Statistics and Probability Letters 167 (2020) 108917

Contents lists available at ScienceDirect

Statistics and Probability Letters

journal homepage: www.elsevier.com/locate/stapro

Hidden Markov Models for multivariate functional data

Andrea Martino *, Giuseppina Guatteri, Anna Maria Paganoni

Department of Mathematics, Politecnico di Milano, Via Edoardo Bonardi, 13, 20133 Milan, Italy

ARTICLE INFO

Article history:
Received 4 July 2019
Received in revised form 29 July 2020
Accepted 16 August 2020
Available online 24 August 2020

Keywords:
Clustering
Functional data
Statistical modeling

ABSTRACT

In this paper we extend the usual Hidden Markov Models framework, where the observed objects are univariate or multivariate data, to the case of functional data, by modeling the temporal structure of a system of multivariate curves evolving in time.

1. Introduction

Hidden Markov Models (HMMs) represent a well-known method for the study of time series involving sequences of data, widely used in many fields like biostatistics (Martino et al., 2020), bioinformatics (Durbin et al., 1998) and finance (Paas et al., 2007). In the literature of HMMs, there are several examples where the outcome consists of univariate or multivariate data, with both discrete and continuous observations; in particular, in Cappé et al. (2006) a very general definition of such processes is provided. In this paper, we want to extend the usual HMM algorithms from the finite dimensional framework to the infinite dimensional one. Therefore, we focus on the functional setting, where each observed data is considered as a multivariate random curve, that can be also seen as the realization of a stochastic process taking values in $\mathbb{R}^n$, $n \geq 1$. Thanks to their versatility, these models can be applied in many research contexts, since lately more and more data are collected as suitable curves observed on a continuum domain. Let us consider, for instance, a collection of biomedical signals, as ECGs and EEGs, measured at different time points for several patients; they can be seen as a collection of multivariate curves evolving in time. With the application of our model to this type of data, we can retrieve some information about the evolution of the shape of the curves, that can lead to determine, for example, the onset of a certain pathology or the rate of transitions among healthy and non-healthy states.

The natural context to develop the statistical models and tools to describe this kind of data is the *Functional Data Analysis* (FDA) (see, e.g. Ramsay, 2004; Ramsay and Silverman, 2007; Ferraty and Vieu, 2006; Horváth and Kokoszka, 2012). Working with functional data can be a difficult task because of the dimensionality of the spaces of the data; moreover, the usual HMM requires the definition of a probability density that generates the observations, which may be lacking for functional random processes. Therefore, since we want to consider the most general case without making any assumptions on the law of the process that generated the data, we construct a similarity function built on distances between curves to evaluate the emission of an observation by a certain state. We consider a hidden Markov chain evolving in time where each state emits a multivariate random curve and we solve two problems. First, we estimate the parameters of the underlying Markov process to understand the time series system that generated the data; then we solve a clustering problem by finding the best sequence of states that generated the data in order to classify the curves in clusters.

* Corresponding author.
E-mail address: andrea.martino@polimi.it (A. Martino).

https://doi.org/10.1016/j.spl.2020.108917
0167-7152/

The paper is organized as follows: in Section 2 we present the model, adding some information about the theory of HMMs and functional data. In Section 3 we present a simulation study to assess the performance of the model while in Section 4 we see a case study application to a dataset regarding a climate problem. Finally, in Section 5 we give some discussion and conclusions. All the analysis have been carried out using the statistical software R (R Core Team, 2017) and a package is in development. The codes are available upon request.

