

TIME SERIES ANALYSIS OF LIMIT ORDER BOOK DATA

CHRISTOS PAPADAKIS

1. ABSTRACT

In recent years, trading in stocks and other financial instruments increasingly takes place through electronic trading platforms. The majority of electronic exchanges use Limit Order Books (LOBs) to record the interest of potential buyers and sellers. Traders may submit a limit order to buy or sell a certain quantity at a certain limit price, or a market order to buy or sell a certain quantity at the best currently available limit price. In the second scenario the order is executed immediately, while in the first scenario a limit order may be modified or even canceled before execution. In this paper we deal exclusively with limit orders while we try to explore the potential of this high frequency data to derive useful information, but also to make predictions using time series theory.

2. INTRODUCTION

The dataset used for the purpose of this paper contains the history of the LOB of the Turkish GARAN stock for the 253 trading days between January 2 and December 29, 2017. From now on we will refer to a buy order as *bid* and to a sell order as *ask*. Each day reports the top 5 levels of bids and asks price, along with their respective volume, or size as well as the date and time, measured in milliseconds, at which the limit orders are submitted.

Date-time	bid1	bsize1	...	bid5	bsize5	ask1	asize1	...	ask5	asize5
2017-01-02 09:55:08.179	7.61	110328	...	7.57	150126	7.61	378930	...	7.65	443326
2017-01-02 09:55:08.323	7.60	242605	...	7.56	54095	7.61	268592	...	7.65	443326
2017-01-02 10:00:00.123	7.60	242105	...	7.56	54095	7.61	266342	...	7.65	443326
2017-01-02 10:00:00.227	7.60	132105	...	7.56	54095	7.61	266342	...	7.65	443326
2017-01-02 10:00:00.334	7.60	132105	...	7.56	64095	7.61	266342	...	7.65	443326

TABLE 1. First 5 rows of original data

By construction of the limit order book, the prices are arranged in such a way that the order for the bid prices is descending and ascending for the ask counterparts,

Received by the editors 19 Feb. 2021.
Prof. Dr. Joseph Teichmann.
Dr. Wahid Khosrawi.
Dr. Erdinc Akyildirim.

as we confirm in *Table 1*. That is, the first level of bids and asks shows the best bid and ask quotes available at time $t \in T$, for some specified collection of times T . It is also worth to mention that the time difference between the entries is not constant. That is because, as previously described, an order may be submitted in the LOB at any time, at a very high frequency, for all trading days.

Now, say we are interested in the variable/column $X = 'bid1'$ for a specified range of times $t \in T$. We choose a range of times, or in other words a time window T and gather a sequence of observational time - value pairs (t, X_t) with strictly increasing observation times. Then the collection of $((X_t))_{t \in T}$, forms a time series, whose components are not equally spaced, because the difference of any two consecutive observation times corresponding to its component, X_t and X_{t+1} , is not constant, as we commented above.

Time series whose components are unevenly spaced (like $X = ((X_t))_{t \in T}$ above) are called *irregular*, or *unevenly spaced time series*. Irregular time series often appear in many other scientific domains, such as natural disasters, astronomy and longitudinal studies, where events typically occur at irregular time intervals. An interesting and worth mentioning fact is that due to computational limitations, the foundations of time series theory are based on equally spaced data and thus our data lack many of the nice features that regular time series enjoy. As result, until now not many methods have been developed specifically for unevenly spaced data [Eck12]. That is why most of the available financial data are at daily or lower frequency [Gen+01].

Probably the most popular approach to analyzing unevenly spaced time series is to transform it into equally spaced by using a form of interpolation (usually linear) and then use the common tools from the theory. However, interpolation of data means in other words, generating new artificial data and that would introduce a significant bias that is hard to quantify. The reason is that linear interpolation fails to capture the stochasticity around the conditional mean, while the latter takes into account the available information up to and included the time of our interest. Andreas Eckner (2014) [Eck12] provides several examples that expose many significant drawbacks when using interpolation and Dacorogna and others [Gen+01] provide a rich and more sophisticated methodology on how to deal with high-frequent big data. However, in this paper we will not use any of these methods or any kind of interpolation.

3. DATA PREPARATION

As described previously, the data consists of recordings between around 10:00 - 18:00, measured in milliseconds, for every each and one of 252 trading days. That is more than 3.5 millions data points and we hope that it is now clear why we need a frequency reduction technique. But before reducing the frequency, let us first have a look at the distribution of our data.

From *Table 2* and *Fig.1* below, we observe that the distribution of the data is unbalanced, in the sense that, we have a number of 41.946 entries on 12-01-2017, while only 3.335 on 31-08-2017, which we considered to remove from the dataset eventually. In fact, we conclude that there is right skewness in the daily sample size distribution of our data, as we observe in the histogram of *Fig.1* below.

count	mean	std	min	25%	50%	75%	max
253	14.485,668	5.731	3.335	10.308	13.431	17.290	41.946

