Artif Intell Rev
DOI 10.1007/s10462-017-9581-3

A survey of feature selection methods for Gaussian mixture models and hidden Markov models

Stephen Adams[1] · **Peter A. Beling**[1]

Abstract Feature selection is the process of reducing the number of collected features to a relevant subset of features and is often used to combat the curse of dimensionality. This paper provides a review of the literature on feature selection techniques specifically designed for Gaussian mixture models (GMMs) and hidden Markov models (HMMs), two common parametric latent variable models. The primary contribution of this work is the collection and grouping of feature selection methods specifically designed for GMMs and for HMMs. An additional contribution lies in outlining the connections between these two groups of feature selection methods. Often, feature selection methods for GMMs and HMMs are treated as separate topics. In this survey, we propose that methods developed for one model can be adapted to the other model. Further, we find that the number of feature selection methods for GMMs outweighs the number of methods for HMMs and that the proportion of methods for HMMs that require supervised data is larger than the proportion of GMM methods that require supervised data. We conclude that further research into unsupervised feature selection methods for HMMs is required and that established methods for GMMs could be adapted to HMMs. It should be noted that feature selection can also be referred to as dimensionality reduction, variable selection, attribute selection, and variable subset reduction. In this paper, we make a distinction between dimensionality reduction and feature selection. Dimensionality reduction, which we do not consider, is any process that reduces the number of features used in a model and can include methods that transform features in order to reduce the dimensionality. Feature selection, by contrast, is a specific form of dimensionality reduction that eliminates feature as inputs into the model. The primary difference is that dimensionality reduction can still require the collection of all the data sources in order to transform and reduce the feature set, while feature selection eliminates the need to collect the irrelevant data sources.

✉ Stephen Adams
sca2c@virginia.edu

Peter A. Beling
pb3a@virginia.edu

[1] University of Virginia, 151 Engineer's Way, Charlottesville, VA 22904, USA

Published online: 25 September 2017

Keywords Feature selection · Gaussian mixture model · Hidden Markov model

1 Introduction

Gaussian mixture models (GMMs) and hidden Markov models (HMMs) are two common parametric latent variable models. GMMs are often used for clustering or modeling multimodal data. HMMs, on the other hand, are often used for modeling time series data. However, both of these models have similar traits in that the distribution of the observed or collected data is dependent upon a hidden or latent random variable. In the case of GMMs, the latent variables are independent and identically distributed, while the latent variable corresponding to a specific observation for an HMM is dependent upon the previous latent variable in the sequence. The successful application of GMMs and HMMs, domains ranging from speech recognition to financial data analytics, has led to a corresponding explosion in the amount and variety of data being collected for analysis using these methods. This growth in data is not limited to the number of observations but extends to the type of information collected or features.

Generally, the more information that can be observed the better a machine learning method will perform. However, there are a broad set of issues, collectively known as the *curse of dimensionality*, associated with high-dimensional data. Training times and algorithmic complexity, storage space requirements, and the noise in data sets all may scale poorly with the number of features. Many pattern recognition and machine learning techniques have trouble dealing with a large number of irrelevant features. Both GMMs and HMMs can handle high-dimensional data. However, increasing the dimensionality of the data, or adding more features to the data set, can decrease the accuracy of a model by introducing noise, a phenomenon known as *peaking* (Jain et al. 2000). Further, the number of observations needed to accurately train a model increases with the number of dimensions or features due to class conditional sparsity.

Feature selection is the process of reducing the number of collected features to a relevant subset of features and is often used to combat the curse of dimensionality. Feature selection can increase the performance of models by eliminating noise in the data, increase the training and prediction speed of the model, improve model interpretation, decrease the risk of overfitting, and decrease the cost of a system by eliminating the need to collect certain features. Feature selection has been widely studied in multiple fields and for various models (Dash and Liu 1997; Guyon and Elisseeff 2003; Liu and Yu 2005). For example, in the field of bioinformatics where high-dimensional data is prevalent, feature selection has become a prerequisite for constructing mathematical models (Saeys et al. 2007). There is a clear and growing need for feature selection methods as the complexity of collected data grows. In 2003, at the dawn of the big data age, Guyon and Elisseeff (2003) wrote a review of feature selection techniques and stated that most of the papers they cite deal with hundreds to tens of thousands of features. The number of features available in most domains has surely grown since then as a result of the many successful applications of artificial intelligence and machine learning and in correspondence with the capabilities of systems and sensors for collecting contextual and transactional data.

Extensive research has been performed on feature selection for clustering and the specific case of GMMs. However, the work on feature selection for HMMs is limited. In this paper, we survey and review the literature on feature selection techniques specifically designed for GMMs and HMMs. We cover both GMMs and HMMs because of the similar nature of

these two models and that techniques designed for one can often be applied to the other. General feature selection techniques can be applied to these models. However, there are two challenges that must be considered when selecting a feature selection technique to be used with these models. First, both GMMs and HMMs are latent variable models, and the latent variables might not be available in the collected data set. Therefore, an unsupervised feature selection technique (Dy 2008), which can estimate the set of relevant features without knowledge about the latent variables, could be required. Second, if unsupervised methods are used for feature selection, they are often iterative and highly computational. Feature selection can be a highly computational process when all the information is known, and this trait often limits the feature selection process in application. Therefore, a highly computational feature selection technique is often undesirable. Feature selection methods developed specifically for GMMs and HMMs often take into account these challenges. Further, the specific feature selection techniques often outperform general techniques that can be applied to any type of model.

Given the vast amount of general feature selection methods and the large number of methods specific to GMMs and HMMs, a survey of the feature selection methods for these models is needed. The primary contribution of this work is the collection and grouping of feature selection methods specifically designed for GMMs and for HMMs. An additional contribution lies in outlining the connections between these two groups of feature selection methods. Often, feature selection methods for GMMs and HMMs are treated as separate topics. In this survey, we propose that methods developed for one model can be adapted to the other model. For example, the feature saliency feature selection technique was originally developed for GMMs but was later extended to HMMs. Further, this survey finds that the number of feature selection methods for GMMs outweighs the number of methods for HMMs and that the proportion of methods for HMMs that require supervised data is larger than the proportion of GMM methods that require supervised data. We conclude that further research into unsupervised feature selection methods for HMMs is required and that established methods for GMMs could be adapted to HMMs.

In this survey, we primarily focus on Gaussian mixture models and HMMs that have a Gaussian observation distribution. Other distributions, such as the gamma distribution or the Student's t-distribution, could be used in place of the Gaussian for both models. Further, HMMs are often modeled with discrete observations and non-parametric observation distributions. Our review of the literature on feature selection techniques for these models reflects the popularity of the Gaussian distribution. However, we believe that most of the methods discussed in this review could be easily extended to models with non-Gaussian distributions.

Feature selection can also be referred to as dimensionality reduction, variable selection, attribute selection, and variable subset reduction. In this paper, we would like to make a distinction between dimensionality reduction and feature selection. Dimensionality reduction is any process that reduces the number of features used in a model and can include methods that transform features in order to reduce the dimensionality. Feature selection is a specific form of dimensionality reduction that eliminates feature as inputs into the model. The primary difference is that dimensionality reduction can still require the collection of all the data sources in order to transform and reduce the feature set, while feature selection eliminates the need to collect the irrelevant data sources. Principal component analysis (PCA) is an example of dimensionality reduction that we do not consider feature selection.

This paper is organized as follows. Section 2 outlines GMMs and HMMs. Section 3 describes feature selection and general feature selection methods. Section 4 describes the feature selection methods specific to GMMs, and Sect. 5 describes the feature selection

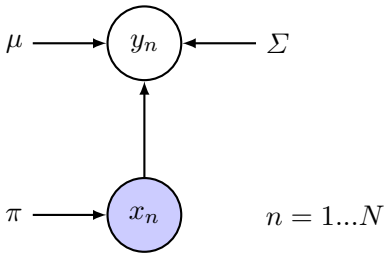

Fig. 1 Graphical model of a GMM

methods specific to HMMs. Finally, Sect. 6 concludes with a discussion and outlines needed research areas.

2 Gaussian mixture models and hidden Markov models

In this section, we review the technical aspects of GMMs and HMMs. GMMs are used for clustering unsupervised data and detect hidden structure among the individual data points. Clustering algorithms are often divided into two types: distance-based and model-based. The former method clusters data using a distance metric while the latter fits a predefined model to the data. GMMs are a model-based clustering approach that assumes each cluster has a multivariate normal distribution. The cluster assignment is considered a random variable. Model-based clustering offers the advantage of a framework for assessing the number of clusters and the significance of each variable in the clustering process (Maugis et al. 2009a).

Similar to GMMs, HMMs are probabilistic models consisting of two sets of correlated random variables. The first set is a hidden state sequences X, which is modeled as a Markov chain. The second set consists of the observations or emissions Y, which are modeled with state-conditional distributions. The emission distribution is often represented by a Gaussian distribution; but it can take on any parametric or non-parametric distribution and be either continuous or discrete. These models are widely applied in finance, speech recognition, medicine, activity recognition, and manufacturing. In some cases, the observations for HMMs are modeled as GMMs.

2.1 GMMs

GMMs are parametric models used for clustering data. Given N observations and I mixtures, let $X = \{x_1, \ldots, x_N\}$ represent the observation's cluster or mixture assignment random variable. Let $Y = \{y_1, \ldots, y_N\}$ represent the multivariate data where y_{ln} is the lth component of the nth observation. Specifically for GMMs, we assume that each mixture emits the data from a multivariate Gaussian $\mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i)$, where $\boldsymbol{\mu}_i$ is the mixture dependent mean vector and Σ_i is the mixture dependent covariance matrix. A graphical model for a GMM is displayed in Fig. 1. The likelihood of observing the data given the model is

$$P(Y|\Lambda) = \prod_{n=1}^{N} \sum_{i=1}^{I} \pi_i \mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i), \tag{1}$$

where $\pi_i = P(x = i)$ and Λ is the set of model parameters $\{\pi_1, \ldots, \pi_I, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_I, \Sigma_1, \ldots, \Sigma_I\}$. If the covariance matrix is diagonal, which represents the assumption that each feature is independent, the likelihood can be rewritten as

$$P(Y|\Lambda) = \prod_{n=1}^{N} \sum_{i=1}^{I} \pi_i \prod_{l=1}^{L} \mathcal{N}(y_{ln}|\mu_{il}, \sigma_{il}^2), \tag{2}$$

where σ_{il}^2 is the variance of the lth feature of the ith mixture.

Let $\mathfrak{x}_{in}$ indicate if the nth observation belongs to the ith cluster, i.e $\mathfrak{x}_{in} = 1$ if $x_n = i$ and $\mathfrak{x}_{in} = 0$ otherwise. Using this indicator, the joint probability of the mixture random variable X and the observations Y can be written as

$$P(X, Y|\Lambda) = \prod_{n=1}^{N} \prod_{i=1}^{I} [\pi_i \mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i)]^{\mathfrak{x}_{in}}. \tag{3}$$

In many cases, the expectation-maximization (EM) algorithm is used to solve for the parameters in the model (Bilmes 1998). In the E-step, calculate the probability that each observation belongs to the ith mixture using

$$P(x_n = i|y_n, \Lambda) = \frac{\pi_i \mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i)}{\sum_{i=1}^{I} \pi_i \mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i)}. \tag{4}$$

In the M-step, update the parameter estimates using

$$\pi_i = \frac{1}{N} \sum_{n=1}^{N} P(x_n = i|y_n, \Lambda), \tag{5}$$

$$\boldsymbol{\mu}_i = \frac{\sum_{n=1}^{N} y_n P(x_n = i|y_n, \Lambda)}{\sum_{n=1}^{N} P(x_n = i|y_n, \Lambda)}, \tag{6}$$

and

$$\Sigma_i = \frac{\sum_{n=1}^{N} P(x_n = i|y_n, \Lambda)(y_n - \boldsymbol{\mu}_i)(y_n - \boldsymbol{\mu}_i)^T}{\sum_{n=1}^{N} P(x_n = i|y_n, \Lambda)}. \tag{7}$$

There are numerous methods for estimating the parameters of a GMM including variational Bayesian methods (Corduneanu and Bishop 2001) and Bayesian sampling methods (Frühwirth-Schnatter 2001; Jasra et al. 2005; Richardson and Green 1997).

2.2 HMMs

HMMs (Rabiner 1989) are parametric models used for representing time series data. Two sequences of random variables are represented by a joint probability distribution. The hidden state sequence $X = \{x_1, \ldots, x_T\}$ is modeled as a Markov chain and follows the Markovian property that the current state is solely based on the previous state. The Markov chain has parameters π representing the initial state distribution and a_{ij} representing the transition probability between state i and j. The transition probabilities are collected into the transition matrix A. At each time step t, an observable emission $Y = \{y_1, \ldots, y_T\}$ is generated from a state dependent distribution. When there are L features, the lth feature is represented by y_{lt}. For an HMM with I states, the state dependent distribution is represented by $p(y_t|x_t = i, \theta)$ where θ represents the parameters of the distribution. The set of all model parameters for the HMM is represented by $\Lambda = \{\pi, A, \theta\}$. A graphical model for an HMM is displayed in Fig. 2.

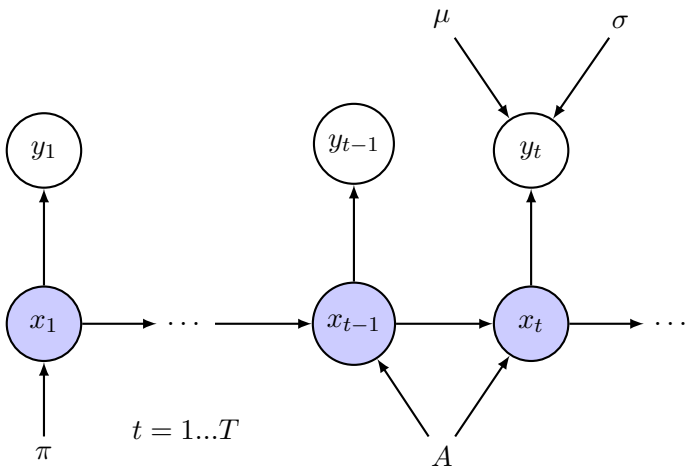

Fig. 2 Graphical model of an HMM

The joint probability for the state sequence X and the emission sequence Y is given by

$$P(X, Y|\Lambda) = \pi_{x_1} f_{x_1}(y_1) \prod_{t=2}^{T} a_{x_{t-1}, x_t} f_{x_t}(y_t), \tag{8}$$

where $f_{x_t}(y_t)$ is the emission distribution given $x_t = i$. The probability for the emission sequence can be found by summing Equation (8) over all possible state sequences.

