Louvain School
of Management

Working paper
Using self-organizing maps to adjust intra-day seasonality
149/05

Walid Ben Omrane, Eric de Bodt

Academic Year 2005 - 2006

Louvain School of Management

Using self-organizing maps to adjust intra-day seasonality

Walid Ben Omrane[1] Eric de Bodt[2,3]

September 26, 2005

Abstract

The existence of an intra-day seasonality component within financial market variables (volatility, volume, activity,...), has been highlighted in many previous works. To adjust raw data from their cyclical component, many studies start by implementing the intra-daily average observations model (IAOM) and/or some smoothing techniques (e.g. the kernel method) in order to remove the day of the week effect. When seasonality involves only a deterministic component, IAOM method succeed in estimating periodicity almost without estimation error. However, when seasonality contains both deterministic and stochastic components (e.g. closed days), we show that either the IAOM or the kernel method fail to capture it. We introduce the use of the self-organizing maps (SOM) as a solution. SOM are based on neural network learning and nonlinear projections. Their flexibility allows capturing seasonality even in the presence of stochastic cycles.

Keywords: self-organizing maps, currency market, intra-day seasonality, high frequency data.

JEL Classification : F31, C22

[1]Department of Business Administration (IAG), Finance Unit, Catholic University of Louvain, Place des Doyens 1, 1348 Louvain-la-Neuve, Belgium, e-mail: benomrane@fin.ucl.ac.be. Tel: +32 10 47 84 49, Fax: +32 10 47 83 24.

[2]Department of Business Administration (IAG), Finance Unit, Catholic University of Louvain, Place des Doyens 1, 1348 Louvain-la-Neuve, Belgium, e-mail: debodt@fin.ucl.ac.be. Tel: +32 10 47 84 47, Fax: +32 10 47 83 24.

[3]ESA, University of Lille 2, Place Déliot, BP 381, Lille F-59020, France.

The authors would like to thank Luc Bauwens, Marie Cottrell, Christian Gouriéroux, Michel Verleysen and participants at the 11^{th} ACSEG International Conference in Lille1, France, for helpful discussions and suggestions. Work supported in part by the European Community Human Potential Programme under contract HPRN-CT-2002-00232, Microstructure of Financial Markets in Europe. The usual disclaimers apply.

1 Introduction

Evidences of intra-daily seasonality in financial market behaviors has been highlighted in many prior studies. Some recent references include Degennaro and Shrieves (1997), Andersen and Bollerslev (1998), Melvin and Yin (2000), Cai, Cheung, Lee, and Melvin (2001), Bauwens, Ben Omrane, and Giot (2003) and Ben Omrane and Heinen (2004). These works illustrate the existence of seasonality in many microstructure variables (e.g. FOREX volatility and quoting activity). Two categories of methods are most often used in order to remove this seasonality. Some studies like Degennaro and Shrieves (1997), Andersen and Bollerslev (1998), Cai, Cheung, Lee, and Melvin (2001) and Ben Omrane and Heinen (2004) adopt a linear projection technique. They regress variables (affected by the seasonal component) on a set of dummy variables (or flexible Fourier form) in order to capture intra-day cycles. Other authors adjust raw data from seasonality using a direct correction factor, obtained by intra-daily average (Dacorogna, Muller, Nagler, Olsen, and Pictet, 1993, Eddelbuttel and McCurdy, 1998, Melvin and Yin, 2000, and Bauwens, Ben Omrane, and Giot, 2003) or a smoothing kernel (Engle and Russell, 1998, Bauwens and Giot, 2000 and Veredas, Rodriguez-Poo, and Espasa, 2002).

This work builds on the previous literature to explore the limits of the classical approaches and to introduce a solution in the case of stochastic cycles. We show that the more the raw data involves a deterministic seasonality, the more the classical methods, particularly the intra-daily average observations model, succeed in estimating the cycles. However, in the presence of stochastic cycles (or the combination of deterministic and stochastic cycles), such as the ones generated by closed days (among others), classical methods reveal their limits. We introduce a method based on the self-organizing maps algorithm (Kohonen (1995)). The self-organizing maps (SOM) allows capturing both deterministic and stochastic cyclical components and purging endogenous variables from the seasonal component.

Our evidences are based on both Monte Carlo simulations and an application to a real data set (taken from the foreign exchange (FX) market). Our Monte Carlo simulations adopt a five-step framework. We begin by generating an auto-regressive process. We then simulate either only a deterministic seasonality, or both deterministic and stochastic cycles which we add to the auto-regressive variable. After that we deseasonalize the endogenous variable, using the three methods cited above. We finally re-estimate the process coefficients on the deseasonalized data series. Better is the deseasonalization, closer should be the estimated coefficient to the simulated one, and lower should be the root mean square error (RMSE). This allows use comparing the performance of the above cited methods in a controlled setup.

Our empirical evidence use a high frequency data set of 5-minute regularly time-spaced Euro/Dollar quotes. The time period ranges from May 15, 2001 through May 15, 2002. Our results confirms

that in the presence of both deterministic and stochastic cycles, the SOM method reveals itself more powerful to neutralize the seasonality than both the intra-daily average observations model (IAOM) and the Nadaraya-Watson kernel smoothing method. We base our comparison on an analysis of the autoregressive correlation function (ACF) of the deseasonalized variables. In such a way, the quality of the adjustments is inferred from the persistence of the cycles in the ACF. Our results are consistent with the Monte Carlo simulation ones, knowing that our financial data sample presents some stochastic perturbations of the cyclical component due, amongst others, to the presence of five closed days.

This paper is divided into six sections. In Section 2 we present a brief review of literature related to intra-day seasonality (focusing on the FX market as we use it as empirical setting in Section 5). We detail our deseasonalization methods in Section 3. We present the Monte Carlo simulation in Section 4. We show the empirical evidences in Section 5. We finally conclude.

2 The Foreign Exchange Seasonality

A large segment of the foreign exchange microstructure literature documents that the market opening/closing, news announcements and days of the week lead to significant cyclical factors into many microstructure variables (e.g. volatility and quoting activity (Bollerslev and Domowitz, 1993, Andersen and Bollerslev 1996, Degennaro and Shrieves, 1997, Melvin and Yin, 2000, 2003 and Ben Omrane and Heinen, 2004)). A typical case is highlighted in Andersen and Bollerslev (1998) and Bauwens, Ben Omrane, and Giot (2003). These authors show that scheduled news announcements have a seasonal impact on volatility. These news events exhibit both a cyclical and a stochastic component, the latter being the news content not fully anticipated by the market participants. However, the cyclical news component could itself be either deterministic or stochastic, since the hour of some announcements change from a week to another.

