Detecting spoofing in high frequency trading using machine learning techniques.

Iryna Veryzhenko*
Conservatoire national des arts et métiers (LeCnam) 292 rue Saint-Martin 75141
Paris Cédex 03, France
Laboratoire interdisciplinaire de recherche en sciences de l'action (LIRSA
Research Center)
https://orcid.org/0000-0003-3396-7886

Nohade Nasrallah
EM Business School Strasbourg, France, 61 avenue de la forêt-noire,
67085 Strasbourg Cédex, France, Mobile: +33 6 11 93 32 92
LaRGE Research Center
nnasrallah@em-strasbourg.eu

Henri Garcia
École Nationale Supérieure d'Arts et Métiers
henri.garcia@ensam.eu

* Corresponding Author
iryna.veryzhenko@lecnam.net

Abstract

This study focuses on spoofing detection in high frequency trading using machine learning techniques. The primary goal of this study is to explain how effective these techniques are in detecting manipulative orders within real-world order books updated at microsecond time grain. To conduct this research we use a supervised learning algorithm for classification, k-nearest neighbors (KNN). The outstanding feature of this study is a combination of research approaches: agent-based modeling and a rich empirical study. We use order books from the artificial financial market, which guarantees a perfect traceability of the actions of its participants, to train the algorithm, then we apply it to empirical order books from the Euronext stock exchange. Findings indicate the KNN algorithm demonstrates robust detection capabilities, albeit exhibiting sensitivity to the nuances of the training data.

Keywords: Spoofing, machine learning, market microstructure, high-frequency trading

1. Introduction

In an era dominated by fast and algorithmic trading, there is a rising concern about momentum-ignition strategies that can exacerbate the market quality and accentuate asymmetric information and toxic flow (Biais et al., 2015; Hoffmann, 2014; Foucault et al., 2017). Regulatory settlements suggest that markets highly populated with algorithmic limit orders could be particularly susceptible to manipulative trading tactics (CFTC, 2018; DOJ, 2020). Pirrong (2017) defines price manipulation as 'intentional conduct that causes market prices to diverge from their competitive level'. Following (Kyle & Viswanathan, 2008), market manipulation is a trading strategy with the intent to pursue a scheme that undermines economic efficiency both by making prices less accurate as signals for efficient resource allocation and by making markets less liquid for risk transfer.

Notably, the attention to electronic limit order books has been growing due to the 2010 Dodd-Frank Act, which explicitly makes 'spoofing' a criminal act in the commodities and futures markets. Spoofing is an illegal act of artificially modifying the supply to temporarily drive prices in a given direction for profit (Xuan Tao Andrew Day & Drapeau, 2022). It is described as a manipulation (Angel & McCabe, 2013; Cartea et al., 2020; Shubber & Stafford, 2020; FCA2021) where the spoofer exercises some sort of devious means to control the market and in doing so harms the legitimate market activities. It involves the manipulator posting limit buy or sell orders on a trading platform, signaling their intent to trade a specific amount of shares at a declared price, with the option to revoke these orders at any time(Khorasanee, 2024). This tactic is employed to influence the market price to benefit a different trade the manipulator intends to make. After achieving a more advantageous price for this separate deal, the manipulator typically withdraws the initial orders before they can be finalized and executed. In this sense, traders adopt stealth trading and order-splitting strategies to disguise limit orders' true size (Barclay and Warner, 1993; Engle et al., 2012; Chan and Lakonishok, 1995; Chou and Wang, 2009; Pérold, 1988; Yeo, 2005).

Although most jurisdictions do not explicitly outlaw spoofing, there is an intense reflection about their unethical repercussions. Nonetheless, the debate about market microstructure has long encountered many difficulties and empirical results and definitions diverge in the identification, perception, and analysis of trading strategies. The problem is rooted in many areas. First, the dynamic evolution of the limit order book is complex and nonlinear. Statistical characteristics and relationships with economic and political variables change over time. Second, there has been a keen interest in technological development, allowing some high-frequency traders to exploit these advancements and use ultra-fast algorithms to increase profitability and stay competitive. Third, the data access is limited and some features and privileges are not provided to all users. Fourth, there are some inconsistencies in the identification proxies of different traders. Fifth, some methodological approaches failed to circumvent inherent caveats and biases that might result while accounting for important endogenous and exogenous factors. For instance, empirical evidence of predictability is often impaired by the non-stationarity of such time series.

Nowadays, the advent of Machine Learning (ML) might overcome such shortcomings as traditional methods are often black-box models that need more transparency and interpretability (Han et al., 2022). ML is a part of the field of artificial intelligence whose methods are capable of making decisions based on mathematical models. ML algorithms are generally used to analyze data through a prior learning process. This learning process involves the capitalization of data, studies, and other models to build a discrimination base for decision-making. Mankad et al. (2013) propose a dynamic ML method to uncover and analyze the ecosystem of an electronic financial market. It aims to identify and understand the relationships among various market participants, such as high-frequency traders, liquidity providers, and other market agents. Han et al. (2022) propose an explainable ML framework for discovering the dynamics of high-frequency trading in financial markets.

The prior literature has investigated market manipulations using ML techniques. Yet, our paper serves to fill the gap in behavioral finance indicating that strategic investors take advantage of others' behavioral biases, most likely those of individual investors, and profit from them through spoofing. Thus, we develop an ML numeric tool to detect spoofing in high-frequency order books. In ML, there are two families of models: Supervised ML algorithm requires the use of labeled data to learn how to perform classification. Following this learning step, the system can provide a label for any new input data. Unsupervised ML algorithm is used when there is no information about the classification or the label. The system studies the data to model the classification model. This type of model is useful when we do not have much information on how to classify the data.

The outstanding feature of this project is a combination of research approaches: agent-based modeling and a rich empirical study. We developed a simulator, which acts as an artificial financial market. The main advantage of the agent-based methodology used to build this artificial market is the perfect traceability of results and observations. This methodology characteristic is important in the training stage of machine learning algorithms. All spoofing episodes are identified. We analyze the sets of data generated by 1,000 fundamentalists, 100 liquidity pressure followers, and 1 spoofer, who randomly manipulates the market (on average 5 times daily). Market participants are classified based on the following features: price, volume, type timing and lifetime of each order, volume and price of the best bid and best ask at the moment of order submission. Additionally, each trader is characterized by a vector of the timing of his transactions. We apply supervised algorithms, such as the Support Vector Machine (SVM), supervised Classification and Regression Tree (CART) algorithm, and K-Nearest Neighbors (K-NN) algorithm, to detect typical characteristics of a spoofing episode. We show that the K-NN effectively deals with the classification of manipulative and non-manipulative orders with a 99.375% degree of precision. We get 6x10-4 % of false positive and 1.3872 % of false negative order classifications.

Then, all algorithms, trained based on simulated data, run through the real market order flow. For this purpose, we use the rich BEDOFIH AMF – Euronext Paris High-Frequency database, a source that has not yet been exploited in spoofing examinations. This source includes all the messages received by the market operator over a trading session and assigns a particular marker to distinguish three types of traders: pure High-Frequency Traders (HFTs), traders operating both high-frequency and non-high-frequency (MIXED HFTs - investment banks), and non-high-frequency traders (NON-HFTs). Thus, such data enables us to distinguish the effects of activities among different categories of traders on price efficiency during a trading session.

