1 Introduction

The Foreign Exchange (FX) market, the largest and most liquid financial market globally, facilitates the exchange of currencies among participants with a daily trading volume surpassing $6 trillion. Unlike centralized exchanges, the FX market operates over-the-counter (OTC), meaning trades are conducted directly between parties through electronic platforms without a central exchange.

2 Participants in the FX Market

The FX market comprises various participants, each playing a crucial role. Central banks, such as the Federal Reserve, the European Central Bank, and the Bank of Japan, influence currency prices through monetary policy, interest rate decisions, and currency interventions. Commercial and investment banks are major liquidity providers, executing currency transactions for clients and engaging in speculative trading and risk management. Large multinational corporations use the FX market to hedge currency risks associated with international trade, while hedge funds and speculative traders aim to profit from short-term exchange rate movements, adding liquidity and aiding price discovery. Retail traders, though smaller in volume, contribute to the market's depth due to the proliferation of online trading platforms. Market makers, including financial institutions and firms, ensure continuous trading by offering buy and sell quotes for currency pairs and profit from the bid-ask spread. Additionally, governments and sovereign wealth funds participate in FX transactions for reserves management and international investments.

3 The Role of an FX Market Maker

Market makers are essential in maintaining liquidity by continuously quoting buy and sell prices for currency pairs. Their primary responsibilities include ensuring there is always a market for currencies, thereby keeping the market functioning smoothly, and profiting from the spread between bid and ask prices. They manage risk through careful balance of their books, hedging positions, and analyzing market conditions.

4 Importance of Modeling Client Flow

Modeling client order flow is critical for market makers as it provides insights into the volume and types of transactions made by clients, including speculative trades and hedging activities. Understanding client flow is pivotal for anticipating exchange rate movements, managing risk, and optimizing profitability. For instance, large client demand can influence currency prices, and by predicting such flows, market makers can adjust pricing and manage risk more effectively. Analyzing client flows also sheds light on market sentiment and participant behavior, aiding in strategy adjustment. Furthermore, understanding client flows helps in improving liquidity management, reducing slippage, and enhancing client relationships by offering better pricing and execution.

5 Challenges in Order Flow Modelization and Prediction in FX Markets

Order flow modelization and prediction in FX markets present several challenges. The decentralized nature of the FX market, with its OTC trading and lack of centralized order books, complicates the gathering of granular data essential for detailed analysis. The Request-for-Quote (RFQ) system further limits real-time order book data, adding to the difficulty. Additionally, the diversity of FX participants—ranging from central banks to retail traders—makes modeling more complex compared to equity markets. Large client trades can still affect prices despite the market's depth, and predicting their impact is crucial for effective liquidity management and risk mitigation.

Addressing these challenges involves understanding how different client types influence order flow, predicting market responses to macroeconomic news and geopolitical events, and mitigating risks associated with adverse selection and information asymmetry. Market makers must also navigate the impact of technological advancements, such as algorithmic and high-frequency trading, and leverage machine learning and AI to enhance prediction models. Furthermore, optimizing responses to RFQ systems and developing new methods to track client behavior in a decentralized market are essential for improving order flow models.

In conclusion, while much of the research on order flow and market impact focuses on equities, the FX market's unique characteristics require tailored approaches to understanding and modeling client order flow. Addressing these complexities is crucial for effective risk management, pricing strategy optimization, and liquidity provision in the global FX market.

6 Method and Analysis

6.1 Introduction

The Foreign Exchange (FX) market, with daily trading volumes exceeding $6 trillion, represents one of the largest and most liquid financial markets globally. Analyzing the impact of client trades on price changes in such a vast market presents unique challenges and opportunities. To study these impacts effectively, it is essential to segment clients into meaningful clusters based on their trading behaviors. This segmentation allows us to assess the significance of each cluster's influence on market dynamics. Building on previous research that identifies client co-activities, our objective is to refine our understanding of client behavior through a robust methodology for quantifying and comparing client actions.

6.2 Quantifying Client Activity: The Imbalance Ratio

A critical aspect of our analysis involves defining a quantitative measure to capture client activity, specifically the imbalance ratio. This ratio is calculated as follows:

$$\text{Imbalance Ratio} = \frac{\text{Buy Orders} - \text{Sell Orders}}{\text{Buy Orders} + \text{Sell Orders}} \tag{1}$$

The imbalance ratio reflects the net trading position of a client, indicating whether their trades are predominantly buy or sell orders. High positive values suggest a buying bias, while high negative values indicate a selling bias. This metric is crucial for understanding the intensity and directionality of client trading activity. For instance, in market making, the imbalance ratio helps predict price movements and manage risk, making clients with high imbalance ratios significant for developing effective trading strategies.

