UMEÅ UNIVERSITY

Evaluating clustering algorithms with human-in-the-loop for identifying trader firm groupings

Machine Learning and Human Insight in Trader Firm Clustering

Yun Nyan Liu

Yun Nyan Liu
Spring 2025
Degree Project in Computing Science and Engineering, 30 credits
Supervisor: Esteban Guerrero Rosero
Extern Supervisor: Anders Jönsson
Examiner: Henrik Björklund
Master of Science Programme in Computing Science and Engineering, 300 credits

Abstract

Financial markets generate vast amounts of data, making it increasingly challenging for analysts to identify meaningful patterns and insights. This thesis explores clustering techniques for categorizing trader firms within the Nordic equity markets, incorporating a **human-in-the-loop** approach to enhance interpretability. Using order book data from Nasdaq Nordic, trader behavior is analyzed, and **k-means**, **hierarchical**, and **spectral** clustering algorithms are applied to classify firms based on their trading patterns.

Due to the lack of ground truth, clustering effectiveness is assessed through intrinsic evaluation metrics and expert validation, revealing a discrepancy between intrinsic measures and expert assessments. The results underscore the importance of **human-in-the-loop** approach and demonstrate the potential of unsupervised learning to support market analysis. This research bridges machine learning methodologies with domain expertise and provides a foundation for future clustering-based analytics in financial markets.

Keywords: Clustering · Unsupervised learning · Trader behavior · Financial markets · Human-in-the-loop

Acknowledgements

First and foremost, I would like to express my profound gratitude to all who supported me throughout the completion of this Master's thesis.

I extend my sincere appreciation to my academic supervisor, **Esteban Guerrero Rosero**, whose positive attitude, insightful ideas, and consistent encouragement after each meeting provided the motivation and energy I needed to persevere to complete this thesis.

I am deeply grateful to Nasdaq for providing the opportunity to conduct this project, along with the essential data and resources that were instrumental to my work. The experience of working within such a large organization has given me valuable experience beyond the scope of this thesis.

Special thanks to my supervisor at Nasdaq, **Anders Jönsson**, whose time, guidance, and support helped me develop a deeper understanding of the project's underlying challenges. I would also like to thank **Magnus Kling** and **Vasko Jovanovski** for their invaluable assistance in helping me interpret the order book data.

I am particularly indebted to **Satchit Sagade**, whose financial expertise provided critical insights and whose guidance was instrumental to this research.

Finally, I owe my deepest gratitude to family and friends for their constant support and encouragement throughout my academic journey.

Contents

1 Introduction

Financial markets today are fast-paced with an abundance of financial datasets [1][2]. Advancements in storage technology have made it faster and cheaper to store large amounts of data. However, the sheer volume of these data sources makes it difficult for human analysts to identify valuable insights or patterns. Consequently, effectively extracting valuable insights from the datasets is becoming increasingly challenging [2].

Accurate market data are essential for traders, as they provide valuable market insights and enable informed decision making [1][3]. Market data enables various groups such as trading participants, investors, index providers to evaluate the value of different assets. It also provides access to historical prices, which is valuable for developing future strategies [3][4].

Nasdaq Data Link is a leading provider of financial, economic, and alternative datasets trusted by analysts at top hedge funds, asset managers, and investment banks [5]. One such offering is the Investor Activity Indicator (IAI) datasets, which offer valuable insights into the factors driving market activity in the Nordic and Baltic regions. They help investors understand sentiment and activity trends across key market segments, enabling more informed trading decisions [6] and only datasets proven to deliver valuable insights are offered as products on the platform [5].

Although the use of machine learning in financial markets has been extensively studied, research has predominantly focused on forecasting returns for individual assets [7] [8] and detecting illegal trading activities as explored in [9]. Consequently, numerous other challenges within the financial sector have been largely overlooked or ignored [7]. For example, predicting financial crises [10], assessing credit risk [11], and segmentation of customers for targeted marketing activities [2]. This study aims to explore clustering techniques for financial data analysis, incorporating a human-in-the-loop approach.

1.1 Purpose

This project, conducted in cooperation with Nasdaq, aims to provide deeper insights into category flows within financial markets using clustering algorithms. The goal of this project is to identify if and how trader firm categories can be singled out of the larger Nordic Equity trading data sets. Specifically, the aim is to explore clustering algorithms for trader firm groupings to determine if any patterns can be found. The data sets consist order book data, which contains buy and sell orders for specific securities or financial instruments [12]. This task presents a unique challenge due to the nature of the data, which includes big data with time-series and mixed data, as well as dynamic trader behavior.

The core problem is to determine whether distinct trader groups with specific characteristics or behaviors can be identified within order book and trade data. The hypothesis is that there

are distinct trader groups that exhibit unique trading behaviors. Given $T = \langle x_1, x_2, \ldots, x_n \rangle$, where T represents the entire set of traders and x_n represents a key characteristic, it is possible to identify trader subgroups such that $T_a, T_b, \ldots, T_m \subseteq T$. The key characteristics $(x_1, x_2, \ldots, x_n)$ defining these unique behaviors are further detailed in Section 4.1.3. Therefore, understanding the complexities of the data is crucial for effectively applying clustering techniques. Ultimately, the end goal with the findings is to present data on volumes, prices, and turnover across these identified categories, with the goal of offering it as a new IAI dataset.

1.2 Problem formulation

The problem this project aims to solve is determining whether clustering algorithms can be used to analyze order book and trade-data to categorize trading firms into different categories. Additionally, the project will compare the performance of the difference clustering algorithms with domain expert evaluation. Thus, this project aims to answer the following questions:

1. How do different clustering algorithms perform in categorizing trading firms based on the intrinsic cluster metrics derived from clustering of order book and trade-data?
2. Which clustering algorithm most accurately identifies trading firm categories based on domain expert evaluations?

1.3 Structure of the thesis

The structure of this thesis is organized as follows. Chapter 1 introduces the background, motivation, objectives, and research aims of the project. Chapter 2 provides central information regarding the financial domain and data clustering relevant to this study. Chapter 3 reviews and examines prior work in data clustering, focusing on finance applications. Chapter 4 provides an overview of the data, outlines the limitations, and describes the clustering methods and evaluation techniques used to address the research questions. Chapter 5 presents the results obtained from applying the different cluster algorithms and the results from the qualitative analysis. Chapter 6 discusses the methodology and results; it also discusses potential improvements for future work. Finally, Chapter 7 summarizes the key contributions and provides concluding remarks.

2 Background

This chapter provides the background information necessary to understand the purpose and goals of this thesis. It presents a brief overview of clustering and various clustering techniques, along with an overview of the relevant financial domain concepts.

2.1 Financial domain

This section provides a brief overview of some financial terms and concepts.

Liquidity

Liquidity in finance refers to how quickly and easily an asset or security can be converted into cash without causing a significant change in its market price [13].

Order book

An order book is an electronic list of buy and sell orders for a specific security or financial instrument. Shows the number of shares being bid on or offered at each price point. The order book helps traders make informed decisions by providing valuable trading information, such as price, availability, and who initiates transactions [12] [14].

Market maker

A market maker maintains firm bid and ask prices in a given security by standing ready to buy or sell at publicly quoted prices, which helps trades occur smoothly. Market makers profit from the spread. The spread is the difference between the bid and ask prices [15].

Liquidity provider

Liquidity providers, often the largest banks or financial institutions, play a crucial role in maintaining market stability and efficiency by ensuring that there is sufficient liquidity. They provide the necessary funds to execute trades in financial markets. Although they function similarly to market makers, there are some key distinctions. Liquidity providers act as intermediaries between buyers and sellers and typically do not maintain an inventory of assets [16] [17].

Retail trader

Retail traders invest their capital for personal gain, rather than trading on behalf of a company or institution [18]. In contrast, institutional traders operate on behalf of another group or institution. The key characteristics of retail traders are that they typically invest in assets such as stocks, bonds, options, and futures but typically trade in smaller volumes compared to institutional traders [19].

Algo-trading firm

An algo-trading firm is a company that uses computer algorithms, advanced mathematical models, and high-frequency trading techniques to execute trades at high speeds and volumes in financial markets. These firms rely on automated trading strategies to make decisions based on market data, price, timing, and volume [20].

Proprietary trader

Proprietary trading occurs when financial institutions, such as investment banks and hedge funds, trade financial instruments using their capital rather than client funds. Unlike traditional brokerage activities, where firms earn commissions from client transactions, proprietary trading enables institutions to retain all the profits generated from their investments [21].

2.2 Time series data

Time series data are temporal data in which each value represents one or more observations recorded chronologically. In other words, a time series is a sequence of ordered data points, denoted as $T = \langle t_1, t_2, \ldots, t_d \rangle$, where $T \in \mathbb{R}^d$ consists of observations with d real values, where $d \in \mathbb{N}^+$ [22] [23]. A multivariate time series is a set of $n \in \mathbb{N}^+$ time series $D = \{T_1, T_2, \ldots, T_n\}$ [22].

2.3 Data clustering

The task of clustering is to partition a set of unlabeled data points into groups that are as similar as possible. Clustering is an unsupervised machine-learning algorithm and a common technique in data mining. There are numerous applications for clustering, such as summarization, learning, segmentation and target marketing [24] [23] [25]. Additionally, clustering can uncover dataset patterns or identify whether new samples belong to the same dataset or are outliers [26].

2.3.1 Clustering algorithms

This section outlines some standard clustering algorithms. Generally, clustering can broadly classified into six groups: Partitioning, Hierarchical, Grid-based, Model-based, Density-based clustering and Multi-step clustering algorithms [23].

However, the two most popular clustering algorithms are partitional and hierarchical clustering due to their simplicity, ease of implementation and a wide range of applicability [24].

K-means clustering

The k-means clustering algorithm is a popular partitional clustering algorithm and can be observed in Algorithm 1. The algorithm begins by selecting initial representative points as K initial centroids. Each data point is assigned to the nearest centroid according to a chosen proximity measure. Centroids are then updated for each cluster and repeated until convergence [24]. The main idea of k-means is to minimize the total distance (typically Euclidean distance) between all objects in a cluster from their cluster center [23]. The known drawbacks of k-means are the initialization of the algorithm and estimating the number of K clusters [24].

Algorithm 1 K-Means clustering

Input: N Objects $\{X_1, X_2, \ldots, X_N \in \mathbb{R}^d\}$, K
Output: K clusters $C = \{C_1, C_2, \ldots, C_K\}$

1: Select K points as initial centroids.
2: **repeat**
3: Form K clusters by assigning each point to its closest centroid
4: Recompute centroid of each cluster
5: **until** convergence criterion met

Hierarchical clustering

Hierarchical clustering addresses the clustering problem by constructing a binary tree-based data structure known as a dendrogram. This dendrogram represents a hierarchy of clusters. Initially, each data object is treated as it own cluster. The algorithm iteratively merges the most similar pairs of clusters. Clusters can be merged using various methods such as single linkage, which merges based on shortest distance among all their members, complete linkage, which merges based on the shortest maximum distance, and average linkage, which merges based on the average distance between all members of the clusters [24].

Algorithm 2 Agglomerative Hierarchical clustering

Input: N Objects $\{X_1, X_2, \ldots, X_N \in \mathbb{R}^d\}$
Output: K clusters $C = \{C_1, C_2, \ldots, C_K\}$

1: Compute the dissimilarity matrix between all data points.
2: **repeat**
3: Merge clusters as $C_{a\cup b} = C_a \cup C_b$. set new cluster's cardinality as $N_{a\cup b} = N_a + N_b$.
4: Insert a new row and column containing the distances between the new cluster $C_{a\cup b}$ and the remaining clusters.
5: **until** Only one maximal cluster remains.