2. Literature review

The dynamics of limit order books have been explored through stochastic process theories and queueing theory. This field of research is well-documented by significant contributions from Cont *et al.* (2010), Hult and Kiessling (2010), Cont and De Larrard (2013), Huang *et al.* (2015), Huang and Rosenbaum (2017) and Muni Toke and Yoshida (2017). To reduce the inherent complexity of these dynamics, key assumptions were commonly adopted to make these models more manageable by applying the Markovian properties to the limit order book, treating independently the order flows at distinct levels, and the standardizing order sizes.

In this regard, a deeper understanding of the origins and nature of price changes provides a conceptual bridge between the microeconomic mechanics of order matching and the macroeconomic concept of price formation. To wit, the use of trading strategies with the intention of misleading other market participants is called "market manipulation." For instance, illegal price manipulation includes corners and squeezes, pump-and- dump manipulation, and failure to make required disclosures. Theoretically, the step-function theory of Fox et al. (2021) explains that asymmetry in the volumes posted on the best bid and the best ask is interpreted by market participants as good or bad news about the asset. Chakraborty and Yilmaz (2004) and Mei et al. (2004) analyzed the pump-and-dump manipulation in a stock market. Allen & Gale (1992) discussed the possibilities of trader-based manipulation and showed that a manipulator could pretend to be informed and mislead the market. Allen & Gorton (1993) showed that the asymmetry between the information associated with buying and selling (i.e., a buy contains more information than a sell) leads the manipulator to buy, causing a higher effect on the price and sell with a lower effect. More centrally, Egginton et al. (2016) and Gao et al. (2015) look at cancellation activity rates and find that a large number of cancellations is associated with lower market quality. Van Ness et al. (2015) find a negative relationship between cancellations and market quality. Shorter and Miller (2015) point out that high-frequency trading firms may engage in potentially manipulative strategies involving the usage of quote cancellations. To emphasize, it is used by a high-frequency trader to build market power by taking advantage of both the behavioral weaknesses of individual investors and microstructural loopholes of trading venues (Dalko et al., 2020).

Narrowly, the prior literature about spoofing emerges unconsolidated in many areas. Some research argue that spoofing is illegal (Shubber & Stafford, 2020) while others pinpoint the innocuous effect of such strategies (Khorasanee, 2024). Others argue that some strategies are natural and are unintentional. For instance, Lee et al. (2013) use a proprietary dataset with trader identification from the Korea Exchange to show that spoofing achieves substantial extra profits and spoofing tends to target stocks with higher return volatility, lower market capitalization, lower price level, and lower managerial transparency. Wang (2019) uses data from the Taiwan Futures Exchange to show that market participants spoof the order book in stocks that exhibit high volumes of trading, high volatility, and high prices. Cartea et al. (2020) derive an optimal spoofing strategy to acquire or liquidate a large position where they explicitly encode spoofing as an intentional action. Furthermore, recent technical studies endeavor to demonstrate that algorithms can learn to coordinate their spoofing when they learn together. This highlights the unintentional effects of algorithms that learn to collude (Calvano et al., 2021; Colliard et al., 2022), and Dou et al., 2023) and provides critical analysis about the convergence to collusive equilibria.

The recent advent of ML models for limit order book data helped to further investigate the many questions that are still dilemmic with regard to the different conditions that surround market microstructures and dynamic environments. ML models overcome shortcomings of traditional methods, often considered as black-box models that need more transparency and interpretability (Han et al., 2022). ML algorithms are generally used to analyze data through a prior learning process. This learning process involves the capitalization of data, studies, and other models to build a discrimination base for decision-making. One of the most important question is the ability of the existing models to estimate the probability of frauds and market manipulations. For instance, the supervised learning methods are used for fraud recognition in high-frequency markets using daily or intraday data with emphasis on market quality, efficiency, fairness, and stability. Supervised learning is a method of detecting market manipulators that are similar to trends that are already known to be manipulative activities. Data mining methods can detect market fraud and experimental results in the literature are encouraging. However, there are many challenges in designing and developing data mining methods for detecting price manipulation in the market, including heterogeneous data, privacy, performance, and legal implications.

Yet, there are several supervised learning models and several successful examples of using ML. Oğut et al. (2009) use ANN and SVM models to detect fraudulent activity. The hypothesis of their work is that price (hence return), volume, and volatility increase during the manipulation period and decrease in the post-manipulation phase. Their study shows that ANN and SVM models are better-performing ML methods than multilinear statistical techniques (56% versus 54%). On another echelon, Diaz et al. (2011) discuss the challenges of applying data mining techniques to detect stock price manipulation by mining financial variables (mainly ratios and textual sources). The different data sources that were combined to analyze over 100 million trades and 170,000 quotes in this study include: profiling information (trading venues, market capitalization, and beta), intraday trading information (price and volume over the course of a year), and financial information and filing reports. The researchers train their clustering algorithms on a training set of data from their database. Their study confirms the existence of higher liquidity, return, and volatility during a manipulation episode in agreement with the previous hypothesis of Oğut et al. (2009). Another study by Golmohammadi et al. (2014) reuses the database of Diaz et al. (2011) to rank the performance of some supervised learning classification methods. The researchers test CART, conditional inference trees, C5.0, Random Forest, Naive Bayes, neural networks, SVM, and KNN methods for the classification of manipulated samples. They show that Naive Bayes, KNN, CART, and Random Forest methods perform better than the other algorithms with higher sensitivity and accuracy. They emphasize that the performance of these algorithms must be interpreted correctly. Indeed, some models exceed accuracy rates of 90%. This is due to the number of "manipulated" class samples being much lower than the number of "non-manipulated" class samples.

Many studies tackle the effect of such manipulations on market quality. Kercheval and Zhang (2015) use support vector machines to forecast movements of the mid-price and price spread crossing. They find indications of short-term predictability of price movements. Their dataset covers a complete trading day for five stocks listed at the Nasdaq. The features extracted from this dataset included the prices and volumes for ten levels of the ask and bid side of the order book as well as bid-ask spreads. They used support vector machines with various feature sets and evaluated their models with cross validation. However, they did not test their forecasting procedure in a realistic trading experiment but used only a period of four hours to carry out some sort of plausibility check. More advanced work attempts to predict mid-price movements with different architectures of neural networks, including RNNs (Dixon 2018) and convolutional neural networks (CNNs) (Doering *et al.* 2017). Mankad et al. (2013) propose a dynamic ML method to uncover and analyze the ecosystem of an electronic financial market. It aims to identify and understand the relationships among various market participants, such as high-frequency traders, liquidity providers, and other market agents. Sirignano (2019) uses a neural network architecture for modelling the joint distribution of ask and bid prices at a future time. Nousi et al. (2019) and Han et al. (2015) also found indications for the predictability of mid-price movements using similar sets of features but different observation periods (ten days and thirty minutes, respectively) and other machine learning algorithms (neural networks and random forests, respectively) in addition to support vector machines.

Moreover, Golmohammadi et al. (2014) explain that labeled data (transactions characterized as fraudulent or non-fraudulent for market fraud detection) are very rare because: Data tagging is very expensive and usually requires a survey by auditors; The number of positive samples (fraud cases) is a very small percentage of the total number of samples. To overcome this problem, some researchers have carried out simulations to produce more representative databases that better describe the fraudulent behavior of certain traders. In the study by Ladley (2023), an artificial order book is simulated to study the impact of manipulations on the quality of the market. This artificial market allows them to test the profitability of manipulative orders under different market conditions: average trading volume, fundamental value volatility and tick size. These same simulated data are used by Youssef (2020), thanks to the data simulated by Ladley (2023), manages to create a CART model aiming at detecting manipulative orders.