6.3 Clustering Client Activities: Methodological Challenges

Once client activity is quantified, the next step is to group clients based on their trading behaviors. Traditional clustering methods, such as k-means, are commonly used for this purpose. However, k-means clustering, which relies on minimizing the within-cluster variance using Euclidean distance, may struggle with capturing complex client activity patterns. The Euclidean distance assumption—that clusters are spherical and of similar size—may not align with the diverse patterns observed in FX trading. For example, clients with correlated trading activities might form distinct clusters not easily captured by k-means. This limitation necessitates the use of alternative clustering approaches that accommodate the nuances of client behavior.

6.4 Advanced Clustering with Statistically Validated Networks (SVNs)

To address the limitations of traditional clustering methods, we employ Statistically Validated Networks (SVNs). SVN techniques leverage statistical validation to identify meaningful clusters of clients based on their trading activities. Unlike k-means, SVN does not rely on rigid assumptions about cluster shapes or distances. Instead, it uses statistical measures to assess the validity and stability of clusters over time. SVN offers flexibility, robustness, and insightfulness, making it adaptable to complex trading patterns and revealing underlying structures in client activities that simpler methods may miss.

6.5 Analyzing Market Impact: Challenges and Considerations

Market impact refers to the effect that trades have on market prices. In the FX market, understanding market impact involves analyzing how client trades influence price movements. While substantial research has been conducted on market impact in equities, studies focusing on FX markets are relatively scarce. Factors such as liquidity and trade volume play a crucial role in market impact. Highly liquid currencies like EUR/USD exhibit smaller price impacts per trade, while less liquid currencies like USD/JPY or GBP/USD may experience more significant price changes. The availability of trade data and volume within the dataset are critical for accurate market impact predictions.

6.6 Predictive Modeling: Tree-Based Models

For predicting market impact and price changes, we utilize Tree-Based Models, specifically LightGBM (LGBM). Tree-Based Models are well-suited for capturing complex, non-linear relationships in data. LGBM, employing gradient boosting techniques, builds an ensemble of decision trees, optimizing performance through advanced algorithms. Its efficiency and scalability in handling large datasets, coupled with its ability to model non-linear relationships, make it an ideal choice for predicting market impacts. Preliminary results indicate that the LGBM model provides a robust framework for analyzing market impact, capturing the intricate dynamics of the FX market.

6.7 Conclusion

This methodology offers a comprehensive approach to understanding client impacts in the FX market. By defining client activity through the imbalance ratio, employing SVN for advanced clustering, and utilizing Tree-Based Models for prediction, we gain deeper insights into market dynamics. This approach overcomes the limitations of traditional methods and leverages advanced statistical and machine learning techniques to deliver actionable insights for market analysis and strategy development.

7 Dataset Analysis

Our analysis leverages an extensive dataset of trades executed by NP during the second semester of 2022, encompassing major global financial markets including the US, European, and Asian markets. This dataset provides a comprehensive view of trading activities across diverse regions and currencies, reflecting the high-volume nature and complexity of the Foreign Exchange (FX) market.

The dataset captures a rich array of trading behaviors, offering valuable insights into market liquidity and client activity. It includes detailed attributes for each trade, such as the timestamp, currency pair, trade size, and trade direction. However, data quality is a concern due to the presence of negligible trades, which constitute a small percentage of the total trades but can skew analysis if not properly addressed. To focus on significant trades, we filter the dataset to include only those above the 1% quantile, ensuring that our analysis concentrates on trades with meaningful impact on market dynamics. This approach is visualized through cumulative distribution functions (CDFs) or threshold-based histograms, which highlight the concentration of impactful trades while filtering out less significant data.

Trade volume and liquidity are key factors in understanding market dynamics. The dataset reveals a substantial number of trades across a wide array of currency pairs, offering insights into different liquidity conditions. By comparing high liquidity pairs such as EUR/USD with lower liquidity pairs like USD/JPY, we can assess the relative trading activity and liquidity of each pair. Graphical representations, such as histograms and kernel density plots, help visualize the trade volume distribution and liquidity metrics, uncovering patterns related to market depth and price stability.