The Baum–Welch algorithm (Bilmes 1998; Rabiner 1989) is the most common algorithm used for unsupervised parameter estimation and is the EM algorithm applied to HMMs. In the expectation step, the forward and backward probabilities are calculated

$$\alpha_t(i) = p(y_1, y_2, \ldots, y_t, x_t = i|\Lambda), \tag{9}$$

and

$$\beta_t(i) = p(y_{t+1}, y_{t+2}, \ldots, y_T|x_t = i, \Lambda). \tag{10}$$

These probabilities are used to calculate the posterior probabilities for the state and the transitions

$$\begin{aligned} \gamma_t(i) &= P(x_t = i, |Y, \Lambda) \\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{I} \alpha_t(i)\beta_t(i)}, \end{aligned} \tag{11}$$

and

$$\begin{aligned} \xi_t(i, j) &= P(x_t = i, x_{t+1} = j|Y, \Lambda) \\ &= \frac{\alpha_t(i) a_{ij} f_j(y_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{I} \sum_{j=1}^{I} \alpha_t(i) a_{ij} f_j(y_{t+1}) \beta_{t+1}(j)}. \end{aligned} \tag{12}$$

In the M-step, the Markov chain parameters are updated using

$$\pi_i = \gamma_1(i), \tag{13}$$

and

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \sum_{j=1}^{I} \xi_t(i,j)} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \tag{14}$$

When the emission distribution is assumed to be Gaussian and the features are assumed to be conditionally independent, the mean μ and standard deviation σ are updated using the following equations

$$\mu_{il} = \frac{\sum_{t=1}^{T} \gamma_t(i) y_{lt}}{\sum_{t=1}^{T} \gamma_t(i)}, \tag{15}$$

and

$$\sigma_{il} = \sqrt{\frac{\sum_{t=1}^{T} \gamma_t(i)(y_{lt} - \mu_{il})^2}{\sum_{t=1}^{T} \gamma_t(i)}}. \tag{16}$$

Other popular methods for estimating the parameters of an HMM given an observation sequence include variational Bayes estimation (Chatzis and Kosmopoulos 2011; Ji et al. 2006; McGrory and Titterington 2009; Wei and Li 2011), descent methods (Bagos et al. 2004; Cappé et al. 1998), Bayesian sampling methods such as Markov chain Monte Carlo or Gibbs sampling (Boys and Henderson 2001; Robert et al. 2000; Rydén 2008; Scott 2002), and maximum mutual information estimation (Bahl et al. 1986; Merialdo 1988). Online or incremental learning algorithms for HMMs are surveyed by in Khreich et al. (2012).

The Viterbi algorithm (Rabiner 1989) can be used to predict the most likely state sequence given an emission sequence. If classes or labels are mapped to the states of an HMM, the Viterbi algorithm can be used for classification and prediction. Another formulation of a classification problem using HMM maps a class or label to an individual HMM. In this formulation, a different HMM is trained for each class. During prediction, the likelihood that the observed data was generated from each HMM is calculated. The observed sequence is labeled with the class corresponding to the HMM with the highest likelihood of generating the observed data.

3 Feature selection literature review

Feature selection is the process of reducing the set of collected features $\mathcal{L}$ to a subset of relevant features $\mathcal{L}'$; see Fig. 3 for a visual representation of the feature selection process. Feature selection is also called variable selection, attribute selection, or feature subset selection. Feature selection speeds the learning process, improves model interpretation, reduces the risk of overfitting, and alleviates the effects of the curse of dimensionality. When the feature set is

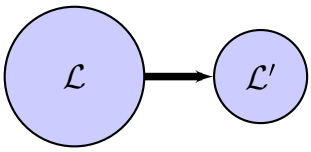

Fig. 3 Feature selection reduces the number of collected features $\mathcal{L}$ to a relevant subset of features $\mathcal{L}'$

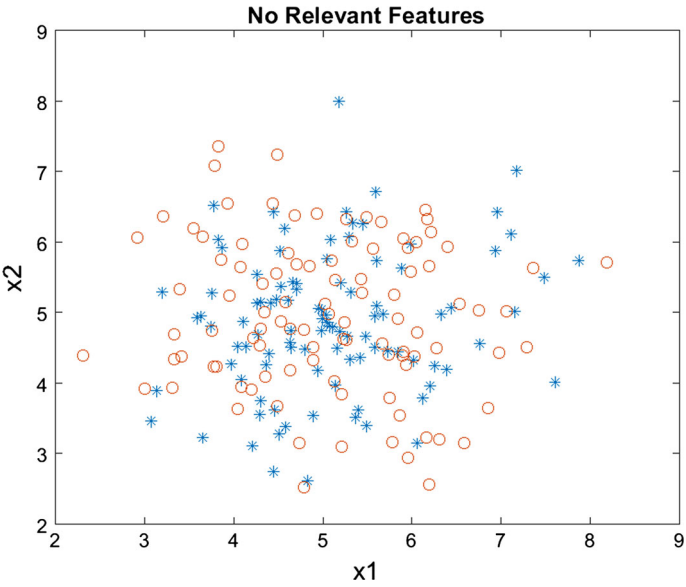

Fig. 4 Two-class example with no relevant features

small, an exhaustive search can be performed. As the number of features grows, this method becomes impractical, because the number of possible feature subsets grows exponentially.

As an example, suppose that two features can be collected on a two-class problem. Let the two features be designated by $x1$ and $x2$. In Fig. 4, neither feature is relevant. In Fig. 5, $x1$ is a relevant feature and $x2$ is irrelevant. In Fig. 6, both features are relevant, however only a single feature is needed to distinguish between the two classes. Thus, $x1$ and $x2$ are considered redundant features. Figure 7 shows an example of the two-class problem where both features are needed to adequately distinguish between classes. A good feature selection algorithm should be able to identify the single relevant feature in Fig. 5 and select both features in Fig. 7.

Feature extraction is a separate problem from feature selection. Feature selection identifies relevant features from a candidate set of features, while feature extraction calculates new features from a given set. Feature selection does not alter the original features, while extraction generates new features. Dimensionality reduction is usually applied to extracted features. Principal component analysis (PCA) (Murphy 2012) is a feature extraction method. Dimensionality reduction is performed by selecting the first m principal components. This method differs from feature selection, however, because all input features must still be collected in order to extract the new principal components. Conversely, after feature selection has been performed, the irrelevant features no longer need to be collected. Further, the selected subset of features can give insight into the process and be interpreted by a domain expert. However, methods such as PCA reduce the noise in the extracted features, and can provide more discriminating features than the original raw features. Independent component analysis (ICA) (Murphy 2012) is another form of feature extraction similar to PCA.

John et al. (1994) and Kohavi and John (1997) outline four definitions for relevant features that were current in the literature in the mid 1990's. Using a correlated XOR problem, they show that different definitions can lead to the selection of different feature subsets, and, after arguing that definitions of strong and weak relevance are necessary, they provide such.

Fig. 5 Two-class example with one relevant features

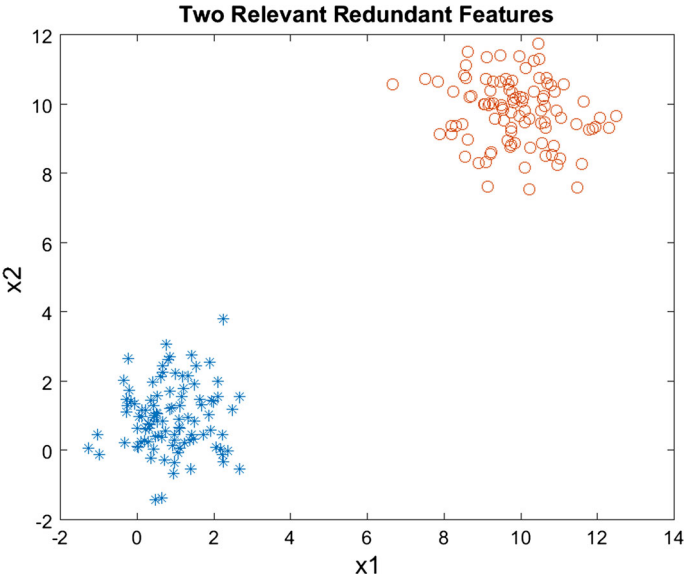

Fig. 6 Two-class example with two relevant redundant features

However, Kohavi and John (1997) go on to show, using an example, that relevant features are not always included in the optimal feature set and irrelevant features are not always excluded from the optimal feature set. Blum and Langley (1997) provide definitions of feature relevance that include a target concept and a measure of complexity of the selected feature subset. Blum and Langley's final definition of relevance, which first appeared in

Fig. 7 Two-class example with two relevant features

Caruana and Freitag (1994), introduces the idea of relevance with respect to a specific learning algorithm.

Guyon and Elisseeff (2003) lay out several steps for general feature selection. These are in the form of questions, and the answers lead to specific actions to be taken or a particular feature selection method to be used. For example, their fifth question is, "Do you need to assess features individually?" If the answer is yes, a procedure that ranks features should be used.

Molina et al. (2002) test and compare 10 feature selection algorithms that require the use of supervised data. They characterize each algorithm based on search optimization (exploring the feature space), generation of successors (selecting the next feature to add or subtract from the proposed feature set), and the evaluation measure. The authors evaluate each algorithm based on the number of relevant or irrelevant features included in the final feature subset, as well as the number of redundant features included in the final subset. Their tests demonstrate that the algorithm that performs the best is highly dependent upon the data, leading to the conclusion that there is no optimal feature selection algorithm for all data sets.

Feature selection methods can be divided into three groups: *filters*, *wrappers*, and *embedded methods*. Filters assess feature relevance by investigating the feature's properties. They address the problems of selecting features and building models independently. Wrappers assess feature relevance with regard to a specific learning algorithm. In most cases, a model is built with respect to a subset of features and the model's performance is evaluated based on specified criteria. Wrappers then move through the subset space evaluating feature subsets with regard to the evaluation function. Embedded methods simultaneously select features and construct models. Figure 8 gives visual displays of each of these methods. There are numerous feature selection methods that can be applied to any type of model, therefore we briefly outline and group a selection of popular feature methods that could be applied to GMMs and HMMs but are not specific to these models.

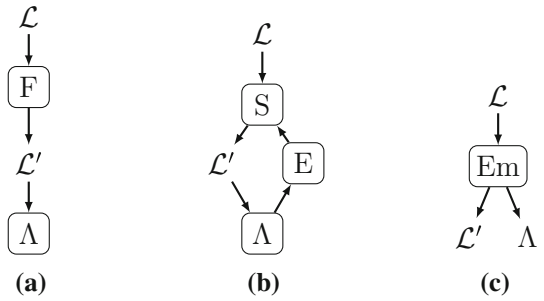

Fig. 8 Types of feature selection methods. **a** The filter F takes as inputs the set of all features $\mathcal{L}$ and reduces it to the relevant subset $\mathcal{L}'$ independent of the model parameters Λ, which are constructed or estimated only on $\mathcal{L}'$. **b** The wrapper first selects a subset of features $\mathcal{L}'$ using the a search algorithm S, estimates the model parameters Λ on $\mathcal{L}'$, and then evaluates $\mathcal{L}'$ the using an evaluation function E. This process continues until a stopping criterion is met. **c** The embedded method Em takes the full feature set $\mathcal{L}$ and outputs both the reduced set $\mathcal{L}'$ and the model parameters Λ

3.1 General filters

Filters treat feature selection as a preprocessing step and select features with no regard to the model, i.e. filters only consider the properties of the collected features and how they distinguish themselves from one another or how they relate to a target class label. These methods are generally fast in terms of computation, but can result in feature subsets that do not yield satisfactory predictive accuracy. Certain types of filters can be applied to unsupervised data. In addition, filters usually scale well to the number of features. Under certain conditions, e.g. if supervised data is available, the filters described in this section can be applied to feature sets before training GMMs or HMMs.

The simplest filtering technique is to select features based on their correlation with the class or continuous response variable. More complex filters include the FOCUS (Almuallim and Dietterich 1991) and Relief (Kira and Rendell 1992) algorithms. Extensions of Relief (Kononenko 1994; Robnik-Šikonja and Kononenko 2003) can be applied to multi-class problems and unsupervised data. Feature-similarity feature selection (Mitra et al. 2002) is an unsupervised filter that groups the k closest features. A single feature from each group is chosen to represent the group in the reduced feature subset. The idea behind this method is that features that are similar by some metric are redundant and can be removed with insignificant change in prediction accuracy. Some filters rank or assign weights to features. Correlation-based methods (Yu and Liu 2003) rank features based on their association with a class label. Forman (2003) compares several metrics for ranking features. The evaluation is specific to text classification, so results may not generalize to all classification problems. The author concludes that the bi-normal separation metric outperforms other filters on the chosen data sets.

Bins and Draper (2001) propose using both filters and a traditional wrapper in a three-stage feature selection method. The first stage removes the irrelevant features using a filter. The second stage removes the redundant features also using a filter. The first two stages reduce the number of features to a point where a traditional wrapper can be applied without significant computational cost. The authors suggest using either forward or backward search depending on how many features remain after the first two stages.

3.2 General wrappers

Wrappers test feature subsets given a model and an evaluation function. These methods are more computationally expensive than filters but take the model into account and often yield feature subsets with better predictive ability. However, wrappers generally require the use of supervised data for the evaluation function. In a wrapper approach, data are typically divided into three groups: training, evaluation, and testing. A model using a subset of the candidate features is trained on the training set, then evaluated using the evaluation set. The feature subset is then augmented in some fashion and the process repeated. The feature subset that optimizes the evaluation function is chosen as the final feature subset and tested on the withheld testing set. When data are scarce, the evaluation set can be eliminated by evaluating the feature subset on the training data. However, this can cause a poor generalization error and increase the likelihood of overfitting to the training data. Cross validation can be used in the case of small data sets.

Sequential forward search and sequential backward search are two types of exploration algorithms (John et al. 1994; Kohavi and John 1997). These are considered greedy algorithms, as opposed to an exhaustive search of the feature subsets. Sequential search methods are often referred to as hill-climbing strategies, because they look for improvement in the evaluation function. However, a non-exhaustive search cannot guarantee the optimal feature subset (Cover and Campenhout 1977). Aha and Bankert (1995) compare forward and backward search algorithms to filters. They found that these wrappers, and some of the tested variants, generally outperform filters on the tested data sets and that backward search does not significantly outperform forward search as previously claimed by Doak (1992). Another exploration method, the stepwise search, combines forward and backward sequential search so that at each step in the algorithm a feature can either be added or removed. When compared to unidirectional methods and filters, stepwise search algorithms outperform unidirectional search and some filtering techniques (Caruana and Freitag 1994).

Kohavi and John (1997) propose a best-first-search feature-exploration technique with compound operators. Compound operators add or remove groups of features, as opposed to adding or removing a single feature in sequential forward search and sequential backward search. The branch-and-bound algorithm (Narendra and Fukunaga 1977; Somol et al. 2004) can yield an optimal feature subset, but requires a monotonic evaluation function that is generally not practical. Floating-search methods (Pudil et al. 1994a, b) add and remove different numbers of features to avoid the nested-feature problem. Ng (1998) presents theoretical bounds on the performance of wrappers and a new search procedure called ORDERED-FS that only searches over all subsets of the same size to reduce computational cost.

When using wrappers, the choice of an evaluation function greatly affects the outcome of the method. Dash et al. (2000) compare a consistency measure to distance measures, information measures, and dependence measures. The authors argue for a consistency-measure evaluation function, because it is monotonic and lacks search bias. Obviously, the type of evaluation function is based on the available data. Functions that require the class label, such as consistency or accuracy, cannot be implemented in unsupervised learning.

While wrappers can be used to select features for both GMMs and HMMs, there are two primary drawbacks to these methods both having to do with the unsupervised data. First, the wrapper will need to use an unsupervised evaluation metric. In unsupervised feature selection, these metrics often trade off the likelihood the model fits the data with the number of parameters or features included in the model. These metrics can be inaccurate and uninformative for feature selection. Second, the computation required for the search procedure is

amplified due to the fact that unsupervised learning techniques for GMMs and HMMs are often iterative and computationally expensive.

3.3 General embedded techniques

Embedded techniques simultaneously select features and construct models. Therefore, these techniques have the wrapper's advantage of selecting feature subsets with respect to a specific learning algorithm, and the filter's advantage of being more computationally efficient than wrappers. Classification and regression trees (CART) (Murphy 2012) are one example of an embedded feature selection method. CART recursively divides the feature space to create a classification model. Irrelevant features will not be selected for inclusion in the tree. Daelemans et al. (2003) show that jointly selecting features and optimizing model parameters in a natural-language-processing context outperforms optimizing the two processes separately.

