



UNIVERSITÀ DEGLI STUDI DI MACERATA

*Dipartimento di Economia e Diritto*

## A Mixed Integer Linear Programming Approach to Markov Chain Bootstrapping

R. Cerqueti, P. Falbo, C. Pelizzari, F. Ricca, A. Scozzari

Quaderno di Dipartimento n. 67

Novembre 2012

# A Mixed Integer Linear Programming Approach to Markov Chain Bootstrapping

*R. Cerqueti, P. Falbo, C. Pelizzari, F. Ricca, A. Scozzari*

## Abstract

Bootstrapping time series is one of the most acknowledged tools to make forecasts and study the statistical properties of an evolutive phenomenon. The idea underlying this procedure is to replicate the phenomenon on the basis of an observed sample. One of the most important classes of bootstrap procedures is based on the assumption that the sampled phenomenon evolves according to a Markov chain. Such an assumption does not apply when the process takes values in a continuous set, as frequently happens for time series related to economic and financial variables. In this paper we apply Markov chain theory for bootstrapping continuous processes, relying on the idea of discretizing the support of the process and suggesting Markov chains of order  $k$  to model the evolution of the time series under study. The difficulty of this approach is that, even for small  $k$ , the number of rows of the transition probability matrix is too large, and this leads to a bootstrap procedure of high complexity. In many practical cases such complexity is not fully justified by the information really required to replicate a phenomenon satisfactorily. In this paper we propose a methodology to reduce the number of rows without losing “too much” information on the process evolution. This requires a clustering of the rows that preserves as much as possible the “law” that originally generated the process. The novel aspect of our work is the use of Mixed Integer Linear Programming for formulating and solving the problem of clustering similar rows in the original transition probability matrix. Even if it is well known that this problem is computationally hard, in our application medium size real-life instances were solved efficiently. Our empirical analysis, which is done on two time series of prices from the German and the Spanish electricity markets, shows that the use of the aggregated transition probability matrix does not affect the bootstrapping procedure, since the characteristic features of the original series are maintained in the resampled ones.

*Roy Cerqueti*, Università degli Studi di Macerata.

E-mail: roy.cerqueti@unimc.it.

*Paolo Falbo*, Università degli Studi di Brescia.

E-mail: falbo@eco.unibs.it.

*Cristian Pelizzari*, Università degli Studi di Brescia.

E-mail: pelizcri@eco.unibs.it.

*Federica Ricca*, Sapienza Università di Roma.

E-mail: federica.ricca@uniroma1.it.

*Andrea Scozzari*, Università degli Studi “Niccolò Cusano” - Telematica Roma.

E-mail: andrea.scozzari@unicusano.it.

# 1 Introduction

After the seminal paper by Efron (1979), several developments and applications of bootstrap methods appeared in the literature. Bootstrap methods following the original idea by Efron and based on re-sampling of model errors have been largely applied in Economics and Finance. The reader is referred to Freedman (1984), Freedman and Peters (1984), Efron and Tibshirani (1993) for a methodological discussion and to Brock et al. (1992), Sullivan et al. (1999) for an application to financial markets. However, the original approach of Efron suffers, in general, of model miss-specification risk and requires observations to be time independent. To overcome such limitations, nonparametric, model-free bootstrap methods have been proposed in the literature. In Bühlmann (2002) several bootstrap methods of this type are compared, such as the *block*, the *sieve*, and the *local methods*. The advantage of nonparametric, model-free methods is that they do not require the observations to be time independent: data themselves capture the dependency structure of the time series, thus relieving the researcher of the responsibility of choosing a model. Among the nonparametric bootstrap methods, a relatively recent group appeared in the literature which is based on Markov chain theory (see, e.g., Anatolyev and Vasnev, 2002). The major issue in this research area is the estimation of the true dimension of the transition probability matrix, which, in turn, consists of estimating the relevant states and the order of the process, that is, the length of its “memory”. Even if these two estimates refer to different objects, they are not completely independent. They have been intensively examined in the area of Information Theory to model alphabet processes (see, for example, Bühlmann and Wyner, 1999; Bühlmann, 2002; Zhu et al., 2002). Brilliant applications, such as efficient data compression, have been obtained therein. However, these studies concern only discrete processes and, since discrete states are usually considered equally relevant in such literature, the main issue has been to estimate the true order of a Markov chain. It must be pointed out that, in some cases, the order  $k$  of a Markov process is specified in advance, depending on the observed series. This may cause that the number of rows in the transition probability matrix is too large w.r.t. the information contained in each single row. In this case, one would like to reduce the number of rows without losing too much information about the process evolution. In Zhu et al. (2002) the problem is solved by searching for a suitable clustering of the rows of the matrix, putting together the rows which are “similar”. Their application relies on the compression algorithm devised by Spears (1998) for square transition probability matrices.

When the process is continuous, the above approach does not apply. Nevertheless, a novel application of Markov chain theory for bootstrapping continuous processes was proposed in the literature. The importance of this application is mainly due to the increasing interest in the economic and financial areas, for modeling continuous valued processes as “Markov switching regimes” (see, e.g., Hamilton, 1996; Jeanne and Masson, 2000; Hamilton, 2005). This kind of processes shows different evolutive features depending on which regime characterizes a given period. In such models, identifying the states that significantly influence the evolutive properties of the process is as important as estimating the correct length of its memory. From a complexity viewpoint, if a Markov chain with  $n$  states is supposed to have a memory of  $k$  time lags, the number of possible paths driving the evolution of the process is  $n^k$ , which increases exponentially with  $k$ . Such paths correspond to the rows of the transition probability matrix, whose size therefore increases exponentially with  $k$ , as well. Following Zhu et al. (2002), in Cerqueti et al. (2009) the authors developed the idea of clustering the rows of the transition probability matrix to reduce its dimension and yet preserving as much as possible the “law” that generated an observed path. They state an optimization problem in which, given a memory of  $k$  time lags, the set of the  $n^k$  rows is partitioned into  $q \leq n^k$  classes by minimizing a distance indicator between rows which is based on the corresponding transition probabilities. The main difficulty of this approach is related to the computational complexity for solving the optimization problem. In fact, the proposed solution method relies on complete enumeration which cannot be applied in real situations, even if the authors provide a pre-processing that drastically reduces the size of the solution space<sup>1</sup>. Partitioning problems are among the most studied in Operations Research (see, e.g., Anily and Federgruen, 1991). In the large majority of cases these problems have been solved by means of heuristic algorithms, like the  $k$ -means algorithm introduced by Forgy (1965) and MacQueen (1967), graph-theoretic approaches to data clustering (see, e.g., Ricca et al., 2008; Wu and Leahy, 1993, and references therein), Tabu Search-based heuristics (see,

---

<sup>1</sup>The size of the solution space is reduced from  $B(n^k)$  to  $[B(n)]^k$ , where  $B(i)$  is the Bell number of  $i$ .

e.g., Cerqueti et al., 2011; Ricca and Simeone, 2008), Ant Colony algorithms as in Trejos et al. (2004), evolutionary algorithms (see, Ma et al., 2006), block diagonal matrix decompositions (see, Courtois, 1977).

In this paper we are interested in the practical solution of the problem of partitioning the set of rows of the transition probability matrix of a Markov process of (fixed) order  $k$ , the final aim being to exploit the optimal partition in the corresponding bootstrap procedure. We propose a new Mixed Integer Linear Programming formulation for the above partitioning problem that can be solved via common optimization packages, like CPLEX<sup>2</sup>. This allows to efficiently solve problems where the number of rows of the matrix ranges from 40 to 60, which corresponds to the typical size of some economic and financial phenomena. As an application of this method we consider the problem of bootstrapping the series of the Spanish and German electricity prices observed daily for 6 and 7 consecutive years, respectively.

The paper is organized as follows. In Section 2, we discuss in detail Markov chain bootstrapping and describe the Mixed Integer Linear Programming model proposed for partitioning the rows of a transition probability matrix. Section 3 is dedicated to the application of our approach to real-life problems: after the description of the available datasets, we illustrate our results providing a statistical analysis of the bootstrapped series which are compared to the original ones. In Section 4, we collect some final thoughts and concluding remarks.

## 2 Markov chain bootstrapping based on Mixed Integer Linear Programming

In this section we introduce the Markov chain resampling problem and propose an optimization approach to solve it. The basic ideas underlying the problem are first introduced, together with the corresponding notation and definitions.

Consider a time-varying phenomenon. Under the hypothesis that the phenomenon evolves according to a  $k$ -th order Markov chain  $(X(t), t \geq 0)$ , our aim is to resample it through a bootstrap procedure. The estimation of the transition probability matrix associated to  $(X(t), t \geq 0)$  relies on an available time-ordered sample of observations of the investigated phenomenon.

In resampling procedures, two aims are pursued which are somehow conflicting: on the one hand, the exact replication of the sample at each simulation should be avoided (*diversification* or *multiplicity* criterion); on the other hand, the statistical properties of the original sample should be reproduced as much as possible in the replications (*similarity* criterion). In order to generate a bootstrapped series satisfying the two above properties, we proposed a method based on the aggregation of the rows of the transition probability matrix of  $(X(t), t \geq 0)$ . The problem is formulated as a partition problem of the st of rows of the transition probability matrix, and it is modeled as a Mixed Integer Linear Program (MILP).