TABLE 2

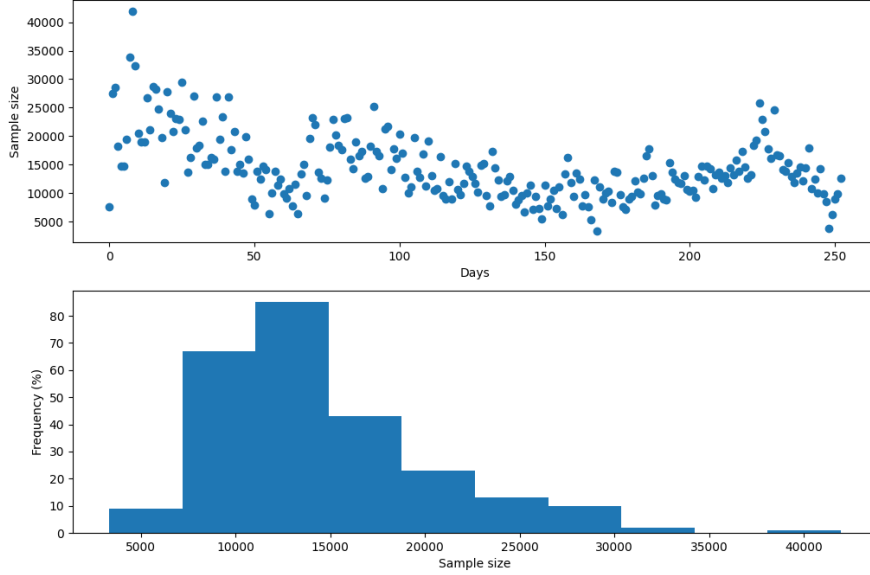


FIG. 1. Data distribution

However, we did not turn our data into balanced data as we thought we would lose an important amount of potentially useful information.

With that being said, we continue by lowering the frequency of the data. In particular, our goal is to represent the data as minute-wise recordings between 10:10 - 12:50 AM and 14:10 - 17:50 PM. Without using interpolation, or any other fancier method, we lowered the frequency to minute-wise measurements while keeping original instances and not artificial ones. Our approach is to assign the values at the latest available data inside a time period of one minute and for the minutes that no data were recorded, we used the last data available like illustrated below :

<i>Available Times</i>		<i>Assign value at</i>	<i>for:</i>
09:55:00.108		09:55:00.248	09:55
09:55:00.248		09:55:00.248	09:56
10:00:00.128		09:55:00.248	09:57
10:00:00.230	→	09:55:00.248	09:58
...		09:55:00.248	09:59
10:00:59.752		10:00:59.752	10:00
10:01:00.554	
...			

Date-time	bid1	bsize1	...	bid5	bsize5	ask1	asize1	...	ask5	asize5
2017-01-02 10:10:00	7.60	600846	...	7.56	196010	7.61	334394.0	...	7.65	733891
2017-01-02 10:11:00	7.58	340088	...	7.54	255062	7.59	123218	...	7.63	307934
2017-01-02 10:12:00	7.59	228477	...	7.55	260544	7.60	653313	...	7.64	545518
2017-01-02 10:13:00	7.59	270245	...	7.55	260544	7.60	457197	...	7.64	545518
2017-01-02 10:14:00	7.59	520966	...	7.55	260454	7.6	515724	...	7.64	545518

TABLE 3. First 5 rows of processed data

4. ANALYSIS

Now we have finished with the preparation of the data, we proceed in the next section with some interesting findings that we came across during our analysis. We start by calculating the density of the first five levels of bid and ask prices. In more detail, we view bid and ask as realizations of two discrete random variables that take 5 possible values, which correspond to the 5 levels, weighted by,

$$(4.1) \quad w_i = \frac{s_i}{\sum_{i=1}^5 s_i}$$

where s_i is the bid or ask size for level i .

Having obtained the density functions, allows us to calculate the first four moments of bid and ask by the formula,

$$(4.2) \quad \mu_t^s = \sum_{i=1}^5 w_i x_i^s$$

where x_i the bid or ask price for level i and $s = 1, 2, 3, 4$.

Figure 2 below displays the correlation between all 4 moments for bid and ask. Interestingly, there is a very strong correlation as we can also see numerically in Table 4.

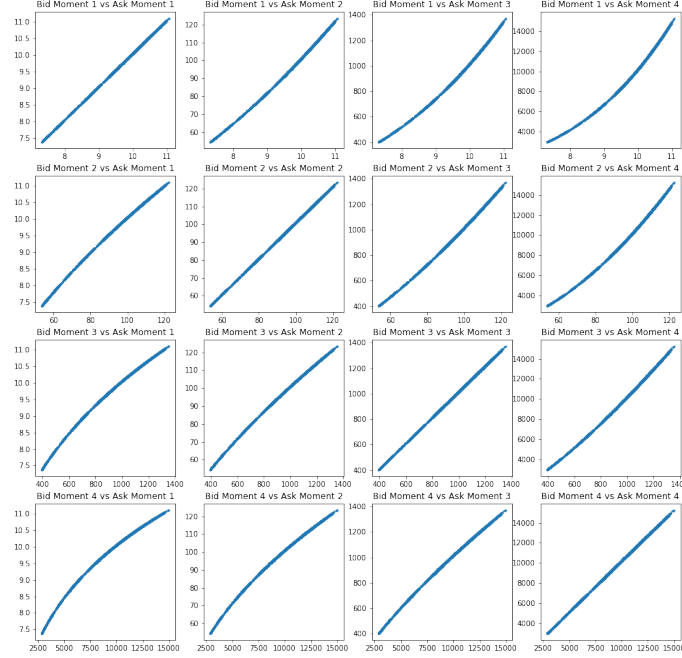


FIG. 2. Moments

	Ask Moment 1	Ask Moment 2	Ask Moment 3	Ask Moment 4
Bid Moment 1	0.999981	0.998488	0.994205	0.987490
Bid Moment 2	0.998460	0.999979	0.998599	0.994645
Bid Moment 3	0.994120	0.998541	0.999977	0.998719
Bid Moment 4	0.987326	0.994505	0.998636	0.999975

TABLE 4. Moments

However, this shall not seem so surprising, since we notice in *Table 3* above that the values of bid and ask prices are pretty close to each other.