A number of methods have been recently put forward in the literature to capture these cyclical behaviors into high frequency data. Melvin and Yin (2000) and Bauwens, Ben Omrane, and Giot (2003), amongst others, use the intra-daily average observations model (IAOM)[1] to adjust volatility and quoting activity variables from seasonality. They divide returns (quoting activity) by the square root of the cross sectional average volatility (cross sectional average quoting activity) to clean the series from its cyclical components (see Section 3.2). The more the data involve a deterministic seasonality,[2] the more IAOM is successful in purging the seasonality. Degennaro and Shrieves (1997) use, in addition, dummy variables to pick the "hour of the day" cyclical effects (but they do not discriminate between different days of the week). On the other hand,

[1] This method is equivalent to the one based on hourly dummy variables.

[2] Deterministic seasonality corresponds to continued cycles without gaps or discontinuity generated, for instance, by closed days.

Andersen and Bollerslev (1998) and Bauwens, Ben Omrane, and Giot (2003) show that workdays are characterized by specific cyclical behaviors. To deseasonalize volatility, they allow therefore for a specific seasonality for each day of the week (but they assume that the day of the week effect is constant from week to week).

Still to capture intra-day cycles, Andersen and Bollerslev (1998), Cai, Cheung, Lee, and Melvin (2001) and Ben Omrane and Heinen (2004) use the flexible Fourier form (a sum of sinusoids). Dacorogna, Muller, Nagler, Olsen, and Pictet (1993) and Eddelbuttel and McCurdy (1998) deseasonalize volatility using an adjustment factor. This factor is proportional to the (mean) absolute value of the returns over a time interval divided by the size of the time interval. Engle and Russell (1998) and Bauwens and Giot (2000) adjust duration variables from seasonality using the cubic splines technique. Veredas, Rodriguez-Poo, and Espasa (2002) adopt the kernel estimator to adjust duration and show that their method is more relevant than the cubic splines.

To sum up, there are two broad categories of seasonality adjustment methods used in the literature. The first one is a one-step procedure and it consists in removing seasonality through a regression. Seasonality is captured through some added variables (like dummies or the flexible Fourier form). The second category is a two-step procedure. Before implementing the regression, we begin in a first step by adjusting the raw data from seasonality. The next step consists in regressing the adjusted variable on the set of exogenous ones.

In this work we focus on the seasonality removal part of the procedure. We compare both the IAOM and the kernel methods to a new type of algorithm: the self-organizing maps (SOM) introduced by Kohonen (1995). This algorithm has lead to many applications in physics and engineering and also in finance (de Bodt, Cottrell, Henrion, and Van Wijmeersch, 1998, de Bodt, Lendasse, Cardon, and Verleysen, 2003 and DeBoeck and Kohonen, 1998, amongst others). The SOM model is based on neural network learning and nonlinear discrete projection (see Section 3.1). As we show it in the sequel, the SOM algorithm will allow us to deal with the stochastic component of the seasonality.

3 Deseasonalization Methods

We present in the following section the three seasonality identification methods. We start with a presentation of the SOM algorithm as its usage is the key contribution of this study. We then review the well known IAOM and kernel smoothing approaches. For a more convenient presentation of the different method details, we take some examples picked out from both the simulated or the real data series.

3.1 The Self-Organizing Maps Model (*SOM(p,q)*)

The self-organizing maps (SOM) introduced by Kohonen (1995) can be considered as a method of data analysis which allows, through a (discrete) projection, to reduce the dimension of the data space (as principle component analysis methods do). Simultaneously it allows, through vector quantization, to summarize the data projected in specific mean profiles. The projection step is carried out on a discrete data space.

Before turning to a more formal presentation of the SOM algorithm, we introduce it starting from an example. Imagine that we are faced to a two dimensional data matrix such as the one presented at Table 1. There are 21 observations and, for each observation, two measures has been taken (e.g., the size and the weight). Figure 1 - Panel A shows the two-dimensional input data space (each observation is represented by a dot). Three natural clusters clearly emerge. The two situated at left are closer to each other than to the one situated at the right of the chart. We use a self-organizing map (SOM) both to capture the proximity relations among clusters and to summarize (to quantify) the information contained in each cluster. Our map is introduced at Figure 1 - Panel B. It is a SOM(2,2) - this is to say composed of two rows and two columns (four nodes). Each node is identified by its location in the map (the row and column indices). To each node is associated a vector of coordinates. This vector defines the location of the node in the input space. Figure 1 - Panel C displays the locations of the four nodes after random initialization of their coordinates' vectors. In technical terms, the map is said to be folded: the proximity relations among the nodes into the map do not respect the proximity relations into input space. In the input space, the node (1,1) is closer to the node (2,2) than the node (1,2) while it is not the case in the map. Moreover, the nodes' locations maintain no relation with the clusters of data. The SOM algorithm is the numerical procedure by which the map will be unfolded and displaced toward the data clusters (see Figure 1 - Panel D). At that time, if everything goes right, the neighborhood relations in the map will correspond to the ones observed in the input space. The node coordinates' vectors will represent homogeneous clusters of data (as it is the case here for nodes (1,1), (2,2) and (2,1) but not (1,2)). Note that in the present case, in order to allow a visual representation of the learning process, we have realized a projection of the two-dimensional input space onto a two-dimensional map. Most frequently, in real applications, the input space dimensions is far higher and the two dimensional map provides a convenient way to observe the neighborhood relation among cluster of individuals. Nothing forbids the use of higher dimensional maps but, in that case, visual representation becomes difficult (if not impossible).

