase, AR parameters and probabilistic dependencies between features can play an important role in explaining this behavior [5]; since FS-HMM assumes full independence, these dependencies were ignored. In the case of B4, only FS-AsHMM was close to the best scoring model with a relative difference of 3.93% and FS-HMM 16.47%.

In terms of parameters, FS-HMM obtained the least amount of parameters in all cases. Since this model assumes full independence variables, no further parameters are added as opposed to the asymmetric model. On the other hand, for

TABLE XIV

LEARNED FEATURE SALIENCIES FOR EACH MODEL IN THE CASE OF B3. COLUMN M REFERS TO THE MODEL, 1 IS THE FS-HMM MODEL, 2 IS FS-ASHMM, AND 3 IS SHMM-LFS. COLUMN Q REFERS TO HIDDEN STATE; IT IS ONLY USED WHEN A MODEL HAS RELEVANCIES THAT DEPEND ON THE HIDDEN STATE (MODEL 3)

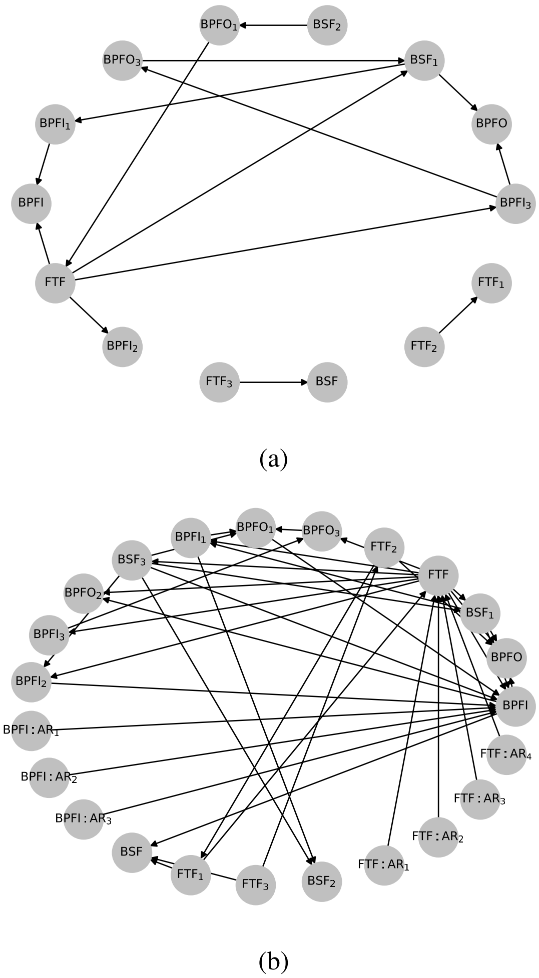
		BPFO				BPFI			
M	Q	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6	ρ_7	ρ_8
1		0.83	0.85	0.81	0.80	0.88	0.83	0.83	0.77
2		1.0	0.99	0.99	0.96	1.0	1.0	1.0	0.98
3	1	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	2	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	3	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	4	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
		BSF				FTF			
M	Q	ρ_9	ρ_{10}	ρ_{11}	ρ_{12}	ρ_{13}	ρ_{14}	ρ_{15}	ρ_{16}
1		0.80	0.82	0.8	0.86	0.90	0.82	0.82	0.78
2		0.98	0.99	0.94	0.99	0.99	0.98	0.98	0.96
3	1	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	2	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	3	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	4	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93

all the ball bearings, SHMM-LFS uses the highest amount of parameters. Observe that, for each ball bearing, the number of parameters changes again for each asymmetric model in spite of the fact that all of them are of the same kind. In particular, notice that, in ball bearings B1 and B2, the amount of parameters is much lower than in B3 and B4; this is because, in the training phase, B3 and B4 fail.

Although the proposed FS models in this case study did not obtain the best results in BIC or LL when compared to AsHMM, we will see that the feature saliencies provided by the proposed models can give further data insights about which AsHMM is not able to provide. From the dataset description, it is known that the training and testing ball bearing B3 fails. Therefore, more attention to the feature saliencies from B3 is given. In Table XIV, the learned feature saliencies for each model are shown. The 16 features are divided into four groups. Features with indices $1, 5, 9$, and 13 correspond to the fundamental frequencies BPPFO, BPFI, BSF, and FTF, respectively. The next three indices for each fundamental frequency correspond to its first, second, and third harmonics, e.g., indices $2, 3$, and 4 are the first, second, and third BPFO harmonics.

