CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Universidad Carlos III de Madrid e-Archivo

A New Distance Measure for Model-Based Sequence Clustering

Darío García-García,
Emilio Parrado Hernández, *Member*, *IEEE*,
and Fernando Díaz-de María, *Member*, *IEEE*

Abstract—We review the existing alternatives for defining model-based distances for clustering sequences and propose a new one based on the Kullback-Leibler divergence. This distance is shown to be especially useful in combination with spectral clustering. For improved performance in real-world scenarios, a model selection scheme is also proposed.

Index Terms—Clustering, sequential data, similarity measures.

1 Introduction

One of the most common assumptions made in Machine Learning is that the observation vectors are independently and identically distributed (i.i.d.). This is a reasonable assumption in a wide range of scenarios and provides useful simplifications that enable the development of efficient learning algorithms. Nonetheless, there are lots of areas where this assumption is far from being valid. For example, sometimes, the useful information is encoded in the way that data vectors evolve along time, so emphasis is required in modeling the system dynamics. Clearly, this cannot be optimally done from an i.i.d. perspective: It requires a sequential approach, where the minimal meaningful unit is not a data vector but a sequence of vectors. Moreover, each of these sequences can have a different length, this being an additional difficulty for the traditional machine learning methods, which mostly rely on comparing patterns living in the same vector space.

A first step toward the development of efficient machine learning techniques to address these problems is obtaining an adequate modeling that enables pattern comparison. There has been a lot of research in generative models for sequential data, some of the most well-known and successful paradigms being the hidden Markov models (HMMs) [1] and their extensions: hierarchical HMMs [2], buried Markov models [3], etc. They offer a good trade-off between computational complexity and expressive power, while at the same time being adequate models for lots of real-life processes.

In this work, we address the problem of clustering sequential data. Clustering is one of the most common and useful tasks in machine learning, so it is a well-studied problem. Many efficient algorithms exist for the usual case of equal-length feature vectors, and, among them, Spectral clustering (SC) [4] stands out as a state-of-the-art nonparametric technique that allows unsupervised classification without making any assumption about the underlying distribution of the data. Hierarchical clustering (HC) [5] is another widely used technique, especially when real hierarchical relations exist in the data. Results obtained by this clustering procedure can be highly descriptive and informative.

In order to apply these well-known clustering methods to the sequential data scenario, we face the difficulty of defining a meaningful distance measure for sequences. A popular framework is to first generate adequate models for the individual sequences in the data set and then use these models to obtain likelihood-based distances between sequences [6]. Based on this work, several other distance measures based on a likelihood matrix have been proposed [7], [8], [9], all of them being very similar in their philosophy.

We propose exploring a different approach and define a distance measure between sequences under the aforementioned framework by looking at the likelihood matrix from a probabilistic perspective. We regard the patterns created by the likelihoods of each of the sequences under the trained models as samples from the conditional likelihoods of the models given the data. This point of view differs largely from the existing distances. One of its differentiating properties is that it embeds information from the whole data set or just a subset of it into each pairwise distance between sequences. This gives rise to highly structured distance matrices, which can be exploited by spectral methods to give a very high performance in comparison with previous proposals. Moreover, we also tackle the issue of selecting an adequate representative subset of models, proposing a simple method for that purpose when using SC. This greatly increases the quality of the clustering in those scenarios where the underlying dynamics of the sequences do not adhere well to the employed models.

The rest of this paper is organized as follows: In Section 2, we review the general framework for clustering sequential data, together with the most employed tools within that framework, namely HMMs as generative models and hierarchical and SC, whose main characteristics are briefly outlined. The existing algorithms under this framework are also reviewed. Section 3 introduces our proposal of a new distance measure between sequences. Performance comparisons are carried out in Section 4, using both synthetic and real-world data. Finally, Section 5 collects the main conclusions of this work and sketches some promising lines for future research.

2 A Framework for Clustering Sequential Data

The work of Smyth [6] proposes a probabilistic model-based framework for sequence clustering. Given a data set $\mathcal{S} = \{S_1, \ldots, S_N\}$ of N sequences, it assumes that each of them is generated by a single model from a discrete pool. The main idea behind this framework is to model the individual sequences and then use the resulting models to obtain a length-normalized log-likelihood matrix $\mathbf{L}$, whose ijth element l_{ij} is defined as

$$l_{ij} = \log p_{ij} = \frac{1}{\mathrm{length}(S_j)} \log f_{\mathbf{S}}(S_j; \theta_i), \quad 1 \leq i, j \leq N, \tag{1}$$

where S_j is the jth sequence, θ_i is the model trained for the ith sequence, and $f_{\mathbf{S}}(\cdot; \theta_i)$ is the probability density function (pdf) over sequences according to model θ_i. Based on this matrix, a distance matrix $\mathbf{D}$ can be obtained so that the original variable-length sequence clustering problem is reduced to a typical similarity-based one.

