ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

Clustering Functional Data with the SOM algorithm

Fabrice Rossi, Brieuc Conan-Guez and Aïcha El Golli

Projet AxIS, INRIA, Domaine de Voluceau, Rocquencourt, B.P. 105
78153 Le Chesnay Cedex, France
CEREMADE, UMR CNRS 7534, Université Paris-IX Dauphine,
Place du Maréchal de Lattre de Tassigny, 75016 Paris, France

Abstract. In many situations, high dimensional data can be considered as sampled functions. We show in this paper how to implement a Self-Organizing Map (SOM) on such data by approximating a theoretical SOM on functions thanks to basis expansion. We illustrate the proposed method on real world spectrometric data for which functional preprocessing is very successful.

1 Introduction

Many real-world applications produce high dimensional data that are quite difficult to handle with traditional methods. A solution to overcome this type of problems is to identify internal structure in the data and to use the corresponding prior knowledge to simplify data analysis. A very general internal structure model can be obtained by assuming that a high dimensional vector is in fact a discretized function. This model covers for instance time series (which are mappings between a date and a value), weather data (which are time-varying geographical mappings), spectrometric data (a spectrum is a mapping between wavelengths and "answers" from the observed object), etc. Functional Data Analysis (FDA, see [10]) is an extension of traditional data analysis methods to this kind of functional data. In FDA, each individual is characterized by one or more real valued functions, rather than by a vector of $\mathbb{R}^p$.

FDA methodology has numerous advantages over a basic multivariate analysis of high dimensional data. FDA allows for instance to work with irregular measurement points in functional data by replacing sampled functions by simple functional representations, thanks to a B-splines expansion or more generally an expansion on any functional basis. Of course, the functional representation can also be applied to regularly sampled functions. A side effect of the representation is that it can be used to smooth the data either individually or globally (see [2]). Another interesting point is that most FDA methods can work directly on the numerical coefficients of the basis expansion, leading to far less computational burden. An additional advantage of dealing with functions is

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

the possibility of using functional preprocessing such as derivation, integration, etc.

Many traditional data analysis methods have been adapted to functional data, especially linear methods, such as principal component analysis (PCA) and linear regression [10]. Recent developments of FDA include non linear models such as multilayer perceptrons [3, 11] and non parametric models [6, 7]. Unsupervised functional data analysis has not received a lot of attention and is currently limited to PCA and k-means like methods [1, 8].

In this paper, we propose an adaptation of the Self-Organizing Map [9] to functional data. The proposed approach is a FDA inspired generalization of previous works on curves clustering with the SOM such as [4, 5].

2 Function representation

The core of FDA methods consists in representing observations as smooth functions. Let us consider indeed a high dimensional observation vector $y \in \mathbb{R}^n$. We assume that there are corresponding observation points $x \in \mathcal{O}^n$, a function g from $\mathcal{O}$ to $\mathbb{R}$ and measurement errors $\epsilon \in \mathbb{R}^n$ such that:

$$\forall k \in \{1, \ldots, n\},\ y_k = g(x_k) + \epsilon_k$$

In this model, $\mathcal{O}$ can be $\mathbb{R}$ or a higher dimensional input space. Observation points can be explicitly given or can be calculated thanks to prior knowledge on data acquisition. In general, g is restricted to belong to a L^2 functional space defined on $\mathcal{O}$. We don't request the observation points to be identical for all inputs and therefore, the first observation vector can belong to $\mathbb{R}^{100}$ whereas the second one belongs to $\mathbb{R}^{125}$, etc.

Rather than working on y, FDA works on g. This function is reconstructed from the observations thanks to traditional function approximation methods. Such an approximation has to be performed for each observation vector and to avoid computational problems, it is more efficient to use linear methods. More precisely, we choose a topological basis of the considered functional space (e.g., a Hilbert basis of $L^2(\mathcal{O})$), that is a series of functions $(\phi_i)_{i \in \mathbb{N}}$ that is dense in the functional space. Then g is replaced by its projection on the vectorial space spanned by the first p basis functions, where p is a meta-parameter of the analysis. More precisely, the observation vector $y \in \mathbb{R}^n$ is replaced by a vector $\alpha \in \mathbb{R}^p$ such that the following reconstruction error is minimal:

$$\sum_{k=1}^{n} \left(y_k - \sum_{i=1}^{p} \alpha_i \phi_i(x_k) \right)^2$$

As stated in the introduction, this approach allows to deal easily with irregular sampling, as the approximation can be calculated independently for each input vector y, even if each y belongs to a different vectorial space and has its own observation points vector x. As long as the basis remains fixed, each input vector is translated into a fixed size coordinate vector on the truncated basis.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

Choosing a low number of basis functions, i.e. a small p, is a crude way to get rid of the measurement noise. A more sophisticated approach is to keep a high number of basis functions and to add a roughness penalty to the reconstruction error. Details can be found in [10] when each curve is smoothed independently and in [2] when the smoothing is performed on the whole set of curves.