On a wider spectrum, ML models serve to uncover and analyze the ecosystem of an electronic financial market. It aims to identify and understand the relationships among various market participants, such as high-frequency traders, liquidity providers, and other market agents (Mankad et al., 2013). Han et al. (2022) propose an explainable ML framework for discovering the dynamics of high-frequency trading in financial markets. In the present study, we attempt to detect spoofing using ML by relying on the rich data of BEDOFIH AMF – Euronext Paris High-Frequency database. The objective is to develop a simulator, identify spoofing episodes based on real data, and distinguish the spoofing's effects among different categories of traders on price efficiency during a trading session

3. Model and data description

For this study, we rely mainly on the articles by Ladley et al. (2023) and Youssef (2020). This paper aims to complement their studies by adding an empirical dimension and a new method of analysis. We take the order book data simulated by Ladley et al (2023) and then we deepen the study of Youssef (2020) through an empirical study applied to a new classification method.

We pose the following research question: to what extent are supervised learning machine algorithms able to detect spoofing in empirical order books? For this purpose, we use supervised machine learning methods trained on order books simulated by Ladley et al. (2023) and then test the model on empirical order books from the Euronext exchange.

In the next section, we explain the origin of the collected data, how it was cleaned, and how the quality of this data was controlled.

3.1 Artificial market data

The simulated data used in this study are from the work of Veryzhenko and Oriol (2019). That study relies on ArTifcial Open Market (ATOM) (Brandouy 2013), a highly flexible simulation platform that allows different parameter settings for the microstructure and traders' behaviors for different scenarios. Spoofers send a large-volume buy/sell limit order, to give a false impression of strong buying/selling pressure and lead others to create an upward or downward price trend. This creates an illusion of liquidity in the order book. Once the trend is initiated, the spoofer cancels the large-volume limit order and submits an ask/bid market order on the opposite side of the book at an artificially better price than before the spoof order. All traders interact in a non-trivial way through the central limit order book. All orders are executed according to the Euronext rules.

The main advantage of agent-based methodology used to build this artificial market is the perfect traceability of results and observations. This methodology characteristic is important in the training stage of machine learning algorithms. All spoofing episodes are clearly identified. We analyze the sets of data generated by 1,000 fundamentalists, 100 liquidity pressure followers and 1 spoofer, who randomly manipulates the market (Ladley et al, 2023).

We employ 50 simulated datasets, each representing a single day, with an average of 215,778 orders per file. Consequently, our dataset comprises a total of 10,788,894 orders derived from these simulated order books. Among all these orders we have 43,877 so-called spoofing orders.

3.2 Euronext high-frequency data

We use the rich BEDOFIH AMF - Euronext Paris High-Frequency database, a source that was not yet exploited in fairness examinations. This source includes all the messages received by the market operator over a trading session, indicating high-frequency traders' complex behavior and the effect on market fairness.

This data enables us to distinguish the effects of activities among different categories of traders on price efficiency during a trading session. We consider

three types of traders: pure High-Frequency Traders (HFTs), traders operating both high frequency and non-high-frequency (MIXED HFTs - investment banks), and non-high-frequency traders (NON-HFTs). Once a trader is classified, it is immutable.
We effectively reconstruct the complete depth of the order book in an event-driven fashion by capturing the evolving state of the order book at each update, akin to taking snapshots. ATOM (Brandouy et al., 2013) is employed for this purpose. The specific order book under consideration pertains to the assets of Air France – KLM throughout the month of June 2016.

Considering the microsecond frequency of our observations and the computational time required, we have opted to utilize a 3-day observation period for the initial phase of this study. To underscore the significance of our efforts, we aim to present the sheer volume of data received. Each set of print screens capturing the total depth of the bid/ask sides of the central order book, stored as raw text, amounts to 40 gigabytes for a single security per day. Subsequently, we plan to extend this study to encompass the entire 22 days of June 2016.

Given that the Euronext order book processes 506,482 orders submitted for buying or selling Air France equities, our analysis involves working with a total of 1,519,448 order book screens. So, 1,519,448 orders should be classified.

3.3 Data quality

To be able to compare simulated and empirical order books, we proceed to a normalization of the numerical values. We choose to make a standard normalization according to the following formula:

$$x_{normalized} = \frac{(x-\mu)}{\sigma}$$

μ denotes the average value per tick, and σ – standard deviation per tick.

In the same way, we carry out an encoding for the algebraic values. We choose to encode them according to the rules described above:

- For the nature of the orders : $Nature=\begin{cases}0: Bid\\ 1: Ask\end{cases}$

- For order type: $Type = \begin{cases} 0: others \\ 1: limits \end{cases}$

- For the nature of the traders : $Trader = \begin{cases} 0: others \\ 1: spoofers \end{cases}$

Additionally, we compute the tick-by-tick spread. Here is the corresponding formula:

$$Spread_{ticki} = max\left(Bid_{ticki}\right) - max\left(Ask_{ticki}\right)$$

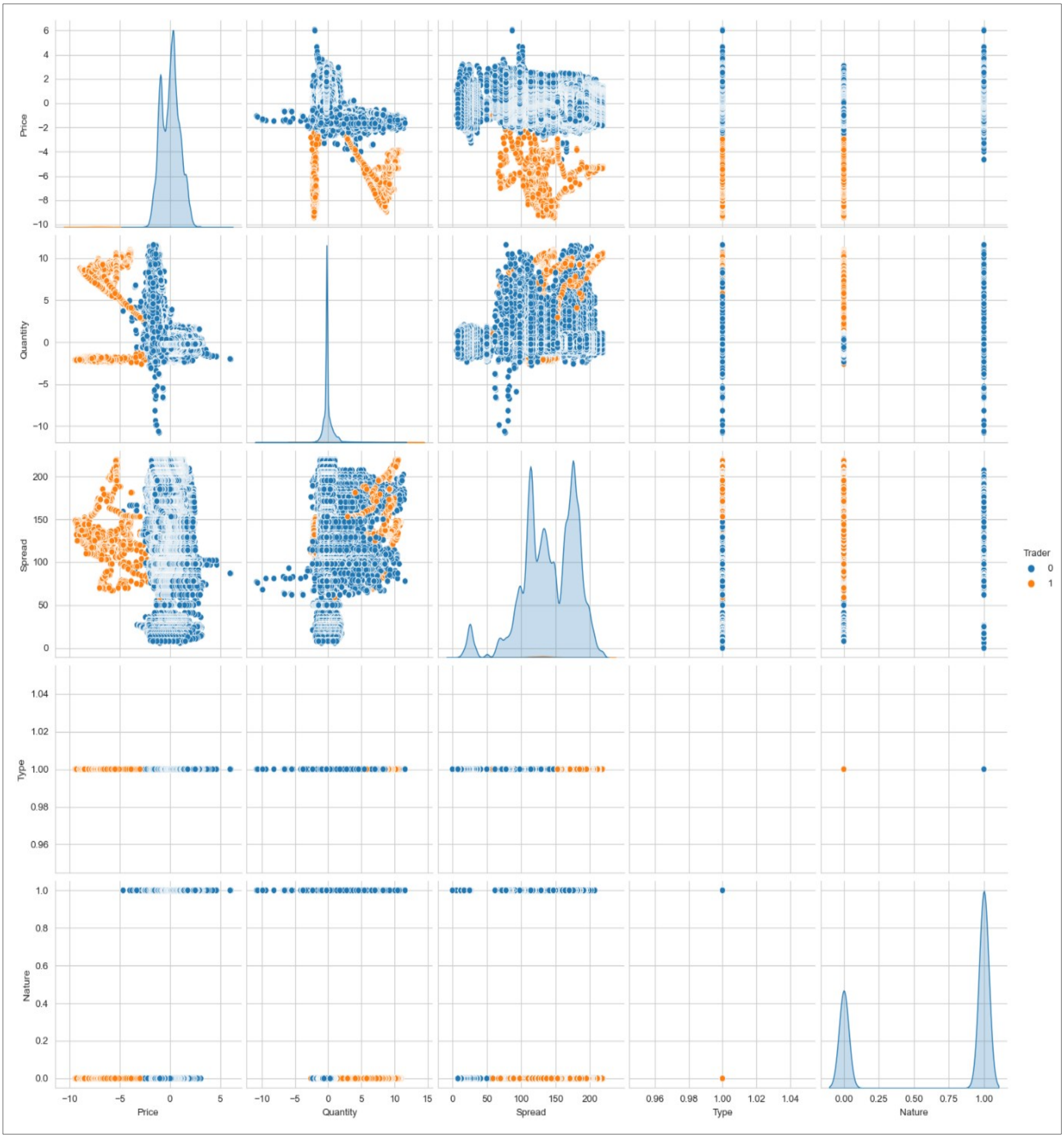


Figure 4 - The simulated data after normalization and encoding

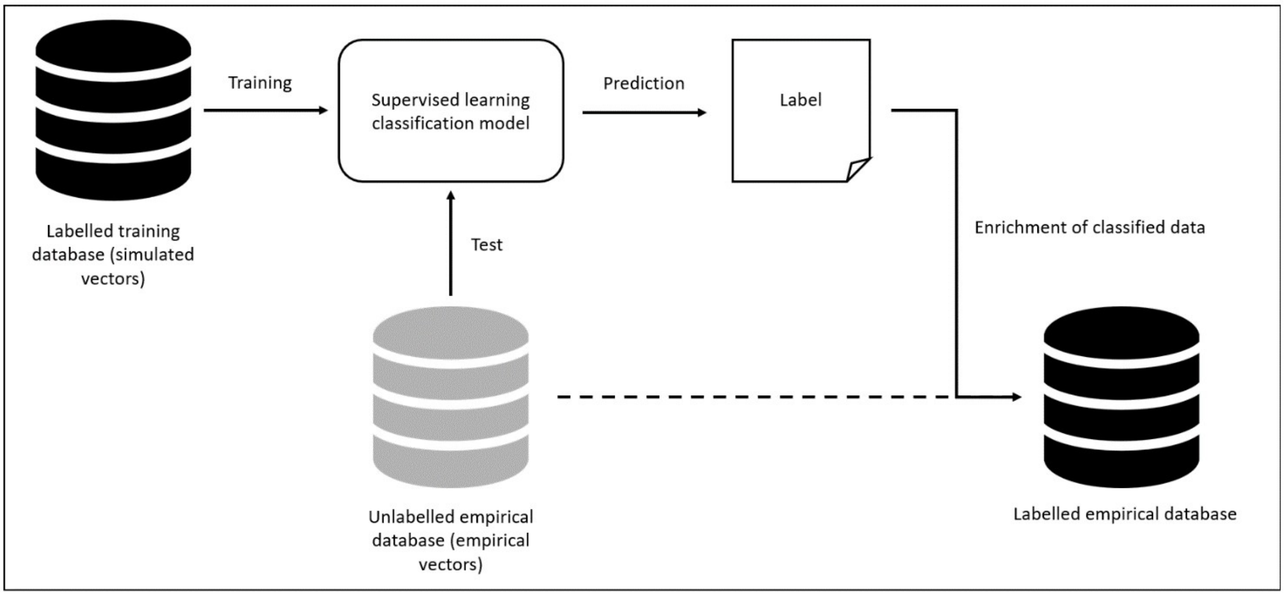
In Figure 4, the distribution of the simulated data is illustrated, allowing each variable to be observed in relation to the others. This visualization provides multiple insights into the strategies employed by the spoofer within the artificial market.

It is observed that spoofers tend to place buy orders at relatively low prices with significantly higher volumes compared to the average. Spoofers aims to avoid immediate execution of their manipulative orders. The spoofers amplify the volume of the manipulative order book, creating an illusion of buying pressure to prompt other market participants to follow suit. Additionally, the presence of spoofers is associated with higher spreads and larger trade volumes, consistent with findings from Öğüt et al. (2009) and Diaz et al. (2011). Moreover, heightened liquidity, returns, and volatility are noted during manipulative episodes.

3.4 Supervised learning process

This section delves into the methodology employed to establish our supervised learning model. We provide a comprehensive overview of the chosen method and outline the empirical and simulated vectors selected for training and testing the model.

A classification method based on supervised learning is employed for detection. The training dataset with labeled data is derived from the artificial market, where each vector is pre-categorized as manipulative or non-manipulative. This is in contrast to the empirical real market vectors, which lack predefined labels. Consequently, the primary objective of the study is to assign labels to real market orders based on their inherent characteristics. The schematic representation of this process is elucidated in Figure 5 below.


Training
Supervised learning classification model
Prediction
Label
Labelled training database (simulated vectors)
Test
Enrichment of classified data
Unlabelled empirical database (empirical vectors)
Labelled empirical database

Initially, the model undergoes training to establish a discrimination criterion capable of classifying vectors. Subsequently, we apply the model's criteria to classify the empirical vectors, and finally, we enhance the empirical dataset by incorporating the labels predicted by the model.

3.4.1. Parameters vector

Following the supervised learning approach, the selection of training and empirical vectors is based on the simulated data. In the parameter selection process, we maximize the information available to us. The parameters chosen for the training vector include the order price, volume, bid/ask book spread at the time of order submission, order type, order direction, and trader profiles.

$$\text{training vector} = \begin{bmatrix} \text{Price} \\ \text{Volume} \\ \text{Spread} \\ \text{Type} \\ \text{Direction} \\ \text{Trader profile} \end{bmatrix}$$

The label for this training vector is determined by the trader parameter, which indicates whether the associated order vector is manipulative or non-manipulative. The 5 other parameters are thus common to the empirical vector, which is written as follows

$$\text{empirical vector} = \begin{bmatrix} \text{Price} \\ \text{Volume} \\ \text{Spread} \\ \text{Type} \\ \text{Direction} \end{bmatrix}$$

The price parameter is important because if a trader is inclined to manipulate the market, he often places orders with minimal chances of execution, causing the price to deviate significantly from the market-accepted price for the stock (asset). At the same time, the order must be placed in the top 5 of the order book to ensure visibility to other market participants. Volume also plays a critical role, as manipulators tend to distort information about market liquidity by executing large volume orders when buying or selling assets. Spread provides additional insight into the quality of order placement. The Type and Nature parameters help refine the model's classification, categorizing orders based on limit/market order and buy/sell information. In particular, in this study, the vector parameters are calculated per tick, which includes the treatment of all orders in the order book.

3.4.2 K-Nearest Neighbors Model

In this section, we present the classification model capable of labeling the empirical vectors. We also describe in detail the method used to calculate the performance of the chosen model.

To carry out this study, we use the K-Nearest Neighbors (KNN) classification model. This model belongs to the supervised machine learning methods. According to the study conducted by Golmohammadi et al. (2014), this model has performances and characteristics that are suitable for fraud detection.