While the number of general embedded feature selection techniques discussed in this section is limited, there are numerous embedded techniques for GMMs and HMMs that will be discussed in the following sections.

4 Feature selection for GMM literature review

Feature selection methods specific to GMMs are discussed in this section. It begins with an overview of the research that uses some form of feature selection with GMMs in an applied setting. Later in the section, filters, wrappers, and embedded techniques specifically designed for GMMs are outlined. Table 1 lists the feature selection methods for GMMs discussed in this section.

Ribeiro and Santos-Victor (2005) combine several types of feature selection methods in an application of feature selection to human activity recognition from video. The authors do not propose a new feature selection technique but rather test four different methods with the ultimate goal of finding the best subset for their application. They limit their search by using methods with a predefined number of features. The four methods are a brute search (testing all possible combinations of subsets contain the preselected number of features), a lite search (grouping the individual features with the best predictive ability into a single subset), a lite-lite search (similar to the lite search but ranking features), and the Relief algorithm. In their testing, they look for feature subsets of only 1, 2, or 3 features, thus limiting the computation. They find that the brute search method with 3 features yields the highest recognition accuracy on the tested data.

In another applied setting, Godino-Llorente et al. (2006) use two feature selection methods to select features derived from mel frequency cepstral coefficients in a voice impairment assessment tool. The two methods, F-ratio and Fisher's discriminant ratio, both compare metrics derived from class assignments. Therefore, observations must be classified using the learned model or supervised data must be provided to utilize the techniques proposed in this study. The work of Godino-Llorente, Gomez-Vilda, and Blanco-Velasco is included in this review to present another application of feature selection when GMMs are used to model data.

Kerroum et al. (2010) propose a feature selection method for classification that uses GMMs but is not specifically designed for selecting features for a GMM. This method is a wrapper that employs mutual information as the evaluation function. GMMs are used to model the data during feature selection but, in the end, the selected features are fed to a standard

Table 1 Feature selection methods for Gaussian mixture models

Filters	Wrappers	Embedded
Krishnan et al. (1996)	Dy and Brodley (2004)	Pan and Shen (2007)
	Raftery and Dean (2006)	Pan et al. (2006)*
	Maugis et al. (2009a)	Wang and Zhu (2008)
	Maugis et al. (2009b)	Xie et al. (2008b)
	Galimberti et al. (2017)	Bhattacharya and McNicholas (2014)
	Scrucca (2016)	Guo et al. (2010)
	Marbac and Sedki (2017)	Xie et al. (2008a)
	Zeng and Cheung (2009)	Zhou et al. (2009)
	Liu et al. (2002)	Xie et al. (2010)
		Galimberti et al. (2009)
		Maugis and Michel (2011)
		Bouveyron and Brunet-Saumard (2014)
		Carbonetto et al. (2003)
		Pudil et al. (1995)*
		Novovicová et al. (1996)*
		Figueiredo et al. (2003)
		Law et al. (2002)
		Law et al. (2004)
		Valente and Wellekens (2004)
		Constantinopoulos et al. (2006)
		Li et al. (2009)
		Chang et al. (2005)
		Graham and Miller (2006)
		Allili et al. (2010)
		Tadesse et al. (2005)
		Kim et al. (2006)
		Swartz et al. (2008)
		Vannucci and Stingo (2010)
		Vellido (2006)
		Vellido et al. (2006)
		Vellido and Velazco (2008)

Starred methods require some form of supervised data

classifier such as a support vector machine. More specifically, the mutual information of two continuous random variables is

$$I(X, Y) = \int p(x, y) \log \frac{p(x, y)}{p(x)p(y)} dxdy. \tag{17}$$

Let X represent the data, and C represent the class. The mutual information between the input X and the output C is

$$I(X; C) = H(C) - H(C|X), \tag{18}$$

where $H(C)$ and $H(C|X)$ are entropy and conditional entropy. If the number of classes is known, the entropy $H(C)$ is fixed and the information is dependent upon the conditional

differential entropy. A mixture model is used to model the data in the conditional probability $p(c|x)$. Let g_c be the number of Gaussian mixtures in class c and N_c be the number of classes. Then

$$p(c|x) = \frac{\sum_{r=1}^{g_c} \pi_{cr} f(x|\theta_{cr})}{\sum_{k=1}^{N_c} \sum_{r=1}^{g_k} \pi_{kr} f(x|\theta_{kr})}, \tag{19}$$

where $f(x|\theta)$ is a Gaussian distribution, π is the mixture prior probability, and θ is the set of parameters for the Gaussian distribution. The GMM is used to get a better fit to the data, but this proposed feature selection method can only be used if the class labels are provided.

Steinley and Brusco (2008) conducted an empirical study of eight feature selection techniques for clustering including two techniques specific to GMMs (Law et al. 2004; Raftery and Dean 2006). Numerical experiments were only performed on simulated data. The authors conclude that the feature selection techniques using GMMs did not perform as well as the other methods on the synthetic data. They also conclude that a major limitation of these methods is that the number of clusters must be known *a priori*. We feel that an updated empirical study is needed which includes a wider variety of techniques and numerical experiments performed on real data sets.

4.1 Filters for GMMs

Krishnan et al. (1996) propose a pruning feature selection technique specific to GMMs based on the Fisher ratio. We refer to this method as a pruning technique rather than a filtering technique because it requires the model parameters, where filters are independent of the model. However, this method does not iterate through the feature space nor simultaneously estimate model parameters and feature subsets so we consider it closer to a filter than either a wrapper or an embedded method. This method assumes that each class has a multi-modal distribution modeled as a GMM. The Fisher ratio between two classes is

$$F_{ijk}^{u} = \frac{(\mu_{ik} - \mu_{jk})^2}{\sigma_{ik}^2 + \sigma_{jk}^2)}, \tag{20}$$

where μ_{ik} and μ_{jk} are the cluster means for the kth feature of classes i and j, and σ_{ik} and σ_{jk} are the corresponding standard deviations of the kth feature. This ratio can be extended to multiple classes and averaged

$$F_{k}^{u} = \frac{1}{J(J-1)} \frac{\sum_{i=1}^{J} \sum_{j=1}^{J} P_i P_j F_{ijk}^{u}}{\sum_{i=1}^{J} \sum_{j=1}^{J} P_i P_j}, \tag{21}$$

where J is the number of classes, and P_i and P_j represent the prior probability of belonging to each class. Discriminating features have a higher Fisher ratio, and all features in the data set can be ranked based on this metric. Krishnan et al. generalize this metric to GMMs and use it for feature selection.

4.2 Wrappers for GMMs

Dy and Brodley (2004) evaluate a general feature selection criterion for clustering on GMMs. This method, called feature subset selection using expectation-maximization (FSSEM), was previously presented in Dy (2000) but not specified to GMMs. FSSEM uses one of two methods for evaluating features: scatter separability and maximum likelihood. Both of these

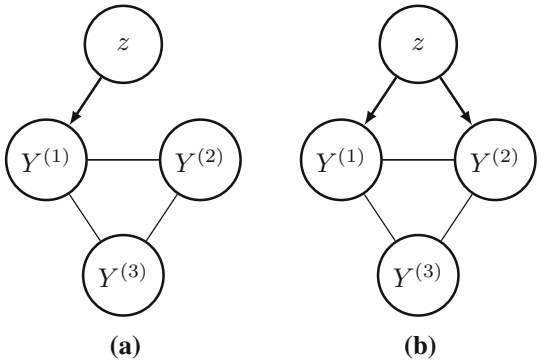

Fig. 9 Graphical representation of $M1$ (left) and $M2$ (right) from Raftery and Dean (2006)

criteria contain dimension-based bias. Therefore, the authors propose a normalization to reduce bias. Dy and Brodley (2004) cite the lack of research on unsupervised feature selection and attempt to explore wrappers as a method for feature selection for unsupervised learning, identify issues in the area of unsupervised feature selection, propose ways to address these issues, point out lessons learned, and give direction to future research. The numerical experiments on both synthetic and real data using GMMs as the clustering algorithm demonstrate that feature selection outperforms using the entire feature set, one evaluation criteria is not better than the other, and both criteria still contain bias. The maximum likelihood approach selects features that yield a high-density clustering while the scatter separability approach selects features that maximize separability.

Raftery and Dean (2006) formulate the feature selection process for clustering as a comparison between competing models. In this method, variables are divided into three groups: those relevant to clustering, those irrelevant to clustering, and those proposed for inclusion or exclusion from the set of variables relevant to clustering. At each step, models are compared using the Bayesian information criterion (BIC). Raferty and Dean define BIC as

$$BIC = 2 \times \log(\text{maximized likelihood}) - (\#\text{ of parameters}) \times \log(n), \tag{22}$$

where n is the number of observations. Let $Y^{(1)}$ represent the set of variables already selected for clustering, $Y^{(2)}$ be the set of variables being considered for inclusion in the model, and $Y^{(3)}$ be the set of remaining variables. The proposed feature selection method compares model $M1$

$$P\left(Y^{(1)}, Y^{(2)}, Y^{(3)}|z\right) = P\left(Y^{(3)}|Y^{(1)}, Y^{(2)}\right) P\left(Y^{(2)}|Y^{(1)}\right) P\left(Y^{(1)}|z\right), \tag{23}$$

to model $M2$

$$P\left(Y^{(1)}, Y^{(2)}, Y^{(3)}|z\right) = P\left(Y^{(3)}|Y^{(1)}, Y^{(2)}\right) P\left(Y^{(1)}, Y^{(2)}|z\right), \tag{24}$$

where z represents the class memberships. Figure 9 displays a graphical representation of $M1$ and $M2$. A greedy stepwise search through the feature space is performed until convergence. BIC is an unsupervised evaluation metric which assesses both features in the model and the number of clusters. Therefore, this technique can both select features and the number of clusters.

Maugis et al. (2009a) propose a feature selection method for GMMs based on backward stepwise selection. This method is an extension of the work previously conducted by Raftery and Dean (2006) as the authors use a similar method for dividing variables into sets of relevant, irrelevant, and proposed for inclusion or exclusion. However, the method in Maugis

et al. (2009a) allows for the possibility that irrelevant variables can be independent of the relevant variables. Further, the authors investigate the more general case where blocks of variables cannot be split. Like the method in Raftery and Dean (2006), the method proposed by Maugis et al. can be used to select features and the number of clusters. Maugis, Celeux, and Martin-Magniette later generalize this technique by assuming that some of the irrelevant features can be linked to some of the relevant features while other irrelevant features remain independent of the relevant features (Maugis et al. 2009b). Celeux et al. (2014) present a study comparing the method proposed in Maugis et al. (2009b) to a regularization method specific to the K-means algorithm (Witten and Tibshirani 2010). This study concluded that feature selection improves clustering accuracy and that the model selection method outperformed the regularization method on simulated data. Galimberti et al. (2017) further generalize this concept by allowing features to provide information about multiple clustering structures. This is accomplished by modeling the dependence between subgroups of features as a multiple regression linear model.

Scrucca (2016) suggests using the evaluation criteria proposed by Raftery and Dean (2006) but a genetic algorithm to search the feature space. A genetic algorithm is composed of several parts. The genetic coding scheme represents the feature subsets, where inclusion in the subset is indicated by a binary variable. The population size is an input parameter into the genetic algorithm and represents the number of models to generate. The fitness function evaluates the feature subset. In the proposed method, the BIC criteria from Raftery and Dean (2006) is used as the fitness function. The genetic operators mutate the genetic code with the intention of improving the fitness function. The genetic algorithm tends to select smaller feature sets than the stepwise search while maintaining accuracy.

Marbac and Sedki (2017) cast the feature selection problem as a model selection problem and propose a new information criterion for model evaluation, the maximum integrated complete likelihood (MICL). This method assumes that an irrelevant feature will have the same estimates for μ and σ across all clusters. A binary variable is introduce that indicates the relevance of each feature. Specifically, if $\omega_l = 1$ the lth feature is relevant, and if $\omega_l = 0$ the lth feature is irrelevant. A model $\mathbf{m}$ is defined by ω and the number of clusters I. Feature selection is performed by finding the model that maximizes the posterior distribution. If a uniform prior is assumed over the models, then

$$\mathbf{m}^* = \text{argmax}_{\mathbf{m}\in\mathcal{M}} P(Y|\mathbf{m}), \tag{25}$$

where $\mathcal{M}$ is the set of possible models, and

$$P(Y|\mathbf{m}) = \int_{\Lambda} P(Y|\mathbf{m}, \Lambda) P(\Lambda|\mathbf{m}) d\Lambda. \tag{26}$$

The integrated complete-data likelihood is

$$P(Y, X|\mathbf{m}) = \int_{\Lambda} P(Y, X|\mathbf{m}, \Lambda) P(\Lambda|\mathbf{m}) d\Lambda. \tag{27}$$

The binary relevance variable ω is incorporated into the modeling process through the prior distribution on the model parameters $P(\Lambda|\mathbf{m})$. The MICL evaluation criterion is

$$MICL(\mathbf{m}) = \log P(Y, X^*_{\Lambda}|\mathbf{m}), \tag{28}$$

where

$$X^*_{\mathbf{m}} = \text{argmax}_X \log P(Y, X|\mathbf{m}). \tag{29}$$

An iterative optimization algorithm alternates between estimating X given Y and $\mathbf{m}$ and maximizing ω given Y and X. We classify this method as a wrapper because of this iterative process of selecting the feature set and then estimating the cluster assignment. However, it is generally more computationally efficient than other wrappers because estimating the model parameters has been integrated out.

Zeng and Cheung (2009) construct two new metrics for evaluating feature sets. The objective of the first is to select features that are relevant to clustering. The objective of the second is to remove redundant features. An iterative process of constructing a clustering model and performing feature selection is proposed with the objective of determine both the optimal number of clusters and the relevant non-redundant feature set. The authors build on their previous work that proposes using the rival penalized EM algorithm to determine the optimal number of clusters for a GMM (Cheung 2004, 2005). The relevant feature metric utilizes the ratio of cluster specific variance to total variance

$$SCORE_l = \frac{1}{I}\sum_{i=1}^{I}\left(1 - \frac{\sigma_{il}^2}{\sigma_l^2}\right), \tag{30}$$

where σ_l is the total variance of the lth feature. This scoring metric is used to rank features, and features with a score below a given threshold can be removed. Redundant features are found using a Markov blanket. Let M_l represent the Markov blanket for the lth feature, F represent the entire feature set, and F_l represent the lth feature. If the Markov blanket for F_l can be found in F, the information in F_l is redundant and can be eliminated without affecting the ability of the model to predict the cluster. We consider this method a wrapper because of its iterative nature, and its reliance on constructing a model.

Some feature selection methods for clustering find cluster-specific feature sets. To illustrate this point, we present the example from Li et al. (2008), a study on localized feature selection for general clustering. Consider a four cluster problem with four features. Clusters 1 and 2 have two relevant features, designated $X1$ and $X2$, and two irrelevant features, designated $X3$ and $X4$. Clusters 3 and 4 have $X2$ and $X3$ as relevant features and $X1$ and $X4$ as irrelevant features. The model parameters are $\mu_{C1} = [0.5, -0.5, 0, 0]$, $\mu_{C2} = [-0.5, -0.5, 0, 0]$, $\mu_{C3} = [0, 0.5, 0.5, 0]$, $\mu_{C4} = [0, 0.5, -0.5, 0]$, $\sigma_{C1} = \sigma_{C2} = [0.2, 0.2, 0.6, 0.6]$, and $\sigma_{C3} = \sigma_{C4} = [0.6, 0.2, 0.2, 0.6]$. A hundred observations are generated from each cluster and the simulated results are presented in Fig. 10. Li, Dong, and Hua concluded that a general clustering algorithm may not be able to adequately cluster this data due to each cluster having an irrelevant feature. Further, a global feature selection method could find it difficult to determine a relevant feature subset for all 4 clusters. A localized feature selection method, which finds a relevant feature subset for each cluster, is needed for this type of problem.

