Autocorrelation of trade signs on electronic stock markets

Matthieu Félix, Charles Bine

May 2016

1

Contents

1 Introduction

This project aims to study the evolutions in the structure of transaction sign autocorrelation on electronic stock markets between 2000 and 2013. Previous works on this topic, based on data from the first half of the 2000's, describe a very long memory of these signs (with a power-law decrease of autocorrelation). It is then natural to ask whether the massive rise of high-frequency trading and automated execution strategies have induced significant evolutions of this structure.

The high-frequency data used to conduct this study was provided by the Chair of Quantitative Finance at CentraleSupélec.

2 Literature review and state of the art

This study follows up on previous research, notably by Jean-Philippe Bouchaud and Fabrizio Lillo on the long memory of trade signs on stock markets.

2.1 Autocorrelation

In order to examine the trade sign long-memory hypothesis, we will study their autocorrelation, i.e. correlation of the signal with itself. Autocorrelation allows detecting regularities or repeated patterns in a signal, even if it is noisy.

The autocorrelation of a random process $(X_n)_{n\in\mathbb{N}}$ with average μ and variance σ^2 is defined as:

$$\varphi(k) = \frac{1}{\sigma^2}\mathbf{E}[(X_i - \mu)(X_{i+k} - \mu)].$$

Autocorrelation takes its values in $[-1, 1]$. A value of 1 shows perfect correlation between the process and its shifted copy.

If we observe n values $(x_1, x_2, \ldots, x_n)$ of a stationary process, the empirical autocorrelation φ can be written as:

$$\varphi(k) = \frac{\sum_{i=1}^{n-k}(x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

Some of the studies we quote here give results on the autocovariance, not autocorrelation, of trade sign; these two functions being proportional, this does not fundamentally change the speed of decrease.

We computed autocorrelation using the `acf` function in R.

2.2 Measures of exponents of trade sign autocorrelation laws

Studying the London Stock Exchange (LSE), Fabrizio Lillo described a power-law behavior for the autocorrelation of order sign. The exponent of this power law ranges from $\gamma = 0.39$ to 0.6.[8] He also observed a change of this exponent around a lag of 500. This study was particularly concerned with data from the Vodafone stock (at the time the most traded stock on the LSE). The same study also describes a long-memory phenomenon with trade signs on the New-York stock exchange.

Independently, Bouchaud applied these methods to the Paris stock exchange, notably to France Télécom and Total, and finds exponents ranging from $\gamma = 0.2$ to $\gamma = 0.7$.[2]

These observations were studied in subsequent articles by models based either on hidden orders,[1] or on traders blindly following market movement.[2] In any case, the exponent γ is inferior to 1, which shows that autocorrelation is non-integrable and corresponds to a long-memory process.

2.3 Other criteria for long-memory hypothesis validation

Lillo[8] notes that when studying of long-memory processes, autocorrelation can be a delicate indicator because statistical errors can be important. The study suggests another criterion for validating the long-memory hypothesis, based on Lo's version of Hurst's R/S test.

This test essentially consists in comparing the maximum and minimum of the partial sums of the difference between observations and empirical mean:

$$\max_{1 \le k \le n} \sum_{j=1}^{k} (X_j - \bar{X}_n) - \min_{1 \le k \le n} \sum_{j=1}^{k} (X_j - \bar{X}_n)$$

The difference between the minimum and the maximum is larger for long-memory processes; once this difference is normalized by the empirical variance of the sample, one can compute the confidence intervals for the long-memory hypothesis. On the LSE, both Hurst's original R/S test and Lo's version confirm the long-memory hypothesis.

2.4   Trade sign computation

The most common algorithm to determine the sign of trades from other information (volume, date, price) was described by Lee and Ready.[6] Although studies from the end of the 1990's evaluated its precision to about 85%,[4] it was nonetheless criticized,[3] notably following recent changes on the markets. New algorithms were developed recently to address its shortcoming, notably in CentraleSupélec.[9]

3   Methodology

3.1   Choice of stocks

We worked on a set of 15 stocks, all of which are traded on the Paris stock exchange: Accor (ACCP), Air France (AIRF), Alstom (ALSO), Axa (AXAF), BNP Paribas (BNPP), Crédit Agricole (CAGR), Eurazeo (EURA), Casino (CASP), LVMH, Sanofi (SASY), Société Générale (SOGN), Total (TOTF), Ubisoft (UBIP), Vallourec (VLLP) and Zodiac Aerospace (ZODC). Liquidity data for these stocks are summed up in annex A.

The data are considerably less noisy with the more liquid stocks (Total, Société Générale, etc.). This is coherent with the literature and led us to study them in priority to refine our analysis and optimize the processing. The less liquid stocks are also more prone to unpredictable capital movements involving a large portion of the (relatively small) capital.

We have also checked that the month of March, which we privileged in our study, was not a usual month of dividend payment for the stocks we used.

3.2 Data retrieval

Availability of historical data CentraleSupélec's Chair of quantitative finance made trade and quotes data for various exchanges available to us. We thank the Chair for these data, without which the study would not have been possible. For this study, we used trade and quotes data from the Paris stock exchange between 2000 and 2013, the most recent available year.

Trade signs were precomputed by the Chair using Muni Toke's Algorithm.

Data pre-treatment The data we retrieved had some artifacts we had to remove before we could process them. Some trades have no computed sign, notably at the beginning and end of each day. These trades cannot be used and were removed from our dataset.