In more formal terms, SOM defines a mapping from the input data space Ω, onto a K-dimensional array of output nodes. In order to visualize the outputs, K has not to go beyond two (a grid). Let $x\,(\,represents\,one\,observation\,) \in \Omega$ be a stochastic data vector. A vector

quantization φ is an application from the continuous space Ω, endowed by the same probability density function $f(x)$, to a finite subset F composed by n nodes $m_1, \ldots, m_n$. These nodes, which are located at a specific location on the map, are associated to a coordinates vector, which will allow in our case, capturing the common cyclical component of the observations associated to a specific node. After learning, the position of a node is a result of the neighborhood structure of the data into the input space. The SOM algorithm is defined as follow:

- The structure of the map is first defined (number of rows (p) and columns (q)),
- The coordinates vectors of the nodes $m_1, \ldots, m_n$ are randomly initialized (in the input space),
- Each node occupies a specific location in the map,
- At each iteration t of the algorithm:
  - An observation x is randomly drawn according to the density $f(x)$,
  - The winning node $m_{k^*,t}$ is identified by minimizing the classical Euclidean norm:

$$\|x_i - m_{k^*,t}\| = \min_k \|x_i - m_{k,t}\| \tag{1}$$

  - The class m_{k^*} and its neighbors in the map are updated by

$$m_{k^*,t+1} = m_{k^*,t} + \varepsilon_t(x_i - m_{k^*,t}), \tag{2}$$

    where ε_t is an adaptation parameter which satisfies the Robbins and Monro (1951) conditions ($\sum \varepsilon_t = \infty$ and $\sum \varepsilon_t^2 < \infty$). Note that the set of the neighbors adapted at each iteration can either be kept fixed or be progressively decreased through the learning.

The learning process combines a projection and a quantization. Nodes begin to be distant from each other and then converge gradually to the barycenter of clusters of observations. At the end of the learning process, their coordinate vectors represent the "average individual" of a given cluster of observation. The adaptation parameter, ε_t (called also the learning coefficient), drops progressively. At the starting of the learning, nodes are moved by a large steps in order to bring them closer to their convergence zone and then, their position are progressively computed with more precision.

Once learning is achieved, each observation i is affected to its winning node $m_{k^*,t}$, identified by its map coordinates. This correspondence is a kind of projection on a discrete subspace.

In our empirical study (Section 5), Ω is the data matrix containing in rows the number of open days along the year (258) and in columns, the number of the five-minutes observations

(288). The seasonality is captured by the value of the node coordinates vector after learning. The deseasonalization is done through two step process. Each observation is affected to a winning node; then, to adjust the observation from seasonality we divide it by (or we withdraw it from) the corresponding winning class mean profile. We implement the division or the substraction according to how the cyclical component is involved into the raw data. For instance, volatility and quoting activity have to be adjusted from seasonality through the division operator. However, both simulated AR(P) processes ($y_t^{'}$ and $z_t^{'}$), computed in the section 4, have to be adjusted from the cyclical component by the substraction operator.

Finally, note that selecting a given map structure (the number of rows (p) and the number of columns (q)) is a trial and error process. As for selecting the right number of lags within an ARMA(p,q) model, a convenient guidance tool is the ACF.

3.2 The Intra-daily Average Observations Model (*IAOM*)

To estimate seasonality, we compute the intra-daily average observations at time n_k of day k (called mv_{n_k}). We divide each day into Q intervals of time. We assume for simplicity that we have exactly S weeks of data. For each interval endpoint per day of the week over the S week period, we have one observation for the random variable, Y. We thus compute in principle Q values mv_{n_k} for each day of the week, that makes a total of W $(5 \times Q)$ values over a week. Formally,

$$mv_{n_k} = \frac{1}{S} \sum_{s=1}^{S} Y_{f(s,k,n_k)}, \tag{3}$$

where

$$f(s,k,n_k) = W\,(s-1) + \sum_{j=1}^{k-1} N_j + n_k, \tag{4}$$

$s = 1, \ldots, S.\ k = 1, \ldots, 5.\ N_1 = N_2 = N_3 = N_4 = N_5 = Q.\ n_1 = 1, \ldots, Q.\ n_2, n_3, n_4$ and n_5 likewise.

To adjust the different variables for seasonality, we implement the same methodology used for the SOM adjustment. We just divide/withdraw them at the endpoint of each five minute interval by/from the corresponding value of the intra-daily average observation. That means, for example, that all quoting activity at 12h on Thursday in the sample are withdrawed/divided by the same value (the average quoting activity at 12h on Thursday).

3.3 The Smoothing Method

It consists in smoothing the raw data using the Nadaraya-Watson kernel estimator and then adjusting each raw observation by the correspondent value on the smooth curve. The adjustment is done as for the SOM and IAOM methods.

The *Nadaraya-Watson* kernel estimator $\hat{Y}_t$ of $Y_(t)$ is:

$$\hat{Y}_t = \frac{\sum_{j=1}^{T} K_h(t - t_j) Y_t}{\sum_{j=1}^{T} K_h(t - t_j)} \,. \tag{5}$$

t is the vector of time, T corresponds to the number of observations, and h is the bandwidth parameter. Choosing the appropriate bandwidth is an important aspect of any local-averaging technique. In our case we select a Gaussian kernel with a bandwidth, h, computed by Silverman (1986):

$$K_h(x) = \frac{1}{h\sqrt{2\pi}} e^{-\frac{x^2}{2h^2}} \tag{6}$$

$$h = \left(\frac{4}{3}\right)^{1/5} \sigma_k \, l^{-1/5}, \tag{7}$$

where σ_k is the standard deviations for the observations.

4 Monte Carlo Simulation

4.1 Simulation Procedure

In order to compare the three seasonality identification methods (IAOM, SOM, NW-kernel) we implement a five-step simulation procedure.

1) We start by generating a P-lag autoregressive process, y_t^*, (AR(P), $P = 1, 5$):

$$y_t^* = \sum_{p=1}^{P} \beta_p y_{t-p}^* + \epsilon_t, \tag{8}$$

where β is equal to 0.95 if $P = 1$, and if $P = 5$, then $\beta_1 = 0.5$, $\beta_2 = 0.09$, $\beta_3 = 0.08$, $\beta_4 = 0.07$, and $\beta_5 = 0.06$. ϵ_t is distributed as a standard Normal.

2) We partition y_t^* by block of Q observations each one representing a day of the week.

3) We simulate a deterministic seasonality $S_{t,i}^{det}$ and we add it to the above AR(P) process, such that:

$$y_{t,i} = y_{t,i}^* + S_{t,i}^{det}. \tag{9}$$

Let $y_{t,i}^*$ represent one such block, where i is an index corresponding respectively to the open days of the week ($i = 1, \ldots, 5$). $S_{t,i}^{det}$ is generated by the following procedure: we divide the block of observations, corresponding to each day of the week, into three time frame (let say the morning, the noon, and the afternoon). Then, we add a defined constant to the AR(P) process depending on the specific time frame in which the observation is located. One set of constants is chosen for

each day of the week, since we generate a deterministic seasonality. In such a way, y_t becomes an autoregressive variable which involves a deterministic seasonality.