The following sections will describe the most usual tools under this framework: HMMs for the individual sequence models and hierarchical and SC for the actual partitioning of the data set. Then, we briefly address the existing algorithms in the literature under this framework.

2.1 Hidden Markov Models

HMMs [1] are a type of parametric discrete state-space model, widely employed in signal processing and pattern recognition. Their success comes mainly from their relative low complexity compared to their expressive power and their ability to model

- *The authors are with the Department of Signal Theory and Communications, University Carlos III of Madrid, Avda. de la Universidad, 30, 28911, Leganés, Madrid, Spain. E-mail: {dggarcia, emipar, fdiaz}@tsc.uc3m.es.*

Manuscript received 25 June 2008; revised 18 Sept. 2008; accepted 23 Oct. 2008; published online 31 Oct. 2008.
Recommended for acceptance by K. Murphy.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-06-0382.
Digital Object Identifier no. 10.1109/TPAMI.2008.268.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.

naturally occurring phenomena. Its main field of application has traditionally been speech recognition [1], but they have also found success in a wide range of areas, from bioinformatics [10] to video analysis [11].

In an HMM, the (possibly multidimensional) observation $\mathbf{y}_t$ at a given time instant t (living in a space $\mathbf{Y}$) is generated according to a conditional pdf $f_{\mathbf{Y}}(\mathbf{y}_t|q_t)$, with q_t being the hidden state at time t. These states follow a time-homogeneous first-order Markov chain so that $P(q_t|q_{t-1}, q_{t-2}, \ldots, q_0) = P(q_t|q_{t-1})$. Bearing this in mind, an HMM θ can be completely defined by the following parameters:

- The discrete and finite set of K possible states $\mathcal{X} = \{x_1, x_2, \ldots, x_K\}$.
- The state transition matrix $\mathbf{A} = \{a_{ij}\}$, where each a_{ij} represents the probability of a transition between two states: $a_{ij} = P\big(q_{t+1} = x_j | q_t = x_i\big), 1 \leq i, j \leq K$.
- The emission pdf $f_{\mathbf{Y}}(\mathbf{y}_t|q_t)$.
- The initial probabilities vector $\pi = \{\pi_i\}$, where $1 \leq i \leq K$ and $\pi_i = P(q_0 = x_i)$.

The parameters of an HMM are traditionally learned using the Baum-Welch algorithm [1], which represents a particularization of the well-known Expectation-Maximization (EM) algorithm [12]. Its complexity is $O(K^2T)$ per iteration, with T being the length of the training sequence. An HMM can be seen as a simple Dynamic Bayesian Network (DBN) [13], an interpretation that provides an alternative way of training this kind of models by applying the standard algorithms for DBNs. This allows for a unified way of inference in HMMs and their generalizations.

2.2 Hierarchical Clustering

HC [5] algorithms organize the data into a hierarchical (tree) structure. The clustering proceeds in an iterative fashion in the following two ways: Agglomerative methods start by assigning each datum to a different cluster, and then merging similar clusters up to arriving at a single cluster that includes all data. Divisive methods initially consider the whole data set as a unique cluster that is recursively partitioned in such a way that the resulting clusters are maximally distant. In both cases, the resulting binary tree can be stopped at a certain depth to yield the desired number of clusters.

2.3 Spectral Clustering

Spectral clustering (SC) [4] casts the clustering problem into a graph partitioning one. Data instances form the nodes of a weighted graph whose edges represent the adjacency between data. The clusters are the partitions of the graph that optimize certain criteria. These criteria include the normalized cut that takes into account the ratio between the cut of a partition and the total connection of the generated clusters. To find these optimal partitions is an NP-hard problem, which can be relaxed to a generalized eigenvalue problem on the Laplacian matrix of the graph.

The spectral techniques have the additional advantage of providing a clear and well-founded way of determining the optimal number of clusters for a data set, based on the eigengap of the similarity matrix [14].

2.4 Existing Algorithms

The initial proposal for model-based sequential data clustering of [6] aims at fitting a single generative model to the entire set $\mathcal{S}$ of sequences. The clustering itself is part of the initialization procedure of the model. In the initialization step, each sequence S_i is modeled with an HMM θ_i. Then, the distance between two sequences S_i and S_j is defined based on the log-likelihood of each sequence, given the model generated for the other sequence

$$d_{SYM}^{ij} = \frac{1}{2}\big(l_{ij} + l_{ji}\big), \tag{2}$$

where l_{ij} represents the (length normalized) log-likelihood of sequence S_j under model θ_i. In fact, this is the symmetrized distance previously proposed in [15]. Given these distances, the data are partitioned using agglomerative HC with the "furthest-neighbor" merging heuristic.

The work in [7] inherits this framework for sequence clustering but introduces a new dissimilarity measure, called the BP metric

$$d_{BP}^{ij} = \frac{1}{2}\left\{\frac{l_{ij} - l_{ii}}{l_{ii}} + \frac{l_{ji} - l_{jj}}{l_{jj}}\right\}. \tag{3}$$

The BP metric takes into account how well a model represents the sequence it has been trained on, so it is expected to perform better than the symmetrized distance in cases where the quality of the models may vary along different sequences.