Similarly, a few very large trades are reported outside the opening hours of the market; these trades represent aggregates of trades that occurred outside the normal trading period and on which we have little coherent data. We have thus also removed the first and last 15 minutes of each day from our dataset.

Order concatenation Frequently enough, a set of multiple trades corresponds to a single buy or sell order. We group these trades during preprocessing since the actual desire of the agent was only to pass one order, but the insufficient liquidity of the market split it into several orders. Trades are thus grouped if they are reported at the same time and with the same price and sign.

Not grouping these trades would artificially augment trade sign autocorrelation since all trades in such a group have the same sign.

3.3   Segmentation

To best use our results, we have tried to smooth the differences that exist between stocks (especially their varying liquidities). A comparison of trade sign autocorrelation between the different stocks was made in annex B. There also exists differences between two months on the same stock; in particular, daily traded volume significantly increases year over year, as can be seen in annex A.

In order to do so, we have grouped trades together in order to create blocks that are comparable from one stock to another and from one year to another. The autocorrelation is then computed on the bocks instead of the raw trades. Several ways to segment trades are conceivable; we will detail some of them in the following sections.

The major problem this method causes is that grouping successive trades make trade sign autocorrelation decrease faster, since one unit of lag for the grouped trades will correspond to several units for ungrouped trades. Another problem is deciding how to assign signs to the newly created blocks.

In the following parts, we will call *event time* the computation of sign autocorrelation without performing any grouping: individual trades are thus considered to happen at uniformly spaced time intervals.

3.4   Physical time segmentation

A first approach to make data from different stocks more comparable is to compute autocorrelation depending on physical time instead of event time. To do so, a time interval is chosen and trades are grouped according to the interval in which they belong.

Depending on the stocks, there can be as few as one or as many as a few hundred trades per minutes.

3.5   Volume time segmentation

3.5.1   Absolute volume time segmentation

Volume time segmentation is a way to group trades according to traded volume. In *absolute* volume time, the volume of a group (V_{group}) is the same

Figure 1: Example of absolute volume time segmentation. All groups have the same volume, regardless of the considered stock.
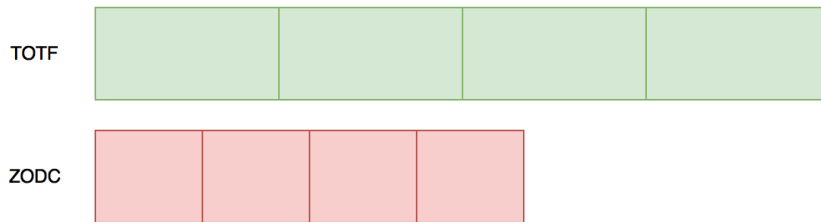

Figure 2: Example of relative volume segmentation. Each group's volume depends on the stock, but the total number of groups is constant across all stocks.

for all stocks. An illustration of this segmentation method is shown on figure 1.

For instance, if we set $V_{\text{group}} = 500$ and if during the month of March 2009, $500\,000$ Total and $50\,000$ Zodiac Aerospace shares have been traded, trades will be grouped in packs of 500 in both cases; there will then be respectively 1000 and 100 trade groups. This method does not compensate for the differences in total number of trades during a month, but allows comparing two stocks by smoothing the differences caused by differences in average order size.

For each stock s, we have $V^s_{\text{group}} = V_{\text{group}}$.

3.5.2   Relative volume segmentation

Another way to perform volume segmentation is to choose a number N of trade groups, and leave it constant across all stocks. We then have $V^s_{\text{group}} = N \cdot V_{\text{group}}$ for each stock s. This option is illustrated on figure 2.

Each (period, stock) pair will thus have its own group volume. This volume represents a $1/N$ share of the total considered period. This method's

8

main advantage when compared to absolute volume segmentation is that all groups have comparable influence overall.

3.6   Choice of sign for a trade group

When segmenting, several methods can be considered to decide the sign of a set of grouped trades.

Most frequent sign Using this method, we count the number of +1 and -1 trades in the group, and set the sign of the group to be the most common. When thus have $\varepsilon_{\text{group}} = \pm 1$. This method can seem somewhat counter-intuitive, but reduces the problem of trade aggregation lowering autocorrelation artificially. A variant of this method is to take the volume of trades into account when determining the most frequent sign.

Average Taking the average of all trades in the group is an obvious solution. We then have $\varepsilon_{\text{group}} \in [-1, 1]$. A variant of this method is to weight each trade with its volume.

We have used the second method for this study, because it gives a fairer evaluation of each trade's contribution, and it is not generally problematic to have a non-integer sign. What is more, the most frequent sign method could lead to overestimating autocorrelation by erasing heterogeneity in trade signs.

3.7   Methods to split large trades into groups

Although the problem seems relatively simple, some important questions do remain. Trade volumes are very variable, which makes it difficult to group them nicely: what if a trade is larger than the volume we've chosen for a group ?