From now on, we focus our interest on the midpoint price as it is commonly used for the evaluation of the current price of a stock. In order to calculate the midpoint price, we first find the best (maximum) bid and best (minimum) ask among the 5 different levels, and then use the formula,

$$(4.3) \quad MP_t = \frac{P_{bid} + P_{ask}}{2}$$

for every minute $t \in T$. The table below (*Table 5*) summarizes some basic descriptive statistics for each month.

	Mean	Std	Max	Min
January	7.739723	0.228324	8.355	7.345
February	8.756813	0.174639	9.115	8.245
March	8.909965	0.077821	9.095	8.645
April	9.431348	0.295864	9.805	8.835
May	9.536650	0.127705	9.935	9.265
June	9.678561	0.107282	9.955	9.475
July	10.292391	0.258248	10.685	9.805
August	10.684081	0.146347	10.995	10.415
September	10.335317	0.278066	10.810	9.605
October	10.027817	0.223441	10.475	9.415
November	10.103401	0.496519	11.095	9.170
December	10.070113	0.264077	10.785	9.385

TABLE 5. Midpoint price statistics

Notice that the midpoint price follows a positive trend as time goes by, while it is worth mentioning that on November there is the largest variance, which we can also interpret as volatility. By illustrating the evolution in time of midpoint price in *Figure 3* below, we observe that there are significant differences between the months.

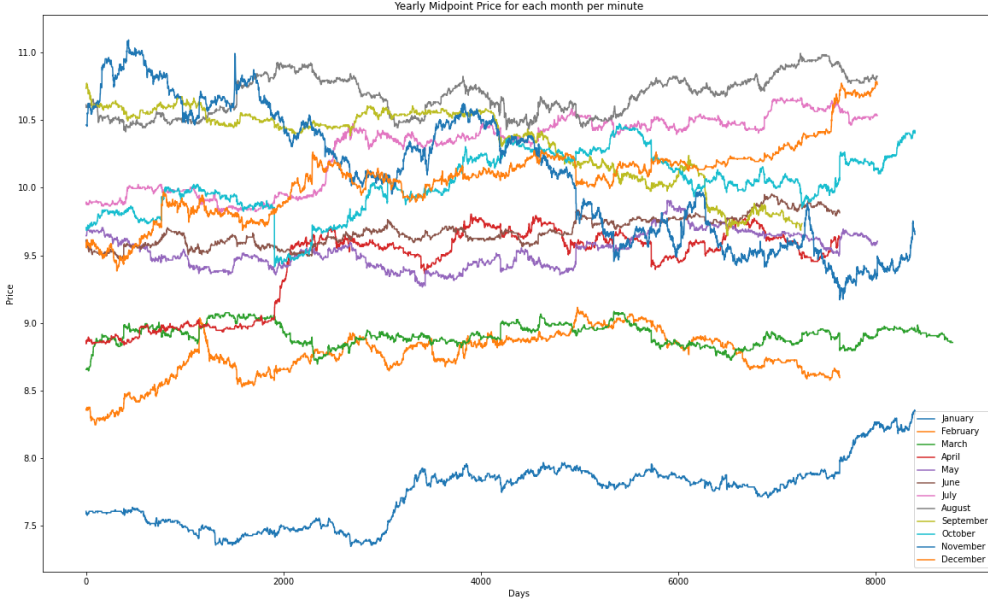


FIG. 3. Midpoint price for every month

5. TIME SERIES ANALYSIS

In this session we perform time series analysis on the midpoint price which will allow us to find a suitable model that will be used for forecasting. In more detail, we create a time series whose first component represents the midpoint price of GARA stock at 10:10 AM 2017-01-02 and the last one at 17:50 PM 2017-12-29 with the intermediate components representing the midpoint price by one minute difference from the next one. One can see below how the midpoint price evolved during the year 2017.



FIG. 4. Midpoint price in 2017

5.1. Model Selection. As we can observe, there is an increasing trend with a very weak seasonality, while we conclude that the time series does not seem to be stationary. Empirical time series like the above, exhibit homogeneity in the sense that there are local or trending patterns that reappear along their trajectories. Box, Jenkins, and Reinsel [Box+15] provide the class of *Auto Regressive Integrated Moving Average* processes, which are suitable for describing such homogeneous non-stationary behavior and are denoted by $ARIMA(p,d,q)$.

The meaning of the term 'Auto Regressive' is that it is about a linear regression model, which uses its own lags as predictors. Then $AR(p)$ denotes an Auto Regressive model which uses p of its own lags plus an error term. We can formulate this via,

$$(5.1) \quad Y_t = \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \dots + \beta_p Y_{t-p} + \epsilon_t$$

where β_0 is the intercept, β_j are the coefficients of the p lags Y_{t-j} , for $j = 1, 2, \dots, p$ and ϵ_t are Gaussian white noise, namely independent normally distributed random variables with mean zero and variance σ_ϵ^2 .

Likewise, the terminology $MA(q)$ denotes a Moving Average model where Y_t depends only on its own q lagged forecast errors. Again, following a similar terminology, the formula for a $MA(q)$ model is given by,

$$(5.2) \quad Y_t = \gamma_0 + \gamma_1 \epsilon_{t-1} + \gamma_2 \epsilon_{t-2} + \dots + \gamma_q \epsilon_{t-q} + \epsilon_t$$

where γ_0 is the intercept, γ_j are the coefficients of the q lagged errors ϵ_{t-j} , for $j = 1, 2, \dots, q$ and the noise. ϵ_t is assumed to be Gaussian white noise.

