École Polytechnique Fédérale de Lausanne

Financial Big Data Project

Real-time intraday bubbles trading

FIN-525, Fall 2024

Guillaume Ferrer (311582)
Gustave Paul Besacier (376507)
Edouard Bueche (319539)

January 26, 2025

Contents

Abstract

This report explores high-frequency financial big data, including order book and trading quotes for 87 stocks over five years, to design trading strategies targeting intra-day bubbles. These bubbles are identified using three criteria: price, volatility, and volume. The study aims to develop strategies to detect and exploit market inefficiencies to generate returns. Additionally, various hyper-parameters are tested to identify distinct market regimes. We eventually find changes in strategy performance over time, supporting the presence of regimes. We also find that volume-based strategies tend to provide better results.
Keywords: High-frequency data, financial bubble, intraday data, volume trading.

I Introduction

Empirical financial data often exhibit patterns refuting the Efficient Market Hypothesis, namely asset prices temporarily deviating from their fair value. Looking at very small time intervals, we observe small deviations from longer run trends, that we define as bubble. This creates trading opportunities that can be exploited using quantitative strategies. In high-frequency trading (HFT), where trades are executed in milliseconds, the ability to identify intraday bubbles—brief but significant price swings—is especially important. These bubbles are typically triggered by abrupt market changes, liquidity imbalances, or large order executions, and detecting them in real-time can offer both theoretical insights and positive returns.

This project focuses on detecting and trading these intraday bubbles using high-frequency second-level data. The challenge lies in distinguishing between random price fluctuations and genuine bubbles, which requires the application of statistical methods and data-driven strategies. We aim to develop a framework that identifies price anomalies based on three key dimensions:

1. `Price`: Sudden price movements that deviate significantly from trends.
2. `Volatility`: Spikes in price fluctuation, indicative of uncertainty or market stress.
3. `Volume`: Abnormal trading activity, often preceding or accompanying price movements.

By analyzing these factors, we seek to create strategies that can effectively capture these sudden market moves in volatile market conditions. We want to analyze how these strategies perform over time and detect whether the market shows various regimes.

II Data Analysis

A Data Acquisition

The data consists of order book (`bbo`) and trade (`trade`) record of 88 stocks of the SP100, between January 2004 and December 2008, acquired from the course Switch Drive repository[1].

The raw data contains a folder for the `bbo`, that contains a folder per ticker. Each in turns contains a compressed `.tar`, encapsulating a series of `gz`-compressed daily records, for each trading day of the 2004-2008 period. A typical file is a time series of the bid and ask price and volume (namely `xltime, bid-price, bid-volume, ask-price, ask-volume`). More details about the structure can be found in the appendix, on Figure 14.

[1] SWITCHdrive file `sp100_2004-8.tar`.

The `trade` file is organized identically, each daily file consists of a time series of trading price and volume, as well as operation categorization (`xltime, trade-price, trade-volume, trade-stringflag, traed-rawflag`).

In Figure 1 we summarize the time coverage of each stock over the full 2004-2008 period for both `bbo` (very similar for `trade`). A black cell corresponds to existing data for the specific [ticker, date] pair, and a white cell stands for the contrary. We clearly see that the data is almost complete for the whole stocks on the period.

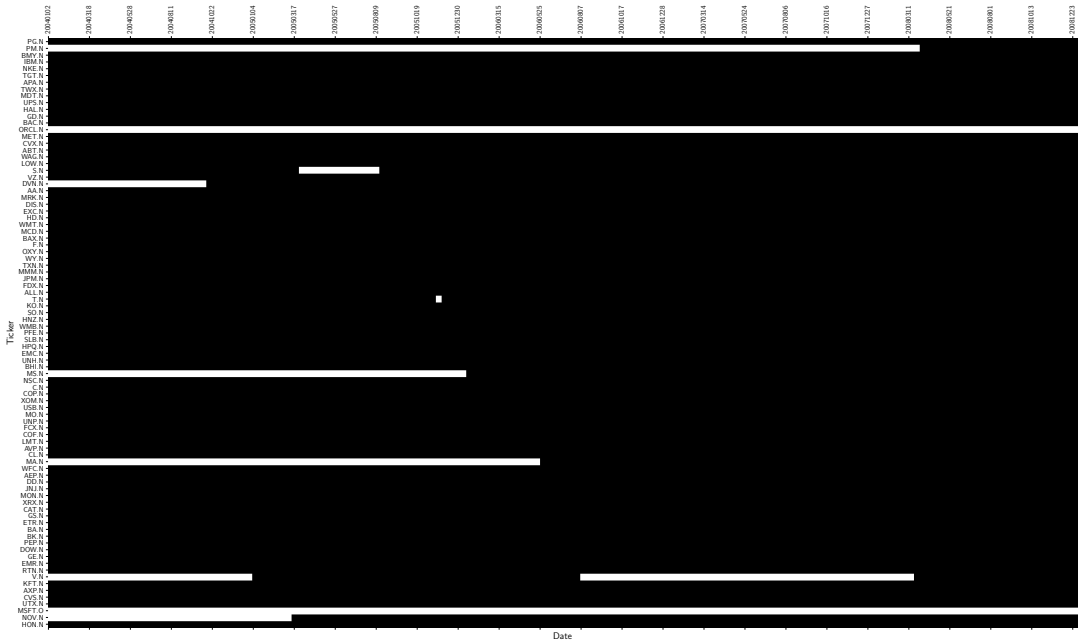


Figure 1: `bbo` coverage

B Data Analysis

B.1 Intraday Volume

To understand the volume dynamics , we analyzed the traded volume per 5-minute interval over multiple years for some stock. We computed both the mean and median traded volume to distinguish between overall liquidity trends and typical trading activity. The results, visualized in Figure 2, show that trading volume is highest at market open (9:30 AM) and close (3:55 - 4:00 PM). This pattern aligns with institutional order executions and end-of-day portfolio rebalancing. This is a well accepted stylized fact known as *liquidity smile*.

The comparison between mean and median traded volume reveals that the mean consistently exceeds the median, indicating that large institutional trades significantly impact total volume. The median, being less sensitive to outliers, highlights that most trades are relatively small in size.