3.7.1   Segmentation with overfilling

In this method, we concatenate trades up to volume V_{group} at least. If we denote by $\tau = \{T_i\}_i$ the set of trades in chronological order, V_i the volume of T_i, and G_n the nth group:
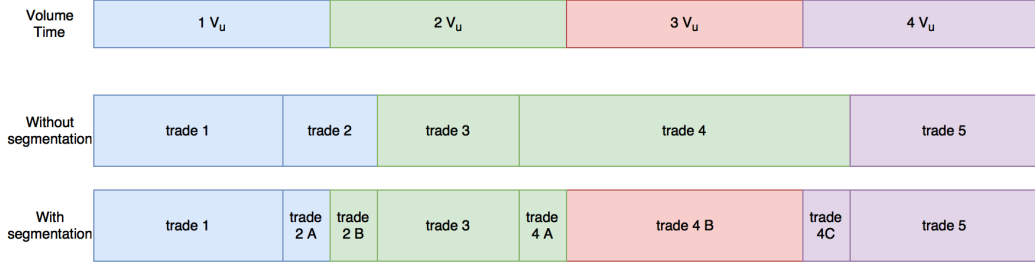
Figure 3: Segmentation of trades into isovolume groups, with and without overfilling. The first bar represents volume time: each colored section corresponds to unit volume V_u. The other two bars represent 5 trades of different volumes. The color with which the trades are represented assigns them to the group in which they are counted. The second bar shows segmentation with overfills, the third one segmentation without overfills. Note this figure is not to scale and that in practice group volumes should be much larger than average trade size.

$$G_n = \left\{ T_i \in \tau, \sum_{k=0}^{i-1} V_i < n \times V_{\text{group}} \text{ and } \sum_{k=0}^{i} V_i > n \times V_{\text{group}} \right\} \qquad (1)$$

In this configuration it is possible (and indeed almost guaranteed) that groups will be overfilled, i.e. $V_n > V_{\text{group}}$ for some groups. Groups will thus not have the exact same volume. This method is illustrated on the second line of figure 3. In this example, group 1 is overfilled, since trade 2 should be splits over groups 1 and 2 but is only counted in group 1.

3.7.2 Segmentation without overfilling

This method consists of splitting trades that do not fit in one group. This way, all groups will have the same volume, but a given trade can be counted in several groups.

The last line of figure 3 illustrates this principle. Trade 2, which does not fit, is split into trades 2A and 2B, which are respectively counted in volume time 1 and volume time 2.

This method is coherent with the convention of computing group signs with a weighted average of the trades they are made up of.

3.8   Method choices

3.8.1   Method of segmentation

In order to compare stocks together in the best possible way, we have used relative volume time and event time. While these two methods give different results, both seem reasonable ways to make data comparable.

3.8.2   Parameter choice

Once a segmentation method is chosen, we must decide the value of its parameters. In our case the value of N is the main question. Let us first discuss the impact this choice can have on our results.

N **is too small**   If the number of subdivisions of the studied period is too small, the volume of each group will be very large, and a large amount of trades will be merged in one volume-time instant of volume V_{group}. This causes a loss of information on the evolution of signs.

N **is too large**   If N is too large, the opposite happens. V_{group} becomes too small to contain more than one trade and we essentially go back to event time, rendering segmentation useless.

To avoid these two problems, N must be as small as possible while remaining larger than the average size of a trade. That is, N is chosen in such a way that the resulting V_{group} for all pairs of (period, stock) is larger than the average size of a trade for that stock during that period.

3.8.3   Method of splitting

In the following, we have usually chosen the overfilling method, for several reasons. First, splitting trades to avoid overfills was an extra data manipulation, which could have introduced artifacts, and whose usefulness was limited if the number of large overfills remained relatively low. This is relatively easy to guarantee by choosing a large V_{group}.

Additionally, splitting trades can change the autocorrelation in some cases. For instance, suppose that one trade has volume $V \gtrsim k \times V_{group}$,

with $k \geq 2$. Such a situation is shown by trade 4 in figure 3. If this trade is split over several groups, it will influence the sign of several volume instants, which over-evaluates its impact on autocorrelation. Specifically, trade 4 is split into trade 4A, which counts for the second instant, trade 4B, which counts for the third instant, and trade $4C$ which counts for the fourth instant. If we choose not to split trades, this trade counts only for instant 2.[1]

4 Results

4.1 Averaging, segmentation

To compare the evolution of trade sign autocorrelation year to year, we've averaged them in two ways:

- First, we averaged sign autocorrelations for the month of March over the 15 stocks, year by year (i.e. the March 2000 data for Accor, Air France, Crédit Agricole,...were averaged to give the general average for March 2000; all March 2001 data was averaged to give that of 2001, etc.)

- Then, we averaged the data of all months, year by year, for a single stock. Given the very large amounts of data involved, we only computed this average for Total and Crédit Agricole. These two stocks were chosen because they're very liquid and are part of two very different industries.

In order to study the evolution of of trade sign autocorrelation decrease between 2000 and now, the most usable plots are those of annual averages over one or several stocks. Indeed, simply plotting the data without any averaging often leads to noisy plots; averaging also helps correcting small variations that can occur from one month to the next.

Generally speaking, plotting the data in event time and in volume time (whether physical or absolute) gives similar results. In volume time, autocorrelation decreases faster as the size of segments gets higher. This is expected: if trades are divided into a small number of trade groups, trades in one trade

---

[1]In this case, group 3 will be empty. Empty groups are removed before computing autocorrelation.

group will on average be far from those in the next. Physical time plots are generally somewhat noisier.

We have also checked that the noisier look of the plots of less liquid stocks was actually due to statistical noise and not to different behavior of sign autocorrelation for these stocks. Averaging the four stocks of our study which were the least liquid in 2013 (Ubisoft, Eurazeo, Zodiac Aerospace and Casino) gives essentially the same results as those we describe in the next section, although the plot remains a bit noisier for large lags.