Client activity varies significantly, with certain clients exhibiting notably higher trading volumes and market impact than others. This non-uniform distribution suggests that some clients play a more substantial role in influencing market dynamics. To streamline our analysis, we focus on the top 100 clients based on trade volume. This selection allows for an in-depth examination of the most influential market participants, providing insights into their trading behaviors and their impact on market activity. Bar charts or pie charts illustrating the trade volume distribution among top clients can effectively highlight the key contributors to market activity.

Overall, this dataset provides a robust foundation for analyzing client behavior, liquidity dynamics, and trading impact in the FX market. By concentrating on substantial trades and key market players, we can uncover meaningful patterns and insights into market activities. The next phase of our analysis will involve clustering these top clients to reveal deeper insights into their trading behavior and its impact on market dynamics.

8 Method Application Pipeline

To tackle the challenges associated with order flow modeling, we define the activity of a client i using the imbalance ratio, which is calculated for each trade and aggregated over a specific time period. This approach provides a comprehensive framework for understanding client behavior.

Signed USD Amount Calculation The signed USD amount for a trade is calculated as follows:

$$\text{signedUSDAmount} = \text{Buy} \times \text{USDAmount} - \text{Sell} \times \text{USDAmount} \tag{2}$$

Here:

- Buy and Sell indicate the direction of the trade,
- USDAmount is the amount of USD traded.

State Definition For each time slice $I = [t, t + S]$, the client state a_i is determined based on the signed USD amount:

$$a_i = \begin{cases} 1 & \text{if signedUSDAmount} \geq p \\ 0 & \text{if signedUSDAmount} < p \end{cases} \tag{3}$$

where p is a threshold value that identifies significant trading activity.

Graphical Representation To visualize client activity, plots of client states over time are generated. These plots illustrate the evolution of trading intensity, highlighting periods of high or low activity, and providing insights into trading behavior patterns.

9 Statistically Validated Networks (SVNs)

Statistically Validated Networks (SVNs) offer a sophisticated approach to clustering and network analysis, addressing the limitations of traditional methods like K-Means and

Spectral Clustering. SVN integrates statistical validation to assess the significance of detected network structures, making it particularly suitable for analyzing complex, high-dimensional, and temporal data typical in financial markets.

Network Construction The initial step in SVN involves constructing a network where nodes represent entities (e.g., clients or traders) and edges represent relationships or interactions. Key metrics for network construction include:

- **Imbalance Ratio:** Measures the difference between buy and sell orders:

$$\text{Imbalance Ratio} = \frac{\text{Buy Volume} - \text{Sell Volume}}{\text{Buy Volume} + \text{Sell Volume}} \tag{4}$$

- **Correlation Measures:** Quantify how similarly two clients trade. The Pearson correlation coefficient between clients i and j is calculated as:

$$w_{ij} = \text{corr}(x_i, x_j) \tag{5}$$

Statistical Validation SVNs incorporate rigorous statistical validation to assess the reliability of detected clusters and network structures. This process involves comparing observed network structures with null models and using permutation tests to ensure that identified patterns are statistically significant.

- **Null Models:** Generate random networks with properties similar to the observed network. Common null models include:
  - **Erdos-Renyi Model:** Random graphs with a fixed probability of edge inclusion.
  - **Configuration Model:** Preserves the degree distribution of the observed network while randomly connecting nodes.
- **Permutation Tests:** Involve permuting the data and recalculating network metrics to assess significance. For example, comparing observed metrics with distributions from multiple null networks.
- **Statistical Significance:** Measures such as p-values are used to evaluate if clusters are significantly more cohesive in the observed network compared to null models. Modularity Q is a common measure of cluster significance:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \tag{6}$$

Stability Analysis Assessing the stability of network structures over time is crucial for ensuring that identified clusters reflect persistent patterns rather than transient artifacts. This includes:

- **Temporal Stability:** Examining how network structures change across different time periods. Techniques like dynamic community detection can track evolving clusters.
- **Cross-Validation:** Involves dividing the dataset into multiple folds and clustering each fold to verify the consistency and robustness of detected clusters.

10 Comparative Advantages of SVN

SVNs offer several advantages over traditional clustering methods, especially in financial data analysis:

Handling Complex Data Structures

- **K-Means Clustering:** Effective for well-separated, spherical clusters but struggles with complex structures and varying densities.
- **Spectral Clustering:** Handles non-convex clusters but can be computationally intensive and may face challenges with high-dimensional data.
- **Statistically Validated Networks (SVNs):** Designed to capture complex, non-spherical clusters by integrating statistical validation and handling varying data structures effectively.