Figure 2 depicts these metrics for a single stock (EXC) for which we clearly remark it follows a typical intraday volume pattern. This behavior is consistent across most stocks, where the mean traded volume forms this *smile* shape, peaking at the market open and close. In contrast, the median traded volume remains relatively flat, with small spikes at the beginning and end of the trading session, indicating that most trades are of smaller size, while a few large trans-

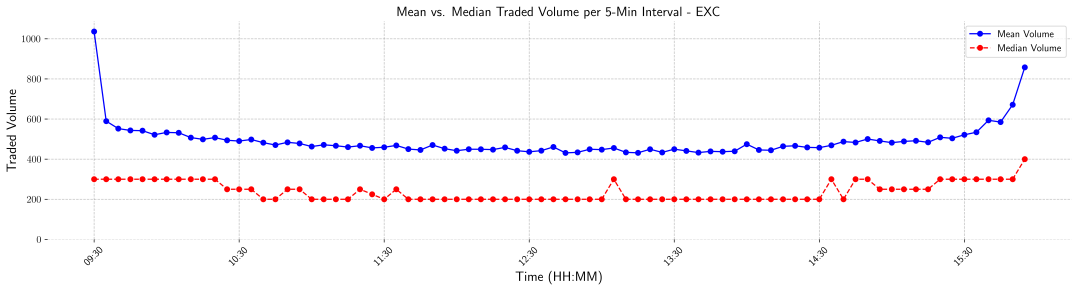


Figure 2: Mean vs. Median Traded Volume per 5-Min Interval

actions influence the mean. This graphs for more stocks are available on the project's GitHub repository[2].

B.2 Intraday Bid-Ask Spread

The intraday bid-ask spread is illustrated in Figure 3 for a stock (EXC). At the market open (9:30 AM), spread tends to be larger due to lower initial liquidity and heightened uncertainty, as market participants react to overnight news and pre-market trading activity.

As the trading session progresses, the spread gradually narrows, reaching its lowest levels around midday. This contraction is attributed to increased market participation and liquidity, as institutional investors and market makers become more active.

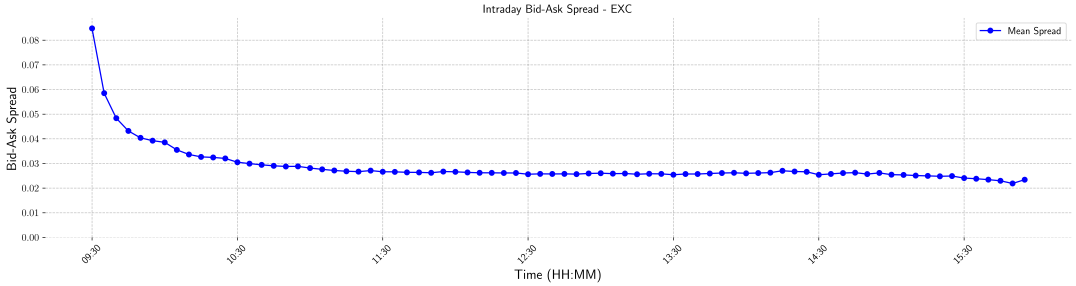


Figure 3: Mean of spread per 5-Min Interval

Once again for the spread, the results are depicted for a single stock but the trend is observed across most companies of the dataset (more graphs can be found on the GitHub repository).

III Data Preprocessing

In this section, we details the processing we applied to the raw data in order to make it usable for our task. This work can be decomposed into two parts. First, we processed and cleaned the data, then we organized it in efficient storage architecture.

[2]See the GitHub repository

A Order book and trading price conciliation

The main goal is to merge the `bbo` and `trade` files to get the following architecture:

`date`	`bid-price`	`bid-volume`	`ask-price`	`ask-volume`	`trade-price`	`trade-volume`

Table 1: File architecture type.

Individual file cleaning

In this part, we proceed to the individual cleaning of the `bbo` and `trade` files[3]. For both files, we first replaced the excel-type of the date by a standard date object, taking into account the New-York (US) timezone. Then, we filtered potential errors by removing the observations for which the ask and bid price were negative, and where the ask price was smaller than the bid price – by definition impossible. Finally, we kept only the core trading session observations, namely between 09:30:00 and 16:00:00[4].

For the order book files (`boo`), we additionally handle the case in which multiple lines exist for the same timestamp by keeping the most recent observation only.

For the trading files, we filtered the observations for `trade-stringflag` equal to `uncategorized`. This aligns with our desire to solely focus on strategies on *traditional* trading, excluding block for example. Similarly to `bbo`, we also kept a single observation per timestamp; and eventually only kept the trading volume and price information, as other columns are irrelevant for our analysis.

File merging

With cleaned `bbo` and `trade` files, we then proceed to the merging. We full merged on the date of the files (keeps all rows when there is a match in either left or right table[5]), then we grouped the data by second, so it increased the compatibility of the files, which timestamp were precise to the millisecond. We aggregated the data by averaging the prices (for bid, ask, and trade prices) and summing for the respective volumes. At this point, the data contained many missing value, hence we decided to use interpolation to fill it. We eventually get files as depicted in Table 2.

date	**bid-price**	**bid-volume**	**ask-price**	**ask-volume**	**trade-price**	**trade-volume**
2004-01-27 09:31:33	43.5	10.0	43.57	10.0	43.57	0.0
2004-01-27 09:31:35	43.56	1.0	43.57	10.0	43.57	1100.0
2004-01-27 09:31:36	43.56	1.0	43.57	15.0	43.57	0.0
2004-01-27 09:31:37	43.56	1.0	43.57	20.0	43.57	0.0

Table 2: Example of merged file (for ticker ABT on January 27th, 2004)

B Dataset architecture

With our data cleaning function built, we proceed to the storage of the data. We decided to store the merged dataset as a single file per month (as in Table 2), stored in annual folders,

[3]Note that the data processing is closely inspired by the course material, available on the course webpage.
[4]NYSE market Holidays and Trading hours, nyse.com.
[5]Polars documentation.

for each ticker. The precise architecture of the clean dataset can be found in Figure 15 of the appendix. In order to get an efficient process, we levered the `polars` library and the `polars.LazyFrame`[6] objects that only performs the operations at once, after optimizing the process and uses parallelization. We also used efficient concatenation (to concatenate the daily to monthly dataframes) using parallelization as well.

