A Data Paradigm for Robustification of Parametric Estimation: Realized Volatilities and Kernels from Non-Synchronous NASDAQ Quotes [Running Title: LOB Data Paradigm]

Ranjan R. Chakravarty[1] and Sudhanshu Pani[1]

[1]School of Business Management, NMIMS University, Mumbai - 400 056, India.

Abstract

Ultra High Frequency (UHF) quotes and trades are examined in high resolution. Patterns which do not correspond to plausible market activity as in Brownlees and Gallo (2006) are observed. Non-microstructure noise is identified and diagnostic methods are evaluated. Extending Barndorff-Nielsen et al. (2009), a paradigm of data handling that synthesizes statistical technique and limit order book modeling is developed. Empirical evidence from the NASDAQ 100 demonstrates that removal of non-microstructure noise from the limit order book robustifies estimation across techniques and levels of market depth.

KEYWORDS: Robustification, Data Handling, Limit Order Book, Model Fit, Estimation, Non-Microstructure noise, Ultra High Frequency.

THE MOTIVATION behind this work is to provide researchers in market microstructure a paradigm with which to better understand the information content of quote and trade level Ultra High frequency (UHF) data. In this context, it presents focused data techniques designed to improve parametric estimation in empirical microstructure research.

Market microstructure data consists of trade and quotes, signals and outliers. Our focus is on the information content of quotes. We observe in UHF data that white noise emanates not only from outliers but also from specific stochastic processes. This paper demonstrates how model fit can be enhanced by reduction in such white noise without resorting to any model re-specification.

Invoking Brownlees and Gallo (2006), we specify that in auction markets quotes that participate in the auction in search of a trade are considered to be plausible market activity. Quotes that do not comprise plausible market activity are termed non-microstructure noise. Identifying and removing those quotes that do not represent plausible market activity are seen to be key to improved estimation in practice.

This paper makes three specific contributions to the existing UHF empirical microstructure literature. First, it shows that in limit order markets there exist stochastic noise processes in addition to and different from microstructure noise. Secondly, it proposes a new paradigm to identify these noise processes and remove them. Third, it empirically demonstrates how the application of this paradigm results in vastly improved parametric estimation.

The remainder of this paper is divided into four parts. The next section analytically explains the formulation of non-microstructure noise. Section 2 presents a three stage paradigm to identify and treat non-microstructure noise. An empirical demonstration of the efficacy of this paradigm is presented in section 3 via the estimation of Realised Kernels (RK) in the presence of non-microstructure noise. The final section provides a summary and conclusions.

1 Non-Microstructure Noise

Microstructure Noise (MN) may be viewed as a gauge-tolerance equivalent. It captures a variety of frictions inherent in the trading process, major among which are the bid-ask bounce, the discreteness of price changes, differences in trade sizes, the informational content of price changes, the gradual response of prices to a block trade, strategic components of the order flow, inventory control effects, processing costs, asymmetric information, auto-correlation in the order flow, stale prices, power laws, order cancellations, relative prices and relative depth profile, all of which have been dealt with in the existing literature. (Ait-Sahalia and Yu (2009), Jacod et al. (2017), Gould et al. (2013))

In contrast, non-microstructure noise (NMN) arises from sources different from friction in the trading process. Principal sources of non-microstructure noise are gaps in market design and incidences of systematic quote stuffing and spoofing in high frequency markets. Such noise can potentially distort the information content and statistical properties of high frequency data. It is first examined from the perspective of aggregate effects in double auction limit order, and conjectures the possible agent behaviour that leads into such effects.

1.1 Stochastic price and noise processes

Let the log price of an asset at time t, Y_t obey a semimartingale process on some filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, P)$. We consider the case where Y is a Brownian Semimartingale and a single period [0,T]. In the most general form Y is given as

$$Y_t = \int_0^t a_u du + \int_0^t \sigma_u dW_u + \alpha \int_0^t \sigma_v dW_v \tag{1}$$

where a_u is a predictable locally bounded drift, $\sigma_u$and $\sigma_v$are a cadlag volatility process, and $W_u$and W_v are Brownian motion. $\alpha = 1$ if V is mixed up with U, $\alpha = 0$ otherwise. The intuition is that V participates in the price process of an asset only if it is mixed up with U. In other instances it does not participate in the price process, although it is involved in the noisy observation as shown in (2). Note that in the above we could also work with a process that is BSMJ, Poisson or Hawkes process.

X_{ij} is a noisy observation of Y_{ij}.

$$X_{ij} = Y_{ij} + U_{ij} + V_{ij} \tag{2}$$

The evolution of Y is through X as an OU process. U is not independent of Y i.e $(U_{i0}, U_{i1}, ..., U_{in})$ are mutually independent but jointly dependent on Y. But U and V are independent. U represents the origins of MN and V that of NMN.

We assume U and V as noise but from different independent processes. U results from an Ornstein-Ulhenbeck (OU) process and V from an OU or Geometric Brownian Motion (GBM). Our object of interest is V. V may evolve independently and simply add up or at times it may get mixed up with U. We assume $E(V_{ij}) = 0$, and $Var(V_{ij}) = \omega^2$. However, at times in absence of V, there could also exist a white noise component from U (as we show later) as a result of its construction.

1.1.1 V evolves independently

When V independently adds up, it evolves from the following stochastic differential equations (SDE) (refer a similar treatment in Wilkinson (2010)):

$$dV = adt + \sigma dW_{2v} \tag{3}$$

1.1.2 V gets mixed up [1]

When V remains independent but gets mixed up, let us start with the following SDE,

$$dx(t) = ax(t)dt + \sigma_u x(t)dW_{ut} + \sigma_v dW_{vt} \tag{4}$$

$$\phi_t = \exp((a - (1/2)\sigma_v^2)t + \sigma_v W_{vt}) \tag{5}$$

(5) above is a geometric Brownian motion. Using integration by parts for Ito processes,

$$d(x(t)\phi_t^{-1}) = x(t)\phi_t^{-1}((-a + \sigma_v^2)dt + \sigma_v dW_{vt})$$

$$+\phi_t^{-1}(ax(t)dt + \sigma_u dW_{ut} + \sigma_v dW_{vt}) - x(t)\phi_t^{-1}\sigma_v^2 dt$$

$$= \sigma_u \phi_t^{-1} dW_{ut} \tag{6}$$

Since the two processes U and V are independent, the co-variance of the two Ito processes $x(t)$ and $\phi_t^{-1}[x(t), \phi_t^{-1}]_t = -\sigma_v^2 \int_0^t x(s)\phi_s^{-1} ds$.

[1] This solution is attributed to Nawaf Bou-Rabee, Department of Mathematical Sciences, Rutgers University Camden.

Now integrating what remains of (6) we get the solution as given below if initial condition is x_0 :

$$x(t) = \phi_t(x_0 + \sigma_u \int_0^t \phi_s^{-1} dW_{us}) \tag{7}$$

1.1.3 Noise in trade price series

The above two scenarios represent datasets that comprise of both trades and quotes or quotes alone. In a dataset that comprises only trades, V should be 0 as we do not expect non-microstructure noise in such data. However, even in such datasets, it is seen that removal of data improves the results. This phenomenon may be understood if one models the trades as evolving from an OU process. As we show below the OU process is the solution to an SDE. And the solution itself can be shown to comprise of the OU noise and a white noise. Removal of the white noise may possibly improve volatility calculations. Let the initial conditions be that $\xi_O U(0)$ is a gaussian random variable with mean 0 and variance $(2\tau)^{-1}$. The OU noise is solution of the following SDE:

$$d\xi_{OU}/dt = (-1/\tau)\xi_{OU}(t) + (1/\tau)\xi_w(t) \tag{8}$$

The solution (from San Miguel and Toral (2000)) to the above is given below, where H is a gaussian random process and can be shown to be a white noise:

$$\xi_{OU}(t+h) = \xi_{OU}(t)e^{-h/\tau} + H_h(t)$$

$$H_h(t) = \tau^{-1}e^{-(t+h)/\tau} \int_t^{t+h} ds\xi_w(s)e^{s/\tau} \tag{9}$$

Though intuitive there may be no motivation in general to clean a dataset of trades for outliers and noise. However, the above result indicates that conceptually, there may exist a case to remove white noise from trade price series as well.

Identification of non-microstructure noise

Quotes that contribute to non-microstructure noise can be identified by three indicators. First, if the quotes do not eventually participate in the auction. This is from Brownlees and Gallo and we shall develop it further in section 1.2 and section 2. Second, by a measure of the deviation of the quote from the center of the distribution of quotes (measured by the Euclidean distance of the quote from a measure of the centroid of the quote distribution). An established technique to identify outliers, this is further extended to handle outliers and noise processes in section 2. The third is from observation of visual patterns in quotes. Figure 1 gives the plots for raw trades and quotes. The arrows indicate stochastic processes and outliers that together represent non-microstructure noise. Although visual identification does not help in the removal of noise, it does confirm the presence of noise. In Figure 1, we observe patterns along the price axis and time axis. Some patterns are closer to the trade signal and auction process and others are clearly distant.

1.2 'Plausible market activity' and agent behaviour

In order to create a construct that identifies what constitutes plausible market activity in double auction limit order markets, we posit that quoting activity in such markets can be represented by queues that are refreshed at every tick. This refresh is based on a price and time priority. The time priority comes into play for quotes with identical prices.

Technically any quote that is not "Best Bid" or "Best Offer" does not have an opportunity to trade or get executed. Conversely, every quote that converts into a trade has to necessarily be the best bid/offer, even if instantaneously. However, the quoting pattern of agents in the auction is clearly conditional on the queue, the depth (quantity on bid and ask side and their distance), the arrival rate of quotes and service rate (execution of trades). Hence, quotes beyond best bid and ask can also participate in the auction.

When quotes are placed further from best bid/offer it indicates an expectation of higher return. On occasion, it could also signal an informed trader. The agent necessarily trades off this expectation with the risk of non-execution. In the event of non-execution the agent is expected to modify his bid/offer or cancel and place a new order. There should exist a possibility for the new order or modified order to move into the best bid/offer position.

Two scenarios emerge, any of which could unfold during the bid-ask bounce. First, that the agent holds their return expectation and the market players on the opposite side move to their position. This scenario is possible if the current position of the agent in the demand/supply curve is crossing the price-quantity schedule of agents on opposite side or the stochastic evolution of their price bids/asks over time. Second, the agent moves along their price-quantity schedule or stochastically evolving price over time and offers the best bid/offer matching or closer to the opposite best offer/bid.

Ignoring the incidence of quote stuffing and spoofing where the intent to trade is questionable, there exist quotes where the demand and supply schedules are distant from the trade signal and inflexible enough to not participate in the auction or price process. Some exceptions can exist where quoting activity commingles with microstructure noise and appears to jointly participate in the auction process. These are observable during phenomena like early hour trade volatility, block deals and jumps.

Our purpose is not to provide an economic model for outliers, but to distinguish the behaviour of quotes that participate in the auction from those that do not. Let $\{N_i(t),\ t\epsilon[0,\infty)\}$ be a stochastic process, $N_i(t),\ \epsilon\{0,1,...\}$ representing the number of discreet ticks in which an order has been placed in the exchange (double auction) by the ith agent upto time t. $A_{in}(t_{in})$ be the amount demanded (or supplied) by agent i at his nth tick that occurs at time t_{in}. The composite process,

$$X_i(t) = \sum_{n=1}^{N_i(t)} A_{in}(t_{in}) \tag{10}$$

represents the total amount demanded / supplied by the ith agent in the interval [0,t=T].[2]In our model of limit order markets, agents are classified according to those whose behaviour represents market activity and those whose do not. For the former, we continue with the above notation, whereas for the others we introduce the subscript 'nmn' (representing non-microstructre noise).

$$Xnmn_i(t) = \sum_{n=1}^{N_i(t)} Anmn_{in}(t_{in}) \tag{11}$$

[2] This representation is similar to Garman (1976)

The two groups can be distinguished in two aspects. Firstly, there is an inelasticity between the stochastically evolving price of the asset (Y_t) and quantity (demanded/supplied) $Xnmn_i(t)$ in case of the NMN group. While theoretically an auction may be possible using the price-quantity schedules of the NMN group, practically this group does not participate in the auction because the market liquidity prevents such scenarios from emerging. Secondly, the NMN group seems to be demanding a return far higher than what the dominant agents in the market are prepared to offer. If we denote the stochastic asset price of NMN group by $Ynmn_t$, then there exists the following condition at all times,

$$|Y_t - Ynmn_t| > 0 \tag{12}$$

We use the above intuitions to identify and remove NMN in the next section.

2 A Paradigm to handle UHF data in presence of NMN

Though the existing literature, Brownlees and Gallo (2006) and Barndorff-Nielsen et al. (2009) deals with data in a comprehensive manner, the approach presented here is additionally diagnostic in nature. Figure 2 illustrates our intent with the new paradigm. The first plot gives the raw data for MSFT (NASDAQ, 30-Oct-2017), the second plot is the data after it has been handled with BNHLS method and the third is the new paradigm.

The data is restricted to the general market hours of 9.30 am to 4.00 pm. Pre-preparation of data, specific to the dataset, if required, can be done at this stage. For trades only start directly with step 3.