4.2   Study of the decrease in autocorrelation

Lillo[8] found the trade sign autocorrelation of the Vodaphone stock on the LSE in 2004 to be significant up to a lag of 10^4. For the Total stock, we find a coherent autocorrelation up to about $3 \cdot 10^3$ in 2005 et 10^3. However the noise level is significant after a lag of 500, especially after averaging all stocks in our study (which includes stocks far less liquid, and thus far more noisy, than Total). For the remainder of this study, we have focused on lags up to 300.

With both methods of averaging (over several stocks or over one stock), we can observe a significant evolution starting in 2010. Until 2009, autocorrelation follows a power law (figure 4)[2]. From 2010 on, autocorrelations can be described with two successive power laws of differing exponents, changing around $t = 10$.

These results are clearly visible on double logarithmic scale plots: in this scale, trade sign autocorrelations form a straight line until 2009, then angle. This phenomenon is visible on both methods of averaging. Using a linear regression on the log-log plots, one finds an exponent of $\gamma \approx 0.28$ to 0.38 up to 2009, then about $\gamma \approx 0.50$ to 0.60 on the first piece and $\gamma \approx 0.17$ to 0.20 on the second. At a lag of 300, autocorrelation is approximatively equal for all years (to about 10^{-2}). At a lag of 10, however, it is equal to about $5 \cdot 10^{-2}$ for 2010-2013, contrasting with a value of 10^{-1} for the 2000-2009 period.

These variations can also bee seen on relative volume time plots. Physical time plots (with 60 second segments) suffer from higher noise; the break in slope is not clearly visible, but autocorrelation at lag 2 is much higher in 2010-2013 than before. The following slope is also less steep.

---

[2]Figure 4 shows averaged data grouped by absolute volume time, in 100-trade groups.

Overall, this change is relatively intriguing: in the later years, the initial decrease is much faster but is compensated at larger lags by a slower decrease of the tail.

4.3   Analysis

The most significant changes on financial markets since 2007-2008 are without doubt market deregulation and the growing share of algorithmic trading in exchanges. The impact of these changes can be relatively ambiguous: high-frequency trading, for instance, will tend to cause a large amount of trades, which depending on the algorithm can be of correlated signs or not.

Deregulation allows more actors to trade on the exchanges and have an impact on trade signs. Trade sign autocorrelation may then have a steeper decrease because of this increased liquidity.

Additionally, the growing role of algorithmic trading (in 2008, on the German stock exchange, 50 to 60% of all traded volume was due to algorithmic trading[5]). This also adds liquidity to the market, and possibly noise because of the large number of trades.

In any case, it remains difficult to explain why the power law exponent grows in the second phase.