Spectral clustering

Unlike traditional clustering methods such as k-means, spectral clustering can handle more complex scenarios and arbitrarily shaped nonlinear clusters. It does not rely on assumptions about the underlying cluster structure. This is achieved by embedding the data in a lower-dimensional representational space where the cluster structure becomes more evident [24].

Algorithm 3 Spectral clustering

Input: N Objects $\{X_1, X_2, \ldots, X_N \in \mathbb{R}^d\}$
Output: K clusters $C = \{C_1, C_2, \ldots, C_K\}$

1: Construct Similarity Graph
2: Let $\mathcal{W}$ be the weighted adjacency matrix
3: Calculate normalized Laplacian L_s
4: Compute first k eigenvectors $l_1, l_2, \ldots, l_k$ of L_s
5: Let $U \in \mathbb{R}^{n \times k}$ be the matrix having vectors $l_1, l_2, \ldots, l_k$ as columns
6: Normalize rows of U to norm 1 to obtain matrix $T \in \mathbb{R}^{n \times k}$, i.e., set $t_{ij} = u_{ij}/(\sum_k u_{ik}^2)^{1/2}$
7: Let $y_i \in \mathbb{R}^k$ be the ith row of U for $i = 1, 2, \ldots, n$
8: Cluster points y_i for $i = 1, 2, \ldots, n$ into k clusters $C_1, C_2, \ldots, C_k$ using k-means algorithm

2.4 Similarity measures

Distances between data points are crucial to creating clusters and serve as a standard for evaluating cluster performance. This concept involves numerical measures of dissimilarity or similarity between individual data points, which are often calculated based on data features [27].

2.4.1 Euclidean

Euclidean distance, also known as ℓ_2 norm quantifies the straight-line geometric distance between two vectors P, Q in d-dimensional space [24]. Derived from the Pythagorean theorem, Euclidean distance is a widely used measure for numerical attributes or features within Euclidean space defined as [28]:

$$d_{L2}(P,Q) = \sqrt{\sum_{i=1}^{d} (P_i - Q_i)^2} \tag{2.1}$$

2.4.2 Cosine similarity

Consider two vectors P, Q, then Cosine similarity measures the cosine of the angle between two vectors defined using an inner product and is defined as [24]:

$$cos(P,Q) = \frac{P \cdot Q}{\|P\| \|Q\|} \tag{2.2}$$

2.5 Dimension reduction

Dimension reduction tools help handle high-dimensional data, visualize data, and provide an understanding of cluster structure by transforming high-dimensional data into a meaningful representation of reduced dimensionality. However, dimension reduction can also be misleading because it may represent cluster structures that were not present in the original data, i.e., data points are further or closer to each other in the projected space but not so in the original space [29] [30].

Principal component analysis

Principal Components Analysis (PCA) is a standard linear technique for dimensionality reduction. PCA performs the reduction by embedding data into a linear subspace of lower dimensionality [30]. Therefore, the objective of PCA is to learn low-dimensional representation $z \in \mathbb{R}^q$ of the data $x \in \mathbb{R}^p$ where $q < p$ and the goal is to retain as much information as possible where information is measured in variance [31]. Thus, PCA preserves the dimensions with the most variation in the data, preserving the global structure of the high-dimensional data [32].

T-distributed Stochastic Neighbor Embedding

T-distributed Stochastic Neighbor embedding (t-SNE) is used to visualize high-dimensional data. It converts similarities between data points into joint probabilities. It aims to minimize the Kullback-Leibler divergence between these joint probabilities in the low-dimensional embedding and the original high-dimensional data [33]. t-SNE is a dimensionality reduction technique that preserves the local structure of the data. Specifically, it focuses on maintaining relationships between nearby points, ensuring that close observations in the high-dimensional space remain close in the low-dimensional representation [32].

2.6 Logistic regression

Logistic regression is a supervised machine learning algorithm for modeling conditional class probabilities. Logistic regression can be interpreted as an adaptation of a linear regression model tailored for classification tasks rather than regression. Consider a linear regression given by [31]:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p \tag{2.3}$$

The key idea of logistic regression is to squeeze z from 2.3 into a restricted interval of [0, 1], which can therefore be interpreted as a probability. This can be done by using the logistic function of $h(z) = \frac{e^z}{1+e^z}$ resulting in [31]:

$$P(y = 1) = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}} \tag{2.4}$$

Generalization

Generalization measures a machine learning model's effectiveness in predicting new unseen data beyond training data. Two challenges to generalization are overfitting and underfitting. Overfitting occurs when the model excessively learns the training data, including the noise and outliers. The "Curse of Dimensionality" refers to problems created by high-dimensional data; An overfitting problem is an example. Dimensionality reduction methods, described in Section 2.5 that map the original data on smaller dimensions, are therefore less prone to overfitting. Conversely, underfitting happens when the model is overly simplistic, failing to predict both the training data and new data accurately [34]. Overfitting and underfitting are illustrated in Figure 1.

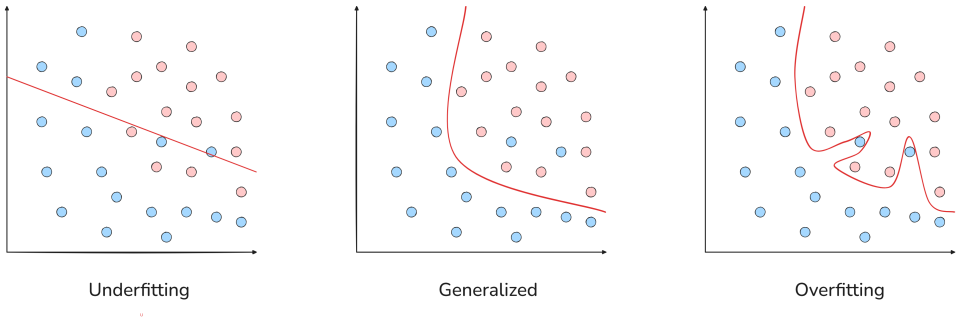


Figure 1: Illustration of underfitting, overfitting and generalized.

Regularization can be applied to improve model generalization. Regularization tries to limit the effect of useless features by adding a "penalty term" called regularizer [35]. Selecting the optimal model can be achieved through cross-validation and grid search. Cross-validation provides a simple and effective model selection and performance evaluation method, while hyperparameter optimization relies on grid search. Grid search evaluates a predefined subset of the hyperparameters space of the training algorithm [36] [37].

A widely used cross-validation technique is k-fold cross-validation. This method randomly divides the dataset into k equally sized subsets. Each subset serves as the validation set once, while the remaining k-1 subsets are used for training. This process repeats until every subset has been used for validation. The model's performance is then averaged across all iterations, which helps to identify the optimal parameters [36].

3 Related work

As mentioned in Chapter 1, while extensive research has been conducted on machine learning applications in financial markets, many areas have been largely overlooked or underexplored. Nonetheless, some studies have been related to the application of clustering within the financial domain, including trader classification and time-series clustering.

Several surveys have reviewed the application of clustering in finance. For example, Cai et al. [2] examine a variety of clustering algorithms to analyze different financial data sets that range from time series to transactions. They conclude that density-based clustering does not suit financial datasets, and their work represents a first step in clustering financial datasets.

Another example is the review by Tang et al. [38], which focuses on clustering methods in financial applications. Tang et al. review standard clustering techniques in finance, focusing primarily on popular topics such as credit scoring, trading strategy, portfolio analysis, and the stock market. They conclude that clustering applications can still be more widely applied and that further development of evaluation techniques is needed, which this thesis will address by exploring trader groupings and domain expert evaluation.

Previous work by Millberg [39] evaluated various clustering techniques on financial time-series data. Similarly, Tangsirisakul [40] conducted a comparative analysis of clustering techniques. However, these studies focused exclusively on stock data, whereas this thesis aims to provide deeper insights into trader distinctions.

The comparison of cluster techniques is not a novel area of research. For instance, Yin et al. [27] conducted a comprehensive review of existing clustering algorithms. As they noted, there has been a shift in the focus of clustering algorithm research from solely improving the underlying algorithms to more targeted applications in specific fields. The increasing recognition of the diverse and complex data challenges in various domains drives this shift.

This thesis builds upon previous work related to the clustering of traders. Mankad et al. [41] proposed a dynamic machine-learning approach to classify traders into five distinct groups. Wright et al. [42] introduced machine learning methods for clustering active traders for intraday classification. Ruan et al. [43] classified short-term strategies and identified clusters of market makers and directional traders. Cont et al. [44] provided a detailed representation of the order book and analyzed client order flow, segmenting clients into various clusters.

The closest work to this thesis is by Cartea et al. [45], who explored statistical predictions of trading strategies in electronic markets. This thesis builds on their work by replicating the set of features they proposed for analyzing trader behavior. However, it extends their approach by exploring a broader range of clustering techniques and offering insights into the generalizability of their method.

4 Method

An overview of the method is illustrated in Figure 2. The project begins with an in-depth analysis of the order book and trade data. Next, the data is preprocessed and transformed into viable features that can be utilized for the end goal. Following this, various clustering algorithms are applied to the identified key characteristics of trading behaviors. Finally, domain experts evaluate the resulting clusters.

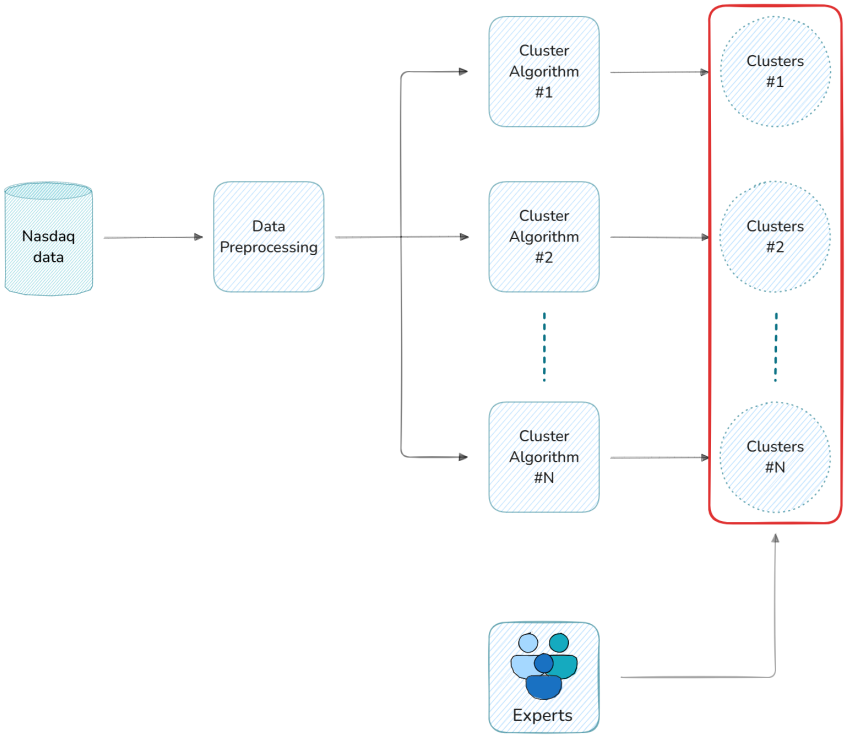


Figure 2: Method overview.