From Table XIV, we can observe that, for FS-HMM (Model 1), the FTF fundamental frequency is the most relevant feature with relevancy of 0.9. This result is unexpected since, in the training phase, the whole degradation process is being observed, and therefore, more harmonics and fundamental frequencies should be relevant. Yet, it is remarkable that the relevancy of some fundamental frequencies, say BPFI and FTF, is higher than the relevancy of their harmonics. Meanwhile, for BPFO, only the first harmonic has greater relevancy than the fundamental frequency, and in the case of BSF, the fundamental frequency has lower relevancy than its harmonics, being the third harmonic frequency with the largest relevancy among the BSF frequencies.

In the case of FS-AsHMM, all the frequencies (fundamental and harmonics) are relevant since their relevancy fulfills the condition $\rho_i \geq \overline{\rho}$; this can be explained because all the degradation processes of the ball bearing are being considered during the training phase. It is worth mentioning that, for the BPFO and FTF frequencies, the larger the harmonic, the lower the relevancy. Also, as in the case of FS-HMM, BSF harmonics could obtain greater relevancies compared to the relevancy of the fundamental frequency. If the relevancy threshold was tighter, for example, $\overline{\rho} = 0.99$, the relevant frequencies would have changed. In this case, the third harmonic of the BPFO and BPFI frequency, the fundamental and second harmonics of the BSF frequency, and the first, second, and third harmonics of the FTF frequency would be irrelevant. In addition, for the FS-HMM model, no variable would be relevant.



(a)

(b)

Fig. 12. Learned context-specific Bayesian structures from the FS-AsHMM model for the ball-bearing B3. Graph learned in (a) low degradation state and (b) high degraded state.

As previously mentioned, the proposed models do not assume full probabilistic dependency or independence between variables as other models do. Fig. 12 shows a pair of context-specific Bayesian networks learned by the FS-AsHMM corresponding to low and high degradation levels.

In the case of the Bayesian network in Fig. 12(a), it represents a low degradation level. We can observe that different probabilistic relationships appear. For instance, the FTF fundamental frequency has the most number of descendants: the BSF first harmonic and the BPFI fundamental, second, and

TABLE XV
TIMES TO LEARN A MODEL FOR EACH BALL BEARING IN SECONDS

Model	t-B1 (s)	t-B2 (s)	t-B3 (s)	t-B4 (s)
AsHMM	551.57	127.48	144.31	220.75
FS-HMM	24.25	48.00	46.02	**15.57**
FS-AsHMM	27.64	57.28	108.91	78.19
SHMM-LFS	**16.29**	**16.49**	**15.85**	19.14

third harmonics. In addition, the BSF fundamental depends on the third FTF harmonic, and the BPFO fundamental relies on the first BSF harmonic, which also depends on the FTF fundamental. From this kind of information, it is plausible to say that, in a low degradation state, for this bearing, the cage of the ball bearing is leading the dynamical system. Fig. 12(b) shows another learned context-specific Bayesian network from a more degraded ball-bearing state. In this case, more variables and dependencies appear in the graph; in particular, AR variables are considered. For example, three and four AR values of the BPFI and FTF fundamental frequencies, respectively, appear and can be helpful to describe the late degradation stages of the ball bearing.

Regarding the learning times for this application, they are displayed in Table XV. In this case, we observe that the slowest algorithm was AsHMM, followed by FS-AsHMM, FS-HMM, and SHMM-LFS. As in the previous datasets, SHMM-LFS was the fastest algorithm in most scenarios, but it captured little relevant information from the data. On the other hand, note that FS-AsMM had big differences in the learning times compared to FS-HMM. We must recall that B3 and B4 broke, and therefore, their dynamic behavior is more complex compared to B1 and B2. This is evidenced by looking at Table XIII where B3 and B4 models for FS-AsHMM obtained more parameters than B1 and B2 FS-AsHMM models.