Another alternative distance within this framework is proposed in [8], namely,

$$d_{POR}^{ij} = \big|p_{ij} + p_{ji} - p_{ii} - p_{jj}\big|, \tag{4}$$

with p_{ij} as defined in (1).

Recently, the popularity of SC has motivated work in which these kinds of techniques are applied to the clustering of sequences. Yin and Yang [9] propose a distance measure resembling the BP metric,

$$d_{YY}^{ij} = \big|l_{ii} + l_{jj} - l_{ij} - l_{ji}\big|, \tag{5}$$

and then apply SC on a similarity matrix derived from the distance matrix by means of a Gaussian kernel. They reported good results in comparison to traditional parametric methods using initializations such as those proposed in [6] and [16], called Dynamic Time Warping (DTW).

Another example of applying SC to sequential data can be found in [17]. In this novel approach, the similarities between the probability distributions defined by the different HMMs are measured via a probability product kernel (PPK). This way, the calculation of the likelihood matrix is avoided and the similarity between two sequences is obtained using just the parameters of the models trained on each of them. Hence, this method falls out of the scope of the present paper.

3 Proposed Algorithm

Our proposal is based on the observation that the aforementioned methods define the distance between two sequences S_i and S_j solely using the models trained on them (θ_i and θ_j). We expect a better performance if we add into the distance some global characteristics of the data set. Moreover, since distances under this framework are obtained from a likelihood matrix, it seems natural to take the probabilistic nature of this matrix into account when selecting adequate distance measures.

Bearing this in mind, we propose a novel sequence distance measure based on the Kullback-Leibler (KL) divergence [18], which is a standard measure for the similarity between probability density functions.

The first step of our algorithm involves obtaining the likelihood matrix $\mathbf{L}$ as in (1) (we will assume at first that an HMM is trained for each sequence). The ith column of $\mathbf{L}$ represents the likelihood of the sequence S_i under each of the trained models. These models can be regarded as a set of "intelligently" sampled points from the model space, in the sense that they have been obtained according to the sequences in the data set. This way, they are expected to lie in the area of the model space $\boldsymbol{\theta}$ surrounding the HMMs that actually span the data space. Therefore, these trained models become a good discrete approximation $\tilde{\boldsymbol{\theta}} = \{\theta_1, \ldots, \theta_N\}$ to the model subspace of interest. If we normalize the likelihood matrix so that each column adds up to one, we get a

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.

new matrix $\mathbf{L}_N$ whose columns can be seen as the probability density functions over the approximated model space conditioned on each of the individual sequences

$$\mathbf{L}_N = \left[f_{\tilde{\boldsymbol{\theta}}}^{S_1}(\theta), \ldots, f_{\tilde{\boldsymbol{\theta}}}^{S_N}(\theta)\right].$$

This interpretation leads to the familiar notion of dissimilarity measurement between probability density functions, the KL divergence being a natural choice for this purpose. Its formulation for the discrete case is as follows:

$$D_{KL}(f_P \| f_Q) = \sum_i f_P(i) \log \frac{f_P(i)}{f_Q(i)}, \tag{6}$$

where f_P and f_Q are two discrete pdfs. Since the KL divergence is not a proper distance because of its asymmetry, a symmetrized version is used

$$D_{KL_{\mathrm{SYM}}}(f_P \| f_Q) = \frac{1}{2}\left[D_{KL}\big(f_P \| f_Q\big) + D_{KL}\big(f_Q \| f_P\big)\right]. \tag{7}$$

This way, the distance between the sequences S_i and S_j can be defined simply as

$$d^{ij} = D_{KL_{\mathrm{SYM}}}\left(f_{\tilde{\boldsymbol{\theta}}}^{S_i} \| f_{\tilde{\boldsymbol{\theta}}}^{S_j}\right). \tag{8}$$

This implies a change of focus from the probability of the sequences under the models to the likelihood of the models, given the sequences. Distances defined this way are obtained according to the patterns created by each sequence in the probability space spanned by the different models. With this approach, the distance measure between two sequences S_i and S_j involves information related to the rest of the data sequences, represented by their corresponding models.

This redundancy can be used to define a representative subset $\mathcal{Q} \subseteq \mathcal{S}$ of the sequences, so that $\tilde{\boldsymbol{\theta}} = \{\theta_{Q_1}, \ldots, \theta_{Q_P}\}, P \leq N$. In this way, instead of using the whole data set for the calculation of the distances, only the models trained with sequences belonging to $\mathcal{Q}$ will be taken into account for that purpose. The advantage of defining such a subset is twofold: on the one hand, computational load can be reduced since the number of models to train is reduced to P and the posterior probability calculations drop from $N \times N$ to $P \times N$. On the other hand, if the data set is prone to outliers or the models suffer from overfitting, the stability of the distance measures and the clustering performance can be improved if $\mathcal{Q}$ is carefully chosen. Examples of both of these approaches are shown in the experiments included in Section 4. Obtaining this measure involves the calculation of $N(N-1)$ KL divergences, with a complexity linear in the number of elements in the representative subset. Therefore, its time complexity is $O(PN(N-1))$. Nevertheless, it is remarkable that the processing time devoted to the distance calculation is minimal in comparison to those involved in training the models and evaluating the likelihoods.

