Information Content of Book and Trade Order Flow at Different Trading Volume Time Scales *

Mariol Jonuzaj[†] Alessio Sancetta[‡] Yuri Taranenko[§]

September 30, 2024

Abstract

This paper studies information spillovers from the NASDAQ Limit Order Book (LOB) and assesses their impact on price predictability. Using LOB data for 35 large cap US stocks from March 2019 to February 2023, we aggregate data at different trading volume time scales and train various machine learning algorithms: Linear Discriminant Analysis, Ridge Classifiers, Random Forests and a Deep Neural Network. Our empirical findings suggest that trade order flow information is the most persistent and prices are predictable with respect to it. We document that machine learning models are able to predict mid price directions accurately, yet this informational advantage dissipates within the first 10 milliseconds. Moreover, our findings suggest that model complexity does not necessarily ensure higher financial returns. Using information on quoting activity from other exchanges, we also conclude that market participants may choose to quote more heavily on the NASDAQ, but they do so without leaking more information. Additionally, employing both panel and cross sectional analysis, we examine how stock-specific and market determinants affect intra-day predictability across different days. Overall, more liquid stocks with higher market beta exhibit higher intraday returns. We show that there is persistency in high frequency performance and the dynamic adjustment towards the long-run average lasts up to three trading days. Finally, we document that over time, the value of order flow has decreased. Then, it is plausible to infer that the growing use of algorithmic trading has increased market competition and consequently enhanced market efficiency.

Key Words: high frequency trading, information dissemination, machine learning, order book.

JEL Codes: G14, G17.

*We thank Francesco Cordoni for help with the datasets. The first two authors acknowledge financial support from the Leverhulme Trust Grant Award RPG-2021-359.

[†]Department of Economics, Royal Holloway University of London, Egham TW20 0EX, UK. Email: mariol.jonuzaj@rhul.ac.uk.

[‡]Corresponding Author. Department of Economics, Royal Holloway University of London, Egham TW20 0EX, UK. Email: asancetta@gmail.com.

[§]Abu Dhabi Investment Authority, 211 Corniche Road, Abu Dhabi, UAE. Email: Yuri.Taranenko@adia.ae.

1 Introduction

This paper studies the information generated from order book flow. This is constructed aggregating orders from the order book. It comprises all messages sent to the exchange, including trades. Hence, trade order flow is a special case (Evans and Lyons, 2001). We document information spillovers over different intraday intervals. By this we mean that information derived from order flow can be used to predict future returns. Such spillovers come mostly from trade order flow. The impact of book order flow at high frequency is well documented (Cont et al., 2015). The role of aggregated trade and book order flow on lower frequency returns has not been systematically studied. Here, lower frequency is measured in volume time and aggregation is over nearly equally spaced volume time intervals. To summarise the wealth of results we obtain in a single sentence, we conclude that, relative to book order flow, trade order flow contains information that persists more at lower frequencies and prices are predictable with respect to it.

In general a number of studies have shown that information extracted from quotes and trades at high frequency can be used to predict price movements. This information can take the form of carefully crafted derived features (Kercheval and Zhang, 2015, Sancetta, 2018, Aï-Sahalia et al., 2022, Mucciante and Sancetta, 2022, 2023), features that mimic the way the exchange sends order book updates in the form of order book flow (Cont, 2014) and features learned directly from the order book using convolution neural networks (Zhang et al., 2019). Lucchese et al. (2023) provides a comparison of different order book representations within the context of predictions using deep neural networks.

All this research focuses on ultra high frequency events. The conclusions drawn are localised to very short time intervals. All the above papers concur on the fact that the order book helps substantially in explaining much of the price variation and improves predictions of up or down movement in price. This study builds upon these documented evidence and extends the literature by examining information spillovers in Limit Order Book (LOB) data through volume based time scales. We rely on NASDAQ level three data provided by LOBSTER (Huang and Polak, 2011)[1]. Our strategy is to train various machine learning-based algorithms and investigate the informativeness of the different features. Our empirical analyses focuses on 35 large cap US stocks from March 2019 to February 2023, encompassing pivotal events like the COVID-19 sell-off, the following market recovery, and the Russian-Ukrainian conflict's impact[2]. This timeframe is of particular interest as it allows us to explore significant market dynamics and their influence on information diffusion.

It is important to note that U.S. equity markets are highly disaggregated. NASDAQ is one of the main trading venues, but not necessarily the largest. For example, Apple shares (AAPL) are most heavily quoted on NYSE ARCA and not on the NASDAQ. The ratio of orders on AAPL between the two venues in roughly one order on the NASDAQ for any four orders on Arca. As of today, there are no consolidated Level 3 order book data on stocks that are accessible under subscription.

[1]LOBSTER is based on the NASDAQ Historical TotalView-ITCH: `https://nasdaqtrader.com/Trader.aspx?id=ITCH`

[2]The exact list of stocks can be found in Table 7 in Section 4.1.

In this paper, information content is meant in the sense of predictability of the mid price direction, i.e. the average of the best bid and ask. High frequency messages are very noisy as market participants try often to conceal their intentions. This is for example the case when trading on private information (Grossman and Stiglitz, 1980). Similarly, when executing a large order, say, to buy some shares, clients of brokerage houses employ relatively advanced algorithms to avoid passing this information to other market participants. In particular, due to limited liquidity in the market, an informed trader who wants to buy a large number of shares faces a dilemma. They can buy a large number of shares over a very short time or be patient and buy very small quantities of shares at the time, over a very long period. In the former case, a large order will deplete the liquidity of the market and execute at a relatively high average price. In the latter case, they may wait that any little depleted liquidity is replenished not to upset the market liquidity. In this case, they face the risk that the price may move away considerably from the initial price as time elapses. For example, execution trading algorithms attempt to find a balance between two extremes, based on client preferences. Then, it seems plausible that on average, over certain time horizons, aggregation of order book messages may more vividly reveal patterns that emerge from execution algorithms.

When making predictions, we focus on training prediction algorithms with increasing level of complexity. The training goal is to classify the mid price change over the next volume time scale to be positive or negative or zero. The advantage of using classifiers rather than regression is twofold. Most importantly, it simplifies the problem, avoiding the necessity to deal with the many outliers commonly observed in intraday prices changes. Second, classification can be seen as a simple decision rule on the price direction. We can directly compute the profit and loss of buying and selling the stock based on the classifier's output. The models we use are Linear Discriminant Analysis, Ridge Classifiers, Random Forests and a Deep Neural Network. We did not fine tune the models on the data in order to avoid possible issues with data snooping. As in Aït-Sahalia et al. (2022) we attempt to gauge any gain in predictability using relatively similar modelling effort. In this case, our results suggest that the choice of model/estimation method is not critical. Our results provide a rough estimate on the economic value of book and trade order flow.

We also assess the value of speed when classification is seen as a decision rule to buy or sell. Our decisions are taken once we are in a new volume time interval and a new order book update is received. This way, decisions are never taken on stale quotes. In consequence, we quantify the value of speed by comparing the change in mid price after a fixed time interval. The time intervals are within the range of observed delays for trading firms relying on different infrastructures. In particular, we consider 50 and 500 microseconds as well as one and 10 milliseconds. We find that the mid price moves towards the predicted direction within the first 10 milliseconds, indicating that the information revealed in the market dissipates rapidly. This highlights the importance of specialised software, hardware and dedicated infrastructure when maximizing profits at high frequencies.

Given that we want to assess the value of information at different frequencies, we consider different volume time scales. These are 0.1%, 0.5% and 1% of the expected average daily volume.

Our findings highlight the importance of aggregation in generating profitable trading strategies. Our results suggest that as we aggregate more orders in volume time, predictability decreases. This translates also on the reduction of the achieved daily financial returns when we use the classification output as a trading rule and for benchmarking purposes we assume virtual execution at mid-price. Specifically, as we decrease the volume time scale from 1% to 0.1% of the expected average daily volume, the cumulative returns increase significantly for the majority of the stocks, over the entire sample period.

Finally we also attempt to explain how the value of order flow information can change from day to day based on changing market conditions. To this end, we carry out an in depth analysis of the prediction results using variables computed at daily frequencies. This is similar to Aït-Sahalia et al. (2022). However, we use slightly different explanatory variables which we clearly motivate. Overall, we provide evidence pointing to the direction that higher algorithmic profitability is associated with procyclical and high-volume stocks. In addition, our empirical estimates highlight the importance of market impact in exploiting profitable trading strategies, ceteris paribus. General macroeconomic and central bank announcements seem to have a mixed effect depending on the level of aggregation. Moreover, we find that the generated economic value tend to be persistent over time, and their adjustment process towards the long-run average lasts approximately three trading days. Finally, our findings suggest that returns are decreasing over the years, highlighting the fact that the growing use of algorithms has enhanced the level of price efficiency.

1.1 Motivation

High frequency data is very noisy. Time aggregation reduces the noise. Then, analysis of such aggregated data allows us to identify some average behaviour. For example, Cordoni and Sancetta (2023) present an application where they study the causal links between aggregated order book imbalances and price changes. Specifically, aggregation is in terms of trading activity rather than clock time. The application is of interest to execution algorithms that require a fine balance between posting passive orders on the order book and crossing the book paying the spread.

From now on, we shall simply use the term order flow when referring to aggregated order flow in trading activity time. The order flow is generated by the arrival of orders. We refer to this as book order flow. Some of these orders are actually executed and in this case, we refer to them as trade order flow. Hence, trade order flow is a subset of book order flow.

To motivate our analysis, we first consider how book and trade order flow can explain contemporaneous price variation. The information is extracted in the form of aggregated order flow coming from the first ten levels of book quotes and trades. The exact definitions are given in Section 3.3. We find that across 35 stocks that we use in the present paper, these order flow variables explain a sizable portion of the price return variation. Interestingly, the same regression that only uses the book order flow explains almost as much as the regression that uses both book and trade order flow. Hence, adding trade order flow does not improve the contemporaneous explanation result. This confirms the importance of order book information in explaining intraday return as found in

the high frequency trading literature. These remarks pertain to returns and order flow variables computed over the same time intervals. These are not predictive regressions. However, given the conclusions of the existing literature, which is based on data that has not been aggregated in time, we know that it is mostly the order book that explains the price changes rather than the other way around.

It is then natural to repeat the same analysis using a predictive regression where the returns are for one period ahead, relatively to the order flow variables. We stress that by one period we mean a time period measured at a fixed volume time scale (Section 3.2 discusses the construction of different volume time scales). Unsurprisingly, the R^2 of the predictive regressions drops. Surprisingly, the order flow coming from the book becomes negligible compared to the predictive power of the trade order flow. This shows that most of the predictability that comes from the book is localised in time and the information is quickly absorbed by the market.

Figure 1 illustrates our remarks. It shows the boxplot (for 35 stocks over 4 years) of the adjusted R^2 for the different regressions of log returns on book and trade order flow at a volume time frequency of 1/1000 of average daily volume. Specifically, for each stock and each day, we compute log returns over different volume time scales. These are the times over which we compute aggregation accordingly to Section 3.2. For simplicity, we have dropped the dependence on the day and the stock. For the results in Figure 1 we used 0.1% average volume time between time intervals. The log returns $\ln\left(\text{mid}_{T_i}/\text{mid}_{T_{i-1}}\right)$ are winsorized using the 99% quantile of the absolute value over the whole sample of the individual stock. Then, we capped by this value and floored by the negative of this value. Using a stronger winsorization, e.g. 97.5% quantile, leads to a higher R^2 but does not change the relative comparison among the regressions. The book and trade order flow are the ones used throughout the study.