### 2.1 Notation and definitions

Assume that the possible realizations of an evolutive phenomenon vary in an interval  $[\alpha, \beta] \subseteq \mathbb{R}$ . Consider the following series of time-ordered realizations of the phenomenon observed in the time horizon  $(1, \dots, t, \dots, T)$ :

$$e(T) = (e_1, \dots, e_t, \dots, e_T). \quad (1)$$

We will also refer to  $T$  as the *length* of the observed series. Let  $[b_0, b_1), \dots, [b_{z-1}, b_z), \dots, [b_{n-1}, b_n]$  be a partition of  $[\alpha, \beta]$  into  $n$  intervals such that  $b_0 = \alpha$  and  $b_n = \beta$ . In order to simplify the notation, we refer to the  $z$ -th interval  $[b_{z-1}, b_z)$  as  $\alpha_z$ . This is only a matter of notation, and  $\alpha_z$  is not meant to be a single point replacing the whole interval  $[b_{z-1}, b_z)$ . One has:

$$\begin{cases} \alpha_i \cap \alpha_j = \emptyset, & i, j = 1, \dots, n, i \neq j \\ \bigcup_{z=1}^n \alpha_z = [\alpha, \beta]. \end{cases}$$

We collect the elements of such partition in a set  $A = \{\alpha_1, \dots, \alpha_z, \dots, \alpha_n\}$  so that  $A$  represents the set of all possible states for the time-ordered series under analysis. States in the set  $A$  are called *theoretical*. We

<sup>2</sup>[www.ibm.com/software/integration/optimization/cplex-optimizer/](http://www.ibm.com/software/integration/optimization/cplex-optimizer/).

will refer to the above partition of  $[\alpha, \beta]$  as the *initial partition*. A *state* corresponds to one of the intervals  $\alpha_z$ ,  $z = 1, \dots, n$ , which are then called the *initial intervals* or *initial states*. The choice of adopting the term “state” is taken to refer explicitly to the Markov chain framework described later in this section. Relying on the above discretization, one has that, for any observed value  $e_t$  (point), there is a unique  $\alpha_z$  (interval) in  $A$  such that  $e_t \in \alpha_z$ . As a consequence, a time series as (1) will be represented as a sequence of states of a Markov chain  $a(T) = (a_1, \dots, a_t, \dots, a_T)$ , where  $a_t = \alpha_z$  when  $e_t \in \alpha_z$ , for  $z = 1, \dots, n$ . Given the set  $A$  of the initial states, for a fixed  $k$  the Cartesian product  $A^k$  collects all the  $k$ -tuples formed by states in  $A$  which we then call *theoretical  $k$ -states*. Thus, given states  $\alpha_z \in A$ ,  $z = 1, \dots, n$ , we denote a generic theoretical  $k$ -state by  $\alpha^h(k) = (\alpha_{h_k}, \alpha_{h_{k-1}}, \dots, \alpha_{h_1})$ , with  $h_w \in \{1, \dots, |A|\}$  and  $w = 1, \dots, k$ .

We denote a sequence of  $k < T$  consecutive elements extracted from  $e(T)$  as:

$$e(\tilde{t}, k) = (e_{\tilde{t}}, \dots, e_{\tilde{t}+j}, \dots, e_{\tilde{t}+k-1}) \quad \text{for some } \tilde{t} \in \{1, \dots, T - k + 1\}. \quad (2)$$

Notice that in formula (2)  $e(\tilde{t}, k)$  depends on two parameters, namely, the *length*  $k$  of the above sequence and  $\tilde{t}$ , which identifies the starting time of the sequence. An *observed  $k$ -state* (or simply  *$k$ -state*)  $a(\tilde{t}, k)$  is the sequence of the  $k$  observed states corresponding to  $e(\tilde{t}, k)$ . More precisely,  $a(\tilde{t}, k) = (a_{\tilde{t}}, \dots, a_{\tilde{t}+j}, \dots, a_{\tilde{t}+k-1})$  is obtained by setting  $a_t = \alpha_z$  when  $e_t \in \alpha_z$ , for  $t \in [\tilde{t}, \tilde{t} + k - 1]$ .

Let  $\mathcal{O}_k$  denote the set of observed  $a(\tilde{t}, k)$ , that is

$$\mathcal{O}_k = \left\{ a(\tilde{t}, k), \tilde{t} \in \{1, \dots, T - k + 1\} \right\},$$

for which one has  $|\mathcal{O}_k| = T - k + 1$ .

Assume that the evolutive phenomenon is modeled as a Markov chain of order  $k \geq 1$ ,  $(X(t), t \geq 0)$ , with state space  $A$ . Then, the  $k$ -th order transition probability matrix is estimated by using the definition of Ching et al. (2008), which involves the estimation of the empirical frequencies.

We introduce the transition probability for the process to reach state  $\alpha_z$  immediately after the sequence of states in  $\alpha^h(k)$ , that is:

$$\mu_{\alpha^h(k), \alpha_z}^{(k)} = \text{Prob}(X(t) = \alpha_z | X(t-1) = \alpha_{h_1}, \dots, X(t-k) = \alpha_{h_k}). \quad (3)$$

Let us denote by  $M$  the  $k$ -th order transition probability matrix, with  $|A^k| = n^k$  rows and  $|A| = n$  columns. To avoid a cumbersome notation, we set  $n^k = m$ . Matrix  $M$  is built starting from the observed sample  $e(T)$  and its generic entry  $(h, z)$  contains the value  $\mu_{\alpha^h(k), \alpha_z}^{(k)}$  corresponding to  $k$ -state  $\alpha^h(k)$  in  $A^k$  and state  $\alpha_z$  in  $A$ .

According to the approach in Ching et al. (2008), we estimate  $\mu_{\alpha^h(k), \alpha_z}^{(k)}$  as follows<sup>3</sup>:

$$\mu_{\alpha^h(k), \alpha_z}^{(k)} = \begin{cases} \frac{\Lambda(\alpha^h(k), \alpha_z)}{\sum_{j=1}^n \Lambda(\alpha^h(k), \alpha_j)} & \text{if } \sum_{j=1}^n \Lambda(\alpha^h(k), \alpha_j) \neq 0 \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

where

$$\Lambda(\alpha^h(k), \alpha_z) = \left| \bigcup_{\tilde{t}=1}^{T-k} \{a(\tilde{t}, k+1) \in \mathcal{O}_{k+1} : a(\tilde{t}, k+1) \equiv (\alpha^h(k), \alpha_z)\} \right| \quad (5)$$

is the number of times that a sequence  $(\alpha^h(k), \alpha_z)$  has been observed in the sample.

For a given  $k$ , each row of the transition probability matrix  $M$  contains the transition probabilities from a  $k$ -state  $\alpha^h(k) \in A^k$  to the states  $\alpha_z \in A$ . Formula (4) is used to compute  $M$  on the basis of the original sample. Notice that, when a  $k$ -state is observed only once in the original series  $e(T)$ , the estimation of its transition probabilities is equal to 1 for the unique state to which the corresponding trajectory

<sup>3</sup>For the sake of simplicity, we avoid introducing here a specific notation for the estimates of the probabilities.

evolved, while it is equal to 0 for all the other states. We refer to this case as *deterministic k-states*, and the corresponding rows of the transition probability matrix are called *deterministic rows*. Each row not completely filled with zeros reports the estimated probabilities from an observed  $k$ -state to an observed state, which substantially means that only sequences  $\alpha^h(k)$  such that  $\alpha^h(k) \equiv a(\tilde{t}, k)$  for some  $\tilde{t} \in \{1, \dots, T - k + 1\}$  and states  $\alpha_z \equiv a_t$  for some  $t \in \{1, \dots, T\}$  are considered.

To introduce our MILP model, in the following we indifferently refer to  $\alpha^u(k)$  and  $\alpha^v(k)$  of  $A^k$  and to the corresponding rows,  $u$  and  $v$  respectively, of matrix  $M$ . Similarly, we will indifferently refer to the set  $A^k$  and to the corresponding set of rows  $V$  of matrix  $M$ .

A simple *dissimilarity measure* between pairs of rows  $u$  and  $v$  of  $M$  can be defined as follows:

$$d_{uv} = \sum_{z=1}^n \left| \mu_{\alpha^u(k), \alpha_z}^{(k)} - \mu_{\alpha^v(k), \alpha_z}^{(k)} \right|. \quad (6)$$

It is easy to check that this dissimilarity measure takes values in the interval  $[0, 2]$  (see Cerqueti et al., 2009). As we will argue below, the dissimilarity  $d_{uv}$  can be viewed as a proxy for the “cost” of putting  $k$ -states  $\alpha^u(k)$  and  $\alpha^v(k)$  *together*, i.e. in the same class of a partition.

## 2.2 An optimization approach to the aggregation of the rows of a transition probability matrix

Let us introduce our partitioning problem of the set  $V$  of the rows of the transition probability matrix  $M$ . Consider a partition of  $V$  into  $q$  classes,  $\pi = \{C_1, \dots, C_p, \dots, C_q\}$ , with  $1 \leq q \leq m$ , and let  $\Pi$  be the set of all possible such partitions.

The *diameter of a class*  $C_p$  is defined as the maximum dissimilarity between two elements in  $C_p$ , and it is denoted by  $\delta(C_p)$ :

$$\delta(C_p) = \max_{u, v \in C_p} d_{uv}. \quad (7)$$

The *diameter of a partition*  $\pi \in \Pi$  is defined as

$$\Delta(\pi) = \max_{C_p \in \pi} \delta(C_p).$$

Formula (7) suggests that in the particular case of a class  $C_p$  including only one element we have  $\delta(C_p) = 0$ , since  $d_{uu} = 0$  for each  $u \in V$ .

For a given  $\pi \in \Pi$ , we denote by  $C_{\bar{p}}$  a class for which  $\Delta(\pi) = \delta(C_{\bar{p}})$ , that is:

$$C_{\bar{p}} \in \operatorname{argmax}\{\delta(C_p), C_p \in \pi\}.$$

In our partitioning problem we want to minimize both the number of classes and the diameter of the partition. These two objectives are obviously in conflict, since the diameter of a partition tends to increase when the number of its classes decreases and vice versa. In fact, we are faced with a bi-objective problem which we handle by minimizing the diameter of the partition while controlling for the number of its classes. In fact, we introduce in the model a parameter  $\gamma \geq 0$  for bounding the value of the diameter  $\Delta(\pi)$  from below, which implies a constraint on the cardinality of the partition. By means of  $\gamma$  we implement a control in the resampling procedure to meet the multiplicity criterion, i.e., to avoid the exact replication of the original sample. Hence, given a fixed value  $\gamma \geq 0$  and an integer  $1 \leq q \leq m$ , we formulate the following partitioning problem:

*among all the possible partitions  $\pi \in \Pi$  of  $V$  with at most  $q$  classes and such that  $\delta(C_{\bar{p}})$  is at least equal to  $\gamma$ , find a partition  $\pi^*$  that minimizes  $\Delta(\pi)$ .*

The problem can be formulated as a Mixed Integer Linear Program, which we illustrate in the following.

To formalize the optimization model, let us observe that a partition  $\pi = \{C_1, \dots, C_p, \dots, C_q\}$ , with  $1 \leq q \leq m$ , can be equivalently written as  $\pi = \{C_1, \dots, C_p, \dots, C_m\}$ , with

$$\begin{cases} C_p \neq \emptyset, & \text{for } p = 1, \dots, q \\ C_p = \emptyset, & \text{for } p = q + 1, \dots, m \end{cases}. \quad (8)$$

**Definition 1.** Given a partition of  $V$ ,  $\pi = \{C_1, \dots, C_p, \dots, C_m\}$ , the cluster  $C_p$  is active if and only if  $C_p \neq \emptyset$ .