To simulate an AR(P) process which contains stochastic seasonality added to the deterministic one, we go through the following procedure:

- We generate an AR(P) process:

$$z_t^* = \sum_{p=1}^{P} \beta_p z_{t-p}^* + \epsilon_t, \tag{10}$$

- We add to this process a deterministic and stochastic seasonality:

$$z_{t,i} = z_{t,i}^* + S_{t,i}^{det} + S_{t,i}^{sto}, \tag{11}$$

$S_{t,i}^{det}$ is generated as described above, and $S_{t,i}^{sto}$ is the stochastic seasonality. The difference between the latter seasonality and the former one consists on the manner of which we add constants to the time frame in the weekdays. In the stochastic seasonality case, days are selected randomly to be subject for added seasonality. Moreover, seasonality changes from a week to another.

4) The fourth step consists in estimating and removing seasonality from the two simulated processes (y_t and z_t) using respectively the IAOM, the NW-kernel and the SOM methods. The deseasonalization methodology consists in using a linear substraction of the estimated seasonality, ϕ_t^{det} and ϕ_t^{sto} respectively from the analysed variables y_t and z_t, such that:

$$y_t^{'} = y_t - \phi_t^{det}, \tag{12}$$

$$z_t^{'} = z_t - \phi_t^{sto}. \tag{13}$$

5) Finally, we estimate both AR(P) processes, based respectively on the deseasonalized variables, using ordinary least square estimation:

$$y_t^{'} = \sum_{p=1}^{P} \beta_p^{'} y_{t-p}^{'} + \epsilon_t^{'}, \tag{14}$$

$$z_t^{'} = \sum_{p=1}^{P} \gamma_p^{'} z_{t-p}^{'} + \nu_t^{'}. \tag{15}$$

The all procedure is iterated 1000 times. To assess the performance in terms of seasonality adjustment of each of the three methods, we compute the root mean square error (RMSE) of the estimated coefficients $\beta_p^{'}$ and $\gamma_p^{'}$ relative to the initially simulated one, β_p. The closer the estimated coefficients to β_p, the lower is the RMSE and better is the seasonality adjustment approach. It is worth pointing out that the SOM algorithm is initialized with the IAOM outputs (which in practice, seems to be a judicious choice).

4.2 Results

Estimation results for the Monte Carlo simulation are presented in Tables 2 and 3. The latter table displays the estimation results for AR(5) process, and the former one presents those of AR(1). The panels A in both tables display the mean, the standard deviation, and the RMSE corresponding to 1000 estimation of the autoregressive coefficient for equation (14) in presence of deterministic seasonality. The panels B illustrates the same results for the stochastic seasonality (equation (15)). The variables, in this case, are deseasonalized from their deterministic and stochastic seasonality. The RMSE in both panels characterize the estimation error generated by the added seasonality. It is the root mean square difference between the simulated coefficients and the recovered ones after adding, estimating and removing seasonality, $\beta^{'}$ and $\gamma^{'}$.

Starting with deterministic seasonality results, the estimated coefficients for the non-deseasonalized autoregressive parameters corresponding to equation (14) and (15) (see the second column of Tables 2 and 3) shows a higher error level in Panel B results than in panel A. The more there is seasonality into the process, the more important is the error in the estimated coefficients. This is the reason why previous studies try to get rid from the cyclical component involved in their microstructure variables.

The IAOM deseasonalization displays interesting results in panel A. The estimated coefficients is very close to simulated ones with an insignificant error equal to 0.01% for AR(1) and around 0.35% for the different coefficients of the AR(5). We conclude that the IAOM method succeeds in capturing almost the whole deterministic seasonality component. The IAOM method can therefore be recommended as an effective tool for seasonality adjustment when the cyclical component is strictly deterministic. This means that the time series should not include gaps due to missing values (due, e.g., to closed days, data recording problems, ...). Panel B presents very different results. In the presence of stochastic cycles, the IAOM method leads to a significant estimation error level. The corresponding RMSE is much higher than the error obtained by estimating the model with deterministic seasonality. When the seasonality involves both deterministic and stochastic elements, the IAOM does in fact not capture the whole cyclical component of the process. When there are good reasons to think that the seasonality could display some stochastic behavior, the IAOM approach should no be used.

The SOM method seems to be far more robust to the presence of stochastic cycles. The panel B of Tables 2 and 3 exhibits, in the forth row, respectively the estimation result for the seasonality adjusted AR(1) and AR(5) processes. The corresponding RMSE is low compared to the IAOM case. Contrary to the IAOM method, SOM(1,5) succeeds in capturing seasonality involving both deterministic and stochastic cycles. Results displayed by panel A, for both tables, show that SOM(1,5) is however less efficient than the IAOM method when seasonality involves

only deterministic cycles. In such a case, the estimation error generated by SOM(1,5) is much higher than the one generated by IAOM method. The choice between the IAOM and SOM depends therefore on the presence of stochastic cycles.

The kernel results displayed in Table 2, Panel A, show that the method captures deterministic seasonality with a low error level. This finding is consistent with previous research which opt for the kernel method as a step in their deseasonalization process in particular when their samples exhibit some deterministic cycles. However, the kernel adjustment is less accurate than IAOM. Nevertheless, Table 3, Panel A, displays a higher RMSE for the kernel method, especially for the first three coefficients of the AR(5).

Panel B results, corresponding to Tables 2 and 3, show that the kernel method generates an estimation error level much higher than the one generated by SOM and IAOM but quite smaller than non-adjusted data.

These findings are consistent with the intuition. By construction, in the case of deterministic seasonality, the IAOM method, built on the computation of the cross sectional means, can easily capture the seasonality. The IAOM algorithm relies indeed on the law of large numbers: the deterministic component estimation amounts to an estimation of the hour by hour cycle expected value by its sample average. The SOM algorithm and kernel methods can as well capture deterministic cycles but much less efficiently than the IAOM. Nonetheless, in case of cycle irregularities (as often observed in financial data), using hour by hour sample average to capture the seasonality becomes problematic. The SOM model goes beyond the limits of IAOM and the kernel models. It estimates, efficiently, the seasonality which contains both deterministic and stochastic cycles.

5 Empirical Evidence

5.1 Data Description

We use in this section the same data set as the one used in Bauwens, Ben Omrane, and Giot (2003). The data chosen are two microstructure variables (volatility and quoting activity) picked out from the currency market. The Euro/Dollar foreign exchange market is a market maker based trading system, where three types of market participants interact around the clock (i.e. in successive time zones): the dealers, the brokers and the customers from which the primary order flow originates. The most active trading centers are New York, London, Frankfurt, Sydney, Tokyo and Hong Kong. A complete description of the FOREX market is given by Lyons (2001).