4.1 Data description

The data available for this thesis comprises order record-keeping data for the Nasdaq Nordic and Baltic markets trading on INET, Genium INET and the Nordic Derivatives Trading System (NDTS) as specified in [46]. INET is a trading platform for equities and the Spotlight stock market [47]. Genium INET is the clearing platform for Nordic equity derivatives, fixed income derivatives, commodities products and trading platform for fixed income and commodities [48]. NDTS is a platform for trading equity derivatives [49]. The datasets contain the fields as specified in [50].

4.1.1 Limitations

To limit the scope of this thesis, this work exclusively utilizes data from INET, as detailed in Section 4.1. The analysis focuses on a single stock called VOLVO with the ticker VOLV B over a period spanning from 02 October 2023 to 21 January 2024. In addition, retail traders, as defined in Section 2.1, are excluded from this study. Thus, the data set only keeps records of professional traders, which is indicated as algorithmic. As shown in Table 1, professional traders dominate the market activity. Lastly, this work only considers new-order events. As such, other order types, such as cancellations and executions, are filtered out in the preprocessing step.

Table 1 Statistics for traders between 2023-10-02 and 2024-01-21

trader	**message count**	**message percentage (%)**
Professional	19822007	95
Retail	1061402	5

As previously mentioned, this study focuses on VOLVO. It was selected because it offers sufficient liquidity for analysis, being one of the largest Nordic stocks, while its order book is not so deep as to cause computational challenges. The statistics for VOLVO can be observed in Table 2.

Table 2 Statistics for VOLVO between 2023-10-02 and 2024-01-21

Average daily message count	**Average daily traded volume €**
15331	112,937,019

4.1.2 Members and Traders

In this work, members are referred to as individuals, corporations, partnerships or entities allowed to trade on the INET Nordic Trading Platform. These members are allowed to trade for themselves or others. Moreover, they can also trade under one or multiple dealing capacities. The exchange determines the classification of capacities and is not regulated by a regulator; Nasdaq Nordic has nine different dealing capacities. However, in this thesis, only eight are interesting, as one of the capacities are retail. Furthermore, only four of these capacities are present in the dataset. In this thesis, a trader is defined as a combination of a member's ID, dealing capacity, and the trade executor ID.

4.1.3 Features

Not all fields presented in Section 4.1 may be relevant; some fields might be redundant, while others could misguide the clustering results. Furthermore, it has been proven that selecting a subset of the original features enhances performance. However, in an unsupervised setting where the ground truth is absent, identifying the most important features presents a significant challenge [51].

An AFM study [52] shows that many trading algorithms utilize features such as order book imbalance, volume in the order book, and price trends, all measured over various time horizons. Moreover, the most recent data have the most predictive power. In this thesis, 53

features[1]. proposed by [45] are used to reflect the state of the order book, and the set of features can be seen in Appendix A.

Finally, the unique trading behaviors, according to [45], are direction (buy or sell), price, and volume. The probability of a trader choosing direction, price, and volume can be modeled with logistic regression from the proposed features. The resulting regression coefficients capture individual trading tendencies and can be used to group similar trading behaviors. Therefore, the final set of features for the clustering of traders will be the coefficients from the logistic regression of the trading direction. Thus, the unique trading behavior in this work is direction and the key characteristics $(x_1, x_2, \ldots, x_n)$ mentioned in Section 1.1 are the 53 regression coefficients from the logistic model.

4.2 Data preprocessing

The data preprocessing stage involves transforming the available order book data described in Section 4.1 into the required target features, as described in Section 4.1.3. Moreover, it also involves filtering out data as described in Section 4.1.1. Lastly, logistic regression is performed to model trade direction and obtain the coefficients used for clustering.

The data preprocessing was primarily done in AWS Redshift, involving the development and optimization of SQL queries. This part of the work was particularly challenging due to the large volume of data, as processing billions of rows and handling massive datasets placed significant strain on computing resources. To manage this, the data was processed one week at a time for the 16-week period, resulting in 16 weekly datasets.

The 16 weeks of data were further processed by keeping only traders present in all 16 weeks and sufficient data for logistic regression analysis. The data was split into four consecutive weeks (i.e., weeks 1-4, 2-5, etc.), resulting in 13 data frames.

The 13 data frames were further split into training (70%) and test (30%) sets. Subsequently, trade direction was modeled using logistic regression, with each trader's resulting coefficients serving as the clustering analysis's final feature set.

4.3 Clustering

The datasets containing the features outlined in Section 4.1.3 represent the state of the order book for the time just before each order event. The data is aggregated using logistic regression as described in Section 2.6 to model each trader's probability of direction.

The clustering algorithms described in Section 2.3.1 are used to partition the datasets consisting of regression coefficients derived from the logistic regression models of trading behavior. The hierarchical clustering uses complete linkage and cosine affinity. Moreover, the spectral clustering uses nearest-neighbor affinity.

This thesis utilizes the implementations provided by Scikit-learn for the aforementioned clustering techniques: `KMeans`, `AgglomerativeClustering` and `SpectralClustering` from `sklearn.cluster`.

[1] Detailed descriptions of the proposed features are provided in Appendix B of [45].

4.4 Evaluation methods

Evaluating the effectiveness of the clustering results is challenging due to the lack of a definitive ground truth and is still an open problem. The definition of the clusters largely depends on the user and the domain, which is subjective [23]. Nonetheless, it is possible to evaluate clusters with intrinsic measures as described in Section 4.4.1. These metrics provide insights into the internal quality of the clustering and help address problem formulation one as outlined in Section 1.2 [25].

However, it is important to note that these metrics primarily evaluate the quality of the clusters and do not directly measure the correctness of the clusters. The correctness of the clusters would have to depend on extrinsic measures as described in Section 4.4.2; due to the lack of definitive ground truth, this thesis relies solely on domain expert evaluation to address problem formulation two, as outlined in Section 1.2.

4.4.1 Intrinsic measures

Intrinsic measures evaluate the quality of the cluster structure using only the information contained within a dataset, without relying on any external information. The evaluation focuses solely on the relationships between data points to determine the effectiveness of the clustering [25] [23]. Typical objective functions are designed to achieve high intra-cluster similarity, ensuring that objects within the same cluster are similar, while achieving low inter-cluster similarity, ensuring that objects from different clusters are dissimilar [23].

Silhouette score

The silhouette score measures the overall quality of a clustering result by computing the mean silhouette coefficient across all data points. The silhouette coefficient evaluates how well a data point is assigned to its cluster by considering both the intra-cluster and inter-cluster distances [24].

For a given data point x_i, the silhouette coefficient is calculated in two steps. First, let a_i be the average distance between x_i and all other points in the same cluster. Next, let b_i be the average distance between x_i and all points in the nearest neighboring cluster. Using these two values, the silhouette coefficient for point x_i is defined as [24]:

$$s_i = \frac{b_i - a_i}{max(a_i, b_i)} \tag{4.1}$$

The silhouette score is obtained by calculating the mean of all individual silhouette coefficients resulting in [24]:

$$S = \frac{\sum_{i=1}^{N} \frac{b_i - a_i}{max(a_i, b_i)}}{N} \tag{4.2}$$

A silhouette score of 1 indicates that the data points are well-clustered and correctly assigned. A score around 0 implies that the clusters overlap. Lastly, a score of -1 suggests that the points have been misclassified and assigned to the wrong clusters [53].

Davies-Bouldin index

The Davies-Bouldin index evaluates the quality of a clustering result by considering for each cluster C the similarity between it and all other clusters [24]. Consider a cluster C with members $X_1, X_2, \ldots, X_m \in E_p$, where E_p is a Euclidean space of p dimensions. Let $S(X_1, X_2, \ldots, X_m)$ denote a dispersion measure that satisfies the following [54]:

$$S(X_1, X_2, \ldots, X_m) \geq 0 \tag{4.3}$$

$$S(X_1, X_2, \ldots, X_m) = 0 \quad \text{iff} \quad X_i = X_j \forall X_i, X_j \in C \tag{4.4}$$

The dispersion measure S_i for a cluster C_i is computed as the average distance between each point in the cluster and its centroid. The inter-cluster distance M_{ij} is defined as the distance between the centroids of two clusters C_i and C_j. Using these measures, the pairwise cluster similarity between clusters C_i and C_j is given by :

$$R_{ij} = \frac{S_i + S_j}{M_{ij}} \tag{4.5}$$

The Davies-Bouldin index is then defined as the average of the maximum similarity R_{ij} for each cluster :

$$R = \frac{1}{N} \sum_{i=1}^{N} \max_{i \neq j} R_{ij} \tag{4.6}$$

As seen in Equation 4.6, each cluster is assigned the highest similarity value relative to any other cluster, and the final index is obtained by averaging all cluster similarities. The smaller the index indicates more compact and well-separated clusters [24].

Calinski–Harabasz index

The Calinski–Harabasz index, also called the Variance Ratio Criterion (VRC), is a measure that evaluates the quality of clustering results by measuring the ratio of between-cluster dispersion to within-cluster dispersion. For a dataset with n points partitioned into k clusters the Calinski–Harabasz index is given by [55]:

$$VRC = \frac{BGSS}{k-1} / \frac{WGSS}{n-k} \tag{4.7}$$

Where WGSS is the Within Group Sum of Squares and BGSS is the Between-Group Sum of Squares, these are given by [55]:

$$WGSS = \sum_{g=1}^{k} \sum_{i=1}^{n_g} d_{i,g}^2 \tag{4.8}$$

$d_{i,g}^2$ denotes the squared distance between the point P_i in cluster g and the centroid of that cluster, and n_g represents the number of data points in cluster g.

$$BGSS = \sum_{g=1}^{k} n_g \cdot d_{g,c}^2 \tag{4.9}$$

$d_{c,g}^2$ denotes the the squared distance between the centroid of cluster g and the overall centroid of the dataset. WGSS measures the dissimilarity within clusters, while BGSS measures the dissimilarity between clusters. Thus, a higher value of VRC indicates a better partition [55].

4.4.2 Extrinsic measures

Extrinsic measures evaluate the quality of clusters by comparing them to externally supplied class labels or ground truth. Extrinsic measures are the most commonly used approach for assessing clustering performance [23]. However, this thesis lacks the ground truth and commonly used extrinsic measures such as F-score, Purity or Rand index cannot be utilized in this work[2].

One extrinsic measure that is explored in this thesis is domain expert evaluation. The domain expert evaluation provides additional insights into the cluster results besides the intrinsic measures mentioned in Section 4.4.1.

Expert evaluation

As mentioned in Section 4.4, there is no definitive ground truth. Thus, this project intends to include domain expert evaluation of the clustering results. Domain expert evaluation can serve as an important qualitative measure of the model's effectiveness and real-life applications. Moreover, it addresses problem formulation two outlined in Section 1.2.

The approach for expert evaluation involves presenting the clustering results as trader group statistics as seen in Appendix B to domain experts. The domain experts will rate how well the clustering algorithm identifies distinct groupings/categories of traders on a scale of 1 to 7. The form also includes detailed information about the clusters, including specific traders in each cluster.