Liu et al. (2002) develop a cluster-specific feature selection method for document clustering where the features are word occurrences. In this method, a GMM is trained on all the features using the EM algorithm, and then each document is assigned to a cluster. Discriminative features are determined using a discriminative feature metric. A word or feature is considered discriminative if that word has the highest number of occurrences inside the assigned cluster and a low number of occurrences outside the assigned cluster. This feature selection algorithms iterates between finding discriminative features and assigning documents to clusters based on those features, therefore we classify the algorithm as a wrapper.

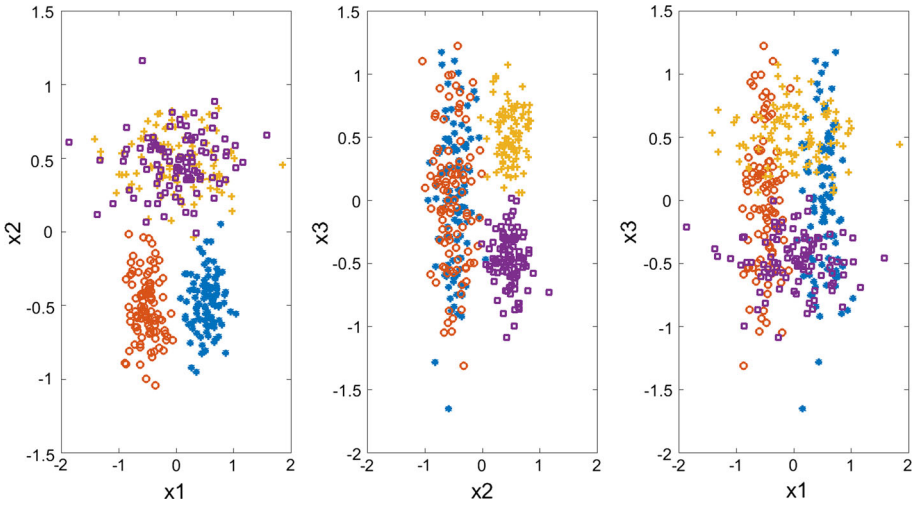

Fig. 10 Data generated by model presented in Li et al. (2008)

4.3 Embedded feature selection techniques

In this section, embedded feature selection techniques for GMMs are reviewed. There are numerous types of embedded methods for GMMs so this section is broken down into several subsections, each focused on a different type of embedded technique.

4.3.1 Penalized model-based clustering

Pan and Shen (2007) propose a penalized model-based clustering algorithm for feature selection. This method is specifically designed to address data sets that are "high dimension, low sample size". The primary idea behind this feature selection technique is that if the cluster means for the lth feature is close to 0, then the lth feature is irrelevant. The cluster means are driven toward 0 by penalizing the cluster means during estimation. Similar methods are used in regression to penalize coefficients.

The log-likelihood can be found by taking the logarithm of Eq. (1)

$$\log P(Y|\Lambda) = \sum_{n=1}^{N} \log \left(\sum_{i=1}^{I} \pi_i \mathcal{N}(y_n|\boldsymbol{\mu}_i, \Sigma_i) \right). \tag{31}$$

Similarly, when assuming that each feature is independent, the log-likelihood of the joint probability of X and Y is

$$\log P(X, Y|\Lambda) = \sum_{n=1}^{N} \sum_{i=1}^{I} \mathfrak{r}_{in} \left[\log \pi_i + \sum_{l=1}^{L} \log \mathcal{N}(y_{ln}|\mu_{il}, \sigma_{il}^2) \right], \tag{32}$$

where $\mathfrak{r}_{in} = 1$ if the nth observation belongs to the ith mixture and 0 otherwise.

Pan and Shen regularize the complete log-likelihood by adding a penalty term

$$\log P(X, Y|\Lambda) = \sum_{n=1}^{N} \sum_{i=1}^{I} \mathfrak{r}_{in} \left[\log \pi_i + \sum_{l=1}^{L} \log \mathcal{N}(y_{ln}|\mu_{il}, \sigma_{il}^2) \right] - \lambda \sum_{i=1}^{I} \sum_{l=1}^{L} |\mu_{il}|. \tag{33}$$

Equation 33 replaces the conventional complete log-likelihood in the EM algorithm, and BIC is used for selecting the number of mixtures. In an earlier paper, Pan et al. (2006) developed a semi-supervised version of this penalized mixture model and applied it to microarray classification.

Wang and Zhu (2008) build on the work of Pan and Shen by proposing two new penalty terms that "group" the parameters of each cluster. Let $\mathcal{L}(\Lambda)$ represent the complete log-likelihood from Eq. (32) and $\mathcal{L}_P(\Lambda)$ the penalized form of the complete log-likelihood. The first novel penalized log-likelihood is called the Adaptive L_∞ Penalized Gaussian Mixture Model (ALP-GMM)

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sum_{l=1}^{L} \max_i(|\mu_{1l}|, \ldots, |\mu_{Il}|). \tag{34}$$

Weights w_i can be added to the penalty to give preference to different variables

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sum_{l=1}^{L} w_i \max_i(|\mu_{1l}|, \ldots, |\mu_{Il}|). \tag{35}$$

The second novel penalty term yields a model named the Adaptive Hierarchically Penalized Gaussian Mixture Model (AHP-GMM). First, the cluster means are reparameterized $\mu_{il} = \gamma_l \theta_{il}$ for $i = 1, \ldots, I$ and $l = 1, \ldots, L$ where $\gamma_l \geq 0$. The hierarchical penalized complete log-likelihood is

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda_\gamma \sum_{l=1}^{L} \gamma_l - \lambda_\theta \sum_{i=1}^{I} \sum_{l=1}^{L} |\theta_{il}|. \tag{36}$$

As with the previous penalty term, weights can be added

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda_\gamma \sum_{l=1}^{L} w_i^\gamma \gamma_l - \lambda_\theta \sum_{i=1}^{I} \sum_{l=1}^{L} w_i^\theta |\theta_{il}|. \tag{37}$$

The EM algorithm is used to estimate model parameters, however ALP-GMM does not have a closed form solution for the cluster mean update. Therefore, a quadratic program is derived. Through tests on simulated data and microarray data, the authors conclude that both ALP-GMM and AHP-GMM outperform the method in Pan and Shen (2007), but ALP-GMM is more computationally expensive due to the quadratic program.

Xie et al. (2008b) develop vertical and horizontal penalties. The vertical penalty term assumes that if a feature is irrelevant, the means of all mixtures will be equal to 0. This results in the following penalized log-likelihood

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sqrt{I} \sum_{l=1}^{L} ||\mu_{\cdot l}||, \tag{38}$$

where $||\mu_{\cdot l}|| = \sqrt{\sum_I \mu_{il}^2}$. The horizontal penalty term assumes that features can be divided into subgroups. Establishing the subgroups often requires some form of prior knowledge. The subgroups are penalized using the following log-likelihood

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sum_{i=1}^{I} \sum_{m=1}^{M} \sqrt{L^m} ||\mu_i^m||, \tag{39}$$

where there are M subgroups of features and L^m represents the number of features in subgroup m. These two penalties can be combined into the following log-likelihood

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sum_{m=1}^{M} \sqrt{IL^m}||\mu^m||. \tag{40}$$

The vertical, horizontal, and combined penalties are shown to outperform the original penalized method in Pan and Shen (2007) on both synthetic and real data.

Several other penalty methods have been suggested in the literature. Bhattacharya and McNicholas (2014) propose a LASSO-like penalty term. The log likelihood function which is maximized during parameter estimation is

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - n\lambda_n \sum_{i=1}^{I} \pi_i \sum_{l=1}^{L} |\mu_{il}|, \tag{41}$$

where λ_n is dependent upon the observation. The authors use a modified form of the EM algorithm. Guo et al. (2010) propose a pairwise penalty term with equal covariance across the clusters

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda \sum_{1 \le i < i' \le I} \sum_{l=1}^{L} |\mu_{il} - \mu_{i'l}|. \tag{42}$$

The previous penalty methods discussed shrink the cluster means of non-informative features to 0, while the pairwise penalty shrinks non-informative cluster means towards each other.

In addition to the L_1 regularization proposed by Pan and Shen (2007), Xie et al. (2008a) penalize the variance. They propose two schemes and both penalize the mean and variance of each mixture. The first penalized log-likelihood is

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda_1 \sum_{i=1}^{I} \sum_{l=1}^{L} |\mu_{il}| - \lambda_2 \sum_{i=1}^{I} \sum_{l=1}^{L} |\sigma_{il}^2 - 1|, \tag{43}$$

and the second is

$$\mathcal{L}_P(\Lambda) = \mathcal{L}(\Lambda) - \lambda_1 \sum_{i=1}^{I} \sum_{l=1}^{L} |\mu_{il}| - \lambda_2 \sum_{i=1}^{I} \sum_{l=1}^{L} |\log \sigma_{il}^2|. \tag{44}$$

The previously discussed penalized methods all assume a diagonal covariance matrix. In practice, it is unlikely for this assumption to hold. Zhou et al. (2009) relax the assumption of a diagonal covariance matrix by penalizing the terms of the precision matrix W. More specifically, $\lambda_2 \sum_{l=1}^{L} \sum_{l'=1}^{L} |W_{ll'}|$ is added to Eq. (33). Xie et al. (2010) build on their prior work by generalizing to non-diagonal covariance matrices. This generalization leads to a significant increase in the number of parameters that need to be estimated. Further, there are computational issues stemming from the requirement that the estimated covariance matrix be positive definite. To overcome these obstacles, Xie, Pan, and Shen utilize a mixture of factor analyzers (McLachlan and Peel 2000) and model the covariance matrix using latent variables. As with the previous penalized methods, the EM algorithm is used for estimating parameters of the penalized GMM.

The concept of a penalized likelihood can be extended to a mixture of factors (Galimberti et al. 2009). In this method, it is assumed that the observation vector y is generated by a linear factor model

$$y = \tilde{\Lambda}\tilde{z} + \tilde{\mu}, \tag{45}$$

where $\tilde{\Lambda}$ is a factor loading matrix, $\tilde{z}$ is an R-dimensional latent variable, and $\tilde{\mu}$ is a L-dimensional Gaussian term with a mean of 0 and covariance Ψ. It is also assumed that the latent variable $\tilde{z}$ is a mixture of Gaussian distributions $\tilde{z} \sim \sum_{i=1}^{I} \pi_i \mathcal{N}(\tilde{z}|\tilde{\mu}_i, \Sigma_i)$. The log-likelihood for y can now be written as

$$l(y) = \sum_{n=1}^{N} \log \sum_{i=1}^{I} \pi_i \mathcal{N}(y|\tilde{\Lambda}\tilde{\mu}_i, \tilde{\Lambda}\Sigma_i\tilde{\Lambda}^T + \Psi). \tag{46}$$

The penalty term for the mixture of factors shrinks the components of the factor loading matrix $\tilde{\Lambda}$ towards zero.

Maugis and Michel (2011) derive a non-asymptotic penalized criterion for model and feature selection. In their approach, the "best" model, in terms of both the number of clusters and the relevant feature subset, minimizes the estimation error between the estimated likelihood given the data and the true but unknown likelihood. The authors consider four collections of Gaussian mixtures, where each collection is defined by the structure of the variance matrix for both the relevant and irrelevant sets of features. One major drawback to this method is that it is purely theoretical. Maugis and Michel acknowledge that these theoretical results are not immediately usable because they rely upon unknown constants and the mixture parameters are unbounded.

Bouveyron and Brunet-Saumard (2014) propose a feature selection method for a discriminative latent mixture (DLM) model using penalization to impose sparsity in the projection matrix. A DLM, as described in Bouveyron and Brunet (2012), models the observations as a linear combination of an unobserved random vector

$$Y = UX + \epsilon, \tag{47}$$

where U is the loading matrix and $\epsilon \sim \mathcal{N}(\epsilon|0, \Psi)$. The unobserved variable is dependent upon the cluster, represented by Z, and follows a Gaussian distribution $P(X|Z = i) \sim \mathcal{N}(X|\boldsymbol{\mu}_i, \Sigma_i)$. The conditional probability of Y given both X and Z also follows a Gaussian distribution $P(Y|X, Z = i) \sim \mathcal{N}(Y|UX, \Psi)$. The marginal distribution of Y is

$$f(y) = \sum_{i=1}^{I} \pi_i \mathcal{N}(y|m_i, S_i), \tag{48}$$

where $m_i = U\mu_i$ and $S_i = U\Sigma_i U^T + \Psi$. A modified version of the EM algorithm, which adds an intermediate F-step to estimate the loading matrix, is derived and used to estimate model parameters. The discriminative variables are linear combinations of the original variables making interpretation difficult. Bouveyron and Brunet-Saumard (2014) propose three methods for penalizing the loading matrix which imposes sparsity and allows for feature selection. The first method adds an L_1 regularization term to the estimation of U in the F-step, the second method implements a penalized regression criterion, and the third method implements penalized singular value decomposition.

4.3.2 Bayesian methods

Carbonetto et al. (2003) propose a Bayesian feature weighting method for unsupervised feature selection for GMMs. In this Bayesian technique, the means and covariance matrices of the Gaussian distributions are treated as random variables and assigned prior distributions. The component or cluster means, designated by μ_c, are assigned a Gaussian prior with the mean designated by μ^* and a diagonal covariance matrix designated by T with elements τ^2.

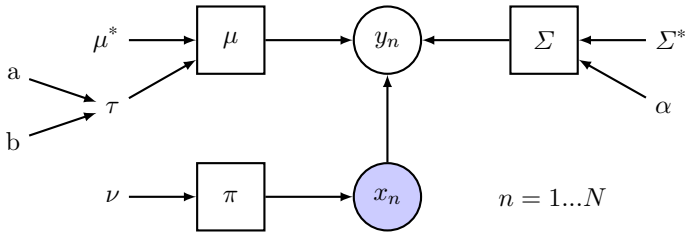

Fig. 11 Graphical model of Bayesian GMM from Carbonetto et al. (2003)

μ^* is the sample mean of the observed data, and the τ's are estimated using the EM algorithm. Estimated τ's near 0 are assumed to be irrelevant. An inverse Gamma distribution is placed on each τ. Figure 11 displays a graphical model of this Bayesian GMM. This method could be viewed as being similar to the penalized likelihood methods of Pan and Shen (2007) and Wang and Zhu (2008), but instead of shrinking the mean of the cluster to 0, Carbonetto et al. shrink variance of the of the prior distribution on the cluster means to determine the relevance of features.

4.3.3 Feature saliency

Feature saliency, which recasts the feature selection problem as a parameter-estimation problem, is an embedded feature selection technique first developed for GMMs. New parameters, called feature saliencies, are added to the conditional distribution of the GMM. The emission probability consists of mixture-dependent and mixture-independent distributions, and feature saliencies represent the probability of belonging to the mixture-dependent distribution. The saliencies can be interpreted as the probability that a feature is relevant. The feature-saliency selection method is an embedded feature selection technique, because it simultaneously constructs a model and performs selection. However, the estimated feature saliency can be used to rank feature relevance, as is the case with many filtering techniques.