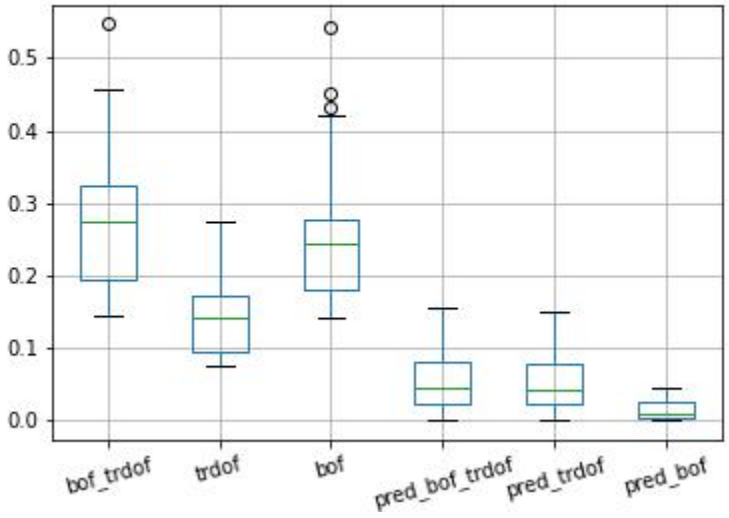
The two takeaways from these regression results are (1) returns should be predictable using order flow information and (2) the power of the book order flow in explaining returns fades faster in time than the trade order flow.

We shall build on this observation and analyse the predictability of price changes over volume times of the order of 1/1000, 5/1000 and 10/1000 of daily volume. Complementary to the recent literature on predictability at ultra high frequency, our results show that the information from the order book quickly becomes irrelevant under aggregation. Nevertheless, returns at lower frequencies are still predictable, though mostly through the information that is extracted from the trade order flow. We can conclude that information is disseminated in the market through different channels at different frequencies: at ultra-high frequencies, information is extracted from the book order flow, while at lower frequencies, the trade flow becomes the major source of information.

1.2 Outline

The paper is organized as follows. Section 2 revises how the Limit Order Book works. Section 3 outlines the paper's methodology. Section 4 presents the empirical findings. Section 5 contains concluding remarks.

Figure 1: Boxplot of Adjusted R^2 for Regression of Returns on Book and Trade Order. The boxplot is relative to 35 stocks traded on the NASDAQ. The R^2 for the regression of log returns on book and trade order flow is estimated using different subsets of these explanatory variables. The data is aggregated at volume intervals of 1/1000 of average daily volume. The bars bof_trdof, trdof, and bof stand for regression returns on book and trade order flow, trader order flow, and book order flow, respectively. The bars with the pred prefix stand for the predictive regression of returns on lag values of the corresponding predictive variables.



2 The Limit Order Book

This section describes briefly the structure of a Limit Order Book. Specifically, we shall present an example of the "Message" and "Order Book" files based on the Limit Order Book obtained from the LOBSTER database (Huang and Polak, 2011). The message file contains information on all the orders sent to the NASDAQ: the exchange timestamp in nanosecond resolution, the type of order (e.g. insertion, cancellation, trade), the size and the price. From now on, we may write market instead of NASDAQ.

The order book is reconstructed by LOBSTER and presents a snapshot of the first ten levels of the order book as a result of each message. The book levels refer to the number of shares available at different bid and ask prices. The first level gives the number of shares at the highest bid price and the lowest ask price. The next best bid and ask prices and sizes refer to the second level and so on.

Table 1 presents an illustrative example of a message book. Each row of the message book corresponds to a different trading event. Specifically, for each event we have the following information presented in a separate column[3]:

1. Time measures the arrival of each event in seconds after midnight. Specifically, the LOBSTER

[3]In addition, the message book includes the trades unique ID. However, we decide to exclude it from this illustrative example as it is irrelevant for our analysis.

Table 1: Example of Message Updates.

Time	Type	Size	Price	Bid/Ask
⋮				
34200.4031783900	1	50	16544700	Bid
34200.4598887180	4	50	16544700	Bid
34200.4600102470	1	100	16560000	Ask
34200.4637017390	1	50	16520100	Bid
⋮				

dataset uses nanoseconds time resolution as the exchange. For example, a time equal to 34200.4637017390 stands for clock time 09:30:00.463702am EST.

2. Type represents the different type events categorized as:

   - Type '1': Submission of a limit order
   - Type '2': Cancellation
   - Type '3': Deletion
   - Type '4': Execution of a visible limit order
   - Type '5': Execution of a hidden limit order

   Type '1' limit orders (passive orders) are buy or sell orders send to the exchange platform at a specified price and their execution is not guaranteed. Type '4' orders are buy or sell orders that are executed. Type '4' orders can be either limit or market order. All we know is that they are executed at a given price. In addition, Type '2' and '3' orders represent partial or complete cancellations: the agents remove orders that previously placed in the market. Lastly, Type '5' orders represent executions against an hidden order, i.e. orders that are not visible in the reconstructed order book. The example depicted in Table 1 includes only Type '1' and '4' orders.

3. Size is the number of shares available at a given price.

4. Price corresponds to the order price and is expressed in dollars times 10000. For example, a price of $165.6 is given by 1656000.

5. Bid/Ask tells us if the order is a bid or an ask. Bid orders are sent by traders who would like to buy the asset, while ask orders are sent by agents who would like to sell.

In addition, in Table 2, we present the order book corresponding to exactly the same trade events depicted in Table 1. The order book is reconstructed recursively from the message updates. For ease of illustration, our example includes only the first two levels.

As depicted in Table 2, for each trade event we have the resulting ask/bid prices and sizes for the different levels. These illustrate the sell/buy orders that agents are willing to transact, sorted

Table 2: Example of Order Book Snapshots.

Ask 1		Bid 1		Ask 2		Bid 2		...
Price	Size	Price	Size	Price	Size	Price	Size	
⋮								
16544700	50	16510000	125	16560000	1	16505700	20	
16560000	1	16510000	125	16574500	100	16505700	20	
16560000	101	16510000	125	16574500	100	16505700	20	
16560000	101	16520100	50	16574500	100	16510000	125	
⋮								

according to price. Regarding the order book dynamics, we can see that the best ask and bid prices change constantly depending on the orders arriving from the agents. For instance, the second order in the message book is an execution on the ask side as a result of a high bid. As we can see from the second row in Table 1, this order corresponds to a bid of 16544700 for 50 shares. The first row in the order book snapshot in Table 2 shows that there was an Ask 1 at the same price for the same number of shares. Hence, there is a match and the order is transacted. Given that there were exactly 50 shares available on the Ask 1, the new Ask 1 becomes the offer that used to be at level 2. The new state of the order book as a result of the order is given in the second row of Table 2.

As a final example, consider the third row in Table 1. This is an ask order at 16560000 for 100 shares. This order joins the existing Ask 1 in the second row of Table 2 and results in a total size of 101 shares available at that price, as shown in the third row of Table 2.

These examples show how orders change the order book. From the order book one can obtain valuable information such as the mid price, the spread and order flow information. These can be utilized for generating useful features for training trading models, as we will detail in Section 3.3.

3 Methodology

3.1 Notation

We shall aggregate data and evaluate predictions for different stocks over multiple days. This can result in confusion on what the quantities are. We introduce some notation that we shall adhere to.

For any positive integer n, $[n] = \{1, 2, ..., n\}$ and for any set $\mathcal{S}$, we denote by $|\mathcal{S}|$ its cardinality. We use the index d to denote day d where $d \in [n_{\text{days}}]$ (n_{days} is the total number of days). A stock s is an element in a set $\mathcal{S} := [35]$ (there are 35 stocks in our study). For each stock s on day d, the set of times of message updates together with the exchange opening time is denoted by $\mathcal{T}_{d,s}$. The update times are also known as tick times.

We shall aggregate the messages sent by the exchange based on an aggregation method to be discussed in Section 3.2. The method depends on a parameter that controls how much data is being aggregated. The parameter will be an element a in a set $\mathcal{A}$. For the aggregation scheme with parameter a, we shall define a subset $\mathcal{T}^{(a)}_{d,s}$ of $\mathcal{T}_{d,s}$ and aggregate (effectively by computing

sums) the variables between points in $\mathcal{T}_{d,s}^{(a)}$. To help the reader with the notation, we shall mostly use the lower capital letter t when referring to elements in $\mathcal{T}_{d,s}$ and the capital letter T when referring to an element in $\mathcal{T}_{d,s}^{(a)}$. In particular, we may write $\mathcal{T}_{d,s} = \{t_i : i = 0, 1, 2, ..., n_{d,s}\}$ and $\mathcal{T}_{d,s}^{(a)} = \left\{T_i : i = 0, 1, 2, ..., n_{d,s}^{(a)}\right\}$ without making explicit the dependence of the times t_i and T_j on d, s and a. Throughout, $n_{d,s}$ is the number of message updates on day d for stock s, while $n_{d,s}^{(a)}$ is the number of updates after aggregation. We refer to the latter as the number of periods on day d for stock s given aggregation parameter a. Note that $t_0 = T_0$ is the time when regular trading starts. By construction we always have that $\mathcal{T}_{d,s}^{(a)} \subset \mathcal{T}_{d,s}$. The prediction models/methods are defined by an index m with values in set $\mathcal{M} = \{\text{LDA}, \text{Ridge}, \text{RF}, \text{DNN}, \text{Avg}\}$. The meaning of the prediction methods will be made clear in Section 3.4.

3.2 Aggregation Scheme for Different Volume Time Scales

We consider aggregation based on volume time scales for each specific stock. For each stock s we compute the geometric average of the daily volume over the full sample, say AvgVlm_s. We then consider fractions of that: $q_s^{(a)} := a \times \text{AvgVlm}_s$ with $a \in \mathcal{A} := \{0.001, 0.005, 0.01\}$[4]. The number $q_s^{(a)}$ represents the number of shares traded corresponding to a fraction a of average daily volume for stock s, e.g. 1% of average daily volume when $a = 0.01$. The number $q_s^{(a)}$ is fixed and does not change from day to day. Then, on each day d, we construct a subset $\mathcal{T}_{d,s}^{(a)}$ of the time updates $\mathcal{T}_{d,s}$ that satisfies the following property. Start with $\mathcal{T}_{d,s}^{(a)} = T_0$, i.e. the trading start time (9:30am). Find the first time update in $\mathcal{T}_{d,s}$ such that the traded volume is at least $q_s^{(a)}$. Denote this by T_1 and redefine $\mathcal{T}_{d,s}^{(a)} := \mathcal{T}_{d,s}^{(a)} \cup \{T_1\}$ [5]. Find the first update after T_1 in $\mathcal{T}_{d,s}$ such that the traded volume starting from time T_1 is at least $q_s^{(a)}$. Denote this by T_2. Redefine $\mathcal{T}_{d,s}^{(a)} := \mathcal{T}_{d,s}^{(a)} \cup \{T_2\}$. Repeat until we reach the last element in $\mathcal{T}_{d,s}$. Always set the last element in $\mathcal{T}_{d,s}^{(a)}$ equal to the last element in $\mathcal{T}_{d,s}$ irrespective of whether the traded volume from the last update is at least $q_s^{(a)}$. It is worth noting that the number of updates in $\mathcal{T}_{d,s}^{(a)}$ is not twice the number of updates in $\mathcal{T}_{d,s}^{(2a)}$ due to the discrete nature of orders coming to the market and the way we construct the set $\mathcal{T}_{d,s}^{(a)}$.

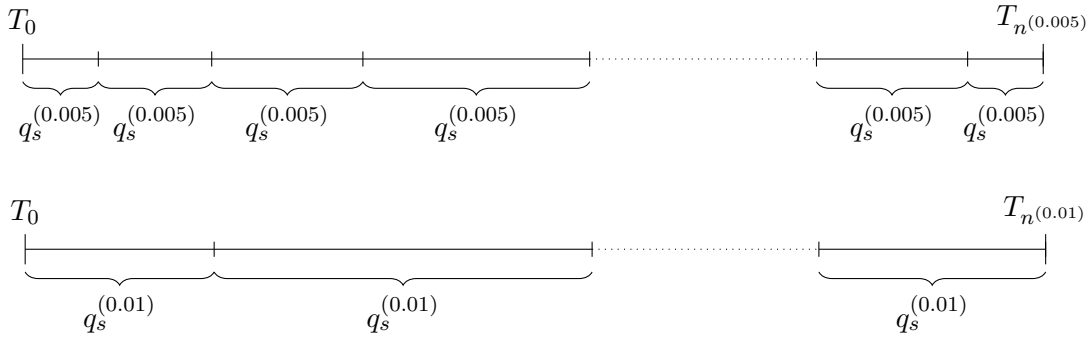
Figure 2 shows two illustrative examples of different volume time scales. Consider a trading day d and a stock s. Given a number a representing a fraction of average daily volume, the straight horizontal line depicts the trading day starting from T_0 and ending to $T_{n^{(a)}}$. Here, for simplicity, we drop the subscripts d, s in $n^{(a)}$. The trading period is split into $n_{d,s}^{(a)}$ intervals, where in each interval $a \times 100$ percent of the expected total volume is traded.