VI. Conclusion and Future Work

In this article, we have introduced a new model that extends As-HMMs. This new model is capable of estimating the relevant features and local-optimal context-specific Bayesian networks for the selected relevant features, all during their learning phase. The parameter learning procedure for each model is detailed and proved. Also, a restriction to the space of context-specific Bayesian networks is imposed in order to consider only relevant features in the graph construction.

Experiments with synthetic data and real data from grammar facial expressions and ball-bearing wearing data are considered. For the experiments, another two other state-of-the-art models were considered for comparative purposes, namely, FS-HMM [7] and SHMM-LFS [8]. For the latter model, a theoretic argument was given to validate its little usefulness to detect relevant features in the experiments. In addition, these two models consider that all variables are independent, which, in many real case scenarios, is not true.

From the experiments, we observed that the proposed model can obtain fair results in fitness. In the case of synthetic data, we observed that the proposed models are capable of detecting irrelevant features. When evaluating grammar facial expressions, FS-AsHMM and FS-HMM obtained good results in accuracy and F-score, in spite of the fact that AsHMM obtained better results in fitness. In addition, the proposed model was capable of determining which variables were noise and not useful for prediction.

From the algorithm complexity point of view, big O notation bounds were provided for the different routines that the algorithm can perform. On the other hand, the learning times for the proposed algorithm were among the highest for all the datasets. However, it was observed that the times were shorter when the data were simpler (see the ball bearing case). This property caused high variance in the obtained times, which indicates that the model is capable of giving further data insights when needed at the cost of higher computational times; otherwise, the times are closer to the ones obtained by simpler models.

In conclusion, although models such as AsHMM can obtain better results in BIC and LL than the FS models, when noise variables are present, the learning and testing performance of AsHMM can be senseless: it is learning and predicting noise. When feature saliencies are added to the model, these were capable of discriminating between noise and relevant features. In addition, it was proven that the proposed model was capable of generating context-specific Bayesian networks at the same time as the feature saliencies were estimated. In this manner, the restriction of total independence in FS models is overcome.

We are aware that the Gaussian hypotheses may not be suitable for many dynamic data scenarios. Therefore, as future work, we would like to explore the application of nonparametric distributions inside the context-specific Bayesian networks. In this manner, the models can be more precise and avoid this issue.

References

[1] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, Oct. 2007.

[2] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1154–1166, Sep. 2004.

[3] R. Kohavi and G. John, "Wrappers for feature selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, Dec. 1997.

[4] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Readings in Speech Recognition*. San Mateo, CA, USA: Morgan Kaufmann, 1990, pp. 267–296.

[5] C. Puerto-Santana, P. Larra naga, and C. Bielza, "Autoregressive asymmetric linear Gaussian hidden Markov models," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–17, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9387117

[6] M. L. Bueno, A. Hommersom, P. J. Lucas, and A. Linard, "Asymmetric hidden Markov models," *Int. J. Approx. Reasoning*, vol. 88, pp. 169–191, Sep. 2017.

[7] S. Adams, A. P. Beling, and R. Cogill, "Feature selection for hidden Markov models and hidden semi-Markov models," *IEEE Access*, vol. 4, pp. 1642–1657, 2016.

[8] Y. Zheng, B. Jeon, L. Sun, J. Zhang, and H. Zhang, "Student's t-hidden Markov model for unsupervised learning using localized feature selection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2586–2598, Oct. 2018.

[9] S. Adams and P. A. Beling, "Feature selection for hidden Markov models with discrete features," in *Intelligent Systems and Applications* (Advances in Intelligent Systems and Computing). Cham, Switzerland: Springer, 2020, pp. 67–82.

[10] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller, "Context-specific independence in Bayesian networks," in *Proc. 25th Int. Conf. Uncertainty Artif. Intell. (UAI)*. San Mateo, CA, USA: Morgan Kaufmann, 1996, pp. 115–123.

[11] M. A. Hall, "Correlation-based feature selection for machine learning," M.S. thesis, Dept. Comput. Sci., Univ. Waikato, Hamilton, New Zealand, 1999.

[12] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Proc. Nat. Conf. Artif. Intell.*, vol. 2, 1992, pp. 129–134.

[13] H. Zhou, M. You, L. Liu, and C. Zhuang, "Sequential data feature selection for human motion recognition via Markov blanket," *Pattern Recognit. Lett.*, vol. 86, pp. 18–25, Jan. 2017.