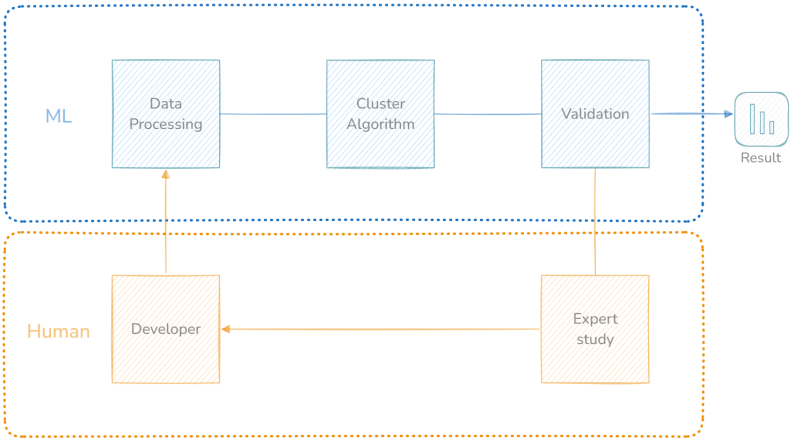


Figure 3: Process of the human-in-the-loop

[2]These metrics require true class labels for evaluation. See Table 23.1 in [24] for a detailed overview.

5 Results

This Chapter presents the logistic regression and clustering results of the 125 unique traders present on 2023-10-02 and 2024-01-21 (16 weeks). In Section 5.3, the results of three clusters are presented using k-means, hierarchical and spectral clustering. Similarly, in Section 5.4, the results of four clusters are presented.

5.1 Logistic regression

In this section, the results of the logistic regression are presented. Table 3 shows the mean accuracies of the top 5, 10, 20 and 50 traders ranked by order count. The top 5 traders will have the most orders and, therefore, the most data points for training and validation.

Table 3 Accuracies of the logistic regression models for VOLVO, calculated over thirteen exercises. Accuracies are reported in % with ± the standard deviation.

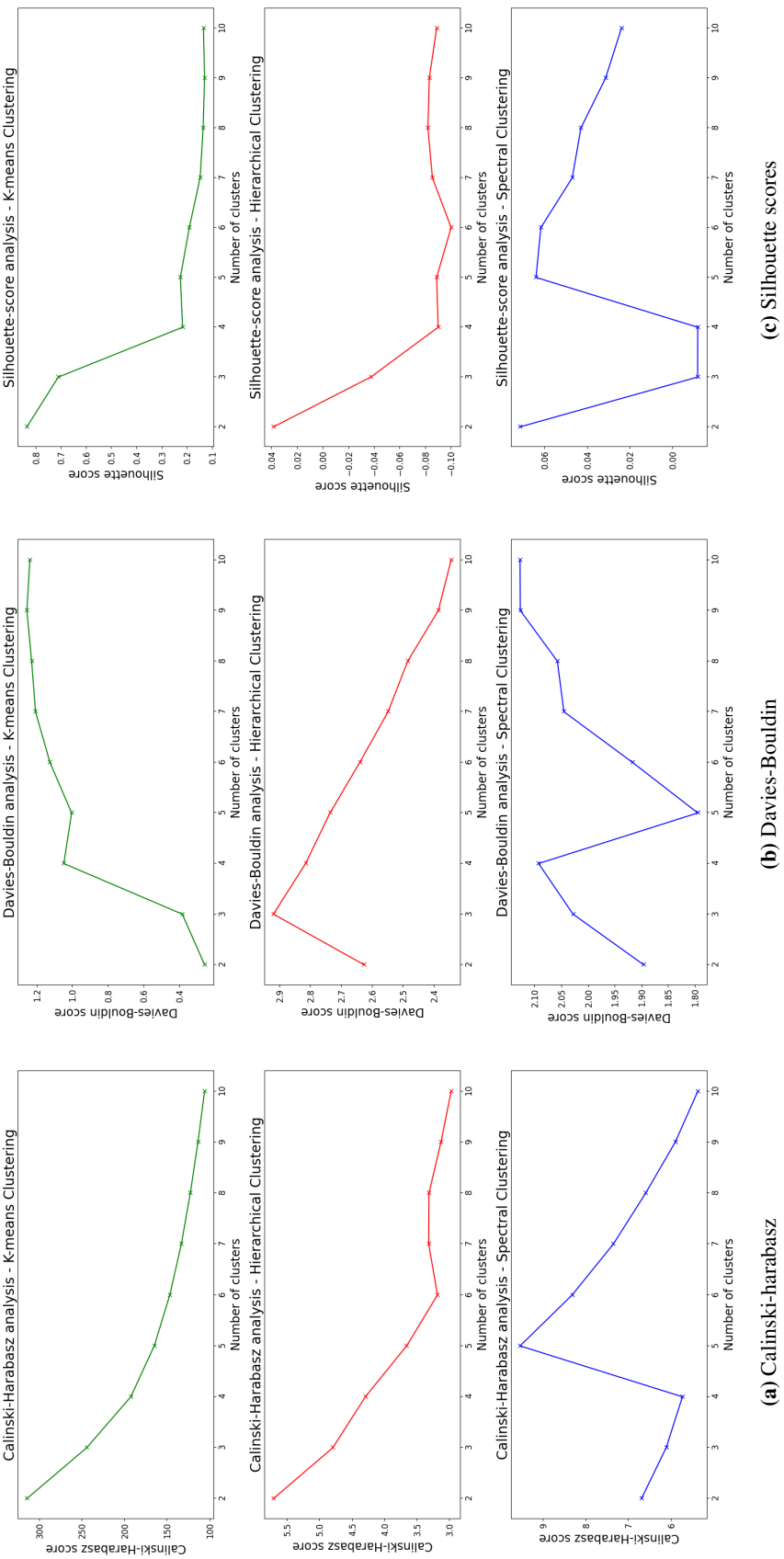
	Accuracy (%)	σ
Top 5	76,5	15,9
Top 10	70,3	12,2
Top 20	71,5	12,6
Top 50	77,9	12,0
All	77,6	15,2

5.2 Optimal number of clusters

This section presents the analysis of the optimal number of clusters. Table 4 shows the average Calinski-harabasz, Davies-Bouldin and Silhouette scores for three cluster algorithms with different *k* ranges, from 2 to 10, across 13 exercises. Figure 4 displays the plots.

Table 4 Calinski-harabasz, Davies-Bouldin and Silhouette scores for cluster k.

	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
Kmeans-c	313.26	206.97	184.29	154.61	135.26	122.90	110.75	103.76	95.94
hierarchical-c	5.70	4.79	4.29	3.66	3.18	3.32	3.31	3.13	2.97
spectral-c	6.69	6.11	5.69	9.53	8.31	7.38	6.59	5.95	5.35
kmeans-d	0.25	0.86	0.89	1.26	1.08	1.27	1.26	1.25	1.44
hierarchical-d	2.63	2.92	2.82	2.74	2.64	2.55	2.48	2.39	2.34
spectral-d	1.90	2.03	2.09	1.80	1.92	2.04	2.06	2.11	2.12
kmeans-s	0.84	0.42	0.29	0.16	0.17	0.13	0.13	0.12	0.09
hierarchical-s	0.04	-0.04	-0.09	-0.09	-0.10	-0.09	-0.08	-0.08	-0.09
spectral-s	0.07	-0.01	-0.01	0.06	0.06	0.05	0.04	0.03	0.02



(a) Calinski-harabasz **(b)** Davies-Bouldin **(c)** Silhouette scores

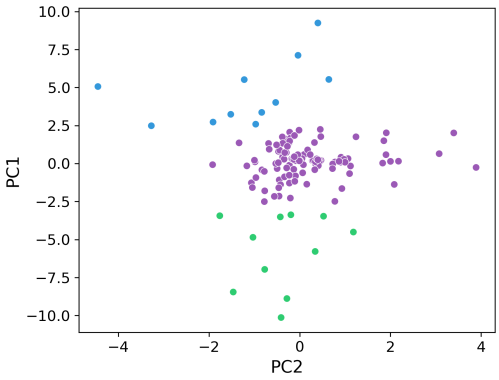
Figure 4: Plots showing Calinski-harabasz, Davies-Bouldin and Silhouette scores for cluster k.

5.3 Results with three clusters

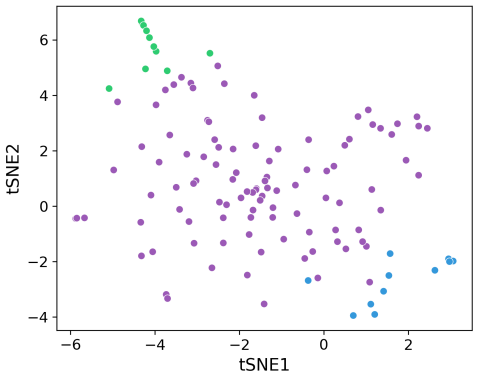
This section presents the results of the three cluster algorithms with $k = 3$. Section 5.3.1 shows the clustering results with k-means, and Section 5.3.2 shows the clustering results with hierarchical clustering. Lastly, section 5.3.3 shows the clustering results with spectral clustering.

5.3.1 K-means

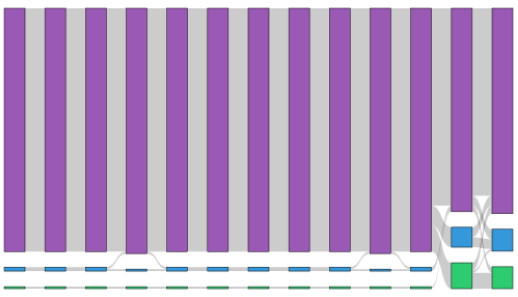
Figures 5a and 5b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 5c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 5d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



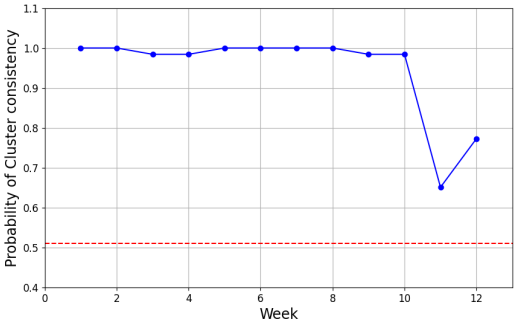
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

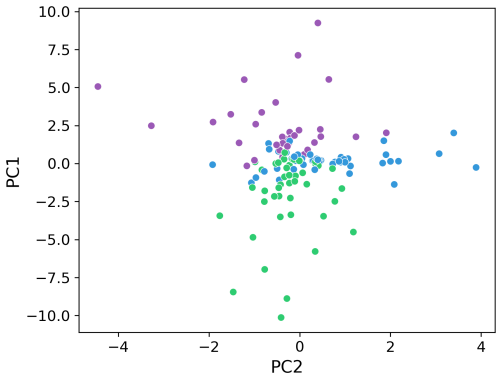


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

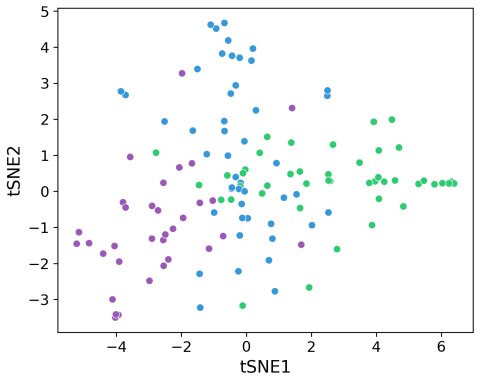
Figure 5: Results of k-means with three clusters.