Let  $v \in V$  be a generic row of matrix  $M$ . We introduce the binary variables  $x_{vp}$  and  $y_p$ , with  $p = 1, \dots, m$ , such that:

$$x_{vp} = \begin{cases} 1, & \text{if row } v \text{ belongs to cluster } C_p \\ 0, & \text{otherwise} \end{cases},$$

$$y_p = \begin{cases} 1, & \text{if cluster } C_p \text{ is active} \\ 0, & \text{otherwise} \end{cases}.$$

Consider now  $u, v \in V$ . It is easy to see that the product  $x_{up} \cdot x_{vp}$  is equal to 1 if and only if the rows  $u$  and  $v$  belong to the same cluster  $C_p$  of the partition:

$$x_{up} \cdot x_{vp} = \begin{cases} 1, & \text{if both } u \text{ and } v \text{ belong to cluster } C_p \\ 0, & \text{otherwise.} \end{cases}.$$

Hence, the *cost* of assigning both  $u$  and  $v$  to cluster  $C_p$  is given by:

$$d_{uv} \cdot x_{up} \cdot x_{vp}.$$

With this notation, the diameter of cluster  $C_p$  can be rewritten as  $\delta(C_p) = \max_{u,v \in V} \{d_{uv} \cdot x_{up} \cdot x_{vp}\}$ . The problem of finding the partition of the set of rows  $V$  into *at most*  $q$  classes that minimizes the maximum diameter of a cluster can be formulated as follows:

$$\begin{aligned} \min_{\pi \in \Pi} \quad & \max_{C_p \in \pi} \quad \max_{u,v \in V} \{d_{uv} \cdot x_{up} \cdot x_{vp}\} \\ (1) \quad & \sum_{p=1}^m x_{vp} = 1 && \forall v \in V \\ (2) \quad & x_{vp} \leq y_p && \forall v \in V, \quad p = 1, \dots, m \\ (3) \quad & \sum_{p=1}^m y_p \leq q \\ & x_{vp}, y_p \in \{0, 1\} && \forall v \in V, \quad p = 1, \dots, m. \end{aligned} \tag{9}$$

Notice that in the above formulation we bound the number of classes of the partition by  $q$ . Thus, varying  $q$  in  $\{1, \dots, m\}$  allows us to tackle the bi-objective problem as a sequence of  $m$  single-objective ones. In problem (9) the first set of constraints states that each row must belong to only one class of the partition. According to constraints (2), a row can belong to a class  $C_p$  only if  $C_p$  is active. Constraint (3) provides an upper bound on the number of classes of  $\pi$ . Problem (9) is an integer nonlinear program which can be linearized by introducing additional *association* variables  $w_{uvp}$ , with  $0 \leq w_{uvp} \leq 1$ :

$$w_{uvp} = x_{up} \cdot x_{vp} = \begin{cases} 1, & \text{if both } u \text{ and } v \text{ belong to cluster } C_p \\ 0, & \text{otherwise} \end{cases},$$

and the following set of constraints:

$$\begin{aligned} w_{uvp} &\leq x_{up} && \forall u, v \in V, \quad p = 1, \dots, m \\ w_{uvp} &\leq x_{vp} && \forall u, v \in V, \quad p = 1, \dots, m \\ w_{uvp} &\geq x_{up} + x_{vp} - 1 && \forall u, v \in V, \quad p = 1, \dots, m. \end{aligned} \tag{10}$$

Thus, if both  $u$  and  $v$  belong to cluster  $C_p$  one has  $x_{up} = x_{vp} = 1$ , and therefore  $w_{uvp} = 1$ . On the other hand, if either  $x_{up} = 0$  or  $x_{vp} = 0$  one has  $w_{uvp} = 0$ , too. This also implies that the bounds  $w_{uvp} \leq 1$  are

always satisfied, and hence they need not to be included explicitly in the problem formulation. Notice that constraints (10) guarantee that variables  $w_{uvp}$  assume only values 0 or 1, and, therefore, they can be introduced in the model as real variables.

The objective function can be written in terms of the new variables as follows:

$$\min_{\pi \in \Pi} \max_{C_p \in \pi} \max_{u, v \in V} \{d_{uv} \cdot w_{uvp}\}.$$

Then, the objective function can be linearized introducing a new variable  $d$  which replaces the expression  $\max_{u, v \in V} \{d_{uv} \cdot w_{uvp}\}$  and adding the following set of constraints:

$$d_{uv} \cdot w_{uvp} \leq d, \quad \forall u, v \in V, \quad p = 1, \dots, m.$$

Let us notice that the above model does not prevent the optimal objective function value to become very small (it may even turn out to be 0). We recall that the core of the bootstrap procedure is to generate resamplings of the observed series  $e(T)$  which must be sufficiently “similar” to  $e(T)$ , in order to guarantee that they can be seen as different realizations of the same phenomenon which also generated  $e(T)$ . However, they must be also sufficiently “diversified” from  $e(T)$ , in order to have some degree of “variability” among them. The optimal partition applied to  $M$  should then be structured so as to guarantee these requirements as much as possible. If, on the one hand, similarity is pursued in the model via the minimization of the diameter, on the other hand, diversification can be controlled by imposing the diameter to be not smaller than a prefixed threshold  $\gamma$ . This additional constraint prevents from choosing the trivial *singleton partition*, where each class is formed by a single element. In fact, for this partition, the value of the objective function is zero, but no aggregation of the rows of  $M$  is actually performed.

In order to formalize the additional condition related to the threshold  $\gamma$ , we introduce the binary variables  $t_{uvp}^\gamma$  defined as follows:

$$t_{uvp}^\gamma = \begin{cases} 1, & \text{if } d_{uv} \cdot w_{uvp} \geq \gamma \\ 0, & \text{otherwise} \end{cases} \quad \forall u, v \in V, \quad p = 1, \dots, m.$$

Adding further suitable constraints involving  $t_{uvp}^\gamma$  and  $\gamma$ , we obtain the MILP for our specific partitioning problem:

$$\begin{aligned} & \min d \\ (1) \quad & \sum_{p=1}^m x_{vp} = 1 && \forall v \in V \\ (2) \quad & x_{vp} \leq y_p && \forall v \in V, \quad p = 1, \dots, m \\ (3) \quad & \sum_{p=1}^m y_p \leq q \\ (4) \quad & w_{uvp} \leq x_{up} && \forall u, v \in V, \quad p = 1, \dots, m \\ (5) \quad & w_{uvp} \leq x_{vp} && \forall u, v \in V, \quad p = 1, \dots, m \\ (6) \quad & w_{uvp} \geq x_{up} + x_{vp} - 1 && \forall u, v \in V, \quad p = 1, \dots, m \\ (7) \quad & d_{uv} \cdot w_{uvp} \leq d && \forall u, v \in V, \quad p = 1, \dots, m \\ (8) \quad & d_{uv} \cdot w_{uvp} \geq t_{uvp}^\gamma \cdot \gamma && \forall u, v \in V, \quad p = 1, \dots, m \\ (9) \quad & \sum_{p=1}^m \sum_{u, v \in V} t_{uvp}^\gamma \geq 1 \\ & x_{vp}, y_p \in \{0, 1\}, t_{uvp}^\gamma \in \{0, 1\} && \forall u, v \in V, \quad p = 1, \dots, m \\ & d \geq 0, w_{uvp} \geq 0 && \forall u, v \in V, \quad p = 1, \dots, m. \end{aligned} \tag{11}$$



In the above model, constraints (7)-(8) describe the relation between the variables  $w$  and the new variables  $t^\gamma$ . Constraint (9) forces at least one variable  $t^\gamma$  to be equal to 1, thus guaranteeing that the diameter of the optimal partition is at least  $\gamma$ .

### 2.3 Bootstrap procedure

In the following, we describe the bootstrapping procedure based on the optimal partition  $\pi^*$  of model (11). The algorithm takes as inputs  $\pi^*$  and the aggregated transition probability matrix  $M^*$  associated to  $\pi^*$ , as well as, the observed time series  $e(T) = (e_1, \dots, e_t, \dots, e_T)$ , and it returns a resampled series. It is not required that the resampled series and the original one have the same length. We denote by  $\ell$  the length of a resampled series (that we fix in advance), and by  $x(\ell) = (x_1, \dots, x_\ell)$  the time ordered sequence of its values.

The optimality of  $\pi^*$  should guarantee that, although generated through a coarser structure of the information in the aggregated transition probability matrix  $M^*$ , a resampled series maintains simultaneously a good statistical similarity w.r.t. the original sample and a satisfactory diversification from other resampled series.

We briefly present the main steps of the bootstrapping procedure, listed in the following pseudo-code, where we use the notation  $a(\tilde{t}, k) \in C$  to denote that the sequence of states in  $a(\tilde{t}, k)$  belongs to class  $C$  of partition  $\pi^*$ . The main step of the algorithm concerns the (random) choice of a  $k$ -state among those belonging to a given class of  $\pi^*$ .

To avoid the rare but not impossible case of a premature stop of the bootstrap procedure, the last  $k$ -state observed in  $e(T)$  is excluded from the random choice if it has never been observed before.

---

**algorithm:** BOOTSTRAP( $\pi^*$ ,  $M^*$ ,  $e(T)$ ,  $\ell$ )

---

**input:** The optimal partition  $\pi^*$ , the corresponding transition probability matrix  $M^*$ , the observed series  $e(T)$  of at least  $2k + 1$  observations, and an integer number  $\ell \geq k + 1$ .