2.1 Reconstructing the limit order book to identify and remove NMN

This step, called 'LOBclean' involves reconstruction of the limit order book. It involves recreating the auction at every tick, identifying the best bid, best ask and recording all the messages (add orders, trade executions, deletions, modifications etc) in the book. Specifically for 'LOB-clean' we need to identify the price level at which every quote is placed and its trajectory in the book before it is cancelled, modified or executed. The trajectory is identified in terms of the price level at which it exists at every tick while it is still in the book.

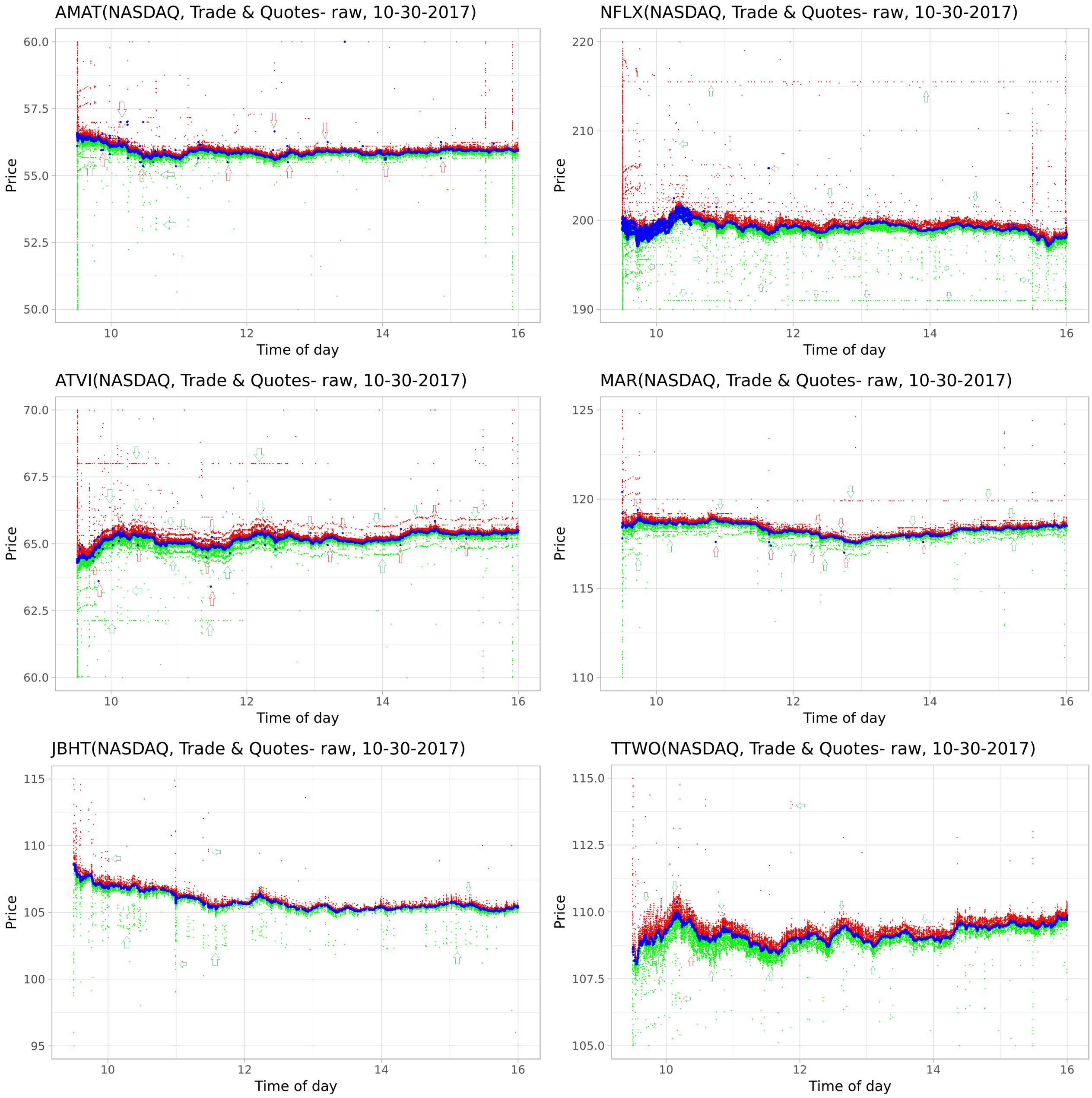

Figure 1: Plots of trade and quotes from NASDAQ, 30-Oct-2017. Price in USD and time in hours Trades represented in blue, bids in green and ask in red. Arrows in red point to trades and arrows in green to the quotes. Although the activity on bid and ask sides can be symmetrical at times, they are more often different, suggesting different stochastic processes in action.

The next step is the application of the concept of plausible market activity to the data. Using a conservative definition (based on observation from a number of experiments), a quote that begins its journey in the book beyond the tenth price level and never comes within the top ten price levels is classified as NMN. Once identified, such quotes are removed from the dataset. We note that even with a conservative definition, the LOBclean step is a highly effective method to remove non-microstructure noise as we shall see in the empirical illustration in section 3. This price level could be revised if the researcher needs to aggressively clean out data that are distant from the best bid and best ask.

Price levels in the limit order book

The distance of a quote from the best bid /offer is given in terms of the price levels. The price levels in a limit order book are a natural distance classifier. This intrinsic scale is used to check for participation in market activity. We assume that the intent of every quote is to result in a trade. In limit order markets, quotes represent a trade off between expected returns and the risk of not getting traded. If a quote does not participate in the auctioning process, it does not represent plausible market activity as noted in section 1 above.

Impact of an aggressive definition for LOBclean

Beyond a particular level of data removal, the risk of information loss through removal of U starts to emerge. To illustrate the impact of the definition for market activity, in one experiment we defined as noise the quotes placed beyond fifth price level. We also did not take cognisance of whether the quote moved into lower price levels or got traded. This is an example of an aggressive stance. Table I gives the results for AAPL (NASDAQ) data for 30-Oct-2017 and contrasts with the definition we have offered in step 1 above. The aggressive criteria classifies 88% of quotes as not representing market activity. This set also includes 50% of quotes that result into trades. The conservative criteria identifies 3.9% of quotes as noise.

Table I : APPL (NASDAQ) LOB 5th price level without recovery vs 10th price level with recovery

Message	NASDAQ Specification	Message type	Number of Raw Messages	Quotes that start beyond 5th level	%	Quotes that never enter the 10 best price levels	%
A	Add Order Message	Quote	517109	455017	87.99	19679	3.81
F	Add Order Message	Quote	4317	4307	99.77	2544	58.93
U	Order Replace Message	Quote - modification	46493	39738	85.47	115	0.25
		Total	567919	499062	87.88	22338	3.93

2.2 Separating the bid and ask price series to identify the nature of distributions

Bids and asks represent different stochastic processes. The two price series may be driven by non-identical influence from the fundamental price signal, microstructure noise, intraday volatility and asynchronous price revisions. In step 3, we present a bouquet of methods to remove NMN. Some of the methods are more effective in case of normal distributions and others in case of non-normal distributions. A QQ plot, that is a scatter plot between the data and a theoretical normal distribution, is a useful tool to check normality. The QQ plot can be used to identify non-normal distribution and chose the appropriate method in step 3. Further, it helps in deciding the trade off between efficiency gain and data removal.

The timestamp correction step is an important step for several studies. Essentially, the raw data consists of a number of datapoints bearing the same time stamp. Following BNHLS we have taken the median of the bids and asks to represent the bid and ask for the timestamp. Some of the other options have been outlined in Barndorff-Nielsen et al. (2009). To these alternatives, we would recommend future studies to explore choosing the incoming quote with the highest bid or lowest ask to represent the bid and ask at that timestamp.

Since all modern datasets clearly identify the bid and ask quotes, we can separate the two to form independent series. This is an important preparatory step for step 3. While we attempt to identify the center of the data in step 3 and clean outliers and noise that are farthest from the center, separate bid and ask series can help because the center of both these datasets is different. In case, the study needs a single price series, we combine the two again after step 3.

2.3 Noise identification and removal based on distance from center

The bid and ask price series are handled separately in this step. We choose from the solution set comprising MADMean, BNHLS, MADMed for symmetric distributions and S_n, Q_n, $ScaleTau2$ for assymetric distributions. The methods for symmetric distributions are based on mean absolute deviation or median absolute deviation or a combination of the two. The methods for assymetric distributions selected in the solution set have been designed for robust estimations even in case of assymetric distributions. An aspect that distinguishes the four methods we include into the basket for UHF financial data is the way we handle the methods. Since these methods are used upon the separated bid and ask price series, we do not treat the data locally (that is, in a neighbourhood of datapoints) and hence there is no subjective decisions taken on the basis of the dataset.

Let $p_i(i = 1toN)$ be an ordered tick-by-tick price series.

2.3.1 Methods for symmetric distributions

Mean Absolute Deviation

Brownlees and Gallo (2006) propose the following method:

$$(|x_i - \bar{x}_i(k)| < 3si(k) + \gamma), \tag{13}$$

true observation i is kept, false observation i is removed, where $\bar{x}_i(k)$ and $si(k)$ denote respectively the δ -trimmed sample mean and sample standard deviation of a neighborhood of k observations around i and γ is a granularity parameter. The neighborhood of observations is always chosen so that a given observation is compared with observations belonging to the same trading day. That is, the neighborhood of the first observation of day are the first k ticks of the day, that of the last observation of the day are the last k ticks of the day, the neighborhood of a generic transaction in the middle of the day is made by approximately the first preceding k/2 ticks and the following k/2 ones, and so on. The neighbourhood function is linearly logical.

A percentage of trimming δ, directly proportional to the frequency of outliers is chosen. The parameter k should be chosen on the basis of the level of trading intensity. If the trading is not very active k should be "reasonably small", so that the window of observations does not contain extreme prices (the contrary is true if the trading is very active). The role of the γ parameter is to avoid zero variances produced by sequences of k equal prices. The choice of γ should be a multiple of the minimum price variation allowed for the specific stock. We note that this method is high on subjective implementation.

BNHLS

The BNHLS method combines Mean Absolute Deviation and Median Absolute Deviation as a secondary method of cleaning. The cleaning band they employ is 10 times of the deviation local mean of 50 observations (not including the observation under study) from the median absolute deviation of the dataset. This step however is preceeded by their primary method that is based on the spread represented by the quote. A datapoint is considered an outlier if its spread is 50 times the median spread of the day. These two steps when combined one after the other, represent an effective data handling technique.

Median Absolute Deviation (Hampel (1974)

This estimator is the median absolute deviation about the median. It is given by:

$$MADmedian = b * med_i |x_i - med_j x_j| \tag{14}$$

The MAD has the best possible breakdown point at 50 percent. Its influence function is bounded.The constant b is needed to make the estimator consistent for the parameter of interest. For Gaussian distributions we need to set b = 1.4826. For each $x_i$we declare it as an outlier if its distance from the MADMedian is over 3. i.e if,

$$|x_i - med_j x_j| > 3 * MADmedian. \tag{15}$$

While the MADMedian is an extremely useful estimator, it has some drawbacks. Its efficiency at Gaussian distributions is low and since it gives equal importance to positive and negative deviations from the median, it takes a symmetric view on the dispersion. This makes it less useful for highly skewed distributions, a situation where the following three options are better.

2.3.2 Methods for Asymmetric Distributions

S_n (Rousseeuw and Croux (1993))

The estimator $S_n$is given as :

$$S_n = c * med_i(med_j|x_i - x_j|) \tag{16}$$

In (13) above for each i we compute the median of $(xi - xj(;j = 1,...,n)$. The median of the n numbers we obtain gives our final estimate S. The factor c has a default value of 1.1926 and is used for consistency. We consider any x_i as outlier if its distance from the median is greater than 3 times Sn.

$$|x_i - med_j x_j| > 3 * Sn. \tag{17}$$

The outer median in (14) is a low median, which is the order statistic of rank $(n+1)/2$, and the inner median is a high median, which is the order statistic of rank $(n/2)+1$.

Q_n (Rousseeuw and Croux (1993))

The estimator Q_n is given as :

$$Q_n = d * (|x_i - x_j|; i < j)_{(k)} \tag{18}$$

where d is a constant factor, 2.21914 and $k = \binom{h}{2} \approx \binom{n}{2}/4, h = (n/2)+1$, here k refers to the pairs to compute the pair-wise distances.

Q_n has a 50 percent breakdown point and is suitable for asymmetric distributions. Both, S_n and Q_n are robust but computationally intensive. However, Croux and Rousseeuw have constructed fast algorithms for them. These have also been implemented in the R package "robustbase", making computation easier.

ScaleTau2 (Maronna and Zamar (2002))

The robust τ (tau)-estimate is :

$$s(X)^2 := s_0^2 * (1/n) \sum_i \rho_c 2((x(i) - \mu(X))/s_0), \tag{19}$$

where $\rho_c(u) = min(c^2, u^2)$.

In the event that research study requires it, the bid and ask price series can be recombined at this stage using the timestamp available with the tick data. This is presented in the empirical illustration in the next section.