For example, let's assume that for a specific stock s the geometric average of the daily volume over the full sample is $\text{AvgVlm}_s = 1,000,000$ shares. Thus, for the different $a \in \{0.001, 0.005, 0.01\}$ aggregation schemes, the corresponding number of traded shares within each interval should be at

[4]The specific values are somewhat arbitrary, but cover a wide range. For example, the trading volume during the first few seconds after the open and before the close can be as high as 1%. On the other hand during the middle of the day it may take relatively long time to trade 1% of the average trading volume. Hence, we also seek values that are much smaller in order to capture high frequency order flow. This is why we consider a value as small as $a = 0.001$.

[5]To reduce the notational burden, we use the same symbol $\mathcal{T}_{d,s}^{(a)}$ after taking union. This should cause no confusion.

Figure 2: Volume Time Scales. Two examples of volume time scales are constructed for a fraction a equal to 0.005 and 0.01, i.e. 0.5% and 1% of the expected average daily volume. For simplicity, the figures assume that the number of intervals for $a = 0.005$ is twice the number of intervals obtained for $a = 0.01$. Due to the nature of the discrete volume updates, this is not necessarily the case.



least $q_s^{(0.001)} = 1,000$, $q_s^{(0.005)} = 5,000$ and $q_s^{(0.01)} = 10,000$, respectively. Hence, the length of each time scale depends on how fast those $q_s^{(a)}$shares are traded and they vary across the different stocks s and trading days d. It is clear, that during days of high trading activity, the time required for the $q_s^{(a)}$ shares to be traded will be shorter. Therefore, the cardinality of $\mathcal{T}_{d,s}^{(a)}$ will be larger. This implies that also the number of trading decisions will be increasing, as well, as the $q_s^{(a)}$ is fixed for all trading days. For instance, during a day when $1,200,000$ shares are traded, for an aggregation scheme $a = 0.01$ (i.e. $q_s^{(0.01)} = 10,000$), there will be at most 120 different time updates $n_{d,s}^{(0.01)}$. On the other hand, during a day with low trading activity, i.e. only $900,000$ shares are traded, there will be at most 90 time updates[6].

Given the U-shaped pattern of intraday volume, the intervals are expected to be shorter at the start and the end of each trading day.

Recall the definition of the times $\mathcal{T}_{d,s}^{(a)} = \left\{T_i : i = 0, 1, 2, ..., n_{d,s}^{(a)}\right\}$. Let $X := \{X_t : t \in \mathcal{T}_{d,s}\}$ be variables observed at times $\mathcal{T}_{d,s}$. For ease of reference, we shall define $\text{Agg}\left(X, \mathcal{T}_{d,s}^{(a)}\right)$ to mean that we take X and return $\left\{X_{T_i}(a) : T_i \in \mathcal{T}_{d,s}^{(a)}\right\}$ where $X_{T_i}(a) := \sum_{t \in \mathcal{T}_{d,s} \cap (T_{i-1}, T_i]} X_t$. This means that we aggregate within the intervals specified by the times in $\mathcal{T}_{d,s}^{(a)}$. Note that we defined $\mathcal{T}_{d,s}^{(a)} \subset \mathcal{T}_{d,s}$. With this notation,

$$X_{T_i}(a) := \text{Agg}(X, \{T_{i-1}, T_i\})$$

represents the variables X summed over the times $(T_{i-1}, T_i]$. We shall also define

$$\text{RelAgg}\left(X, \mathcal{T}_{d,s}^{(a)}\right) \tag{1}$$

[6]To avoid look forward bias, $\mathcal{T}_{d,s}^{(a)}$ is constructed recursively. Given the nature of trade sizes, the number of elements in $\mathcal{T}_{d,s}^{(a)}$ will often be less than the number of traded shares in a day divided by $q_s^{(a)}$.

as the array of $n_{d,s}^{(a)}$ variables whose i^{th} element is defined as

$$\text{RelAgg}\left(X, \{T_{i-1}, T_i\}\right) = \frac{\text{Agg}\left(X, \{T_{i-1}, T_i\}\right)}{\text{Agg}\left(\{|X_t| : t \in \mathcal{T}_{d,s}\}, \{T_{i-1}, T_i\}\right)}.$$

The above is such that the aggregated variable takes values in $[-1, 1]$ by construction.

3.3 Order Flow and Targets

In this section we define the book and trade order flow and the target variable used for training the classification algorithms.

3.3.1 Order Flow for Intra-Day Predictions

In what follows, to ease notation, we shall suppress the indexes that refer to the day and stock ticker, i.e. we shall omit the indexes $s \in \mathcal{S}$ and $d \in [n_{\text{days}}]$.

Book flow. We start from order book snapshots of the first ten levels. These snapshots represent the full supply and demand for ten best price levels on the bid and ask side. In particular, for a given stock and day, and update $t \in \mathcal{T}_{d,s}$, every level $l \in [10]$ on the bid contains a price $P_t^B(l)$ and a size $S_t^B(l)$, and similarly for the ask, using a superscript A instead of B. On each day, we transform these quantities into bid order flow

$$O_{t_i}^B(l) := S_{t_i}^B(l) 1_{\left\{P_{t_i}^B(l) \geq P_{t_{i-1}}^B(l)\right\}} - S_{t_{i-1}}^B(l) 1_{\left\{P_{t_i}^B(l) \leq P_{t_{i-1}}^B(l)\right\}}$$

and ask order flow

$$O_{t_i}^A(l) := -S_{t_i}^A(l) 1_{\left\{P_{t_i}^A(l) \leq P_{t_{i-1}}^A(l)\right\}} + S_{t_{i-1}}^A(l) 1_{\left\{P_{t_i}^A(l) \geq P_{t_{i-1}}^A(l)\right\}}.$$

The order flow imbalance is computed as the sum of the two and is given by

$$\text{OI}_t(l) = O_t^B(l) + O_t^A(l).$$

The last definition is the one in Cont et al. (2014) applied to multiple levels of the order book. The intuition is that the if prices do not move on the bid side, an increase in the posted size from $S_{t_{i-1}}^B(l)$ to $S_{t_i}^B(l)$ leads to a positive bid order flow equal to $S_{t_i}^B(l) - S_{t_{i-1}}^B(l)$. If there is a price increase on the bid, the bid order flow is $S_{t_i}^B(l)$, whereas a decrease in the bid price is associated to a negative bid order flow equal to $-S_{t_{i-1}}^B(l)$. For ask order flow, the argument is reversed, as an increase in the ask size or a lower ask price imply an increase in sell pressure.

Given $\mathcal{T}_{d,s}^{(a)} = \left\{T_i : i = 0, 1, 2, ..., n_{d,s}^{(a)}\right\}$, define

$$\text{BF}_{T_i}(l) := \text{RelAgg}\left(\{\text{OI}_t : t \in \mathcal{T}_{d,s}\}, \{T_{i-1}, T_i\}\right).$$

Table 3: Features for High Frequency Prediction.

	Short Name
Order Book Flow	BF (l); level $l = 1, 2, ..., 10$
Trade Flow	TF
Mid Price Change	dMid

We call this quantity the book order flow. For simplicity, we have also dropped dependence on the aggregation parameter $a \in \mathcal{A}$.

Trade flow. To define trade order flow, we first define the signed trade size as

$$\text{SignedTradeSize}_t = \text{sign}\left(\text{TradeSize}_t\right) \times \text{TradeSize}_t$$

where $\text{sign}\left(\text{TradeSize}_t\right)$ is 1 if the trade is executed at the prevailing mid price or above, -1 if executed below the prevailing mid price, and zero if there was no trade at update $t \in \mathcal{T}_{d,s}$[7].

We aggregate the above and define the trade flow:

$$\text{TF}_{T_i} := \text{RelAgg}\left(\left\{\text{SignedTradeSize}_t : t \in \mathcal{T}_{d,s}\right\}, \left\{T_{i-1}, T_i\right\}\right).$$

Mid price change. Mid price changes are used to account for momentum in prices. The mid price is defined as $\text{mid}_{t_i} := 0.5\left(P^B_{t_i}(1) + P^A_{t_i}(1)\right)$. Then, the mid price change is defined as

$$\text{dmid}_{t_i} := \text{mid}_{t_i} - \text{mid}_{t_{i-1}}$$

where $t_i \in \mathcal{T}_{d,s}$. The aggregated mid price change is defined as

$$\text{dMid}_{T_i} = \text{RelAgg}\left(\left\{\text{dmid}_t : t \in \mathcal{T}_{d,s}\right\}, \left\{T_{i-1}, T_i\right\}\right).$$

Features summary. The total number of covariates is 12. We refer to these as features. For ease of reference, we report the features in Table 3.

Many more features can be constructed, but the focus of this paper is on understanding the information content of book and trade order flow. A number of other studies have consider large number of features (Kercheval and Zhang, 2015, Sancetta, 2018, Aït-Sahalia et al., 2022, Mucciante and Sancetta, 2023).

We want to focus on the order flow and how the information it provides dissipates over different time scales. It is easy to add lag values and other information and end up with hundreds of features. However, this would lead to a lack of focus in our analysis. For this reason, we do not consider any lag values of the aggregated features.

[7] A very low fraction of traded prices was at mid. Classifying a trade price as a buy is a simplifying assumption. This assumption has no consequences for our conclusions on the economic value of predictions. To estimate the economic value of predictions, the target is price change.

3.3.2 Target

The target is the variable that we want to predict when we train the prediction algorithms using supervised learning. For reasons that will become apparent the target is different from the mid price change or its sign. To ease notation, we drop the dependence on the day d, stock s and aggregation parameter a in what follows. For each day and stock, let

$$\text{avgMid}_{T_{i+1}} = \text{Avg}\left(\{\text{mid}_t : t \in \mathcal{T}_{d,s}\}, \{T_i, T_{i+1}\}\right).$$

Then, define

$$Y_{T_{i+1}} = 1\left\{\text{avgMid}_{T_{i+1}} > \text{mid}_{T_i}\right\} - 1\left\{\text{avgMid}_{T_{i+1}} < \text{mid}_{T_i}\right\} \tag{2}$$

with values in $\{-1, 0, 1\}$, for $T_i < T_{i+1} \in \mathcal{T}_{d,s}^{(a)}$. Recall that the time intervals $(T_{i-1}, T_i]$ are used to construct the aggregated variables. Throughout, $1\{\cdot\}$ is the indicator function: it takes value one if the argument is true and zero otherwise. Using the aggregate features for update T_i, we train different algorithms to predict $Y_{T_{i+1}}$. The advantage of using a discrete variable is that noise in the estimation is reduced. This clearly comes at a loss of information. However, prediction of high frequency returns can be very noisy and this can reduce the performance of more complex models. Under the assumption that prices are corrupted by market microstructure noise, we predict the sign of $\text{avgMid}_{T_{i+1}} - \text{mid}_{T_i}$ as this should include the same information as the sign of the mid change, but with averaged microstructure noise. Given that we sample mid prices at times when trades occur at relatively high frequencies, averaging should lead to a less noisy target variable. Finally, the use of the average reduces the occurrence of zeros in the target variable. We can encounter a large proportion of zeros when using mid price changes if we aggregate over relatively small volume time scales.