A method that does not estimate feature saliencies but uses a similar idea of mixture-dependent and mixture-independent distributions was developed by Pudil et al. (1995). In this model, each observation in Y is considered to belong to one of C classes $\Omega = \omega_1, \omega_2, \ldots, \omega_C$, and the labels for each observation are assumed to be known. The class conditional distribution $P(Y|\omega)$ is assumed to be a mixture of distributions $f(y|b)$, where b represents the distribution's parameters. The probability density function can be written as

$$P(Y|\omega) = \sum_{i=1}^{I_\omega} \pi_i^\omega \prod_{l=1}^{L} [f(y_l|b_{0l})^{1-\phi_l} f(y_l|b_{il}^\omega)_l^\phi], \tag{49}$$

where ϕ_l is binary $\{0, 1\}$, b_{0l} represents the parameters for the background distribution, and b_{il}^ω are the parameters for the mixture and class dependent distribution. Feature selection is performed by finding the set of ϕ's that minimize the Kullback-Liebler divergence between the true and postulated class-conditional probability density function. The EM algorithm is used to estimate model parameters. This method was later refined and specified for a two class problem in Novovicová et al. (1996).

In the work that would eventually lead to feature saliency, Figueiredo et al. (2003) use the idea of splitting the likelihood into relevant and irrelevant distributions. In their formulation, the available features are divided into two subsets $y^{\mathscr{U}}$ and $y^{\mathscr{N}}$, which represent the disjoint "useful" and "non-useful" feature sets. The data likelihood can be written as

$$P(y|\mathscr{U}, \Lambda^{\mathscr{U}}, \Lambda^{\mathscr{N}}) = P(y^{\mathscr{N}}|\Lambda^{\mathscr{N}}) \sum_{i=1}^{I} \pi_i P(y^{\mathscr{U}}|\Lambda_i^{\mathscr{U}}), \tag{50}$$

where $\Lambda^{\mathscr{N}}$ represents the set of model parameters for the "non-useful" feature distribution, $\Lambda_i^{\mathscr{U}}$ represents the set of model parameters for the ith mixture of the "useful" feature distribution, and $\mathscr{U}$ specifies the set of useful features. This feature selection method relies on the assumption that the non-useful features are independent of the mixture. This can be seen by deriving the posterior probability that y belongs to the ith mixture. Let $P(x = i|y, \Lambda) = w_i$, then

$$\begin{aligned} w_i &= \frac{\pi_i P(y^{\mathscr{U}}|\Lambda_i^{\mathscr{U}}) P(y^{\mathscr{N}}|\Lambda^{\mathscr{N}})}{\sum_{i=1}^{I} \pi_i P(y^{\mathscr{U}}|\Lambda_i^{\mathscr{U}}) P(y^{\mathscr{N}}|\Lambda^{\mathscr{N}})} \\ &= \frac{\pi_i P(y^{\mathscr{U}}|\Lambda_i^{\mathscr{U}})}{\sum_{i=1}^{I} \pi_i P(y^{\mathscr{U}}|\Lambda_i^{\mathscr{U}})}, \end{aligned} \tag{51}$$

where w_i can be obtained using the EM algorithm. Let $v_i(\mathscr{N})$ be the expected label using only the set of useful features, and let $\hat{\Lambda}$ be the corresponding model parameters. A measure of non-usefullness can now be defined by

$$\Upsilon(\mathscr{N}) = \frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{I} w_{i,n} \log \frac{w_{i,n}}{v_{i,n}(\mathscr{N})}. \tag{52}$$

This non-usefullness measure is used in a backward sequential search algorithm to rank the features. The cutoff point between the useful and non-useful features is determined using minimum description length

$$\hat{\mathscr{U}} = \text{argmin}_{\mathscr{U}} \left\{ \min_{\Lambda_{\mathscr{U}}, \Lambda_{\mathscr{N}}} \{-\log P(Y|\mathscr{U}, \Lambda_{\mathscr{U}}, \Lambda_{\mathscr{N}})\} + \frac{|\Lambda_{\mathscr{U}}| + |\Lambda_{\mathscr{N}}|}{2} \log(N) \right\}. \tag{53}$$

In their initial studies of feature saliency and GMMs, Law et al. (2002) and Law et al. (2004) used the EM algorithm and the minimum-message length (MML) criterion to estimate model parameters and the number of clusters. The likelihood for the feature saliency GMM is written as

$$P(Y|\Lambda) = \sum_{i=1}^{I} \pi_i \prod_{l=1}^{L} (\rho_l p(y_l|\theta_{il}) + (1 - \rho_l) q(y_l|\lambda_l)), \tag{54}$$

where $p(\cdot|\cdot)$ is a mixture-dependent Gaussian distribution with parameters $\theta_{il} = \{\mu_{il}, \sigma_{il}\}$, $q(\cdot|\cdot)$ is a mixture-independent Gaussian distribution with parameters $\lambda_l = \{\mu_l, \sigma_l\}$, and ρ_l is the feature saliency which represents the probability that a feature is relevant. The MML penalty, which is used to aid in model selection, encourages feature-saliency estimates for irrelevant features to go to zero and helps estimate the number of mixtures by forcing the probability of sparsely occupied components to zero. Law et al. (2004) propose a post-processing step in which feature saliencies are optimized to discriminate between components, after which other parameters of the model are estimated using EM. They also note that a limitation to this method is the assumption that all the features are independent.

There are several variational Bayesian (VB) approaches for estimating feature saliency model parameters for GMMs. VB is an approximate learning method for latent variables models. In a general sense, let Y represent the given data, X represent the hidden variables, and Λ represent the model parameters. The VB approach assumes that the true posterior distributions

for the model parameters $p(\Lambda|Y)$ and the hidden variable $p(X|Y)$ can be approximated by $q(\Lambda|Y)$ and $q(X|Y)$. The objective is to find the approximating distributions that maximize the upper bound of the marginal likelihood

$$F(\Lambda, X) = \int q(\Lambda|Y)q(X|Y) \log \frac{p(X, Y|\Lambda)}{q(X|Y)} d\Lambda dX - D(q(\Lambda|Y)||p(\Lambda)), \tag{55}$$

where $p(\Lambda)$ is the prior distribution of the parameters and $D(\cdot|\cdot)$ is the Kullback-Leibler divergence. The learning procedure is similar to the EM algorithm in that it is based on iteratively updating quantities in two steps

$$q(X) \propto \exp^{\mathbb{E}_\Lambda[\log p(X,Y|\Lambda)]}, \tag{56}$$

and

$$q(\Lambda) \propto \exp^{\mathbb{E}_X[\log p(X,Y|\Lambda)]} p(\Lambda). \tag{57}$$

Valente and Wellekens (2004) and Constantinopoulos et al. (2006) both implement VB on a feature saliency GMM. Valente and Wellekens treat all model parameters as random variables and assign them prior distributions, while Constantinopoulos, Titsias, and Likas only treat some of the model parameters as random variables. Both studies conclude that the VB approach outperforms the MML approach proposed in Law et al. (2002, 2004).

Li et al. (2009) propose a localized feature saliency model which assumes that the relevance of each feature is dependent upon the cluster. A matrix S represents the feature relevance where $s_{il} = 1$ indicates that the lth feature is associated with the ith cluster, and $s_{il} = 0$ otherwise. The feature saliency can now be interpreted as $\rho_{il} = P(s_{il} = 1)$, and the likelihood is rewritten as

$$P(Y|\Lambda) = \sum_{i=1}^{I} \pi_i \prod_{l=1}^{L} (\rho_{il} p(y_l|\theta_{il}) + (1 - \rho_{il}) q(y_l|\lambda_{il})). \tag{58}$$

VB approximation is used to estimate posterior distributions for model parameters. The numerical experiments demonstrate that the localized feature selection method outperforms the global method proposed in Constantinopoulos et al. (2006).

Expectation propagation (EP) is another Bayesian inference technique that Chang et al. 2005 apply to a feature saliency GMM model for feature selection. EP is an extension of assumed density filtering (ADF) (Minka 2001), a sequential method for estimating posterior distributions. In a general ADF formulation, let $Y = \{y_1, y_2, \ldots, y_N\}$ represent the observed data and Λ represent the parameters for the model that generated the data. The joint distribution of Y and Λ is given by $P(Y, \Lambda) = P(\Lambda) \prod_{n=1}^{N} p(y_n|\Lambda)$. ADF assumes that the joint distribution can be factored into simple terms denoted by $t_n(\Lambda)$. The objective of ADF is to estimate the approximating distribution $q(\Lambda|h)$, where h is the set of hyperparameters. EP is an extension of ADF that incorporates iterative refinements of the approximating distribution. More specifically, EP starts by estimating $t_n(\Lambda)$ by $\tilde{t}_n$ and iteratively updating the hyperparameters until convergence. Given $\tilde{t}_n$, the approximate posterior can be written as

$$q^{new}(\Lambda|h^{new}) = \frac{\prod_n \tilde{t}_n(\Lambda)}{\int_{\Lambda'} \prod_n \tilde{t}_n(\Lambda') d\Lambda'}. \tag{59}$$

Let $q^{\setminus i}(\Lambda|h^{\setminus i})$ represent the approximate posterior without the ith observation

$$q^{\setminus i}(\Lambda|h^{\setminus i}) \propto \frac{q^{new}(\Lambda|h^{new})}{\tilde{t}_n(\Lambda)}. \tag{60}$$

The contribution of the ith distribution is incorporated into the posterior distribution through

$$\hat{p}(\Lambda) = \frac{\tilde{t}_n(\Lambda) q^{\setminus i}(\Lambda|h^{\setminus i})}{\int_{\Lambda'} \tilde{t}_n(\Lambda) q^{\setminus i}(\Lambda|h^{\setminus i}) d\Lambda'}. \tag{61}$$

The approximate posterior $q^{\setminus i}(\Lambda|h^{\setminus i})$ is found by minimizing the KL divergence between $\hat{p}(\Lambda)$ and $q(\Lambda)$. The update for $\tilde{t}_n$ is

$$\tilde{t}_n = Z_n \frac{q^{new}(\Lambda|h^{new})}{q^{\setminus i}(\Lambda|h^{\setminus i})}, \tag{62}$$

with $Z_n = \int_{\Lambda'} t_n(\Lambda') q^{\setminus i}(\Lambda|h^{\setminus i}) d\Lambda'$. When EP is applied to the feature saliency GMM, Chang, Dasgupta, and Carin found that it outperformed the maximum likelihood formulation proposed in Law et al. (2004).

Graham and Miller (2006) use the mixture-dependent and mixture-independent model for model and feature selection on general mixture models. They assume that features are mixture dependent and write the likelihood as

$$P(Y|\Theta(I)) = \sum_{i=1}^{I} \pi_i \prod_{l=1}^{L} P(Y_l|\theta_i)^{v_{il}} P(Y_l|\theta_s)^{(1-v_{il})}, \tag{63}$$

where $\Theta(I)$ is the set of model parameters for I mixtures, $P(Y_l|\theta_i)$ and $P(Y_l|\theta_s)$ are mixture-dependent and shared distributions of any form with parameter sets θ_i and θ_s (the s indicates the shared or mixture-independent distribution), and v_{il} is a binary model parameter indicating if the feature belongs to the mixture-dependent or shared distribution. The major differences between this formulation and the feature saliency formulation proposed by Law et al. (2004) is the lack of a feature saliency parameter and treating v_{il} as a model parameter as opposed to a random variable. The learning procedure proposed by Graham and Miller (2006) is composed of two steps. In the first step, BIC is used to estimate the number of mixtures. In the second step, a modified EM algorithm is performed to estimate model parameters and select features. The E-step is modified to estimate the expected complete penalized log-likelihood, where the penalty comes from replacing the complete log-likelihood with BIC in the expectation. The M-step is modified by splitting it into two sub-steps: (1) estimates a subset of the model parameters given a fixed v_{il} and (2) finds the optimal configuration of v_{il}. Graham and Miller derive learning procedures for GMMs and multinomial models.

Allili et al. (2010) adapt the feature saliency model to mixtures of generalized Gaussian distributions and apply it to image and video segmentation. Their primary contribution is the assumption that the mixture-independent distribution can be composed of K mixtures. The updated likelihood is now

$$P(y_n|\Lambda) = \sum_{i=1}^{I} \pi_i \prod_{l=1}^{L} \left(\rho_l p(y_{ln}|\theta_{il}) + (1-\rho_l) \sum_{k=1}^{K} \alpha_{kl} p(y_{ln}|\varphi_{kl}) \right), \tag{64}$$

where $p(\cdot|\cdot)$ is a generalized Gaussian distribution (Allili et al. 2008). As in Law et al. (2004), the minimum message length criteria and the EM algorithm are used to estimate model parameters. Similarly, Boutemedjet et al. (2007) adapt feature saliency to mixtures of general Dirichlet distributions and solve for model parameters using the MML criteria and the EM algorithm.

Tadesse et al. (2005) propose a Bayesian feature selection method based on the idea of mixture-dependent and mixture-independent distributions. In this formulation, the features are not assumed to be independent but come from multivariate Gaussian distributions.

Estimation of the model parameters, including the assignment of features to the mixture-dependent and mixture-independent distributions, is carried out using a Markov chain Monte Carlo (MCMC) algorithm. The MCMC algorithm can split and merge mixtures as well as change the assignment of each feature to the preferred distribution. This method is not tested against any of the other feature saliency methods outlined in this section, but we speculate that the computational cost of this MCMC algorithm would not scale well with the number of features as the possible combinations of feature assignments grows exponentially. This model is expanded to infinite mixtures of distributions using a Dirichlet process (Kim et al. 2006). Swartz et al. (2008) add a known structure to the Tadesse model by assigning groups of observations to specific clsuters. This method is particularly useful for modeling microarray data. Later, Vannucci and Stingo (2010) outline a Bayesian approach to variable selection that is similar to feature saliency: A binary random variable is used to represent belonging to either a component-dependent and component-independent distribution, and priors can be used to convey biological information to the system. The authors outline both supervised and unsupervised models, and suggest that Gibbs sampling be used to estimate model parameters. However, only the supervised model is tested and evaluated on data.

Feature saliency has also been expanded to generative topographical mapping (GTM) (Bishop et al. 1998), a constrained form of the GMM. A GTM is essentially a GMM with equal mixture weights ($\pi_i = I^{-1}\ \forall i$), a shared covariance matrix across the mixtures, and Gaussian means that do not move independently. Vellido (2006) uses feature saliency to select features for a GTM. Vellido et al. (2006) use feature saliency on a GTM that assumes a Student's t-distribution instead of the standard Gaussian. The t-distribution allows the model to be more robust to outliers. This is demonstrated on MRS data. However, the MRS results suggest that the performance of the feature saliency GTM may be affected by low sample sizes and noise in the collected data. Vellido and Velazco (2008) investigate these limitations using synthetic data. They conclude that these limitations are a characteristic of the model and suggest that future research should test if regularization could be added to improve model performance in high noise environments.

5 Feature selection for HMM literature review

While numerous studies have investigated feature selection in general and GMMs specifically, research on feature selection methods specific for HMMs is lacking. This section begins by outlining studies addressing feature extraction and dimensionality reduction for HMMs as opposed to feature selection. These methods include principal component analysis, independent component analysis, and feature pruning. Then, we outline studies which compare different types of general feature selection methods and use HMMs as the predictive model. Finally, there are subsections for filters, wrappers, and embedded feature selection techniques with specific application to HMMs. Table 2 lists the feature selection methods for HMMs discussed in this section.