5.3.2 Hierarchical

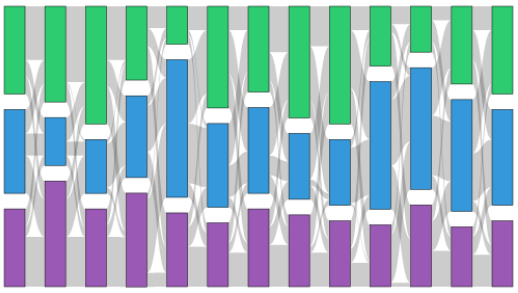
Figures 6a and 6b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 6c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 6d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



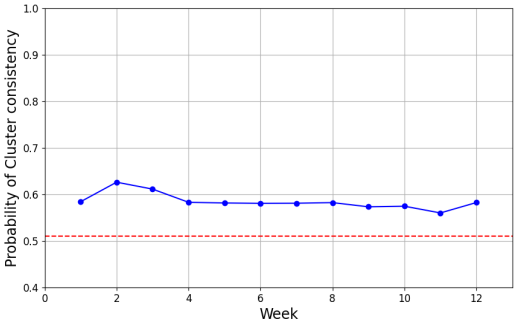
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

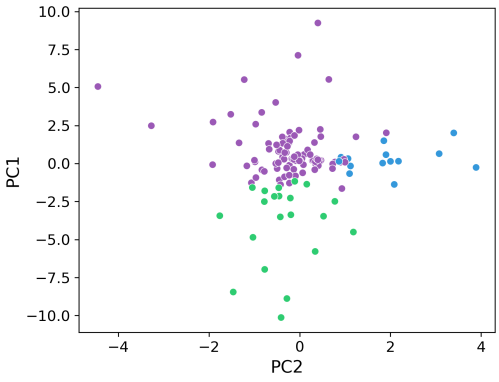


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

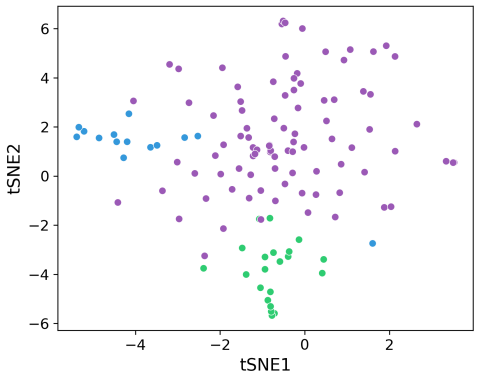
Figure 6: Results of hierarchical with three clusters.

5.3.3 Spectral

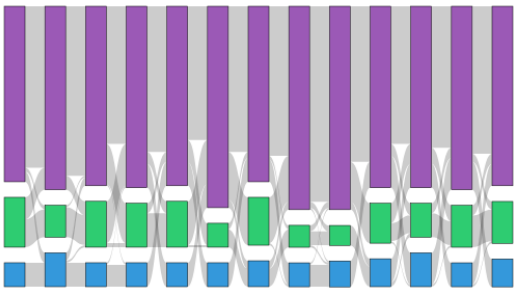
Figures 7a and 7b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 7c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 7d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



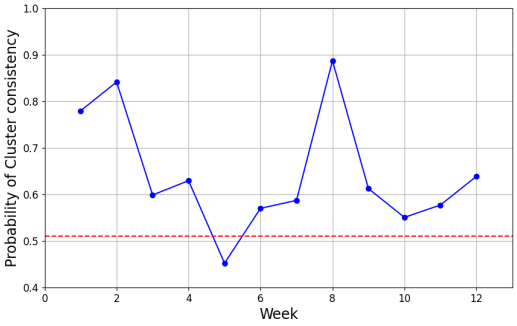
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).



(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

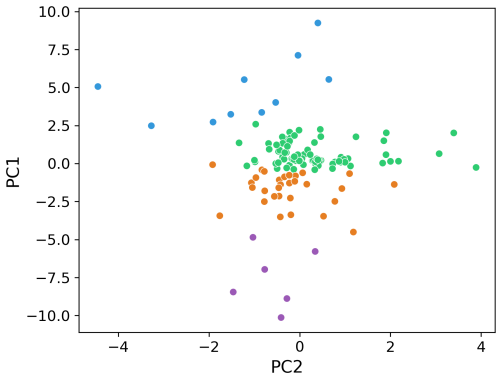
Figure 7: Results of spectral with three clusters.

5.4 Results with four clusters

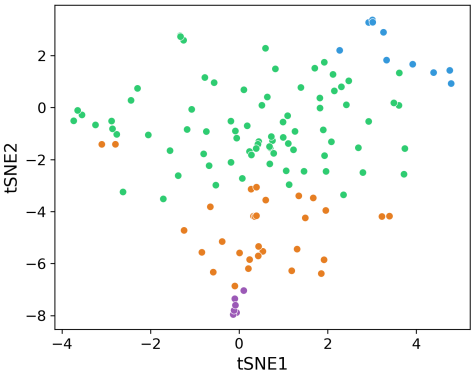
This section presents the results of the four cluster algorithms with $k = 4$. Section 5.4.1 shows the clustering results with k-means, and Section 5.4.2 shows the clustering results with hierarchical clustering. Lastly, section 5.4.3 shows the clustering results with spectral clustering.

5.4.1 K-means

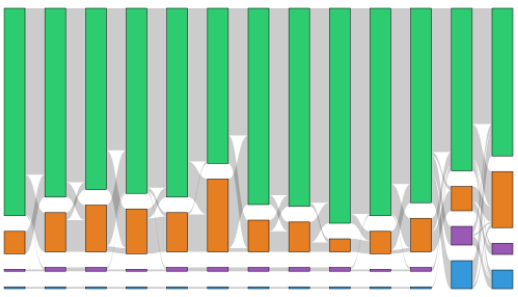
Figures 8a and 8b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 8c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 8d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



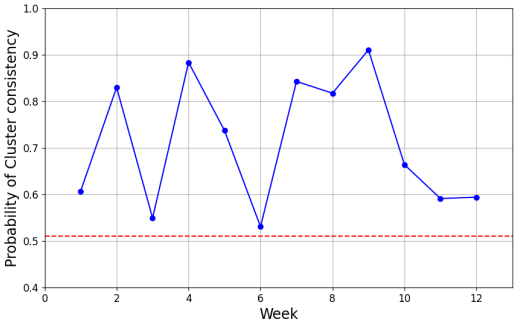
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

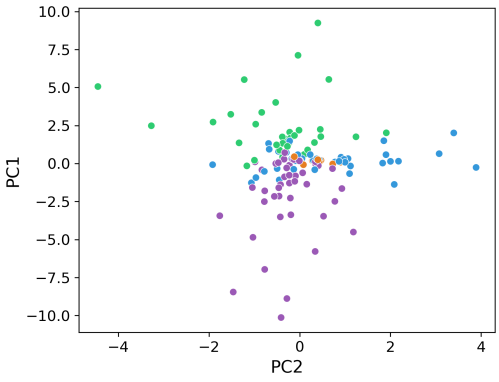


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

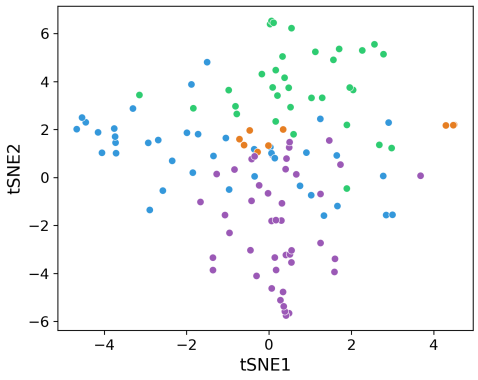
Figure 8: Results of k-means with four clusters.

5.4.2 Hierarchical

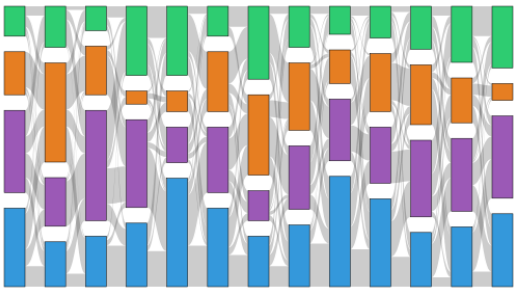
Figures 9a and 9b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 9c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 9d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



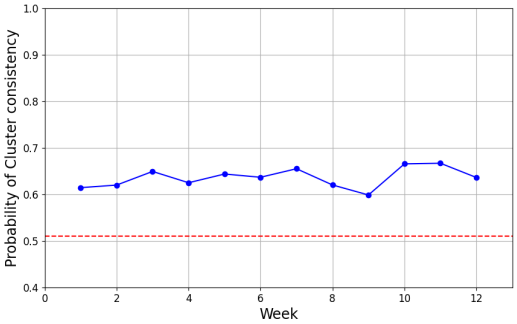
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

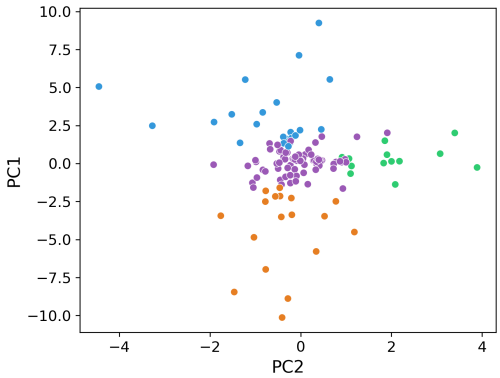


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

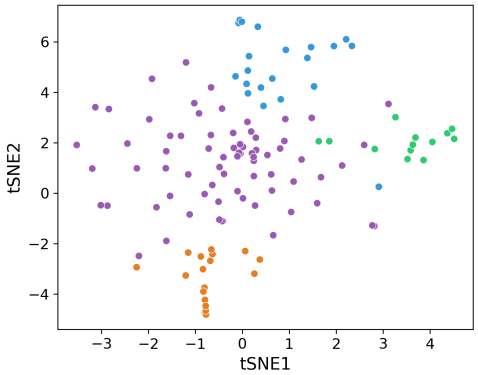
Figure 9: Results of hierarchical with four clusters.

5.4.3 Spectral

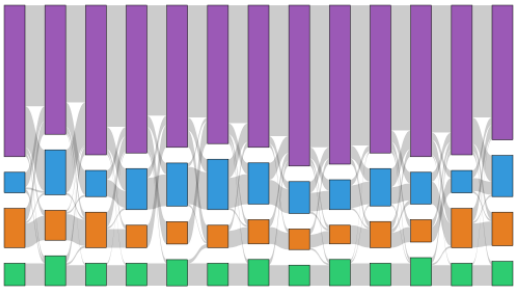
Figures 10a and 10b show scatter plots created using dimension reduction methods as described in Section 2.5. Figure 10c illustrates cluster transitions over 13 time spans of 4 consecutive weeks. Each column in the figure represents a 4-week span of data, i.e., the first column covering weeks 1-4, the second column weeks 2-5, and so on. Lastly, Figure 10d shows the stability of the cluster exercises. The y-axis indicates the probability that pairs of traders in two consecutive clustering exercises, A and B, are either in the same or different clusters in both exercises. The x-axis represents the 13 time spans of clustering exercises, while the red dotted line shows the expected probability if traders were randomly assigned to clusters.