The classification performed by this model uses the Euclidean distance in N dimension. In a first step, the training vectors are placed in a vector space of dimension N. This first step is called training because it is used as a reference for the N discriminative criteria induced by the placement of the vectors in the vector space. In a second step, an empirical vector of the same dimension is placed in the same vector space to compute the Euclidean distances to the previously placed training vectors. Among all the distances computed, the algorithm will select only the k smallest distances. Thus, we will have found the *k* nearest neighbors. It is then sufficient to average the labels of the *k* nearest neighbors to predict the label of the empirical vector.

Mathematical formalization of the KNN model

Let V_{entr} a vector from the training database with n characteristics such that,

$$\forall n \in N, \forall i \in [0,n], a_i \in R, V_{entr} = (a_0, a_1, a_2 \ldots a_{n-1}, a_n) \, with \, a_n \leq vector\ label$$

Let V_{simul} be a vector from the simulation bank with n-1 features such that,

$$\forall n \in N, \forall i \in [0,n-1], b_i \in R, V_{simul} = (b_0, b_1, b_2 \ldots b_{n-2}, b_{n-1})$$

The Euclidean distance between the two vectors is then written :

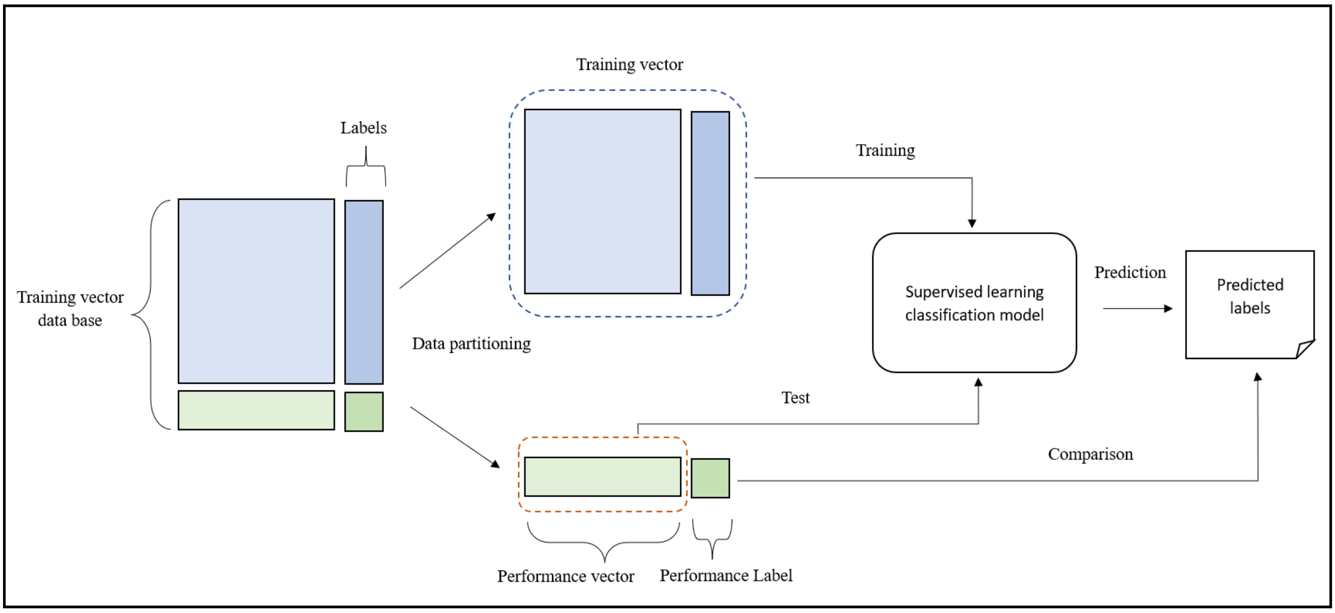
$$dist(V_{entr}, V_{simul}) = \sqrt{\sum_{i=0}^{n-1} (b_i - a_i)^2}$$

Once the set of distances is found, we just have to group the k smallest distances and average the nature of the labels a_n of the simulated vectors to add the label dimension to the vector V_{simul}. The output vector is then written,

$$\forall n \in N, \forall i \in [0,n], b_i \in R, V_{simul} = (b_0, b_1, b_2 \ldots b_{n-1}, b_n) \, with \, b_n \leq vectorlabel$$

Method of calculating the performance of the model

To test the performance of a supervised learning algorithm, we simply use training vectors. In other words, we split the database of training vectors into two data sets. The first is used to train the algorithm, and the second is used to test its performance. The advantage of using vectors from the training data is that they are labeled. Therefore, it is easy to compare the result predicted by the algorithm with the real result. Here is an illustration of the process:



To conduct performance calculations, we commence by partitioning the training vector database. Typically, this partition involves allocating 2/3 of the vectors for training and the remaining 1/3 for performance assessment. In the subsequent step, we train our model using the designated 2/3 of training vectors. Once the algorithm is trained, we evaluate its performance on the reserved performance vectors. The labels predicted by the algorithm are then scrutinized and compared with the actual performance labels. This testing methodology aids in defining the optimal number 'k' of nearest neighbors, ensuring the model attains the highest accuracy. It also assists in determining the appropriate number of training vectors to prevent "overtraining" the algorithm and facilitates the creation of a confusion matrix capable of gauging the error probabilities of the model.

4. Results

4.1 Performance of the model

In this section we analyze the performances of our algorithm. We will show the general performance results through precision graphs and confusion matrices.

Training volume

To estimate the performance of the algorithm, we need to determine the number of data that will constitute the database of training vectors. To do this, we calculate the accuracy of the algorithm using a simulated order book. That is, we concatenate the orders of the order books to find the volume of orders that will give the best accuracy. The accuracy of the algorithm corresponds to the percentage of good predictions. Here is the graph describing the accuracy of the model as a function of the number of order books making up the training database, with a number k of nearest neighbors arbitrarily set at k=5:

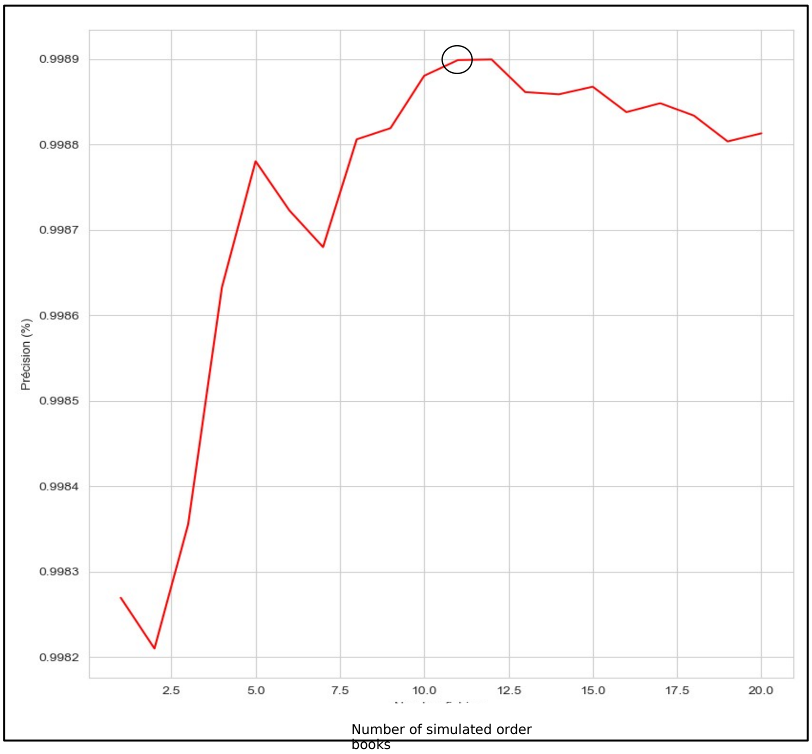


Figure 7 - Prediction accuracy as a function of the number of simulated order books