The final monthly data is stored in `csv` format which allows for fast and easy retrieval. More details can be found in Table 15.

IV Strategies

As mentioned earlier in Part I, bubbles are characterized by the dynamics of volume, price, and volatility. Therefore, we aim at detecting these bubble using strategies that detect regime shifts in volume, volatility, and price using rolling window computations. As stated in the course, "STRATEGY = SCIENTIFIC MODEL = MEASURING DEVICE,"[7], thus we will aim to develop strategies capable of measuring the "bubble-state". It in turns triggers trading decision exploiting significant discrepancies that emerge between short-term and longer-term metrics.

The following subsections detail the single-stock strategies employed to capture these market anomalies and their performance under various conditions.

A Price momentum

The simplest strategy that we implement is a basic price momentum of moving average. Focusing on a single stock, the idea is to compute a short- and long-window moving average of the price, and use it as a trend indicator. When the long-term moving average (`LMA`) crosses the short-term moving average (`SMA`) from below, we expect the price to increase and trigger a buy action, whereas when the `LMA` crosses the `SMA` from above, we expect the stock price to decrease and trigger a sell action. This can be visualized in Figure 4.

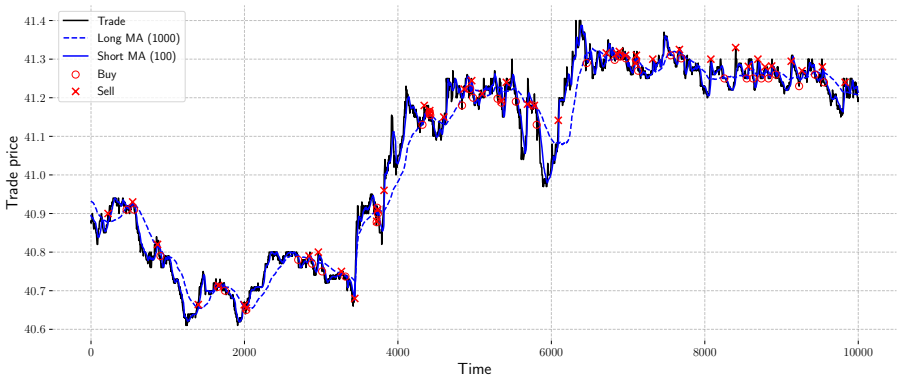


Figure 4: Example of momentum for the last 10000 observations of February 2004 for APA.

[6] Polars documentation.
[7] Week 13 of the course.

B Excess Volume Strategy

The goal of this strategy is to exploit large changes in the volume traded. Two elements need to be distinguished. On the one hand, we have a pair of volume moving averages, with a short and a long window (respectively `SMAv` and `LMAv`, `v` stands for *volume*). On the other hand, we have a pair of price moving averages with again a short and a long window (respectively `SMAp` and `LMAp`, `p` stands for *price*). The trading decision is triggered by the volume moving averages, while the trading direction is determined by the price moving averages.

More specifically, we have the following trading process. When the `SMAv` crosses the `LMAv` by above, we consider trading and we check the price moving averages. If `SMAp` is above `LMAp`, we buy (as we expect the price to increase) while we sell if `SMAp` is below `LMAp`. On top of that, we decided to include an extra condition to forbid short selling, that is we only sell if we have an open position, and we only buy if the net position is currently zero.

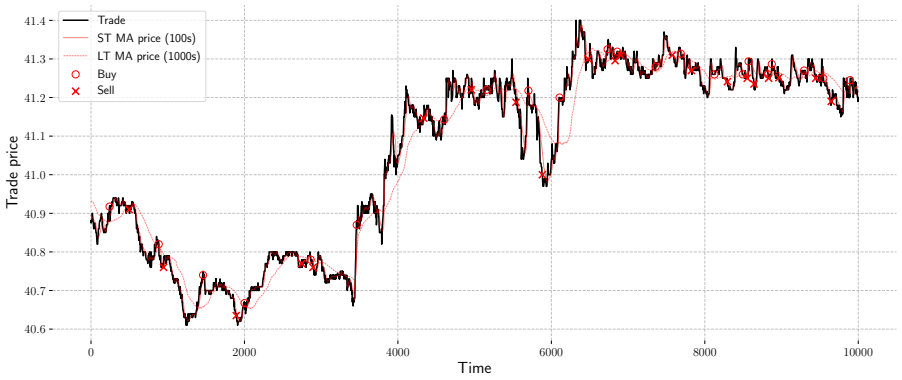


Figure 5: Example of volume based momentum strategy for the last 10000 observations of February 2004 for APA.

C Volatility-Based Strategy

In that strategy we aim at interpreting change in log return volatility as a trading signal. Indeed, it can interpreted as an important information about market competitors trading direction, and is likely to generate gains if spotted fast enough. We first started to compute the log return of the trade price. Then, we applied the same methodology as earlier and create a short and a long window rolling standard deviation (respectively $\texttt{SMA}_\sigma$ and $\texttt{LMA}_\sigma$). The trading signal is $\texttt{SMA}_\sigma$ crossing from below of $\texttt{LMA}_\sigma$ for a buy, and from above for a sell. Using this ensures we are restricted to long positions only, as desired. An example is displayed in Figure 6.