Finally, before applying an SC, the distance matrix $\mathbf{D} = \{d_{ij}\}$ must be transformed into a similarity matrix $\mathbf{A}$. A commonly used procedure is to apply a Gaussian kernel so that $a_{ij} = \exp(\frac{-d_{ij}^2}{2\sigma^2})$, with σ being a free parameter representing the kernel width. Next, a standard normalized-cut algorithm is applied to matrix $\mathbf{A}$, resulting in the actual clustering of the sequences in the data set. In the sequel, we will refer to this combination of our proposed KL-based distance and SC as KL+SC.

4 Experimental Results

This section presents some experimental results concerning several synthetic and real-world sequence clustering problems. Synthetic data experiments aim at illustrating the performance of the different sequence clustering algorithms under tough separability conditions but fulfilling the assumption that the sequences are generated by HMMs. This way, we focus the analysis on the impact of the distance measures as we isolate the adequateness of the modeling (except in overfitting). Besides, we also use two real-world scenarios, namely, speech data and electroencephalogram (EEG) data, to show a sample application of sequence clustering in two fields where HMMs have typically been used as rough approximate generative models.

The different methods to be compared are the following:

- SYM—Symmetrized distance (2)
- BP—BP distance (3)
- POR—Porikli distance (4)
- YY—Yin-Yang distance (5)
- KL—Proposed KL distance (8).

All of them will be paired with both an agglomerative HC using the furthest neighbor merging heuristic, as in [6], and a normalized-cut SC. For the SC algorithm, the value of parameter σ of the Gaussian kernel is selected empirically in a completely unsupervised fashion as the one that maximizes the eigengap for each distance measure in each case (as proposed in [14]). It is also remarkable that the k-means part of the SC algorithm, due to its strong dependence on the initialization, is run 10 times at each iteration and we choose as the most adequate partition the one with the minimal intracluster distortion, defined as

$$D_{cluster} = \sum_{k=1}^{K} \sum_{i \in \mathcal{C}_k} \|\mathbf{x}_i - \mathbf{c}_k\|^2,$$

where K is the number of clusters, $\mathcal{C}_k$ is the set of the indexes of points belonging to the kth cluster, $\mathbf{c}_k$ is the centroid of that cluster, and $\mathbf{x}_i$ is the ith data point. This distortion can be seen as the "tightness" of the resulting clusters, and it is also well known that this minimum distortion criterion implies a maximum separation among centroids [19].

Both the code and the data sets for the following experiments can be found at the authors' website.[1]

4.1 Synthetic Data

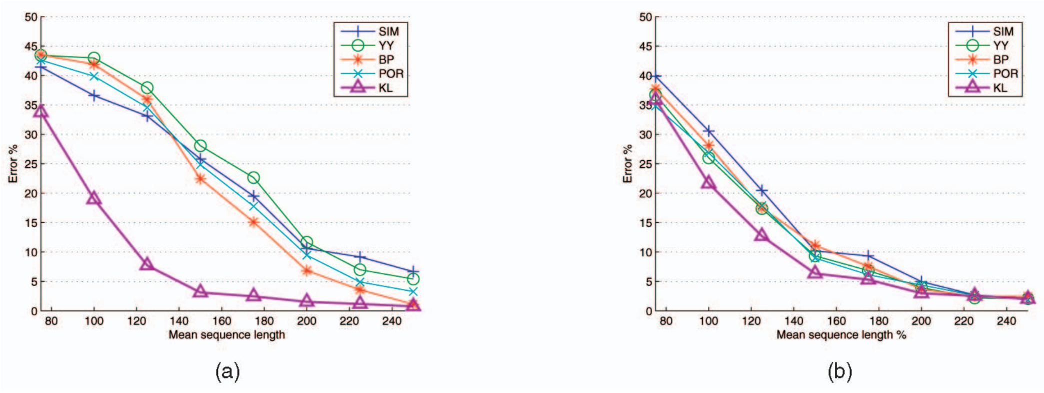
The first scenario under which the comparison is carried out is the original example from [6]: Each sequence in the data set is generated with equal probability by one of two possible HMMs θ_1 and θ_2, each one of them having two hidden states ($m = 2$). Transition matrices for the generating HMMs are given by

$$\mathbf{A}_1 = \begin{pmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{pmatrix}, \qquad \mathbf{A}_2 = \begin{pmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{pmatrix}.$$

Initial states are equiprobable and emission probabilities are the same in both models, specifically $N(0,1)$ in the first state and $N(3,1)$ in the second. This scenario represents a very appropriate test bed for sequence clustering techniques since the only way to differentiate sequences generated by each model is to attend to their dynamical characteristics. These, in turn, are very similar, making this a hard clustering task. The length of each individual sequence is obtained by sampling a uniform pdf in the range $[\mu_L(1 - V/100) \quad \mu_L(1 + V/100)]$, where μ_L is the sequence's mean length and V is a parameter which we will refer to as the percentage of variation in the length. All the given results are averaged over 50 randomly generated data sets.