Determining the optimal volume for the training vectors is crucial to avoid overtraining the model. This challenge is highlighted by Golmohammadi et al. (2014), who highlight the imbalance between the number of samples in the "manipulated" class, which is significantly smaller than the number in the "unmanipulated" class. The graph above illustrates this phenomenon, showing a decline in accuracy beyond 11 days of order book snaps, which is indicative of overtraining symptoms. While the variations are subtle, the sensitivity of supervised learning models to training samples requires careful consideration of the training vector volume. Simulation results indicate that the optimal volume is achieved with 11 simulated days of order book snaps. In subsequent steps, these 11 simulated order books are chosen randomly, as the distribution of volumes for manipulated and unmanipulated orders remains consistent across simulated order books (see Figure 1 and 2 for volume comparisons). In summary, the model achieves an accuracy of 99.89% when the training database consists of 11 simulated days of order book snaps, representing approximately 2,200,000 orders, including 8,500 manipulated orders.

Number k of nearest neighbors

After establishing the volume of the training vectors, the optimal number 'k' of nearest neighbors that maximizes the model's accuracy can be determined. Accuracy is calculated following the method outlined in section 3.3. To identify the maximum accuracy, we vary the number of nearest neighbors. The resulting graph from the simulation is depicted in Figure 8.

In the depicted Figure 8, the accuracy is illustrated in relation to the number of nearest neighbors. It is observed that the optimum precision is achieved when the number of nearest neighbors is k = 3. Additionally, as the number of nearest neighbors increases, there is a corresponding decrease in accuracy. This decline highlights the algorithm's high sensitivity to training samples, indicative of a potential symptom of overtraining. Thus, the optimal number of nearest neighbors is determined to be k = 3.

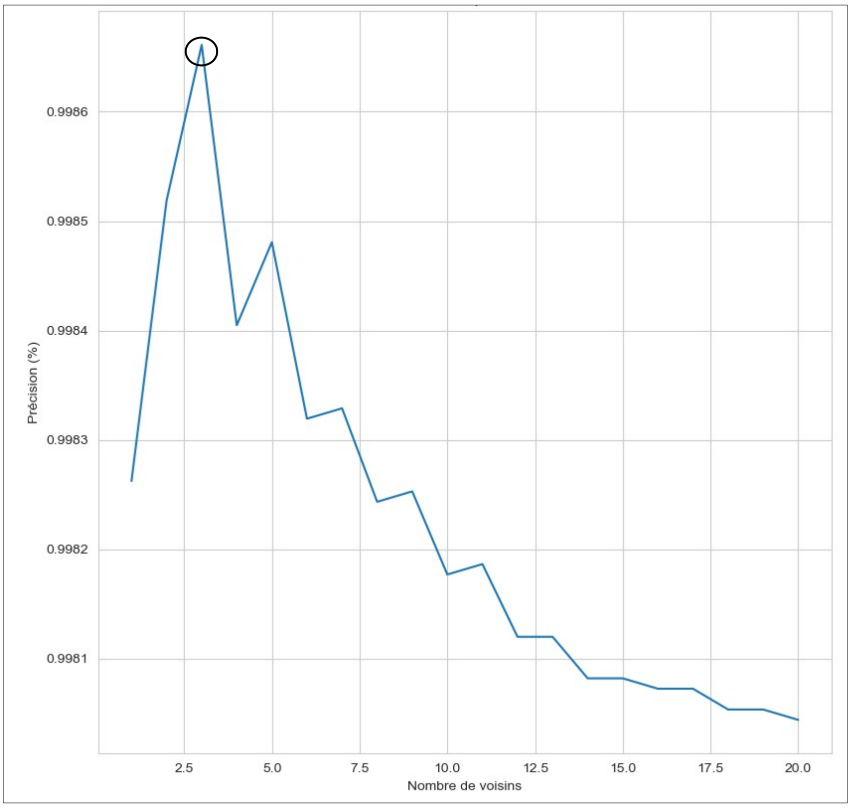


Figure 8 - Prediction accuracy as a function of the number of nearest neighbors

Confusion matrix

After identifying the training volume and the optimal number of nearest neighbors, our next step is to assess the algorithm's ability to predict accurate labels. To achieve this, we construct a confusion matrix.

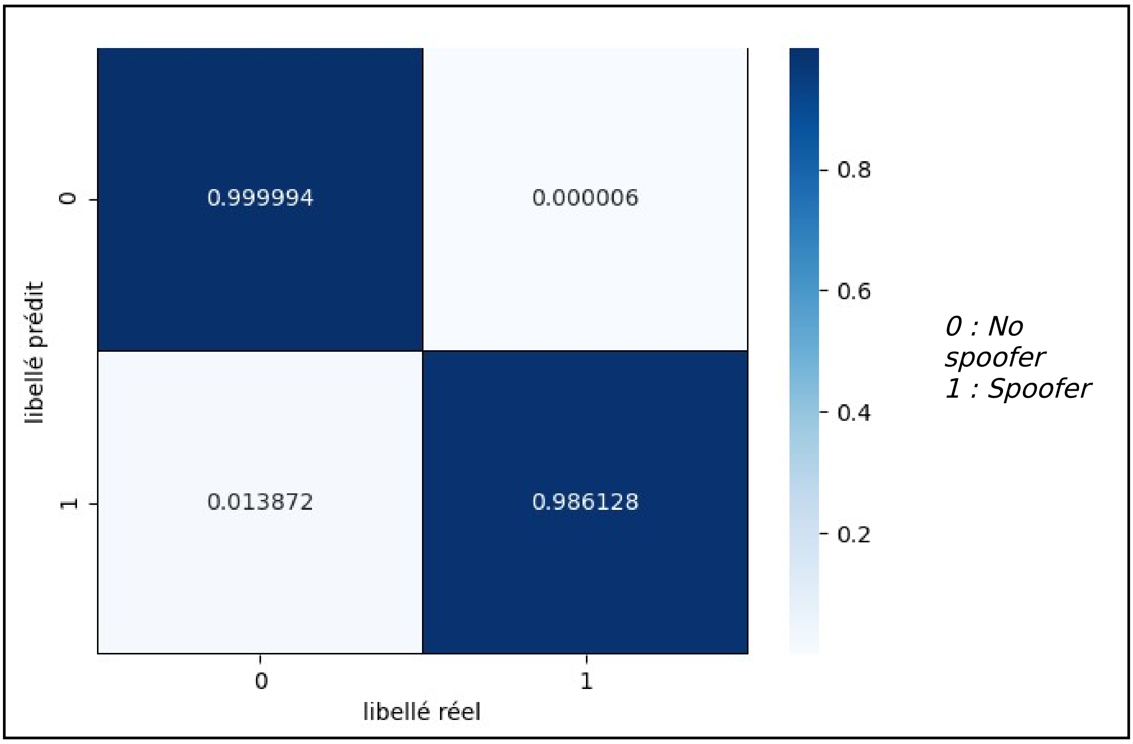


Figure 9 - Model confusion matrix

In the confusion matrix, we have a true positive of 0.999994 and a true negative of 0.986128, which means that 99.9994% of non-spoofers were classified as non-spoofers by our algorithm and 98.6128% of spoofer were classified as spoofer. The false positives and false negatives are $6x10^{-6}$ and 0.013872, respectively, which means that $6x10^{-4}$ % of non-spoofers were classified as spoofer and 1.3872% of spoofer were classified as non-spoofers.