**output:** A resampled series  $x(\ell) = (x_1, \dots, x_\ell)$ .

---

1. **begin**
  2. Let  $e(k + 1, k) = (e_{k+1}, \dots, e_{2k})$  be the  $k$  elements of  $e(T)$  observed at times  $k + 1, \dots, 2k$ .
  3. Set  $(x_1, \dots, x_k) := (e_{k+1}, \dots, e_{2k})$ .
  4.     **for**  $j = 1$  to  $\ell - k$  **do**
  5.         Consider  $(x_j, \dots, x_{j+k-1})$   
and the corresponding sequence of states  $a(j, k) = (a_j, \dots, a_{j+k-1})$ .
  6.         **if**  $a(j, k) \in \mathcal{O}_k$
  7.             Let  $C$  be the class of  $\pi^*$  including  $a(j, k)$ .  
Based on  $M^*$  randomly choose one of the  $k$ -states belonging to  $C$   
among those that have passed the *absolute continuity filter*.
  8.             Let  $a(\bar{\tau}_j, k)$  be the selected  $k$ -state.
  9.             Set  $x_{j+k} := e_{\bar{\tau}_j+k}$ .
  10.         **else**
  11.             Set  $r := 1$ .
  12.             Consider  $(x_{j+r}, \dots, x_{j+k-1})$   
and the corresponding sequence of states  $a(j + r, k - r) = (a_{j+r}, \dots, a_{j+k-1})$ .
  13.             **if**  $a(j + r, k - r) \in \mathcal{O}_{k-r}$
  14.                 Among all the  $(k - r)$ -states  $a(\tau, k - r) = a(j + r, k - r)$ ,  
select uniformly one, say  $a(\bar{\tau}_{j+r}, k - r)$ .
  15.                 Set  $x_{j+k} := e_{\bar{\tau}_{j+r}+(k-r)}$ .
  16.             **else**
  17.                 Set  $r := r + 1$  and go to step 12.
  18.             **end if**
  19.         **end if**
  20.     **end for**
  21. **end**
- 

In steps 2.-3., the first  $k$  resampled values  $(x_1, \dots, x_k)$  are initialized to the  $k$  observations  $e_{k+1}, \dots, e_{2k}$  in the original series  $e(T)$ . Although the estimation of the transition probability matrix  $M^*$  was performed on all the values of the original series, we exclude from the bootstrap procedure the first  $k$  observed values in  $e(T)$ ,  $e_1, \dots, e_k$ . Indeed, none of these values can represent the possible evolution of a  $k$ -state, because, by construction, it is not possible to identify a whole  $k$ -state preceding each of the values  $e_1, \dots, e_k$ .

The other  $\ell - k$  resampled values are obtained in the loop performed in steps 4.-20. In particular, relying on the optimal partition  $\pi^*$ , steps 5.-9. generate the  $(k + j)$ -th value of the series when the last resampled  $k$ -state  $a(j, k)$  is one of those observed (i.e., it belongs to  $\mathcal{O}_k$ ) as follows: suppose that  $C$  is the class of  $\pi^*$  which  $a(j, k)$  belongs to. The algorithm first selects the  $k$ -states belonging to  $C$  that have passed the “absolute continuity filter” and, then, to obtain the next resampled value, it performs a random choice that is respectful of the transition probabilities of class  $C$  contained in  $M^*$ . To this purpose, we follow uniform resampling from the  $k$ -states included in class  $C$  (net of the filtering mentioned before). Although the technique is a standard one, the following example briefly illustrates how it works.

**Example.** Consider the observed series  $a(T) = \{1, 3, 3, 3, 1, 3\}$ , with  $T = 6$ , and for which  $\mathcal{O}_2 = \{(1, 3), (3, 3), (3, 3), (3, 1), (1, 3)\}$ . Suppose that the class  $C$  of  $\pi^*$  is formed by the 2-states  $(1, 3)$  and  $(3, 3)$ . According to formula (4), in  $M^*$  the probabilities of evolving from class  $C$  to the three states  $\alpha_1 = 1$ ,  $\alpha_2 = 2$  and  $\alpha_3 = 3$  are  $1/3$ ,  $0$ , and  $2/3$ , respectively, since the observed 2-state  $(1, 3)$  evolved to

3, as well as one of the two observed 2-states (3, 3), while the other observed 2-state (3, 3) evolved to 1.  $\square$

The *absolute continuity filter* is a statistical tool used here to exclude from the selection those  $k$ -states of  $\mathcal{C}$  that have transition probabilities not “absolutely continuous” w.r.t. those of  $a(j, k)$ . More specifically, suppose that the  $k$ -state generated by the procedure, say  $a(j_1, k)$ , belongs to the same class of another  $k$ -state, say  $a(j_2, k)$ . Then, the procedure considers  $a(j_2, k)$  for possible selection only if its transition probability is *absolutely continuous*<sup>4</sup> w.r.t. the one of  $a(j_1, k)$ , i.e.

$$Prob(a_z|a(j_1, k)) = 0 \Rightarrow Prob(a_z|a(j_2, k)) = 0, z = 1, \dots, n. \quad (12)$$

This device is meant to prevent (even if it is not guaranteed) the generation of resampled series that evolve following sequences of states that have never been observed.

Steps 10.-19. are designed to handle the generation of a new resampled value when a  $k$ -state that has not been observed in  $\mathcal{O}_k$  arises. In fact, in such cases, relying only on  $M^*$ , we would not be able to extract the next value for the resampled series. To get over such hurdles, we resort to a local “reduction of memory” that amounts to progressively reducing the order of the Markov chain until, for some  $r$ , a  $(k - r)$ -state is observed in  $\mathcal{O}_{k-r}$ . Let  $a(\bar{\tau}_{j+r}, (k - r))$  be the observed  $(k - r)$ -state, the next resampled value is set equal to  $e_{\bar{\tau}_{j+r}+(k-r)}$ . Notice that, since we do not compute transition probabilities for  $(k - r)$ -states, for this local step we perform a random uniform selection among all the  $(k - r)$ -states in  $\mathcal{O}_{k-r}$  showing the same sequence of states as  $a(\bar{\tau}_{j+r}, (k - r))$ .

### 3 Application and results

In this section we describe an experimental application of our bootstrapping method. It relies on data taken from the Spanish and German electricity markets which we are going to describe in detail in the following. The analyzed series show several interesting features which make them difficult to treat for a bootstrapping method, therefore they represent challenging tests for evaluating the performance of our new approach.

#### 3.1 Data

In our application we study two time series, namely the “Mibel Spanish Electric System Arithmetic Average Price” and the German “EEX Phelix Day Base Price”. The series have been observed daily in the period from January 1st, 1998 to December 31st, 2003 (Spain) and from June 17th, 2000 to May 8th, 2007 (Germany). The prices are expressed in euros and refer to 1 MWh. The Spanish time series consists of  $T = 2190$  observations, while for the German series we have  $T = 2517$ .

Figures 1 and 2 show the two time series which are characterized by the following features:

- a weekly and annual seasonality;
- a slightly positive trend;
- stochastic volatility;
- nonlinear dependency of data;
- two clear regimes of prices: *normal trading* and *occasional spiking* periods.

Spikes are occasional, since they usually correspond to unexpected shortages on the supply side of the electricity system, or unexpected and temporary increases of the demand (e.g., sudden meteorological events driving to high consumption). Since our data consist of daily data, intra-day seasonality is neglected in this analysis. Because of the joint presence of such features, in the literature, electricity price

<sup>4</sup>In probability theory, a probability  $\mu$  is said *absolutely continuous* w.r.t. a probability  $\nu$  if, for every event  $B$ , one has  $\mu(B) = 0$  whenever  $\nu(B) = 0$ .

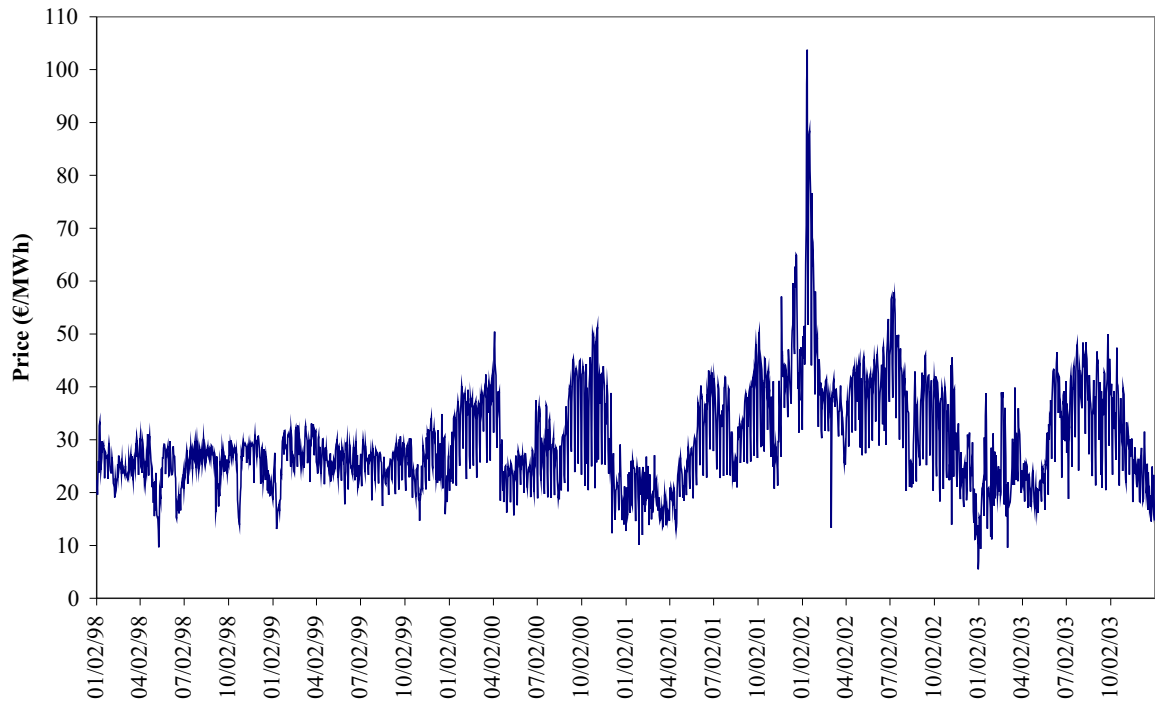


Figure 1: Spanish daily electricity prices.

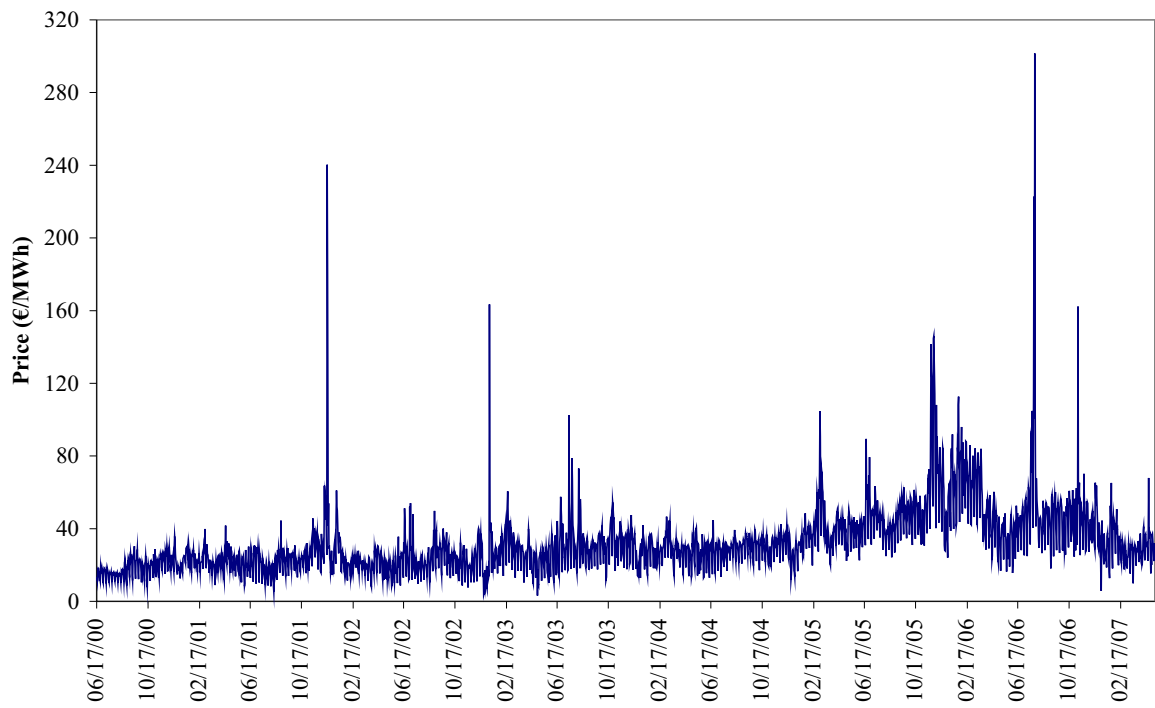


Figure 2: German daily electricity prices.