[1] http://www.tsc.uc3m.es/~dggarcia.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 1. Clustering error percentage achieved by the compared methods against different mean sequence lengths ($V = 40\%$, $N = 80$). (a) SC. (b) HC.

Fig. 1 shows the results of the performance comparison of the different distance measures and clustering methods against variations of the mean length μ_L of the data sequences for a fixed length variation V of 40 percent in a data set comprised $N = 80$ sequences. It can be seen that, as expected, the longer the sequences the more accurate the clustering. It is also clear that our proposed distance measure outperforms the previous proposals under both hierarchical and SC, attaining specially good results using the latter technique. Specifically, the proposed KL+SC method yields the best performance for every mean sequence length, showing consistent improvements which are more dramatic for short mean sequence lengths ($\mu_L < 200$). Models trained with such short sequences suffer from severe overfitting, not being able to adequately capture the underlying dynamics, and thus giving unrealistic results when evaluated using the sequences in the data set. This results into incoherent distance matrices using the typical methods which render the use of SC algorithms unproductive. Nonetheless, our proposal is more resilient against this issue since it takes a global view on the data set that allows for the correct clustering of sequences even if the models are rather poor. Evaluating the sequences on a large enough number of individually inadequate models can generate patterns that our distance measure can capture, which translates into a consistent distance matrix very suitable for applying spectral methods. This shows that our approach is efficient even when the models are poor so they cannot be expected to correctly sample the model space. In these scenarios, the probabilistic interpretation of the proposed distance is not clear and it takes more of a pattern matching role.

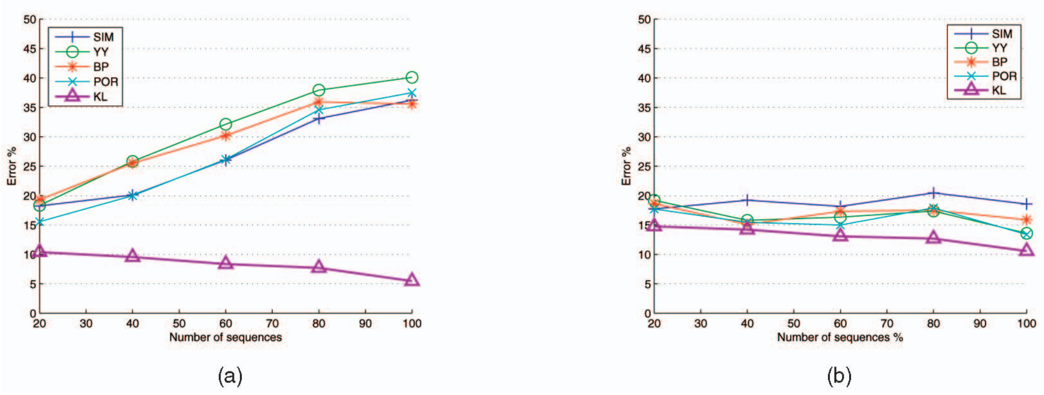
Agglomerative HC is more forgiving of loosely structured distance matrix since it merges clusters based on pairwise distance comparisons instead of taking a global view. Therefore, it seems more suitable than SC methods for its use with the previously proposed model-based sequence distances. On the other hand, it also implies that it cannot benefit from the use of our proposed distance as much as spectral techniques can.

Fig. 2 displays the evolution of the error along the number of sequences in the data set. As more sequences are present in the data set, the aforementioned problems of the previous proposals in combination with SC become clearer, while our method manages to improve its performance. Using HC, all the distances achieve stable results irrespective of the number of sequences, but once again, this comes at the expense of an inferior performance compared to the KL+SC combination.

Fig. 3 shows the results for a multiclass clustering with $K = 3$ classes. The sequences being clustered were generated using the two previously employed HMMs (θ_1 and θ_2) and a third one θ_3 that differs only from them in the transition matrix. Specifically,

$$\mathbf{A}_3 = \begin{pmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{pmatrix}.$$

