Interactive Interpretation of Kohonen Maps Applied to Curves

Anne DEBREGEAS, Georges HEBRAIL

ELECTRICITE DE FRANCE (EDF)
Research Division - Statistics, Optimization and Decision-Aid Group
1, Av. du Général de Gaulle
92141 CLAMART CEDEX - FRANCE
Anne.Debregeas@edfgdf.fr, Georges.Hebrail@der.edfgdf.fr

Abstract

For large customers, Electricité de France - the French national electric power company - stores every 10' the amount of electric power they consume. For each customer, these measures lead to curves called electric load curves. Clustering of electric load curves is a key problem for understanding the behavior of these customers. Several methods have been used but Kohonen maps give a very nice solution to this problem thanks to the visualization of the map. The work we present here describes a software for interactive construction and interpretation of a Kohonen map clustering, in the case of curves. The user can run the Kohonen map clustering, visualize the map, see external characteristics of curves linked to each cell of the map, find the cells figuring curves having some chosen external characteristics, define classes of cells, add comments on cells. User interaction is largely based on mouse clicking on the map cells and on bars of barcharts figuring external characteristics of the curves. This software is not dedicated to electric load curve analysis but can be used on any type of curve, for instance to analyze time series in finance.

Introduction

Electricité de France studies the behavior of its large customers in terms of power consumption. The standard way to do it is to collect the energy consumed by some of these customers every 10 minutes all along the year. Once data are collected, a load curve is built for each customer, with a duration of a day, a week, a month, or a year. Analysis of load curves is very useful to understand the customers behavior, in particular in relation to the pricing policy. The standard way of analyzing a large set of load curves is to perform a clustering of the curves, so that experts look at a small number of classes (i.e. clusters) of similar curves instead of at the whole set of curves.

Much attention has been given recently to curve and time-series analysis: see for instance (Agrawal *et al.* 1996), (Keogh and Smyth 1997). Many studies have been carried out in our research center on the problem of load curve clustering: see (Chantelou, Hébrail and Muller 1996) for some references. Both hierarchical clustering and self-organizing maps (Kohonen maps) have been tried out on load curves: see (Chantelou, Hébrail and Muller 1996) and (Boudaillier and Hébrail 1997). Self-organizing maps appear to be preferred for three reasons:

- they offer a good visualization capability, even with a large number of clusters,
- it is possible to build maps with a large number of clusters (i.e. around 100) which enables a microscopic analysis of the dataset,
- the self-organizing algorithm complexity is proportional to the number n of curves, while hierarchical clustering is n^2: large datasets can still be analyzed.

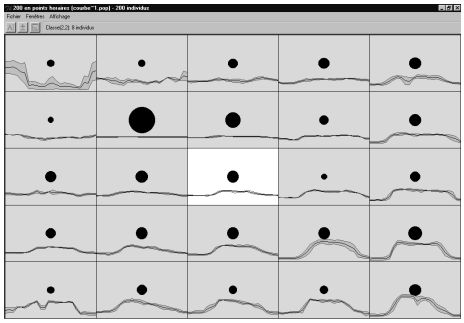
In this paper, we present a software (developed in our research center) which helps load curve experts to build self-organizing maps of large datasets of curves. The main features of this software are the following:

- it follows a user's task model which leads to the association of comments with each cell of the map, and to the creation of groups of cells,
- the self-organizing map algorithm has been improved by performing a principal component analysis to initialize the centers of the cells: this speeds up the convergence of the algorithm,
- the map can be visualized interactively and a special window shows a planar representation of distances between the cells of the map,
- interactive barcharts figuring external characteristics of curves (as pricing, activity, season, ...) are linked with the map, as a help to give an interpretation to cells of the map,
- interactive clustering of the cells into a small number of classes is possible and can be done by different means.

Once the load curve expert user has interpreted a Kohonen map clustering, the result is a map with some comments associated with each cell or each group of cells. This map contains the expertise of the analyst user and can be used by some other (operational) users to compare curves of new customers to curves appearing in the map. The last section is devoted to a short presentation of a first classification module featuring these capabilities.