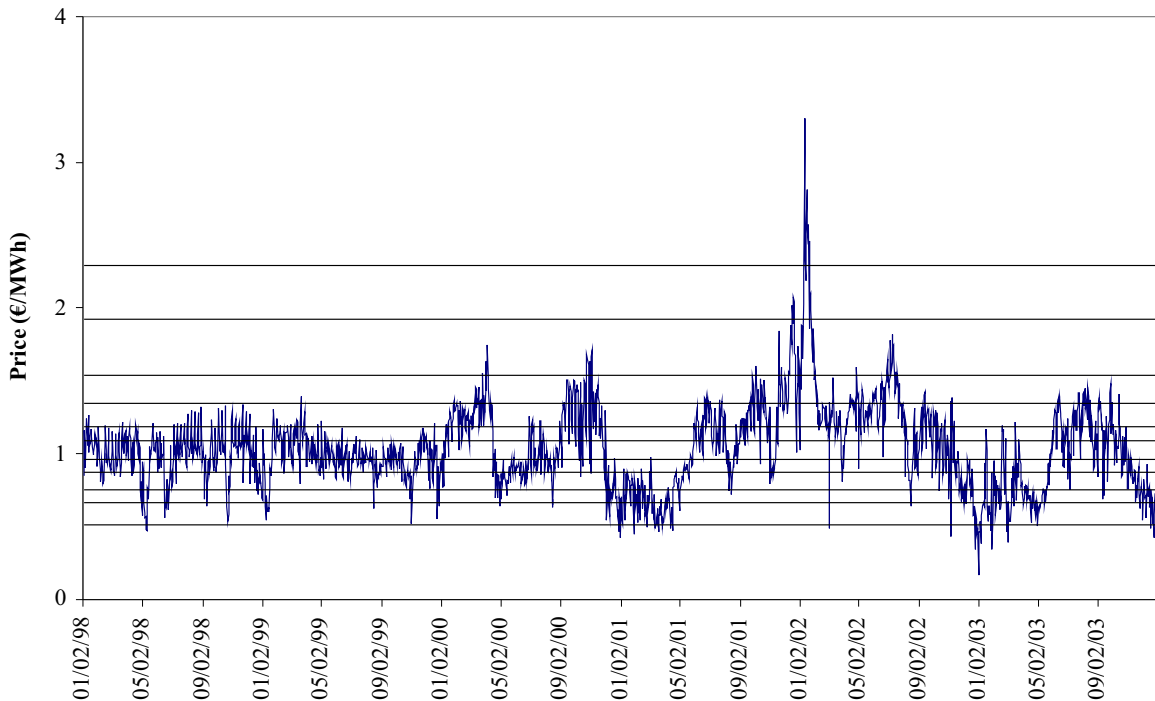


Figure 3: Spanish daily electricity prices without exponential trend and exponential weekly seasonality. Horizontal lines mark the extremes of the intervals  $\alpha_1, \dots, \alpha_{12}$ .

series are usually considered as “hard to model” cases. For a review of the difficulties in modeling electricity prices and the methods developed to solve them, see, for example, Bunn (2004), Huisman and Mahieu (2003), Weron et al. (2004), and Weron (2006). Raw data prices have been removed of an (exponential) weekly seasonal component as well as of an (exponential) trend. Two main reasons justify this pre-treatment. Removing the weekly seasonality lets us free to reduce the order of the Markov chain below 7, which corresponds to a great reduction in the complexity of our problem. The removal of the trend component makes the series more stationary. Of course both components are added back to the bootstrapped series. The estimation of exponential (rather than linear) components is recommended to avoid that this removal/reintroduction process generates occasional negative prices. In Appendix A we give the details about this data treatment.

### 3.2 Preliminary segmentation of the support

As already explained in Subsection 2.1, to discretize our continuous valued process, a preliminary segmentation of the support  $[\alpha, \beta]$  is performed by partitioning it into  $n$  initial intervals  $\alpha_1, \dots, \alpha_z, \dots, \alpha_n$ . In our application we set  $n = 12$  which corresponds to an order of magnitude larger than the number of states commonly considered in the literature for the regimes of electricity prices (usually 2 or 3, see, e.g., Huisman and Mahieu, 2003).

The segmentation of the support into 12 initial states was performed through the minimum-variance clustering procedure provided in Ward (1963), after having removed trend and weakly seasonality from the original series (see, Appendix A for further details). Figures 3 and 4 show the series of Spain and Germany, together with the initial 12 intervals  $\alpha_1, \dots, \alpha_{12}$  (separated by horizontal lines). Appendix B details this preliminary segmentation.

The transition probability matrices  $M_S$  and  $M_G$  of order  $k = 2$ , estimated for the Spanish ( $S$ ) and German ( $G$ ) markets, respectively<sup>5</sup>, were computed using formula (4).

The set of observed 2-states,  $\mathcal{O}_2$ , is composed by 2188 elements for the Spanish instance and 2515 elements

<sup>5</sup>They are available at the web address: <http://chiara.eco.unibs.it/~pelizcri/Partitioning.html>.

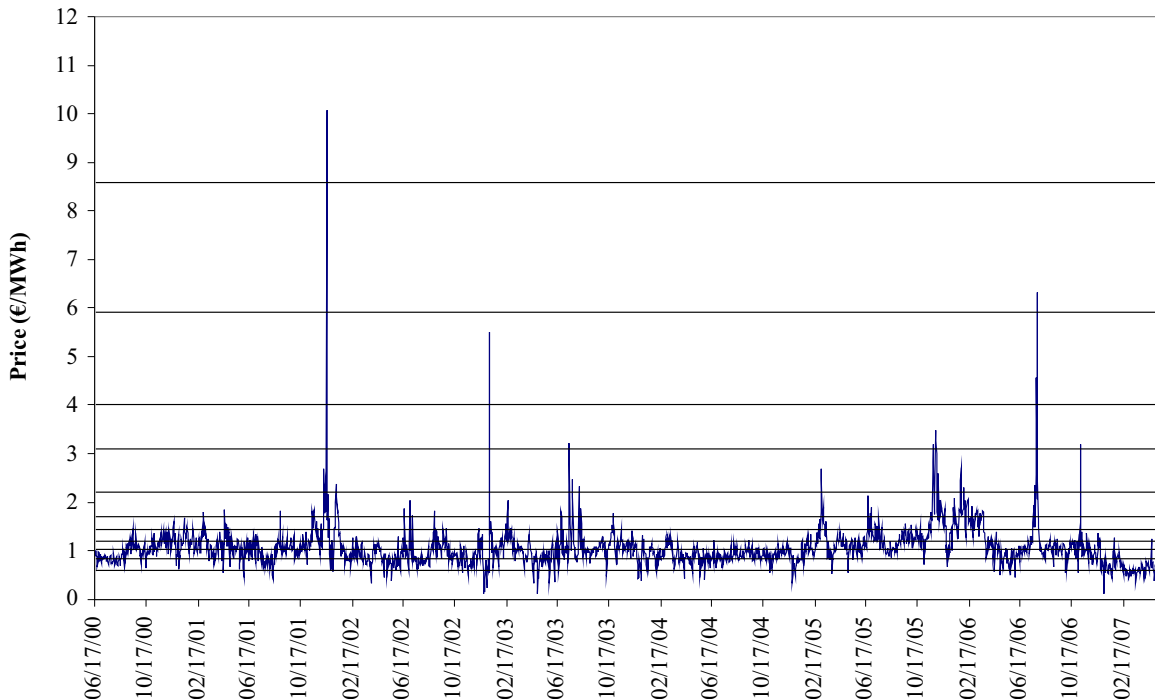


Figure 4: German daily electricity prices without exponential trend and exponential weekly seasonality. Horizontal lines mark the extremes of the intervals  $\alpha_1, \dots, \alpha_{12}$ .

for the German one<sup>6</sup>. The percentage of deterministic 2-states over all the observed ones is about 1% in both cases (23 for Spain and 21 for Germany). The remaining observed 2-states are probabilistic (2165 for Spain and 2494 for Germany). In partitioning the rows of the transition probability matrices  $M_S$  and  $M_G$ , we actually consider only the probabilistic 2-states. On the other hand, we define a priori a single class collecting all the non observed 2-states, while we set up as many classes as the number of observed deterministic 2-states. For  $n = 12$  we obtain 63 and 44 non deterministic rows in the transition probability matrix of Spain and Germany, respectively, while the deterministic rows are 19 in both cases.

### 3.3 Optimization phase

The central element in the optimization model is given by parameter  $\gamma \geq 0$ . This parameter provides a lower bound on the minimum distance between two rows belonging to the cluster with maximum diameter in an optimal solution. Fixing  $\gamma$  to a positive value guarantees some degree of diversity between the rows of such a cluster, and one can be sure that at least one cluster of the partition is not formed by a single row. When  $\gamma = 0$ , no bound is imposed on the objective function value, so that the trivial singleton partition corresponding to one row per class can be always obtained as the optimal solution, provided that the maximum number of clusters  $q$  is large enough (it suffices to set  $q = m$ ). Since the optimal value of the singleton partition is always 0, it can be considered as the partition with maximum similarity within clusters<sup>7</sup>, and, therefore, it can be taken as a benchmark for the evaluation of the partition finally chosen to perform the bootstrapping procedure.