2. The model

The aim of this paper is to consider and study a proper Hidden Markov Model (HMM) in the multivariate functional framework. Let us consider a multivariate random curve $\mathbf{X} = \{\mathbf{X}(t)\}_{t\in I} = \{X_1(t), \ldots, X_J(t)\}_{t\in I}$, with $J \geq 1$ and I compact interval of $\mathbb{R}$, as defined in Horváth and Kokoszka (2012). Starting from the definition in Cappé et al. (2006), we define a HMM in the multivariate functional context as a bivariate process $\{(Q_k, \{\mathbf{X}_k(t)\}_{t\in I})\}_{k\geq 1}$ on a given probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where

- $\{Q_k\}_{k\geq 1}$ is a Markov chain with a discrete and finite state space $S = \{s_1, \ldots, s_N\}$, with $N \geq 1$, transition matrix $A = \{a_{ij}\} = \mathbb{P}(Q_k = s_j | Q_{k-1} = s_i)$ and initial distribution $\boldsymbol{\nu}$, where $\nu_i = \mathbb{P}(Q_1 = s_i)$;
- $\{\mathbf{X}_k(t)\}_{t\in I}$ is a multivariate random curve, i.e. a stochastic process taking values in $\mathbb{R}^J$. Given the process $\{Q_k\}_{k\geq 1}$, $\{\{\mathbf{X}_k(t)\}_{t\in I}\}_{k\geq 1}$ is a sequence of conditionally independent multivariate functions and $\{\mathbf{X}_k(t)\}_{t\in I}$ only depends on Q_k for each k. Differently from the literature, in this case the emission functions are not probability density functions. Let us consider a functional observation $\mathbf{x}_k$ and denote the emission function of $\mathbf{x}_k$ conditionally on the event $\{Q_k = s_i\}$ with $b_{\mathbf{X}_k|Q_k=s_i}(\mathbf{x}_k ; \boldsymbol{\mu}_i)$, for any $i = 1, \ldots, N$, where $\boldsymbol{\mu}_i$ is a functional parameter representing the mean of the curves emitted by state s_i. We construct the emission functions $b_{\mathbf{X}_k|Q_k=s_i}(\mathbf{x}_k ; \boldsymbol{\mu}_i)$, $i = 1, \ldots, N$, based on distances between curves. Specifically, we assume that, for each state s_i, the emission function can be written as

$$b_{\mathbf{X}_k|Q_k=s_i}(\mathbf{x}_k ; \boldsymbol{\mu}_i) = h\big(d(\mathbf{x}_k , \boldsymbol{\mu}_i)\big), \qquad i = 1, \ldots, N \tag{1}$$

  where $h : \mathbb{R} \to \mathbb{R}$ is a function that transforms the distance into a similarity measure. In particular, in this work we will use the function $h(y) = 1/y^2$ and the L^2 distance. For a further analysis on the choice of the function $h(y)$, along with some simulations, see the Supplementary Material. An expression function like (1) can be interpreted as a similarity function that behaves like a likelihood; in particular, when the observation $\mathbf{x}_k$ and the functional parameter $\boldsymbol{\mu}_i$ are close to each other, with respect to the L^2 distance, the distance is small so the value of the emission function is high, which suggests that the observation $\mathbf{x}_k$ is very similar to the curves emitted by state s_i and represented by $\boldsymbol{\mu}_i$. Instead, when the observation $\mathbf{x}_k$ and the functional parameter $\boldsymbol{\mu}_i$ are far from each other with respect to the L^2 distance, the distance is high, so the value of the emission function is low, which suggests the observation $\mathbf{x}_k$ is not very similar to the curves emitted by state s_i and represented by $\boldsymbol{\mu}_i$. Finally, we can completely define our HMM with the set of parameters $\lambda = (\boldsymbol{\nu}, A, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N)$.

An important step of our algorithm is the initialization. Since we want our algorithm to be as robust as possible, we perform a functional k-means algorithm on the dataset of curves to find the initial centroids, see Tarpey and Kinateder (2003) for further details. After a random selection of a set of N fixed initial centroids $\{\boldsymbol{\mu}_1^{(0)}(t), \ldots, \boldsymbol{\mu}_N^{(0)}(t)\}$ the algorithm iteratively repeats the two steps described before. Formally, at the mth iteration of the initialization step, $m \geq 1$, in the algorithm:

1. each curve $\mathbf{x}_k$, $k = 1, \ldots, K$, is assigned to the cluster whose centroid minimizes the L^2 distance. The mth cluster assignment $C_k^{(m)}$ for the kth statistical unit is

$$C_k^{(m)} := \underset{i=1,\ldots,N}{\operatorname{argmin}}\, d_{L^2}(\mathbf{x}_k, \boldsymbol{\mu}_i^{(m-1)});$$

2. the centroids for the clusters are computed as

$$\boldsymbol{\mu}_i^{(m)} := \underset{\boldsymbol{\mu}\in L^2(\Omega\times I;\, \mathbb{R}^J)}{\operatorname{argmin}} \sum_{k:C_k^{(m)}=i} d_{L^2}(\mathbf{x}_k, \boldsymbol{\mu})^2, \qquad i = 1, \ldots, N.$$

   In our case, we can rewrite the equation for any state s_i, $i = 1, \ldots, N$, as:

$$\mu_{ij}^{(m)} := \frac{1}{K_i} \sum_{k:C_k^{(m)}=i} x_{jk}, \qquad j = 1, \ldots, J.$$

   where K_i is the number of curves assigned to the ith cluster in the step m. After obtaining the same cluster assignments at two subsequent iterations, the initialization step ends and we obtain the preliminary estimates of the functional parameters $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N\}$ for the states of the HMM.

Let us denote by $\mathbf{x}$ an output sequence of observation functions of the HMM and with $\mathcal{L}(\lambda|\mathbf{x})$ the objective function of all the parameters of the model given $\mathbf{x}$. In the literature of HMMs, there are usually three problems to tackle (see, e.g., Zucchini et al., 2016; Rabiner, 1989): find $\mathcal{L}(\lambda|\mathbf{x})$ for the realization $\mathbf{x}$, find the set of parameters λ^* that maximizes $\mathcal{L}(\lambda|\mathbf{x})$ and find the best state sequence $Q = (Q_1, \ldots, Q_K)$ that explains $\mathbf{x}$, given $\mathbf{x}$ and λ.

The sequence of states of a HMM is not observed, so the usual approach, see Dempster et al. (1977), consists in treating the states as missing data and apply an EM algorithm combined with the forward–backward procedure, to find the objective function and then deduce the best estimates of the parameters. This algorithm is known in the literature as Baum–Welch algorithm (see for instance Baum et al., 1970; Welch, 2003; Bilmes, 1998). To fully describe our algorithms, we need to introduce two further quantities. We define $\xi_k(i, j)$, the probability of being in state s_i at time k, and state s_j at time $k + 1$, given the model and the observations, i.e.

$$\xi_k(i,j) \;=\; \mathbb{P}(Q_k = s_i, Q_{k+1} = s_j \mid \mathbf{X}_1 = \mathbf{x}_1, \ldots, \mathbf{X}_k = \mathbf{x}_k, \lambda)$$

and the probability of being in the state s_i at time k, given the observations and the model as

$$\gamma_k(i) \;=\; \mathbb{P}(Q_k = s_i \mid \mathbf{X}_1 = \mathbf{x}_1, \ldots, \mathbf{X}_k = \mathbf{x}_k, \lambda) \;=\; \sum_{j=1}^{N} \xi_k(i,j).$$

As usually done in the literature, we use the forward–backward procedure, which is described in the Supplementary Material. After computing the objective function with the forward–backward algorithm, we apply an EM-type algorithm to compute all the parameters of the HMM, maximizing the objective function. Since the maximization of the initial and the transition probabilities is performed using the estimators already known in literature (see, e.g., Rabiner, 1989; Zucchini et al., 2016), in this work we mainly focus on the maximization of the state dependent parameters, which depend on the emission functions. Even though until now our formulas only consider a single observation sequence, they can be extended to the more general case of multiple observations. Let us denote the set of L observation sequences as $\mathcal{X} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(L)})$ where $\mathbf{x}^{(l)} = (\mathbf{x}_1^{(l)} \mathbf{x}_2^{(l)} \cdots \mathbf{x}_{K_l}^{(l)})$ is the lth observation sequence of length K_l. We assume all the sequences to be independent from each other; our goal is to adjust the parameters of the model λ to maximize the function $\mathcal{L}(\lambda|\mathcal{X}) = \prod_{l=1}^{L} \mathcal{L}(\lambda|\mathbf{x}^{(l)})$. Moreover, since in our case we only consider sequences of equal length between the statistical units, we can assume $K_l = K$ for every $l = 1, \ldots, L$. The term we want to maximize, in the setting of multiple sequences, becomes, in principle:

$$\text{term 3} = \sum_{k=1}^{L\cdot K} \sum_{i=1}^{N} \gamma_k(i) \log\, b_{\mathbf{X}_k|Q_k=s_i}(\,\mathbf{x}_k\,;\,\boldsymbol{\mu}_i). \tag{2}$$

We stress that this is the first attempt of using HMMs to deal with infinite dimensional objects and the solution seems not trivial. In particular, differently from the finite case, because of the choice we made in (1), the emission functions are not probability density functions, so we do not proceed as in the classical case but we extend all the estimators commonly used in the functional data framework into the theory of functional HMM. We thus introduce the most natural estimator for the means:

$$\widehat{\boldsymbol{\mu}}_i = \frac{\sum_{k=1}^{L\cdot K} \gamma_k(i) \boldsymbol{X}_k}{\sum_{k=1}^{L\cdot K} \gamma_k(i)}, \qquad i = 1, \ldots, N. \tag{3}$$

Notice that it corresponds to the argmax of the following term

$$\text{term 3} = \sum_{k=1}^{L\cdot K} \sum_{i=1}^{N} \gamma_k(i)\, b_{\mathbf{X}_k|Q_k=s_i}(\,\mathbf{x}_k\,;\,\boldsymbol{\mu}_i). \tag{4}$$

Let now assume for simplicity set $J = 1$, so that $\boldsymbol{\mu_i}$ becomes a scalar function and each $X_k(t)$ are real processes for each k. Then, since our data are curves defined on a finite set of points, we can see them as realizations of a sequence of random vectors $X_k(t_1), \ldots, X_k(t_h)$ (where h is the length of the grid of points) such that, given the state $Q_k = j$, have mean $\boldsymbol{\mu}_j^h = (\boldsymbol{\mu_j}(t_1), \ldots, \boldsymbol{\mu_j}(t_h)) \in \mathbb{R}^h$ and a certain variance $\Sigma_j^h \in M(h \times h)$. Now assume for a moment that our emission variables are gaussian processes in I , then our estimator (3) coincides with the one proposed in Rabiner (1989), derived from the EM algorithm and we know from the literature that this estimator inherits the properties of an MLE estimator, see Baum et al. (1970) and Dempster et al. (1977) and in particular it is consistent, see Kamgaing (2013) for further details. So the idea is to bypass the problem that the emission function is not a "real" probability density function as follows: we estimate the parameters involved using the finite dimensional conditional laws of the stochastic processes $(X_k(\cdot))$; thus, we use the typical estimation of the mean that in several situations, as the fundamental Gaussian case, has good asymptotical properties. In the next section, we show empirically that this estimate gives good results.

Finally, the third problem regarding HMM, as we stated before, consists in finding the best state sequence $Q = (Q_1, \ldots, Q_K)$ that explains a certain observation. To solve this problem we use the Viterbi algorithm, see Viterbi (1967). We refer to the Supplementary Materials for the description of the algorithm.

Table 1
AMSE (SD) of the HMM parameters along 100 replications of the Baum–Welch algorithm with $N = 3$ states for the HMM.