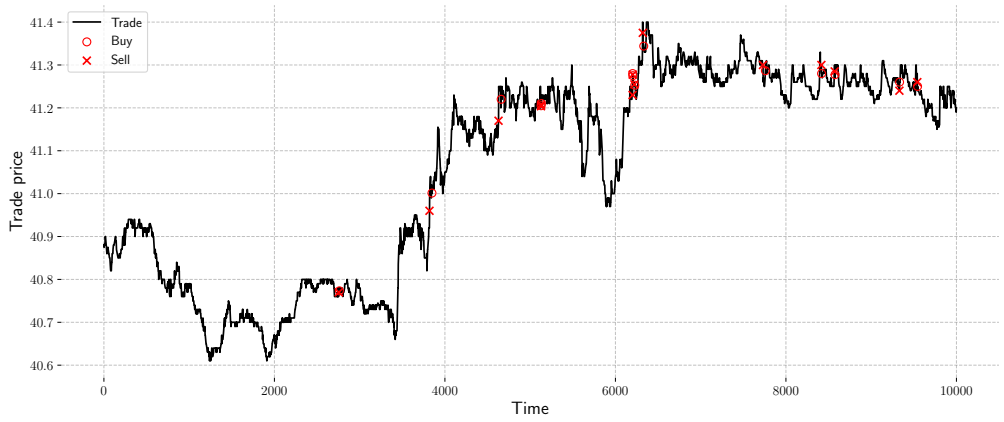


Figure 6: Example of volatility-based strategy for the last 10000 observations of February 2004 for APA.

D Strategy storage

After we built the strategies, we ran them on each stocks for the whole period, trying multiple parameters for the rolling window. To store all this data, we generated a CSV file for each strategy (a strategy is either a volume-based, price-based or volatility-based approach, for a specific set of parameters), containing the daily returns for each stock, for the whole period of time. Table 3 presents an example of a strategy file. The files are then stored in folder `data/strategies`, using the following nomenclature: `strategyFamily_parameters.csv`. For instance, a momentum-price strategy with a long moving average window of 1000 and a short moving average window of 100[8], the corresponding CSV file will be `momentum_price_s100_l1000.csv`. This structured naming convention ensures clarity and facilitates easy retrieval of strategy results for further evaluation and comparison.

day	VZ	CAT	...	PFE
2004-01-02	-0.0291	0.0426	...	-0.0223
2004-01-05	-0.0212	-0.0228	...	-0.0108
...	...	...	...	...
2004-12-31	-0.0249	-0.0396	...	0.0278

Table 3: Strategy file architecture type.

V Results

A simple way to compare the strategies is to find the best strategies (highest returns) for a particular stock on a given date are represented in the heatmap 10. The results of the strategy are summarized in the heat map of Figure 10. For each day and ticker, the color cell corresponds to the best performing strategy. The correspondence between the color id and the trading strategy can be found in the appendix, in Table 4. We observe some regime change before

[8]Note that the all the rolling windows for average and standard deviation are computed in terms of seconds (opposed to simple 'rows'). This increases the comparison power as it puts each stock on the same playing ground field (as the time delta between each observation is not necessarily constant nor identical over stocks).

and after 2007, end of year. The two separated heat maps for the corresponding periods are presented in Tables 16 and 17 of the appendix. After 2007, the price-based momentum seem to overperform other strategies for most of the stocks. This is likely to results from the financial instability slowly approaching.

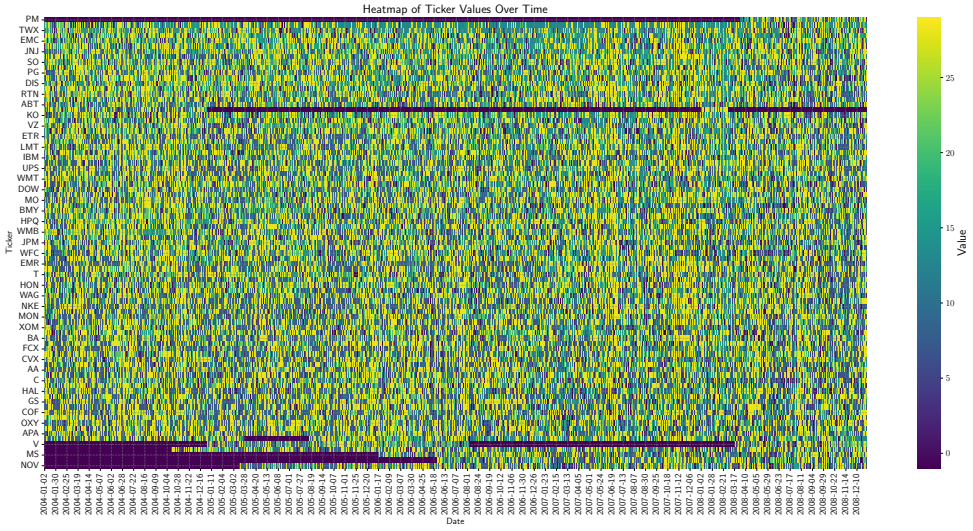


Figure 7: Best strategy over time for each ticker (values mapping in Table 4)

In Figure 8, we reduced the dimensionality of the heat map by regrouping the performance by strategy family (volume, price and volatility), to have an easier visualization of the performance over all parameters tried. It clearly exhibits vertical lines around particular dates, which may indicates a change of state in the market where now another family of strategy (maybe with different parameters) is better on average on that period.

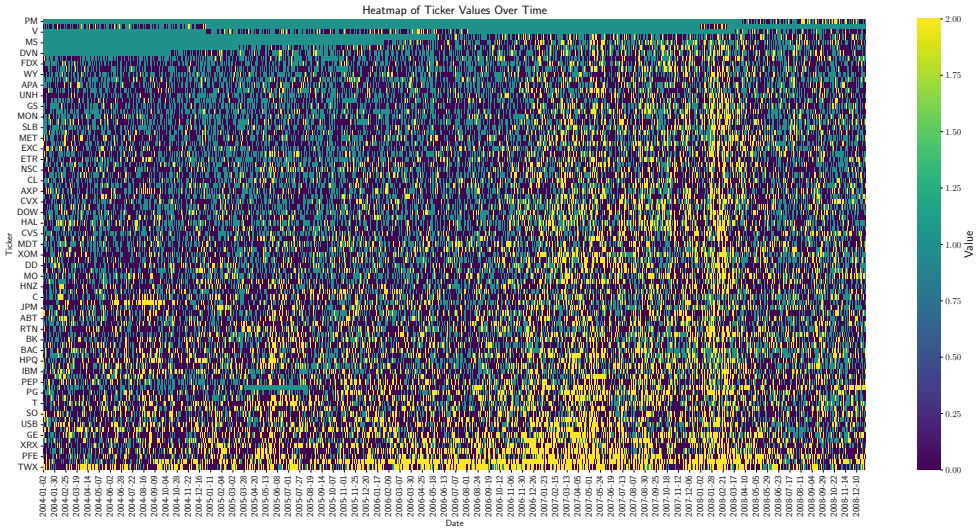


Figure 8: Best family of strategy over time for each ticker (values mapping in Table 5)

We are also interested in analyzing how the hyper-parameters inside a family change over time. Figure 9 shows the family of excess volume strategies. Again we can see clearly that there is a

change in regime after 2007 and some stocks seem to be more adapted for certain strategy.