[14] F. Lv and R. Nevatia, "Recognition and segmentation of 3-D human action using HMM and multi-class Adaboost," in *Proc. Eur. Conf. Comput. Vis.*, Berlin, Germany: Springer, 2006, pp. 359–372.

[15] M. Momenzadeh, M. Sehhati, and H. Rabbani, "A novel feature selection method for microarray data classification based on hidden Markov model," *J. Biomed. Informat.*, vol. 95, Jul. 2019, Art. no. 103213.

[16] M. Dash and Y.-S. Ong, "RELIEF-C: Efficient feature selection for clustering over noisy data," in *Proc. IEEE 23rd Int. Conf. Tools Artif. Intell.*, Nov. 2011, pp. 869–872.

[17] W. Yue, Y. S. Wong, and G. S. Hong, "Adaptive-VDHMM for prognostics in tool condition monitoring," in *Proc. 6th Int. Conf. Autom., Robot. Appl. (ICARA)*, Feb. 2015, pp. 131–136.

[18] M. M. M. Farag, T. Elghazaly, and H. A. Hefny, "Face recognition system using HMM-PSO for feature selection," in *Proc. 12th Int. Comput. Eng. Conf. (ICENCO)*, Dec. 2016, pp. 105–110.

[19] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc., B, Methodol.*, vol. 58, no. 1, pp. 267–288, 1996.

[20] R. Setiono and H. Liu, "Neural-network feature selector," *IEEE Trans. Neural Netw.*, vol. 8, no. 3, pp. 654–662, May 1997.

[21] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 2. Cambridge, MA, USA: MIT Press, 2014, pp. 2204–2212.

[22] H. Zhu, Z. He, and H. Leung, "Simultaneous feature and model selection for continuous hidden Markov models," *IEEE Signal Process. Lett.*, vol. 19, no. 5, pp. 279–282, May 2012.

[23] Y. Li, M. Dong, and J. Hua, "Simultaneous localized feature selection and model detection for Gaussian mixtures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 953–960, May 2009.

[24] L. Guerra, C. Bielza, V. Robles, and P. Larrañaga, "Semi-supervised projected model-based clustering," *Data Mining Knowl. Discovery*, vol. 28, no. 4, pp. 882–917, Jul. 2014.

[25] T. M. Nguyen, Q. M. J. Wu, and H. Zhang, "Asymmetric mixture model with simultaneous feature selection and model detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 400–408, Feb. 2015.

[26] Z. Song, S. Ali, and N. Bouguila, "Bayesian inference for infinite asymmetric Gaussian mixture with feature selection," *Soft Comput.*, vol. 25, no. 8, pp. 6043–6053, Apr. 2021.

[27] J. A. Bilmes, "Buried Markov models for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, Mar. 1999, pp. 713–716.

[28] S. Kirshner, S. Padhraic, and R. Andrew, "Conditional Chow-Liu tree structures for modeling discrete-valued vector time series," in *Proc. 20th Conf. Uncertainty Artif. Intell.* Arlington, VA, USA: AUAI Press, 2004, pp. 317–324.

[29] N. Städler and S. Mukherjee, "Penalized estimation in high-dimensional hidden Markov models with state-specific graphical models," *Ann. Appl. Statist.*, vol. 7, no. 4, pp. 2157–2179, 2013.

[30] I. Jolliffe, "Generalizations and adaptations of principal component analysis," in *Principal Component Analysis*. Cham, Switzerland: Springer, 1986, pp. 223–234.

[31] X. Chang, F. Nie, S. Wang, Y. Yang, X. Zhou, and C. Zhang, "Compound rank-*k* projections for bilinear analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1502–1513, Jul. 2016.

[32] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, "HITON: A novel Markov Blanket algorithm for optimal variable selection," in *Proc. AMIA Annu. Symp.*, vol. 2003, 2003, pp. 21–25.

[33] C. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," *Comput. Graph.*, vol. 21, no. 4, pp. 24–34, 1987.

[34] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc., B, Methodol.*, vol. 39, no. 1, pp. 1–38, 1977.

[35] N. Friedman, "The Bayesian structural EM algorithm," in *Proc. 14th Conf. Uncertainty Artif. Intell.* San Mateo, CA, USA: Morgan Kaufmann, 1998, pp. 129–138.