Parameter	AMSE (SD)
a_{11}	$4.49 \cdot 10^{-3}$ $(4.01 \cdot 10^{-3})$
a_{12}	$6.70 \cdot 10^{-3}$ $(6.80 \cdot 10^{-3})$
a_{13}	$1.20 \cdot 10^{-3}$ $(1.60 \cdot 10^{-3})$
a_{21}	$3.47 \cdot 10^{-3}$ $(2.69 \cdot 10^{-3})$
a_{22}	$2.30 \cdot 10^{-2}$ $(4.63 \cdot 10^{-2})$
a_{23}	$3.68 \cdot 10^{-2}$ $(7.46 \cdot 10^{-2})$
a_{31}	$1.78 \cdot 10^{-5}$ $(9.29 \cdot 10^{-5})$
a_{32}	$4.06 \cdot 10^{-2}$ $(3.54 \cdot 10^{-2})$
a_{33}	$4.23 \cdot 10^{-2}$ $(9.31 \cdot 10^{-2})$

Table 2
CCR and AMSE (SD) between the real and estimated means along 100 replications of the Viterbi and k-means algorithm.

	Viterbi Algorithm	k-means algorithm
CCR	0.849	0.591
AMSE($\boldsymbol{\mu}_1; \widehat{\boldsymbol{\mu}}_1$) (SD)	0.079 (0.023)	0.131 (0.286)
AMSE($\boldsymbol{\mu}_2; \widehat{\boldsymbol{\mu}}_2$) (SD)	0.697 (0.097)	0.890 (0.279)
AMSE($\boldsymbol{\mu}_3; \widehat{\boldsymbol{\mu}}_3$) (SD)	0.944 (0.031)	1.051 (0.059)

3. Simulation studies

We generate three samples of length $n = 2000$ of realizations on a grid of 100 points for three independent bivariate random curves $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ in $L^2(\Omega \times I;\ \mathbb{R}^J)$, with $J = 2$. Each sample is emitted from a different state of a 3-state HMM having the following parameters:

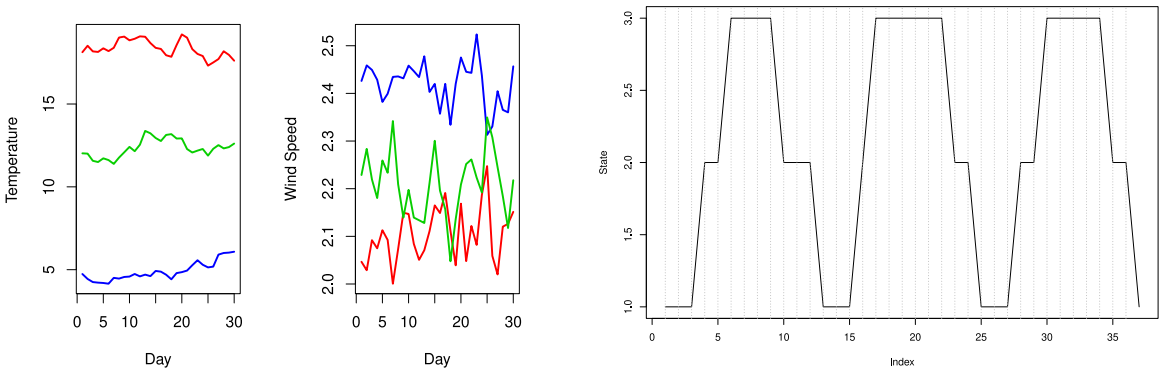
- **State 1**: $\nu_1 = 1, a_{11} = 0.6, a_{12} = 0.3, a_{13} = 0.1, \boldsymbol{\mu}_1(t) = \begin{pmatrix} t(1-t) \\ 2t \end{pmatrix}$;
- **State 2**: $\nu_2 = 0, a_{21} = 0.1, a_{22} = 0.8, a_{23} = 0.1, \boldsymbol{\mu}_2(t) = \begin{pmatrix} t^2(1-t) \\ t^2 \end{pmatrix}$;
- **State 3**: $\nu_3 = 0, a_{31} = 0, a_{32} = 0, a_{33} = 1, \boldsymbol{\mu}_3(t) = \begin{pmatrix} t(1-t)^2 \\ \frac{1}{2}t^3 \end{pmatrix}$.