3.4 Models

We consider four different models for prediction of the multi-class variable $Y_{T_{i+1}}$. In order of complexity, we predict using Linear Discriminant Analysis (LDA), Ridge Classifier (Ridge), Random Forest (RF) and Deep Neural Networks (DNN). The LDA estimator is computed with no penalty, in its standard specification. The Ridge uses a one versus all approach for classification. For each class value $y \in \{-1, 0, 1\}$ it converts the target to one if the target take the specified value y and minus one otherwise. It then runs a ridge regression for each value of y. The ridge parameter is chosen over a grid of values using generalised cross-validation. This is almost as fast as linear discriminant analysis, however, it does include a regularization. This can be helpful, as the ten level of the book order can be highly correlated. The RF estimator is computed aggregating 500 trees, with maximum depth of 15 and some additional constraints to reduce noise on the final leaves. For the DNN we use a simple architecture with two hidden layers that maps the 12 features into 64 nodes. The second layer takes this 64 nodes and maps them again into 64 nodes. We then use a softmax activation function to map the 64 nodes into three classes. The decision to use 64 nodes is

Table 4: High Frequency Prediction Methods.

Method	Short Name
Linear Discriminant Analysis	LDA
Ridge Classification	Ridge
Random Forest	RF
Deep Neural Network	DNN
Average Prediction	Avg

to allow the network to discover features. Finally, we predict using the average prediction from the above methods. We shall refer to each method using the short name reported in Table 4.

The estimation are done using the Python libraries scikit-learn and keras for DNN.

3.5 Performance Evaluation

We discuss a number of metrics that will be used to assess the value of the information content in order flow variables.

3.5.1 Classification Measures

Given that the target variable in (2) only takes three values, the prediction problem is the one of classification. To keep notational burden to the minimum, in this section, we omit dependence on the day $d \in [n_{\text{days}}]$, the stock $s \in \mathcal{S}$, the aggregation parameter $a \in \{0.001, 0.005, 0.01\}$, and the prediction model $m \in \{\text{LDA}, \text{Ridge}, \text{RF}, \text{DNN}, \text{Avg}\}$ as in Table 4. Then, we shall denote the prediction for Y_{T_i} by $\hat{Y}_{T_i}$. We consider standard classification measures: accuracy, precision and recall. Because of the use of future average price in the definition of the target in (2), the occurrence of zeros is negligible relatively to the values -1 and 1. Hence, to simplify the interpretation of results, we exclude the class $\{0\}$ when computing precision and recall. Hence, we effectively compute the sample estimates of the following quantities

$$\frac{1}{3} \sum_{y \in \{-1,0,1\}} \Pr\left(\hat{Y}_{T_i} = y, Y_{T_i} = y\right) \text{ (population accuracy)}$$

$$\frac{1}{2} \sum_{y \in \{-1,1\}} \Pr\left(Y_{T_i} = y | \hat{Y}_{T_i} = y\right) \text{ (population precision excluding } y = 0)$$

$$\frac{1}{2} \sum_{y \in \{-1,1\}} \Pr\left(\hat{Y}_{T_i} = y | Y_{T_i} = y\right) \text{ (population recall excluding } y = 0)$$

The accuracy is an unweighted quantity across the classes $\{-1, 0, 1\}$, while the others are equivalent to giving zero weight to the class $\{0\}$ and equal weight to the elements in $\{-1, 1\}$. Together, these quantities provide a reasonably complete picture of the classification performance.

3.5.2 Returns Based Measure

Standard classification measures of performance may not necessarily provide an indication of the value of the economic information. For example, one model may predict better than another, but in practice this may not materialize into a potentially higher economic profit. Suppose that one method only classifies well price changes that are very small. Another method better classifies relatively large price changes. Now assume that there are relatively more small price changes than large ones. This can imply that the former method has a higher accuracy than the second. However, this may not make it a preferable method. For this reason, we look at a measure that also accounts for price magnitude and turns classifications into decisions. We use the following simple measure. On day d for stock s, the value of the order flow prediction is

$$\text{Ret}_d = 100 \times \frac{\text{Sum}\left(\left(\text{mid}_{T_i} - \text{mid}_{T_{i-1}}\right)\hat{Y}_{T_{i-1}}\right)}{\text{openMid}_d} \tag{3}$$

where $\text{Sum}(\cdot)$ stands for the average over a given day, mid_T is the mid price of the stock at time T during the day and openMid_d is the mid price at the opening of day d. As usual, the times are the ones in $\mathcal{T}_{d,s}^{(a)}$ as in Section 3.1. We scale by openMid_d so that quantities are in the same scale across stocks. We multiply by 100 so that the return is expressed in percentages. Here, $\hat{Y}_{T_i}$ is the prediction for Y_{T_i} obtained, out of sample, by the classification rule from the models described in Section 3.4.

In the actual performance analysis, we shall work with the average return per period

$$\text{AvgRet}_d = 10^4 \times \frac{\text{Avg}\left(\left(\text{mid}_{T_i} - \text{mid}_{T_{i-1}}\right)\hat{Y}_{T_{i-1}}\right)}{\text{openMid}_d}, \tag{4}$$

where Avg stands for average within the day. For our analysis we choose to compute (4) instead of (3) to ensure that results are invariant with respect to days of high daily volumes. A typical example is the period of the Covid pandemic. The factor 10^4 is used to express this quantity in basis points rather than percentages[8].

3.5.3 The Value of Features in Price Discovery

We evaluate the information content in the different levels of the order book and the other variables. We do so by independent randomly reshuffling of the rows of the features for which we want to assess the importance. To ease notation, we drop dependence on the day d, the stock s, the aggregation parameter a and the prediction model m. Let f_T be the vector of features (after aggregation) at time T for a given stock. The features are as defined in Section 3.3.1 (see Table 3). Let $\hat{Y}_{T_{i+1}} := \hat{P}(f_{T_i,s})$ be the prediction from an estimated model, i.e. $x \mapsto \hat{P}(x)$ maps features to

[8]This is to ensure that that the intraday relative changes in (4) are in a scale that is not minuscule. For (3) the use of percentages is more natural, as basis points would produce very large numbers when computing returns over a daily horizon.

Table 5: Groups for Features' Importance.

BF (1)
BF (2), BF (3)
BF (l), $l > 3$
TF
dMid

predictions of the variable $Y_{T_{i+1}}$, as defined in (2). Let $f_{T_i}^{(j)}$ be the j^{th} feature. We consider the importance of groups of features. In particular, let $\mathcal{J} \subset \{1, 2, ..., 12\}$ (recall that we have 12 features) be the index corresponding to some group of features. For each $j \in \mathcal{J}$ we replace $\left\{ f_T^{(j)} : T \in \mathcal{T}_{d,s}^{(a)} \right\}$ with $\left\{ f_{\sigma(T)}^{(j)} : T \in \mathcal{T}_{d,s}^{(a)} \right\}$ where $\left\{ \sigma(T) : T \in \mathcal{T}_{d,s}^{(a)} \right\}$ is a random permutation of the updates in $\mathcal{T}_{d,s}^{(a)}$. Note that permutation is performed independently for each feature in $\mathcal{J}$. If we fix j and view $f^{(j)} := \left\{ f_T^{(j)} : T \in \mathcal{T}_{d,s}^{(a)} \right\}$ as a column vector, we just reshuffle the rows of $f^{(j)}$. Let $\left\{ f_T^{\mathcal{J}} \right\}$ be the data where we only permute the rows of $\left\{ f_T^{(j)} : T \in \mathcal{T}_{d,s}^{(a)}, j \in \mathcal{J} \right\}$ and leave the remaining as they are. We compute the prediction $\hat{Y}_{T_{i+1}}^{\mathcal{J}} = \hat{P}\left(f_{T_i}^{\mathcal{J}} \right)$ and the performance measures from Sections 3.5.1 and 3.5.2. We then compare the performance measures when using $\hat{Y}_{T_{i+1}}$ with the ones computed using $\hat{Y}_{T_{i+1}}^{\mathcal{J}}$ by taking their difference. In particular, we compute the following features importance score:

$$\text{Score}^{\mathcal{J}} := \frac{\text{Avg}\left(\text{Performance}_d - \text{Performance}_d^{\mathcal{J}}\right)}{\text{Avg}\left(|\text{Performance}_d|\right)} \tag{5}$$

where the average is across all days, Performance_d is the performance on day d, and $\text{Performance}_d^{\mathcal{J}}$ is the performance when we permuted the features corresponding to index $\mathcal{J}$. The performance measures can be statistical, as in Section 3.5.1, or returns as in Section 3.5.2. We expect that if the features corresponding to the indexes in $\mathcal{J}$ are irrelevant, out of sample, the t-statistic for the estimated difference will be small and insignificant. This is because random permutation is equivalent to replacing the value of that feature with a random one from the unconditional distribution of the feature. If the feature is not important, the prediction should not be affected in a significant way. The groups of features we test for are reported in Table 5. The choice is motivated by some evidence that the first three levels are the most important for high frequency trading. The top of book is the one where there is most competition in order to be filled relatively quickly without crossing the spread. This motivates our decision to split the ten levels of the book flow into three separate groups. We then also consider the remaining variables, i.e. the trade flow and the price change to assess their role in price discovery. We also note that the importance of the variables can change based on the level of aggregation we choose and the prediction method.

3.5.4 The Value of Speed Advantage

Here, we shall quantify the value of speed advantage when taking a decision. Again, we shall suppress dependence on the day d, the stock s, the aggregation parameter a and the prediction

model m. The Ret in (3) is equivalent to immediate execution at the prevailing mid price. We consider the deterioration in the prediction when mid_{T_i} in (2) is replaced by $\text{mid}_{T_i+\Delta}$, i.e. when there are Δ microseconds delay. We allow $\Delta \in \{50, 500, 1000, 10000\}$. A delay of 50 microseconds to parse the exchange message, compute a signal, turn it into a decision and send an order to the exchange is exceptional. This can only be achieved by market participants who are co-located with the exchange matching engine and possibly with a server in a rack on the immediate vicinity of the matching engine. A delay of 500 microseconds is possible for co-located market participants with leading hardware and software infrastructure. A delay of 10 milliseconds (i.e. 10,000 microseconds) is on the other hand of the spectrum, but not achievable by a retail client with internet connection[9].

At time $T_i \in \mathcal{T}_{d,s}^{(a)}$, we compute the cost of reacting with a delay equal to Δ microseconds as

$$\text{costdelay}_{T_i}(\Delta) := (\text{mid}_{T_i+\Delta} - \text{mid}_{T_i})\,\hat{Y}_{T_i}. \tag{6}$$

where $\text{mid}_{T_i+\Delta}$ is the prevailing mid price at clock time $T_i + \Delta$. Note that $\text{costdelay}_{T_i}(\Delta)$ depends on the prediction $\hat{Y}_{T_i}$. This is rightly so because we want to measure the cost in the direction of the predicted stock direction. Thus, equation (6) captures the potential profitability missed due to execution delay. We compute the average for each day scaled by openMid_d:

$$\text{CostDelay}_d(\Delta) = 10^4 \times \text{Avg}\left(\text{costdelay}_{T_i}(\Delta)\right)/\text{openMid}_d.$$

This quantity can be scaled by the return in (4) to give

$$\text{RelCostDelay}_d(\Delta) = \frac{\text{CostDelay}_d(\Delta)}{\text{AvgRet}_d} \tag{7}$$

where AvgRet_d is the return given in (4). In simpler terms, $\text{RelCostDelay}_d(\Delta)$ measures the cost of latency relative to the return a trader would have if their trades were executed without any delays. Therefore, for a given delay Δ, if the ratio is close to 0, it indicates minimal cost of latency. On the hand, higher values of the ratio would suggest higher profitability losses due to software, hardware and infrastructure constraints.

3.6 Explaining the Performance

We carry out an explanatory analysis of the performance results, where performance is as defined in (4). We define a set of explanatory variables and provide a short motivation for their choice.