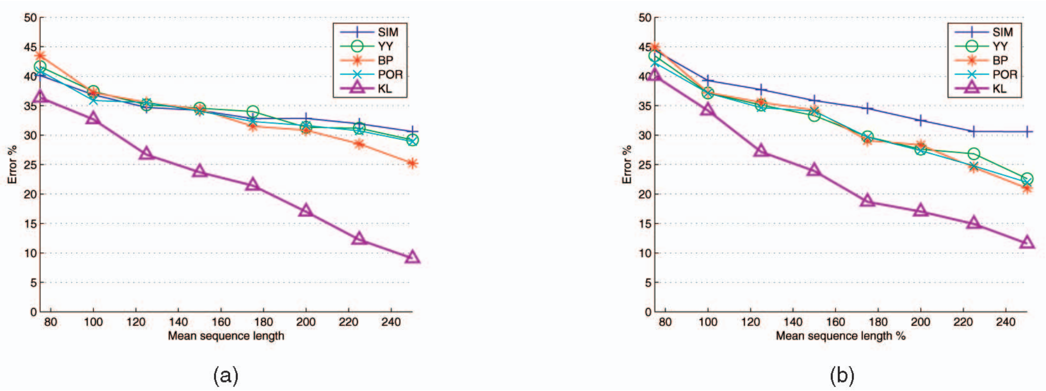
The additional class makes this a harder problem than the two-class scenario, so it is logical to assume that lengthier sequences are required to achieve comparable results. Nonetheless, the use of our proposed distance still shows significant improvements over the rest of the distances, all of which give almost identical results.



(a) (b)

Fig. 2. Clustering error against number of sequences in the data set ($\mu_L = 125, V = 40\%$). (a) SC. (b) HC.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.



(a)

(b)

Fig. 3. Performance in a multiclass ($K = 3$) clustering task against different mean sequence lengths ($V = 40$ percent, $N = 100$). (a) SC. (b) HC.

4.2 Real-World Data

In this section, different sequential-data clustering algorithms will be evaluated on real-world scenarios. The first scenario is speaker clustering: We are given a set of audio files, each one of them containing speech from a single speaker, and the task is to group together files coming from the same speaker (two speakers per experiment). Two different databases are employed, namely,

- **AHUMADA** [20]. A database specially tailored for speaker identification. We use a free subset[2] consisting of 25 speakers and choose the isolated digits task: each speaker records 24 digits, which are further concatenated in groups of two, giving 12 sequences per user with a mean length of 0.7 seconds.
- **GPM-UC3M**. A database recorded at the Multimedia Processing Group of the University Carlos III of Madrid using a PDA. It consists of 30 speakers with 50 isolated words for each one of them. Every single word is considered an individual sequence and its mean length is around 1.3 seconds.

The audio files were processed using the freely available HTK software,[3] a standard parametrization consisting of 12 Mel-frequency cepstral coefficients (MFCCs), an energy term, and their respective increments (δ), giving a total of 26 parameters. These parameters were obtained every 10 ms with a 25-ms analysis window. The resulting 26-dimensional sequences were fed into the different clustering algorithms without any further processing.

The other scenario used for testing purposes is clustering of EEG signals. We employ the database recorded by Zak Keirn at Purdue University.[4] This database consists of EEG recordings of seven subjects performing five different mental tasks, namely, baseline (rest), math calculations, composing a letter, rotating a geometrical figure, and counting. Each recording comprises measures taken from seven channels at 250 Hz for 10 seconds. We divide them into sequences of 125 samples, and the only preprocessing applied to them is a first-order derivative so they adjust better to a Markov model. Given a subject, the purpose is to find clusters of sequences representing the same activity. Concretely, we perform seven clusterings (one for each subject) of 50 sequences (10 per mental activity, randomly chosen) into five groups.

Since real-world data are inherently noisy and the sequences do not perfectly fit a Markovian generative model, the property of embedding information about the entire set of sequences in each pairwise distance can become performance-degrading. Thus, it becomes interesting to select only an adequate subset of the models for obtaining the distance matrix. This way, we will be performing the clustering in a reduced subspace spanned just by the chosen models.

For this purpose, we propose a simple method to determine which models to include in the KL+SC method: First, since models coming from lengthier sequences are expected to be less influenced by outliers and to provide more information about the underlying processes, the general heuristic is to keep these models. The remaining question is, how many models should be considered? To answer this, the percentage of modeled sequences is swept and at each step a heuristic h is obtained as

$$h = \frac{\lambda_{K+1}}{\lambda_N}, \tag{9}$$

where λ_j is the jth eigenvalue of the SC GEV problem, sorted by increasing magnitude. Intuitively, this value can be seen as representing a normalized measure of the energy preserved by taking only K eigenvalues, and can therefore be regarded as a measure of the clustering quality. It is then natural to select as the appropriate percentage of sequences to model the one that maximizes h. Similar approaches can be found in the PCA literature for selecting the optimum number of principal components to retain [21]. As previously stated, this is a simple method with no aspirations of being optimum but developed just for illustrating that an adequate selection of models can be advantageous, or even necessary, for attaining good clustering results. It is also worth noting that, using this method, the model selection is carried out based on a likelihood matrix obtained using all the sequences in the data set. We refer to the KL+SC method coupled with this model selection scheme as KL+SC+MS.