where $\boldsymbol{\nu} = (\nu_i)$ is the vector of the initial probabilities of the state, $A = (a_{ij})$ is the transition matrix and $\boldsymbol{\mu}_i(t), i = 1, \ldots, N$, represent the real means of each sample. For each state, the sample is generated from a Gaussian process using the same exponential covariance kernel $C(s, t) = ae^{-b|s-t|}$, $a = 0.1$, $b = 0.3$, using the `R` package `roahd` (Ieva et al., 2019). The first problem consists in the choice of the number of states, since it is a priori unknown. We begin by running our algorithm for $N = 2, \ldots, 5$ number of states and by computing each time the AIC and BIC criteria to choose the optimal number of states that models our data. In particular, we compute the values as follows:

$$\text{AIC} = -2\log(\mathcal{L}(\lambda|\mathbf{x})) + 2p \qquad \text{BIC} = -2\log(\mathcal{L}(\lambda|\mathbf{x})) + p\log(n) \tag{5}$$

where p is the number of parameters of the HMM. For both criteria, the model exhibiting the lowest value is usually preferred which, in this case, is reached for $N = 3$. From now on we consider, for this simulation, $N = 3$ as the "optimal" number of states for the HMM. After choosing the number of states, we summarize our results along 100 repetitions of our algorithm to estimate the parameters of the HMM. To have a better understanding of the results, we compute the average mean square error (AMSE) and the standard deviation (SD) of the estimates, along 100 repetitions of the algorithm, and we show the obtained results in Table 1. As we can see, all the parameters are very well estimated, both in terms of mean and standard deviation of the parameters.

Moreover, we can obtain some further information about the clustering structure of our data. Specifically, we use our model and apply the Viterbi algorithm on the output obtained from the Baum–Welch algorithm, to estimate the best state sequence and compare it with the output of the k-means algorithm, based on the same distance. In particular, in Table 2 we compare the Correct Classification Rate (CCR) of the number of curves obtained by applying both methods along with the MSEs of the estimates of functional parameters of a state $\widehat{\boldsymbol{\mu}}_i$, $i = 1, 2, 3$, i.e. the values of the distances between the real and the estimated means, computed over 100 replications of the algorithm. By comparing the results, we see obvious advantages of our method, since the CCR is higher and all the AMSEs and standard deviations are smaller. We conclude that, not only our method is able to detect the temporal structure behind the sequences of functional data and estimate all the parameters of the underlying hidden states but, applying the Viterbi algorithm, we can also cluster the curves with good values of accuracy.



(a) Plots of the functional parameters.

(b) Output of the Viterbi algorithm

Fig. 1. Results of the Baum–Welch and Viterbi algorithms for the case study.

4. Case study: Weather data

In this last part, we apply the described model to a real dataset regarding the weather in Basel, Switzerland, extracted from the website www.meteoblue.com. In particular, our data consists of daily registrations of temperature and wind speed from 2008 to 2018. We consider each month as an observation of a statistical unit, in order to have 12 multivariate functional observations for every year. First, we apply our algorithm to the weather data with $N = 2, \dots, 6$ states and compute every time the AIC and BIC of the model, as in (5). We assume $N = 3$ to represent the optimal number of states, since both criteria exhibit the lowest value.

After choosing the number of states, we continue our analysis by applying the Baum–Welch and Viterbi algorithms. In particular, from the parameter estimation algorithm we obtain the following results:

$$\nu = (1, 0, 0) \qquad A = \begin{pmatrix} 0.708 & 0.292 & 0.000 \\ 0.079 & 0.498 & 0.423 \\ 0.000 & 0.181 & 0.819 \end{pmatrix},$$

i.e., the initial probabilities vector, the transition matrix and the functional parameters of the states for temperature and wind speed, which can be seen in Fig. 1(a). The three states are denoted by the Blue, Green and Red color, respectively. Since each statistical unit starts being observed during January, the vector ν of the initial probabilities only takes probability 1 on the first state, which is the state with the lowest temperature and highest wind speed, representing the colder months. Moreover, the transition matrix shows how states 1 and 3 are the ones with higher probabilities for the model to remain in the state while state 2, representing the "middle seasons", is the most unstable state and can be basically considered as a transition state between the other two.