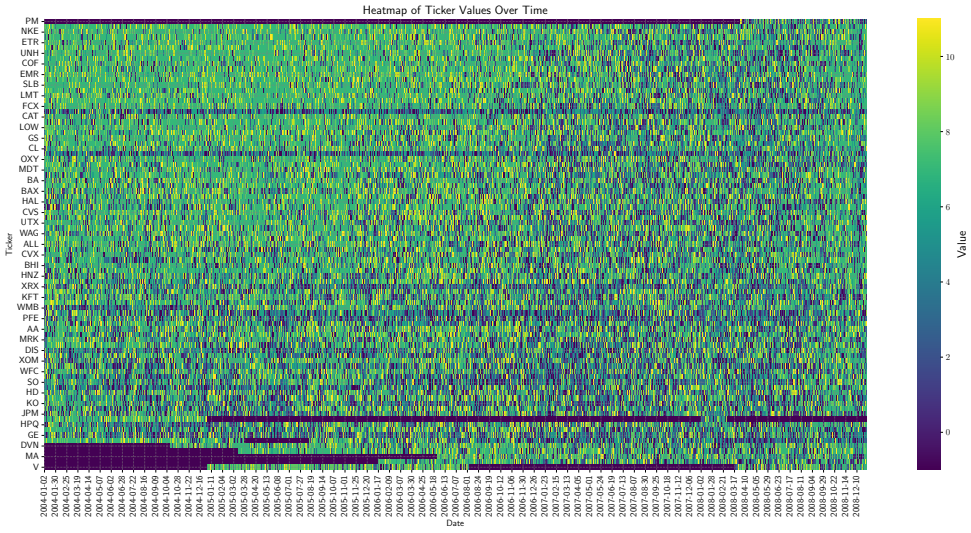


Figure 9: Best family of strategy over time for each ticker (values mapping in Table 6)

We also adopted another visualization approach in Figure 10, which illustrates the best-performing strategy over all stocks (equally weighted portfolio) on each day. This puts in evidence that some strategies outperform others significantly, and consistently over time. However, this does not necessarily indicate that they are optimal trading strategies, but only that, on specific days, they performed better than the other strategies we developed. Some strategies under perform consistently, while others exhibit strong performance under specific parameter configurations but fail under different conditions. This variability highlights the importance of parameter tuning and adaptive strategy selection in algorithmic trading.

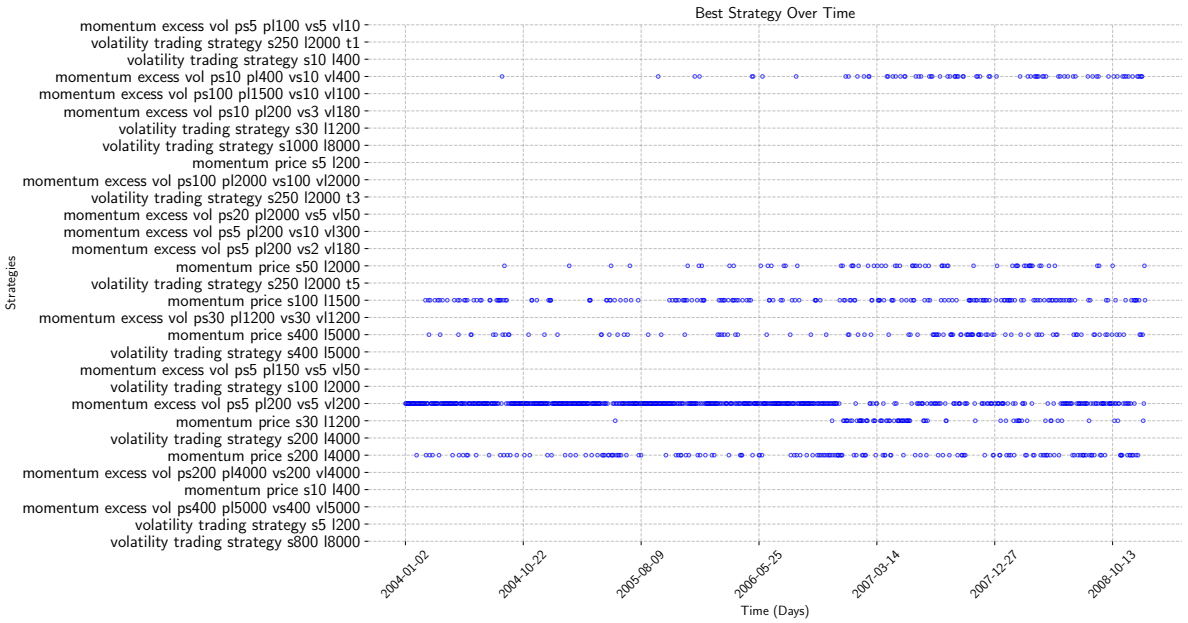
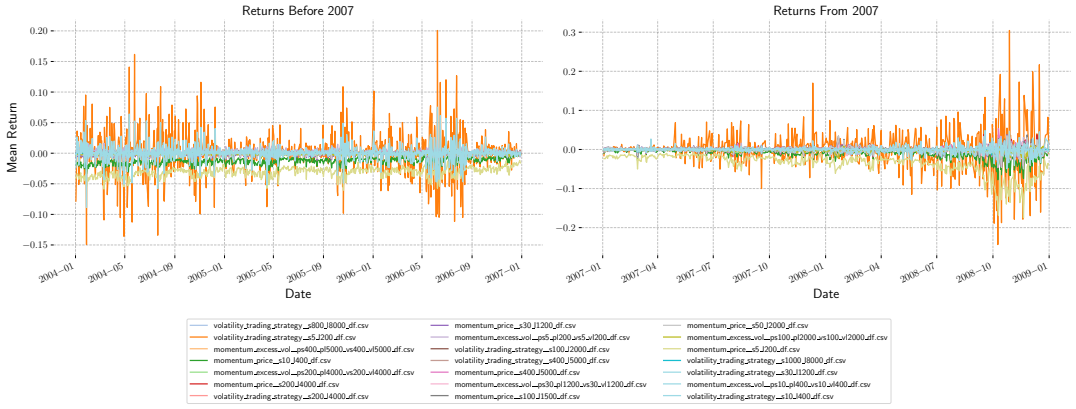
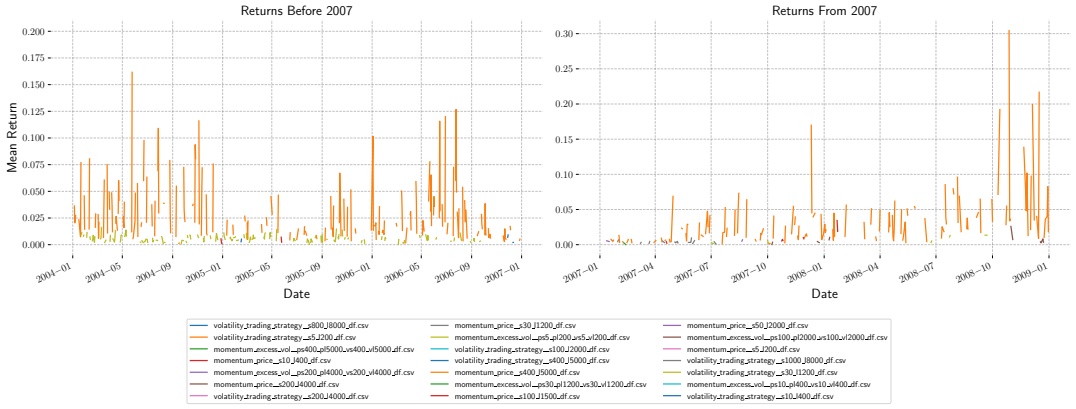