Validation and Robustness

- **K-Means Clustering:** Lacks inherent validation and is sensitive to initial conditions and the choice of k.
- **Spectral Clustering:** Provides some validation but lacks a rigorous statistical framework for assessing significance.
- **Statistically Validated Networks (SVNs):** Incorporates rigorous statistical validation methods, including null models and permutation tests, to ensure the reliability of detected structures.

Temporal Dynamics and Stability

- **K-Means Clustering:** Typically static and does not account for temporal dynamics.
- **Spectral Clustering:** Also static and may require additional techniques for time-series analysis.
- **Statistically Validated Networks (SVNs):** Explicitly addresses temporal dynamics through stability analysis and dynamic community detection, making it well-suited for evolving trading behaviors.

11 Conclusion

Statistically Validated Networks (SVNs) provide a robust framework for clustering and network analysis, overcoming many limitations of traditional methods such as K-Means and Spectral Clustering. SVN's focus on statistical validation, complex data handling, and temporal stability makes it a valuable tool for understanding client activities in order flow modeling. Future research should continue to refine SVN techniques and explore their applications in other domains with complex, high-dimensional, and temporal data.

12 Method Application Pipeline

The application of Statistically Validated Networks (SVNs) to client trading imbalance ratio activity involves several key steps: data preprocessing, network generation, projection, link validation, multiple hypothesis testing correction, and detection of under-expressed links. This section outlines these steps in a cohesive manner.

12.1 1. Data Preprocessing and Network Generation

1.1 Calculating Signed USD Amount To quantify the net buying or selling activity of each client, we compute the signed USD amount for each trade as follows:

$$\text{signedUSDAmount} = (\text{Buy} \times \text{USDAmount}) - (\text{Sell} \times \text{USDAmount}), \tag{7}$$

where:

- **Buy** and **Sell** are binary indicators for whether the transaction was a buy or sell,
- **USDAmount** represents the dollar amount of the transaction.

1.2 Defining Client States For each time slice $I = [t, t + S]$, where S is the sampling period, we categorize each client's trading behavior into states based on their signed USD amount. The client's state a_i is defined as:

$$a_i = \begin{cases} 0 & \text{if signedUSDAmount} < p, \\ 1 & \text{if signedUSDAmount} \geq p, \end{cases} \tag{8}$$

where p is a threshold value separating high trading activity from low activity.

1.3 Constructing the Bipartite Network Using the defined client states, we construct a bipartite network $G = (V, E)$ where:

- One set of nodes V_A represents the clients,
- The other set V_B represents the trading states,
- An edge between a client $v_i \in V_A$ and a state $v_j \in V_B$ indicates that client v_i was in state v_j during the period.

12.2 2. Projection and Link Validation

2.1 Projecting the Network To analyze client interactions more directly, we project the bipartite network onto V_A, creating a projected network G'. This network contains edges between clients who share common trading states, reflecting their co-occurrence in specific activities.

2.2 Statistical Validation of Links To validate the links in the projected network, we follow these steps:

2.2.1 Null Hypothesis Assuming that connections between clients are random while preserving the degree distribution, we model the probability p of two clients i and j sharing X common trading states using the hypergeometric distribution:

$$H(X \mid N_B, N_i, N_j) = \frac{\binom{N_i}{X}\binom{N_B - N_i}{N_j - X}}{\binom{N_B}{N_j}}, \tag{9}$$

where:

- N_B is the total number of states in V_B,
- $\binom{\cdot}{\cdot}$ denotes the binomial coefficient.

2.2.2 p-Value Calculation The p-value for observing $N_{i,j}$ or more common states between clients i and j is computed as:

$$p(N_{i,j}) = 1 - \sum_{X=0}^{N_{i,j}-1} H(X \mid N_B, N_i, N_j). \tag{10}$$

This p-value indicates the likelihood of observing such a number of common states by chance.

12.3 3. Multiple Hypothesis Testing Correction

To address the numerous potential client pairs, we apply corrections for multiple hypothesis testing to control false positives. Two commonly used methods are:

3.1 Bonferroni Correction The Bonferroni correction adjusts the significance level α by dividing it by the number of tests m. The Bonferroni-corrected threshold is:

$$\text{Bonferroni threshold} = \frac{\alpha}{m}. \tag{11}$$

This method reduces the risk of Type I errors but can increase Type II errors, especially with a large number of tests.