3.6.1 Explanatory Variables

We consider a number of explanatory variables to carry out an analysis as in Aït-Sahalia et al. (2022), but with somehow different variables to identify a few stylised facts about high frequency

[9]These claims are based on discussions that one of the authors had with senior technologists at proprietary trading firms. Numbers publishes by Nasdaq suggest that the orders of magnitude of these latencies are correct `https://www.nasdaq.com/docs/2020/01/15/Metro_Millimeter_Wave_FAQ.pdf`.

predictions[10]. We shall compute these variables for every day and use them as regressors for that day. We use the hat notation to mean that population quantities are replaced by sample versions for each day.

Using one minute data, for each day and stock, we estimate the following quantities.

1. Stock specific market beta:

$$\text{Beta} := \frac{\widehat{Cov}\left(\text{RetStock}, \text{RetSPY}\right)}{\widehat{Var}\left(\text{RetSPY}\right)},$$

where RetStock and RetSPY are the log returns of the stock and the SPY, respectively. The SPY is the ETF on the S&P500 index. The stocks studied here are constituents of the S&P500.

2. A proxy for market impact based on the square root model (Grinold and Kahn, 1999),

$$\text{MarketImpactStock} := \frac{\text{VolStock}}{\sqrt{10^{-6}\text{VolumeStock}}},$$

where

$$\text{VolStock} := \sqrt{\text{N}_{\text{RetStock}} \times \widehat{Var}\left(\text{RetStock}\right)}$$

and the variance is computed using the one minute log returns, where $\text{N}_{\text{RetStock}}$ is the number of minutes in a specific the trading day for the stock.

3. High volume regime

$$\text{HighVolumeRegime} := 1_{\{\text{VolumeStock} \geq q_{75}\}},$$

where q_{75} is the 75^{th} quantile of the stock volume over the whole sample and VolumeStock is the traded volume of the stock on the given day.

Using daily data for the VIX, we compute the following:

4. High VIX regime index

$$\text{HighVIX} := 1_{\{\text{VIX} \geq 30\}}.$$

The number 30 in the VIX regime is chosen according to one of the thresholds defined by the CBOE VIX Tail Hedge Index[11]. Overall, during the period between 01/03/2019 and 28/02/2023 there are 135 occurrences of High VIX index regime.

5. Jumps in the daily stock return

$$\text{RetJumpStock} := 1_{\{|\text{RetStockDaily}| \geq 2\text{VolStock}\}}.$$

where VolStock is the volatility of the stock during the day, as previously defined, and RetStockDaily is the daily log return between open and close.

[10] We also found that, in our sample, some of the variables used in Aït-Sahalia et al. (2022) were relatively highly correlated and tended to capture the same idea.

[11] `https://www.cboe.com/us/indices/dashboard/VXTH/`

Table 6: Market Moving Announcements for S&P500. The list of announcements is taken from Kurov et al. (2019).

Announcements
Nonfarm Employment
ISM Manufacturing Index
Initial Jobless Claims
ADP Employment Report
Advance Retail Sales
ISM Non-Manufacturing Index
CB Consumer Confidence Index
Pending Home Sales
Consumer Price Index
Existing Home Sales
GDP Preliminary
Durable Goods Orders
Housing Starts
GDP Advance
UM CSI Preliminary
New Home Sales
GDP Final
Industrial Production

6. Jump in overnight stock return

$$\text{OvNRetJumpStock} := 1_{\{|\text{OvNRetStock}| \geq 2\text{VolStockLag}\}},$$

where OvNRetStock is the return between the closing the day before and the opening of the current day and VolStockLag is the lag of VolStock.

We also consider a number of dummy variables.

7. We generate the Macro binary variable which equals to one for the days in which we have market moving announcements for the S&P500. Market moving announcements are as defined in Kurov et al. (2019). These are reported in Table 6. We refer the reader to Kurov et al. (2019) for their release frequency and other details.

8. We generate the FOMC dummy variable for dates in which there was an Federal Open Market Committee (FOMC) meeting.[12] This was also used in in Aït-Sahalia et al. (2022).

9. Friday dummy: one if the weekday is a Friday.

10. Summer dummy: one if the calendar month is July, August or September.

11. Year 2 dummy: one if the calendar day is in Mar/2020-Feb/2021.

12. Year 3 dummy: one if the calendar day is in Mar/2021-Feb/2022.

[12]Since 2019, the Federal Reserve has held a press conference after every meeting. Before then, press conferences where held every other meeting and believed to have higher market impact, possibly requiring to distinguish between meetings with and without press conference.

13. Year 4 dummy: one if the calendar day is in Mar/2022-Feb/2023.

3.6.2 Remarks on Control Variables

We remark on the choice of each explanatory variable we use.

1. The stock specific market Beta captures the sensitivity of mid price returns on the market movements. The idea is to verify whether algorithmic predictability is associated with the stock's procyclicality: high beta stocks are considered to be more procyclical. We expect this variable to have a positive effect on returns, as stocks with higher market beta generally exhibit higher trading activity[13].

2. The MarketImpactStock index captures the sensitivity of mid price movements due to transaction trades. Stocks with lower MarketImpactStock index are more liquid and therefore their mid price changes less as a result of executed orders. On the other hand, tickers with higher index tends to exhibit higher mid price volatility relative to the total volume traded. We expect predictions based on order flow be more profitable in stocks which are subject to higher market impact, ceteris paribus. This is because the price is more sensitive to execution of orders.

3. The HighVolumeRegime indicator allows to control for period of excess transaction volumes. We expect the returns from predictions to be higher during days of high transaction volumes.

4. The HighVIX index measures the market's perception of risk. During periods of high risk, traders may decide to increase speed of execution, thus potentially revealing more information through trading. This in turn should improve predictions based on order flow.

5. The RetJumpStock captures whether the daily return exceeds the double of its intraday volatility. In this way, we can control for days with high volatility adjusted returns.

6. The OvNRetJumpStock index controls whether the overnight return exceeds the double of the previous intraday volatility. Using this variable we want to control how much market news and events outside the normal trading hours influence the next day's predictability.

7. We include year effects to test whether the profitability of predictions based on order flow has decreased over the years. Over time, more participants should have adopted algorithms to trade, resulting in increased competition and reduced predictability.

8. Finally, we choose to control for summer months and the Friday weekday, as there is reduced trading activity during summer months, while there can be an increase in activity during Fridays. Higher activity should be associated with higher profitability.

[13]The Pearson correlation coefficient between market beta and the trading volume is 0.422.

Table 7: List of Tickers. Tickers are grouped in terms of activity level. High includes the most liquid stocks with volumes traded in the tens of millions, Mid are stocks with volumes traded in the millions and Low are stocks with volumes traded in the hundred of thousands. In addition, we report the average, the min and the max daily volumes traded (in millions).

High				Mid				Low			
	Average	Min	Max		Average	Min	Max		Average	Min	Max
AAPL	8.043	0.119	41.359	CF	0.355	0.001	1.429	BLK	0.058	0.000	0.914
AMD	7.426	0.229	28.784	ROST	0.499	0.006	4.122	ABMD	0.092	0.000	1.115
TSLA	3.603	0.041	24.340	TSN	0.266	0.000	1.429	VRSK	0.162	0.000	0.748
AMZN	1.566	0.005	20.243	COF	0.354	0.001	1.680	CPT	0.071	0.001	0.511
F	3.715	0.057	18.565	ICE	0.289	0.001	1.455	AWK	0.087	0.000	0.342
CCL	2.752	0.031	17.037	OKE	0.366	0.001	2.944	ATO	0.099	0.000	0.314
NVDA	2.744	0.043	14.24	LYB	0.269	0.003	1.577	PSA	0.084	0.000	0.316
T	2.645	0.058	18.327	PGR	0.340	0.000	1.779	WAB	0.169	0.000	1.815
META (FB)	3.266	0.028	24.357	MNST	0.514	0.002	2.664	DVA	0.137	0.000	0.955
INTC	4.372	0.097	26.961	PHM	0.405	0.005	2.270	PKI	0.090	0.000	0.517
BAC	4.478	0.059	24.393								
AAL	4.507	0.126	62.208								
GOOGL	0.824	0.003	16.109								
MSFT	4.723	0.068	17.401								
CMCSA	2.942	0.056	11.695								

4 Empirical Analysis

4.1 The Data

The data come from the LOBSTER database which provides limit order book data for US stocks (Huang and Polak, 2011). This is a Level 3 dataset, meaning that it contains all limit orders and cancellations for the first 10 levels of the order book as well as trades, all in a sequential order.

Our sample period is 1/Mar./2019 to 28/Feb./2023. We consider 35 stocks constituents of the S&P500, sampled on the basis of liquidity considerations. In particular, we consider three groups: most active, somewhat in between, and least active, but not at the very bottom, to avoid data issues with possible missing dates or too little activity. We report them in Table 7. We have considered a disproportional larger number of liquid stocks as these tend to attract more attention and executing trades on them tends to be easier.

4.2 Results

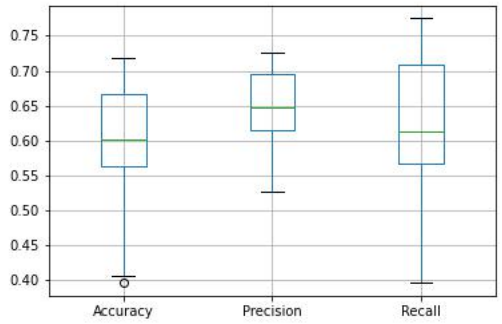
We report results based on the metrics and explanatory variables discussed in Sections 3.5 and 3.6.

4.2.1 Intraday Prediction

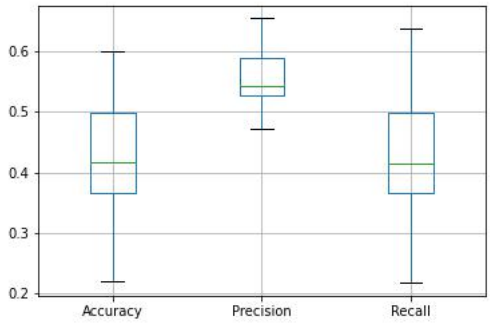
We analyse intraday predictability using various training algorithms, as discussed in Section 3.4. Our predictions are admissible in the sense that we use 6 months rolling window to train the models. In particular, at the end of a month, we use the last six months of data, train the model and predict with this model for the whole of next month. We repeat this process every month.

We present the results using different methods and levels of aggregation $a \in \{0.001, 0.005, 0.01\}$. The statistical performance measures consistently show that there is predictability in the sign of the average price change at all frequency levels. Specifically, algorithmic predictability can be as

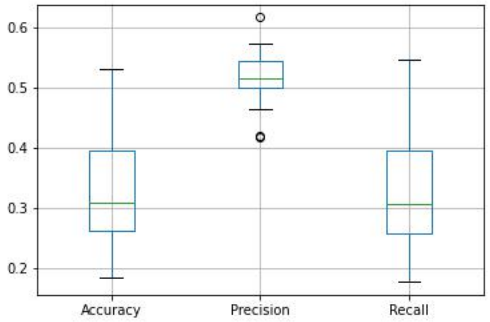
Figure 3: Boxplot of Statistical Performance Measures. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively.



(a)



(b)



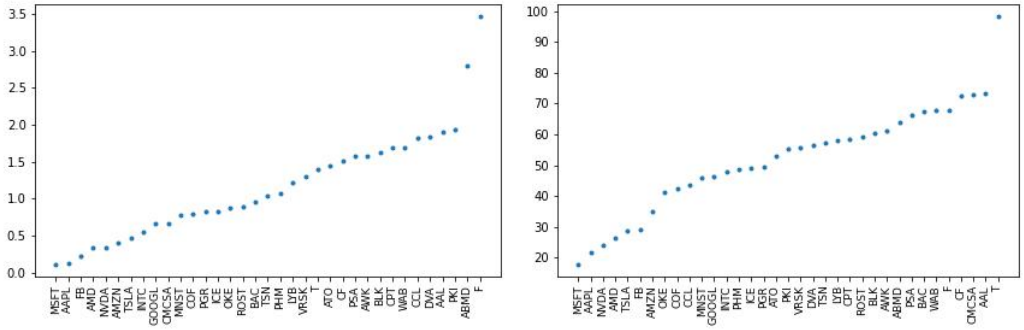
(c)

high as 73%. However, as we decrease the frequency (in volume time), the predictability decreases, as expected. Among the classification metrics, precision is the least affected by changing frequency of aggregation. Precision measures the probability that, conditioning on the predicted direction being $y \in \{-1, 1\}$, the actual price direction turns out to be the same (see Section 3.5.1 for details). Figure 3 shows the results for all 35 stocks summarised in a boxplot.