What is left, is to investigate what 'Integrated' means in the ARIMA model. In several practical applications, like the one presented in this paper, one often has to deal with data for which stationary processes do not seem plausible for modeling. Long-term trends for instance, like the one that the time series of the midpoint price exposes in *Figure 4*, is a strong indication of non-stationarity. Well, the case is that we need to suppose some suitable order of *difference*, in order to make sure that the time series becomes stationary. *Differencing* a time series means subtracting the previous observations from the current observation. This will eliminate trend and seasonality but it will also stabilize the mean of the time series. The *order* of difference, which we denote by d , is the number of times that differencing is applied on the time series.

Summing up the above and having chosen a suitable order of difference d to make the time series stationary, the $ARIMA(p, d, q)$ model can be formulated as,

$$(5.3) \quad Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \dots + \beta_p Y_{t-p} + \epsilon_t + \gamma_1 \epsilon_{t-1} + \gamma_2 \epsilon_{t-2} + \dots + \gamma_q \epsilon_{t-q}$$

with $\beta_p \neq 0$, $\gamma_q \neq 0$ and $\sigma_\epsilon^2 > 0$.

Note that (5.3) expresses a linear regression problem. That is, we can introduce the vector notation,

$$(5.4) \quad \theta^T = (\beta_1, \dots, \beta_p, \gamma_1, \dots, \gamma_q)$$

$$(5.5) \quad Y_t^T = (Y_t, Y_{t-1}, \dots, Y_1)$$

namely, Y_t contains all information up to time t . From the foundations of modern statistics, we know that if we want to estimate the model parameters, the most classical approach is to use the *Least Squares Method*, which uses the *cost function*,

$$(5.6) \quad \sum_{t=0}^T (Y_t - \hat{Y}_t)^2$$

Since (5.6) is an example of a *cost* function, our best benefit is to find the right model which gives predictions, or *estimators* \hat{Y}_t not far away from the underlying variable of our interest Y_t . Equivalently, we are trying to find the right model in the sense that our estimators *minimize* the least squares cost function. However, it is also known that in a linear model, if the error terms ϵ_t are normally distributed, then the solution obtained via the least squares is identical to the solution via the *Maximum Likelihood Estimation* method.

The Maximum Likelihood method computes the likelihood function and then finds the estimator $\hat{\theta}$ which maximizes the likelihood, i.e. the probability that the sample drawn comes from the distribution we assumed. The ML estimator can be obtained analytically, or more commonly in real world data, numerically using numerical analysis methods (e.g. Newton- Raphson method).

We now use this notation to calculate the joint probability distribution, or in other words the Likelihood Function, for all observations up to some time index T given the values of θ and σ_ϵ^2 . Note that for iid data Y_t with some marginal probability density function $f(Y_t; \theta)$, the joint density function for a sample $y_t = (y_t, y_{t-1}, \dots, y_1)$ is simply given by the product of the marginal density functions for each observations. Then the successive application of the probability law $P(AB) = P(A)P(B)$ for independent events A and B , yields the equation

$$(5.7) \quad f(Y; \theta) = f(y_1, \dots, y_T; \theta) = \prod_{t=1}^T f(y_t; \theta)$$

The likelihood function then, is that joint density expressed as a function of the population parameter θ given our sample observations. However, for a sample obtained from a stationary time series the above construction of the likelihood function does not work, as argued with [WLL19]. One approach for calculating the likelihood for a stationary time series is given by [Ham20], which requires, among other things, the derivation of the covariance matrix of Y . Instead, we follow the approach as of [Mad07], which relies on the factorization of the joint density into a series of conditional densities and the density of a set of initial values. Also note that in our case, the model parameters we are trying to estimate are σ_ϵ^2 and θ , as described in (5.4). Thus, by using the law of conditional probabilities $P(A) = P(A|B)P(B)$, this approach yields,

$$\begin{aligned}
L(Y_T; \theta, \sigma_\epsilon^2) &= f(Y_T; \theta, \sigma_\epsilon^2) \\
&= f(Y_T | Y_{T-1}; \theta, \sigma_\epsilon^2) f(Y_{T-1}; \theta, \sigma_\epsilon^2) \\
(5.8) \quad &= \left(\prod_{t=p+1}^T f(Y_t | Y_{t-1}; \theta, \sigma_\epsilon^2) \right) f(Y_p; \theta, \sigma_\epsilon^2)
\end{aligned}$$

This is the general formula for the *likelihood function* for time series data [Mad07] and it only remains to maximize it with respect to θ and σ_ϵ^2 , the variance of the error terms. Now, the error terms ϵ_t defined as the difference between the true and predicted values by the model, denoted by \hat{Y} and are given by,

$$(5.9) \quad \epsilon_t = Y_t - \hat{Y}_{t|t-1}$$

Hence by taking expectations on both sides of (5.9) given the available information up to time $t-1$ and by using the assumption that ϵ_t are supposed to follow a $N(0, \sigma_\epsilon^2)$ distribution, we obtain that,

$$(5.10) \quad \hat{Y}_{t|t-1}(\theta) = E[Y_t | Y_{t-1}, \theta]$$

This gives that,

$$\begin{aligned}
f(Y_t | Y_{t-1}; \theta, \sigma_\epsilon^2) &= \frac{1}{\sigma_\epsilon \sqrt{2\pi}} \exp \left(-\frac{(Y_t - E[Y_t | Y_{t-1}, \theta])^2}{2\sigma_\epsilon^2} \right) \\
(5.11) \quad &= \frac{1}{\sigma_\epsilon \sqrt{2\pi}} \exp \left(-\frac{(Y_t - \hat{Y}_{t|t-1}(\theta))^2}{2\sigma_\epsilon^2} \right) \\
&= \frac{1}{\sigma_\epsilon \sqrt{2\pi}} \exp \left(-\frac{\epsilon_t(\theta)^2}{2\sigma_\epsilon^2} \right)
\end{aligned}$$