To compute the returns used to estimate the volatility, we use the Olsen and Associates database made up of 'tick-by-tick' Euro/Dollar quotes for the period ranging from May 15, 2001 to May 15, 2002 (i.e. one year). It is worth pointing out that our sample involves five closed

days.[3] This database includes 6,088,382 observations. As in most empirical studies on FOREX data, these Euro/Dollar quotes are market makers' quotes and not transaction quotes (which are not widely available).[4] More specifically, the database contains the date, the time-of-day time stamped to the second in Greenwich mean time (GMT), the dealer bid and ask quotes, the identification codes for the country, city and market maker bank, and a return code indicating the filter status. According to Dacorogna, Muller, Nagler, Olsen, and Pictet (1993), when trading activity is intense, some quotes are not entered into the electronic system. If traders are too busy or the system is running at full capacity, quotations displayed in the electronic system may lag prices by a few seconds to one or more minutes. We retained only the quotes that have a filter code value greater than 0.85.[5]

From the tick data, we computed mid-quote prices, where the mid-quote is the average of the bid and ask prices. As we use five-minute returns, we have a daily grid of 288 points. At the end of each interval, we use the closest previous and next mid-quotes to compute the relevant price by interpolation. The mid-quotes are weighted by their inverse relative time distance to the interval endpoint. Next, the return at time t is computed as the difference between the logarithms of the interpolated prices at times $t-1$ and t, multiplied by 10,000 to avoid small values. Volatility is computed as the square of returns.

Because of scarce trading activity during the week-end, we excluded all returns computed between Friday 21h30 and Sunday 24h. In addition, we excluded the first return of each Monday and of each day following a closed day (other than week ends) to avoid possible biases due to the lack of activity during the week-ends and closed days. We take into consideration the day-light saving time adjustment to account for the time changes (to winter and summer time) that occurred on October 29, 2001 and March 25, 2002. This concerns GMT hours from 6h until 21h (corresponding to market times in Europe and the USA).

Next to return volatility, a second important variable is quoting activity. FOREX quoting activity, measured by the number of quotes in five minutes time interval, is considered in many papers as a proxy for volatility and in some other studies as a proxy for private information. Adjustments for week-ends and holidays are computed in the same way as for returns. The total

[3]The dates of the closed days are: the 25^{th} and the 26^{th} of December 2001, the first of January 2002, the 18^{th} of April and the first of May 2002.

[4]Danielsson and Payne (2002) show that the statistical properties of 5-minute dollar/DM quotes are similar to those of transaction quotes.

[5]Olsen and Associates recently changed the structure of their HF database. While they provided a 0/1 filter indicator some time ago (for example in the 1993 database), they now provide a continuous indicator that lies between 0 (worst quote quality) and 1 (best quote quality). While a value larger than 0.5 is already deemed acceptable by Olsen and Associates, we choose a 0.85 threshold to have high quality data. We remove however almost no data records (Olsen and Associates already supplied us with data which features a filter value larger than 0.5), as most filter values are very close to 1.

number of observations for volatility and quoting activity is 72,675.

Table 4 presents summary statistics of the Euro/Dollar returns, and quoting activity. The returns mean is almost equal to zero, their distribution has fatter tails than the normal and features a positive skewness coefficient. The quoting activity mean and standard deviation are relatively high. However its distribution is less leptokurtic than returns but much more asymmetric.

5.2 Results

Our empirical results based on FOREX volatility and quoting activity relies on the ACF analysis for the adjusted data. The presence of closed days could generate a discontinuity in the cycles source of the stochastic seasonality in addition to the deterministic one.

Figure 2 illustrates the ACF for both deseasonalized volatility and quoting activity. The seasonality adjustment is done by the IAOM method. It is clear that despite the adjustment, cycles remains in the series particularly in quoting activity. Figure 3 displays the ACF for the same variables, but adjusted by the kernel method. In the case of volatility, the cycles persist but less pronounced than the previous figure. Figure 4 shows an ACF of which the cycles are almost removed. In such a case, volatility is adjusted through a SOM(2,5) and quoting activity by SOM(6,6).

Table 5 shows the mean, the standard deviation and the autocorrelation coefficient (AC) computed respectively at one-day, two days and three-days lags. The idea is to quantify the peaks in the different ACF cycles in order to simplify the comparison. The AC's corresponding to the different adjustment method are consistent with the figures. For instance, adjusted volatility by IAOM presents higher AC's than the one adjusted by the kernel.

These results are consistent with the features of our sample in terms of the discontinuity of involved cycles. The SOM model is more efficient than the IAOM and the kernel models in term of seasonality adjustment, particularly when seasonality involves deterministic and stochastic components.

6 Conclusion

This paper focus on three seasonality identification methods: the self-organizing maps algorithm (SOM), the intra-day average observation method (IAOM) and the Nadaraya-Watson kernel method. The IAOM and the kernel methods have been used previously in the literature. We introduce the SOM algorithm in order to overcome some of their shortcomings.

We study the ability of each method to capture cycles involving deterministic and stochastic components. We implement a Monte Carlo simulation in which we generate an AR(1) and

AR(5) processes infected by a seasonality involving deterministic and stochastic cycles. Then, we capture and remove the cycles by implementing the three methods. We estimate, afterward, the deseasonalized process and we compute and compare the estimation generated errors.

Furthermore, we implement the three seasonality identification methods to capture and remove the cyclical components of two microstructure FOREX variables: Volatility and quoting activity corresponding to the 5-minutes Euro/Dollar currency quotes, and the period ranging from May 15, 2001 through May 15, 2002.

The simulation outputs carry out the following results: 1-the IAOM model is much more efficient than the kernel and the SOM methods when seasonality contains only deterministic cycles. 2-When seasonality involves both deterministic and stochastic cycles, SOM model outperforms the other methods in capturing and identifying seasonality. The empirical results corresponding to the real financial data yields consistent results with the ones obtained by simulation. The real data sample contains five closed days which trigger discontinuity and stochastic cycles. This reason explains, amongst other, that SOM outperforms, in identifying seasonality, than IAOM and the kernel methods.

References

Andersen, T., and T. Bollerslev (1996): "Heterogeneous information arrivals and returns volatility dynamics: uncovering the long-run in high frequency rendements," NBER Working paper 5752.