Following the solution approach illustrated in Subsection 2.2, for  $\gamma$  we tested a finite number of values equally spaced in the interval  $(0, 2)$ . This is a standard practice adopted when significant values have to be tested for a parameter ranging in a continuous interval. However, it must be noticed that, in our case, the values of  $\gamma$  that one can actually consider for a given dataset depend on the dataset itself. In fact, since the maximum distance between two rows of the transition probability matrix is given, our MILP

<sup>6</sup>For both Spain and Germany the last observed 2-state is excluded from the computation of the cardinality of  $\mathcal{O}_2$ .

<sup>7</sup>Actually, in this case, no aggregation at all is performed on the rows of the transition probability matrix.

Table 1: Summary of the results of the MILP model for the two datasets. The number of classes corresponds to the smallest value of  $q$  for which the optimal objective function value can be attained.

$\gamma$	Spain		Germany	
	Objective function value	Number of classes	Objective function value	Number of classes
0.50	0.50	52	0.50	23
0.75	0.75	49	0.75	16
1.00	1.00	40	1.00	8
1.25	1.257	30	1.428	5
1.50	1.50	25	1.50	5

model turns out to be infeasible for all values of  $\gamma$  greater than this maximum. For the two analyzed datasets, we tested the following values of  $\gamma$ : 0.5, 0.75, 1, 1.25, 1.5. For each value of  $\gamma$ , we applied our MILP model with increasing values for the integer parameter  $q$  that corresponds to the maximum number of clusters of the partition, i.e.  $q = 2, 3, \dots, m-1$ . As  $q$  increases, we observe a converging process towards a stable optimal value of the objective function. Thus, taking into account that our original partition problem is a bi-objective one, for any fixed  $\gamma$ , among the optimal partitions provided by the MILP model for different values of  $q$ , the best choice is to select the partition with the minimum objective function value (similarity criterion) and the smallest  $q$  (multiplicity criterion). Since in the model we minimize the maximum diameter of a cluster  $d$ , and  $\gamma$  is a lower bound for  $d$ , the ideal case would be attaining such a bound. In our experiments, we obtained this result in 8 out of 10 cases, and we got very close to  $\gamma$  in one of the other two. In Table 1, for each dataset and value of  $\gamma$ , we report the optimal value of the objective function and the smallest  $q$  for which such value can be attained.

Solving the MILP model for a single value of  $q$  and fixed  $\gamma$  required a reasonable computational time (less than 3 hours) for the German series. A major effort was performed for solving the case corresponding to the Spanish series, which leads to a much larger MILP than in the German case. Indeed, for a given value of  $q$  and  $\gamma$ , an average of 10 hours was necessary to get the optimal solution in all cases but one for which, however, after a time limit of 20 hours, we found a best solution value of 1.257, which is very close to the lower bound  $\gamma = 1.25$  (see Table 1). For the model solution we used the well-known optimization software AMPL<sup>8</sup>. The experiments have been carried out on a cluster of two machines, each equipped with two Quad-Core Xeon E5520 processors with 2,26Ghz clock rate and 48 GB RAM.

Based also on the results reported in Table 1, and on the statistical analysis of the resampled series in the following subsection, the choice of  $\gamma = 0.5$  seems to be particularly reasonable when performing the bootstrap procedure. Indeed, in this case, the 5000 series simulated to assess the performance of our bootstrapping procedure reflect in a rather satisfactory manner the statistical properties of the original sample. As for the diversity among the generated series, when the value of  $\gamma = 0.5$  is adopted, a very low probability of duplication over the 5000 tested cases is observed. Then, for each dataset, we selected the optimal partition corresponding to  $\gamma = 0.5$  for the bootstrap procedure. We denote them by  $\pi_S^{milp}$  and  $\pi_G^{milp}$ , for the Spanish and German dataset, respectively, while  $M_S^{milp}$  and  $M_G^{milp}$  indicate the corresponding 2<sup>nd</sup> order aggregated transition probability matrices<sup>9</sup>. The two singleton partitions obtained for  $\gamma = 0$  and  $q = m$  are denoted by  $\pi_S$  and  $\pi_G$ , respectively.

Partition  $\pi_S^{milp}$  consists of 72 classes (19 of which correspond to deterministic 2-states, and one contains all the non observed 2-states). Partition  $\pi_G^{milp}$  has 43 classes (19 of which correspond to deterministic 2-states, and one contains all the non observed 2-states).

### 3.4 Statistical analysis of the resampled series

For each market (Spain and Germany), we evaluate the performance of our bootstrap procedure in two different scenarios:

<sup>8</sup>www.ampl.com.

<sup>9</sup>These matrices are available at the web address <http://chiara.eco.unibs.it/~pelizcri/Partitioning.html>.

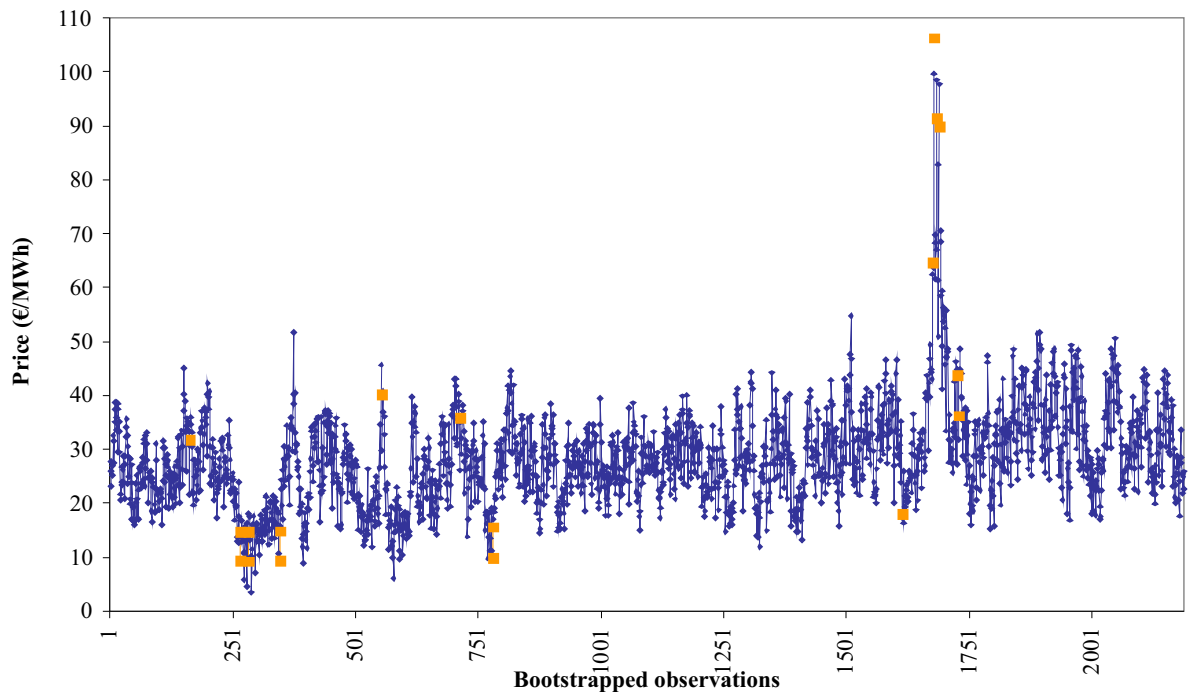


Figure 5: Bootstrapped Spanish electricity prices. The thin mark indicates that the selected value is the ending point of an observed probabilistic 2-state, the thick mark indicates that the bootstrapped values are the last points of observed deterministic 2-states.

- i. a “conservative” scenario, where we consider the two partitions of singletons  $\pi_S$  and  $\pi_G$ , which represent the benchmark situation of minimum multiplicity and maximum similarity for the generation of the bootstrapped series;
- ii. a “progressive” scenario, where we consider the two partitions  $\pi_S^{milp}$  and  $\pi_G^{milp}$  ( $\gamma = 0.5$ ), which are expected to generate higher diversification and lower similarity among the bootstrapped series than  $\pi_S$  and  $\pi_G$ .

We generated 4 sets of 5000 bootstrapped series (one set for each partition) with length  $\ell = 2188$  for the Spanish case and  $\ell = 2515$  for the German one. In the following, we analyze the statistical properties of the bootstrapped series in order to compare them with the ones of the original series.

Before introducing the indices, we want to give an idea of how the bootstrapped series look like. Figures 5 and 6 report two bootstrapped series, one for the Spanish market and one for the German instance, based on partitions  $\pi_S^{milp}$  and  $\pi_G^{milp}$ . Here, the bootstrapped series include the exponential trend and the exponential weekly seasonality initially removed from the original samples (see Appendix A). Each value of the series is classified as deterministic (thick mark) or probabilistic (thin mark).

We can make the following considerations:

- both the bootstrapped series in Figures 5 and 6 reproduce the spikes observed in the original series (see, Figures 1 and 2);
- also the normal trading regime appears satisfactorily reproduced. Indeed, the two series take values in ranges strongly overlapping those of the original one;
- weekly seasonality is clearly distinguishable, as well as, a slightly positive trend;
- the frequency of deterministic values is 1% for both the Spanish and German cases, similarly to the values observed in the original series. The bootstrap procedure probabilistically reproduces sequences of the original series and occasionally such segments are interleaved by deterministically