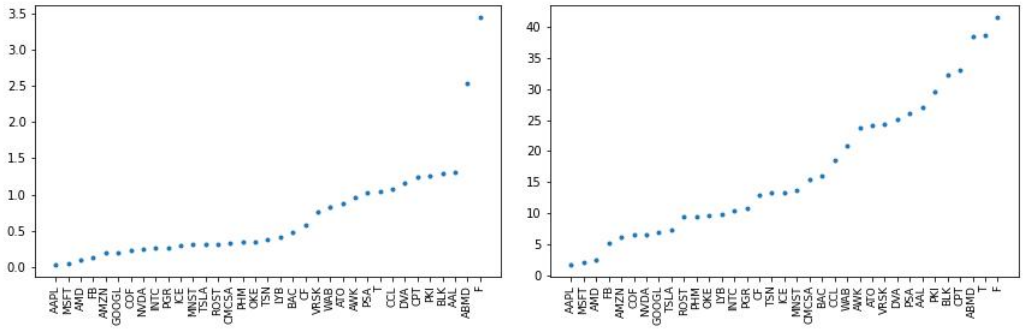
Next, we investigate the economic value of the classification following the methodology discussed in Section 3.5.2. We compute the average return per time period as in (4). The results confirm that there is economic value in the predictions made by the classifiers. However, this decreases as we lower the frequency of aggregation. Most returns per time period are in the order of 1 basis point. The spread for all these stocks is in the order of single digits. Hence, this predictability is compensated by liquidity providers keeping the spread wide enough to protect against adverse selection. However, the predictability at these frequencies is useful for inventory management of liquidity provides and execution algorithms. Figure 4 provides a visual summary of the aforementioned remarks.

Average sample results can mask interesting dynamics regarding predictability. For this reason,

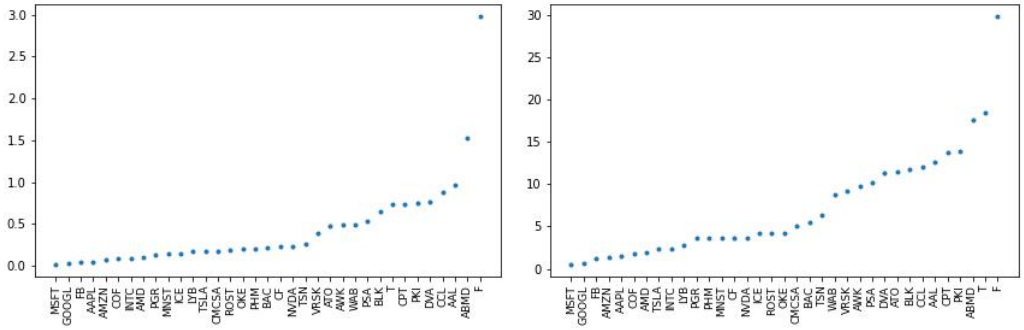
Figure 4: Sorted Average Returns per Period and T-Statistics. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. Within each panel, the left hand side plot is for returns in basis points and the right hand side plot is for t-statistics. For each stock, the average return per time period in (4) is computed and averaged across all days in the sample. For the same quantity, the t-statistic is also computed across all days in the sample. Both results are sorted before plotting.



(a)



(b)



(c)

we also investigated the cumulated return defined in (3) averaged across stocks for each day. We can think of this quantity as the daily return on an equally weighted portfolio that can buy and sell following the predictions based on the order flow, gross of transaction costs. When predicting at higher frequencies, we take more decisions and these can lead to higher daily returns in the order of mid single digit percentage points. During the Covid19 pandemic period in March 2020, the average return were as high as 40% daily. Of course, this is gross of transaction costs. During the pandemic, the average daily volume was considerably higher than during normal periods. Given that we measure time in volume, in terms of a fixed number of shares, we sum over more time periods $n_{d,s}^{(a)}$ (as in Sections 3.1 and 3.2) when constructing the daily return in (3). However, accounting for longer daily time periods in volume time during the pandemic does not explain the higher level of informativeness in the order flow. In fact, volumes have been relatively high also towards the end of our sample. However the daily returns have not gone up. We conclude that during the pandemic market participants were less able (or willing) to conceal their information when trading, possibly due to panic.

Finally, we note that the performance cannot be explained by just a few stocks. In fact, even removing the top 5 stocks in terms of predictive performance, the results are not substantially different. Figure 5 gives a full picture of the dynamics of the information content in the order flow.

4.2.2 The Value of Book and Trades in Price Discovery

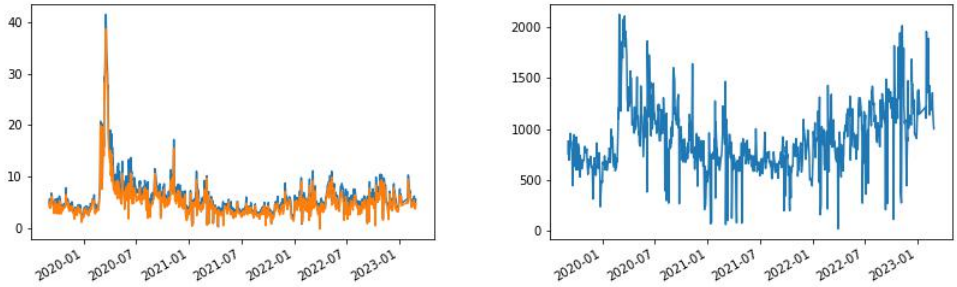
We follow the procedure outlined in Section 3.5.3 and compute (5) to assess the value of different features. In particular, we focus on the average return per period (4) as performance measure in (5). In this case, the features importance score can be interpreted as the drop in return when we omit a specific feature or set of features. We note that by construction, (5) attains the same value if we instead use the daily return (3). This is because, for each day, (3) is a scale factor of (4).

We find that the trade order flow contributes overwhelmingly more than the book order flow to the value of the prediction. This contrasts the results when using non-aggregated order book information as discussed in Section 1.1. The results show that trade flow is very important at all aggregation levels. The returns drop significantly when we omit trade flow. Specifically, the average returns per period (4) decrease by as much as 100%, 78% and 55% (median of the empirical distributions) at time scales of 0.1%, 0.5% and 1% of average daily volume, respectively. This highlights the importance of information derived from the trade flow in the process of price discovery. On the other hand, book order flow features have a modest influence on price discovery, though a positive one. These remarks are consistent with the regression analysis in Section 1.1. The Figure 6 illustrate the above remarks across all the stocks.

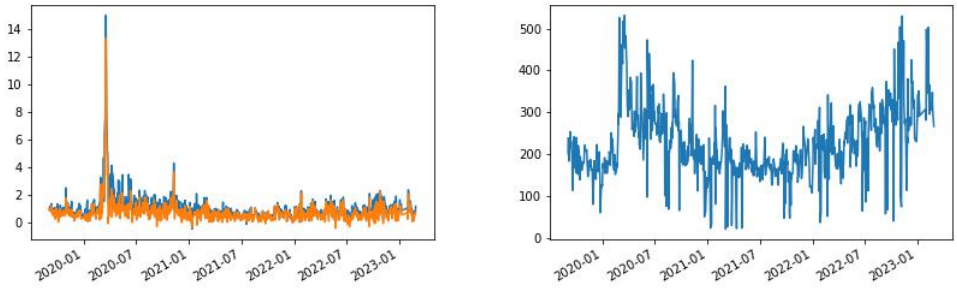
4.2.3 The Value of Speed

When predicting the next period price direction, it is of interest to measure whether the prediction and hence the information dissipates quickly and its value reduces. This can be the result of market participants observing the same information and reacting to it. Then, participants with a

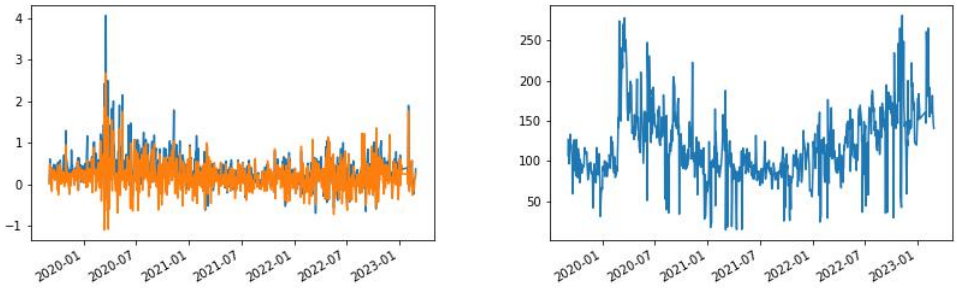
Figure 5: Time Series of Daily Returns and Daily Time Periods Averaged Across Stocks. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. Within each panel, the left hand side plot is for daily returns (3) in percentage points. The daily return is averaged on each day across stocks $s \in \mathcal{S}$ and reported as a blue line. The average removing the 5 stocks with best overall sample performance is reported with an orange line. The right hand side plot is for the number of periods $n_{d,s}^{(a)}$ in each day (as in Section 3.1) averaged across stocks $s \in \mathcal{S}$.



(a)

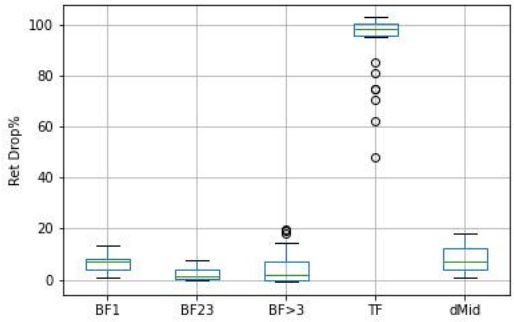


(b)

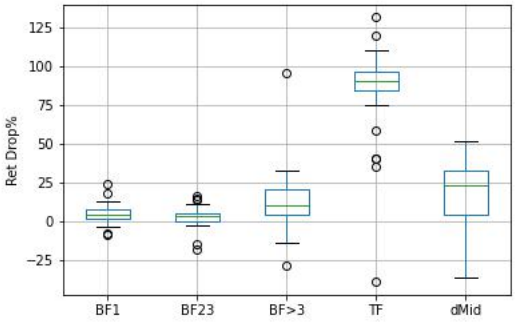


(c)

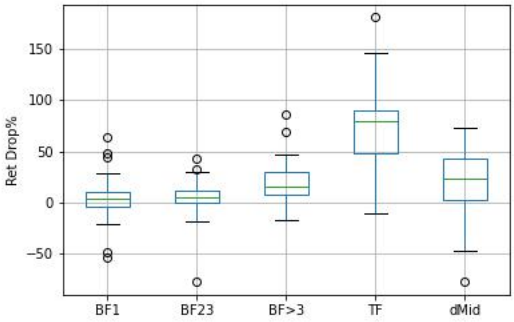
Figure 6: Features Importance. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. The boxplots are computed for the model Avg (see Section 3.4) using the features importance score in (5). The plots report the relative drop in return when the data for the plotted feature is randomly permuted. Higher values mean that the feature is more important. For BF23 and BF>3 we consider random permutation of the second and third level and the levels greater than 3 of the book order flow.