Figure 10: Best strategy as a function of time

We also plotted the daily return of each strategy over time in Figure 11a. Despite the large number of strategies plotted in the same figure, we clearly see that some consistently surpass the other. Also note a change in scale of mean returns of the graph before and after 2007, which is consistent with our previously presented results. Figure 11b presents the same information but only shows the best strategy for each day. Here again, very few strategies dominates.



(a) Daily returns per strategy.



(b) Daily returns of the best strategy.

Figure 11: Returns of the strategies over time.

Finally, to investigate how a strategy evolves with respect to a stock, we visualized the stock's network based on strategy correlations. While we displayed the network for a specific month, the code allows us to explore networks for any time period.

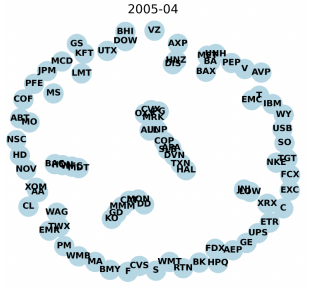


Figure 12: stock's network based on strategy correlations for March 2005 correlation level = 0.6

VI Discussion

The results presented above are encouraging as they highlight the potential of intraday market data to exploit inefficiencies and generate returns. To build upon this direction, and based on Figure 10, we could expand the range of parameters for the strategies and filter out underperforming ones. This approach could enhance their effectiveness and adaptability to various market conditions.

Another noteworthy observation is the critical role of hyperparameters. For instance, `momentum excess vol ps5 pl200 vs5 vl200` is one of the best-performing strategies overall, but reducing the window sizes to `ps5 pl150 vs5 vl50` renders the strategy ineffective. Additionally, we observe changes in market states: `momentum excess vol ps10 pl400 vs10 vl400` and `momentum price s30 l1200` appear to become effective after mid-2006. Analyzing the current state of the market could provide insights into designing more efficient strategies.

Using the heatmaps in Figures 4, 5, and 6, we clearly identify three key components. First, the distribution of strategies over time is not uniform—some strategies outperform others under specific market regimes. Second, certain strategies exhibit a preference for particular stocks. Third, the family of strategies plays a significant role, as a given family tends to favor specific stocks and regimes.

Thus, the next step would be to analyze the relationships between stocks and strategies, examine how market regimes influence stocks, and identify how strategies leverage these regime shifts.

Another area to explore is the clustering of stocks that exhibit similar changes in strategy. Figure 12 reveals that certain stocks share comparable patterns in the strategies that perform best for them. Analyzing the factors driving the formation of these clusters would be valuable. Additionally, it would be insightful to visualize a network of strategies, illustrating how they transition and change in relation to one another.