[36] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, 1978.

[37] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006.

[38] F. Freitas, S. Peres, C. Lima, and F. Barbosa, "Grammatical facial expressions recognition with machine learning," in *Proc. 27th Florida Artif. Intell. Res. Soc. Conf. (FLAIRS)*, Aug. 2014, pp. 180–185.

[39] N. Michael, D. Metaxas, and C. Neidle, "Spatial and temporal pyramids for grammatical expression recognition of American sign language," in *Proc. 11th Int. ACM SIGACCESS Conf. Comput. Accessibility*, 2009, pp. 75–82.

[40] T. D. Nguyen and S. Ranganath, "Facial expressions in American sign language: Tracking and recognition," *Pattern Recognit.*, vol. 45, no. 5, pp. 1877–1891, May 2012.

[41] P. Larrañaga, D. Atienza, J. Diaz-Rozo, A. Ogbechie, C. Puerto-Santana, and C. Bielza, *Industrial Applications of Machine Learning*. Boca Raton, FL, USA: CRC Press, 2018.

[42] Y. Wang and M. Liang, "An adaptive SK technique and its application for fault detection of rolling element bearings," *Mech. Syst. Signal Process.*, vol. 25, no. 5, pp. 1750–1764, Jul. 2010.

[43] J. C. Jáuregui-Correa and A. Lozano-Guzman, *Mechanical Vibrations and Condition Monitoring*. New York, NY, USA: Academic, 2020.

[44] H. Qiu, J. Lee, J. Lin, and G. Yu, "Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics," *J. Sound Vibrat.*, vol. 289, nos. 4–5, pp. 1066–1090, 2006.

Carlos Puerto-Santana received the bachelor's degree in mathematics from the Universidad de los Andes, Bogotá, Colombia, in 2016, and the master's degree in artificial intelligence from the Universidad Politécnica de Madrid, Madrid, Spain, in 2018, where he is currently pursuing the Ph.D. degree.

He is also a Researcher and a Developer at Aingura IIoT, Donostia-San Sebastian, Spain.

Pedro Larrañaga (Member, IEEE) received the M.Sc. degree in mathematics (statistics) from the University of Valladolid, Valladolid, Spain, in 1982, and the Ph.D. degree in computer science from the University of the Basque Country, Leioa, Spain, in 1995.

He earned the Habilitation Qualification for Full Professor in 2003. He has been a Full Professor of computer science and artificial intelligence with the Universidad Politécnica de Madrid (UPM), Madrid, Spain, since 2007. Before moving to UPM, his academic career was developed at the University of the Basque Country (UPV-EHU) at several faculty ranks: an Assistant Professor from 1985 to 1998, an Associate Professor from 1998 to 2004, and a Full Professor from 2004 to 2007. He has published more than 200 papers in impact factor journals and has supervised 33 Ph.D. theses. His research interests are primarily in the areas of probabilistic graphical models, metaheuristics for optimization, data mining, classification models, and real applications, such as biomedicine, bioinformatics, neuroscience, industry 4.0, and sports.

Dr. Larrañaga has been a fellow of the European Association for Artificial Intelligence since 2012 and the Academia Europea since 2018. He was awarded the 2013 Spanish National Prize in Computer Science and the prize of the Spanish Association for Artificial Intelligence in 2018. He received the Excellence Award for his Ph.D. degree.

Concha Bielza (Member, IEEE) received the M.S. degree in mathematics from the Universidad Complutense de Madrid, Madrid, Spain, in 1989, and the Ph.D. degree in computer science from the Universidad Politécnica de Madrid, Madrid, in 1996.

Since 2010, she has been a Full Professor of statistics and operations research with the Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid. She has published more than 150 papers in impact factor journals and has supervised 20 Ph.D. theses. Her research interests are primarily in the areas of probabilistic graphical models, decision analysis, metaheuristics for optimization, classification models, temporal models, and real applications, such as biomedicine, bioinformatics, neuroscience, and industry.

Dr. Bielza was awarded the 2014 UPM Research Prize and the 2020 Machine Learning Award of Amity University (India). She received the Extraordinary Doctorate Award for her Ph.D. degree.