Detection zones

In this section, we showcase the discrimination zones computed by the algorithm. These zones represent the two-dimensional projection of the discrimination volume within the vector space where all algorithm inputs are situated. This visualization provides insights into the calculated discrimination boundaries determined by the model.

Price-Volume

On the graph below we can observe in yellow the projected area for which the algorithm considers that the price and volume parameters correspond to a manipulative order. In purple we can see the projected area for which the algorithm considers that the price and volume parameters correspond to a non-manipulative order.

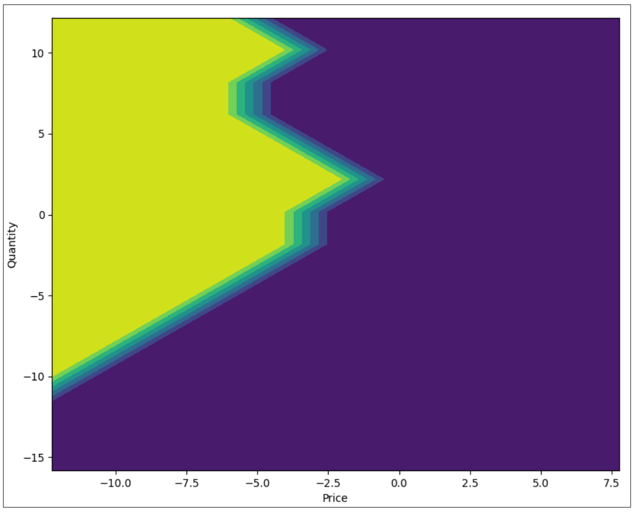


Figure 9 - Projected prediction area on price and volume parameters

The algorithm's delineation aligns with the strategic spoofer model. It categorizes manipulative orders as those with low prices and high volumes. Consequently, these orders have minimal likelihood of execution, yet they inflate the order book volume.

Price - Spread

In the illustrated graph below, the yellow region represents the projected area where the algorithm identifies the price and spread parameters as indicative of a manipulative order. Conversely, the purple region denotes the projected area where the algorithm categorizes the price and spread parameters as characteristic of a non-manipulative order.

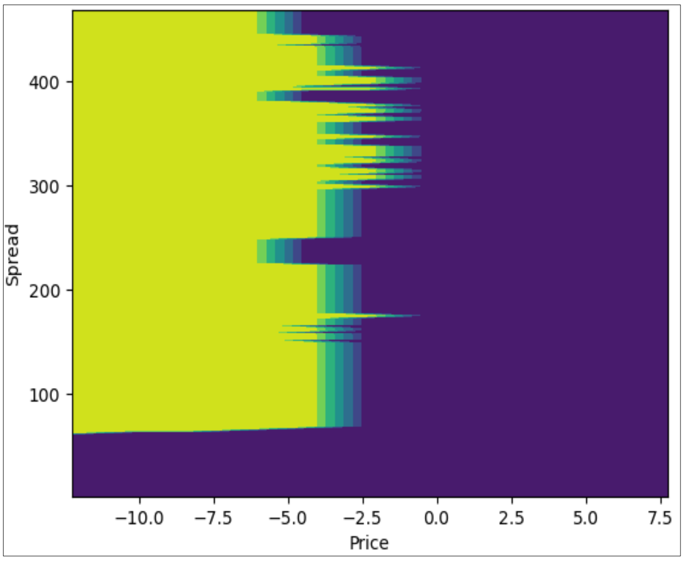


Figure 10 - Projected prediction area on price and spread parameters

The demarcation established by the algorithm aligns with the characteristics of the strategic spoofer model. In the algorithm's assessment, manipulative orders are characterized by low prices and high spreads. The elevated spread serves as an indicator of liquidity, while large volumes are symptomatic of an intent to inflate the order book.

Volume - Spread

In the graph depicted below, the yellow region illustrates the projected area where the algorithm identifies that the volume and spread parameters correspond to a manipulative order. Conversely, the purple region signifies the projected area where the algorithm determines that the volume and spread parameters correspond to a non-manipulative order.

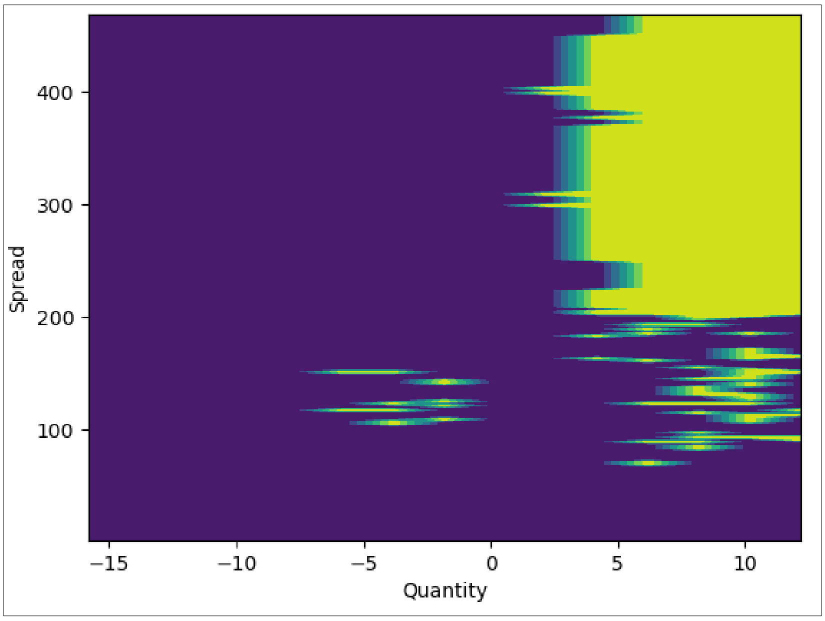


Figure 11 - Projected prediction area on volume and spread parameters

We observe that the prediction zone of the algorithm is in agreement with the data visualized in Figure 4, i.e., spoofer orders with large volumes and spreads higher than the average. The prediction zone realized by the algorithm is therefore consistent with the strategies used by the spoofers.

Detection of manipulative orders on Air France - KLM

In this final section, we apply the models to empirical data, specifically aiming to identify manipulative orders within the Air France - KLM order book sourced from the Euronext exchange. Among the 1,499,445 orders examined, the algorithm detects 39 suspicious orders, accounting for 0.0026% of spoofers in the empirical sample. It is noteworthy that this result is derived from a training database of 2,125,222 simulated orders, with 8,530 orders labeled as manipulative. It's important to emphasize that the spoofer count corresponds to the sum of orders per tick considered as spoofer, and the detection is performed on a per-tick basis rather than by trader. Utilizing the results obtained from the confusion matrix and the model's predictions, we construct the confusion table associated with the tested empirical data.