3 Self-Organizing Map on functions

Kohonen's Self-Organizing Map (SOM) can be theoretically applied to data in a normed vectorial space, regardless of its dimension. Let us indeed recall the main steps of the inner loop of the SOM algorithm: for each input vector, the first step is to calculate the winning neuron defined as the one which minimizes the distance between its prototype vector and the input vector. This operation is based only on the norm on the vectorial space. The second step is to update the prototype of the winning neuron and of its neighbors thanks to the following update formula: $p_{t+1} = p_t + \eta(y - p_t)$, where p_t is the prototype vector and y the input vector. This update step is based only on vectorial operations.

Of course, the actual implementation in a functional space introduces technical problems as it is not possible to exactly manipulate arbitrary functions. The obvious solution is to approximate functions. This was done implicitly in previous works which were focused on regularly sampled functions. As they can indeed be considered as high dimensional vectors, the SOM can be directly applied to them, as proposed in [4, 5]. This solution is quite limited as it cannot deal with irregularly sampled functions. A more important drawback is that the approximation might be quite poor. Let us consider indeed an arbitrary function f. The implicit assumption of the simple multivariate approach is that $\|f\|^2$ is correctly approximated by a quantity proportional to $\sum_{i=1}^{n} f(x_i)^2$, where $(x_i)_{1 \leq i \leq n}$ are the observation points. This approximation is good only if there is no observation noise, if f is sufficiently regular and if n is large. When we depart from those assumptions, it is much better to estimate f from the observations and then to approximate vectorial operation and norm calculation on the representation rather than on the raw data.

We therefore propose to use a more flexible method than the multivariate approach, inspired by the method used in [1] for k-means: rather than working on raw data, we first represent each input function by calculating its approximation on a fixed truncated basis. Then we submit to the SOM the coordinates of the representation on the chosen basis, after having preprocessed them so as to have an equivalence between comparison of coordinates and comparison of functional norms (in [1], the k-means algorithm is implemented directly on the coordinates). Indeed, we need to approximate $\|f\|^2$ for a represented function. If $f \simeq \sum_{i=1}^{p} \alpha_i \phi_i$, we have: $\|f\|^2 \simeq \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j \langle \phi_i, \phi_j \rangle$, where $\langle u, v \rangle$ denotes the scalar product between two functions u and v. In general, the basis is neither orthogonal, nor normed. For instance, when dealing with functions from $\mathbb{R}$ to $\mathbb{R}$, B-spline basis, which are not orthogonal, are often used because of their very interesting practical properties: numerical stability, locality, efficient

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

calculation, etc. Nevertheless, the matrix $\Phi(p)_{ij} = \langle \phi_i, \phi_j \rangle$ is symmetric and positive, and has therefore a Cholesky decomposition, i.e. there is a matrix $U(p)$ such that $\Phi(p) = U^T(p)U(p)$. Then obviously $\|f\|^2 \simeq \|U(p)\alpha\|^2$.

In order to approximate the theoretical functional SOM, we submit to a classical numerical SOM $U(p)\alpha$ where α is the coordinate vector of the considered input function. The representation step is consistent with vectorial calculation as it is linear. If f and g have coordinate vectors α and β on the truncated basis, then $\lambda f + \mu g$ has coordinate vector $\lambda\alpha + \mu\beta$, and therefore, after transformation, is manipulated by the SOM as $U(p)\,(\lambda\alpha + \mu\beta) = \lambda U(p)\alpha + \mu U(p)\beta$. Therefore, vectorial operations can be directly implemented on the transformed coordinates.