3 Empirical Illustration: Realised Kernels in the absence of NMN

We empirically illustrate our proposed paradigm by estimating volatility using Realised Kernels following Barndorff-Nielsen et al. (2009). Our proposed paradigm to handle NMN is expected to improve the estimation of realised kernels. The choice of volatility estimation using realised kernels to illustrate our paradigm is deliberate. Firstly, the realised kernel of BNHLS is sensitive to data handling. Secondly, BNHLS have given us a benchmark and sophisticated technique for data handling. We use results obtained using this technique as a benchmark to compare any improvements obtained by using our proposed methods. Elements of data handling are also adopted from BNHLS.

The dataset comprises of a sample of 10 stocks, one from each decile of NASDAQ 100 ranked on the basis of business time. Business time is defined as the number of messages in the data sample of *NASDAQ Itch* for the trading day 30-October-2017. These messages includes the quotes, quote modifications, trades and quote deletions. We first rank the NASDAQ 100 stocks on the basis of business time. We select the first stock from each decile. [3]

[3] MSFT is the second stock in the first decile has been selected over APPL due to dataset issues. We do not expect this to influence results in anyway.

3.1 Design

The research design comprises estimating realised variance at tick level and realised kernel at tick level for trade data and midquote data. The realised kernel is estimated as per the technique in Barndorff-Nielsen et al. (2009). There is no difference in the data handling for trade data between our method and BNHLS. The important step is to parse out multiple trades with the same timestamp and replace them with their median value. Since trades act as signals, the estimate of realised values for the trade acts as a benchmark. In their work BNHLS have shown that the estimate for realised kernel for trades and midquotes was close, demonstrating the robustness of the realised kernel as an estimate of volatility. Given the difference in midquote price series construction (as discussed later) this result is not expected in the present estimation.

As a first step, the realised kernel values are estimated using the original method from BNHLS so as to compare with the estimates we obtain after removal of NMN. This estimation is referred to as BNHLS.

Second, the original method from BNHLS is reinforced with LOBclean step given in section 2. This involves identifying and cleaning data that never enter the top 10 price levels in the limit order book. There is no ambiguity about this data being noise. In the process, we can evaluate any improvement in the estimation of the realised values. This estimation is referred to as BNHLS+LOBclean.

Third, the realised values are estimated using our paradigm. This involves the limit order book reconstruction and cleaning, separating the bid and ask price series, cleaning data that are beyond the band defined by the techniques in section 2, and finally, joining the two series to get a single midquote price series. While we evaluate the distribution of the bid and ask price series, we carry out the estimation using MADMed, Sn, Qn and ScaleTau2 to compare their performance.

3.2 Data Handling

The data handling steps are summarised in Table II for all the estimating procedures. Data handling for quote preparation involves 5 to 6 distinct stages. Correction for trading time and multiple datapoints in the same timestamp are common steps. The BNHLS based steps have spread correction and later a data cleaning based on a combined Mean absolute deviation and median absolute deviation. Methods in our paradigm involve, limit order book modelling for noise removal and separation of bid and ask price series. The data cleaning procedures are applied to the separated series. LOBclean is also included in the reinforced BNHLS method.

Table II: Morphology of Data Handling Steps.

Data Handling Step	Trade	BNHLS	BNHLS + LOBclean	MADMed	Sn	Qn	ScaleTau2
1. Correction for Market Timing	Y	Y	Y	Y	Y	Y	Y
2. Removal of Erroneous data		Y	Y	Y	Y	Y	Y
3. Correction for Multiple datapoints with same Time Stamp	Y	Y	Y	Y	Y	Y	Y
4. Data Cleaning using Spread as Criterion		Y	Y				
5. Data Cleaning using BNHLS Mean Absolute Deviation from MAD (Median)		Y	Y				
6. Noise Removal in the Limit Order Book			Y	Y	Y	Y	Y
7. Separation of Bid and Ask data series				Y	Y	Y	Y
8. Data Cleaning using Median Absolute Deviation as Criterion				Y			
9. Data Cleaning using Sn as Criterion					Y		
10. Data Cleaning using Qn as Criterion						Y	
11. Data Cleaning using ScaleTau2 as Criterion							Y

3.3 Data Description

We are aware at the outset of the existence of serial correlation. However what is of interest is not the existence of but the nature of the autocorrelation. The analysis of autocorrelation is important to understand the stochastic processes of U and V. Market microstructure noise (U) induces autocorrelation in the intraday returns and it can be observed in the price series as well. This autocorrelation is the source of bias in realised variance estimation Hansen and Lunde (2006). Hansen and Lunde also note that "prewhitening" of intraday returns and kernel based estimators aid the estimation process. We would want to check if there is a difference in influence of V. Our assumption in section 1 was that U and V are independent or mixed stochastic noise processes. The experimental design gives us several datapoints to investigate the role of autocorrelation and the stochastic processes. The impact of LOB cleaning, time cleaning step in trade prices, comparative impact of BNHLS and MADMed (or Sn, Qn, ScaleTau2) data cleaning can throw insights into the stochastic processes.

Table III, summarises our findings with respect to autocorrelation in trade and midquote price series for our sample of stocks. It also juxtaposes the observations against the distributions in the bid and ask price series we observed. Figures 3 to 12 (Refer Appendix) present the autocorrelation function for our sample of stocks. The figures give the autocorrelation in raw trade price series, time cleaned trade price series, raw midquotes, LOBclean midquote price series, price series after BNHLS data cleaning and price series of the MADMed method. The latter serves as a representative method from our paradigm. Although, some consistent observations can be made, these figures indicate the presence of diverse stochastic processes.

3.3.1 Trades

The time cleaning step in Trade price series has a mixed impact. It may induce an increase, decrease or no change in autocorrelation. This step in trade price series handling merits a deeper discussion. The possible reasons for such timestamp clustering is not known to us. As Table IV shows, the loss in datapoints from transaction price signal ranges from 26% in MAR to 44% in HOLX and MSFT. We could not find any way to prevent such a high loss in signal. The need to have a single price value at a given timestamp is important to further generate the

Table III : Autocorrelation in the trade and midquote return series of the sample stocks.

Stock (percentile)	**Autocorrelation function (Trades)**	**Autocorrelation function (Midquotes)**
MSFT (99)	First order negative. Time clean induces higher order autocorrelation	First order negative. LOBclean takes out autocorrelation from 2nd to 10th lag. BNHLS and MADMed clean up higher order autocorrelation. Higher order auto correlation is from V.
AMAT (90)	First order negative	First order negative and Third order positive. LOBclean takes out the third order serial dependence (we infer this to be V). BNHLS and MADMed remove higher order autocorrelation.
ATVI (80)	First order negative. Timeclean removes some higher order autocorrelation.	First order negative autocorrelation and weak lower order. LOB cleaning removes lower order except first order. In BNHLS and MADMed it is completely removed except 1first order. Wavelets seem to be V.
NFLX (70)	First order negative autocorrelation.	Autocorrelation present upto lag 20. Strong first order negative autocorrelation. LOBclean cleans up several lower order autocorrelation but not first order. MADMed clean up all auto correlation except lag 1. BNHLS misses out the first 20 lags.
PCLN (60)	First order negative autocorrelation	Strong first and second order acf. LOBclean impacts the second order autocorrelation. MADMed and BNHLS clean up autocorrelation other than first and second order.
MAR (50)	Strong first order autocorrelation and other weaker autocorrelations in raw and time clean trades.	First order negative autocorrelation and higher orders also present. However LOBclean increases autocorrelation of higher orders. BNHLS and MADMed cleans up higher order autocorrelation.
TTWO (40)	First order negative and weak second order positive autocorrelation. Higher orders autocorrelation also present.	First order negative autocorrelation. BNHLS and MADMed remove all higher order autocorrelation.

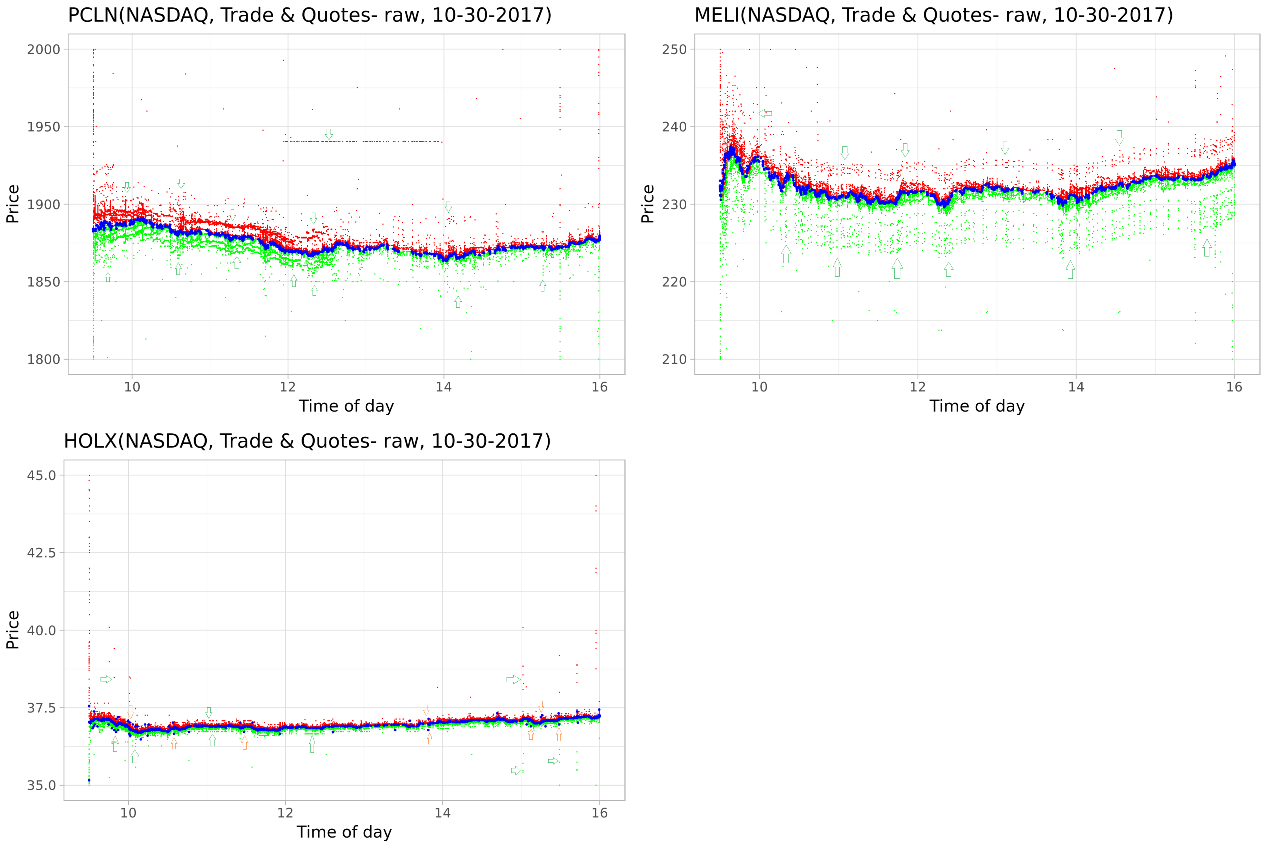

Figure 1 (Contd...): Plots of trade and quotes from NASDAQ, 30-Oct-2017. Price in USD and Time in hours.

Table III (...contd): Autocorrelation in the trade and midquote return series of the sample stocks.

Stock (percentile)	**Autocorrelation function (Trades)**	**Autocorrelation function (Midquotes)**
HOLX (30)	First order negative autocorrelation. No impact of time cleaning.	First order negative autocorrelation. other orders present but not significant. LOBclean has a mixed impact. BNHLS and MADMed have usual impact cleaning up higher order autocorrelation.
JBHT (20)	First order negative autocorrelation but not very strong. higher orders present and LOBclean increases it.	First order negative autocorrelation. BNHLS and MADMed increase the second order. wipe out others.
MELI (10)	Autocorrelation of lower orders and middle orders. Timeclean has mixed impact.	Strong first order negative autocorrelation. No impact of LOBclean. BNHLS and MADMed decrease the first order autocorrelation by 25%. Wipe clean higher order autocorrelation.