(a)

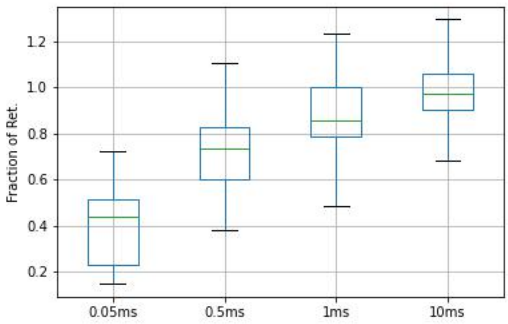


(b)

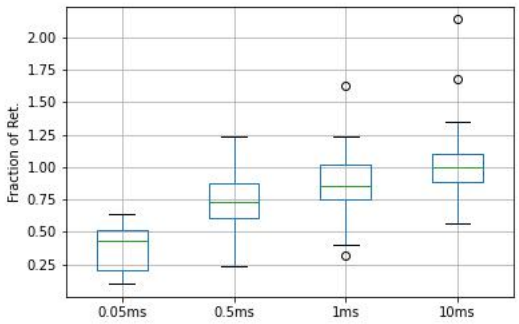


(c)

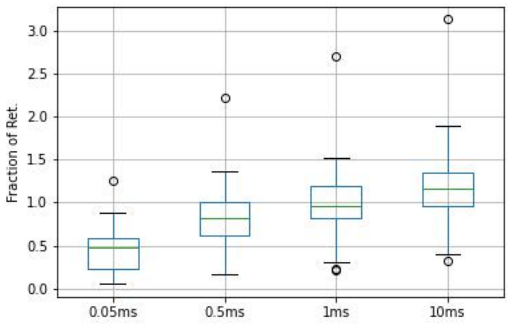
Figure 7: The Value of Speed Advantage. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. The plots show how much the price moves in the direction of the prediction in the first $\Delta \in \{0.05, 0.5, 1, 10\}$ milliseconds. Results are computed as in (7).



(a)



(b)



(c)

speed advantage will react faster and move the price accordingly. We compute how much the price responds to a prediction in the following $\Delta \in \{50, 500, 1000, 10000\}$ microseconds. Specifically, in Figure 7 we report the $\text{RelCostDelay}_d(\Delta)$ in (7). Panels (a), (b) and (c) report the corresponding estimates for the different volume time scales. We find that the information dissipates relatively quickly. Essentially, 100% of the price movement happens within the first 10 milliseconds. On average about 75% of the price movement happens within the first 500 microseconds. These are speeds that are only achieved by high frequency trading firms. This confirms that software, hardware and infrastructure are essential for maximizing returns in high frequency trading.

4.3 Relative Activity and Value of Information

The trading of stocks in the US is highly fragmented. The appearance of off exchange facilities exacerbates this further. However, the NASDAQ, NYSE and CBOE are the exchange venues with

higher market share. A recent post of the CBOE provides a close look at US trade venues and the current state of affairs[14].

Results of this paper are for NASDAQ listed stocks and are based on NASDAQ TotalView data feed. In view of this, it is natural to ask if the NASDAQ order flow of stocks that are more active on the NASDAQ, relative to other venues, may convey more information. More generally if a stock is relatively more active in exchange e, is the order flow from that venue more informative? By more active we do not mean traded volume, but actual activity, in terms of messages sent to the exchange. We then see whether the economic value of NASDAQ order flow (as in (4)) is monotonically increasing for with respect to stocks that are more active on the NASDAQ. In particular, we count the messages sent by five exchanges that cover most of the trading activity [15]. Our choice is to cover most of the US exchanges. Due to off exchange trading, activity can also go through other less regulated venues. For these exchanges, we count the number of messages sent. For all but NYSE and NYSE ARCA (ARCA for short), messages are recorded for the first 30 levels. The feeds for NYSE and ARCA send all the the messages. To construct our index of NASDAQ activity for each stock s, we compute the following:

$$\text{RelCount}\,(s) = \frac{\text{Count}\,(s, \text{NASDAQ})}{\left(\prod_{e\in\mathcal{E}} \text{Count}\,(s,e)\right)^{1/4}} \tag{8}$$

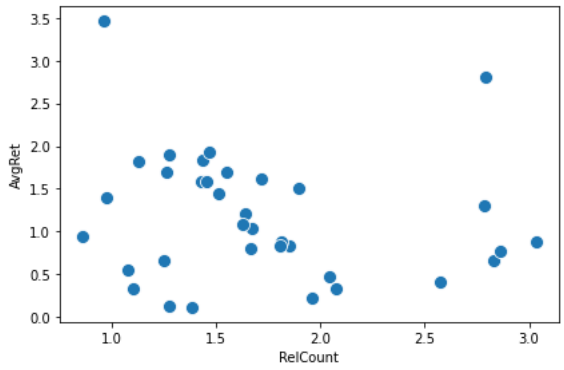
where $\mathcal{E} = \{\text{NASDAQ}, \text{NYSE}, \text{ARCA}, \text{BZX}, \text{BYX}\}$ and $\text{Count}\,(s,e)$ is the number of messages sent for stock s to exchange e. Due to the fact that the NYSE and ARCA feeds records all the messages, we are making the following assumption in the construction of (8). We suppose that for NYSE the count of the full book is $(1 + c_{\text{NYSE}})$ times the count of the first 30 levels, for some constant $c_{\text{NYSE}} > 0$ that does not depend on the stock s. A similar assumption is made for ARCA with possibly different constant c_{ARCA}. Then, the index in (8) is just $[(1 + c_{\text{NYSE}})\,(1 + c_{\text{ARCA}})]^{-1/4}$ the index based on the first 30 levels of the NYSE and ARCA, in the denominator of (8). Given that what matters is the relative size across stocks, this has no impact in the following.

To verify if there is a monotonically increasing relation, we produce a scatter plot of (4) and (8) for different volume time scales a. In particular, for each stock, we compute the time average of (4) and (8). Our results show that there is no monotonic relation between the two (Figure 8). We can then conclude that relatively higher market activity on the NASDAQ for a given stock does not translate to higher information content of the order flow. This is an interesting result, as it indicates that market participants may choose to quote more heavily on the NASDAQ, but they do so without leaking more information.

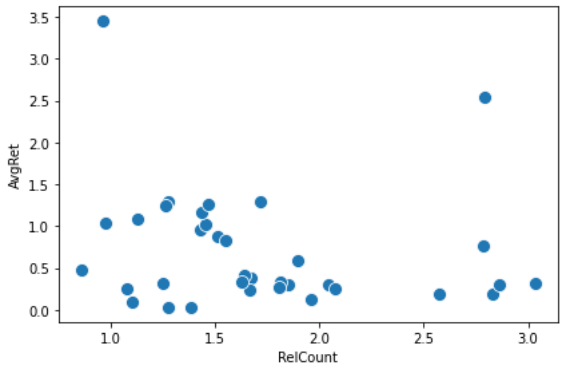
[14] U.S. Equities Trading Venues: A Closer Look, August 10, 2023. `https://www.cboe.com/insights/posts/u-s-equities-trading-venues-a-closer-look/`

[15] The count is for level 3 (MBP – Market By Price) order books. The message counts come from the following five sources: (1) NTV_MBP (NASDAQ TotalView) (2) NY2_MBP (traditional NYSE) (3) ARC_MBP (NYSE ARCA), (4) BYX_MBP (one of CBOE books) (5) BZX_MBP (one of CBOE books). The New York Stock Exchange (NYSE) is a physical and electronic stock exchange, while NYSE ARCA is an electronic communications network (ECN) used for matching orders. The CBOE books are from BATS exchanges. BATS was acquired by the CBOE in 2017

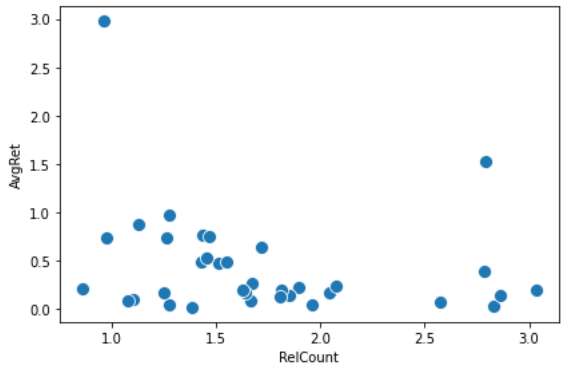
Figure 8: Scatter Plot of Daily Averages of Relative Message Count and Average Return per Period. Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. The average return per period in (4) is for the model Avg (see Section 3.4) and averaged across days, over the full sample for each individual stock. The relative message count in (8) is also averaged across days for each individual stock.



(a)



(b)



(c)

4.3.1 Explaining the Predictions

Finally, we explain the performance of the order book predictions using the covariates in Section 3.6.1. Consider a fixed level of aggregation parameter a so that we can drop the dependence on it. Let $\tilde{Y}_{d,s,m}$ be the average return in (4). To begin with, we pool all the data from the different models in a panel data form in order to understand how prediction methods, stock specific and market specific factors explain predictability. Let $\delta_k(l) = 1$ if $k = l$ and zero otherwise and $\mathcal{M} := \{\text{LDA}, \text{Ridge}, \text{RF}, \text{DNN}, \text{Avg}\}$ with short names as in Table 4. We consider the following specification,

$$\tilde{Y}_{d,s,m} = \sum_{l \in \mathcal{S}} \alpha_l \delta_l(s) + \sum_{k \in \mathcal{M} \setminus \text{Avg}} \gamma_k \delta_k(m) + \sum_{j=1}^{3} \rho_j \tilde{Y}_{d-j,s,m} + X'_{s,d}\beta + \varepsilon_{d,s,m} \tag{9}$$

where the parameters α_l capture the unobserved stock specific effects, and γ_k the model specific effects. Here, we choose the Avg method to represent our benchmark, and thus, we drop the dummy for it. Hence, a positive coefficient γ_k would imply that the specific prediction model performs better than the average of the predictions. We are interested in understanding whether training more complex and sophisticated methods can enhance predictability. We also add three lags of dependent variable $\tilde{Y}_{d,s,m}$, and the vector valued covariate $X_{s,d}$ of stock and market explanatory variables as defined in Section 3.6.1. The term $\varepsilon_{d,s,m}$ is a zero mean error term. Here, we need to highlight why a dynamic setting is used. The variables $\tilde{Y}_{d,s,m}$ are autocorrelated across days d. For fixed s and m, when running the above regression with no lagged dependent variables, the dependence is picked up by the error term $\varepsilon_{d,s,m}$. We found that including 3 lags essentially removed the autocorrelation for every stock s and method m. Hence, by introducing model dynamics we should be confident that no bias should have been introduced[16]. Moreover, we centered all the variables except for the constant. We also capped the Beta and MarketImpact by 5 and 15, respectively. These are upper bounds for their 99% quantiles. Capping is advisable because we observed some large outliers, in some cases.

In Table 8, we report the parameter estimates for the different aggregations. In terms of model predictability, it is interesting that LDA and Ridge perform very similar to the much more complex DNN. In general, the differences between models is not critical. However, this claim has to be put into the right context. Estimation via DNN requires considerable tuning of parameters and an appropriate choice of architecture. This would require a considerable investment of resources. Our goal here is to see whether any model has an advantage conditioning on relatively simple tuning parameters choices. On the other hand, we see that RF performs worse than other methods.

Regarding the stock specific determinants, it is worth noting that higher returns are associated with higher market beta stocks. This indicates a higher degree of predictability for procyclical assets. Nevertheless, we see that this effect is significant conditionally on the time scale level of aggregation. Specifically, for $a \in \{0.005, 0.01\}$ the Beta variable becomes insignificant.

[16] In addition, we estimated the model using the cross sectional correction as proposed by Chudik and Pesaran (2015). These estimates are virtually the same, hence are not reported.

In addition, the estimates provide evidence that machine learning models tend to perform better with assets being subject to higher market impact. Specifically, the MarketImpactStock variable is positive and significant across all aggregations, suggesting that predictions using oder flow improves for stocks whose price is more sensitive to trading activity.