Table 1 shows the results (averaged over 15 iterations) of the compared methods in the different data sets using SC. HC results are not shown because of space restrictions, but they were clearly inferior to those attained via SC. Agglomerative methods fail in these scenarios because the relationships among the data that must be exploited in order to obtain an adequate partition are impossible to capture in a pairwise fashion. Results are given under varying number of hidden states in the range where the different methods perform best for each data set.

All in all, the KL+SC+MS combination noticeably outperforms the alternatives, specially in the speech data sets. The use of KL+SC without model selection does not work as well as in the synthetic experiments because sequences belonging to the same cluster are not actually drawn from a unique HMM. The clustering just relies on the assumption that these sequences lead to similar models. In this scenario, there are no such things as "true" HMMs generating the data set, so the interpretation of the likelihood matrix that gives birth to our proposal loses much of its strength. However, it can be

[2] http://atvs.ii.uam.es/databases.jsp.
[3] http://htk.eng.cam.ac.uk.
[4] http://www.cs.colostate.edu/eeg/eegSoftware.html.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.

TABLE 1
Mean and Standard Deviation of Clustering Accuracy (Percent) on the Real Data Sets Using HMMs with Different Number of Hidden States and SC

Dataset	# of hidden states	SYM	BP	YY	POR	KL	KL+MS
AHUMADA	m=2	79.08 (±0.41)	**82.72** (±0.42)	78.15 (±0.33)	53.90 (±4.01)	77.65 (±0.45)	**82.36** (±0.76)
	m=3	78.83 (±0.35)	75.88 (±0.51)	77.74 (±0.44)	53.88 (±3.99)	77.61 (±0.42)	**82.01** (±0.48)
	m=4	76.80 (±0.74)	71.42 (±1.01)	75.75 (±0.66)	50.00 (±0.01)	75.85 (±0.63)	**81.02** (±0.52)
GPM-UC3M	m=2	84.54 (±3.67)	**87.38** (±3.97)	84.98 (±3.95)	50.09 (±0.19)	84.28 (±4.30)	**90.18** (±3.45)
	m=3	73.05 (±4.34)	75.93 (±5.25)	74.89 (±4.83)	50.18 (±0.29)	76.13 (±4.52)	**90.35** (±2.93)
	m=4	61.68 (±2.73)	61.82 (±3.44)	64.88 (±4.33)	50.26 (±0.26)	70.25 (±3.76)	**89.96** (±2.50)
EEG	m=5	39.43 (±0.49)	62.99 (±0.88)	**72.53** (±0.60)	48.64 (±0.70)	66.53 (±0.82)	70.19 (±0.85)
	m=6	39.44 (±0.40)	65.92 (±0.90)	69.94 (±0.76)	47.25 (±0.82)	66.27 (±0.85)	**71.14** (±0.89)
	m=7	39.41 (±0.44)	68.38 (±0.94)	71.50 (±0.87)	45.62 (±0.71)	70.47 (±0.84)	**73.21** (±0.76)

seen that if the KL+SC combination is coupled with the proposed model selection method, it produces convincing results even in such adverse conditions.

A remarkable fact is that the previously proposed distances suffer from a severe performance loss in the speaker clustering tasks as the number of hidden states increases. This is caused by the models overfitting the sequences in these data sets because of the high dimensionality of the data and the short mean length of the sequences. The evaluation of likelihoods under these models produces results that does not reflect the underlying structure of the data. This distortion severely undermines the performance of previously proposed distances, yielding poorly structured distance matrices that seriously hinders the SC. The use of our proposed KL distance, specially in combination with model selection, has a smoothing effect on the distance matrices. This effect makes it less sensitive to overfit models, resulting in an improved performance relative to the other distances as overfitting becomes more noticeable. This robustness is a very useful property of our proposal since, in practice, it is usually hard to determine the optimum model structure and overfitting is likely to occur. It is also worth noting that the advantage of using our method is clearer in the GPM-UC3M data set because the number of sequences considered in each clustering task is larger. This agrees with the conclusions drawn from the experiments with synthetic data.

The dimensionality of the data in the EEG data set is lower than those in the speaker verification ones. This allows for an increase in the number of hidden states without suffering from overfitting. The KL+SC+MS method also performs best in this data set, followed closely by the YY distance. It is remarkable that the improvement in performance due to the use of model selection is less dramatic in this scenario because of both the absence of overfitting and the equal length of all sequences.

In Table 2, we show the number of models chosen for consideration by the model selection algorithm in each of the clustering tasks. Notice how in the three cases as the complexity of the models (in terms of number of hidden states) increases, the model selection scheme picks a larger number of them. In general, more complex models lead to more varying probabilities when evaluated on the different sequences. This way, the effective dimension of the model-induced subspace where the sequences lie grows with the complexity of the models, which agrees with the aforementioned behavior of the model selection scheme.