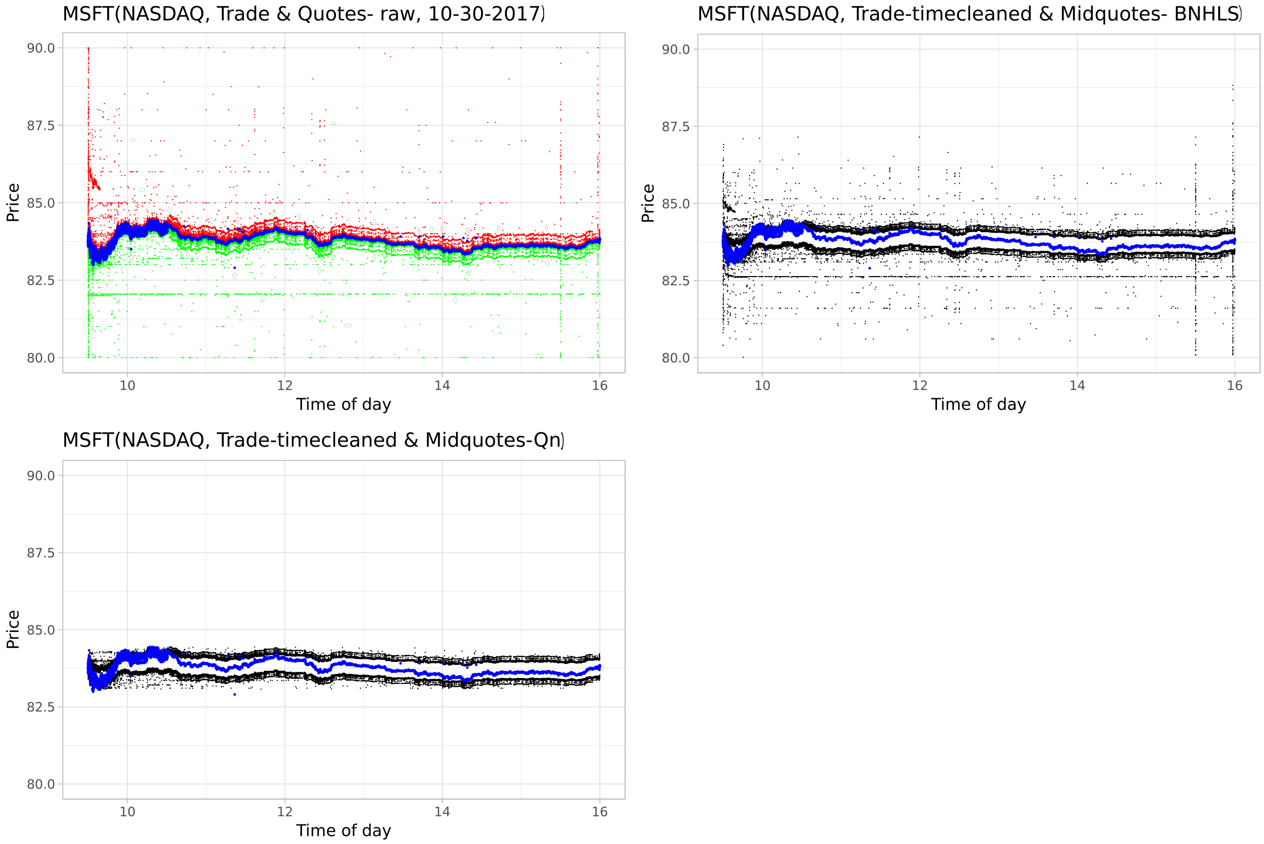

Figure 2: The top left panel plots trades and quotes of MSFT, NASDAQ, 30-Oct-2017 and the top right, the data cleaned with BNHLS method. Trades are represented in blue in both plots. Bids are green and asks in red. Midquotes in black. Green arrows point to quotes and red arrows to trades. In right some stochastic processes still remain after data cleaning using the method of BNHLS. Contrast this with the bottom left panel plot in which data is handled using one of our proposed methods.

return series. Where a return series is not used and in case of tick sampling, one may consider not using this step. Some trade price execution (messages 'X' and 'E' in *NASDAQ Itch*) can be linked to orders using the unique order reference number. Hence, these ticks although with the same timestamp are distinguishable.

3.3.2 Midquotes

The original and earliest way of preparation of a midquote data series is to take the mid of the best bid and best ask. This is adapted into limit order market to be the highest bid and lowest ask at all ticks. The challenges with this manner of midquote series preparation are: First, while the data series so prepared will be a pseudo-trade data series, it does not get influenced by the incoming quotes, unless they are the best priced quotes. Second, the issue of stale quotes, that is, the quote included in the series may be redundant by the time it is included in the series. Third, the non-synchronous updation of bids and asks.

There are two other alternatives. One could use the midquote to represent the mid point of the latest bid and latest ask. The problem here is that the midquote series loses sight of the trade signal. Although the volatility profile is closer to the trade series.

The second alternative is to use the mid of the quote coming in the latest tick and the opposing best bid / offer. We have chosen this as this solves the issue of the connect with the trade series and also the issue of stale quotes. This series by its very construction is expected to have a higher volatility profile than other series. If the objective is to demonstrate the proximity of the estimation from trades and the midquotes this may not be suitable. However, with the dataset having autocorrelation and heteroskedasticity, this technique of midquote data series preparation seems promising.

3.3.3 Autocorrelation

We find first order negative autocorrelation in the midquote price series of all stocks. Some variations such as third order positive autocorrelation in AMAT, autocorrelation upto 10 lags in MSFT and upto 20 lags in NFLX and second order in PCLN. With the exception of HOLX, data handling techniques have not had any impact on the negative first order autocorrelation.

There is a 25% reduction in case of HOLX and some minor reduction in several other stocks. We infer the first order negative autocorrelation to be the main source of autocorrelation and resulting from microstructure noise U. Apart from the first order autocorrelation, all higher order autocorrelation are the result of non microstructure noise (V) or a mix of U and V. Data handling methods employed in BNHLS and our proposed methods (as illustrated in MADMed) remove autocorrelation emanating from V. BNHLS and MADMed (and Sn, Qn and ScaleTau2) clean up higher order autocorrelation. An exception exists in case of NFLX where BNHLS fails to remove autocorrelation in first 20 lags.

3.3.4 LOBclean

The LOBclean step has mixed results with higher order autocorrelation. This method has no impact on the first order autocorrelation, cleans up lower order autocorrelation from 2nd upto 20th in case of the first five deciles (MSFT, AMAT, ATVI, NFLX and PCLN), in MAR it increases higher order autocorrelation and in others it has negligible impact. We see the operation of mixed processes of U and V in the larger stocks and only V in the smaller stocks. This step behaves like a control group in our design. Since, we are sure about characterising this step as impacting V alone, we can infer the stochastic processes in action. When LOBclean has no impact on autocorrelation, we infer an independent V. When it increases autocorrelation, it indicates a mixed process, where clean up removes V and leaves U alone. When it decreases autocorrelation, the autocorrelation originates from V alone or from a mixed U and V process where both get removed.

3.3.5 McLeod-Li test for the ARCH effect

High frequency quote and trade data are known to display volatility clustering. McLeod and Li (1983) proposed a formal test for ARCH effect based on the Ljung-Box test. It looks at the autocorrelation function of the squares of the pre-whitened data, and tests whether the first L autocorrelations for the squared residuals are collectively small in magnitude. Since we want to test the data series directly, this suits our purpose. To illustrate, figure 13 (Appendix) shows the results of the McLeod-Li test for MSFT and NFLX. The null hypotheses of no ARCH effects is rejected. The results for all the stocks in our sample are similar, that is, the null hypotheses of no ARCH effects is rejected.

3.4 Results and Discussion

Table V summarises the results of the estimation of realised variance (RV) with tick level sampling and the realised kernel (RK) estimation. The RK estimation is as per BNHLS (2009). BNHLS serves as the benchmark for the performance of our data handling method. Movement in the direction of the estimate from trades data would be construed as a performance or efficiency improvement.

3.4.1 Trades

The volatility estimates (RV-tick and RK) from trade data for our sample stocks is low. There could be a downward bias as a result of the presence of negative serial correlation of the first order and second order and heteroskedasticity. This could also have resulted from the increase in autocorrelation we had observed after the timestamp correction stage.

3.4.2 Midquotes

Table VI shows the improvement in estimation achieved by our proposed methods (identified by the last step in the method MADMed, Sn, Qn, ScaleTau2) over the benchmark BNHLS.

Our focus is on the estimation of RK. BNHLS when reinforced by LOBclean step results into an improvement in estimation efficiency ranging from 3.6% in JBHT to 17.66% in PCLN. The average improvement is 8.96% with an average data loss of 0.6%. With additional loss of data ranging from 2.54% to 3.23%, the methods MADMed, Sn, Qn and ScaleTau2 can on an average improve efficiency over 25%. The efficiency improvement achieved by the 4 methods range from 13.15% to 43.75%. As expected, given the nature of distributions, Qn performs best in efficiency improvement, among the four methods, in 8 out of 10 stocks. MADMed performs better in ATVI and NFLX. The best performing method successfully removes fat tails, discontinuities and jumps. This is observed in all samples without exception. The improvement in estimation efficiency, thus, has a statistical basis and is not incidental. The tails in the data is mostly V. BNHLS is not able to handle the tails as effectively. We had noticed this visually in section 2 (figure 2). If the distribution is normal, BNHLS and MADMed can be effective methods. Although, the four methods outperform the benchmark, the absolute distance between the four methods themselves is not high. The improvement in estimation efficiency is high. But it comes at a cost of data loss.

The data loss in the four methods ranges from a low of 0.88% to high of 10.56%. Given the struggle for model fit and estimation efficiency in high frequency market microstructure research, we believe it is a favourable trade off. Let us now evaluate what happens if we reward less loss of data in calculating efficiency. This is a theoretical exercise as by design the lost data is mostly non-microstructure noise and the above trade off is a huge gain.

Table VII gives the efficiency addition in RK for every 1 % loss of data over benchmark (BNHLS). This measure is obtained by simply normalising the improvement in efficiency of estimation over benchmark with the loss in data over the benchmark. The average of each of the 5 methods ranges between 16.03 (BNHLS+LOBclean) to 23.21 (Sn). The stock level efficiency addition ranges from a low of 1.29 (AMAT, MADMed) to a high of 97.51 (MAR, MADMed). Using this yardstick, BNHLS+LOBclean is a better method in 5 out of 10 stocks in our sample and MADMed is better in 3 out of 10 stocks.

Table IV : Loss of data from Timecleaning step for trade price series

Stock (percentile)	**Trades before cleaning**	**Trades after timestamp cleaning**	**% loss of data**
MSFT (99)	47892	26677	44%
AMAT (90)	12947	7926	39%
ATVI (80)	15499	8838	43%
NFLX (70)	19220	13282	31%
PCLN (60)	4111	2362	43%
MAR (50)	4524	3351	26%
TTWO (40)	5789	3351	42%
HOLX (30)	4259	2375	44%
JBHT (20)	5243	3498	33%
MELI (10)	4124	2927	29%

Table V: Realised Variance and Realised kernel estimation results. The data handling methods proposed in this paper produce improved estimation in each instance. The best performing method (highlighted with a box) 20% of times comes from a method for symmetric distributions.

			For symmetric distributions			**For asymmetric distributions**		
Stock (percentile)	**Estimation**	**Trades**	**BNHLS**	**BNHLS+ LOB-clean**	**MADMed**	**Sn**	**Qn**	**ScaleTau2**
MSFT (99)	RV-tick	0.0395	6.1922	5.6472	4.8897	4.8906	4.8862	4.8886
	RK	0.0198	3.0980	2.8238	2.4451	2.4456	$\boxed{2.4434}$	2.4445
	Datapoints	26677	451242	448857	447236	447261	447198	447233
AMAT (90)	RV-tick	0.0163	3.5872	3.2978	3.0982	3.1153	3.1109	3.1121
	RK	0.0082	1.7937	1.6490	$\boxed{1.5493}$	1.5578	1.5556	1.5562
	Datapoints	7926	203114	202152	181674	188101	186740	186840
ATVI (80)	RV-tick	0.0041	5.5817	5.2507	4.7080	4.7129	4.6964	4.7050
	RK	0.0021	2.7912	2.6255	2.3541	2.3566	$\boxed{2.3484}$	2.3526
	Datapoints	8838	145754	145045	142073	142683	141657	142042
NFLX (70)	RV-tick	0.0431	3.9910	3.3262	2.4350	2.5193	2.5347	2.5428
	RK	0.0217	1.9963	1.6631	$\boxed{1.2176}$	1.2598	1.2674	1.2715
	Datapoints	13282	106413	104747	100873	102769	103102	103235
PCLN (60)	RV-tick	0.0001	0.5005	0.4127	0.3069	0.3063	0.3038	0.3042
	RK	0.0001	0.2506	1.6631	0.1535	0.1532	$\boxed{0.1519}$	0.1521
	Datapoints	2362	56258	55818	55654	55650	55631	55634
MAR (50)	RV-tick	0.0011	0.0512	0.0488	0.0299	0.0298	0.0288	0.0290
	RK	0.0006	0.0256	0.0245	0.0150	0.0149	$\boxed{0.0144}$	0.0145
	Datapoints	3351	73472	73338	73160	73149	73081	73096
TTWO (40)	RV-tick	0.0002	1.7142	1.6107	1.4734	1.4723	1.4684	1.4701
	RK	0.0002	0.8572	0.8055	0.7367	0.7362	$\boxed{0.7343}$	0.7351
	Datapoints	3949	65393	65124	64720	64699	64625	64660
HOLX (30)	RV-tick	0.0156	0.3666	0.3338	0.3063	0.3047	0.3030	0.3041
	RK	0.0079	0.1832	0.1668	0.1533	0.1525	$\boxed{0.1516}$	0.1521
	Datapoints	2375	52497	52289	52240	52228	52185	52215
JBHT (20)	RV-tick	0.0002	1.8028	1.7376	1.5041	1.4984	1.4896	1.5392
	RK	0.0003	0.9016	0.8691	0.7522	0.7493	$\boxed{0.7449}$	0.7697
	Datapoints	3498	43896	43691	41362	41252	40896	42097
MELI (10)	RV-tick	0.0008	3.6151	3.2718	2.3168	2.3008	2.2946	2.3555
	RK	0.0008	1.8183	1.6451	1.1590	1.1510	$\boxed{1.1480}$	1.1784
	Datapoints	2927	31627	31386	30262	30247	30217	30483