Plugging (5.11) in the product term of equation (5.9) and by conditioning on Y_p yields,

$$\begin{aligned}
L(Y_T; \theta, \sigma_\epsilon^2) &= \prod_{t=p+1}^T f(Y_t | Y_{t-1}; \theta, \sigma_\epsilon^2) \\
(5.12) \quad &= (\sigma_\epsilon^2 2\pi)^{-\frac{T-p}{2}} \exp \left(-\frac{1}{2\sigma_\epsilon^2} \sum_{t=p+1}^T \epsilon_t^2(\theta) \right)
\end{aligned}$$

For numerical convenience, we use the fact that the map $x \rightarrow \log x$ is increasing and differentiable for x lying in the positive half line of real numbers and thus it is equivalent to maximize the *log-likelihood* instead of the likelihood function.

$$(5.13) \quad \log L(Y_T; \theta, \sigma_\epsilon^2) = -\frac{T-p}{2} \log \sigma_\epsilon^2 - \frac{T-p}{2} \log 2\pi - \frac{1}{2\sigma_\epsilon^2} \sum_{t=p+1}^T \epsilon_t^2(\theta)$$

The exact calculation of the MLE estimator for θ requires more sophisticated methods which stray away from the basic purpose of this paper, but we encourage the reader to refer to [OKS97]. However, we decided to provide an implicit formula for the MLE of σ_ϵ^2 , in order to illustrate the mechanics of the Maximum Likelihood Method for the estimation of parameters that ARIMA requires.

We maximize the log-likelihood by differentiating (5.13) with respect to σ_ϵ^2 and setting equal to zero.

$$(5.14) \quad \frac{\partial \log L(Y_T; \theta, \sigma_\epsilon^2)}{\partial \sigma_\epsilon^2} = -\frac{T-p}{2} \frac{1}{\sigma_\epsilon^2} + \frac{1}{2\sigma_\epsilon^4} \sum_{t=p+1}^T \epsilon_t^2(\theta) = 0$$

In several literature (5.14) is known as a so called *normal equation*. Solving for σ_ϵ^2 yields the (unique) solution,

$$(5.15) \quad \hat{\sigma}_\epsilon^2 = \frac{1}{T-p} \sum_{t=p+1}^T \epsilon_t^2(\theta)$$

Now simply by substituting σ_ϵ^2 by its estimation $\hat{\sigma}_\epsilon^2$ in (5.13) it follows that,

$$(5.16) \quad \begin{aligned} \log L(Y_T; \theta, \sigma_\epsilon^2) &= \frac{T-p}{2} \log (T-p) - \frac{T-p}{2} \log \left(\sum_{t=p+1}^T \epsilon_t^2(\theta) \right) \\ &\quad - \frac{T-p}{2} \log 2\pi - \frac{T-p}{2} \end{aligned}$$

Or more compactly, by denoting by c anything that does not depend on θ , hence constant,

$$(5.17) \quad \log L(Y_T; \theta, \sigma_\epsilon^2) = -\frac{T-p}{2} \log \left(\sum_{t=p+1}^T \epsilon_t^2(\theta) \right) + c$$

Therefore, in order to maximize the log-likelihood, we have to minimize the second factor in the product above. Since the logarithm is an increasing function, we can compute the Maximum Likelihood Estimator $\hat{\theta}_{ML}$ by *minimizing*,

$$S^2(\theta) = \sum_{t=p+1}^T \epsilon_t^2(\theta)$$

which is a function of θ . Finally, by simple calculations we obtain that the Maximum Likelihood Estimator for σ_ϵ^2 is given by,

$$(5.18) \quad \hat{\theta}_{ML} = \hat{\sigma}_\epsilon^2 = \frac{S^2(\theta)}{T - p}.$$

The likelihood principle we described above, offers a very rich family of methods for the best model selection; in our case, for the optimal parameters p, d and q . In this paper, we have chosen one of the commonly used methods, which is based on *Akaike's Information Criteria* (AIC), which is defined by,

$$(5.19) \quad AIC = 2k - 2\log(\text{MLE}) = 2k - 2\log(\hat{\theta}_{ML})$$

where k is the number of the estimated parameters in the model. In our case, we need to estimate the parameters $\beta_1, \dots, \beta_p, \gamma_1, \dots, \gamma_q$, therefore $p + q$ in total. The goal is to find the optimal pair (p, q) that minimize AIC, in our particular case,

$$(5.20) \quad AIC = 2(p + q) - 2\log(\hat{\sigma}_\epsilon^2)$$

5.2. Simulation. Now we have described the ARIMA models as well as how to estimate the model parameters, let us now design a simulation study to provide insights into the time series simulated from autoregressive integrated moving average models.

We generated an artificial ARIMA(2,1,1) time series with length $N = 10^6$, which is a suitable sample size for a typical simulation, but here it also makes sense due to the fact that high frequency data, like LOB data which is the fundamental purpose of this paper, often come with immense amount of data. Now, as discussed in the previous section, our model mathematically looks like below,

$$(5.21) \quad Y_t = \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \gamma_1 \epsilon_{t-1} + \epsilon_t$$

where $\beta_2 \neq 0$, $\gamma_1 \neq 0$ and ϵ_t are iid $N(0, \sigma_\epsilon^2)$ -distributed with $\sigma_\epsilon^2 > 0$, for $t = 2, \dots, 10^6$. For the purpose of the simulation, we chose the parameter values $\beta_1 = 0.6$, $\beta_2 = 0.3$, $\gamma_1 = 0.4$ and $\sigma_\epsilon^2 = 0.33$

By looking at *Figure 5* we observe that Y seems to be non-stationary, but we fortunately can test the significance of this hypothesis by performing a statistical test, known as *Augmented Dickey - Fuller Test (ADF)*. What an ADF-Test basically does, is to test the null hypothesis that the time series is non-stationary versus the alternative hypothesis that the time series is stationary at a pre-chosen significance level $\alpha \times 100\%$. The level of significance expresses our level of confidence in the sense that, if we performed our experiment 100 times, then $100 - 100 \times \alpha$ of times our results would be correct.