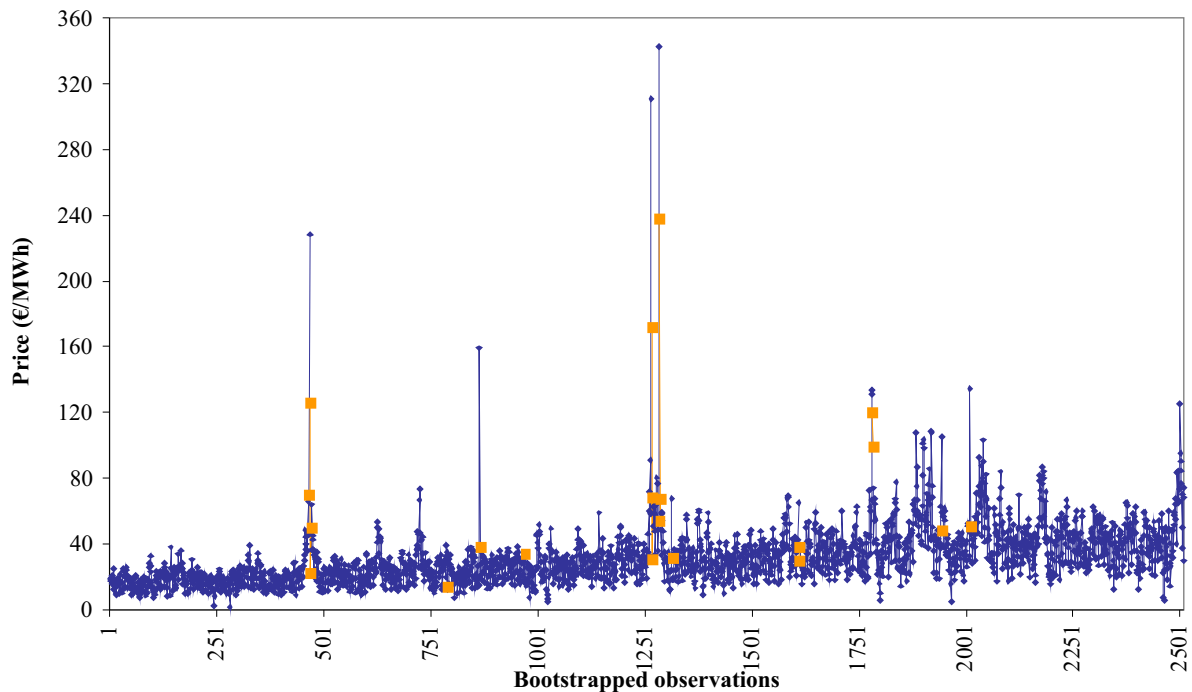


Figure 6: Bootstrapped German electricity prices. The thin mark indicates that the selected value is the ending point of an observed probabilistic 2-state, the thick mark indicates that the bootstrapped values are the last points of observed deterministic 2-states.

chosen values. Let us observe at this point the key advantage of the Markov chain bootstrapping method: depending on the different probability distributions associated to each conditioning event, the resampling method switches from deterministic (e.g., in the case of spikes) to highly unpredictable.

To evaluate more rigorously the statistical properties of the bootstrapped series with respect to their original counterparts, for each bootstrapped series, we calculated the following statistics:

1. average
2. standard deviation
3. skewness
4. kurtosis
5. minimum
6. maximum
7. autocorrelation at lag  $k$ ,  $k = 1, \dots, 8$
8. slope of a linear regression model,  $b$ , with  $\hat{x}_j = a + b \cdot j + \varepsilon_j$ ,  $j = 1, \dots, \ell$ .

The statistics 1. – 6. are concerned with the distribution of prices, while 7. and 8. are more concerned with the dynamic structure of the series. The autocorrelations at lags  $k = 3$  to  $k = 8$  are observed to check if the similarity between the original and the bootstrapped series is kept beyond the order  $k = 2$  used to define the driving process.

For the distribution of each of the above statistics 1. – 8., Tables 2 and 3 report (for Spain and Germany,

Table 2: Percentiles of the distributions of some indices computed over the 5000 bootstrapped series of Spain for two scenarios: the conservative scenario (partition  $\pi_S$ ) and the progressive scenario (partition  $\pi_S^{milp}$ ).

Index	Spain - $\pi_S$				Spain - $\pi_S^{milp}$			
	5 <sup>th</sup>	95 <sup>th</sup>	Value	Percentile rank	5 <sup>th</sup>	95 <sup>th</sup>	Value	Percentile rank
	percen- tile	percen- tile	of original series	of original series value	percen- tile	percen- tile	of original series	of original series value
average	28.318	31.975	29.688	46	28.080	31.541	29.688	56
standard dev.	7.796	12.538	9.562	62	7.788	12.318	9.562	65
skewness	0.211	2.126	1.383	67	0.214	2.099	1.383	67
kurtosis	-0.127	8.590	5.092	70	-0.150	8.630	5.092	69
minimum	3.599	9.466	5.469	65	3.573	9.303	5.469	69
maximum	58.261	110.729	103.758	80	58.020	110.819	103.758	81
aut. at lag 1	0.748	0.868	0.818	66	0.752	0.868	0.818	65
aut. at lag 2	0.613	0.795	0.705	59	0.617	0.794	0.705	59
aut. at lag 3	0.539	0.764	0.680	70	0.543	0.762	0.680	72
aut. at lag 4	0.485	0.736	0.667	78	0.487	0.733	0.667	80
aut. at lag 5	0.442	0.715	0.661	84	0.444	0.714	0.661	85
aut. at lag 6	0.472	0.729	0.721	93	0.470	0.727	0.721	94
aut. at lag 7	0.526	0.755	0.802	99	0.523	0.753	0.802	99
aut. at lag 8	0.391	0.685	0.683	94	0.387	0.680	0.683	95
lin. regr. slope	0.001	0.006	0.004	69	0.001	0.006	0.004	69

respectively) the 5<sup>th</sup> and the 95<sup>th</sup> percentiles, together with the corresponding value for the original series. To evaluate these distributions, we also report the *percentile rank*, i.e., the percentage of cases in which the a statistic is smaller than or equal to the original observed one.

We can make the following observations:

- i. For all the four partitions  $\pi_S$ ,  $\pi_S^{milp}$ ,  $\pi_G$ , and  $\pi_G^{milp}$ , all the statistics computed for the original series take values in between the two above percentiles. This is true also for the autocorrelations at lags  $k = 3$  to  $k = 8$ , with the exception of the autocorrelations at lag  $k = 7$  for the Spanish case.
- ii. The percentile ranks are more fluctuating in the Spanish scenarios than in the German cases: the lowest percentage for Spain is 46% (see, row “average” of partition  $\pi_S$  in Table 2), while it is 51% for Germany (see, row “average” of partition  $\pi_G^{milp}$  in Table 3). The highest percentages in the Spanish and German cases are 99% and 94%, respectively. In general, though, the differences can be considered negligible.
- iii. There seems to be no remarkable difference of values between the 5<sup>th</sup> and the 95<sup>th</sup> percentiles generated with the two partitions  $\pi_S$  and  $\pi_S^{milp}$ , therefore suggesting that the bootstrapped series obtained with the two partitions are rather similar. The same observation applies to the German case.
- iv. Not all the 5000 series generated in each setting showed a spike. This feature reflects the desirable property that a rare event, like a spike, does not appear regularly.

The above results suggest that, both in the Spanish and in the German case, it cannot be confuted that the original series was generated by the same Markov process that produced the 5000 bootstrapped series. In addition, the statistical properties of the series generated by the bootstrapping procedure based on  $\pi_S^{milp}$  and  $\pi_G^{milp}$  do not significantly differ from those of the series generated by the procedure based on  $\pi_S$  and  $\pi_G$ , respectively.

To conclude, our empirical analysis provides evidence that the use of the aggregated transition probability matrix  $M^*$  does not significantly alter the “information” contained in the original matrix  $M$ . Thus  $M^*$

Table 3: Percentiles of the distributions of some indices computed out of the 5000 bootstrapped series of Germany for two scenarios: the conservative scenario (partition  $\pi_G$ ) and the progressive scenario (partition  $\pi_G^{milp}$ ).

Index	Germany - $\pi_G$				Germany - $\pi_G^{milp}$			
	5 <sup>th</sup>	95 <sup>th</sup>	Value	Percentile rank	5 <sup>th</sup>	95 <sup>th</sup>	Value	Percentile rank
	percen- tile	percen- tile	of original series	of original series value	percen- tile	percen- tile	of original series	of original series value
average	30.683	33.497	32.230	60	30.940	33.717	32.230	51
standard dev.	14.017	21.117	18.396	75	14.141	20.925	18.396	77
skewness	1.308	8.187	3.973	62	1.364	7.803	3.973	66
kurtosis	3.457	145.269	35.562	56	3.751	135.671	35.562	61
minimum	1.451	4.051	3.117	81	1.570	4.533	3.117	71
maximum	132.782	493.512	301.542	60	136.361	486.727	301.542	65
aut. at lag 1	0.635	0.772	0.717	60	0.633	0.772	0.717	62
aut. at lag 2	0.428	0.628	0.572	67	0.430	0.623	0.572	70
aut. at lag 3	0.370	0.550	0.507	76	0.368	0.545	0.507	80
aut. at lag 4	0.335	0.516	0.477	79	0.333	0.511	0.477	83
aut. at lag 5	0.339	0.522	0.487	81	0.338	0.515	0.487	85
aut. at lag 6	0.394	0.597	0.586	92	0.395	0.589	0.586	94
aut. at lag 7	0.442	0.673	0.644	84	0.439	0.664	0.644	89
aut. at lag 8	0.355	0.558	0.544	91	0.353	0.549	0.544	93
lin. regr. slope	0.010	0.014	0.013	62	0.010	0.014	0.013	61

can be adopted in the bootstrapping procedure to generate resampled series with the same characteristic features of the original one.

## 4 Conclusions

In this paper we adopted Markov chain theory for resampling series observed from a continuous process. The novel aspect of our work is that for the first time a Mixed Integer Linear Programming model is introduced to formulate and solve the problem of optimally clustering the rows of the estimated transition probability matrix of a Markov chain. Even if it is well known that this problem is computationally hard, in our application on a medium size real-life case, the problem was solved exactly within reasonable computational times. With this approach, also the purpose of simplifying the original transition probability matrix, and yet preserving the law driving a process, can be considered satisfactorily fulfilled (even in the hard case of electricity prices). However, for large scale partitioning problems (i.e., when the transition probability matrix has a very high number of rows), a heuristic approach is still advised. The development of *ad hoc* heuristics for this kind of problems is one of our future lines of research on the subject.

Another interesting aspect of the partitioning problem is the possibility of introducing new constraints for modeling particular relations between the rows of the transition probability matrix. Such an approach aims at reducing the number of feasible partitions and, therefore, it may be a viable strategy to reduce the computational effort for solving our MILP model in the general case.

## Appendix A - Trend and weekly seasonality removal

The estimation of the exponential trend and weekly seasonality is based on the following model:

$$e_t^{(c)} = \exp(rt + \eta_1 \mathbb{I}_1(t) + \eta_2 \mathbb{I}_2(t) + \eta_3 \mathbb{I}_3(t) + \eta_4 \mathbb{I}_4(t) + \eta_5 \mathbb{I}_5(t) + \eta_6 \mathbb{I}_6(t) + \eta_7 \mathbb{I}_7(t) + \varepsilon_t), \quad (13)$$

where  $e_t^{(c)}$  are the raw original prices,  $\mathbb{I}_j(t)$  is the dummy variable signalling whether  $t$  is the  $j$ th day of the week, with  $j = 1, \dots, 7$ ,  $r$  is the growth rate,  $\eta_j$  is the coefficient of dummy variable  $\mathbb{I}_j(t)$ , with

$j = 1, \dots, 7$ , and  $\varepsilon_t$  are the errors. If we take the natural logarithm on both sides of Formula (13), we obtain the following formula:

$$z_t = rt + \eta_1 \mathbb{I}_1(t) + \eta_2 \mathbb{I}_2(t) + \eta_3 \mathbb{I}_3(t) + \eta_4 \mathbb{I}_4(t) + \eta_5 \mathbb{I}_5(t) + \eta_6 \mathbb{I}_6(t) + \eta_7 \mathbb{I}_7(t) + \varepsilon_t,$$

where  $z_t = \ln e_t^{(c)}$ .