4.4   Potential development paths

Previous studies on this topic describe a power-law trade sign autocorrelation decrease on all studied stock markets; hence, this study could be extended to other market to confirm the changes we observe from 2010 onwards. Similarly, it would be relevant to study years 2014 and 2015 to extend our analysis, since these data were not available to us.
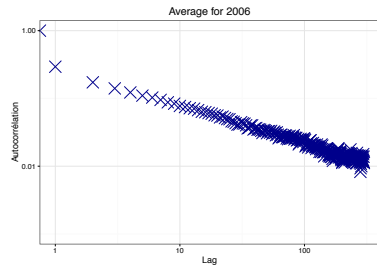
As we mentioned in the literature review, there exist other test criteria for long memory hypothesis, and other quantities beside trade signs have been shown to display long-memory behavior. These could provide relevant extensions to this study in order to confirm or refine our results.
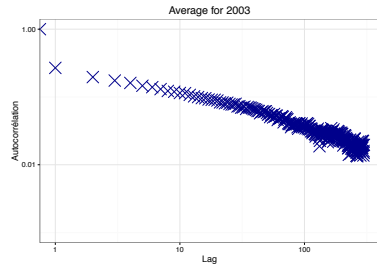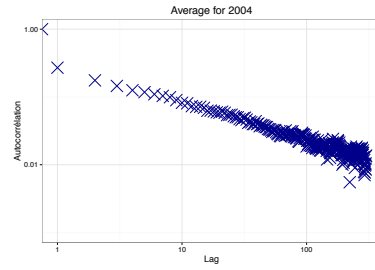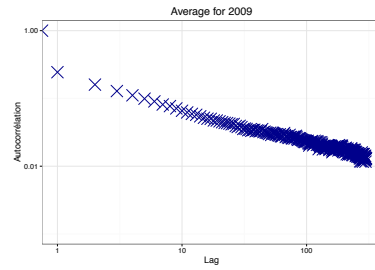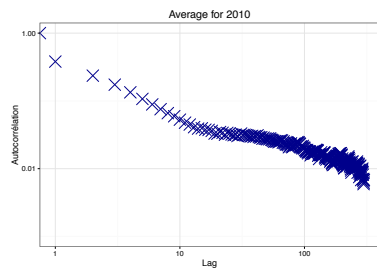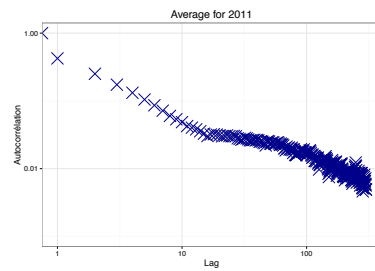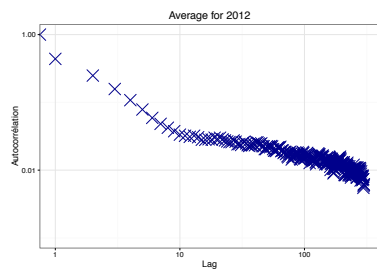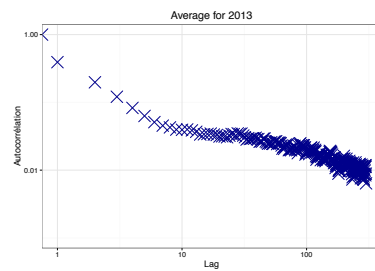
Figure 4: Average trade sign autocorrelation for the month of March over the 15 studied stocks in 2002, 2004, 2006 and 2009 to 2013. Log-log scale.
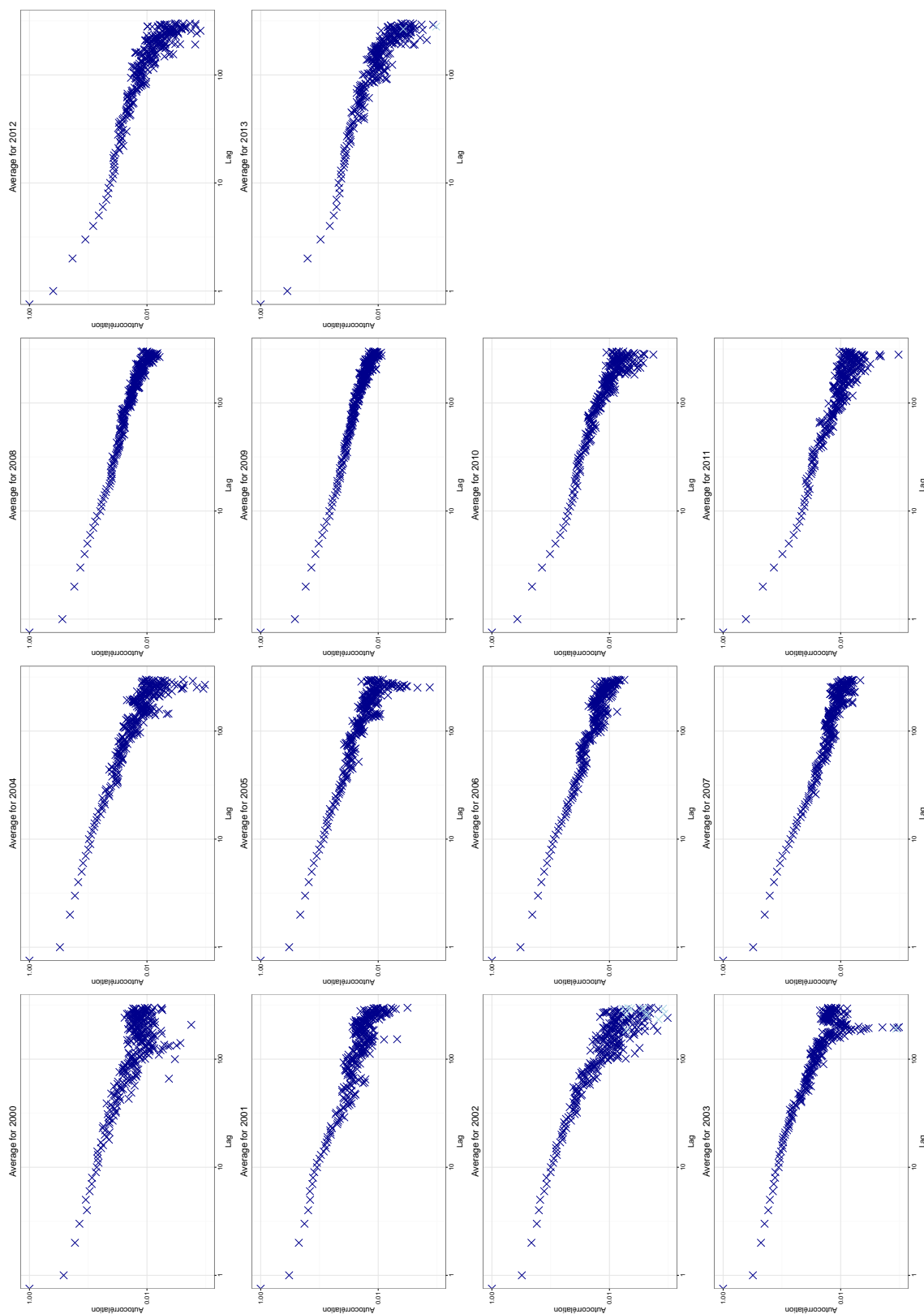
A   Studied stocks, numbers of trades

		Cap. (May 16) MEUR	Trades per month			
			2000	2005	2010	2013
- Total	TOTF	105596	65357	120490	338476	276680
Crédit Agricole	CAGR	23323	104650 (2002)	83786	199122	218274
BNP Paribas	BNPP	60900	61786	81115	329312	403513
Société Générale	SOGN	29204	34313	69226	338761	368382
Sanofi	SASY	94544	27737	104858	213480	283496
Axa	AXAF	55120	50386	115997	224724	216102
LVMH	LVMH	73226	43585	46393	161565	146771
Air France	AIRF	2163	29784	40515	82866	115807
Alstom	ALSO	4938	32556	39708	157750	106303
Vallourec	VLLP	1288	2546	6294	111369	88051
Accor	ACCP	8886	61127	46873	82676	78528
Casino	CASP	6000	24650	29096	51459	77790
Zodiac Aerospace	ZODC	6111	7171	9276	38303	52562
Eurazeo	EURA	4255	2033	5902	24263	20482
Ubisoft	UBIP	3669	34767	9027	50633	16823