TABLE 2
Optimal Number of Models Chosen by the Model Selection Algorithm

Dataset	m=2	m=3	m=4
AHUMADA	66.63%	72.28%	75.3%
GPM-UC3M	43.1%	58.61%	63.20%
Dataset	m=5	m=6	m=7
EEG	57.14%	60.12%	64.46%

5 Conclusions and Future Work

We have proposed a new distance measure for sequential data clustering based on the KL divergence. It embeds information of the whole data set into each element of the distance matrix, introducing a structure that makes it specially suitable for its use in combination with SC techniques. This measure also allows for the use of a reduced representative subset of models, which, if chosen properly, can give an increase in performance in real-world scenarios potentially containing outliers and misleading data.

A model selection scheme for this task has also been presented in the paper with very encouraging results, especially in the presence of overfitting. This method works from a likelihood matrix **L** constructed using all the sequences in the data set. If the selection could be done directly on the sequences, reduced computational cost would be achieved, by not having to fit a model for each sequence in the data set.

The reported results have been obtained using HMMs as generative models for the individual sequences, although the method is independent of this selection. In fact, exploring more expressive models is a straightforward and promising future line of research in order to successfully apply this clustering technique to a wider range of problems, such as video event detection, text mining, etc.

Acknowledgments

This research was conducted with support from Spanish Government CICYT grants TEC2008-02473 and TEC2008-06382. The authors would like to thank the anonymous reviewers and the associate editor for their valuable comments and suggestions that helped to improve the manuscript.

References

[1] L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE,* vol. 77, no. 2, pp. 257-286, Feb. 1989.

[2] S. Fine, Y. Singer, and N. Tishby, "The Hierarchical Hidden Markov Model: Analysis and Applications," *Machine Learning,* vol. 32, no. 1, pp. 41-62, 1998.

[3] J. Bilmes, "Buried Markov Models for Automatic Speech Recognition," *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing,* Mar. 1999.

[4] Z. Wu and R. Leahy, "An Optimal Graph-Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 11, pp. 1101-1113, Nov. 1993.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.

[5] R. Xu and D.W. Wunsch II, "Survey of Clustering Algorithms," *IEEE Trans. Neural Networks,* vol. 16, no. 3, pp. 645-678, May 2005.

[6] P. Smyth, "Clustering Sequences with Hidden Markov Models," *Advances in Neural Information Processing Systems,* vol. 9, pp. 648-654, 1997.

[7] A. Panuccio, M. Bicego, and V. Murino, "A Hidden Markov Model-Based Approach to Sequential Data Clustering," *Proc. Joint IAPR Int'l Workshop Structural, Syntactic and Statistical Pattern Recognition,* pp. 734-742, 2002.

[8] F. Porikli, "Clustering Variable Length Sequences by Eigenvector Decomposition Using HMM," *Proc. Int'l Workshop Structural and Syntactic Pattern Recognition,* pp. 352-360, 2004.

[9] J. Yin and Q. Yang, "Integrating Hidden Markov Models and Spectral Analysis for Sensory Time Series Clustering," *Proc. Fifth IEEE Int'l Conf. Data Mining,* Nov. 2005.

[10] P. Baldi, S. Brunak, and G. Stolovitzky, *Bioinformatics: The Machine Learning Approach.* MIT Press, 1998.

[11] G. Jin, L. Tao, and G. Xu, "Hidden Markov Model Based Events Detection in Soccer Video," *Proc. Int'l Conf. Image Analysis and Recognition,* pp. 605-612, 2004.

[12] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Soc.,* vol. 39, no. 1, pp. 1-38, 1977.

[13] K. Murphy, "Dynamic Bayesian Networks: Representation, Inference and Learning," PhD dissertation, Computer Science Division, Univ. of California Berkeley, July 2002.

[14] A. Ng, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," *Advances in Neural Information Processing Systems,* 2002.

[15] B. Juang and L. Rabiner, "A Probabilistic Distance Measure for Hidden Markov Models," *AT&T Technical J.,* vol. 64, no. 2, pp. 391-408, Feb. 1985.

[16] T. Oates, L. Firoiu, and P.R. Cohen, "Using Dynamic Time Warping to Bootstrap HMM-Based Clustering of Time Series," *Sequence Learning—Paradigms, Algorithms, and Applications,* Springer-Verlag, pp. 35-52, 2001.

[17] T. Jebara, Y. Song, and K. Thadani, "Spectral Clustering and Embedding with Hidden Markov Models," *Proc. 18th European Conf. Machine Learning,* Sept. 2007.

[18] S. Kullback and R. Leibler, "On Information and Sufficiency," *Annals of Math. Statistics,* vol. 22, pp. 79-86, 1951.

[19] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis.* Cambridge Univ. Press, 2004.

[20] J. Ortega-Garcia, J. Gonzalez-Rodriguez, and V. Marrero-Aguiar, "Ahumada: A Large Speech Corpus in Spanish for Speaker Characterization and Identification," *Speech Comm.,* vol. 31, pp. 255-264, 2000.

[21] I. Jolliffe, *Principal Component Analysis,* second ed. Springer, 2002.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: Purdue University. Downloaded on September 11, 2009 at 16:02 from IEEE Xplore. Restrictions apply.