This software has been developed using C++ for all numerical computations (Kohonen algorithm, barcharts,

Figure 1: A self-organizing map with 25 cells



statistical indicators, ...) and with Tcl/Tk for the user interface. It operates both on UNIX and PC/Windows stations.

This paper is presented with a set of curves figuring electric power consumption, but the software is general and can be applied without any change to any type of curves.

Self-Organizing Map Clustering

The Kohonen algorithm is a particular clustering method (Kohonen 1995). It provides a partition of objects into disjoint clusters preserving some proximity constraints between the clusters. The constraints are given by a topology structure defined by the user. In our case, we use a map structure as in Figure 1. Each cell of the map is a cluster of objects and neighbor cells on the map are built to contain more similar objects than clusters far away on the map. This property enables to define maps with a large number of clusters (i.e. 100 with a 10 by 10 map as in Figure 7), which remain readable since cells are ordered by the map topology.

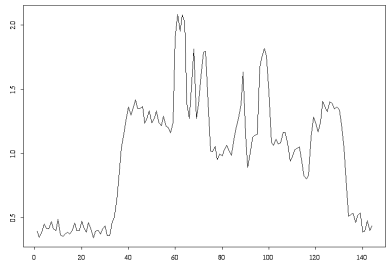
Electric Load Curves

The dataset we consider here is a set of 2,665 daily load curves[2] corresponding to the electric power consumption of 205 customers over 13 Thursdays in winter. A daily load curve is given by a numerical vector of dimension 144, representing the energy consumed during each period of 10 minutes in the day from 0h to 23h50. Figure 2 contains an example of such a daily load curve. Though visualization of a single load curve is really easy to do, there is a problem when we want to have a synthetic idea of the shape of 2,665 load curves.

The need for visualizing a large set of curves comes from the problem of analyzing the behavior of large customers in relation to the pricing policy defined by our company. A sample of 205 medium-voltage customers has been chosen: for these customers, we collect the energy

[2] This dataset has been provided by the « Service Etudes de Réseaux - Département Consommations, Clientèle et Télécommunications » of our research center.

Figure 2: Example of a daily load curve



they consume for every period of 10 minutes. In the raw dataset, the value of each 10' period is the energy consumption in kilo-watt hour. Each value is divided by the mean value of the day. This transformation is done to analyze the shape of curves and not the absolute level of consumption. After this transformation, customers with high and low average consumption during the day both have a mean consumption of 1 during the day.

As for external characteristics describing curves, we consider the observation day and season of the curve, the activity sector of the customer, as well as the characteristics of its contract with EDF. The goal of the analysis is to find classes of load curves according to their shape and to study relationships between these classes and external characteristics.

The special feature with curves is that they have a nice graphical representation:

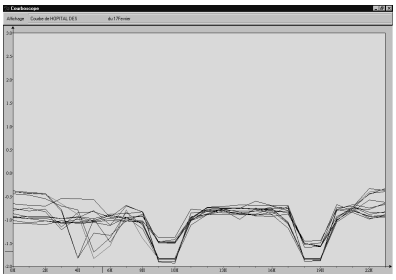
- each curve can be represented by a line plot as in Figure 2,
- a set of curves can be either represented by its mean curve or by juxtaposition of the curves (see Figure 3 for an example of juxtaposition of a set of curves).

Since each cell of a self-organizing map corresponds to a set of curves, a graphical representation can be associated with each cell as shown in Figure 1.

Speeding up the Clustering Process

Another interesting feature of self-organizing maps is that the algorithm complexity is proportional to the number of items to be clustered. So large datasets can still be

Figure 3: Detail of a cell by juxtaposition of curves



processed by this method. But in practice, it is necessary to read many times the dataset to avoid convergence problems of the algorithm. If the number of iterations is too small, the result map can be far away from the optimal map and therefore mislead the user in the interpretation process. To solve this problem, we propose the following approach: centers of the initial map considered by the algorithm are initialized by a *principal component analysis* applied to the dataset of curves, instead of taking randomly some curves in the dataset.