3.2 False Discovery Rate (FDR) The FDR approach, such as the Benjamini-Hochberg procedure, controls the expected proportion of false positives among declared significant results. The procedure involves:

1. **Sorting p-values:** Arrange all p-values in ascending order.
2. **Calculating the FDR threshold:** Determine the threshold for significance using:

   $$\text{FDR threshold} = \frac{i}{m} \times Q, \tag{12}$$

   where i is the rank of the p-value, m is the total number of tests, and Q is the desired FDR level.
3. **Selecting Significant Edges:** Retain edges with p-values below the FDR threshold.

The FDR method offers a balance between false positives and false negatives, making it less conservative than Bonferroni.

12.4 4. Detection of Under-Expressed Links

SVNs also identify under-expressed links, where the observed number of common states is significantly lower than expected. The p-value for under-expression is calculated as:

$$p(N_{i,j}) = \sum_{X=N_{i,j}}^{N_B} H(X \mid N_B, N_i, N_j). \tag{13}$$

This approach reveals less frequent or avoided relationships, providing additional insights into client interactions.

13 Conclusion

The application of Statistically Validated Networks to imbalance ratio activity enables the generation and validation of a network that highlights significant interactions and under-expressed relationships. This methodology facilitates a nuanced understanding of trading behaviors, supporting more informed decision-making and strategy development.

To evaluate the effectiveness of clustering methods used in client interaction studies, two key approaches can be employed: Uniform Manifold Approximation and Projection (UMAP) for visualization and Adjusted Rand Index (ARI) for numerical assessment. This section details the application of each method, including their mathematical foundations, practical implementation, and their role in comparing clustering results from different methodologies, such as the Statistically Validated Network (SVN) pipeline, client category clustering, client location clustering, and client business type clustering.

14 UMAP for Visualization

UMAP (Uniform Manifold Approximation and Projection) is a powerful technique for reducing high-dimensional data to lower dimensions, typically 2D or 3D, to facilitate visualization. This technique helps assess the quality of clustering results by projecting complex relationships into a comprehensible visual format, thereby enabling the evaluation of cluster separation and grouping based on interaction patterns.

14.1 Mathematical Foundation of UMAP

UMAP operates by constructing a graph in the high-dimensional space where each data point (client) is connected to its nearest neighbors. The primary steps in UMAP are:

14.1.1 Constructing the High-Dimensional Graph

UMAP starts with a k-nearest neighbors algorithm to create a graph, where each point is linked to its closest neighbors in the original feature space. The edge weights w_{ij} between points x_i and x_j are computed as:

$$w_{ij} = \exp\left(-\frac{d(x_i, x_j)^2}{2\sigma_i^2}\right)$$

Here, $d(x_i, x_j)$ is the distance between points x_i and x_j, and σ_i is a parameter that controls the neighborhood spread for point x_i.

14.1.2 Optimization in Low-Dimensional Space

UMAP then seeks a lower-dimensional embedding by minimizing the discrepancy between distances in the high-dimensional space and the low-dimensional space. This is achieved through a loss function that resembles cross-entropy:

$$C = \sum_{(i,j)} w_{ij} \log \left(\frac{w_{ij}}{w'_{ij}} \right)$$

where w'_{ij} denotes the edge weights in the low-dimensional projection.

14.2 Applying UMAP to Client Interaction Clustering

When applied to client interaction data, UMAP projects the complex relationships derived from the SVN pipeline into 2D or 3D space. This visualization helps to assess the clarity and separation of clusters, revealing how well clients are grouped based on their trading behaviors or interaction patterns.

15 Adjusted Rand Index (ARI) for Numerical Evaluation

While UMAP provides a visual representation of clusters, the Adjusted Rand Index (ARI) offers a numerical measure to compare different clustering results. ARI assesses the similarity between clusterings while accounting for chance groupings, making it a robust metric for evaluating clustering effectiveness.

15.1 Mathematical Foundation of ARI

ARI is a modification of the Rand Index, which measures how consistently pairs of points are assigned to the same or different clusters in two different clusterings. For clusterings C (ground truth) and C' (from the SVN pipeline), ARI is computed using the following definitions:

- a: the number of pairs of points in the same cluster in both C and C',
- b: the number of pairs of points in different clusters in both C and C',
- c: the number of pairs of points in the same cluster in C but in different clusters in C',
- d: the number of pairs of points in different clusters in C but in the same cluster in C'.