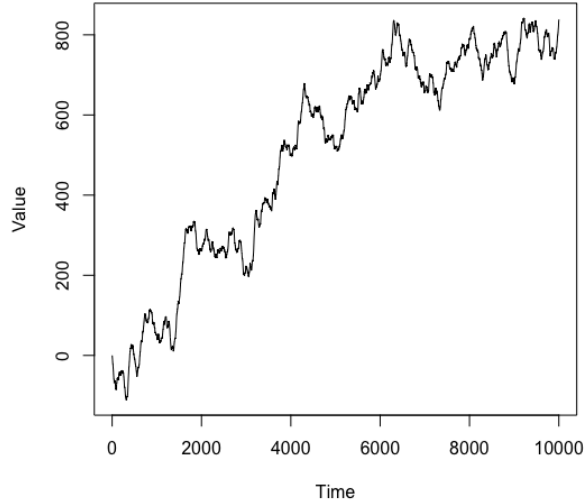


FIG. 5. Trajectory of (5.21)

What determines us to be *correct* is the probability, under H_0 , to sample a *test statistic* whose value is at least as, or more extreme than that we observed in our sample. This probability is rigorously known as *p-value* and determines whether we reject or *fail to reject* our null hypothesis H_0 versus the alternative hypothesis H_1 , where in our case,

$$H_0 : "Y \text{ is not stationary}" \text{ vs. } H_1 : "Y \text{ is stationary}"$$

As for the test statistic, here the Augmented Dickey-Fuller (ADF) statistic is a negative number with the property that the more negative it is, the more likely it is to reject our null hypothesis. We encourage the reader to have a look at [Ful76] if they are interested in more mathematical details for the ADF-Test.

Let us now return to the simulation. After we performed an ADF-test by using `R`, we obtain the following output,

Augmented Dickey-Fuller Test

```
data: data_sim
Dickey-Fuller = -2.5853, Lag order = 21, p-value = 0.3304
alternative hypothesis: stationary
```

The test returns the value of Dickey-Fuller test statistic, always negative as we aforementioned, but it also returns the p-value, which is greater than our significance level $\alpha = 0.05$. We thus fail to reject the null hypothesis that Y is non-stationary, or in more simple words there is a strong indication that our time series is not stationary. That means that we should difference the time series at least once, which

is well justified by the fact that this time series is artificially generated by an actual $ARIMA(2,1,1)$ process, and we know beforehand that the order of difference $d = 1$. But even if we say that this is not a simulation, but the ADF-Test indeed had given us such results, it would be safe to start with a value of $d = 1$ while looking for the other model parameters.

As discussed in the previous subsection, our now goal is to find out the optimal model order (p^*, q^*) for which it holds that,

$$(5.22) \quad AIC(p^*, q^*) \leq AIC(p, q) \text{ for any other } p, q$$

Then the model $ARIMA(p^*, q^*)$ is the one that best describes our time series, and it can be used to obtain very good estimates for the model parameters β_1 , β_2 and γ_1 in (5.21).

In order to give an illustration of the above, we fit the true model $ARIMA(2, 1, 1)$ in the simulated time series with sample size $N = 10^6$. The results are given below.

```
Call:
arima(x = data_sim, order = c(2, 1, 1))

Coefficients:
          ar1          ar2          ma1
      0.5291    0.3656    0.4671
s.e.    0.0365    0.0342    0.0353

sigma^2 estimated as 0.1094:  log likelihood = -31266.88,
aic = 62541.76
```

From the `R` output above, we confirm that the estimated model parameters $\hat{\beta}_1$, $\hat{\beta}_2$, $\hat{\gamma}_1$ and $\hat{\sigma}_\epsilon^2$ are very close to the true ones (0.6, 0.3, 0.4 and 0.33 respectively). We can also check if our results are significant by looking at the confidence intervals of our estimations.

Since none of the confidence intervals above contains 0, we conclude that our estimates are significant. Recall that it should also hold that the value of $AIC = 62541.76$ is the minimum among any other $ARIMA(p, q)$ model for any other model order (p, d, q) . As an example we fit an $ARIMA$ model but now with three moving average terms, that is an $ARIMA(2, 1, 3)$ and we obtain the following output for the model coefficients and their respective confidence intervals.

```
Call:
arima(x = data_sim, order = c(2, 1, 3))

Coefficients:
          ar1          ar2          ma1          ma2          ma3
      0.6868    0.2197    0.3123   -0.0055    0.0036
s.e.    0.0923    0.0855    0.0924    0.0069    0.0026
```

```
sigma^2 estimated as 0.1088:  log likelihood = -309814.2,
aic = 619640.3
```

```

          2.5 %      97.5 %
ar1  0.505842986  0.867743969
ar2  0.052109287  0.387214725
ma1  0.131255036  0.493272683
ma2 -0.019033875  0.007940230
ma3 -0.001600764  0.008780954
```

As expected, the value of $AIC = 619640.3$ is much larger than that of $ARIMA(2, 1, 1)$, where $AIC = 62541.76$. Furthermore, we notice that the confidence intervals for the two additional moving average terms $ma2 = \gamma_2$ and $ma3 = \gamma_3$ both contain 0, which means that these coefficients are not significantly different than 0 and hence we can remove them from the model. As a result, we are led back to the $ARIMA(2, 1, 1)$ model that we used for the sake of this simulation.