In most applications of HMMs, features are pre-selected based on domain knowledge and the feature selection procedure is completely omitted. Xie et al. (2002) use HMMs for analyzing soccer videos. The objective is to classify if the video is showing a soccer game that is in progress or a soccer game that is on a break. Features are extracted from the video that give information on the ratio of grass pixels to non-grass pixels and the motion intensity which estimates the gross motion in the frame. These features are selected purely on the basis of domain knowledge and there is no analysis of their impact on the classification ability of the

Table 2 Feature selection methods for hidden Markov models

Filters	Wrappers	Embedded
Zhu et al. (2008)*	Charlet and Jouvet (1997)	Städler and Mukherjee (2013)
Wissel et al. (2013)*	Günter and Bunke (2003)*	Zhu et al. (2012)
	Lv and Nevatia (2006)*	Adams et al. (2016)
		Olier and Vellido (2008)

Stars indicate methods which require some form of supervised data

HMM. Montero and Sucar (2004) use HMMs for visual human gesture recognition. They use domain knowledge to select different trajectory-based features, and then they experimentally evaluate different feature subsets. The final feature subset is chosen based on the experimental evaluation but they do present a methodology for feature selection. This could be considered a brute-force search method over subsets of features where the subsets are grouped based on objective similarity, i.e. all feature generated by a certain extraction function are in the same feature subset.

A few attempts have been made to reduce the likelihood calculation of the HMM, but these methods are not truly feature selection techniques, as all data streams must be collected. Bocchieri (1993) proposes to use vector quantization to efficiently compute continuous Gaussian likelihoods. In this method, the likelihoods of observations that are not classified as outliers will be calculated using a Gaussian distribution, while the likelihoods of outliers will only be approximated. Vector quantization is used to determine the the boundary between an outlier and a non-outlier. Li and Bilmes (2003) and Li and Bilmes (2005) propose a similar idea where the likelihood for a subset of features is calculated exactly, and the likelihood for the remaining features is approximated. The authors refer to this method as feature pruning, and the amount of pruning is determined by the algorithm. Another attempt to reduce the computation of HMMs was proposed by Gales et al. (1999). In this method, the number of Gaussian components or number of mixtures is trimmed.

Principal component analysis (PCA) is often used to reduce the size of the feature space and extract features for HMMs. Bashir et al. (2007) use PCA to extract features from object trajectories for activity classification using HMMs. Similarly, Liu and Chen (2003) use PCA to extract features from video for face recognition. Independent component analysis (ICA), which is similar to PCA, has also been used with HMMs and attempts to identify independent noise components of a signal. Windridge and Bowden (2005) use ICA to extract features and then perform feature selection on the extracted features to eliminate the noise components. However, this method is dependent on the ICA transformation and the feature selection technique cannot be successfully utilized without ICA. Therefore, we consider the method proposed by Windridge and Bowden to be a dimensionality reduction technique because ICA does not eliminate data streams. Zhou and Zhang (2008) use ICA to extract features for video content analysis using HMMs which they call the *ICA Mixture Hidden Markov Model*.

Other methods for transforming the original feature set for use with HMMs have also been studied. Yin et al. (2004) present asymmetrically boosted HMMs that use a form of AdaBoost (Meyer 2002; Schwenk 1999) to construct a new feature set. Yin et al. (2008) use segmental boosting, which creates an ensemble of weak learners or HMMs to create a new feature space. This method is presented as feature selection, however we classify it as feature extraction because it does not select a subset of relevant features but instead constructs a new relevant feature set.

Gales (1999) introduces the idea of semi-tied covariance matrices for HMMs with Gaussian conditional likelihoods. This method does not select features but reduces the number of parameters in the covariance matrix.

5.1 Feature selection comparison for HMMs

Nouza (1996) compares three feature selection techniques when using HMMs: sequential forward search (SFS), discriminate feature analysis (DFA), and PCA. The HMM used in this study assumes that each feature in the conditional likelihood follows a Gaussian mixture distribution. The PCA feature extraction technique used in this comparison calculates principal components using eignenvalues and eigenvectors.

DFA evaluates the contribution of each feature to correctly classifying an observation. Consider the case where the covariance matrix is diagonal and there is only one mixture in the conditional likelihood. The log-likelihood given model Ψ can be written as

$$\log P(Y|\Psi) = K - \sum_{l=1}^{L} D_l, \tag{65}$$

where K represents the part of the likelihood that is not dependent on the features, and D_l is

$$D_l = \sum_{t=1}^{T} \frac{(y_{lt} - \mu_{lx_t})^2}{2\sigma_{lx_t}^2}. \tag{66}$$

The state at each time period x_t given the observation sequence is estimated using the Viterbi algorithm. Now, let $C_l = \mathbb{E}[D_l]$ when Ψ is the correct model, and $W_l = \mathbb{E}[D_l]$ when Ψ is the incorrect model. The feature significance factor R_l, which is used for evaluating each feature in DFA, is defined as

$$R_l = \frac{C_l}{W_l}. \tag{67}$$

Features with a higher R_l contribute more to classification and should be considered more relevant than features with lower values of R_l.

The SFS method implemented in this comparison uses recognition rate as the evaluation function. The PCA method can be considered an unsupervised feature extraction technique akin to a filter because it does not require class labels and is performed before training the HMM. The DFA method is a supervised filtering feature selection technique. The SFS is a supervised wrapper. This study demonstrates that SFS and DFA both outperform PCA, but require more computation and supervised data.

Paliwal (1992) compares four feature reduction techniques for use on an HMM-based speech recognizer. The first method is a filtering feature selection technique that uses the F-ratio – the between class variance over the pooled within class variance. The second method constructs an HMM for each feature and then tests the recognition rate of each HMM. The top d features are selected based on the top d recognition rates of the corresponding single feature HMMs. The third method extracts features using linear discriminant analysis, and the fourth method extracts features using PCA. The study concludes that the method that evaluates features using recognition rate and single feature HMMs outperforms the other methods. However, a set of features selected based on domain knowledge was shown to outperform all of the feature reduction techniques in this specific application.

5.2 Filters for HMMs

Zhu et al. (2008) present a discriminant feature selection method specific to HMMs and apply the method to estimating the condition of a micro-milling tool. The authors compare their proposed method to PCA and automatic relevance determination (ARD). ARD is a Bayesian single layer neural network. First, we describe ARD in a general application. Let $D = \{x_n, y_n\}$ for $n = 1, \ldots, N$ be a data set drawn from a random distribution where y is the output, and x is the input to the neural network. The distribution of y can be written as

$$P(y|x, D) = \int P(y|x, w)P(w|D)dw, \tag{68}$$

where $P(w|D)$ is the posterior distribution for the weights in the neural network. ARD assumes that the weights have a Gaussian prior so training simplifies to finding the weights that minimize the cost function

$$E(w) = \frac{\beta}{2}\sum_{n=1}^{N}(y_n - f(x_n; w))^2 + \frac{\alpha}{2}\sum_{k=1}^{W} w_k^2, \tag{69}$$

where α and β are hyperparameters that control the cost function. These hyperparameters are reestimated using the evidence framework (MacKay 1992), and the optimal set of hyperparameters is linked to the relevance of each feature, i.e. a small value for the hyperparameter corresponds to a large weight and relevance.

For the proposed feature selection method, Zhu, Hong, and Wong adapt Fisher's linear discriminant analysis (Duda et al. 2001) to the feature selection problem. Given K classes and a data set x with N observations and L features, the sample mean and covariance for each class are represented by μ_k and Σ_k. The within-class covariance Σ_w can be calculated by

$$\Sigma_w = \sum_{k=1}^{K}\sum_{x\in D_k}(x - \mu_k)(x - \mu_k)^T, \tag{70}$$

where D_k is the subset of observations for class k. The between-class covariance Σ_b can be calculated by

$$\Sigma_b = \sum_{k=1}^{K} n_k(\mu_k - \mu)(\mu_k - \mu)^T, \tag{71}$$

where n_k is the number of observations in class k, and μ is the overall mean of the data. In the two class problem, the absolute values of Σ_w and Σ_b are proportional to $s_1 + s_2$ (the sum of the variances for each class) and $(\mu_1 + \mu_2)^2$, and the Fisher's discriminant ratio (FDR) can be written as

$$FDR = \frac{|\mu_1 + \mu_2|^2}{s_1 + s_2}. \tag{72}$$

The multiclass FDR for the lth feature is

$$FDR(l) = \sum_{i=1}^{K}\sum_{j\neq i}^{K}\frac{|\mu_{il} - \mu_{jl}|^2}{s_{il} + s_{jl}}. \tag{73}$$

The proposed method modifies the multiclass FDR because it emphasizes class separability over feature ranking

$$FDR(l) = \frac{\sum_{i=1}^{K} \sum_{j=1}^{K} |\mu_{il} - \mu_{jl}|^2}{\sum_{i=1}^{K} s_{il}}. \tag{74}$$

For feature selection, the FDR is computed for each feature and then ranked in descending order based upon this calculation. The top m features are selected for inclusion in the relevant feature subset.

The three feature selection methods presented in this study are compared based on classification accuracy of the HMM on the tool condition problem. The proposed feature discriminant analysis outperforms PCA and ARD.

Wissel et al. (2013) compare HMMs to support vector machines on the task of classifying finger movements. The study concludes that both classifiers can accurately classify the movements and that separation between the two classifiers relies on the extracted and selected features. The study specifically states that a wrapper method is not appropriate because the classifier could influence the feature selection process. A filtering feature selection algorithm is proposed based on the Davies-Bouldin (DB) index (Davies and Bouldin 1979; Palaniappan and Wissel 2011). For an arbitrary clustering problem, let $\mathbf{r}$ represent a high-dimensional vector with L features and C_k the cluster assignment. The DB index for clusters is

$$R_{ij} = \frac{d_i + d_j}{||\boldsymbol{\mu}_i - \boldsymbol{\mu}_j||}, \tag{75}$$

where

$$d_i = \frac{1}{N} \sum_{\mathbf{r} \in C_k} ||\mathbf{r} - \boldsymbol{\mu}_i||, \tag{76}$$

and

$$\boldsymbol{\mu}_i = \frac{1}{N} \sum_{\mathbf{r} \in C_k} \mathbf{r}. \tag{77}$$

For application to HMMs, the cluster assignment is replaced by the state assignment so this method requires knowledge of the hidden state. The sequence of observations Y is divided into three equal segments. The DB index is calculated on each segment and averaged to form a DB measure for each feature under consideration. The features with the smallest average DB index are added to the feature set until a desired number of features is reached.

5.3 Wrappers for HMMs

A number of search methods through the feature space are explored by Charlet and Jouvet (1997) to find the optimal feature subset for HMMs in the speaker verification problem. Specifically, the four search methods in this study and the number of feature subset evaluations for each of these search methods given L features are

- L-best method: the best feature subset composed of d features is found by selecting the top d when the features are evaluated separately. This requires $2L$ evaluations.
- Ascendant selection: sequential forward search by adding a feature to the feature subset. This requires $\frac{L(L+1)}{2}$ evaluations.
- Knock-out procedure: sequential backward search by removing a feature from the feature subset. This requires $\frac{L(L+1)}{2}$ evaluations.

– Selection by dynamic programming: use dynamic programming to search the feature space (Cheung and Eisenstein 1978) and assume the best feature subset is not embedded in the previous best feature subset. This requires $\frac{L^2(L-1)}{2}$ evaluations.

Charlet and Jouvet develop an evaluation function based on the probability that speaker M produced the speech signal Y. The probability that X was spoken by M for a given word is

$$P(Y, \hat{x}|M) = \pi_{x_1} \prod_{t=2}^{T} a_{x_{t-1},x_t} f_{x_t}(y_t), \tag{78}$$

where $\hat{x}$ is the optimal path. The negative log-likelihood of the emission probability is labeled as the $Score(Y)$. If the emission probability is assumed to be a Gaussian distribution with a diagonal covariance matrix, the total $Score(Y)$ is composed of the sum of the score of the individual features and the feature independent component K

$$Score(Y) = \sum_{l=1}^{L} Score_l(Y_l) + K. \tag{79}$$

A subset of features S can be evaluated using this scoring function by

$$Score_S(Y) = \sum_{l \in S} Score_l(Y). \tag{80}$$

This study concludes that the L-best method performs significantly worse than the other three search methods. The three remaining search methods, ascending, knock-out, and dynamic programming, all reduce the error rate in the speaker verification problem.

Günter and Bunke (2003) propose a fast feature selection method for HMMs. We consider the proposed method a wrapper because it evaluates feature sets using recognition rate. The novelty of this study is the way recognition rate is calculated. However, there is no proposed search method through the feature space and all feature subsets are evaluated. The authors propose two approaches for selecting the final feature subset. The first selects the feature subset with the highest recognition rate. The second evaluates the feature subsets on a validation set then selects the best subset of each size to be evaluated on a withheld test set. The feature subset with the highest recognition rate on the test set is selected as the final feature subset. This method requires supervised data because recognition rate can only be calculated using speaker labels.

The calculation of recognition rate is modified under the assumption that two HMMs with the same topology but different feature subsets often have the same or very similar optimal paths given an observation sequence. This assumption improves the speed of calculating the recognition rate because the optimal path must only be calculated once while other methods using recognition rate require the optimal path to be calculated for each feature subset. This speed up in calculation allows for an exhaustive search of the the feature subset space.

Lv and Nevatia (2006) use boosting to perform feature selection when using HMMs to classify human actions. For each feature and each class, a single HMM is trained. The AdaBoost algorithm is used to learn weights for each feature by increasing the weight for misclassified observations, which requires supervised data or class labels. While this study does not use the proposed method to select features, the learned weights could be used to select features and thus select the most powerful HMMs for the ensemble classifier. Supervised data is required in order to identify misclassified observations.

5.4 Embedded methods for HMMs

The idea of penalizing the model to perform feature selection originally proposed for GMMs can be extended to HMMs. Städler and Mukherjee (2013) propose a penalized estimation for HMMs, however the focus of this method is not feature selection but producing a sparse inverse covariance matrix and estimating the number of states. This method could be easily extended to feature selection by removing the features based on the estimated diagonal elements of the covariance matrix. The model parameters are estimated using a penalized negative log-likelihood function

$$\hat{\Lambda}_{I,\lambda} = \text{argmin}_{\Lambda_{I,\lambda}} - l(\Lambda_{I,\lambda}) + \lambda \text{pen}(\Lambda_{I,\lambda}), \tag{81}$$

where $l(\Lambda_{I,\lambda})$ represents the observed log-likelihood, and $\text{pen}(\Lambda_{I,\lambda})$ is the penalty function. Three penalty functions are tested in this study, but all use L_1 constraints to lead to a sparse inverse covariance matrix. The optimal number of states can be determined using a backward pruning procedure that minimizes a model selection criteria such as BIC or mixture minimum description length (Figueiredo et al. 1999).

Feature saliency, first developed for feature and model selection for GMMs, has been adapted to HMMs. Zhu et al. (2012) first adapt feature saliency to HMMs and use a variational Bayesian (VB) method to jointly estimate model parameters and select features. This method does not require the number of states – or the number of mixtures if a GMM is used for the emission distribution – to be known *a priori*. Let ϕ represent the binary variable for feature relevance, where if $\phi_l = 1$ then the lth feature is relevant and belongs to the state-dependent distribution, and if $\phi_l = 0$ the lth feature is irrelevant and belongs to the state-independent distribution. The emission probability is assumed to be a mixture of Gaussian distributions. The conditional likelihood can be written as

$$P(y_t|\Lambda_i, \phi, \epsilon, \tau) = \sum_{m=1}^{M} c_{im} \prod_{l=1}^{L} \left[p(y_t|\mu_{ilm}, \Sigma_{ilm})^{\phi_{lt}} q(y_t|\epsilon_l, \tau_l)^{1-\phi_{lt}} \right], \tag{82}$$

where Λ_i is the set of model parameters for the state-dependent distribution $p(\cdot|\cdot)$ when in state i, c_{im} is the mixture probability, μ is the mean of $p(\cdot|\cdot)$, Σ is the inverse covariance matrix (note that in most feature saliency formulations the covariance instead inverse covariance is used) of $p(\cdot|\cdot)$, and ϵ and τ are the model parameters for the state-independent distribution $q(\cdot|\cdot)$.