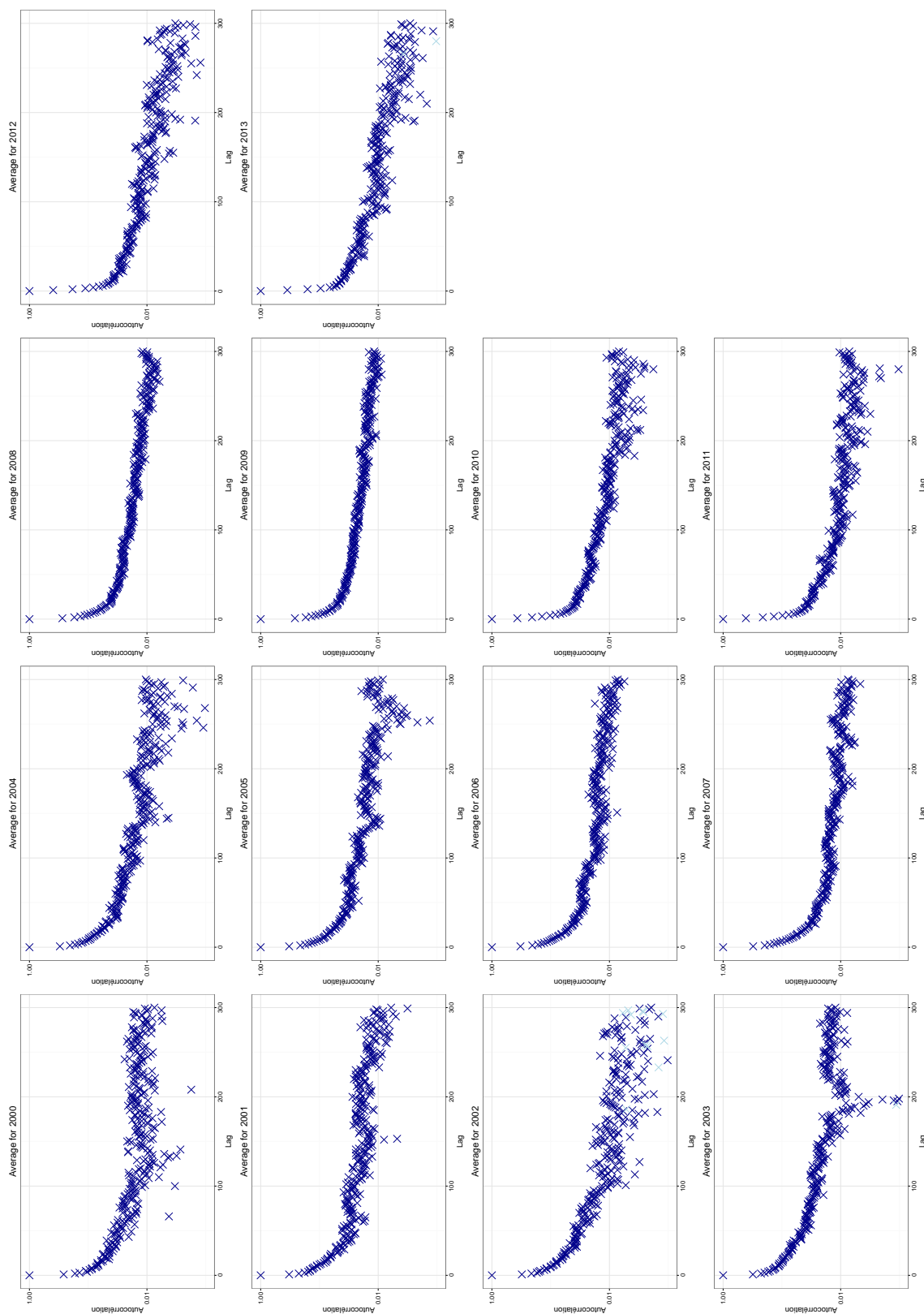Nombre de trades durant le mois de mars pour une action, moyenne sur les 15 actions étudiées
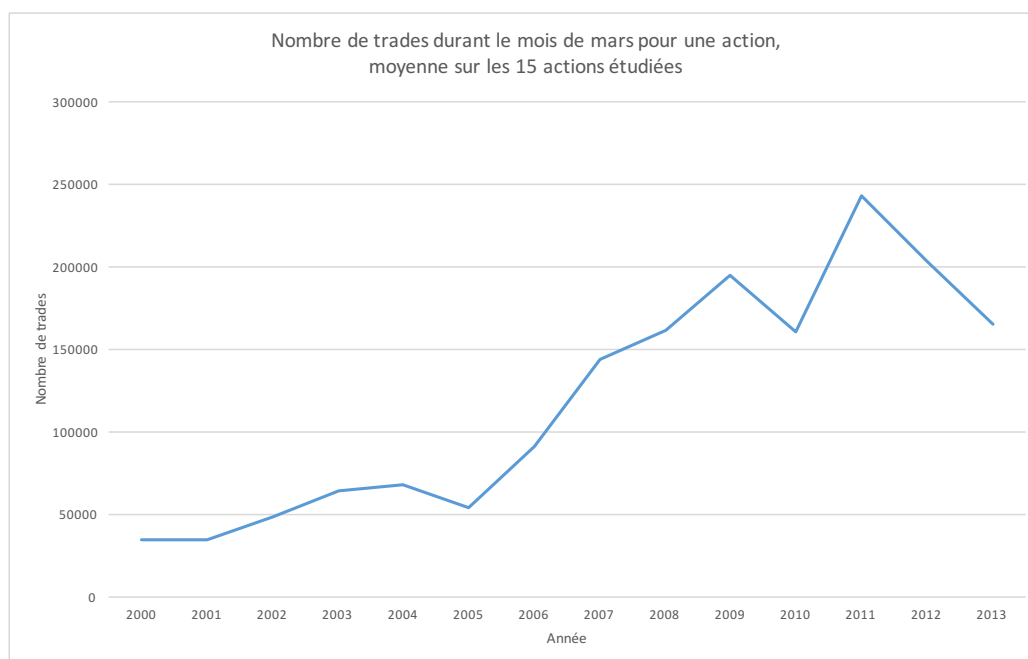
B   Autocorrelation graphs