Table VI: Efficiency addition in RK for unit loss of data over benchmark (BNHLS) (Efficiency change with 1% loss of data)

Technique Sample Stock (Percentiles)	**BNHLS+LOBclean**			**MADMed**		
	Absolute Efficiency improvement (A)	**Data Removal (B)**	**Normalised Efficiency improvement (C=A/B)**	**Absolute Efficiency improvement (A)**	**Data Removal (B)**	**Normalised Efficiency improvement (C=A/B)**
MSFT (99)	-8.85%	-0.53%	16.75	-21.07%	-0.89%	23.74
AMAT (90)	-8.07%	-0.47%	17.03	-13.63%	-10.56%	1.29
ATVI (80)	-5.94%	-0.49%	12.20	-15.66%	-2.53%	6.20
NFLX (70)	-16.69%	-1.57%	10.66	-39.01%	-5.21%	7.49
PCLN (60)	-17.66%	-0.78%	22.58	-38.75%	-1.07%	36.09
MAR (50)	-4.30%	-0.18%	23.56	-41.41%	-0.42%	97.51
TTWO (40)	-6.03%	-0.41%	14.66	-14.06%	-1.03%	13.66
HOLX (30)	-8.95%	-0.40%	22.59	-16.32%	-0.49%	33.34
JBHT (20)	-3.60%	-0.47%	7.72	-16.57%	-5.77%	2.87
MELI (10)	-9.53%	-0.76%	12.50	-36.26%	-4.32%	8.40
Average	-8.96%	-0.61%	16.03	-25.27%	-3.23%	23.06

Technique Sample Stock (Percentiles)	**Sn**			**Qn**		
	Absolute Efficiency improvement (A)	**Data Removal (B)**	**Normalised Efficiency improvement (C=A/B)**	**Absolute Efficiency improvement (A)**	**Data Removal (B)**	**Normalised Efficiency improvement (C=A/B)**
MSFT (99)	-21.06%	-0.88%	23.87	-21.13%	-0.90%	23.58
AMAT (90)	-13.15%	-7.39%	1.78	-13.27%	-8.06%	1.65
ATVI (80)	-15.57%	-2.11%	7.39	-15.86%	-2.81%	5.64
NFLX (70)	-36.89%	-3.42%	10.77	-36.51%	-3.11%	11.73
PCLN (60)	-38.87%	-1.08%	35.96	-39.39%	-1.11%	35.34
MAR (50)	-41.80%	-0.44%	95.07	-43.75%	-0.53%	82.21
TTWO (40)	-14.12%	-1.06%	13.30	-14.34%	-1.17%	12.21
HOLX (30)	-16.76%	-0.51%	32.70	-17.25%	-0.59%	29.02
JBHT (20)	-16.89%	-6.02%	2.80	-17.38%	-6.83%	2.54
MELI (10)	-36.70%	-4.36%	8.41	-36.86%	-4.46%	8.27
Average	-25.18%	-2.73%	23.21	-25.57%	-2.96%	21.22

Table VI (...contd): Efficiency addition in RK for unit loss of data over benchmark (BNHLS) (Efficiency change with 1% loss of data)

Technique		**ScaleTau2**	
Sample Stock (Percentiles)	**Absolute Efficiency improvement (A)**	**Data Removal (B)**	**Normalised Efficiency improvement (C=A/B)**
MSFT (99)	-21.09%	-0.89%	23.74
AMAT (90)	-13.24%	-8.01%	1.65
ATVI (80)	-15.71%	-2.55%	6.17
NFLX (70)	-36.31%	-2.99%	12.16
PCLN (60)	-39.31%	-1.11%	35.44
MAR (50)	-43.36%	-0.51%	84.73
TTWO (40)	-14.24%	-1.12%	12.71
HOLX (30)	-16.98%	-0.54%	31.60
JBHT (20)	-14.63%	-4.10%	3.57
MELI (10)	-35.19%	-3.62%	9.73
Average	-25.01%	-2.54%	22.15

Table VII : Efficiency addition in RK for unit % loss of data over benchmark (BNHLS). To calculate this measure, start with the outperformance over the benchmark and then normalise with the additional loss of data.

Sample Stock (Percentiles)	**BNHLS + LOB-clean**	**MADMed**	**Sn**	**Qn**	**ScaleTau2**
MSFT (99)	16.75	23.74	23.87	23.58	23.74
AMAT (90)	17.03	1.29	1.78	1.65	1.65
ATVI (80)	12.20	6.20	7.39	5.64	6.17
NFLX (70)	10.66	7.49	10.77	11.73	12.16
PCLN (60)	22.58	36.09	35.96	35.34	35.44
MAR (50)	23.56	97.51	95.07	82.21	84.73
TTWO (40)	14.66	13.66	13.30	12.21	12.71
HOLX (30)	22.59	33.34	32.70	29.02	31.60
JBHT (20)	7.72	2.87	2.80	2.54	3.57
MELI (10)	12.50	8.40	8.41	8.27	9.73

4 Summary and Conclusion

This study brings forth a unifying empirical paradigm via the identification and treatment of non-microstructure noise (NMN) in order to induce robustification in empirical microstructure research with UHF data. It identifies the different stochastic processes at play in UHF data and presents methodologies to address the impacts thereof. It synthesizes dominant paradigms in microstructure theory and associated statistical techniques, and offers a robust and universal diagnostic framework for the empirical researcher to apply to the data elements commonly used in such research. Each step is detailed for application by the empirical researcher. The unifying paradigm's effectiveness is demonstrated through dramatic improvements in Realized Kernel (RK) and Realized Variance (RV) estimation efficiencies.

References

Yacine Ait-Sahalia and Jialin Yu. High frequency market microstructure noise estimates and liquidity measures. *The Annals of Applied Statistics*, 3(1):422–457, March 2009.

O. E. Barndorff-Nielsen, P. Reinhard Hansen, A. Lunde, and N. Shephard. Realized kernels in practice: trades and quotes. *Econometrics Journal*, 12(3):C1–C32, November 2009.

C.T. Brownlees and G.M. Gallo. Financial econometric analysis at ultra-high frequency: Data handling concerns. *Computational Statistics & Data Analysis*, 51(4):2232–2245, December 2006.

Martin D. Gould, Mason A. Porter, Stacy Williams, Mark McDonald, Daniel J. Fenn, and Sam D. Howison. Limit order books. *Quantitative Finance*, 13(11):1709–1742, 2013.

Frank R. Hampel. The Influence Curve and Its Role in Robust Estimation. *Journal of the American Statistical Association*, 69(346):383, June 1974.

Peter R Hansen and Asger Lunde. Realized Variance and Market Microstructure Noise. *Journal of Business & Economic Statistics*, 24(2):127–161, April 2006.

Jean Jacod, Yingying Li, and Xinghua Zheng. Statistical properties of microstructure noise. *Econometrica*, 85(4):1133–1174, 2017.

Ricardo A Maronna and Ruben H Zamar. Robust Estimates of Location and Dispersion for High-Dimensional Datasets. *Technometrics*, 44(4):307–317, November 2002.

A. I. McLeod and W. K. Li. Diagnostic Checking ARMA Time Series Models Using Squared-Residual Autocorrelations. *Journal of Time Series Analysis*, 4(4):269–273, July 1983.

Peter J. Rousseeuw and Christophe Croux. Alternatives to the median absolute deviation. *Journal of the American Statistical association*, 88(424):1273–1283, 1993.

Maxi San Miguel and Raul Toral. Stochastic Effects in Physical Systems. In Enrique Tirapegui, Servet Martinez, Enrique Tirapegui, Javier MartÃnez, and Rolando Tiemann, editors, *Instabilities and Nonequilibrium Structures VI*, volume 5, pages 35–127. Springer Netherlands, Dordrecht, 2000.

Michael Wilkinson. Perturbation Theory for a Stochastic Process with Ornstein-Uhlenbeck Noise. *Journal of Statistical Physics*, 139(2):345–353, April 2010.

Appendix

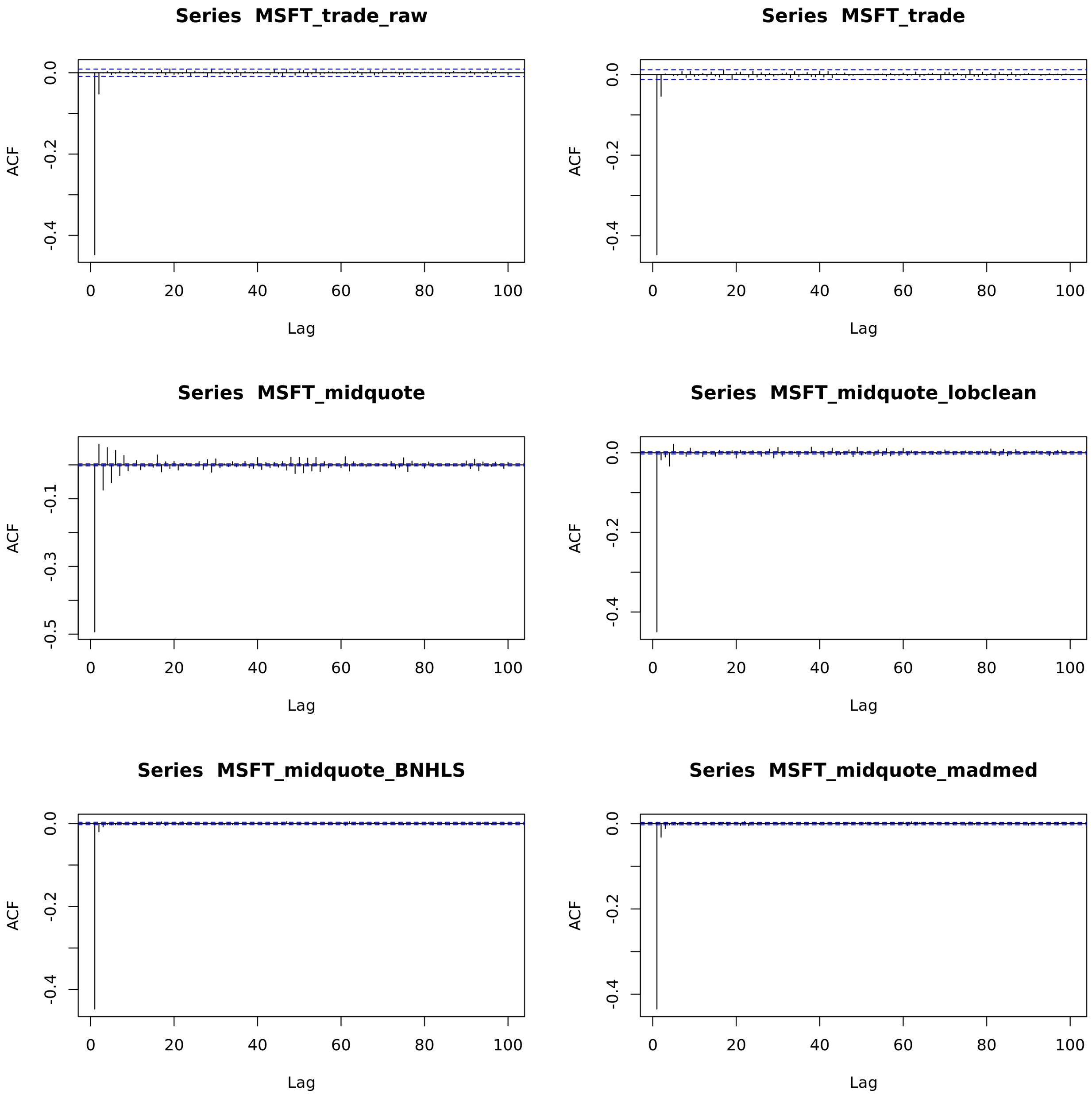

Figure 3: Autocorrelation in MSFT (NASDAQ, 30-October-2017). First order negative serial correlation present in trade price series. Time correction induces autocorrelation of higher order. In the midquote price series we find first order negative autocorrelation. Also seen at lower orders. LOBclean takes out some autocorrelation between 2nd to 10th lag. Both BNHLS and MADMed remove autocorrelation except the first and second order. Higher order autocorrelation thus seems to be coming from V.