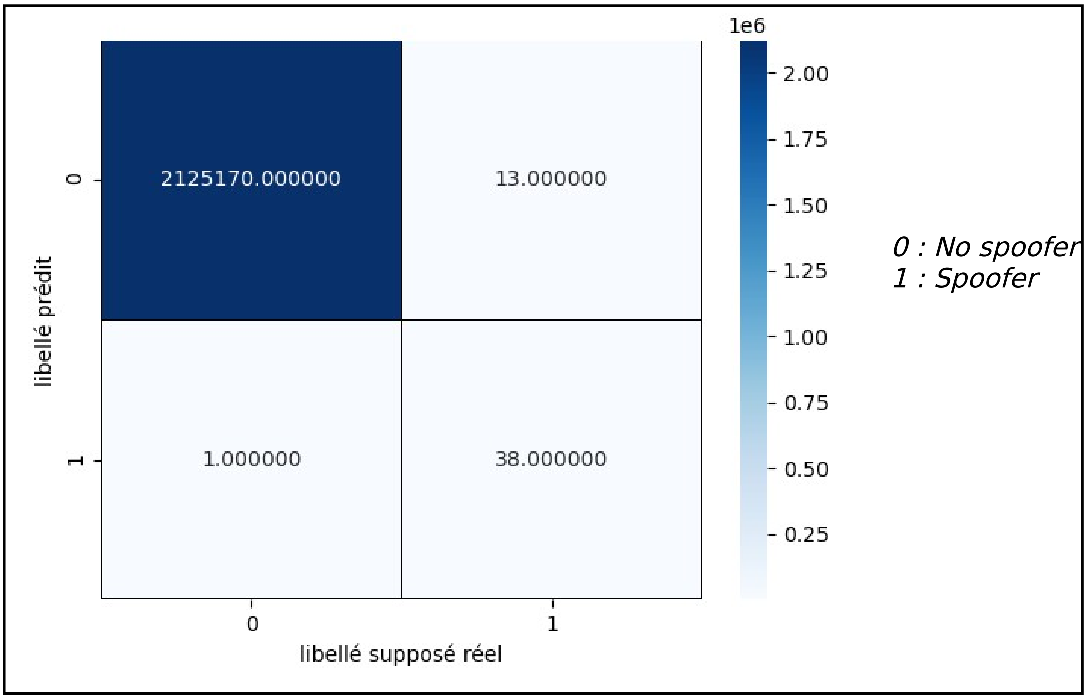


Figure 12 - Confusion matrix applied to the simulation results

In the confusion matrix, we observe a true positive volume of 2,125,170 and a true negative volume of 38. This indicates that the algorithm correctly classified 2,125,170 non-spoofers orders as non-spoofers and 38 spoofer orders as spoofer. Conversely, there is a false positive volume of 13 and a false negative volume of 1, signifying that 13 non-spoofers orders were incorrectly classified as spoofer, and 1 spoofer order was inaccurately classified as non-spoofers.

5. Conclusion

The primary objective of this research is to evaluate the effectiveness of supervised machine learning techniques in identifying spoofing orders. To achieve this goal, we use a database of simulated order books to train the supervised learning model and then apply it to empirical real market data. In particular, unlike previous studies that analyze data on a daily basis (Liu et al., 2021), our approach allows us to detect multiple intraday manipulative episodes with a granularity down to the microsecond level.

First, manipulations are simulated within the framework of an artificial market that faithfully replicates the Euronext architecture, allowing the generation of labeled vectors that characterize manipulative orders. The classification model chosen for this task is the k-nearest neighbors (KNN) method, with an impressive accuracy of 99.375%. The trained algorithm is then evaluated on high-frequency order book information from Euronext Paris, where an average of 100 messages are received every tenth of a second. The sheer volume of orders requires a meticulous processing approach to minimize false positives and especially false negatives. The percentages of false positives and false negatives are exceptionally low at 6x10-4% and 1.3872%, respectively. Although the false positive rate is minimal, approximately 1.4% of manipulative orders remain undetected.

The supervised learning process follows an iterative path, allowing the cyclical reuse of data predicted by the model for retraining. The ultimate goal is to rely solely on training data derived from the algorithm's own predictions. The simulation data is used only for initial training, and this iterative process has the potential to improve model performance while reducing false positives and false negatives. By repeatedly incorporating empirical data labeled by the model, data fidelity increases, restoring the algorithm's high sensitivity to the training data.

One way to further improve the model is to increase the number of parameters in the training and test vectors. A higher number of parameters increases the robustness and precision of the classification algorithm.

Taking the analysis a step further, it becomes interesting to examine the trading activity of different categories of traders at the microsecond level. This approach could reveal trader profiles (pure HFT, mixed HFT, and non-HFT) involved in market manipulation. Such insights are of particular importance to market operators and regulators seeking to instill confidence in the market.

References

Aitken, M. J., Aspris, A., Foley, S., & de B. Harris, F. H. (2018). Market Fairness: The Poor Country Cousin of Market Efficiency. *Journal of Business Ethics*.

Allen, F., & Gale, D. (1992). Stock-price manipulation. *The Review of Financial Studies*, *5*(3), 503–529.

Allen, F., & Gorton, G. (1993). Churning bubbles. *The Review of Economic Studies*, *60*(4), 813–836.

Angel, J. J., & McCabe, D. (2013). Fairness in Financial Markets: The Case of High Frequency Trading. *Journal of Business Ethics*. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84874792183&doi=10.1007%2Fs10551-012-1559-0&partnerID=40&md5=e08a4e995410e3b315fe237b351e8238

Brandouy, O., Mathieu, P. and Veryzhenko, I. (2013). On the design of agentbased artificial stock markets. Communications in Computer and Information Science, 271, 350–364.

Chakraborty, A., & Y\ilmaz, B. (2004). Manipulation in market order models. *Journal of Financial Markets*, *7*(2), 187–206.

Diaz, D., Theodoulidis, B. and Sampaio, P. (2011). "Analysis of stock market manipulations using knowledge discovery techniques applied to intraday trade prices".

Golmohammadi, K., Zaiane, O.R. and Diaz, D. (2014). "Detecting stock market manipulation using supervised learning algorithms".

Heath, E. (2010). Fairness in financial markets. *Finance Ethics: Critical Issues in Theory and Practice (Robert W. Kolb Series)*, 163–178.

Hendershott, Terrence, and Ryan Riordan (2011)." Algorithmic Trading and Information."

Kemme, D. M., McInish, T. H., & Zhang, J. (2022). Market fairness and efficiency: Evidence from the Tokyo Stock Exchange. *Journal of Banking \& Finance*, *134*, 106309.

Kyle, A. S., & Viswanathan, S. (2008). How to define illegal price manipulation. *American Economic Review*, *98*(2), 274–279.

Ladley D., N. Oriol, I. Veryzhenko (2023). "High-frequency spoofing, market fairness and regulation", working paper

Liu Q., C. Wang b , P. Zhang, and K. Zheng, (2021) Detecting stock market manipulation via machine learning: Evidence from China Securities Regulatory Commission punishment cases, *International Review of Financial Analysis*, 78,

Mei, J., Wu, G., & Zhou, C. (2004). Behavior based manipulation: theory and prosecution evidence. *Available at SSRN 457880.*

Öğüt, H., Mete Doğanay, M. and Aktaş, R. (2009). "Detecting stock-price manipulation in an emerging market: The case of Turkey".

SEC - United States Commodities and Futures Trading Commission and Securities and Exchange Commission (2010), "Findings regarding the market events of May 6, 2010," Report of the Staffs of the CFTC and SEC to the Joint Advisory Committee on Emerging Regulatory Issues, September 30, 2010.

Shefrin, H., & Statman, M. (1993). Ethics, fairness and efficiency in financial markets. *Financial Analysts Journal*, *49*(6), 21–29.