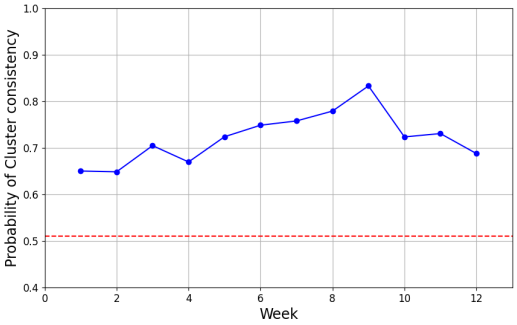
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).



(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

Figure 10: Results of spectral with four clusters.

5.5 Domain expert evaluation

This section outlines the results from the qualitative evaluation of the identified clusters shown in Section 5.3 and 5.4 with the help from domain experts with the methodology described in Section 4.4.2.

Table 5 shows the expert scoring of the clustering results with the number of clusters set to three ($k = 3$), displaying the average score for each algorithm.

Table 5 Expert scoring of the clustering results on a scale of 1-7 for three clusters.

Algorithm	Expert 1	Expert 2	Expert 3	Expert 4	Avg
k-means	5	1	2	4	3.00
hierarchical	6	2	4	6	4.50
spectral	7	3	3	6	4.75

Similarly, Table 6 shows the expert scoring of the clustering results with the number of clusters set to four ($k = 4$), also displaying the average score for each algorithm.

Table 6 Expert scoring of the clustering results on a scale of 1-7 for four clusters.

Algorithm	Expert 1	Expert 2	Expert 3	Expert 4	Avg
k-means	4	3	3	3	3.25
hierarchical	6	3	3	5	4.25
spectral	4	1	4	5	3.50

Moreover, one domain expert provided the following detailed feedback through the evaluation form:

> "To me, it's unclear how accurate the different algorithms are in separating trader groups. I see several reasons why this is the case: (i) predicting buy vs sell is the wrong metric to focus on. Everyone buys and sells over an extended period of time. There is no single trader group that consistently buys or sells over long periods; (ii) looking at a single stock does not allow the full scope of underlying strategies employed by trader groups to be captured. The specific stock in question here could have idiosyncratic features that make it hard to generalize; (iii) capacity and executor ID may not accurately reflect the underlying trader groups due to potential limitations in how these fields are recorded by different members. Overall, while the results do show some features of different trader groups, the metrics reported are not all consistent."

6 Discussion

This chapter discusses the results of Chapter 5 in relation to the research questions outlined in Section 1.2. As the analysis focuses on a single stock, the findings should be viewed as exploratory and context-specific. Section 6.1 examines the results of the logistic regression, Section 6.2 discusses the results of the intrinsic measures, and Section 6.3 covers the insights of the expert evaluations. Finally, Section 6.4 provides directions for future work.

6.1 Trading behavior

As mentioned in Section 4.1.3, the defining trading behaviors are the trader's direction, price, and volume. In this work, only the direction has been considered and modeled using logistic regression, which is the most straightforward approach, providing one coefficient for each feature. However, this also means that the logistic regression step will largely influence the clustering results, as it serves as a form of dimensionality reduction by aggregating all of the order events for a trader. Furthermore, the logistic regression training step introduces some randomness, so the results may vary for each run.

Nonetheless, Table 3 shows that the accuracy for predicting trader direction is relatively good, better than random. Several variables could influence these results. For example, the data points for each trader are not equal in volume. The top 5 traders, as observed in Table 3, have more data points to train on for the logistic regression, which could explain the higher accuracy but also the higher standard deviation, as more data points might introduce greater variability.

Another factor could be that the top 5 traders may employ a more diverse trading strategy and are more sensitive to market conditions, reacting quickly to changes. This adaptability of the traders could affect the logistic regression results.

Lastly, the number of features is a crucial consideration. As mentioned in Section 4.1.3, 53 features are used to reflect the state of the order book. This high dimensionality could lead to overfitting due to the "curse of dimensionality" as described in 2.6. During this work, I attempted to use PCA (dimensionality reduction) as described in Section 2.5 to lower the dimensionality, but it did not yield any improvement. Furthermore, I performed a grid search with k-fold cross validation for each logistic regression model to try to combat overfitting, but this did not result in any improvement in terms of accuracy.

6.2 Intrinsic Evaluation of clusters

Intrinsic cluster measures, as described in Section 4.4.1, were used to analyze the clusters to check the optimal number of clusters and compare the different cluster algorithms. The

first step was to check the optimal number of clusters, which can be observed in Section 5.2. Since this work is highly exploratory and the trader groups are not strictly defined, the number of clusters is unknown. However, that is not entirely true; as stated in Section 4.1.2, there are known dealing capacities of a trader, which could serve as a baseline for determining the number of clusters. Therefore, three or four clusters should be the optimal number of clusters.

6.2.1 Number of clusters

Figure 4 presents the Calinski-harabasz, Davies-Bouldin and Silhouette scores for three clustering algorithms, k-means, hierarchical, and spectral, across a range of cluster numbers from two to ten. Table 4 shows the corresponding values.

Observations regarding the scores show that k-means have the most variation across the metrics. However, it is consistent that lower k-values are optimal as scores degrade when k increases. For $k = 2$, the Calinski-harabasz is the highest, and the score decreases steadily as the number of clusters increases. A higher score indicates dense and well-separated clusters. Similarly, a high Silhouette score indicates well-separated and cohesive clusters, and it starts at a high value and drops as the number of clusters increases. However, for Davies-Bouldin, lower values indicate a better separation, and it could be seen that initial scores are low, and the score rises as the number of clusters grows. The intrinsic measures show that the optimal number of clusters are $k = 2$ and $k = 3$.

The results for hierarchical clustering show consistently poor performance, as can be observed in the values shown in Table 4. The Calinski-harabasz index at its highest at $k = 2$ is very low and decreases even further for higher k values. Moreover, Davies-Bouldin starts high, and while there is a decrease, the value is consistently high across all k. Lastly, the values for the silhouette score lie near zero or even negative, which indicates overlapping clusters. The intrinsic measures show that hierarchical clustering does not find any meaningful structure.

The results for Spectral are similar to those for hierarchical, but with slightly better values, they still indicate overlapping clusters. However, an interesting observation is that $k = 5$ is the optimal number for spectral. However, five clusters are not considered in this work as the intrinsic measures for spectral show weak performance. Although the values for $k = 2$ were consistently good in all clusters, two clusters are not interesting in this work. Therefore, $k = 3$ and $k = 4$ were chosen for the clustering exercises.

6.2.2 Cluster comparison

Table 4 shows for $k = 3$ and $k = 4$ that k-means outperforms the other two algorithms when clustering traders according to each intrinsic measure. However, this result could be misleading. As shown in Section 5.3.1, the results for three clusters with k-means (Figure 5c) reveal that in 11 out of 13 cluster exercises, all traders are grouped into one large cluster. This clustering behavior could explain the high intrinsic measure scores, as these metrics typically evaluate intra-cluster cohesion and inter-cluster separation. With one predominant cluster, the comparison with a few distant outliers would result in a high evaluation score. The stability of k-means shown in Figure 5d also makes sense as all of the traders are grouped into the same big cluster for each timespan; however, for the last 2 cluster exercises,

it was able to identify more outliers, which caused a drop in the stability. Furthermore, when observing the scatter plots in Figure 5a and 5b, it is difficult to see any significant separation between clusters.

In contrast, the hierarchical results in Section 5.3.2 show a quite different picture. Figure 6c shows three prominent clusters for all 13 cluster exercises. However, it is clear that the cluster transition for each timespan is very volatile, and this becomes obvious looking at Figure 6d, which shows that the stability of the hierarchical cluster is slightly better than random, but there is a fair amount of instability in the clusters over time, as indicated by the gap between the blue line and 1. Similarly, as with k-means, it is difficult to see any significant separation between clusters in the scatter plots in Figure 6a and 6b.

Lastly, Spectral clustering was also able to identify three clusters, one big cluster and two smaller ones. as can be observed in Figure 7c. However, the stability of the spectral clustering exhibits considerable variability across the different runs, which could indicate the unreliability of the clustering solution. While the stability reaches a value of 0.89, seen in Figure 5d, it also drops to 0.45 in another case, falling below the random benchmark. One explanation for this variability is that spectral clustering captures the variability in direction, as directionality may not entirely capture unique trading behavior. Moreover, part of the instability might be due to traders updating their trading strategies. If any of these traders change their trading behavior in how they send orders, the coefficients obtained in the logistic regression will also change, resulting in unstable clusters.

The results for the four clusters present similar results to the three clusters when looking at the intrinsic measure scores. However, Figure 8c shows that k-means could identify two clusters; nonetheless, the stability is very volatile. While hierarchical results are similar to those with three clusters, they remain consistent, but the gap persists between the blue line and 1, indicating there is still instability. The results for Spectral improved a bit as Figure 10d shows consistency, and the gap between the blue line and 1 is less than that of hierarchical.

In conclusion, since there is a lack of ground truth, intrinsic measures are the only metrics to evaluate the clusters. Therefore, to answer research question one, relying purely based on these intrinsic measures, the best performing clustering algorithm to categorize trading firms is k-means, spectral in second and hierarchical performing the worst.

6.3 Qualitative Evaluation of clusters

The results from the domain expert evaluation, presented in Section 5.5, indicate consistent trends in the perceived quality of the clustering results. Across both evaluations for $k = 3$ and $k = 4$, k-means was consistently rated low by all four experts, resulting in the lowest average scores in both tables. The domain expert scores suggest that k-means was the least effective in capturing meaningful patterns in the data.

The expert evaluations of hierarchical and spectral clustering are less clear when looking at the results for three clusters ($k = 3$). Two experts rated spectral clustering higher, one expert preferred hierarchical clustering, and the fourth expert rated them similarly. In terms of average scores, spectral clustering achieved the highest mean score (4.75), followed closely by hierarchical clustering (4.50). Therefore, for $k = 3$, spectral clustering performs best,

with hierarchical clustering as a close second and k-means as the least effective.

The results differ slightly in the case of four clusters ($k = 4$). While k-means remains the lowest-performing algorithm on average, hierarchical clustering now receives the highest average score (4.25), followed by spectral clustering (3.50). Interestingly, spectral clustering receives more mixed evaluations for four clusters, suggesting it may not capture meaningful structure at this number of clusters. This observation might align with the results in Table 4, where spectral clustering shows weaker performance for $k = 4$, indicating that spectral clustering fails to identify any meaningful structures for four clusters.

In conclusion, from a subjective, expert-based perspective, k-means is the weakest performer in both evaluations. Spectral clustering performs best with three clusters, while hierarchical clustering performs best with four clusters. Since the average expert scores are higher in the results with three clusters than for four clusters, spectral clustering achieves the highest score in that case. Thus, we can conclude that spectral clustering is the most effective approach for categorizing trading firms in this context, thereby addressing research question two.

6.4 Future work

The detailed feedback provided by the domain expert, as discussed in Section 5.5, indicates that none of the clusters successfully separate traders into clearly distinct groups. While some clustering results suggest some distinct trader groups, the metrics for the trader groups shown in Appendix B are contradictory and do not provide insights of business value or real-world applicability.

One key observation made by the expert is that the clustering approach modeled the direction of trading activity. However, all traders buy and sell over extended periods, leading to overlapping and non-distinct trading behaviors. An alternative approach was explored by modeling price instead; the results of this can be found in Appendix C. While the findings appear promising, there was insufficient time for a qualitative evaluation. This alternative would represent the second iteration in the process illustrated in Figure 3.