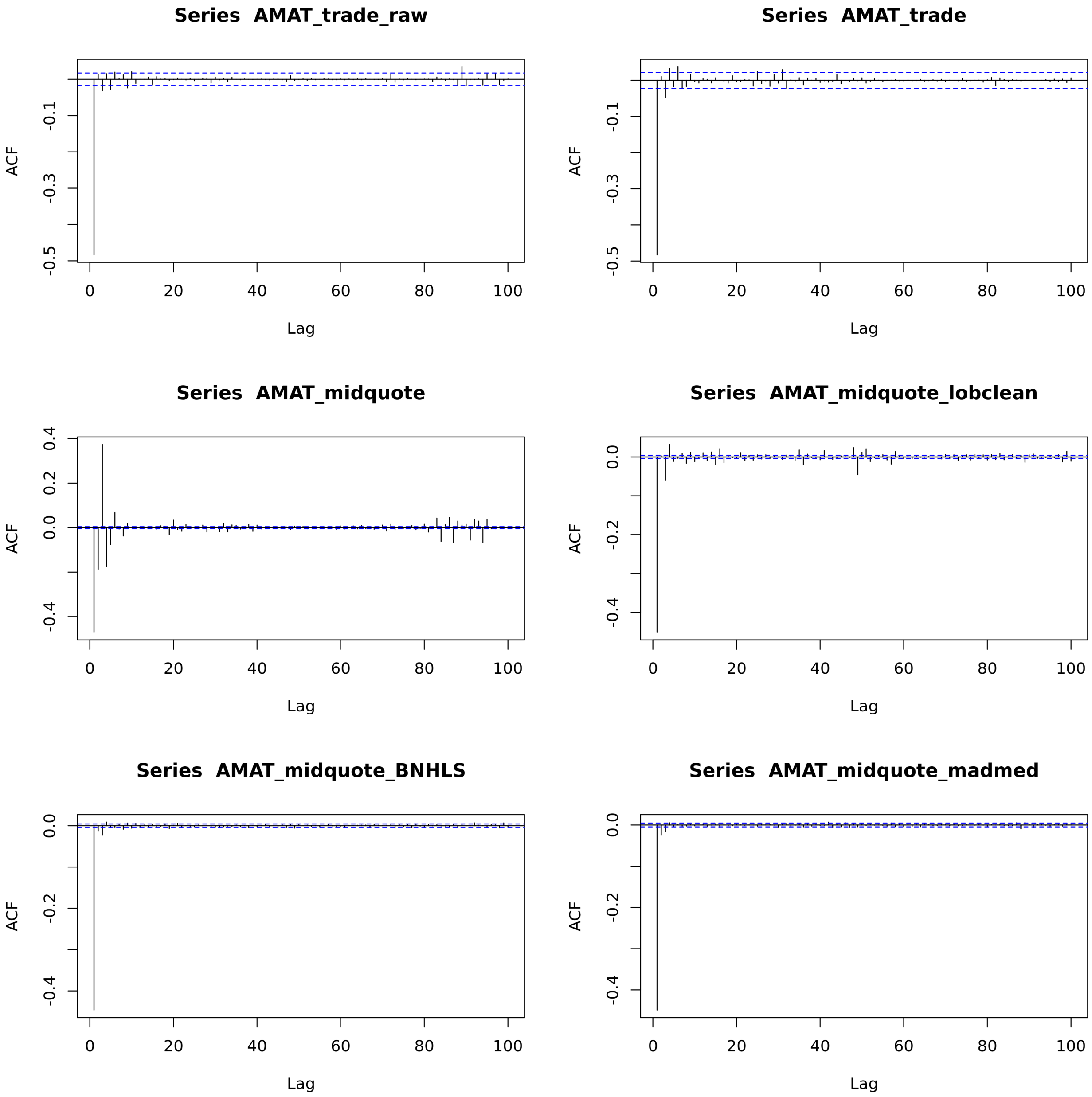

Figure 4: Autocorrelation in AMAT (NASDAQ, 30-October-2017). Both the raw trade and clean trade are first order negative serial dependence processes. Cleaning increases short order autocorrelation and removes long order autocorrelation. In midquotes we find first order negative and third order positive autocorrelation. No impact of LOB cleaning on first order acf but has impact on 3rd order positive autocorrelation. No difference between BNHLS and MADMed.

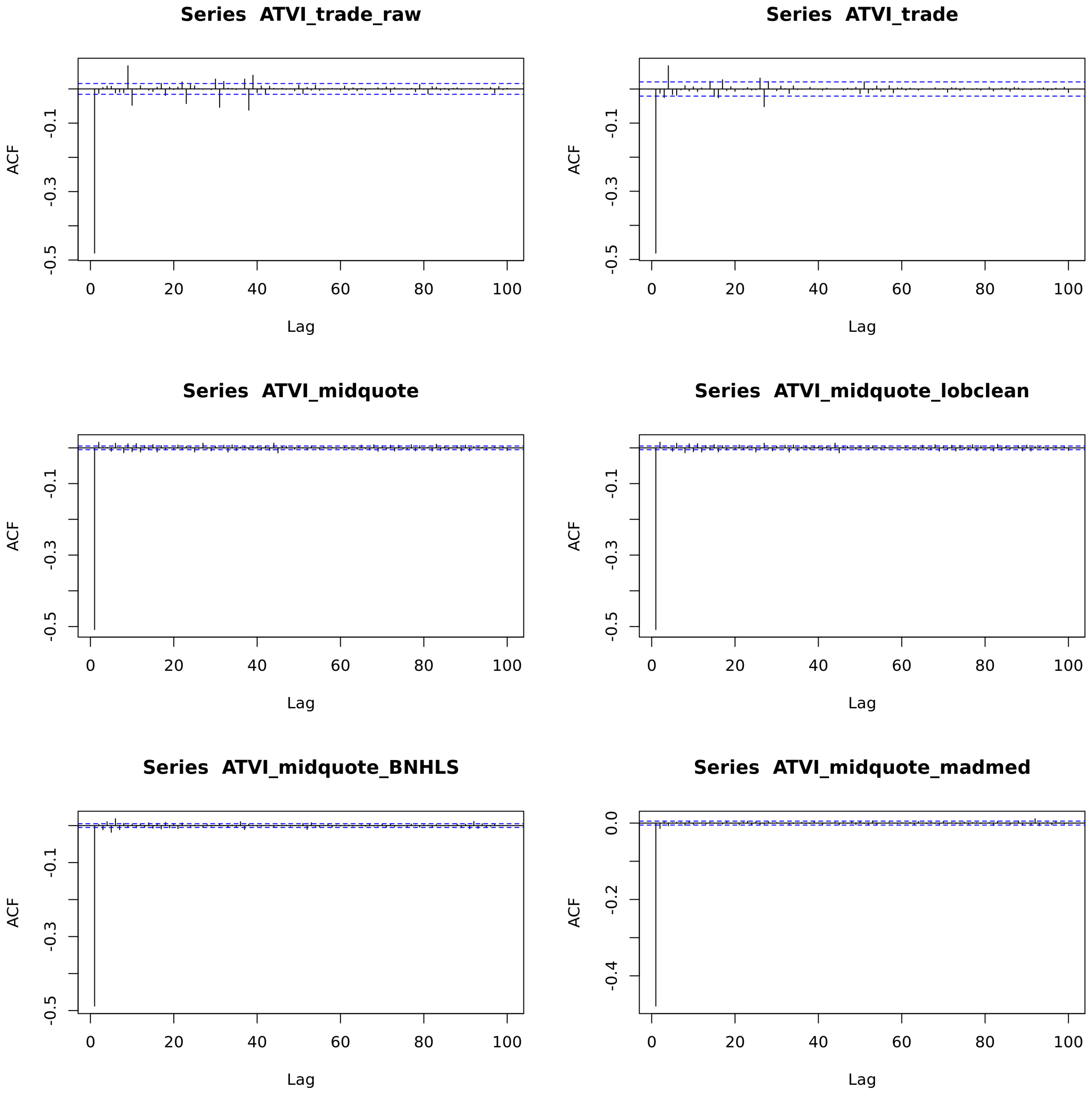

Figure 5: Autocorrelation in ATVI (NASDAQ, 30-October-2017). First order negative autocorrelation in trades. Time clean step reduces autocorrelation except first order dependence. First order negative autocorrelation and weak lower order autocorrelation in midquotes. LOB cleaning takes care of autocorrelation except first order. In BNHLS and MADMed it is completely removed except first order.

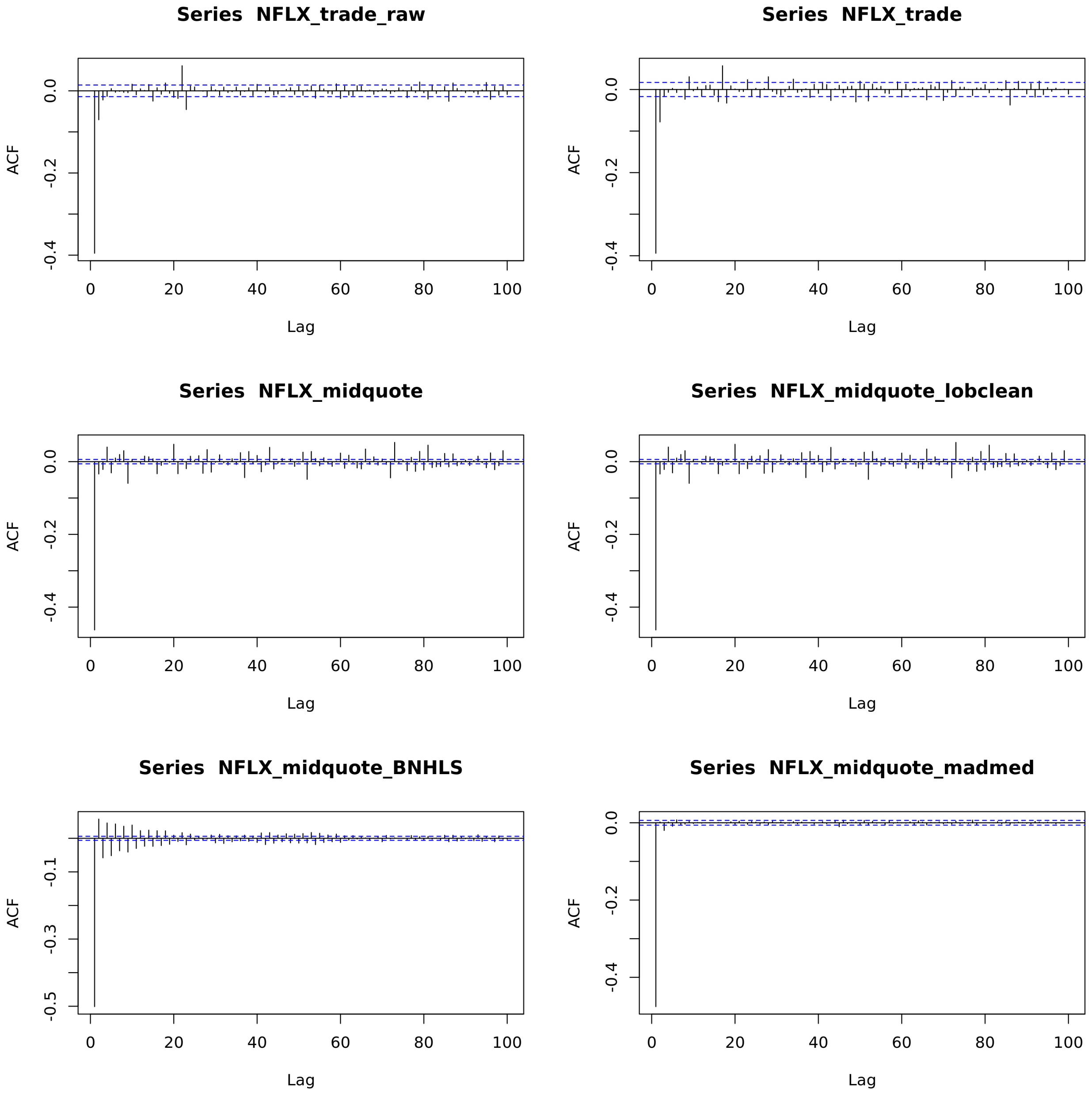

Figure 6: Autocorrelation in NFLX (NASDAQ, 30-October-2017). Trades are first order autocorrelation processes. Time cleaning step increases the autocorrelation but not significantly. In midquote series autocorrelation present upto lag 20. Strong first order negative autocorrelation. LOBclean cleans up several lower order autocorrelation but not first order. MADMed clean up all auto correlation except lag 1. BNHLS misses out the first 20 lags.