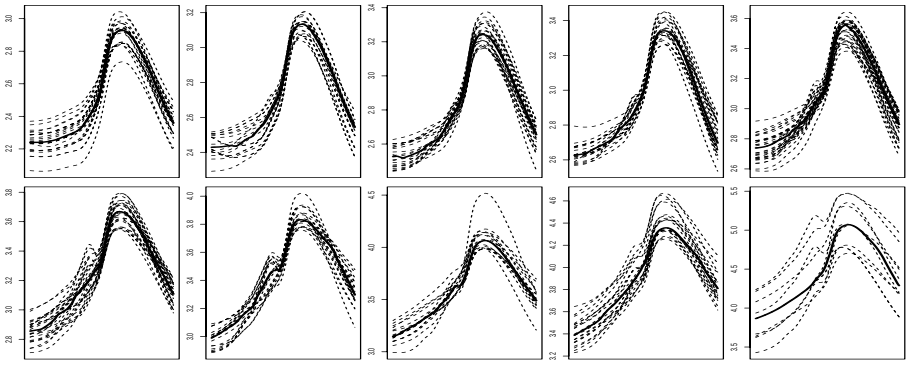


Figure 1: Clustering of spectra

4 An application to spectrometric data

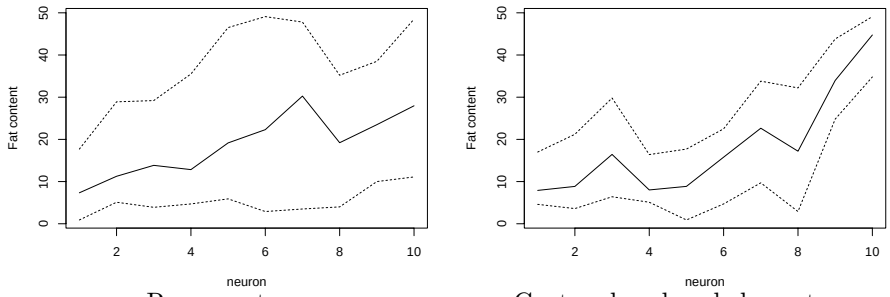
4.1 The data

We study in this section spectrometric data from food industry. Each observation is the near infrared absorbance spectrum of a meat sample (finely chopped), recorded on a Tecator Infratec Food and Feed Analyses (we have 215 spectra). More precisely, an observation consists in a 100 channel spectrum of absorbances in the wavelength range 850–1050 nm. Data can be downloaded from the statlib site[1]. Each spectrum is associated to a content description of the meat sample (obtained by analytic chemistry), that is the percentage of fat, water and protein contained in the sample. Our goal is to classify the spectrum and see whether their shapes are related to the corresponding chemical composition.

The spectrum are very regular and do not seem to contain any measurement noise. Moreover, they are sampled at regular wavelength and with a good resolution (each spectrum belongs to $\mathbb{R}^{100}$). In this situation, the only advantage of the functional approach is to reduce the dimensionality. Indeed, we have projected the spectrum on a B-spline basis using 50 functions, which divides

[1] `http://lib.stat.cmu.edu/datasets/tecator`

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312



Raw spectra Centered and scaled spectra

Figure 2: Maximum, mean and minimum fat contents for each neuron

by 2 the amount of data. As the running time of the SOM is proportional to the dimension of the input space, this reduction allows to perform twice as more runs as with the original data in a given amount of time. In a data exploration context, this allows to test more preprocessing, numerical parameters, etc.

According to the original contributor of the data, the more interesting information is the fat content. We have therefore used a one dimensional SOM with 10 neurons to order the spectra. The result is given by figure 1 (for space reasons, the one dimensional SOM is represented on two rows that should be read from left to right and from top to bottom): each rectangle displays all curves clustered in the corresponding neuron and the median curve in bold. The y range of each cell has been scaled individually so as to enhance readability: a side effect is to hide the fact that the mean value is growing from left to right. Results are almost identical to the one obtained by the multivariate approach (i.e., using a classical SOM on the original $\mathbb{R}^{100}$ input vectors): as the original spectra are noiseless and the sampling is good, the approximation of the functional norm provided with the Euclidean norm in $\mathbb{R}^{100}$ is good enough to reproduce the functional results. It is clear in the obtained classification that the mean value dominates the classification as the shapes of curves associated to one neuron can be quite different.