Principal component analysis - see (Saporta 1990), (Jobson 1992), and (Lebart 1997) - is a standard method of statistical data analysis for dimension reduction of quantitative data. Since curves are described by a numerical vector of 144 dimensions, this method can be applied. Only one read operation of the dataset file is necessary to perform it since it is based on a singular value decomposition of the covariance matrix. This method finds the plane which minimizes the distorsion of the dataset when it is projected onto it. On this plane, a grid is defined and some points are chosen to be the centers of the initial cells of the Kohonen algorithm. Coordinates of these points are transformed into the initial 144 space by the reconstitution formula (Saporta 1990). This choice of initial centers helps the convergence of the Kohonen algorithm since it has been shown that Kohonen maps build a representation which is close to principal component analysis (Lebart 1997).

So, by this process, self-organizing maps become much more usable on very large datasets of curves. Another advantage of this approach is that the global shape of the map does not change if the Kohonen algorithm is run again on the same dataset. As a matter of fact, when the basic algorithm is run several times on the same dataset, results may be different due to some symmetric transformations or convergence to different local optima. A good choice of the orientation of the principal component axes leads to the same global map, even if the algorithm is re-run: this appeared to be very important for the end-user.

Description of the User's Task

The task of the end-user we consider here is the interpretation of a self-organizing map clustering applied to a dataset of curves. The result of this task is double:

- the association, with each map cell, of a comment or label defined by the user after a careful interpretation of the contents of the cell, both in terms of the cell curves and in terms of the external characteristics of these curves,
- the definition of some disjoint groups of cells (clusters of cells we call 'superclasses') which lead to a two-level clustering of the curves. Superclasses are also assigned a comment by the user.

Interpretation of each cell and superclass is made by observing characteristics of the curves assigned to the cell. It is here necessary to distinguish among curve characteristics:

- *curve points* which are used in the clustering process to define a similarity between curves,
- *external* characteristics associated with the curves, which are not used in the clustering process.

Both curve points and external characteristics are useful to build interpretations. The clustering process is also a way for the user to study correspondences between the shape of the curves and their external characteristics.

The result of the interpretation process reflects the expertise of the user, mainly by the definition of the cell and superclass labels. This result is used by another module which is a classification module: another type of user (final user) can merge new curves into the interpreted map.

Browsing the Self-Organizing Map

The first functionality of the software is that the user can import a set of curves associated with some external characteristics. Then a self-organizing map can be built interactively: the user only defines the size of the map (for instance 5 by 5 as in Figure 1). The software then performs the algorithm (with the initialization process described before) and displays the map window on the screen as in Figure 1.

Browsing the Map

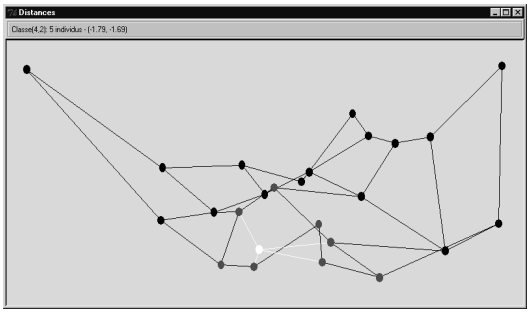
Each cell is represented by its mean curve, and by the curves representing and +/- 1.5 standard error curves of the cells. This gives information about the dispersion of the contents of each cell. In this map window the black circles represent the size of each cell (i.e. the number of curves assigned to the cell): large cells have a large circle, small cells a small one, and empty cells figure nothing (no mean curve and no circle).

If the user wants more detail about a particular cell, it is possible to display a window figuring the juxtaposition of all curves of the cell as in Figure 3. All curves in this window are mouse sensitive so that the curve under the mouse pointer is instantly highlighted in red: it is thus possible to see the shape of some untypical curves among all the mixed curves.

Browsing the Distance Window

The distance window can be displayed on demand and figures the real topology of the map (see Figure 4). Actually, the map window gives a distorted representation of distances between cells: neighbor cells may be much far away one from each other than it is perceived on the map. The distance window represents each cell by its center point projected onto the plane defined by the principal component analysis. This window is linked with the map

Figure 4: Distance window of the map of Figure 1



window so that when the mouse is on a cell, the corresponding center is highlighted in the distance window (neighbors of the cell are also highlighted with a different color). The same kind of interaction is available from the distance window to the map window.