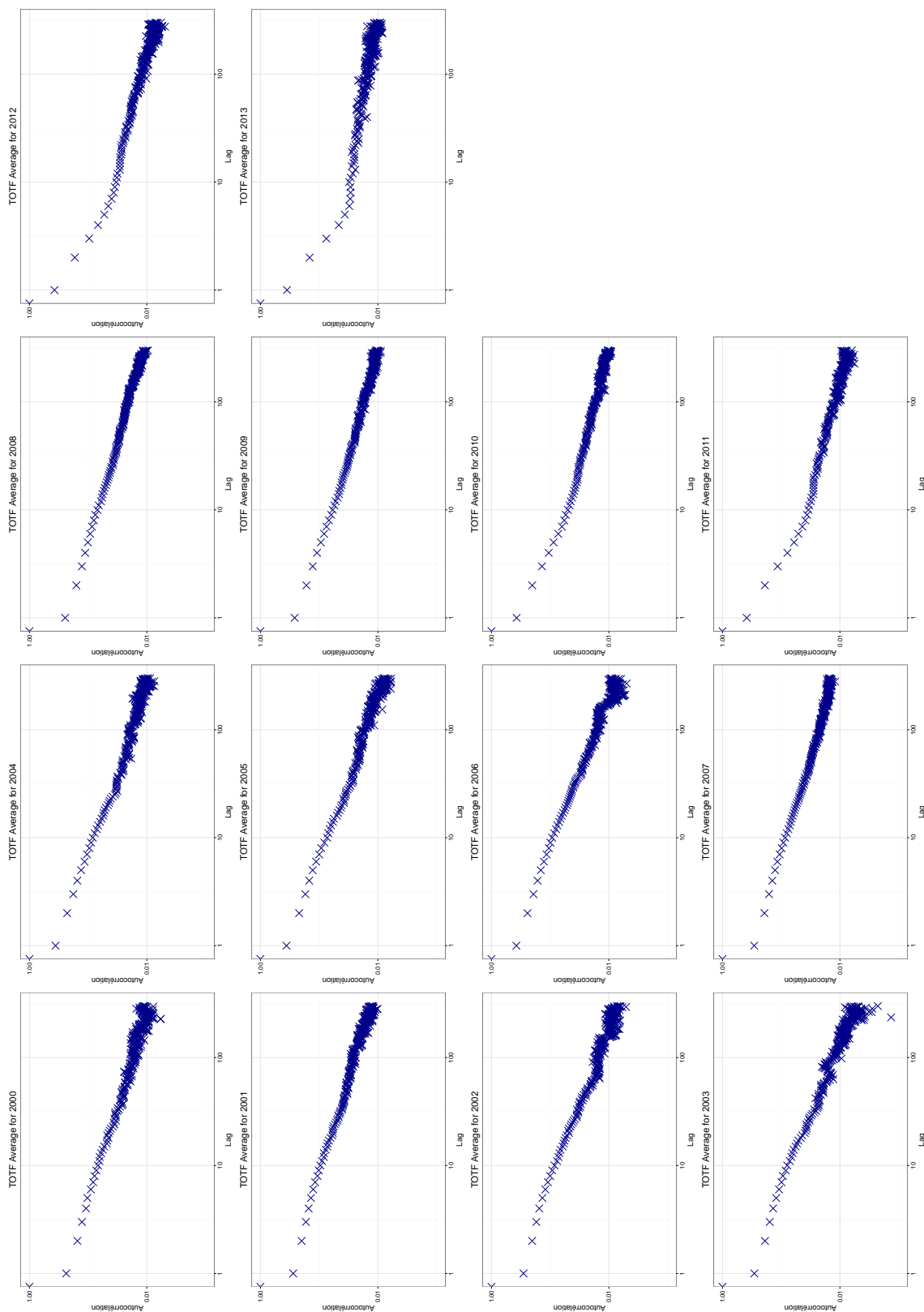## B.1 Global averages over all stocks

Moyennes annuelles tous stocks - temps de volume, blocs de volume 200, échelle log-log
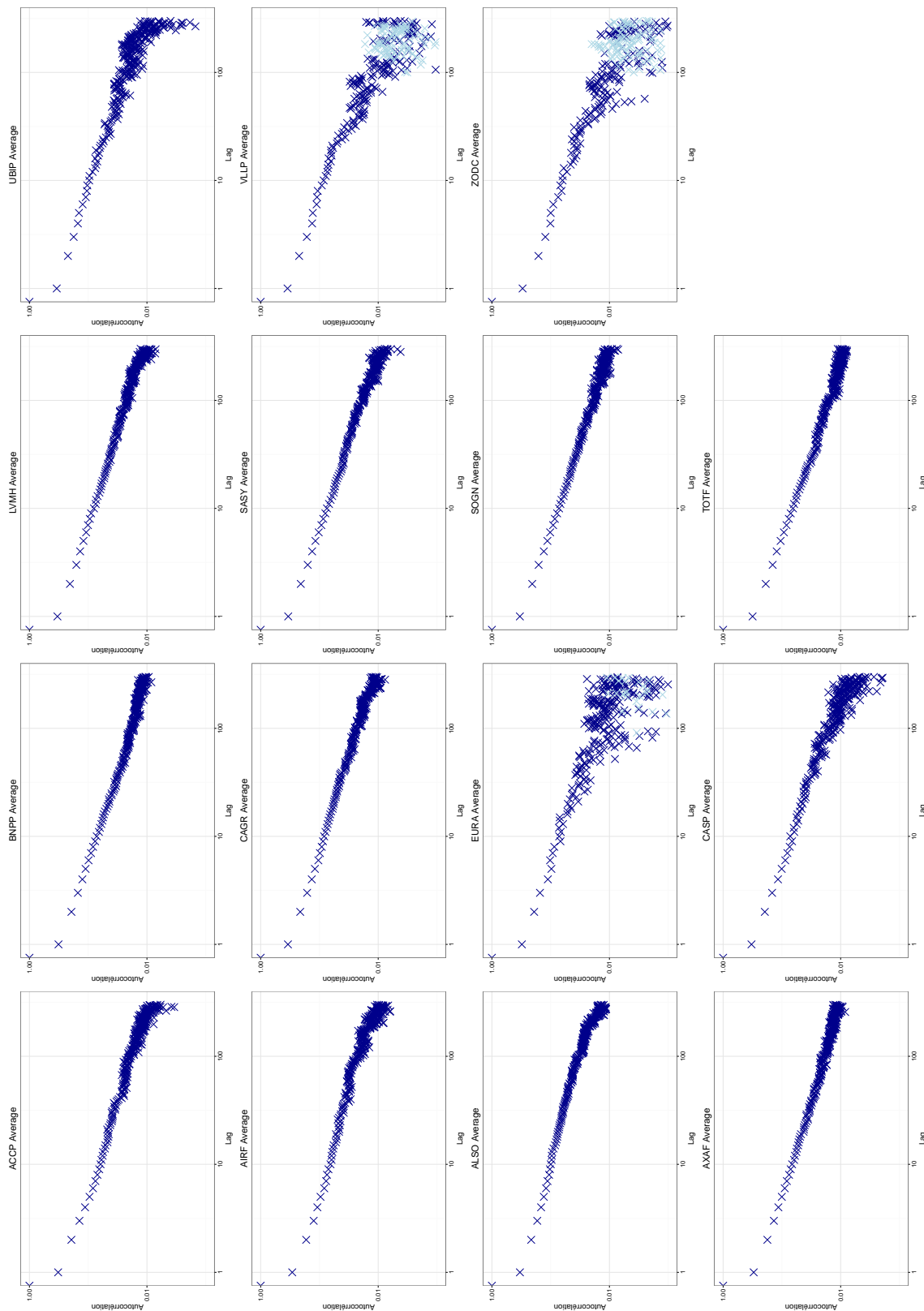
B.2 Global averages over all stocks

Moyennes annuelles tous stocks - temps de volume, blocs de volume 200, échelle log-linéaire

B.3 Global averages for Total (TOTF)

Moyennes annuelles pour Total - temps de volume, blocs de volume 200, échelle log-log

B.4 Global averages by stock

Moyenne globale pour chaque stock - temps de volume, blocs de volume 200, échelle log-log

C   References

[1] Jean-Philippe Bouchaud, J Doyne Farmer, and Fabrizio Lillo. "How markets slowly digest changes in supply and demand". In: *Fabrizio, How Markets Slowly Digest Changes in Supply and Demand (September 11, 2008)* (2008).

[2] Jean-Philippe Bouchaud et al. "Fluctuations and response in financial markets: the subtle nature of 'random'price changes". In: *Quantitative Finance* 4.2 (2004), pp. 176–190.