——— (1998): "Deutsche mark-dollar volatility: intraday volatility patterns, macroeconomic announcements and longer run dependencies," *The Journal of Finance*, 1, 219–265.

Bauwens, L., W. Ben Omrane, and P. Giot (2003): "News Announcements, Market Activity and Volatility in the Euro/Dollar Foreign Exchange Market," *Journal of International Money and Finance, forthcoming.*

Bauwens, L., and P. Giot (2000): "The logarithmic ACD model: an application to the bid-ask quote process of three NYSE stocks," *Annales d'Economie et Statistique*, 60.

Ben Omrane, W., and A. Heinen (2004): "The Information Content of Individual FX Dealers' Quoting Activity," IAG Working paper 120/04.

Bollerslev, T., and I. Domowitz (1993): "Trading patterns and prices in the inter-bank foreign exchange market," *The Journal of Finance*, 4, 1421–1443.

Cai, J., Y. Cheung, R. Lee, and M. Melvin (2001): "Once in a generation yen volatility in 1998: fundamentals, intervention and order flow," *Journal of International Money and Finance*, 20, 327–347.

Dacorogna, M., U. Muller, R. Nagler, R. Olsen, and O. Pictet (1993): "A geographical model for the daily and weekly seasonal volatility in the foreign exchange market," *Journal of International Money and Finance*, 12, 413–438.

Danielsson, J., and R. Payne (2002): "Real trading patterns and prices in the spot foreign exchange markets," *Journal of International Money and Finance*, 21, 203–222.

de Bodt, E., M. Cottrell, E. Henrion, and C. Van Wijmeersch (1998): "Self-Organizing Maps for Data Analysis : an Application to the Belgium Leasing Market," *Journal of Computing Intelligence in Finance*, 6, 5–24.

de Bodt, E., A. Lendasse, P. Cardon, and M. Verleysen (2004): "Self-organizing feature maps for the classification of investment funds," *Journal of Economic and Social Systems*, 17, 183–195.

DeBoeck, G., and T. Kohonen (1998): *Visual Explorations in Finance with Self-Organizing Maps*. Springer.

Degennaro, R., and R. Shrieves (1997): "Public information releases, private information arrival and volatility in the foreign exchange market," *Journal of Empirical Finance*, 4, 295–315.

Eddelbuttel, D., and T. McCurdy (1998): "The impact of news on foreign exchange rates : evidence from high frequency data," Working paper, University of Toronto.

Engle, R., and J. Russell (1998): "Autoregressive conditional duration: a new approach for irregularly spaced transaction data," *Econometrica*, 66, 1127–1162.

Kohonen, T. (1995): *Self-Organizing Maps*. Springer.

Lyons, R. (2001): *The Microstructure Approach to Exchange Rates*. MIT Press.

Melvin, M., and X. Yin (2000): "Public information arrival, exchange rate volatility and quote frequency," *The Economic Journal*, 110, 644–661.

Robbins, H., and S. Monro (1951): "A stochastic approximation model," *Annals of Mathematical Statistics*, 22, 400–407.

Silverman, B. (1986): *Density estimation for statistics and data analysis*. Chapman and hall.

Veredas, D., J. Rodriguez-Poo, and A. Espasa (2002): "On the (Intradaily) Seasonality and Dynamics of a Financial Point Process: A Semiparametric Approach," CORE DP, 2002/23.

Table 1: Input data for the Self-organizing maps example

observations	Xi	Yi
1	18	46
3	21	47
4	19.5	53
5	20	52
6	18	51
7	20	45
8	5	12
9	6	16
10	5.5	5
11	6.7	8
12	4.9	10
13	6.1	9
14	7	13
15	5	52
16	6	56
17	5.5	45
18	6.7	48
19	4.9	50
20	6.1	49
21	7	55

Table 2: Estimation results for the AR(1) processes with seasonality:

$$y_t^{'} = \beta^{'} y_{t-1}^{'} + \epsilon_t^{'},$$
$$z_t^{'} = \gamma^{'} z_{t-1}^{'} + \nu_t^{'}.$$

	Non-Deseas.	*Deseas. IAOM*	*Deseas. SOM(1,5)*	*Deseas. Kernel*
		Panel A	(deterministic seasonality)	
$\beta^{'}$	0.9596	0.9499	0.9433	0.9510
σ	0.10%	0.07%	0.12%	0.12%
$RMSE$	0.96%	0.09%	0.67%	0.14%
		Panel B	(stochastic seasonality)	
$\gamma^{'}$	0.9702	0.9672	0.9464	0.9640
σ	0.22%	0.30%	0.12%	0.20%
$RMSE$	2.02%	1.72%	0.36%	1.40%

$y_t^{'}$ and $z_t^{'}$ are two AR(1) processes generated, through the Monte Carlo simulation. $\beta^{'}$ and $\gamma^{'}$ are respectively the mean of the estimated AR(1) coefficients, through 1000 simulation. The second row presents the estimated coefficients for the non-deseasonalized AR(1) process. Rows three to five, show the estimated coefficients for the deseasonalized AR(1) process respectively by IAOM, kernel and SOM(1,5) methods. RMSE corresponds to the root mean square difference between the estimated coefficient and the simulated one, β.