For the VB method, the set of parameters is divided into a subset of model parameters and random variables. The random variables require a prior distribution. The distribution for ϕ is

$$P(\phi) = \prod_{t=1}^{T} \prod_{l=1}^{L} (\rho_l)^{\phi_{lt}} (1 - \rho_l)^{1-\phi_{lt}}. \tag{83}$$

The prior on μ is assumed to be Gaussian, the prior on Σ is a Gamma distribution, and the mixture probability and initial probability for the Markov chain have Dirichlet priors. In order to select the number of states from an infinite set, a stick-breaking representation (Paisley and Carin 2009) is used for the transition probabilities

$$a_{ij} = V_{ij} \prod_{k=1}^{j-1} (1 - V_{ik}), \tag{84}$$

$$V_{ij} \sim B(1, \alpha_{ij}), \tag{85}$$

and

$$\alpha_{ij} \sim G(e, f). \tag{86}$$

In this representation, $B(1, \alpha_{ij})$ is a beta distribution, and $G(e, f)$ is a Gamma distribution. Using this stick-breaking prior, the number of states can be overestimated when the algorithm is initialized, and then reduced during parameter estimation.

Zhu, He, and Leung propose using the mean field approximation to factor the approximating distribution. However, VB can underestimate the variance for the approximate distribution when using the mean field assumption (Consonni and Marin 2007). Further, Chatzis and Kosmopoulos (2011) demonstrate that VB gives poor estimates for the number of states and model parameters when using an HMM with Gaussian emissions in the presence of outliers. The feature saliency ρ and the parameters for the state-independent distribution ϵ and τ are all considered model parameters.

Adams et al. (2016) further adapt feature saliency to HMMs by allowing for the inclusion of the cost of collecting each feature. They propose using maximum a posteriori (MAP) estimation to solve for model parameters. The cost of each feature is conveyed to the feature selection algorithm through the prior distribution on the feature saliency parameter.

For MAP estimation using the EM algorithm, the $\mathcal{Q}$ function is modified to include the prior distributions $G(\Lambda)$

$$\mathcal{Q}(\Lambda, \Lambda') + \log(G(\Lambda)). \tag{87}$$

The proposed prior on ρ is a truncated exponential distribution with a rate parameter k

$$\rho_l \sim \frac{1}{Z} e^{-k_l \rho_l}, \tag{88}$$

where Z is the normalizing factor. Figure 12 displays a graphical model for the MAP formulation presented in Adams et al. (2016). The rate parameter is tuned to reflect the cost of each feature. A beta prior for ρ is also proposed, but is shown to underestimate the value of relevant features during numerical experimentation. The proposed MAP feature saliency method is compared to a maximum likelihood method that does not use priors on the model parameters and the VB method in Zhu et al. (2012) on a synthetic data set, a tool wear data set, and an activity recognition data set. The MAP method removes more features and targets the more expensive features than the other two methods without a significant drop in accuracy. Further, this method is extended to hidden semi-Markov models (HSMMs). HSMMs are a variant of HMMs which model a duration or sojourn time in each state (Yu 2010).

As with GMMs, there is a constrained form of HMMs called the GTM through time (GTMTT) (Bishop et al. 1997). The constraints take the form of a shared covariance matrix across states and observation means that do not move independently. Olier and Vellido (2008) extend the feature saliency GTM described in Vellido (2006) to a time series setting. The extension is straight forward, and the EM algorithm is used to solve for model parameters. The proposed method is validated on several synthetic and real data sets.

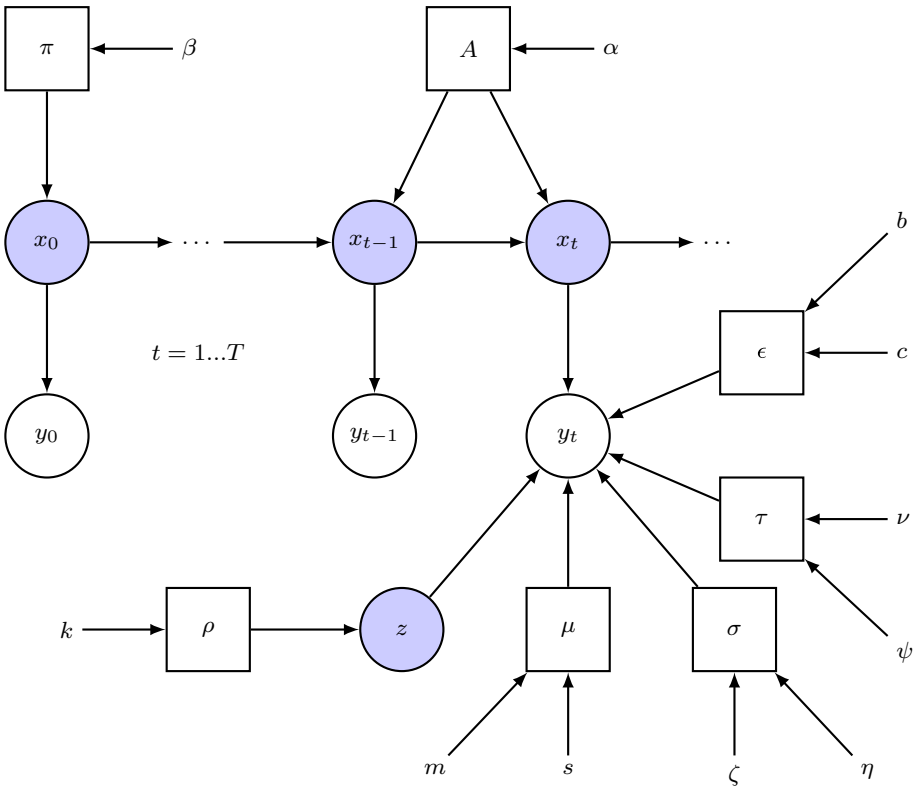

Fig. 12 Graphical model for MAP FSHMM formulation from Adams et al. (2016). Shaded circles represent hidden variables. Circles are observable variables. Squares are model parameters

6 Discussion and conclusion

This survey outlines feature selection techniques specifically designed for GMMs and HMMs. While there are numerous general feature selection techniques which could be used with these latent variable models, the literature demonstrates that custom methods often outperform the general methods. The vast majority of general feature selection methods require some form of supervised data. When applied to GMMs and HMMs, supervised data takes the form of knowledge about the class or label of each observation or observation sequence, or knowledge about the latent variable. However, GMMs and HMMs are often applied in areas where this knowledge is either completely unavailable or difficult to acquire. Therefore, unsupervised feature selection methods are often required when performing feature selection on data modeled using GMMs and HMMs. The GMM and HMM specific feature selection methods often account for the fact that supervised data is unavailable. Further, the general unsupervised feature selection methods in the literature do not take into account the model structure. Thus, the specific methods generally outperform the general feature selection methods.

The number of feature selection methods for GMMs is much greater than the number of methods for HMMs. We discuss both models in this review because they are both latent variable models and have similar properties. We propose that methods developed for one model could be easily transferable to the other model and that the benefits of the feature

selection methods would likewise be transferred. This is illustrated by the feature saliency method which was developed for GMMs but was later adapted and applied to HMMs.

Feature selection methods are divided into three primary types. This survey demonstrates that embedded methods, which simultaneously estimate model parameters and select features, are preferred when developing feature selection techniques for the GMM and HMM communities. The unsupervised training algorithms and parameter estimation procedures for GMMs and HMMs are often iterative and can be computationally expensive. A wrapper technique, which cycles between selecting a feature subset and constructing a model, can quickly become unfeasible in this setting. The embedded procedures help alleviate the high computational cost of feature selection and model training.

The proportion of unsupervised feature selection methods for HMMs is far less than the proportion of unsupervised methods for GMMs. Future work in feature selection for HMMs should focus on two areas: (1) transferring proven methods from GMMs to HMMs, and (2) developing more unsupervised methods.

References

Adams S, Beling PA, Cogill R (2016) Feature selection for hidden Markov models and hidden semi-Markov models. IEEE Access 4:1642–1657

Aha DW, Bankert RL (1995) A comparative evaluation of sequential feature selection algorithms. In: Proceedings of the fifth international workshop on artificial intelligence and statistics

Allili MS, Bouguila N, Ziou D (2008) Finite general Gaussian mixture modeling and application to image and video foreground segmentation. J Electron Imaging 17(1):013,005–013,005

Allili MS, Ziou D, Bouguila N, Boutemedjet S (2010) Image and video segmentation by combining unsupervised generalized Gaussian mixture modeling and feature selection. IEEE Trans Circuits Syst Video Technol 20(10):1373–1377

Almuallim H, Dietterich TG (1991) Learning with many irrelevant features. In: AAAI, vol 91. Citeseer, pp 547–552

Bagos PG, Liakopoulos TD, Hamodrakas SJ (2004) Faster gradient descent training of hidden Markov models, using individual learning rate adaptation. In: International colloquium on grammatical inference. Springer, pp 40–52

Bahl L, Brown PF, De Souza PV, Mercer RL (1986) Maximum mutual information estimation of hidden Markov model parameters for speech recognition. In: Proceedings of ICASSP, vol 86, pp 49–52

Bashir FI, Khokhar AA, Schonfeld D (2007) Object trajectory-based activity classification and recognition using hidden Markov models. IEEE Trans Image Process 16(7):1912–1919

Bhattacharya S, McNicholas PD (2014) A LASSO-penalized BIC for mixture model selection. Adv Data Anal Classif 8(1):45–61

Bilmes J (1998) A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. Int Comput Sci Inst 4(510):126

Bins J, Draper BA (2001) Feature selection from huge feature sets. In: Eighth IEEE international conference on computer vision, 2001. ICCV 2001. Proceedings, vol 2. IEEE, pp 159–165

Bishop CM, Hinton GE, Strachant IG (1997) GTM through time. In: Proceedings of the IEEE fifth international conference on artificial neural networks. Citeseer

Bishop CM, Svensén M, Williams CK (1998) GTM: the generative topographic mapping. Neural Comput 10(1):215–234

Blum AL, Langley P (1997) Selection of relevant features and examples in machine learning. Artif Intell 97(1):245–271

Bocchieri E (1993) Vector quantization for the efficient computation of continuous density likelihoods. In: 1993 IEEE international conference on acoustics, speech, and signal processing, 1993. ICASSP-93, vol 2. IEEE, pp 692–695

Boutemedjet S, Bouguila N, Ziou D (2007) Feature selection for non Gaussian mixture models. In: 2007 IEEE workshop on machine learning for signal processing. IEEE, pp 69–74

Bouveyron C, Brunet C (2012) Simultaneous model-based clustering and visualization in the fisher discriminative subspace. Stat Comput 22(1):301–324

Bouveyron C, Brunet-Saumard C (2014) Discriminative variable selection for clustering with the sparse Fisher-EM algorithm. Comput Stat 29(3–4):489–513
Boys RJ, Henderson DA (2001) A comparison of reversible jump MCMC algorithms for DNA sequence segmentation using hidden Markov models. Comput Sci Stat 33:35–49
Cappé O, Buchoux V, Moulines E (1998) Quasi-Newton method for maximum likelihood estimation of hidden Markov models. In: Proceedings of the 1998 IEEE international conference on acoustics, speech and signal processing, 1998, vol 4. IEEE, pp 2265–2268
Carbonetto P, De Freitas N, Gustafson P, Thompson N (2003) Bayesian feature weighting for unsupervised learning, with application to object recognition. In: Artificial intelligence and statistics (AI & Statistics' 03). Society for Artificial Intelligence and Statistics
Caruana R, Freitag D (1994) Greedy attribute selection. In: ICML. Citeseer, pp 28–36
Caruana R, Freitag D (1994) How useful is relevance? FOCUS 14(8):2
Celeux G, Martin-Magniette ML, Maugis-Rabusseau C, Raftery AE (2014) Comparing model selection and regularization approaches to variable selection in model-based clustering. Journal de la Societe francaise de statistique (2009) 155(2):57
Chang S, Dasgupta N, Carin L (2005) A Bayesian approach to unsupervised feature selection and density estimation using expectation propagation. In: IEEE Computer society conference on computer vision and pattern recognition, 2005. CVPR 2005, vol 2. IEEE, pp 1043–1050
Charlet D, Jouvet D (1997) Optimizing feature set for speaker verification. In: International conference on audio-and video-based biometric person authentication. Springer, pp 203–210
Chatzis SP, Kosmopoulos DI (2011) A variational Bayesian methodology for hidden Markov models utilizing Student's-t mixtures. Pattern Recognit 44(2):295–306
Cheung R, Eisenstein B (1978) Feature selection via dynamic programming for text-independent speaker identification. IEEE Trans Acoust Speech Signal Process 26(5):397–403
Cheung Ym (2004) A rival penalized EM algorithm towards maximizing weighted likelihood for density mixture clustering with automatic model selection. In: Proceedings of the 17th international conference on Pattern recognition, 2004. ICPR 2004, vol 4. IEEE, pp 633–636
Cheung Ym (2005) Maximum weighted likelihood via rival penalized EM for density mixture clustering with automatic model selection. IEEE Trans Knowl Data Eng 17(6):750–761
Consonni G, Marin JM (2007) Mean-field variational approximate Bayesian inference for latent variable models. Comput Stat Data Anal 52(2):790–798
Constantinopoulos C, Titsias MK, Likas A (2006) Bayesian feature and model selection for Gaussian mixture models. IEEE Trans Pattern Anal Mach Intell 28(6):1013–1018
Corduneanu A, Bishop CM (2001) Variational Bayesian model selection for mixture distributions. In: Artificial intelligence and statistics, vol 2001. Morgan Kaufmann Waltham, MA, pp 27–34
Cover TM, Van Campenhout JM (1977) On the possible orderings in the measurement selection problem. IEEE Trans Syst Man Cybern 7(9):657–661
Daelemans W, Hoste V, De Meulder F, Naudts B (2003) Combined optimization of feature selection and algorithm parameters in machine learning of language. In: Machine learning: ECML 2003. Springer, pp 84–95
Dash M, Liu H (1997) Feature selection for classification. Intell Data Anal 1(3):131–156
Dash M, Liu H, Motoda H (2000) Consistency based feature selection. In: Knowledge discovery and data mining. Current issues and new applications. Springer, pp 98–109
Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 2:224–227
Doak J (1992) An evaluation of feature selection methods and their application to computer security. University of California, Computer Science
Duda RO, Hart PE, Stork DG (2001) Pattern classification, 2nd edn. Wiley, New York
Dy JG (2008) Unsupervised feature selection. Computational methods of feature selection, pp 19–39
Dy JG, Brodley CE (2000) Feature subset selection and order identification for unsupervised learning. In: ICML, pp 247–254
Dy JG, Brodley CE (2004) Feature selection for unsupervised learning. J Mach Learn Res 5:845–889
Figueiredo MAT, Jain AK, Law MH (2003) A feature selection wrapper for mixtures. In: Perales FJ, Campilho AJC, de la Blanca NP, Sanfeliu A (eds) Pattern recognition and image analysis. IbPRIA 2003. Lecture notes in computer science, vol 2652. Springer, Berlin, pp 229–237
Figueiredo MA, Leitão JM, Jain AK (1999) On fitting mixture models. In: International workshop on energy minimization methods in computer vision and pattern recognition. Springer, pp 54–69
Forman G (2003) An extensive empirical study of feature selection metrics for text classification. J Mach Learn Res 3:1289–1305
Frühwirth-Schnatter S (2001) Markov chain Monte Carlo estimation of classical and dynamic switching and mixture models. J Am Stat Assoc 96(453):194–209