Of course, we obtain similar results if we fit a model with less parameters, like an $ARIMA(1, 1, 1)$ model. Again the value for $AIC = 620257$ which is again larger than $AIC = 62541.76$, namely that of the $ARIMA(2, 1, 1)$ model, as we can clearly see from the above results:

```
Call:
arima(x = data_sim, order = c(1, 1, 1))

Coefficients:
          ar1          ma1
      0.9217    0.0788
s.e.    0.0004    0.0011

sigma^2 estimated as 0.1089:  log likelihood = -310125.5,  aic = 620257

          2.5 %      97.5 %
ar1  0.92086468  0.92249924
ma1  0.07667995  0.08098086
```

Finally, the confidence intervals above suggest that both AR and MA terms are significant, which further suggests that we should include at least one AR and one MA term and thus it seems that looking in the family of mixed autoregressive moving average models is a good place to start for the model identification process. Even so, none of the two alternative models above turned out to be suitable for our simulated time series, since none achieved the minimal value of AIC .

In practise, one can use Machine Learning methods, such as Cross-Validation, in which we split up our data into a training and a testing set and we *train* our model on the former while test its performance on the latter. Thus, an algorithm is trained to find the optimal model parameters, which are chosen in such a way that the prediction error is the minimal among the ones obtained from the all the

other models, corresponding to the other combinations of parameters. In the next section, we explicitly use this approach in order to check how much accurately the ARIMA model can predict when tested on *future* data for the same optimal parameters, obtained from the training process on *past* data.

5.3. Forecasting. Let us now return our focus back on the midpoint price, for which we wish to fit a model and use it for forecasting. As we argued in the previous section, in the basic ARIMA model we need to provide the parameters (p, d, q) . We start by checking if the time series for the midpoint price is stationary or not. In accordance with what we discussed in the previous section, this can be achieved by performing an Augmented Dickey - Fuller Test (ADF). By fixing a significance level $1 - \alpha = 95\%$, we obtained a p-value equal to $0.27 > 0.05$, and thus we fail to reject the null hypothesis at a 5% level of significance. In other words, there is strong indication that the midpoint price time series is non-stationary and thus, we must make it stationary by some order of difference, before estimating the other model order terms p and q .

While we usually use statistical techniques, as described in the previous section, as well as ACF and PACF plots for determining the values of the tuning parameters (p, d, q) , we decided to use the function `auto.arima` from the R-package `forecast`. This function returns the best values combination of (p, d, q) according to the AIC criterion; that is, for every value of AIC function (5.19), for any possible combination of the parameters (p, d, q) , this function returns the combination corresponding to the minimum value of AIC. We passed through the argument `seasonal = False`, while leaving the default values for the rest of parameters.

From the R output below, we can see that the optimal parameters (p, d, q) for the ARIMA model that best describe the time series of the midpoint price is an ARIMA(1,1,3) model.

```
Series: midpoint_price
ARIMA(1,1,3) with drift

Coefficients:
          ar1          ma1          ma2          ma3  drift
      -0.0236  -0.0236  -0.0163  -0.0091      0
s.e.    0.2443   0.2442   0.0121   0.0044      0

sigma^2 estimated as 5.065e-05:  log likelihood=339462.5
AIC=-678913   AICc=-678913   BIC=-678856.2
```

In order to double-check the robustness of the results above, we review a couple of ACF and PACF plots. A widely known way to find the order of differencing d , after having having obtained from the ADF test that differencing is indeed needed, is to find the lag at which the ACF/PACF exposes a quick cut-off. By consulting the PACF plot below, we can confirm that the choice of $d = 1$ is reasonable.

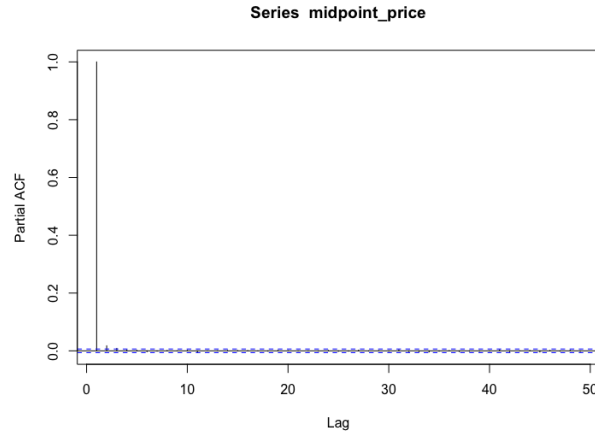


FIG. 6. PACF plot for the midpoint price

Next, another common way to estimate the number of lags p is to take the number of lags that cross the significance limit in the PACF plot of the differenced time series. From *Figure 6* above one can observe that lag 1 is highly significant, since it crosses the limit, represented by the dashed blue line, by far. Note also, that lag 2 and lag 3 turn out to be also significant, but not to the same extent as lag 1. By taking into account the results from `auto.arima`, we decided to fix the order of $p = 1$.