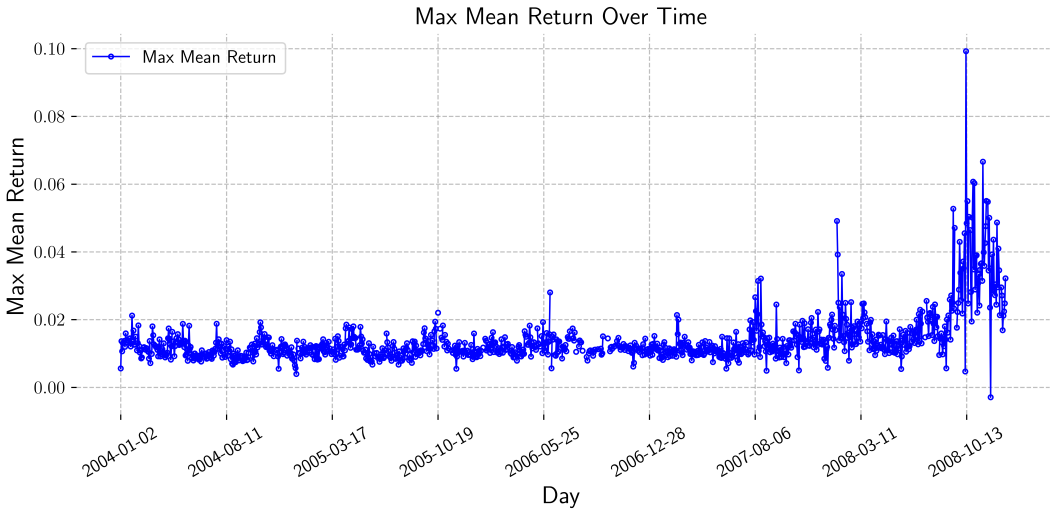


Figure 13: Best strategy as a function of time

Only based on our sample of strategies, and in an ideal –and impossible scenario – where the best strategy for each stock was known in advance, averaging the returns across all stocks would yield the results depicted in Figure 13. Of course, predicting the best strategy for any given day is unrealistic. However, through further analysis, we could attempt to cluster market periods

like it was done in the course, allowing us to determine which strategies perform optimally in each state. For that, we could try exploiting correlation network as presented in Figure 18 for example. This approach would enable a more dynamic and adaptive strategy selection process, helping traders act based on historical market behavior and its resemblance to current conditions.

While some returns are generated from our strategies, some aspects makes it more difficult to put it into practice. In facts, our strategies can easily trade more than five thousand times per day, each generating tiny returns. In practice, it is likely that these small returns would not be enough to absorb associated fees and spread. Another limitation is the ability to compute and allocate in real time using streamed data.

VII Conclusion

We developed a robust financial big data pipeline used to highlight the potential of simple trading strategies in detecting and exploiting intraday bubbles. While some strategies demonstrated strong performance under certain market conditions, the results suggest that a more refined approach could further enhance profitability. Further work could explore multi-asset strategies that trade across all stocks simultaneously rather than applying single-stock strategies. Additionally, expanding the range of strategies tested, incorporating a wider set of parameters, and leveraging clustering techniques to detect market states could significantly improve strategy selection and execution. Finally, portfolio of strategies optimization and using short selling could potentially widen and stabilize the gains.

Appendix

Data structure

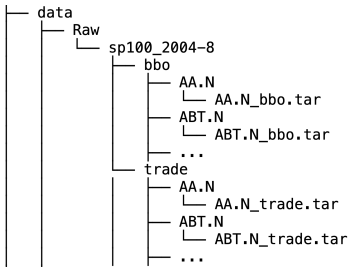


Figure 14: Raw data structure

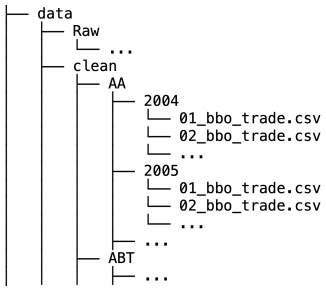


Figure 15: Clean data structure

Best strategy before & after 2007

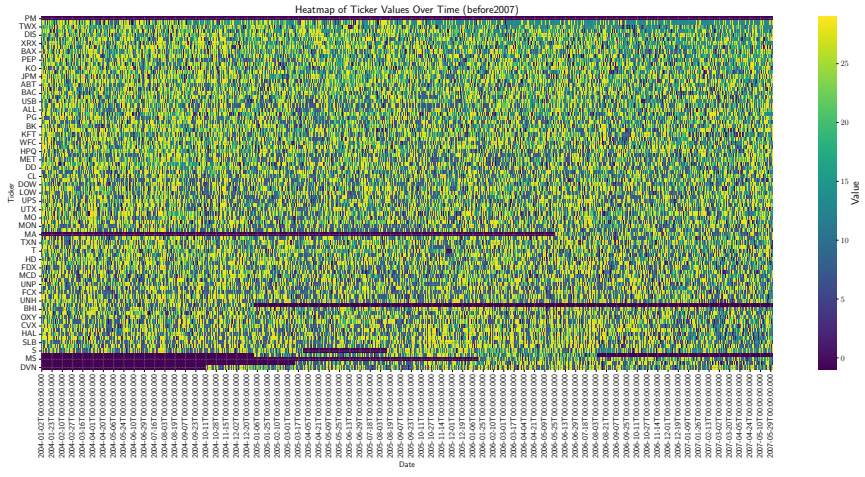


Figure 16: Best strategy over time for each ticker, period before June 2007 (values mapping in Table 4)

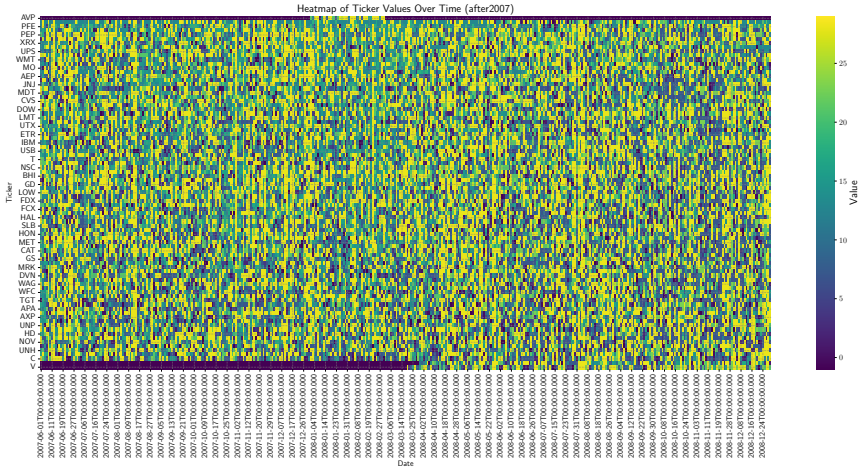


Figure 17: Best strategy over time for each ticker, period after June 2007 (values mapping in Table 4)