Interpretation with External Characteristics

The map browsing described in the previous section could be sufficient to assign a comment to each cell. This is due to the graphical representation of curves. The software also enables the user to use external characteristics to do so.

Linked Barcharts

For each external characteristic describing curves, a barchart can be drawn (see Figure 5). In the case of numerical characteristics, histograms replace barcharts. Barcharts are linked to the map window. When the user clicks on a cell, barcharts represent the distribution of characteristic values of the cell curves (in black) compared to the distribution in the whole dataset (in white).

A special window is available to help the user to select characteristics which are statistically the most linked with the self-organizing map clustering (a chi-square test is performed).

Test-Values

Moreover, when the user clicks on a barchart bar, it is

Figure 5: Barcharts linked with the map

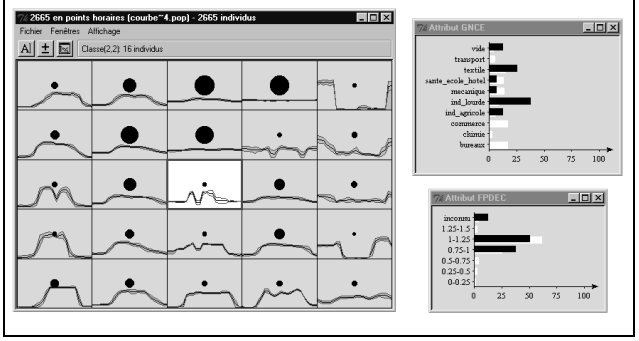
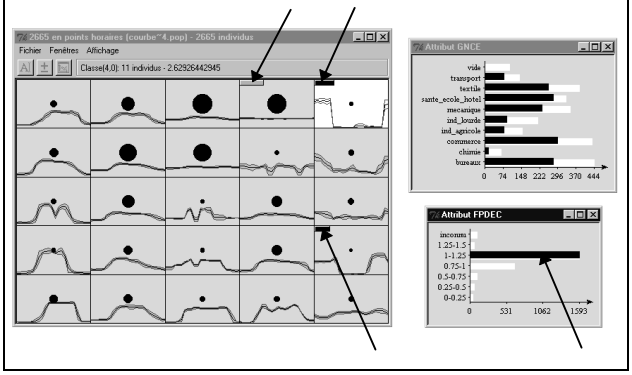


Figure 6: Test-values



equivalent to select all curves having the corresponding characteristic value in the dataset. Then, the following information are displayed (see Figure 6):

- all other barcharts are instantly updated to show the distribution of the selected curves on other characteristics, like in standard statistical software,
- information is displayed to show map cells for which the selected value is characteristic (with a black bar in the cell) or anti-characteristic (with a gray bar).

For each cell and each characteristic value, a statistical indicator is computed to indicate if the set of curves of the cell have a high proportion of curves with the given value or not. This indicator is called a *test-value* (Morineau 1984). Long black (resp. gray) bars in the cell mean the selected characteristic value is very characteristic (resp. anti-characteristic) of the cell, while short bars or no bar mean that there is nothing special.

Labeling

The labeling of one cell can be done at any time by selecting the cell and clicking on the labeling button of the map window. The user is prompted to enter a character string which can be displayed in the cell.

Interactive Partitioning of Cells

The software enables the user to define interactively groups of cells into some disjoint superclasses. This cell clustering can be done manually by the user who creates the superclasses, assigns them a label and a color, and build them by painting some map cells with a superclass color (see Figure 7).

A wizard is also available to initialize automatically some superclasses, by two alternative methods:

- a clustering of the centers of the cells (approximately the mean curve of each cell) is performed by a *k-means* partitioning algorithm (the user defines manually the number of superclasses),

- an external characteristic is chosen by the user and one superclass is created for each value of the characteristic (numerical characteristics are transformed into intervals). Then, each cell is painted with a superclass color if the associated value is very present in the cell[3]. Some cells may remain unpainted if no value is significatively present.

So, these facilities enable the user to build superclasses based either on the shape of the cell curves or on a particular external characteristic of the curves. For instance, it is interesting to initialize superclasses based on the type of contract the customer has signed with EDF.