The problem with this classification is that it has poor explanation power with respect to the fat content, as shown on figure 2 (left). Indeed the mean fat content of spectra clustered in each neuron does not increase with the rank of the neuron in the SOM linear structure. Moreover, the variability is quite important in each cluster. We can calculate the quality of the clustering thanks to the quantization error for the fat, that is the root mean square error obtained when we approximate the fat of each spectrum by the average fat of the spectra of its cluster. We obtain here a very high quantization error: 10.8 (the fat range is $[0.9, 49.1]$). The quantization error for the classical SOM is exactly the same.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

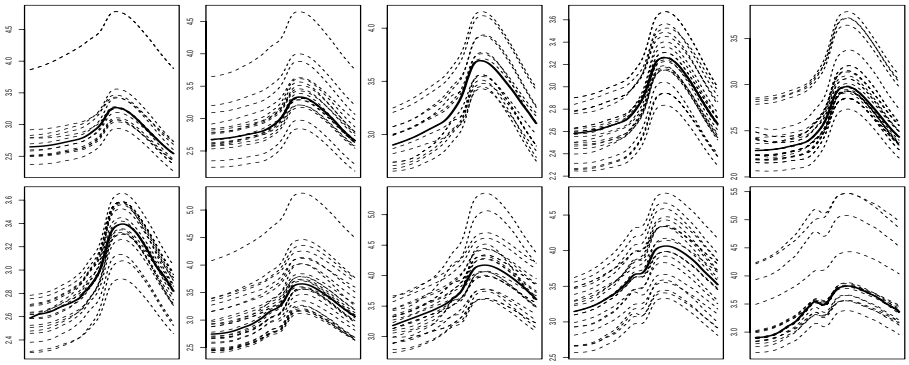


Figure 3: Spectra clustered after centering and scaling

4.2 Centering and scaling

When the mean value dominates the classification, it is tempting to work on individually centered data, i.e. to subtract from each spectrum its mean value. To emphasize even more the shape of the spectrum, we can also scale each spectrum to unit variance. Those preprocessing can be implemented on a multivariate point of view, or on a functional point of view. In the later case, a function g is replaced by g_c defined by $x \mapsto g(x) - \frac{1}{b-a}\int_a^b g(u)du$ (for centering). The resulting function is scaled into g_s defined as:

$$g_s(x) = \frac{g_c(x)}{\frac{1}{b-a}\sqrt{\int_a^b (g_c(u))^2 du}}.$$

The multivariate version can be considered as an approximation of the functional ones. Both versions improve the quality of the clustering with respect to fat content explanation, as summarized in table 1.

	Centered	Centered and scaled
Multivariate	6.82	5.93
Functional	6.74	5.88

Table 1: Quantization errors for various preprocessing

The slight improvement observed in the case of the functional implementation seems to come from a better estimation of the centering and scaling values. Figure 3 gives the classification results (for the functional implementation with scaling). It is obvious that the mean effect has disappeared and that shapes now dominate the classification. Figure 2 (right) shows that the preprocessing (scaling) improves the clustering explanation power with respect to the fat content.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 305-312

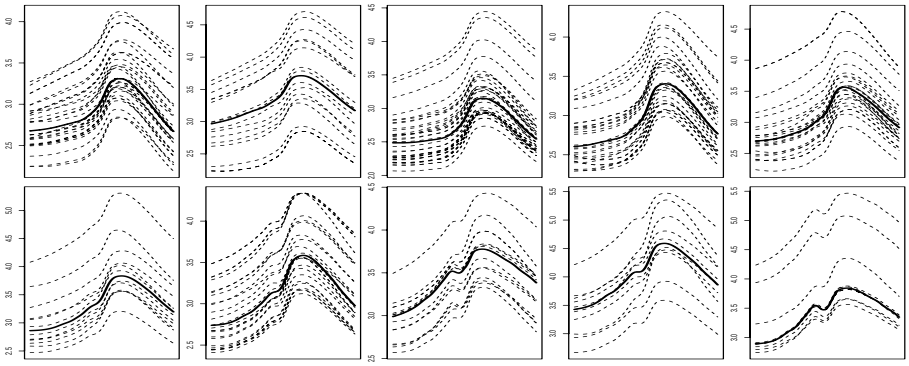


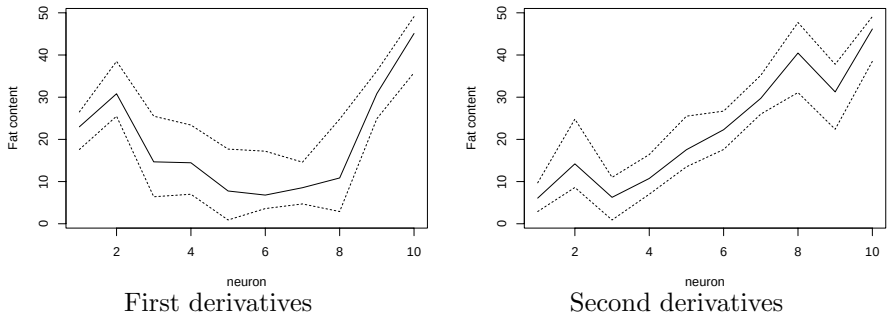
Figure 4: Spectra clustered thanks to their second derivatives