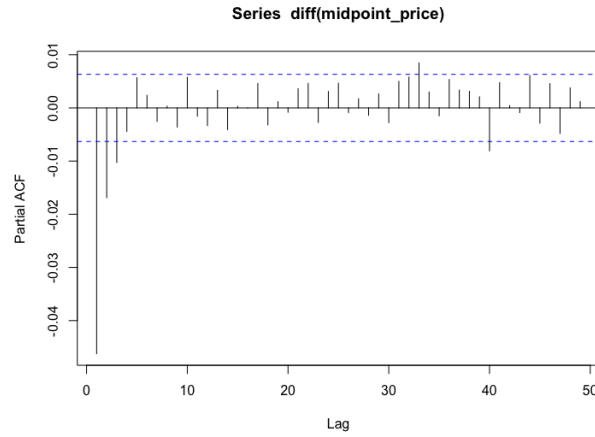


FIG. 7. PACF plot for the midpoint price

Lastly, by looking at ACF plot one can tell how many q terms one needs, to remove any autocorrelation in the stationarized series. Lets us check the autocorrelation plot of the differenced series (*Figure 8*).

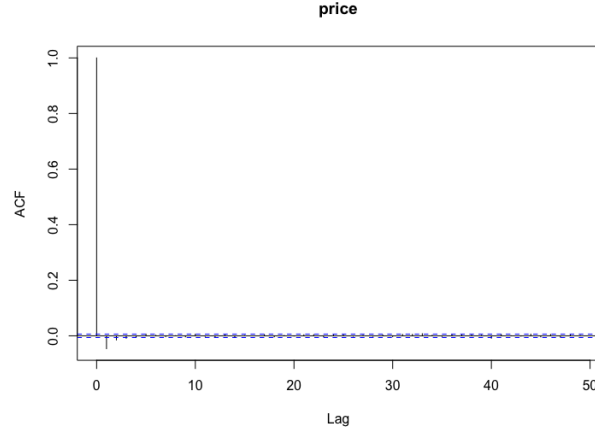


FIG. 8. ACF plot for the differenced time series

We can tell that lag 2 is more significant than lag 3, but let us trust the choice of `auto.arima` and tentatively fix $q = 3$.

6. VALIDATION

Now we have determined our model, we train our ARIMA(1,1,3) model on the first 90% entries, while we keep the rest 10% for testing the results. Again, the results obtained from `auto.arima` function when trained on the training set is an ARIMA(1,1,3), as one can confirm by the R-output below.

```
Series: (midpoint_price.train)
ARIMA(1,1,3)

Coefficients:
      ar1      ma1      ma2      ma3
    -0.0236  -0.0235  -0.0178  -0.0123
s.e.    0.1926   0.1926   0.0097   0.0046

sigma^2 estimated as 4.826e-05:  log likelihood=307604.7
AIC=-615199.3    AICc=-615199.3    BIC=-615152.5
```

The predicted values based on the test set, we kept out for valuation of the model, are plotted in *Figure 9* below in the green line. As we observe in plot the predictions made by using the ARIMA(1,1,3) model are very close to the true ones and one can say that the goodness of fit is more than satisfactory.

For numerical precision of the performance of our model, one can have a look at the table below, where the Mean Error (ME), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Percentage Error (MPE) and Mean Absolute Percentage Error (MAPE) are given. Likewise, one can confirm the goodness of

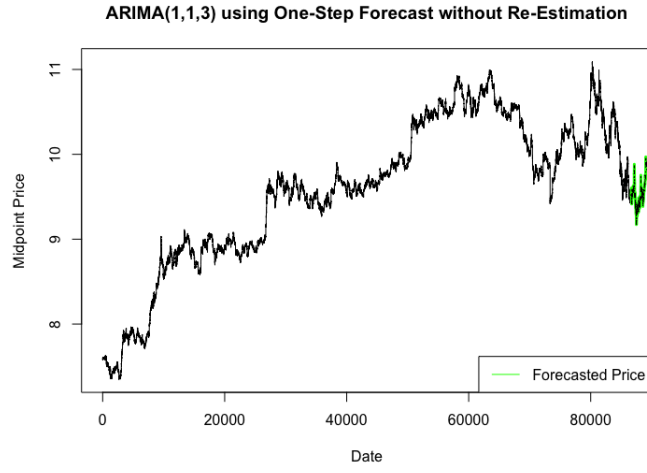


FIG. 9. Predictions for the midpoint price

fit by observing the fact that the generalization error is very small for any metric used.

ME	RMSE	MAE	MPE	MAPE
0.0001403946	0.0084888143	0.0041914873	0.0013461162	0.0424792625

TABLE 6

REFERENCES

- [Box+15] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley Sons, 2015.
- [Eck12] Andreas Eckner. “A framework for the analysis of unevenly spaced time series data”. In: *Preprint* (2012).
- [Ful76] Wayne A Fuller. “Introduction to statistical time series, new york: John-wiley”. In: *FullerIntroduction to Statistical Time Series1976* (1976).
- [Gen+01] Ramazan Gençay, Michel Dacorogna, Ulrich A Muller, Olivier Pictet, and Richard Olsen. *An introduction to high-frequency finance*. Elsevier, 2001.
- [Ham20] James Douglas Hamilton. *Time series analysis*. Princeton university press, 2020.
- [Mad07] Henrik Madsen. *Time series analysis*. CRC Press, 2007.
- [OKS97] John Keith Ord, Anne B Koehler, and Ralph D Snyder. “Estimation and prediction for a class of dynamic nonlinear statistical models”. In: *Journal of the American Statistical Association* 92.440 (1997), pp. 1621–1629.
- [WLL19] Shixiong Wang, Chongshou Li, and Andrew Lim. “Why are the ARIMA and SARIMA not sufficient”. In: *arXiv preprint arXiv:1904.07632* (2019).

Email address: christos.1995.papadakis@gmail.com



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):

First name(s):

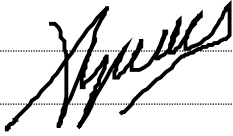
With my signature I confirm that

- I have committed none of the forms of plagiarism described in the '[Citation etiquette](#)' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date

Signature(s)

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.