A First Classification Module

A first version of a classification module has also been developed. It takes as input the result of the previously described module and provides the user with the following features:

- visualization of the map where each cell figures its label and mean curve,
- importation of new curves to be merged into the map, either one by one, or all together. In this case, the user visualizes where the new curves are situated in the map.

This module is intended for final users (non-analyst users). An example of application of this module is to help commercial agents to choose the best pricing for a particular large customer, according to the shape of his (her) load curves.

This merging module also makes a large use of graphical representations of cell curves and curves to be merged. It is designed so that the final user can validate the assignment of new curves to map cells according to his (her) expertise.

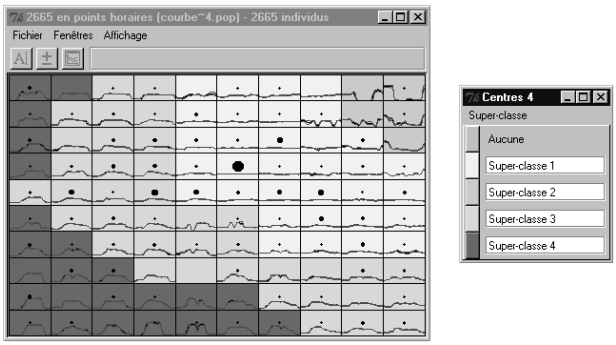
Conclusion and Further Work

The first module of the software presented here helps an analyst user to build and interpret Kohonen maps from a large dataset of electric load curves. The expertise of this analyst user is captured into an interpreted Kohonen map. This module is general and can be applied to any type of curve, for instance in finance, insurance, or marketing. We are currently evaluating this module to improve it from user feedback.

The second module enables final users to merge new curves into an interpreted Kohonen map. This module is not general but dedicated to our application on electric load curve analysis. The merge of a set of curves into an interpreted map can find many other applications in several domains. But here specific modules have to be developed to fit exactly final user needs within the context of the application.

[3] A threshold is defined on test-values.

Figure 7: 100 cells map clustered into 4 superclasses



Acknowledgements. A.Morineau (CISIA) suggested to initialize the self-organizing map clustering by a principal component analysis. C.Muller (EDF), D.Chantelou (EDF), C.Derquenne (EDF), and Y.Lechevallier (INRIA) helped in the design of the software. E.Boudaillier (UNIFIX) and X.Galante (ARCHE 2) programmed it.

References

Agrawal, R., Mehta, M., Shafer, J., Srikant, R., Arning, A., and Bollinger T. 1996. The Quest Data Mining System. In *Proc. of the 2nd Int'l Conference on Knowledge Discovery and Data Mining*, AAAI/MIT Press.

Boudaillier, E., and Hébrail, G. 1997. Interactive Interpretation of Hierarchical Clustering. In *Proceedings of PKDD'97*, Lecture Notes in Artificial Intelligence, Principles of Data Mining and Knowledge Discovery, 288-298, Springer.

Chantelou, D., Hébrail, G., and Muller C. 1996. Visualizing 2,665 Electric Power Load Curves on a Single A4 Sheet of Paper. In *Proceedings of the ISAP'96 Conference*, Orlando (Florida).

Keogh, E., Smyth, P. 1997. A Probabilistic Approach to Fast Pattern Matching in Time Series Databases. In *Proc. of the 3nd Int'l Conference on Knowledge Discovery and Data Mining*, AAAI/MIT Press.

Kohonen, T. 1995. *Self-organizing Maps*. Berlin: Springer.

Jobson, J.D. 1992. *Applied Multivariate Data Analysis.* Vol.II, Springer-Verlag.

Lebart, L. 1997. Méthodes Factorielles. In *Statistique et méthodes neuronales*, S.Thiria, Y.Lechevallier, O.Gascuel, S.Canu Eds, Paris:Dunod.

Morineau, A. 1984. Note sur la Caractérisation Statistique d'une Classe et les Valeurs-test. Bulletin du CESIA, Vol.2, N°1-2, Paris.

Saporta, G. 1990. *Probabilités, Analyse des Données, et Statistique*. France:Editions Technip.