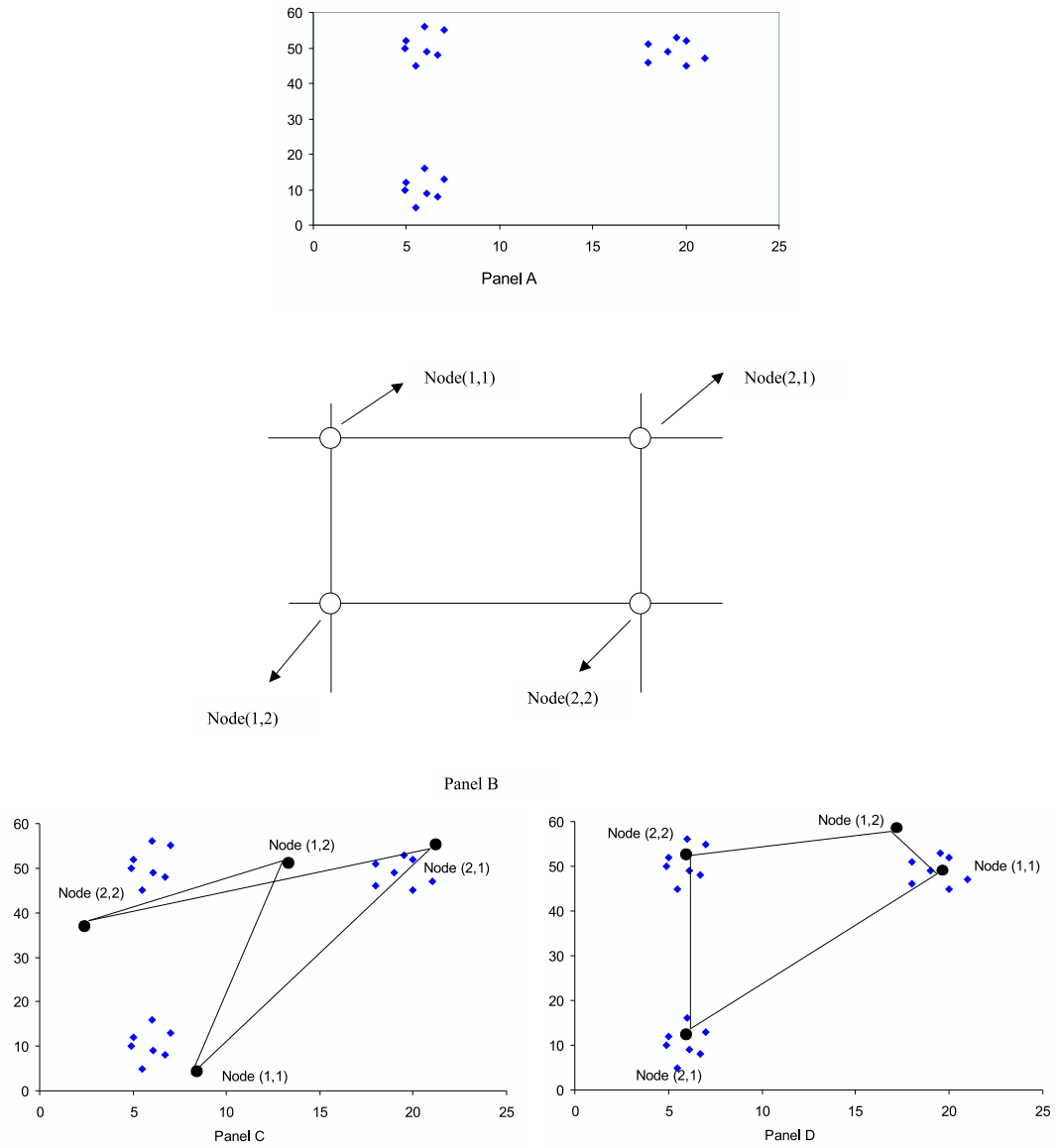

Figure 1: Illustration for the Self-organizing maps algorithm

Table 3: Estimation results for the AR(5) processes with seasonality:

$$y_t^{'} = \sum_{p=1}^{5} \beta_p^{'} y_{t-p}^{'} + \epsilon_t^{'},$$
$$z_t^{'} = \sum_{p=1}^{5} \gamma_p z_{t-p}^{'} + \nu_t^{'}.$$

	Non-Deseas.	*Deseas. IAOM*	*Deseas. SOM(1,5)*	*Deseas. Kernel*
		Panel A	(deterministic seasonality)	
$\beta_1^{'}$	0.595	0.500	0.501	0.590
σ	0.35%	0.36%	0.52%	0.35%
$RMSE$	9.47%	0.29%	0.36%	9.00%
$\beta_2^{'}$	0.114	0.901	0.0904	0.114
σ	0.42%	0.40%	0.44%	0.38%
$RMSE$	2.42%	0.33%	0.35%	2.37%
$\beta_3^{'}$	0.091	0.079	0.0802	0.088
σ	0.42%	0.40%	0.43%	0.36%
$RMSE$	1.09%	0.33%	0.34%	0.84%
$\beta_4^{'}$	0.073	0.070	0.0702	0.072
σ	0.44%	0.42%	0.44%	0.40%
$RMSE$	0.46%	0.34%	0.35%	0.38%
$\beta_5^{'}$	0.064	0.0.059	0.0601	0.059
σ	0.38%	0.39%	0.42%	0.38%
$RMSE$	0.43%	0.31%	0.33%	0.32%
		Panel B	(stochastic seasonality)	
$\gamma_1^{'}$	0.687	0.653	0.516	0.685
σ	2.13%	2.52%	0.61%	2.31%
$RMSE$	18.66%	15.26%	1.63%	18.49%
$\gamma_2^{'}$	0.114	0.116	0.0904	0.114
σ	0.47%	0.45%	0.47%	0.52%
$RMSE$	2.35%	2.55%	0.72%	2.37%
$\gamma_3^{'}$	0.077	0.084	0.085	0.077
σ	0.61%	0.61%	0.45%	0.59%
$RMSE$	0.52%	0.58%	0.59%	0.52%
$\gamma_4^{'}$	0.054	0.062	0.075	0.054
σ	0.67%	0.73%	0.45%	0.70%
$RMSE$	1.60%	0.89%	0.54%	1.57%
$\gamma_5^{'}$	0.036	0.047	0.067	0.037
σ	0.72%	0.87%	0.42%	0.80%
$RMSE$	2.41%	1.36%	0.71%	2.34%

See caption of Table 2.

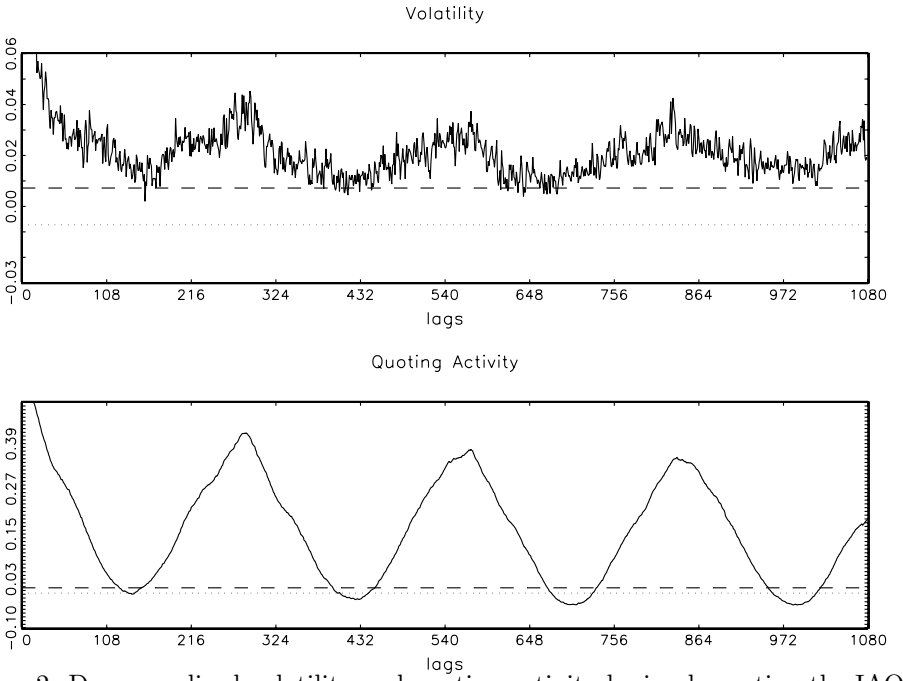

Figure 2: Deseasonalized volatility and quoting activity by implementing the IAOM.