Additionally, among the market determinants presented in Section 3.6.1, the HighVIX variable is negative and statistically significant. This indicates that during periods characterized by negative market expectations, the value of predictions decrease significantly. This is surprising, as anecdotal evidence suggests that higher volatility is associated with higher profitability when trading at high frequency. However, here we need to highlight that the HighVIX indicator is positively correlated with MarketImpactStock and HighVolumeRegime, suggesting that the high VIX regime could indirectly impact AvgRet thought the volume and volatility channel. Nevertheless, as the volume time scale increases ($a \in \{0.005, 0.01\}$), this negative effect vanishes while both MarketImpactStock and HighVolumeRegime remains positive and significant.

The coefficients of FOMC and MACRO which captures monetary and macroeconomic announcements, respectively, do not provide a clear cut interpretation. Specifically, the sign and the statistical significance seem to vary depending on the volume time scale. In regard to the year effects, the regression analysis reveal a reduction in predictability over the years. Assuming that algorithmic trading has become more widespread over the years, we can infer that higher market competition has resulted in a reduction in the economic value of predictions.

For all the different aggregation methods, the estimated autoregressive parameters are positive and statistically significant. This indicates that the generated economic value exhibit a persistent behaviour over time. This highlights the fact that market conditions impact the future economic value of predictions and their cumulative effect could last up to three trading days[17].

Given that the above regression analysis includes very long panels, we extend our analysis by accounting for further heterogeneity in the data. For each individual stock $s \in \mathcal{S}$, we re-estimate the model in (9). Dropping the stock specific index s, we consider the model:

$$\tilde{Y}_{d,m} = \alpha + \sum_{k \in \mathcal{M} \setminus \text{Avg}} \gamma_k \delta_k (m) + \sum_{j=1}^{3} \rho_j \tilde{Y}_{d-j,m} + X_d' \beta + \varepsilon_{d,m} \tag{10}$$

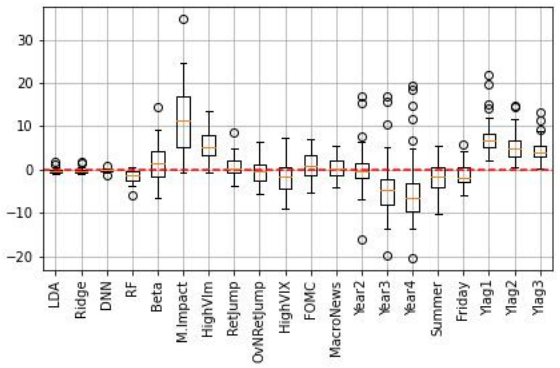
where α is a constant and all other variables are as in (9). The model in (10) enables us to analyze and exploit further the heterogeneous impact of the stock and market variables on returns. For each stock s, we estimate the model using OLS and we obtain the parameter estimates and the corresponding t-statistics. In Figure 9, we visualize the t-statistics for the above regression using a boxplot for each parameter in (10). Visualization of the t-statistics captures both the sign and the statistical significance of each parameter. Overall, the regression results illustrated in Figure (9) confirm our estimates using the panel setting, as both the parameter signs and the corresponding

[17]For any dynamic AR(p) model, the half-life of adjustment can be estimated as HL $:= \frac{ln(0.5)}{ln(|\lambda_{max}|)}$, where λ_{max} is the largest root of the characteristic function. The time needed for full adjustment can be calculated as 2HL. For the regression analysis with $a = 0.1\%$, the half-life is estimated to be approximately 1.2 trading days.

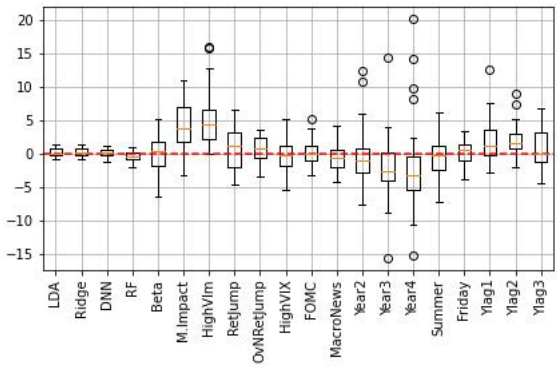
Table 8: Panel Regression Results. The results for the regression in (9) are presented for different levels of aggregation. The column est. and t-stat stand for the parameters estimates and t-statistics. The row Avg. Stocks Effects reports the mean of the fixed effects parameters' estimates and t-statistics. All regressions include stock dummies. We do not report the estimates to save space.

	Aggregation as % of Expected Average Daily Volume					
	0.1%		0.5%		1%	
	est.	t-stat	est.	t-stat	est.	t-stat
Avg. Stocks Effects	1.749	128.75	0.539	31.26	0.222	12.62
Method Effects						
LDA	-0.001	-0.20	0.010	1.34	0.017	2.27
Ridge	-0.001	-0.20	0.011	1.47	0.021	2.76
DNN	-0.005	-0.76	0.005	0.61	0.015	1.98
RF	-0.043	-7.11	-0.011	-1.46	-0.018	-2.37
Ticker Specific Determinats						
Beta	0.099	9.83	0.008	0.67	0.002	0.17
MarketImpactStock	0.249	69.32	0.115	28.82	0.045	11.88
HighVolumeRegime	0.192	38.65	0.197	33.37	0.091	15.28
RetJumpStock	-0.003	-0.28	0.041	3.84	0.092	7.79
OvNRetJumpStock	0.007	0.57	0.015	1.03	0.000	-0.03
Market Determinants						
HighVIX	-0.085	-11.39	-0.047	-4.98	-0.006	-0.67
FOMC	0.079	8.35	0.021	1.74	0.011	0.80
MACRO	0.002	0.53	-0.015	-2.91	-0.015	-2.69
Other Determinants						
Year2	0.012	1.83	-0.003	-0.38	-0.017	-2.16
Year3	-0.152	-27.08	-0.077	-10.86	-0.076	-10.51
Year4	-0.111	-17.49	-0.076	-9.80	-0.070	-8.61
Summer	-0.031	-7.47	-0.020	-3.64	-0.016	-2.59
Friday	-0.023	-4.33	0.007	1.10	-0.005	-0.82
Autoregressive Parameters						
AvgRet_{-1}	0.270	34.63	0.068	25.33	0.018	10.97
AvgRet_{-2}	0.194	29.08	0.071	28.53	0.020	11.82
AvgRet_{-3}	0.163	28.75	0.051	19.96	0.010	6.19
Adj. R^2	0.57		0.19		0.051	

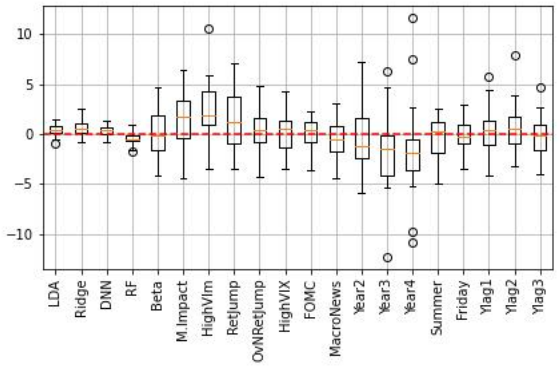
Figure 9: T-Statistics from Regression in (10). Panels (a), (b) and (c) correspond to results for a time scale of 0.1%, 0.5% and 1% of the expected average daily volume, respectively. For each individual regression in (10), the t-statistic of each explanatory variable is computed and then the boxplot is used to plot them.



(a)



(b)



(c)

statistical significance match the results using pooled data.

5 Conclusion

This paper studies the information content extracted from book and trade order flow and its implications for price predictability. Using aggregated data at different volume time scales, we analyse information spillovers and their influence on price predictability. Using Level 3 order book data for 35 constituents of the S&P500 stocks spanning from 1/Mar./2019 to 28/Feb./2023, we document that trade order flow contains persistent information, enabling predictability of mid price direction. However, we see that the information advantage vanishes very fast, as the mid price moves towards the predicted direction within the first 10 milliseconds.

We train various machine-learning based models and explore whether more computationally intensive specifications can outperform simpler ones in predicting price movement. Our empirical

findings indicate that simple models to tune such as Linear Discriminant Analysis and Ridge Classifier demonstrate comparable effectiveness to more computational intensive counterparts like Deep Neural Networks and Random Forests. This underscores the importance of the information used for model training rather than the computational complexity of the model.

Furthermore, the paper examines the economic value of prediction at different volume time scales. We find that as we increase the frequency of decisions, the trained models generate higher economic value over the period of our study. Specifically, for all the different methods, we illustrate that for predictions at 0.1% of average volume time scale, the value of prediction is three times higher than at the 1% time scale.

Additionally, we decompose the economic value of the predictions using stock and market specific factors. Overall, the profitability of high frequency predictions are persistent and its dynamic adjustment process requires up to three trading days. Moreover, the economic values are positively associated with procyclical and high liquidity assets. On the other hand, high market risk, as captured by the VIX index, unexpectedly reduces the values of order flow information. However, this effect dissipates as we increase the volume time scales. Finally, the economic value of these predictions has decreased during the years. This is likely due to an increase in trading algorithms that has led to higher market competition and efficiency.

References

[1] Aït-Sahalia, Y., J. Fan, L. Xue and Y. Zhou (2022) How and When are High-Frequency Stock Returns Predictable? https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4095405

[2] Chudik A., Pesaran M. (2015) Common Correlated Effects Estimation of Heterogeneous Dynamic Panel Data models with Weakly Exogenous Regressors. Journal of Econometrics, Vol 188, Issue 2, 393-420

[3] Cont, R., A. Kukanov and S. Stoikov (2014) The Price Impact of Order Book Events. Journal of Financial Econometrics 12, 47-88.

[4] Evans, M.D.D. and R.K. Lyons (2001) Order Flow and Exchange Rate Dynamics. Journal of Political Economy 110, 170-180.

[5] Grinold, R.C. and R.N. Kahn (1999) Active Portfolio Management : A Quantative Approach for Producing Superior Returns and Selecting Superior Money Managers. New York: McGraw-Hill.

[6] Grossman, S.-J. and J.E. Stiglitz (1980) On the Impossibility of Informationally Efficient Markets. American Economic Review 70, 393-408.

[7] Huang, R. and T. Polak (2011) LOBSTER: The Limit Order Book Reconstructor. School of Business and Economics, Humboldt Universität zu Berlin, Techenical Report.

[8] Kurov, A., A. Sancetta, G. Strasser and M.H. Wolfe (2019) Price Drift Before U.S. Macroeconomic News: Private Information about Public Announcements? Journal of Financial and Quantitative Analysis 54, 449-479.

[9] Lucchese, L., M. Pakkanen and A. Veraart (2023) The Short-Term Predictability of Returns in Order Book Markets: A Deep Learning Perspective. https://arxiv.org/abs/2211.13777.

[10] Kercheval, A.N., Y. Zhang (2015) Modelling High-Frequency Limit Order Book Dynamics with Support Vector Machines. Quantitative Finance 15, 1-15.

[11] Kyle, A. (1985) Continuous Auctions and Insider Trading. Econometrica 53, 1315–1335.

[12] MacKenzie, D. (2017) A Material Political Economy: Automated Trading Desk and Price Prediction in High-Frequency Trading. Social Studies of Science 47, 172-194 .

[13] Mucciante, L. and A. Sancetta (2022) Estimation of a High Dimensional Counting Process Without Penalty for High Frequency Events. Econometric Theory: https://doi.org/10.1017/S0266466622000238.

[14] Mucciante, L. and A. Sancetta (2023) Estimation of an Order Book Dependent Hawkes Process for Large Datasets. Journal of Financial Econometrics: https://doi.org/10.1093/jjfinec/nbad021.

[15] Sancetta, A. (2018) Estimation for the Prediction of Point Processes with Many Covariates. Econometric Theory 34, 598-627.89-107.

[16] Sancetta, A. (2023) Intraday Trades Profile Estimation: An Intensity Approach. Journal of Financial Econometrics 21, 651-677.

[17] Zhang, Z., S. Zohren and S. Roberts (2019) DeepLOB: Deep Convolutional Neural Networks for Limit Order Books. IEEE Transactions on Signal Processing 67, 3001-3012.