[3] Bidisha Chakrabarty, Pamela C Moulton, and Andriy Shkilko. "Short sales, long sales, and the Lee–Ready trade classification algorithm revisited". In: *Journal of Financial Markets* 15.4 (2012), pp. 467–491.

[4] Thomas J Finucane. "A direct test of methods for inferring trade direction from intra-day data". In: *Journal of Financial and Quantitative Analysis* 35.04 (2000), pp. 553–576.

[5] Terrence Hendershott, Charles M Jones, and Albert J Menkveld. "Does algorithmic trading improve liquidity?" In: *The Journal of Finance* 66.1 (2011), pp. 1–33.

[6] Charles Lee and Mark J Ready. "Inferring trade direction from intraday data". In: *The Journal of Finance* 46.2 (1991), pp. 733–746.

[7] Fabrizio Lillo and J Doyne Farmer. "The key role of liquidity fluctuations in determining large price changes". In: *Fluctuation and Noise Letters* 5.02 (2005), pp. L209–L216.

[8] Fabrizio Lillo and J Doyne Farmer. "The long memory of the efficient market". In: *Studies in nonlinear dynamics & econometrics* 8.3 (2004).

[9] Ioane Muni Toke. "Reconstruction of Order Flows using Aggregated Data". In: *arXiv preprint arXiv:1604.02759* (2016).