Best strategies heat maps correspondences

Key	Category	Parameters
-1	momentum_excess_vol	ps100_pl1500_vs10_vl100
0	momentum_excess_vol	ps100_pl2000_vs100_vl2000
1	momentum_excess_vol	ps10_pl200_vs3_vl180
2	momentum_excess_vol	ps10_pl400_vs10_vl400
3	momentum_excess_vol	ps200_pl4000_vs200_vl4000
4	momentum_excess_vol	ps20_pl2000_vs5_vl50
5	momentum_excess_vol	ps30_pl1200_vs30_vl1200
6	momentum_excess_vol	ps400_pl5000_vs400_vl5000
7	momentum_excess_vol	ps5_pl100_vs5_vl10
8	momentum_excess_vol	ps5_pl150_vs5_vl50
9	momentum_excess_vol	ps5_pl200_vs10_vl300
10	momentum_excess_vol	ps5_pl200_vs2_vl180
11	momentum_excess_vol	ps5_pl200_vs5_vl200
12	momentum_price	s100_l1500
13	momentum_price	s10_l400
14	momentum_price	s200_l4000
15	momentum_price	s30_l1200
16	momentum_price	s400_l5000
17	momentum_price	s50_l2000
18	momentum_price	s5_l200
19	volatility_trading_strategy	s1000_l8000
20	volatility_trading_strategy	s100_l2000
21	volatility_trading_strategy	s10_l400
22	volatility_trading_strategy	s200_l4000
23	volatility_trading_strategy	s250_l2000_t1
24	volatility_trading_strategy	s250_l2000_t3
25	volatility_trading_strategy	s250_l2000_t5
26	volatility_trading_strategy	s30_l1200
27	volatility_trading_strategy	s400_l5000
28	volatility_trading_strategy	s5_l200
29	volatility_trading_strategy	s800_l8000

Table 4: Heatmap mapping

Key	Category
0	Excess volume momentum
1	Price momentum
2	Volatility

Table 5: Heatmap families mapping

Key	Category	Parameters
-1	momentum_excess_vol	ps100_pl1500_vs10_vl100
0	momentum_excess_vol	ps100_pl2000_vs100_vl2000
1	momentum_excess_vol	ps10_pl200_vs3_vl180
2	momentum_excess_vol	ps10_pl400_vs10_vl400
3	momentum_excess_vol	ps200_pl4000_vs200_vl4000
4	momentum_excess_vol	ps20_pl2000_vs5_vl50
5	momentum_excess_vol	ps30_pl1200_vs30_vl1200
6	momentum_excess_vol	ps400_pl5000_vs400_vl5000
7	momentum_excess_vol	ps5_pl100_vs5_vl10
8	momentum_excess_vol	ps5_pl150_vs5_vl50
9	momentum_excess_vol	ps5_pl200_vs10_vl300
10	momentum_excess_vol	ps5_pl200_vs2_vl180
11	momentum_excess_vol	ps5_pl200_vs5_vl200

Table 6: Heatmap families mapping

Correlation networks

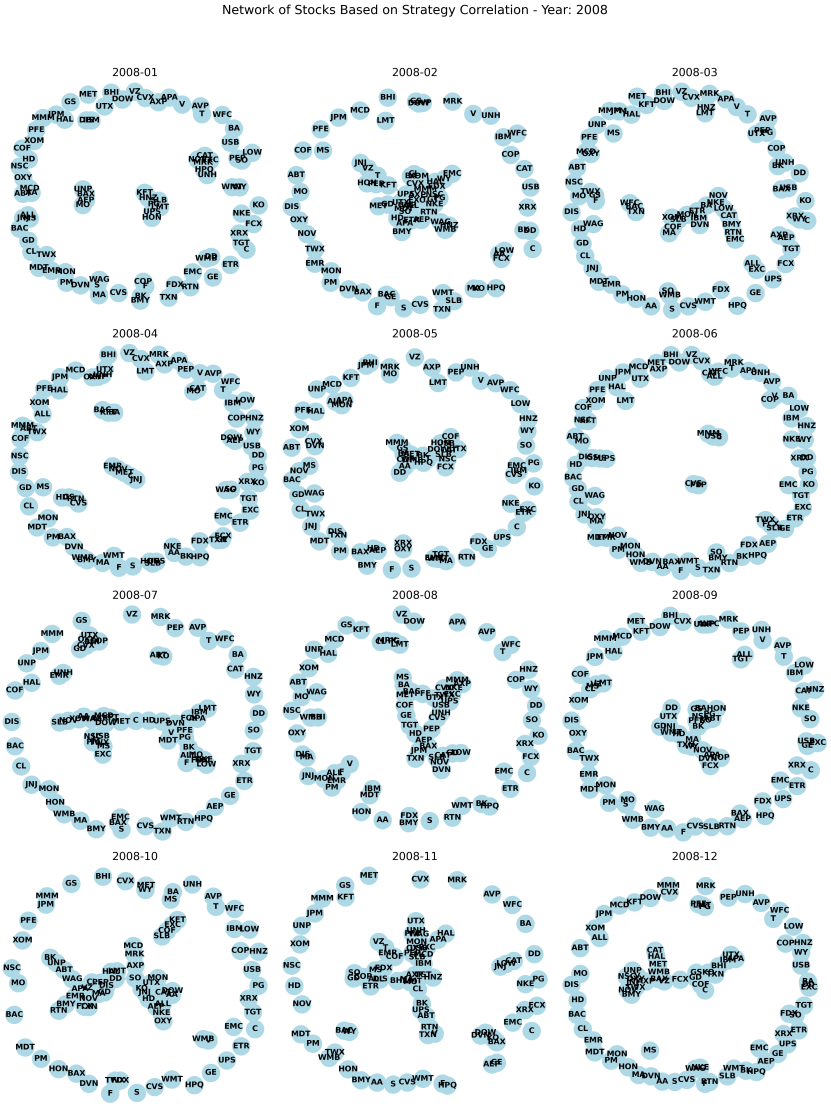


Figure 18: Monthly correlation networks for year 2008