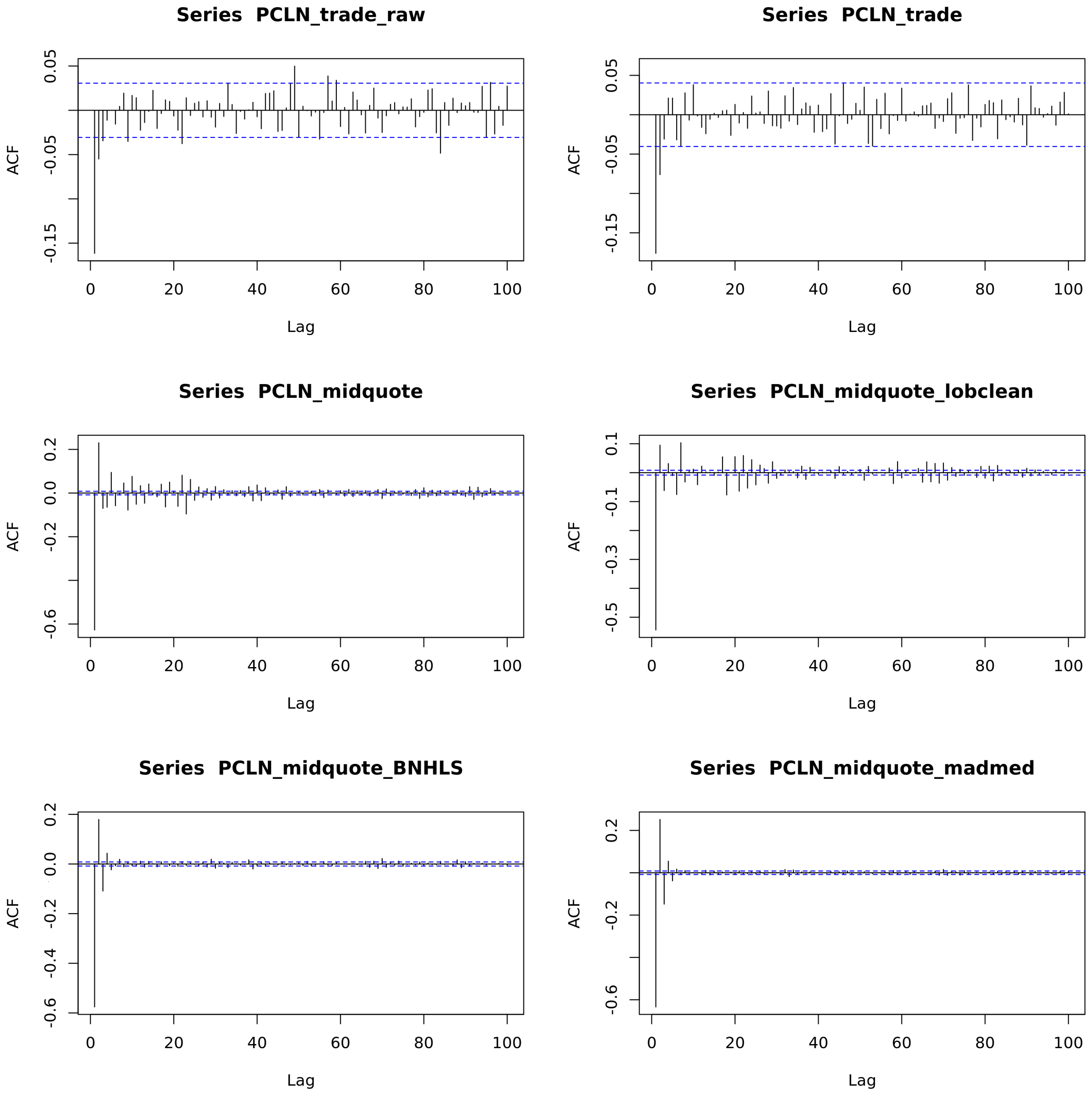

Figure 7: Autocorrelation in PCLN (NASDAQ, 30-October-2017). Strong first order autocorrelation in trades. Timeclean step reduces autocorrelation in trades. In the midquote series, strong first and second order autocorrelation is seen. LOBclean impacts the second order autocorrelation. Not others. MADMed and BNHLS clean up autocorrelation other than first and second order.

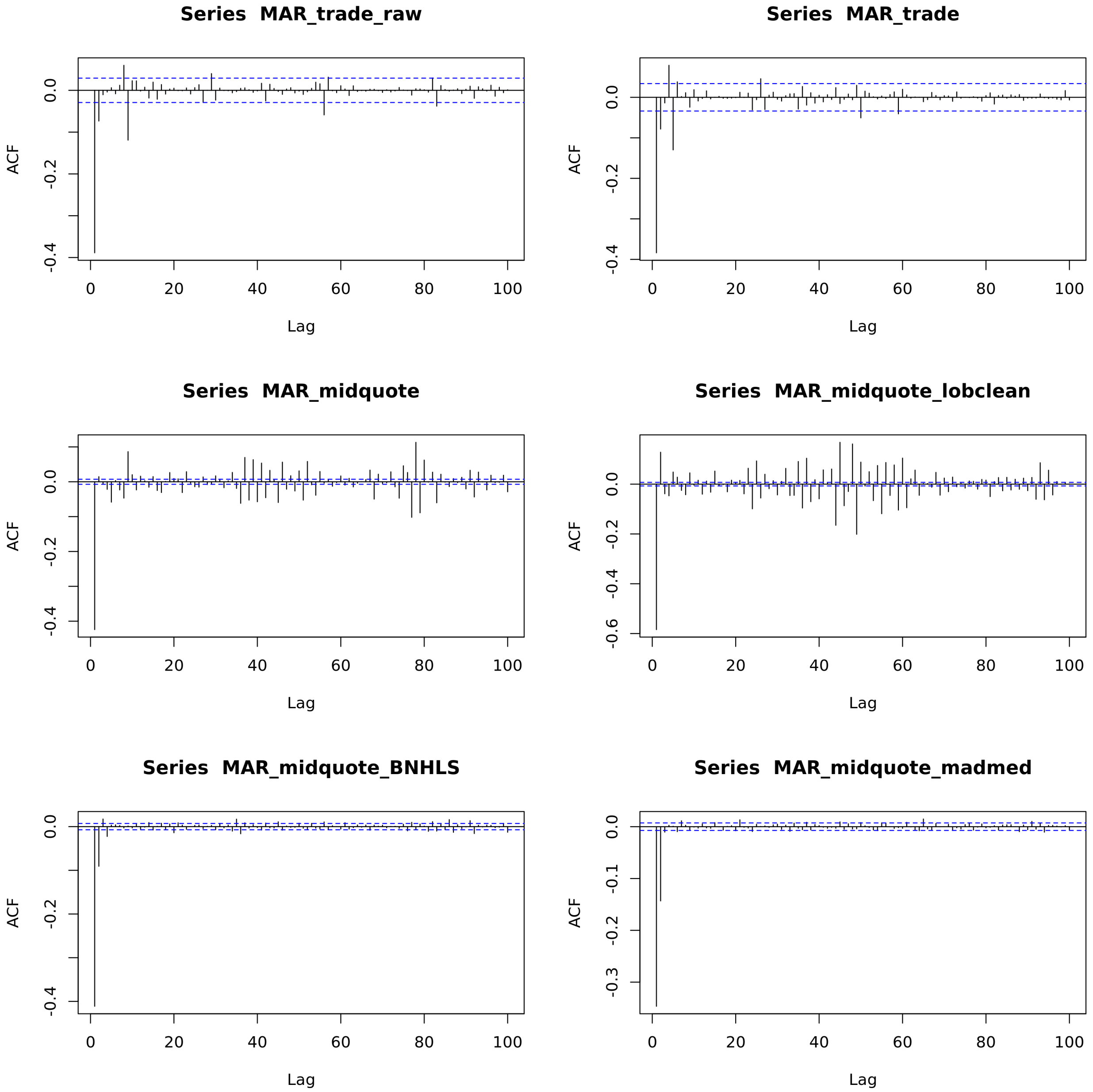

Figure 8: Autocorrelation in MAR (NASDAQ, 30-October-2017). Strong first order autocorrelation and other weaker autocorrelations in raw and time clean trades. In midquote series, first order negative autocorrelation. However LOBclean increases autocorrelation of higher orders. BNHLS and MADMed cleans up higher order autocorrelation.

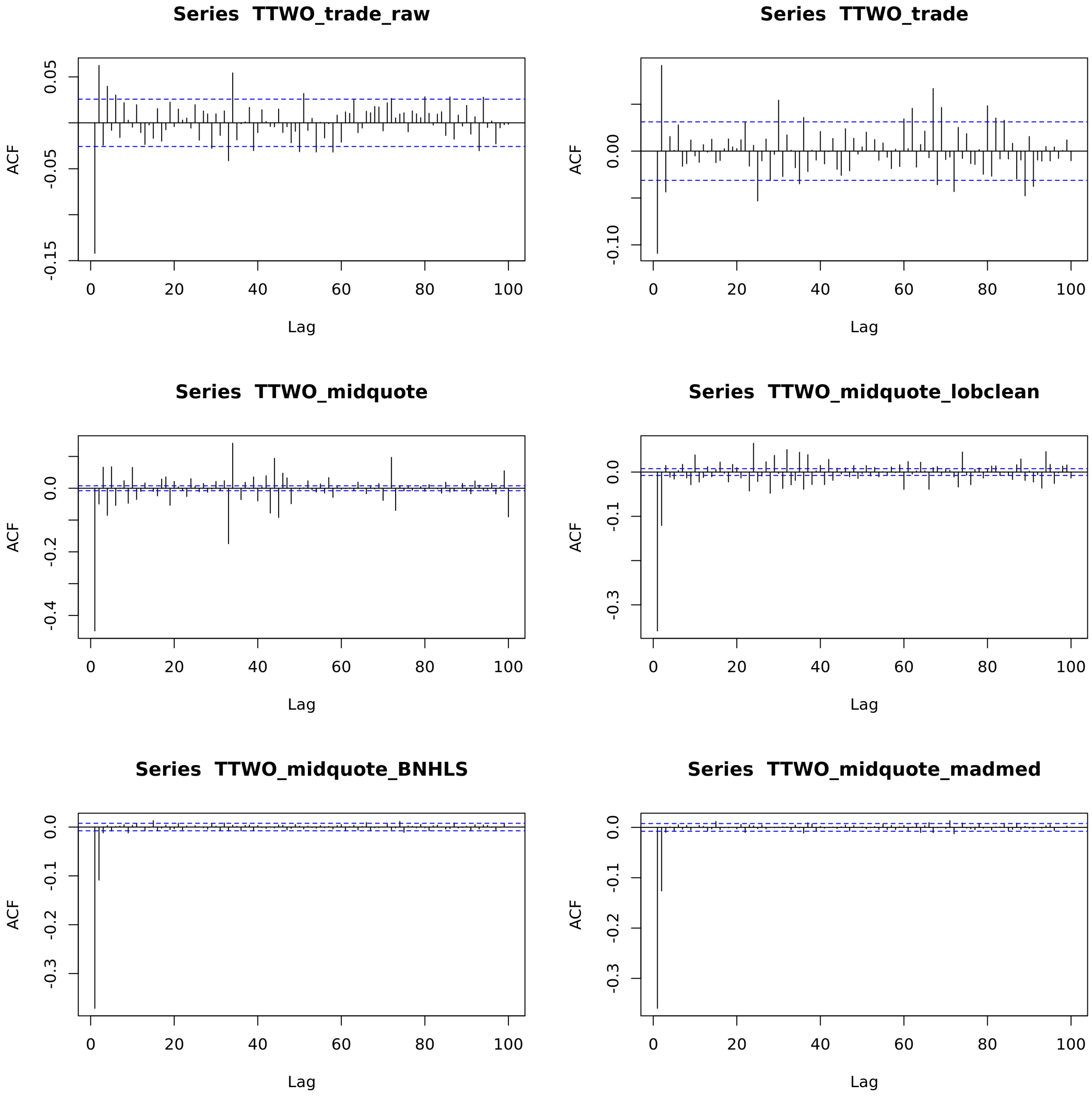

Figure 9: Autocorrelation in TTWO (NASDAQ, 30-October-2017). Trades show first order negative autocorrelation and weak second order positive autocorrelation. Higher orders autocorrelation also present. In midquote series we find first order negative autocorrelation. BNHLS and MADMed remove all higher order autocorrelation.

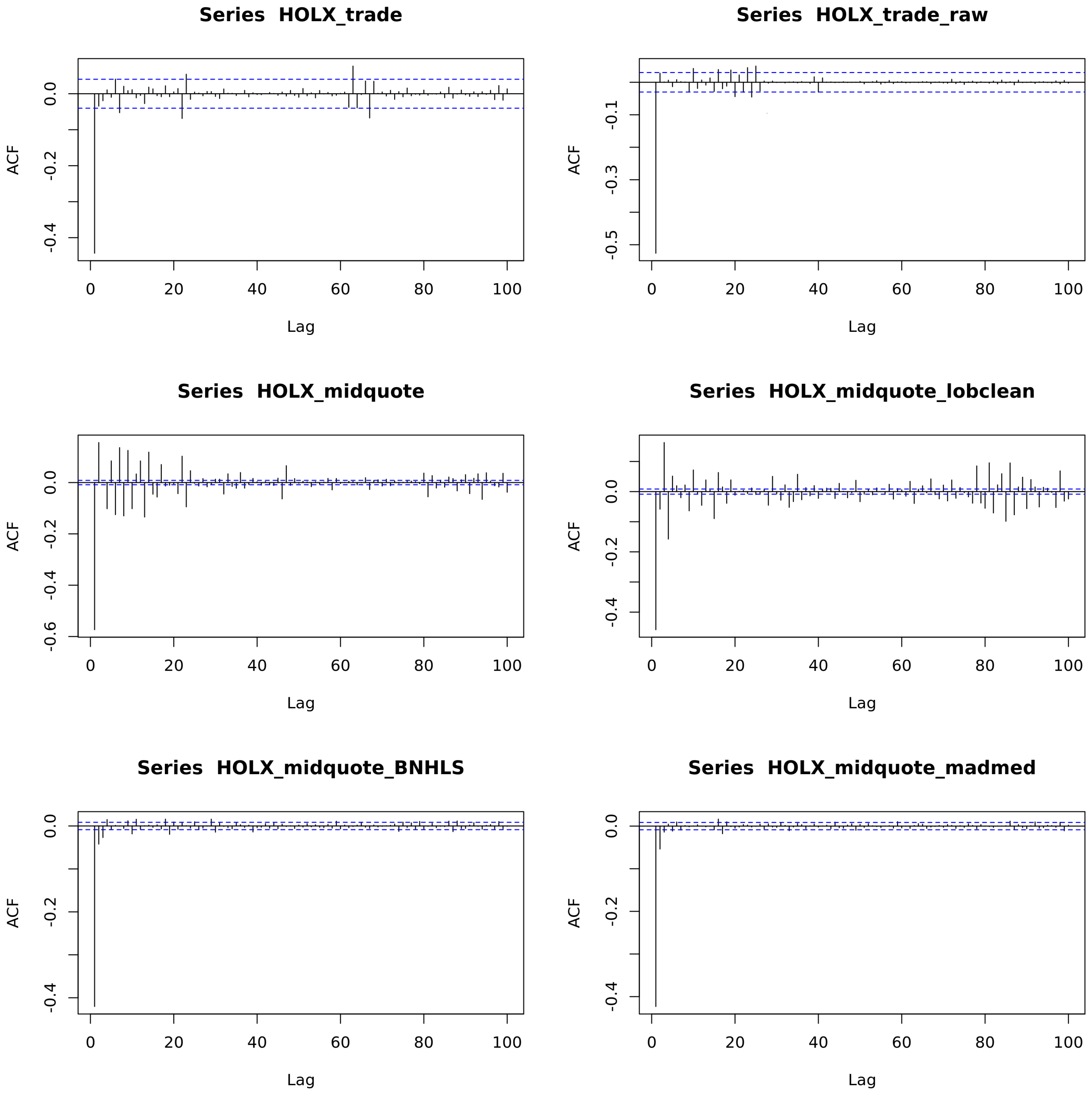

Figure 10: Autocorrelation in HOLX (NASDAQ, 30-October-2017). The trades series displays first order negative autocorrelation. No impact of time cleaning step. In the midquote series we find first order negative autocorrelation. Autocorrelation is also present in other orders, but not significant. LOBclean step has a mixed impact. BNHLS and MADMed remove higher order autocorrelation.