Additionally, the scope of this thesis was intentionally constrained to maintain manageability, as described in Section 4.1.1. These limitations may have influenced the clustering results. For example, important behavioral patterns may have been excluded by focusing solely on new orders. The expert also noted that analyzing only a single stock fails to capture individual traders' broader trading strategies or behaviors.

Moreover, the expert highlighted that fields such as capacity and executor ID may not reliably reflect actual trader identities, as different members may record these fields inconsistently. This inconsistency could further undermine the reliability of the clustering results.

In conclusion, future work should consider shifting the focus from trade direction to other distinctive trading behaviors and expanding the scope of this study. The scope could be broadened by including data from multiple stocks and a broader range of order types. Refining the feature set may help better capture meaningful distinctions between traders. These improvements will likely enhance the quality and applicability of the clustering results.

7 Conclusion

This thesis explored different clustering techniques for identifying groupings of trading firms within financial markets, leveraging machine learning methodologies alongside domain expertise. By applying k-means, hierarchical, and spectral clustering algorithms to trader features derived from logistic regression models, the aim was to uncover meaningful trader categories based on unique behavioral patterns.

The findings indicate that clustering models can offer valuable insights into trader segmentation. Expert evaluations suggest that spectral and hierarchical clustering are more promising than k-means, even though intrinsic evaluation metrics favored k-means. This discrepancy highlights the importance of incorporating human-in-the-loop approaches to improve the interpretability and practical relevance of clustering results.

Despite these advancements, challenges remain in achieving better separation of distinct trader groups. Future research should focus on modeling another unique trading behavior than directionality, expand the scope of the analysis by incorporating data from multiple stocks, a broader set of order types, and more refined feature engineering to better capture trader behaviors.

Ultimately, this thesis contributes to the growing work on machine learning in financial markets by demonstrating the potential of unsupervised learning for analyzing trader behavior. By bridging algorithmic insights with domain expert feedback, this approach lays the groundwork for further developments in clustering-based financial analytics.

References

[1] Bookmap. "Deciding about data: The importance of data when trading." (2024), [Online]. Available: https://bookmap.com/blog/the-importance-of-data-when-trading (visited on 01/30/2025).

[2] F. Cai, N.-A. Le-Khac, and T. Kechadi, *Clustering approaches for financial data analysis: A survey*, 2016. arXiv: 1609.08520 [q-fin.GN]. [Online]. Available: https://arxiv.org/abs/1609.08520.

[3] D. B. Group. "Market data – relevance for the financial market and the economy." (n.d.), [Online]. Available: https://www.deutsche-boerse.com/dbg-en/media/news-stories/explainers/financial-market-know-how/market-data (visited on 01/30/2025).

[4] IG. "Market data definition." (n.d.), [Online]. Available: https://www.ig.com/uk/glossary-trading-terms/market-data-definition#information-banner-dismiss (visited on 01/30/2025).

[5] Nasdaq. "Nasdaq data link." (n.d.), [Online]. Available: https://www.nasdaq.com/nasdaq-data-link (visited on 01/30/2025).

[6] Nasdaq. "Investor activity indicator." (n.d.), [Online]. Available: https://www.nasdaq.com/solutions/investor-activity-indicator (visited on 01/30/2025).

[7] R. Dolphin, B. Smyth, and R. Dong, "A machine learning approach to industry classification in financial markets," in *Artificial Intelligence and Cognitive Science*, L. Longo and R. O'Reilly, Eds., Cham: Springer Nature Switzerland, 2023, pp. 81–94, ISBN: 978-3-031-26438-2.

[8] N. Nazareth and Y. V. Ramana Reddy, "Financial applications of machine learning: A literature review," *Expert Systems with Applications*, vol. 219, p. 119 640, 2023, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2023.119640. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423001410.

[9] C. Poutré, D. Chételat, and M. Morales, "Deep unsupervised anomaly detection in high-frequency markets," *The Journal of Finance and Data Science*, vol. 10, p. 100 129, 2024, ISSN: 2405-9188. DOI: https://doi.org/10.1016/j.jfds.2024.100129. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S240591882400014X.

[10] A. Samitas, E. Kampouris, and D. Kenourgios, "Machine learning as an early warning system to predict financial crisis," *International Review of Financial Analysis*, vol. 71, p. 101 507, 2020, ISSN: 1057-5219. DOI: https://doi.org/10.1016/j.irfa.2020.101507. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1057521920301514.

[11] W. Bao, N. Lianju, and K. Yue, "Integration of unsupervised and supervised machine learning algorithms for credit risk assessment," *Expert Systems with Applications*, vol. 128, pp. 301–315, 2019, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2019.02.033. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417419301472.

[12] W. Kenton. "What is an order book? definition, how it works, and key parts." (2024), [Online]. Available: https://www.investopedia.com/terms/o/order-book.asp (visited on 02/11/2025).

[13] A. Hayes. "Understanding liquidity and how to measure it." (2024), [Online]. Available: https://www.investopedia.com/terms/l/liquidity.asp (visited on 04/14/2025).

[14] IG. "Order book definition." (2024), [Online]. Available: https://www.ig.com/en/glossary-trading-terms/order-book-definition (visited on 02/11/2025).

[15] SmartAsset. "What is a market maker and what do they do?" (2024), [Online]. Available: https://www.nasdaq.com/articles/what-market-maker-and-what-do-they-do (visited on 02/11/2025).

[16] P. Satsuk. "Liquidity providers vs market makers: Everything you need to know." (2024), [Online]. Available: https://www.wlglobal.solutions/blog/liquidity-provider-vs-market-maker/ (visited on 02/11/2025).

[17] YourPropFirm. "Liquidity provider vs market maker: What is the difference?" (2024), [Online]. Available: https://yourpropfirm.com/liquidity-provider-vs-market-maker/ (visited on 02/11/2025).

[18] Nasdaq. "Retail investors." (2024), [Online]. Available: https://www.nasdaq.com/glossary/r/retail-investors (visited on 02/18/2025).

[19] K. Zucchi. "Institutional traders vs. retail traders: What's the difference?" (2023), [Online]. Available: https://www.investopedia.com/articles/active-trading/030515/what-difference-between-institutional-traders-and-retail-traders.asp (visited on 02/18/2025).

[20] J. Chen. "Algorithmic trading: Definition, how it works, pros & cons." (2024), [Online]. Available: https://www.investopedia.com/terms/a/algorithmictrading.asp (visited on 02/18/2025).

[21] A. Farley, "Proprietary trading: What it is, how it works, benefits," *Investopedia*, 2025. [Online]. Available: https://www.investopedia.com/terms/p/proprietarytrading.asp (visited on 05/16/2025).

[22] A. Yeshchenko, C. D. Ciccio, J. Mendling, and A. Polyvyanyy, "Comprehensive process drift detection with visual analytics," *CoRR*, vol. abs/1907.06386, 2019. arXiv: 1907.06386. [Online]. Available: http://arxiv.org/abs/1907.06386.

[23] S. Aghabozorgi, A. Seyed Shirkhorshidi, and T. Ying Wah, "Time-series clustering – a decade review," *Information Systems*, vol. 53, pp. 16–38, 2015, ISSN: 0306-4379. DOI: https://doi.org/10.1016/j.is.2015.04.007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306437915000733.

[24] C. C. Aggarwal and C. K. Reddy, Eds., *Data Clustering: Algorithms and Applications*, 1st. Chapman and Hall/CRC, 2014. DOI: 10.1201/9781315373515.

[25] IBM. "What is clustering?" (2024), [Online]. Available: https://www.ibm.com/think/topics/clustering (visited on 02/12/2025).

[26] S. J. Prince, *Understanding Deep Learning*. The MIT Press, 2023. [Online]. Available: http://udlbook.com.

[27] H. Yin, A. Aryani, S. Petrie, A. Nambissan, A. Astudillo, and S. Cao, *A rapid review of clustering algorithms*, 2024. arXiv: 2401.07389 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2401.07389.

[28] A. Levy, B. R. Shalom, and M. Chalamish, *A guide to similarity measures*, 2024. arXiv: 2408.07706 [cs.IR]. [Online]. Available: https://arxiv.org/abs/2408.07706.

[29] Y. Wang, H. Huang, C. Rudin, and Y. Shaposhnik, *Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization*, 2021. arXiv: 2012.04456 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2012.04456.

[30] L. van der Maaten, E. Postma, and J. van den Herik, "Dimensionality reduction: A comparative review," Tilburg centre for Creative Computing, Tilburg University, Tilburg, The Netherlands, Tech. Rep. TiCC TR 2009–005, Oct. 2009, Available: http://www.uvt.nl/ticc.

[31] A. Lindholm, N. Wahlström, F. Lindsten, and T. B. Schön, *Machine Learning - A First Course for Engineers and Scientists*. Cambridge University Press, 2022. [Online]. Available: https://smlbook.org.

[32] E. E. Seghers, L. A. Briceno-Mena, and J. A. Romagnoli, "Unsupervised learning: Local and global structure preservation in industrial data," *Computers & Chemical Engineering*, vol. 178, p. 108 378, 2023, ISSN: 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2023.108378. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S009813542300248X.

[33] S.-l. developers, *Tsne class*, https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html, Accessed: 2025-04-19, 2025.

[34] C. Aliferis and G. Simon, "Overfitting, underfitting and general model overconfidence and under-performance pitfalls and best practices in machine learning and ai," in *Artificial Intelligence and Machine Learning in Health Care and Medical Sciences: Best Practices and Pitfalls*, G. J. Simon and C. Aliferis, Eds. Cham: Springer International Publishing, 2024, pp. 477–524, ISBN: 978-3-031-39355-6. DOI: 10.1007/978-3-031-39355-6_10. [Online]. Available: https://doi.org/10.1007/978-3-031-39355-6_10.

[35] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, no. 2, p. 022 022, 2019. DOI: 10.1088/1742-6596/1168/2/022022.

[36] G. C. Cawley and N. L. C. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *Journal of Machine Learning Research*, vol. 11, pp. 2079–2107, 2010. [Online]. Available: https://www.jmlr.org/papers/volume11/cawley10a/cawley10a.pdf.

[37] P. Liashchynskyi and P. Liashchynskyi, "Grid search, random search, genetic algorithm: A big comparison for NAS," *CoRR*, vol. abs/1912.06059, 2019. arXiv: 1912.06059. [Online]. Available: http://arxiv.org/abs/1912.06059.

[38] G. Tang, R. Tian, and B. Wu, "An overview of clustering methods in the financial world," in *Proceedings of the 2022 7th International Conference on Financial Innovation and Economic Development (ICFIED 2022)*, Open access article distributed under the CC BY-NC 4.0 license., vol. 211, Atlantis Press International B.V., 2022.

[39] J. Millberg, "Evaluating clustering techniques in financial time series," M.S. thesis, Uppsala University, Uppsala, Sweden, 2023.

[40] D. Tangsirisakul, "Comparing Clustering Algorithms using Financial Time-series data," *International Journal of Science and Business*, vol. 3, no. 2, pp. 146–166, 2019. DOI: https://doi.org/10.5281/zenodo.2617341. [Online]. Available: http://ijsab.com/wp-content/uploads/334.pdf.