4.3 Functional transformation

As we are dealing with functions, we can implement functional preprocessing. In order to focus on the shape of the spectra rather than on the absorbance values, we calculate the second derivatives of the spectra (see [6, 7]). As they are very smooth, we simply estimate those derivatives with finite differences. Then, we represent the obtained functions on a B-spline basis (with 50 functions) and submit transformed coordinates to a linear SOM. Results are summarized by figure 4. It is clear that the mean effect has completely disappeared and that shapes of clustered curves are more similar (curves on the right of the linear SOM have two maxima rather than one). Moreover, figure 5 (right) shows that the clustering has much more explanation power with respect to the fat content than the first one: indeed, the quantization error is now 3.01. In fact, the second maxima associated to high fat content corresponds to the specific peak of absorbance associated to fat alone, a chemical result that can be rediscovered thanks to the functional approach. We obtain similar but slightly worse results with the first derivatives of the spectra. The quantization error grows to 4.08 and the classification itself is less satisfactory: we obtain curves with two peaks at both ends of the SOM linear structure, a fact that appears on figure 5 (left) where fat does not grow with the index of the neuron. Nevertheless, both functional preprocessing give better result than standard centering and scaling: the functional approach introduces new preprocessing schemes that can be very useful on adapted data.

5 Conclusion

We have proposed in this paper a simple way to implement a self-organizing map on functional data. Even on noiseless and regularly sampled curves, the functional approach allows both to consistently reduce the size of the data and to implement functional transformations that extend the practical possibilities of the SOM. Those results have been confirmed on phoneme data from the



First derivatives Second derivatives

Figure 5: Maximum, mean and minimum fat contents for each neuron

TIMIT database: those very noisy data can be reduced from 256 measurements to only 31 splines coefficients which even give a slightly better classification. Further experimental work is needed to investigate the robustness of the proposed approach to irregularly sampled data.

References

[1] Christophe Abraham, Pierre-André Cornillon, Eric Matzner-Lober, and Nicolas Molinari. Unsupervised curve clustering using b-splines. *Scandinavian Journal of Statistics*, 30(3):581–595, September 2003.

[2] Philippe Besse, Hervé Cardot, and Frédéric Ferraty. Simultaneous nonparametric regressions of unbalanced longitudinal data. *Computational Statistics and Data Analysis*, 24:255–270, 1997.

[3] Brieuc Conan-Guez and Fabrice Rossi. Multi-layer perceptrons for functional data analysis: a projection based approach. In José R. Dorronsoro, editor, *Artificial Neural Networks – ICANN 2002*, pages 667–672, Madrid, August 2002. Springer.

[4] M. Cottrell, B. Girard, and P. Rousset. Forecasting of curves using a kohonen classification. *Journal of Forecasting*, 17:429–439, 1998.

[5] Anne Debrégeas and Georges Hébrail. Interactive interpretation of kohonen maps applied to curves. In *Proc. International Conference on Knowledge Discovery and Data Mining (KDD'98)*, pages 179–183, New York, August 1998.

[6] Frédéric Ferraty and Philippe Vieu. The functional nonparametric model and application to spectrometric data. *Computational Statistics*, 17(4), 2002.

[7] Frédéric Ferraty and Philippe Vieu. Curves discriminations: a nonparametric functional approach. *Computational Statistics and Data Analysis*, 44(1–2):161–173, 2003.

[8] Gareth M. James and Catherine A. Sugar. Clustering for sparsely sampled functional data. *Journal of American Statistical Association*, 98:397–408, 2003.

[9] Teuvo Kohonen. *Self-Organizing Maps*. Springer Verlag, New York, 1997.

[10] Jim Ramsay and Bernard Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer Verlag, June 1997.

[11] Fabrice Rossi, Brieuc Conan-Guez, and François Fleuret. Theoretical properties of functional multi layer perceptrons. In *Proceedings of ESANN 2002*, pages 7–12, Bruges, Belgium, April 2002.