The Rand Index (RI) is calculated as:

$$RI = \frac{a + b}{a + b + c + d}$$

To account for chance agreement, ARI adjusts the RI:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]}$$

where $E[RI]$ is the expected Rand Index for random clustering, and $\max(RI)$ is the maximum possible Rand Index. ARI values are normalized between -1 and 1:

- ARI = 1 indicates perfect agreement between the two clusterings,
- ARI = 0 indicates that the clustering is no better than random,
- ARI ¡ 0 suggests a clustering result worse than random, though this is rare.

16 Conclusion

UMAP and ARI are complementary tools for evaluating clustering methods. UMAP provides intuitive visual insights into clustering structure, while ARI offers a rigorous numerical assessment. Together, these methods allow for a comprehensive evaluation of clustering effectiveness in client interaction studies, supporting the validation of clustering results from the SVN pipeline against alternative clusterings.

17 Adjusted Rand Index (ARI) for Clustering Comparison

17.1 Example of ARI Calculation

To illustrate the application of the Adjusted Rand Index (ARI), consider a scenario where 100 clients have been clustered using the SVN pipeline. If this pipeline groups clients according to geographic proximity, we expect a high ARI score when comparing these clusters with a known client location clustering. A high ARI indicates that the SVN clustering aligns well with the location-based clustering, demonstrating that the clustering method effectively captures geographic similarities.

17.2 Comparing SVN Pipeline with Various Clustering Schemes Using ARI

The ARI metric is a valuable tool for comparing the clustering results obtained from the SVN pipeline with various predefined clustering schemes, such as client categories, locations, and business types. By evaluating the ARI scores, we can assess the level of agreement between the SVN clusters and these predefined categories. A high ARI value signifies strong alignment between the SVN-generated clusters and the known categories, indicating that the SVN pipeline effectively reflects the underlying structure present in the data.

18 Conclusion

In summary, integrating UMAP for visualization with the Adjusted Rand Index (ARI) for quantitative evaluation provides a robust framework for assessing clustering performance in client interaction studies. UMAP offers intuitive visual insights into client groupings, while ARI delivers a rigorous, numerical measure of clustering quality. By leveraging ARI to compare SVN clustering results with alternative clusterings based on client categories, locations, or business types, we gain a deeper understanding of client interactions and refine our clustering methods to better capture meaningful relationships in the data.

19 Introduction

In financial markets, understanding the dynamics of price variation in relation to order flow—the net volume of buy and sell orders executed by clients—is crucial for predicting price movements. This paper explores this relationship through a LightGBM (LGBM) regression model, aiming to quantify how client order flow influences price variations.

20 Price Variation as a Percentage Change

To measure price movements, we define price variation as the percentage change between two time periods. The formula is:

$$\text{Price Variation}_{t+1} = \frac{p_{t+1} - p_t}{p_t}, \tag{14}$$

where:

- p_t is the asset price at time t,
- p_{t+1} is the asset price at time $t + 1$.

This formula captures the relative change in price, providing insight into upward or downward movements in percentage terms.

21 Order Flow as a Predictor of Price Variation

Order flow, representing the net buying or selling volume from clients, plays a significant role in price evolution. The total order flow at time t is calculated as:

$$\text{order\_flow}_t = \sum_{i=1}^{N} q_{i,t}, \tag{15}$$

where:

- N is the number of clients,
- $q_{i,t}$ is the trade volume of client i at time t (positive for buy orders, negative for sell orders).

21.1 Hypothesized Relationship Between Order Flow and Price Variation

We hypothesize that order flow at time t influences price variation at time $t+1$. Specifically:

$$\text{order\_flow}_t \sim \text{price\_variation}_{t+1}. \tag{16}$$

This implies that the price variation at time $t+1$ is influenced by the order flow at time t, as imbalances in buy/sell orders lead to price adjustments.

22 LGBM Regression for Modeling Price Variation

We use a LightGBM regression model to explore the relationship between client order flow and price variation. LightGBM, a gradient-boosting framework, builds decision trees iteratively to correct prediction errors.

22.1 Objective Function: Mean Squared Error

The Mean Squared Error (MSE) is used as the objective function, measuring the average squared difference between predicted and actual values:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2, \tag{17}$$

where:

- y_i is the actual price variation at time $t+1$,
- $\hat{y}_i$ is the predicted price variation,
- N is the total number of time steps.