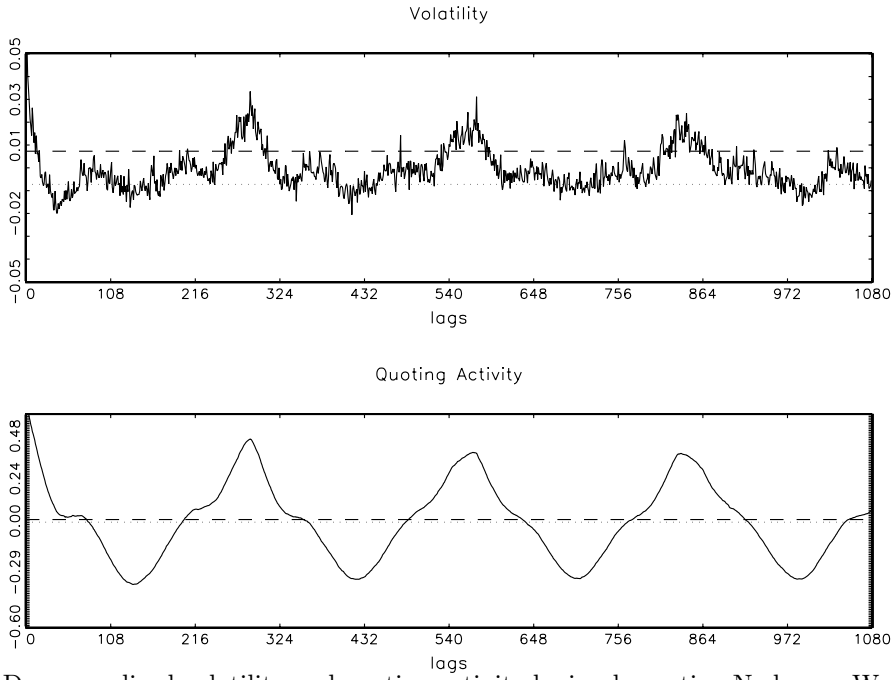

Figure 3: Deseasonalized volatility and quoting activity by implementing Nadaraya-Watson kernel smoothing method.

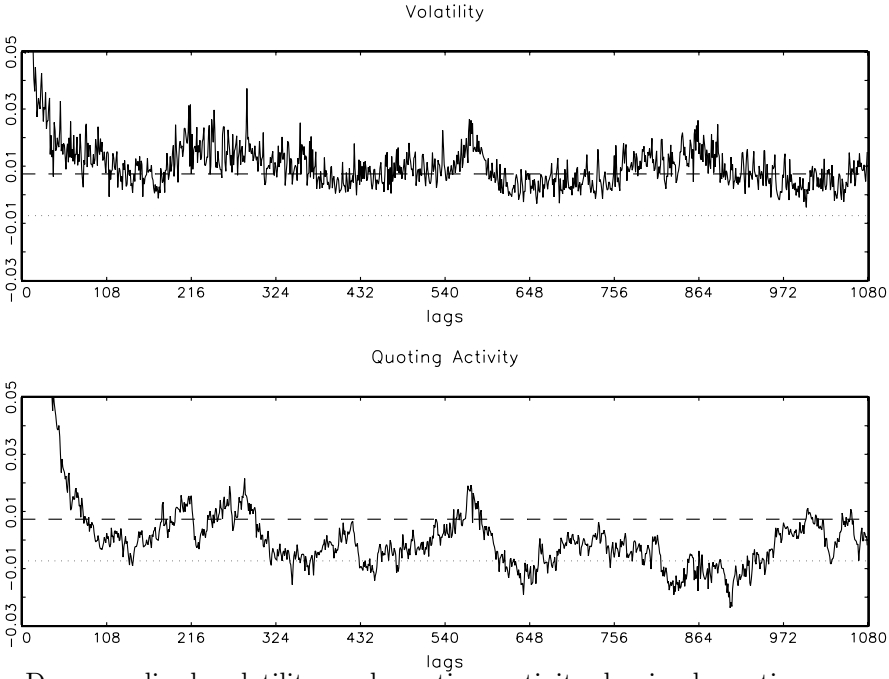

Figure 4: Deseasonalized volatility and quoting activity by implementing respectively the SOM(2,5) and the SOM(6,6).

Table 4: Moments of the Euro/Dollar returns and quoting activity

	Returns	Quoting activity
Mean	0.007	82.31
Standard deviation	3.91	60.11
Skewness coefficient	0.21	1.23
Kurtosis coefficient	15.0	5.43

The 5-minute returns have been pre-multiplied by 10,000 (to avoid small values). The number of observations is 72,675, corresponding to the period from May 15, 2001 to May 15, 2002.

Table 5: Moments and autocorrelation coefficient for the Euro/Dollar deseasonalized volatility and quoting activity

	IAOM V	*IAOM* QA	*SOM(2,5)* V	*SOM(6,6)* QA	*Kernel* V	*Kernel* QA
μ	0.999	1.000	1.038	0.998	0.918	0.923
σ	2.180	0.554	2.629	0.387	2.045	0.440
ρ_{288}	0.037	0.417	0.037	0.014	0.027	0.461
ρ_{576}	0.032	0.372	0.025	0.006	0.031	0.382
ρ_{864}	0.030	0.320	0.026	-0.017	0.011	0.295

The number of observations is 72,675, corresponding to the period from May 15, 2001 to May 15, 2002. The seasonality adjustment was done by implementing the SOM model presented in Section 3.1, the intradaily average observations one (iaom) presented in Section 3.2 and the kernel smoothing method detailed in section 3.3.