After running the Viterbi algorithm, we obtain a set of labels for the observations and we show a subset for 3 years in Fig. 1(b). From the plot, we clearly notice the seasonality trend; moreover, it is clear how the results we obtained from the transition matrix are strengthened. The states corresponding to Summer and Winter are the longest and most stable, since the corresponding states are visited for 40 and 35 months over a total of 132, respectively, while the mid-seasons can be grouped together and are usually the shortest, for the remaining 57 months.

5. Discussion

In this work, we faced the problem of estimating the parameters of a HMM where the output is a multivariate random curve, showing that it is able to detect the underlying structure of the time series system and provide robust results. Then, using the Viterbi algorithm, we also solved a classification problem, noticing that we obtain better results by looking at the time order of the system than by only considering the shape of the curves.

CRediT authorship contribution statement

Andrea Martino: Writing - original draft, Writing - review & editing, Conceptualization, Methodology, Software, Validation, Formal analysis, Data curation, Visualization, Investigation. **Giuseppina Guatteri:** Writing - review & editing, Conceptualization, Methodology, Supervision, Formal analysis, Project administration. **Anna Maria Paganoni:** Writing - review & editing, Conceptualization, Methodology, Supervision, Project administration.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.spl.2020.108917.

References

Baum, L.E., Petrie, T., Soules, G., Weiss, N., 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann. Math. Stat. 41 (1), 164–171.
Bilmes, J.A., 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.
Cappé, O., Moulines, E., Rydén, T., 2006. Inference in Hidden Markov Models. Springer Science & Business Media.
Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B Methodol. 1–38.
Durbin, R., Eddy, S.R., Krogh, A., Mitchison, G., 1998. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, http://dx.doi.org/10.1017/CBO9780511790492.
Ferraty, F., Vieu, P., 2006. Nonparametric Functional Data Analysis: Theory and Practice. Springer Science & Business Media.
Horváth, L., Kokoszka, P., 2012. Inference for Functional Data with Applications, Vol. 200. Springer Science & Business Media.
Ieva, F., Paganoni, A.M., Romo, J., Tarabelloni, N., 2019. Roahd package: robust analysis of high dimensional data. The R Journal 11 (2), 291–307, https://doi.org/10.32614/RJ-2019-032.
Kamgaing, J.T., 2013. Maximum Likelihood Estimators for Multivariate Hidden Markov Mixture Models. Technical Report, Report in Wirtschaftsmathematik 146, Department of Mathematics, University of Kaiserslautern.
Martino, A., Guatteri, G., Paganoni, A.M., 2020. Multivariate Hidden Markov models for disease progression. Stat. Anal. Data Min. ASA Data Sci. J. 1–9, https://doi.org/10.1002/sam.11479.
Paas, L.J., Vermunt, J.K., Bijmolt, T.H., 2007. Discrete time, discrete state latent Markov modelling for assessing and predicting household acquisitions of financial products. J. R. Stat. Soc. Ser. A Stat. Soc. 170 (4), 955–974.
R Core Team, 2017. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, https://www.R-project.org/.
Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE 77 (2), 257–286.
Ramsay, J.O., 2004. Functional data analysis. In: Encyclopedia of Statistical Sciences, Vol. 4. Wiley Online Library.
Ramsay, J.O., Silverman, B.W., 2007. Applied Functional Data Analysis: Methods and Case Studies. Springer.
Tarpey, T., Kinateder, K.K., 2003. Clustering functional data. J. Classif. 20 (1), 093–114.
Viterbi, A., 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inf. Theory 13 (2), 260–269.
Welch, L.R., 2003. Hidden Markov models and the Baum–Welch algorithm. IEEE Inf. Theory Soc. Newslett. 53 (4), 10–13.
Zucchini, W., MacDonald, I.L., Langrock, R., 2016. Hidden Markov Models for Time Series: an Introduction Using R. Chapman and Hall/CRC.