For estimation purposes, we assume that the usual hypotheses of linear regression on the errors  $\varepsilon_t$  hold. We obtain the OLS estimates of  $r$  and  $\eta_j$ ,  $j = 1, \dots, 7$ , and they are significant at a level of 5% (see Table 4).

Table 4: Coefficients estimates of an exponential regression model of trend and weekly seasonality applied to the series of electricity prices of Spain and Germany.

Coefficient estimate	Spain	Germany
$\hat{r}$	0.0001161214	0.0003865792
$\hat{\eta}_1$	2.9922407785	2.4365264186
$\hat{\eta}_2$	3.2673051542	2.9583694626
$\hat{\eta}_3$	3.2697769413	3.0280264215
$\hat{\eta}_4$	3.2754546141	3.0209612161
$\hat{\eta}_5$	3.2888136542	3.0002419013
$\hat{\eta}_6$	3.2762719043	2.9386885451
$\hat{\eta}_7$	3.1382584293	2.7011989900

To the purpose of removing the exponential trend and the exponential weekly seasonality from our original series, we define the series of prices  $e(T) = (e_1, \dots, e_t, \dots, e_T)$ , where:

$$e_t = \exp[z_t - (\hat{r}t + \hat{\eta}_1 \mathbb{I}_1(t) + \hat{\eta}_2 \mathbb{I}_2(t) + \hat{\eta}_3 \mathbb{I}_3(t) + \hat{\eta}_4 \mathbb{I}_4(t) + \hat{\eta}_5 \mathbb{I}_5(t) + \hat{\eta}_6 \mathbb{I}_6(t) + \hat{\eta}_7 \mathbb{I}_7(t))], t = 1, \dots, T.$$

Set  $e(T)$  is an input of the bootstrap procedure, while the output is the bootstrapped series  $x(\ell) = (x_1, \dots, x_\ell)$ . To re-introduce the exponential trend and the exponential weekly seasonality in  $x(\ell)$ , we multiply each point  $x_j$  by  $e^{(\hat{r}j + \hat{\eta}_1 \mathbb{I}_1(j) + \hat{\eta}_2 \mathbb{I}_2(j) + \hat{\eta}_3 \mathbb{I}_3(j) + \hat{\eta}_4 \mathbb{I}_4(j) + \hat{\eta}_5 \mathbb{I}_5(j) + \hat{\eta}_6 \mathbb{I}_6(j) + \hat{\eta}_7 \mathbb{I}_7(j))}$ ,  $j = 1, \dots, \ell$ .

## Appendix B - Initial states, or intervals

Table 5 reports the 12 intervals of the initial partition of the support  $[\alpha, \beta]$  of the series of Spain and Germany after removal of trend and weekly seasonality.

## References

- [1] S. Anatolyev and A. Vasnev, Markov Chain Approximation in Bootstrapping Autoregressions, *Economics Bulletin*, 3(19) 2002, 1-8.
- [2] S. Anily and A. Federgruen, Structured Partitioning Problems, *Operations Research*, 39(1) 1991, 130-149.
- [3] W. Brock, J. Lakonishok and B. LeBaron, Simple Technical Trading Rules and the Stochastic Properties of Stock Returns, *The Journal of Finance*, 47(5) 1992, 1731-1764.
- [4] P. Bühlmann and A. J. Wyner, Variable Length Markov Chains, *The Annals of Statistics*, 27(2) 1999, 480-513.
- [5] P. Bühlmann, Bootstraps for Time Series, *Statistical Science*, 17(1) 2002, 52-72.

Table 5: Elements of the initial partition of the support of the detrended series of electricity prices of Spain and Germany. In both cases, the upper limit of the interval 12, i.e.,  $\beta$ , represents a “high enough” value such that no price can be reasonably thought to be greater than it. For example, in our experiment we take  $\beta = 1,000,000$ .

Interval label	Interval of prices	
	Spain	Germany
$\alpha_1$	[0, 0.51)	[0, 0.59)
$\alpha_2$	[0.51, 0.66)	[0.59, 0.83)
$\alpha_3$	[0.66, 0.76)	[0.83, 1.04)
$\alpha_4$	[0.76, 0.87)	[1.04, 1.20)
$\alpha_5$	[0.87, 0.96)	[1.20, 1.43)
$\alpha_6$	[0.96, 1.09)	[1.43, 1.70)
$\alpha_7$	[1.09, 1.18)	[1.70, 2.20)
$\alpha_8$	[1.18, 1.35)	[2.20, 3.09)
$\alpha_9$	[1.35, 1.54)	[3.09, 4.02)
$\alpha_{10}$	[1.54, 1.92)	[4.02, 5.93)
$\alpha_{11}$	[1.92, 2.29)	[5.93, 8.58)
$\alpha_{12}$	[2.29, $\beta$ ]	[8.58, $\beta$ ]

- [6] D. W. Bunn, *Modelling Prices in Competitive Electricity Markets*, John Wiley & Sons, Chichester UK, 2004.
- [7] R. Cerqueti, P. Falbo, G. Guastaroba and C. Pelizzari, A Tabu Search Heuristic Procedure in Markov Chain Bootstrapping, *Rapporti di Ricerca del Dipartimento Metodi Quantitativi dell’Università degli Studi di Brescia*, 2011.
- [8] R. Cerqueti, P. Falbo and C. Pelizzari, Optimal Dimension of Transition Probability Matrices for Markov Chain Bootstrapping, *Quaderni del Dipartimento di Istituzioni Economiche e Finanziarie dell’Università degli Studi di Macerata*, 53/2009.
- [9] Wai-Ki Ching, M. K. Ng and E. S. Fung, Higher-Order Multivariate Markov Chains and Their Applications, *Linear Algebra and Its Applications*, 428(2-3) 2008, 492-507.
- [10] P. J. Courtois, *Decomposability: Queuing and Computer System Applications*, Academic Press, New York, 1977.
- [11] B. Efron, Bootstrap Methods: Another Look at the Jackknife, *The Annals of Statistics*, 7(1) 1979, 1-26.
- [12] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall, New York, 1993.
- [13] E. W. Forgy, Cluster Analysis of Multivariate Data: Efficiency Vs. Interpretability of Classification - Abstract, *Biometrics*, 21(3) 1965, 768-769.
- [14] D. Freedman, On Bootstrapping Two-Stage Least-Squares Estimates in Stationary Linear Models, *The Annals of Statistics*, 12(3) 1984, 827-842.
- [15] D. A. Freedman and S. C. Peters, Bootstrapping a Regression Equation: Some Empirical Results, *Journal of the American Statistical Association*, 79(385) 1984, 97-106.
- [16] M. R. Garey and D. S. Johnson, Two-Processor Scheduling with Start-Times and Deadlines, *SIAM Journal on Computing*, 6(3) 1977, 416-426.

- [17] O. Jeanne and P. Masson, Currency Crises, Sunspots and Markov-Switching Regimes, *Journal of International Economics*, 50(2) 2000, 327-350.
- [18] J. D. Hamilton, Specification Testing in Markov-Switching Time-Series Models, *Journal of Econometrics*, 70(1) 1996, 127-157.
- [19] J. D. Hamilton, What's Real About the Business Cycle?, *Federal Reserve Bank of St. Louis Review*, 87(4) 2005, 435-452.
- [20] R. Huisman and R. Mahieu, Regime Jumps in Electricity Prices, *Energy Economics*, 25(5) 2003, 425-434.
- [21] J. B. MacQueen, Some Methods for Classification and Analysis of Multivariate Observations, Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability. L. M. LeCam and J. Neyman (Eds.), University of California Press, Berkeley, CA, 1 1967, 281-297.
- [22] C. Patrick, H. Ma, C. Keith, C. Chan, Xin Yao and D. K. Y. Chiu, An Evolutionary Clustering Algorithm for Gene Expression Microarray Data Analysis, *IEEE Transactions on Evolutionary Computation*, 10(3) 2006, 296-314.
- [23] F. Ricca and B. Simeone, Local Search Algorithms for Political Districting, *European Journal of Operational Research*, 189(3) 2008, 1409-1426.
- [24] F. Ricca, A. Scozzari and B. Simeone, Weighted Voronoi Region Algorithms for Political Districting, *Mathematical and Computer Modelling* 48(9-10) 2008, 1468-1477.
- [25] W. M. Spears, A Compression Algorithm for Probability Transition Matrices, *SIAM Journal of Matrix Analysis and Applications*, 20 1998, 60-77.
- [26] R. Sullivan, A. Timmermann and H. White, Data-Snooping, Technical Trading Rule Performance, and the Bootstrap, *The Journal of Finance*, 54(5) 1999, 1647-1691.
- [27] J. Trejos, A. Murillo and E. Piza, Clustering by Ant Colony Optimization, Classification, Clustering, and Data Mining Applications, Proceedings of the Meeting of the International Federation of Classification Societies (IFCS), Illinois Institute of Technology, Chicago. D. Banks, L. House, F. R. McMorris, P. Arabie and W. Gaul (Eds.), Springer-Verlag, Berlin, Germany, 2004, 25-32.
- [28] J. H. Ward, Hierarchical Grouping to Optimize an Objective Function, *Journal of the American Statistical Association*, 58(301) 1963, 236-244.
- [29] R. Weron and M. Bierbrauer and S. Trck, Modeling Electricity Prices: Jump Diffusion and Regime Switching, *Physica A: Statistical Mechanics and its Applications*, 336(1-2) 2004, 39-48.
- [30] R. Weron, *Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach*, John Wiley & Sons, Chichester, UK, 2006.
- [31] Z. Wu, R. Leahy, An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11) 1993, 1101-1113.
- [32] J. Zhu, J. Hong and J. G. Hughes, Using Markov Chains for Link Prediction in Adaptive Web Sites *Lecture Notes in Computer Science*, SoftWare 2002: Computing in an Imperfect World. D. Bustard, W. Liu and R. Sterritt (Eds.), Springer-Verlag, Berlin/Heidelberg, 2002, 55-66.