Gales MJ (1999) Semi-tied covariance matrices for hidden Markov models. IEEE Trans Speech Audio Process 7(3):272–281
Gales MJ, Knill KM, Young SJ (1999) State-based Gaussian selection in large vocabulary continuous speech recognition using HMMs. IEEE Trans Speech Audio Process 7(2):152–161
Galimberti G, Manisi A, Soffritti G (2017) Modelling the role of variables in model-based cluster analysis. Stat Comput 1–25
Galimberti G, Montanari A, Viroli C (2009) Penalized factor mixture analysis for variable selection in clustered data. Comput Stat Data Anal 53(12):4301–4310
Godino-Llorente JI, Gomez-Vilda P, Blanco-Velasco M (2006) Dimensionality reduction of a pathological voice quality assessment system based on Gaussian mixture models and short-term cepstral parameters. IEEE Trans Biomed Eng 53(10):1943–1953
Graham MW, Miller DJ (2006) Unsupervised learning of parsimonious mixtures on large spaces with integrated feature and component selection. IEEE Trans Signal Process 54(4):1289–1303
Günter S, Bunke H (2003) Fast feature selection in an HMM-based multiple classifier system for handwriting recognition. In: Joint pattern recognition symposium. Springer, pp 289–296
Guo J, Levina E, Michailidis G, Zhu J (2010) Pairwise variable selection for high-dimensional model-based clustering. Biometrics 66(3):793–804
Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182
Jain AK, Duin RP, Mao J (2000) Statistical pattern recognition: a review. IEEE Trans Pattern Anal Mach Intell 22(1):4–37
Jasra A, Holmes C, Stephens D (2005) Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. Stat Sci 50–67
Ji S, Krishnapuram B, Carin L (2006) Variational Bayes for continuous hidden Markov models and its application to active learning. IEEE Trans Pattern Anal Mach Intell 28(4):522–532
John GH, Kohavi R, Pfleger K (1994) Irrelevant features and the subset selection problem. In: Machine learning: proceedings of the eleventh international conference, pp 121–129
Kerroum MA, Hammouch A, Aboutajdine D (2010) Textural feature selection by joint mutual information based on Gaussian mixture model for multispectral image classification. Pattern Recognit Lett 31(10):1168–1174
Khreich W, Granger E, Miri A, Sabourin R (2012) A survey of techniques for incremental learning of HMM parameters. Inf Sci 197:105–130
Kim S, Tadesse MG, Vannucci M (2006) Variable selection in clustering via Dirichlet process mixture models. Biometrika 93(4):877–893
Kira K, Rendell LA (1992) The feature selection problem: traditional methods and a new algorithm. AAAI 2:129–134
Kohavi R, John GH (1997) Wrappers for feature subset selection. Artif Intell 97(1):273–324
Kononenko I (1994) Estimating attributes: analysis and extensions of Relief. In: Machine learning: ECML-94. Springer, pp 171–182
Krishnan S, Samudravijaya K, Rao P (1996) Feature selection for pattern classification with Gaussian mixture models: a new objective criterion. Pattern Recognit Lett 17(8):803–809
Law MH, Figueiredo MA, Jain AK (2004) Simultaneous feature selection and clustering using mixture models. IEEE Trans Pattern Anal Mach Intell 26(9):1154–1166
Law MH, Jain AK, Figueiredo M (2002) Feature selection in mixture-based clustering. In: Advances in neural information processing systems, pp 625–632
Li X, Bilmes J (2003) Feature pruning in likelihood evaluation of HMM-based speech recognition. In: 2003 IEEE workshop on automatic speech recognition and understanding, 2003. ASRU'03. IEEE, pp 303–308
Li X, Bilmes J (2005) Feature pruning for low-power ASR systems in clean and noisy environments. IEEE Signal Process Lett 12(7):489–492
Li Y, Dong M, Hua J (2008) Localized feature selection for clustering. Pattern Recognit Lett 29(1):10–18
Li Y, Dong M, Hua J (2009) Simultaneous localized feature selection and model detection for Gaussian mixtures. IEEE Trans Pattern Anal Mach Intell 31(5):953–960
Liu H, Yu L (2005) Toward integrating feature selection algorithms for classification and clustering. IEEE Trans Knowl Data Eng 17(4):491–502
Liu X, Chen T (2003) Video-based face recognition using adaptive hidden Markov models. In: 2003 IEEE computer society conference on computer vision and pattern recognition, 2003. Proceedings, vol 1. IEEE, pp I–340
Liu X, Gong Y, Xu W, Zhu S (2002) Document clustering with cluster refinement and model selection capabilities. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, pp 191–198

Lv F, Nevatia R (2006) Recognition and segmentation of 3-d human action using HMM and multi-class adaboost. In: Computer vision–ECCV 2006. Springer, pp 359–372
MacKay DJ (1992) A practical Bayesian framework for backpropagation networks. Neural Comput 4(3):448–472
Marbac M, Sedki M (2017) Variable selection for model-based clustering using the integrated complete-data likelihood. Stat Comput 27(4):1049–1063
Maugis C, Celeux G, Martin-Magniette ML (2009) Variable selection for clustering with Gaussian mixture models. Biometrics 65(3):701–709
Maugis C, Celeux G, Martin-Magniette ML (2009) Variable selection in model-based clustering: a general variable role modeling. Comput Stat Data Anal 53(11):3872–3882
Maugis C, Michel B (2011) A non asymptotic penalized criterion for Gaussian mixture model selection. ESAIM Probab Stat 15:41–68
McGrory CA, Titterington D (2009) Variational Bayesian analysis for hidden Markov models. Aust N Z J Stat 51(2):227–244
McLachlan GJ, Peel D (2000) Mixtures of factor analyzers. In: Proceedings of the seventeenth international conference on machine learning. Morgan Kaufmann Publishers Inc, pp 599–606
Merialdo B (1988) Phonetic recognition using hidden Markov models and maximum mutual information training. In: 1988 international conference on acoustics, speech, and signal processing, 1988. ICASSP-88. IEEE, pp 111–114
Meyer C (2002) Utterance-level boosting of HMM speech recognizers. In: 2002 IEEE international conference on acoustics, speech, and signal processing (ICASSP), vol 1. IEEE, pp I–109
Minka TP (2001) Expectation propagation for approximate Bayesian inference. In: Proceedings of the seventeenth conference on uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc, pp 362–369
Mitra P, Murthy C, Pal SK (2002) Unsupervised feature selection using feature similarity. IEEE Trans Pattern Anal Mach Intell 24(3):301–312
Molina LC, Belanche L, Nebot À (2002) Feature selection algorithms: a survey and experimental evaluation. In: 2002 IEEE international conference on data mining, 2002. ICDM 2003. Proceedings. IEEE, pp 306–313
Montero JA, Sucar LE (2004) Feature selection for visual gesture recognition using hidden Markov models. In: Proceedings of 5th international conference on computer science, 2004. ENC 2004. IEEE, pp 196–203
Murphy KP (2012) Machine learning: a probabilistic perspective. The MIT Press, Cambridge
Narendra PM, Fukunaga K (1977) A branch and bound algorithm for feature subset selection. IEEE Trans Comput 100(9):917–922
Ng AY (1998) On feature selection: learning with exponentially many irrelevant features as training examples. In: Proceedings of the fifteenth international conference on machine learning. Morgan Kaufmann Publishers Inc, pp 404–412
Nouza J (1996) Feature selection methods for hidden Markov model-based speech recognition. In: Proceedings of 13th international conference on pattern recognition vol 2, pp 186–190
Novovicová J, Pudil P, Kittler J (1996) Divergence based feature selection for multimodal class densities. IEEE Trans Pattern Anal Mach Intell 18(2):218–223
Olier I, Vellido A (2008) Advances in clustering and visualization of time series using GTM through time. Neural Netw 21(7):904–913
Paisley J, Carin L (2009) Hidden Markov models with stick-breaking priors. IEEE Trans Signal Process 57(10):3905–3917
Palaniappan R, Wissel T (2011) Considerations on strategies to improve EOG signal analysis. Int J Artif Life Res 2(3):6–21
Paliwal K (1992) Dimensionality reduction of the enhanced feature set for the HMM-based speech recognizer. Digital Signal Process 2(3):157–173
Pan W, Shen X (2007) Penalized model-based clustering with application to variable selection. J Mach Learn Res 8:1145–1164
Pan W, Shen X, Jiang A, Hebbel RP (2006) Semi-supervised learning via penalized mixture model with application to microarray sample classification. Bioinformatics 22(19):2388–2395
Pudil P, Ferri F, Novovicova J, Kittler J (1994a) Floating search methods for feature selection with nonmonotonic criterion functions. In: Proceedings of the twelveth international conference on pattern recognition, IAPR. Citeseer
Pudil P, Novovičová J, Kittler J (1994b) Floating search methods in feature selection. Pattern Recognit Lett 15(11):1119–1125
Pudil P, Novovičová J, Choakjarernwanit N, Kittler J (1995) Feature selection based on the approximation of class densities by finite mixtures of special type. Pattern Recognit 28(9):1389–1398
Rabiner L (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proc IEEE 77(2):257–286

Raftery AE, Dean N (2006) Variable selection for model-based clustering. J Am Stat Assoc 101(473):168–178
Ribeiro PC, Santos-Victor J (2005) Human activity recognition from video: modeling, feature selection and classification architecture. In: Proceedings of international workshop on human activity recognition and modelling. Citeseer, pp 61–78
Richardson S, Green PJ (1997) On Bayesian analysis of mixtures with an unknown number of components (with discussion). J R Stat Soc Ser B (Stat Methodol) 59(4):731–792
Robert CP, Ryden T, Titterington DM (2000) Bayesian inference in hidden Markov models through the reversible jump Markov chain Monte Carlo method. J R Stat Soc Ser B (Stat Methodol) 62(1):57–75
Robnik-Šikonja M, Kononenko I (2003) Theoretical and empirical analysis of ReliefF and RReliefF. Mach Learn 53(1–2):23–69
Rydén T et al (2008) EM versus Markov chain Monte Carlo for estimation of hidden Markov models: a computational perspective. Bayesian Anal 3(4):659–688
Saeys Y, Inza I, Larrañaga P (2007) A review of feature selection techniques in bioinformatics. Bioinformatics 23(19):2507–2517
Schwenk H (1999) Using boosting to improve a hybrid HMM/neural network speech recognizer. In: 1999 IEEE international conference on acoustics, speech, and signal processing, 1999. Proceedings, vol 2. IEEE, pp 1009–1012
Scott SL (2002) Bayesian methods for hidden Markov models: recursive computing in the 21st century. J Am Stat Assoc 97(457):337–351
Scrucca L (2016) Genetic algorithms for subset selection in model-based clustering. In: Unsupervised learning algorithms. Springer, pp 55–70
Somol P, Pudil P, Kittler J (2004) Fast branch & bound algorithms for optimal feature selection. IEEE Trans Pattern Anal Mach Intell 26(7):900–912
Städler N, Mukherjee S et al (2013) Penalized estimation in high-dimensional hidden Markov models with state-specific graphical models. Ann Appl Stat 7(4):2157–2179
Steinley D, Brusco MJ (2008) Selection of variables in cluster analysis: an empirical comparison of eight procedures. Psychometrika 73(1):125–144
Swartz MD, Mo Q, Murphy ME, Lupton JR, Turner ND, Hong MY, Vannucci M (2008) Bayesian variable selection in clustering high-dimensional data with substructure. J Agric Biol Environ Stat 13(4):407–423
Tadesse MG, Sha N, Vannucci M (2005) Bayesian variable selection in clustering high-dimensional data. J Am Stat Assoc 100(470):602–617
Valente F, Wellekens C (2004) Variational Bayesian feature selection for Gaussian mixture models. In: IEEE international conference on acoustics, speech, and signal processing, 2004. Proceedings.(ICASSP'04), vol 1. IEEE, pp I–513
Vannucci M, Stingo FC (2010) Bayesian models for variable selection that incorporate biological information. Bayesian Stat 9:659–678
Vellido A (2006) Assessment of an unsupervised feature selection method for generative topographic mapping. In: International conference on artificial neural networks. Springer, pp 361–370
Vellido A, Lisboa PJ, Vicente D (2006) Robust analysis of MRS brain tumour data using t-GTM. Neurocomputing 69(7):754–768
Vellido A, Velazco J (2008) The effect of noise and sample size on an unsupervised feature selection method for manifold learning. In: IEEE international joint conference on neural networks, 2008. IJCNN 2008 (IEEE world congress on computational intelligence). IEEE, pp 522–527
Wang S, Zhu J (2008) Variable selection for model-based high-dimensional clustering and its application to microarray data. Biometrics 64(2):440–448
Wei X, Li C (2011) The Student's t -hidden Markov model with truncated stick-breaking priors. IEEE Signal Process Lett 18(6):355–358
Windridge D, Bowden R (2005) Hidden Markov chain estimation and parameterisation via ICA-based feature-selection. Pattern Anal Appl 8(1–2):115–124
Wissel T, Pfeiffer T, Frysch R, Knight RT, Chang EF, Hinrichs H, Rieger JW, Rose G (2013) Hidden Markov model and support vector machine based decoding of finger movements using electrocorticography. J Neural Eng 10(5):056,020
Witten DM, Tibshirani R (2010) A framework for feature selection in clustering. J Am Stat Assoc 105(490):713–726
Xie B, Pan W, Shen X (2008) Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. Electron J Stat 2:168
Xie B, Pan W, Shen X (2008) Variable selection in penalized model-based clustering via regularization on grouped parameters. Biometrics 64(3):921–930
Xie B, Pan W, Shen X (2010) Penalized mixtures of factor analyzers with application to clustering high-dimensional microarray data. Bioinformatics 26(4):501–508

Xie L, Chang SF, Divakaran A, Sun H (2002) Structure analysis of soccer video with hidden Markov models. In: Proceedings of IEEE international conferene on acoustics, speech, and signal processing, vol 4

Yin P, Essa I, Rehg JM (2004) Asymmetrically boosted HMM for speech reading. In: Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition, 2004. CVPR 2004, vol 2. IEEE, p II-755

Yin P, Essa I, Starner T, Rehg JM (2008) Discriminative feature selection for hidden Markov models using segmental boosting. In: IEEE international conference on acoustics, speech and signal processing, 2008. ICASSP 2008. IEEE, pp 2001–2004

Yu L, Liu H (2003) Feature selection for high-dimensional data: a fast correlation-based filter solution. ICML 3:856–863

Yu SZ (2010) Hidden semi-Markov models. Artif Intell 174(2):215–243

Zeng H, Cheung YM (2009) A new feature selection method for Gaussian mixture clustering. Pattern Recognit 42(2):243–250

Zhou H, Pan W, Shen X (2009) Penalized model-based clustering with unconstrained covariance matrices. Electron J Stat 3:1473

Zhou J, Zhang XP (2008) An ICA mixture hidden Markov model for video content analysis. IEEE Trans Circuits Syst Video Technol 18(11):1576–1586

Zhu H, He Z, Leung H (2012) Simultaneous feature and model selection for continuous hidden Markov models. IEEE Signal Process Lett 19(5):279–282

Zhu K, Hong G, Wong Y (2008) A comparative study of feature selection for hidden Markov model-based micro-milling tool wear monitoring. Mach Sci Technol 12(3):348–369