LightGBM minimizes MSE by iteratively improving the predictions through decision trees.

22.2 Model Evaluation Metrics

22.2.1 R-Squared: Coefficient of Determination

The R^2 metric assesses the proportion of variance in price variation explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \bar{y}\right)^2}, \tag{18}$$

where $\bar{y}$ is the mean of the actual values y_i. An R^2 value close to 1 indicates strong explanatory power.

22.2.2 Pearson Correlation

The Pearson correlation coefficient ρ measures the linear relationship between predicted and actual price variations:

$$\rho_{y,\hat{y}} = \frac{\mathrm{Cov}(y, \hat{y})}{\sigma_y \sigma_{\hat{y}}}, \tag{19}$$

where:

- $\mathrm{Cov}(y, \hat{y})$ is the covariance between actual and predicted values,
- σ_y and $\sigma_{\hat{y}}$ are the standard deviations of actual and predicted values, respectively.

A Pearson correlation close to 1 indicates a strong positive linear relationship.

23 Order Flow as Input Features for LGBM

The input features for the LGBM model include the order flows of different client clusters. For cluster i at time t, the order flow is:

$$\text{order\_flow}_{t,i} = \sum_{j=1}^{N_i} q_{j,t}, \tag{20}$$

where N_i is the number of clients in cluster i. The feature vector at time t is:

$$\mathbf{x}_t = \left(\text{order\_flow}_{t,1}, \text{order\_flow}_{t,2}, \ldots, \text{order\_flow}_{t,K}\right), \tag{21}$$

where K is the total number of clusters. The target variable is $\text{price\_variation}_{t+1}$.

24 Handling Differences in Distribution

Order flow often exhibits a skewed distribution, while price variation tends to follow a more continuous distribution. The LGBM model effectively learns non-linear relationships between these differently distributed variables.

25 Choice of LGBM over Other Models

Linear models, which assume a constant linear relationship, are inadequate for capturing the complexities of financial markets. Neural networks, although flexible, may underperform on tabular data and risk overfitting. LGBM offers a balance by handling non-linearity efficiently and performing well on tabular data without the pitfalls associated with neural networks.

26 Conclusion

The LightGBM model provides a robust framework for understanding the relationship between client order flow and price variation. It captures non-linearities, handles distributional differences, and performs efficiently on large datasets. Evaluation metrics such as R^2, MSE, and Pearson correlation ensure that the model delivers valuable insights into price movements.

27 Analysis of Client-Specific Trade Data for Price Variation Prediction

27.1 Introduction

Data scarcity can limit the effectiveness of predictive models. To address this, we propose a novel approach that treats each client's trade as an individual data point, significantly increasing the sample size. This method enhances the model's predictive accuracy by leveraging the collective trading activity of all clients, even when specific clients are inactive.

27.2 Problem Statement

The limited amount of data constrains traditional models' performance. By treating each client's trade as a separate data point, we increase the sample size, improving the model's robustness and predictive power.

27.3 Model Framework

Our model aims to predict future price variation based on individual client trades during a sampling period. The framework is as follows:

- **Client Trade Data Collection:** Each trade is treated as an individual data point, linked to the price variation observed in the subsequent period.
- **Target Variable:** For each trade, the target is the price variation at the end of the next sampling period.
- **Simultaneous Predictions:** We predict future price variation simultaneously for all clients using trades from the same period.

Mathematically:

Let $q_{i,t}$ be the trade volume of client i at time t, and p_{t+1} be the price at $t+1$. The price variation is:

$$\text{price\_variation}_{t+1} = \frac{p_{t+1} - p_t}{p_t}. \tag{22}$$

The model aims to predict $\text{price\_variation}_{t+1}$ using the features derived from client trades.

27.4 Implications and Future Work

By leveraging trades from all clients, the model can better handle periods of inactivity and capture the overall market dynamics. Future work will focus on refining the model to account for varying client influences and exploring additional features that could enhance predictive accuracy.

28 Conclusion

This report has explored various methods for analyzing and predicting price variation based on client order flow. The integration of UMAP for visualization and ARI for evaluation provides a comprehensive approach to understanding clustering results. The application of LightGBM for regression analysis demonstrates its effectiveness in capturing non-linear relationships between order flow and price variation. Future enhancements will focus on addressing data limitations and improving predictive accuracy through advanced modeling techniques.