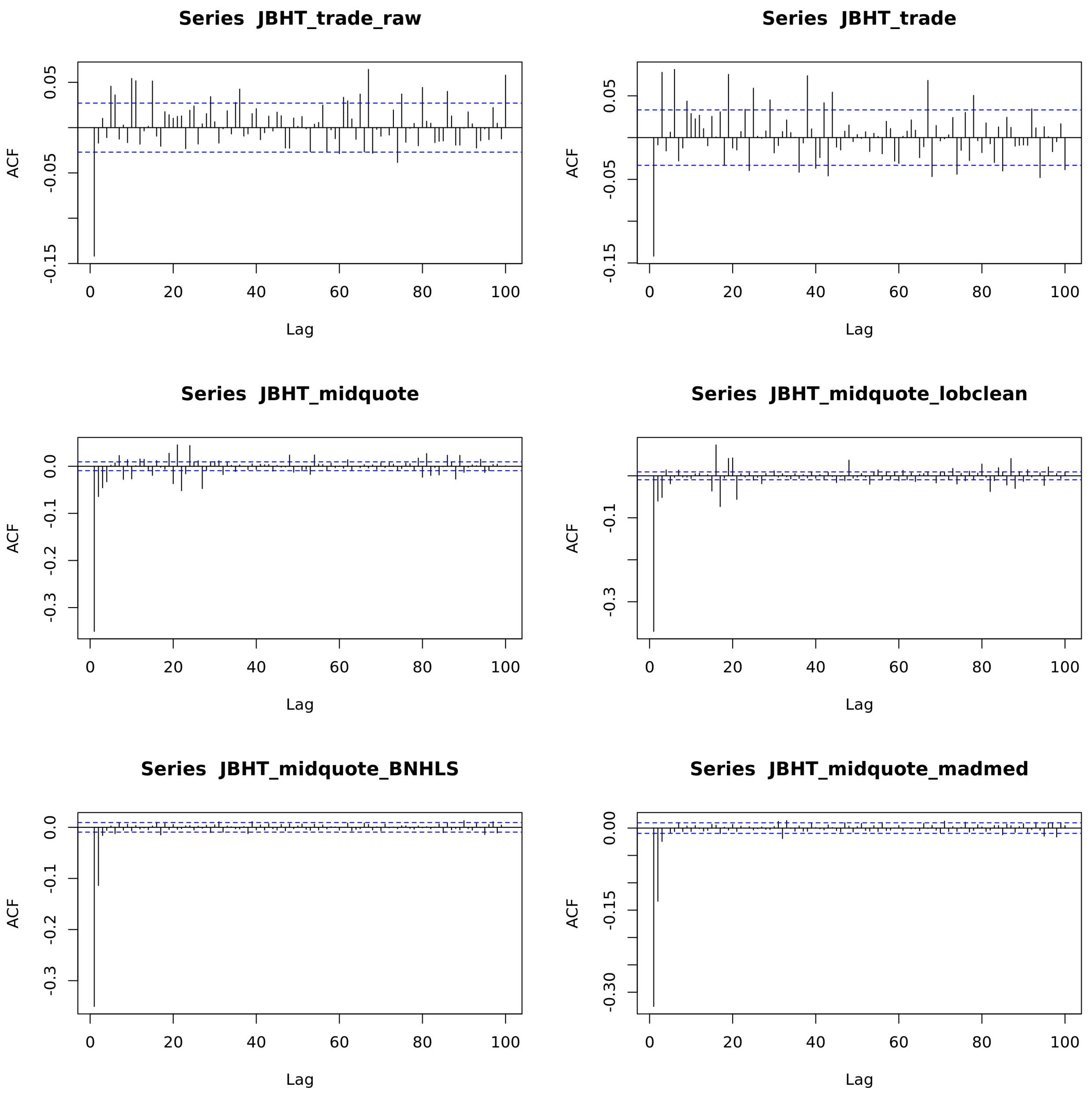

Figure 11: Autocorrelation in JBHT (NASDAQ, 30-October-2017). First order negative autocorrelation is seen in trades series but is not very strong. Higher orders present and LOBclean increases it. In midquote series first order negative serial correlation present. BNHLS and MADMed increase the second order and wipe out others.

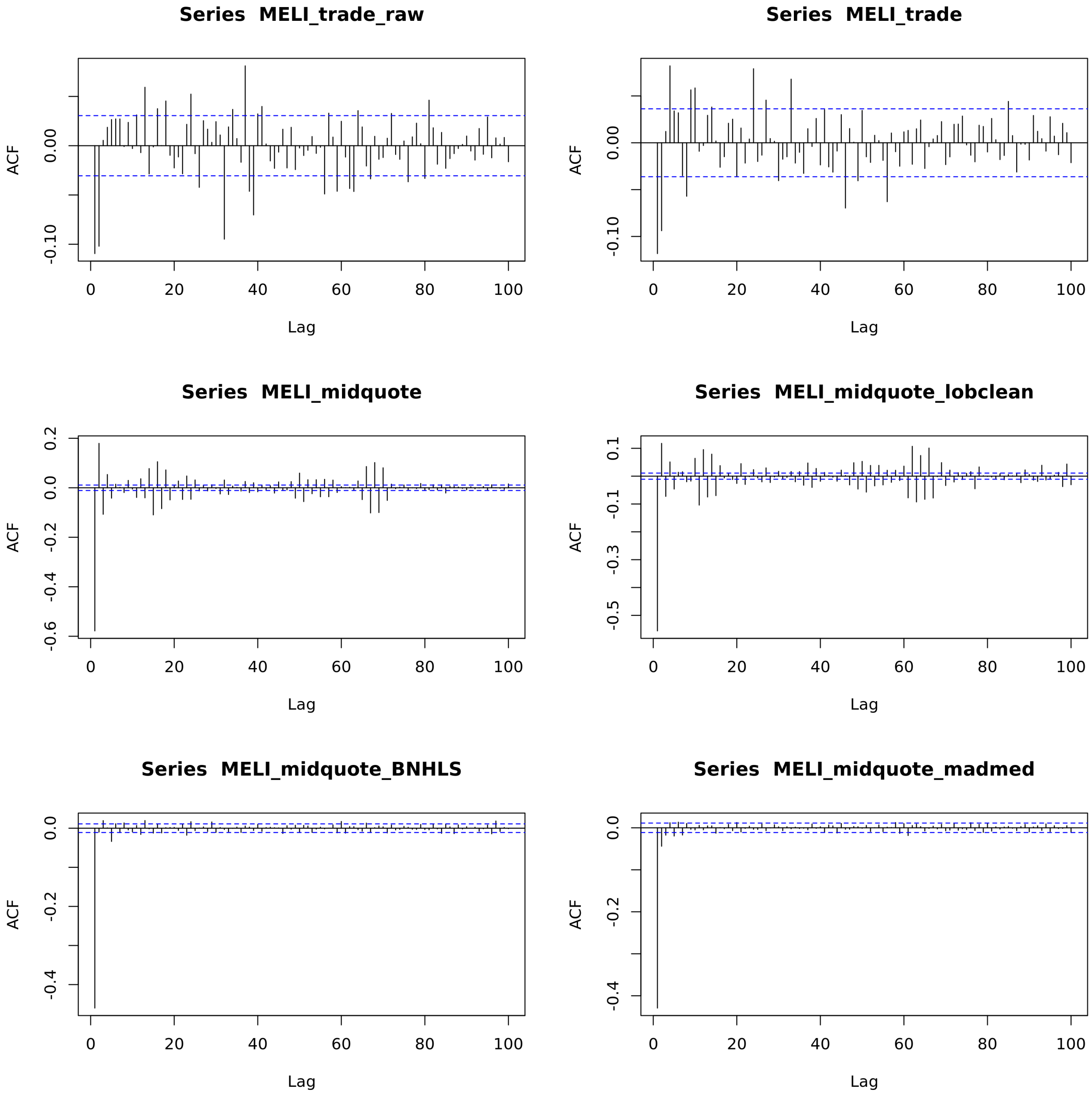

Figure 12: Autocorrelation in MELI (NASDAQ, 30-October-2017). In trades, autocorrelation of lower orders and middle orders is seen. The timeclean step has mixed impact on autocorrelation of differing lags- increases some while it decreases others. In midquote series there is strong first order negative autocorrelation. No impact of LOBclean step. BNHLS and MADMed decrease the first order autocorrelation by 25% and wipe clean autocorrelation in higher orders.

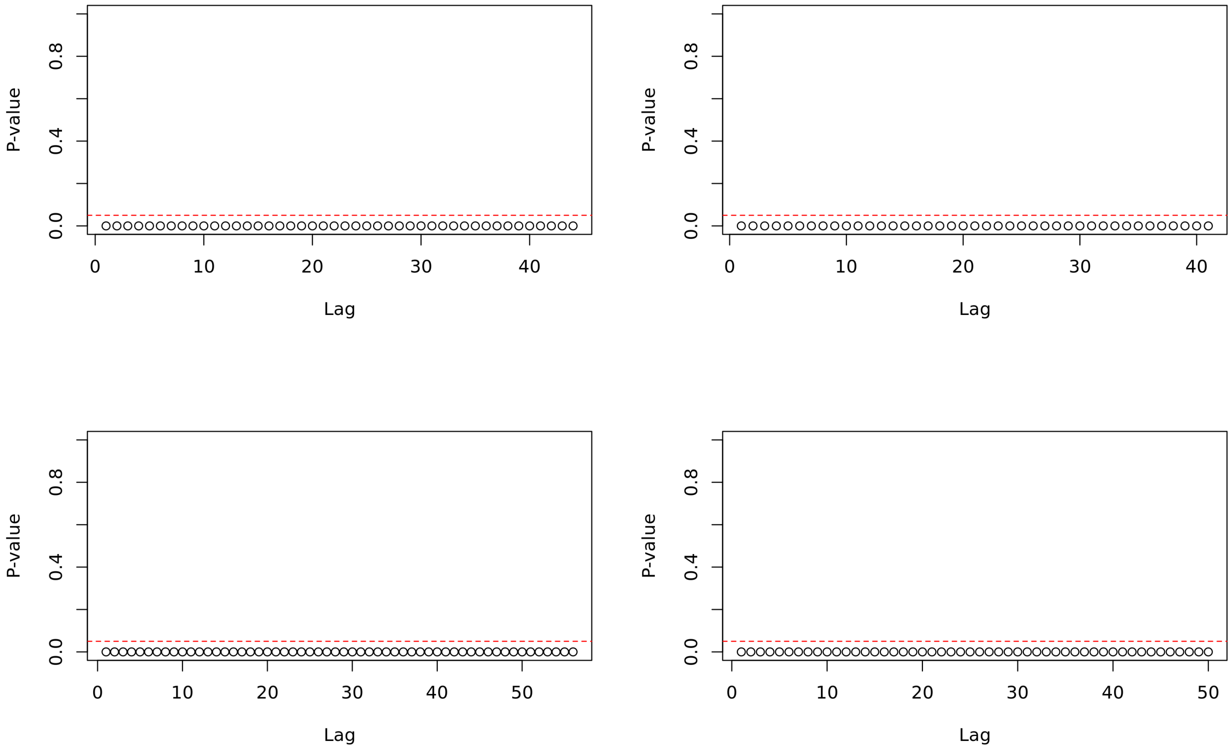

Figure 13: The left panel shows the results of McLeod-Li test for MSFT and the right panel for NFLX. At the top is the test for trades and bottom the test for midquotes. The null hypotheses, of the presence of no ARCH effects is rejected as the p values are close to zero for all the four cases..