[41] S. Mankad, G. Michailidis, and A. Kirilenko, "Discovering the ecosystem of an electronic financial market with a dynamic machine-learning method," *Algorithmic Finance*, vol. 2, no. 2, pp. 151–165, 2013. DOI: 10.3233/AF-13023.

[42] I. D. Wright, M. Reimherr, and J. Liechty, "A machine learning approach to classification for traders in financial markets," *Stat*, vol. 11, no. 1, e465, 2022. DOI: https://doi.org/10.1002/sta4.465. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/sta4.465. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.465.

[43] R. Ruan, E. Bacry, and J.-F. Muzy, *Liquidity takers behavior representation through a contrastive learning approach*, 2023. arXiv: 2306.05987 [q-fin.ST]. [Online]. Available: https://arxiv.org/abs/2306.05987.

[44] R. Cont, M. Cucuringu, V. Glukhov, and F. Prenzel, "Analysis and modeling of client order flow in limit order markets," *Mathematical Institute, University of Oxford, Department of Statistics, University of Oxford, The Alan Turing Institute, JP Morgan*, Dec. 2021.

[45] Á. Cartea, S. N. Cohen, R. Graumans, S. Labyad, L. Sánchez-Betancourt, and L. van Veldhuijzen, "Statistical predictions of trading strategies in electronic markets," *Journal of Financial Econometrics*, vol. 23, no. 2, nbae025, Oct. 2024, ISSN: 1479-8409. DOI: 10.1093/jjfinec/nbae025. eprint: https://academic.oup.com/jfec/article-pdf/23/2/nbae025/59882367/nbae025.pdf. [Online]. Available: https://doi.org/10.1093/jjfinec/nbae025.

[46] Nasdaq, *Nasdaq guide on order record keeping v.2.3*, Accessed: 2025-02-04, 2022. [Online]. Available: https://www.nasdaq.com/docs/2022/09/16/Nasdaq-Guide-on-Order-Record-Keeping-v.2.3.pdf.

[47] Nasdaq. "Inet nordic trading platform." (n.d.), [Online]. Available: https://www.nasdaq.com/solutions/technical-information-inet-nordic-trading-platform (visited on 02/05/2025).

[48] Nasdaq. "Genium inet trading and clearing platform." (n.d.), [Online]. Available: https://www.nasdaq.com/solutions/technical-information-genium-inet-trading-and-clearing-platform (visited on 02/05/2025).

[49] Nasdaq. "Nordic equity derivatives trading system (ndts)." (n.d.), [Online]. Available: https://www.nasdaq.com/solutions/nordic-equity-derivatives-trading-system (visited on 02/05/2025).

[50] European Commission, *Commission delegated regulation (eu) no .../... supplementing regulation (eu) no 600/2014 of the european parliament and of the council with regard to regulatory technical standards for the maintenance of relevant data relating to orders in financial instruments*, 2016. [Online]. Available: https://ec.europa.eu/finance/securities/docs/isd/mifid/rts/160624-rts-24-annex_en.pdf.

[51] J. G. Dy, "Feature selection in unsupervised learning," in *Encyclopedia of the Sciences of Learning*, N. M. Seel, Ed. Boston, MA: Springer US, 2012, pp. 1278–1282, ISBN: 978-1-4419-1428-6. DOI: 10.1007/978-1-4419-1428-6_97. [Online]. Available: https://doi.org/10.1007/978-1-4419-1428-6_97.

[52] AFM. "Proprietary traders make large-scale use of machine learning in trading algorithms." Accessed: 2025-04-19. (2023), [Online]. Available: https://www.afm.nl/en/sector/actueel/2023/maart/her-machine-learning.

[53] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987, ISSN: 0377-0427. DOI: https://doi.org/10.1016/0377-0427(87)90125-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0377042787901257.

[54] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979. DOI: 10.1109/TPAMI.1979.4766909.

[55] T. Caliński and J. H. and, "A dendrite method for cluster analysis," *Communications in Statistics*, vol. 3, no. 1, pp. 1–27, 1974. DOI: 10.1080/03610927408827101.

A Set of Features

Table 7 Set of features.

Index	Features	description
1	vb_5	The buy and sell volumes available in the order book for the top 5, 10 and 20 levels by the trader. These features reflects the recent activity in the order book by the trader.
2	vb_10	
3	vb_20	
4	vs_5	
5	vs_10	
6	vs_20	
7	vb_5_excl	The buy and sell volumes available in the order book excluding the trader for the top 5, 10 and 20 levels. These features reflects the state of the order book.
8	vb_10_excl	
9	vb_20_excl	
10	vb_5_excl	
11	vb_10_excl	
12	vb_20_excl	
13	Imbalance_5	The imbalance of volumes in the order book for the top 5, 10 and 20 levels from the trader, which indicates the current presence of the trader in the order book.
14	Imbalance_10	
15	Imbalance_20	
16	imbalance_5_excl	The imbalance of volumes in the order book for the top 5, 10 and 20 levels excluding volumes from trader, which indicates the current presence of the members.
17	imbalance_10_excl	
18	imbalance_20_excl	
19	imb_change_5	The change of imbalance of volumes in the order book for the top 5, 10 and 20 levels excluding volumes from the trader. This reflects current state of the order book.
20	imb_change_10	
21	imb_change_20	
22	best_bid_volume	The volume of orders at best bid price available in the order book
23	best_offer_volume	The volume of orders at best offer price available in the order book.
24	spread_bsp	The spread in basis points, the percentage change in the value of financial instruments.
25	return_1	The returns over a number of periods, the periods considered are the last 1, 5, 60 and 300 seconds. These features follows from that the most recent have most predictive power.
26	return_5	
27	return_60	
28	return_300	
29	volatility_1	The volatility over a number of periods. The periods considered for the volatility is the last 1, 5, 60 and 300 minutes. These features follows from that the most recent have most predictive power.
30	volatility_5	
31	volatility_60	
32	volatility_360	
33	num_msg_1s	The number of messages over a number of periods. The number of messages are considered over 1 second, 100 milliseconds, 1 millisecond and 100 microseconds.
34	num_msg_100ms	
35	num_msg_1ms	
36	num_msg_100us	
37	agre_diff_1	The difference between aggressive buys and sells over the previous 1, 5, 60 and 360 seconds. The number of aggressive buys subtracted with number of aggressive sells.
38	agre_diff_5	
39	agre_diff_60	
40	agre_diff_360	
41	agre_buy_1	The number of aggressive buys over the previous 1, 5, 60 and 360 seconds. Agressive orders aims to take liquidity.
42	agre_buy_5	
43	agre_buy_60	
44	agre_buy_360	
45	agre_sell_1	The number of aggressive sells over the previous 1, 5, 60 and 360 seconds. Agressive orders aims to take liquidity.
46	agre_sell_5	
47	agre_sell_60	
48	agre_sell_360	
49	v_last_transaction	The volume of the latest recorded transaction.
50	trader_inv	The inventory of the trader (0 at the start of the trade day).
51	mpid_inv	The inventory of the member (0 at the start of the trade day).
52	trader_cash	The cash expense of the trader(0 at the start of the trade day).
53	mpid_cash	The cash expense of the member (0 at the start of the trade day).

B Statistics for clusters

B.1 Three clusters

Table 8 Statistics for traders in clusters 1, 2, and 3 for VOLVO using k-means. Last four weeks of data.

	Cluster 1	Cluster 2	Cluster 3
Number of traders	11	103	11
Order count	7,717	1,268,364	33,050
Trade count	2,967	107,055	11,154
Order to trade ratio	2.60	11.85	2.96
Order count (%)	0.59	96.89	2.52
Trade count (%)	2.45	88.35	9.20
Liquidity taking trade count	1,340	49,765	4,155
Liquidity taking trade count (%)	2.42	90.06	7.52
Average trader volume in ask side of LOB 0-5	0.35	1.96	0.40
Average trader volume in bid side of LOB 0-5	0.00	1.90	1.18
Average absolute imbalance of traders 0-5	0.20	0.27	0.50
Average ask volume of trader 11-20	0.02	0.68	0.02
Average bid volume of trader 11-20	0.00	0.68	0.05
Average daily directionality	0.64	0.04	0.51
Median resting time (milliseconds)	59	3,304	5,640
Average size of orders	1,284	313	341
Average size of orders SEK	63,331	156,114	63,351
Average size of orders Euro	5,677	13,921	5,636

Table 9 Statistics for traders in clusters 1, 2, and 3 for VOLVO using hierarchical. Last four weeks of data.

	Cluster 1	Cluster 2	Cluster 3
Number of traders	48	33	44
Order count	1,008,359	91,994	208,778
Trade count	81,177	23,268	16,731
Order to trade ratio	12.42	3.95	12.48
Order count (%)	77.03	7.03	15.95
Trade count (%)	66.99	19.20	13.81
Liquidity taking trade count	40,593	10,443	4,224
Liquidity taking trade count (%)	73.46	18.90	7.64
Average trader volume in ask side of LOB 0-5	2.07	0.45	1.75
Average trader volume in bid side of LOB 0-5	2.04	0.76	1.58
Average absolute imbalance of traders 0-5	0.24	0.38	0.38
Average ask volume of trader 11-20	0.77	0.23	0.35
Average bid volume of trader 11-20	0.76	0.23	0.35
Average daily directionality	0.03	0.42	0.15
Median resting time (milliseconds)	3,179	3,256	4,308
Average size of orders	291	380	418
Average size of orders SEK	35,061	255,617	526,774
Average size of orders Euro	3,130	22,816	46,923

Table 10 Statistics for traders in clusters 1, 2, and 3 for VOLVO using spectral. Last four weeks of data.

	Cluster 1	Cluster 2	Cluster 3
Number of traders	90	21	14
Order count	1,248,996	28,147	31,988
Trade count	76,365	9,797	35,014
Order to trade ratio	16.36	2.87	0.91
Order count (%)	95.41	2.15	2.44
Trade count (%)	63.02	8.08	28.90
Liquidity taking trade count	17,994	3,244	34,022
Liquidity taking trade count (%)	32.56	5.87	61.57
Average trader volume in ask side of LOB 0-5	1.98	0.69	0.03
Average trader volume in bid side of LOB 0-5	1.96	0.31	0.03
Average absolute imbalance of traders 0-5	0.27	0.36	0.19
Average trader volume in ask side of LOB 11-20	0.69	0.32	0.03
Average trader volume in bid side of LOB 11-20	0.68	0.55	0.03
Average daily directionality	0.03	0.67	0.14
Median resting time (milliseconds)	3,413	1,487	100
Average size of orders	311	988	37
Average size of orders SEK	105,861	828,661	40,122.
Average size of orders Euro	9,448	73,804	3,584

B.2 Four clusters

Table 11 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using k-means. Last four weeks of data.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	10	79	30	6
Order count	30,700	1,228,782	44,095	5,554
Trade count	10,407	91,918	17,101	1,750
Order to trade ratio	2.95	13.37	2.58	3.17
Order count (%)	2.35	93.86	3.37	0.42
Trade count (%)	8.59	75.85	14.11	1.44
Liquidity taking trade count	3,896	43,646	6,809	909
Liquidity taking trade count (%)	7.05	78.98	12.32	1.64
Average trader volume in ask side of LOB 0-5	0.42	2.00	0.60	0.25
Average trader volume in bid side of LOB 0-5	1.25	1.95	0.30	0.00
Average absolute imbalance of traders 0-5	0.52	0.26	0.35	0.20
Average ask volume of trader 11-20	0.02	0.70	0.22	0.02
Average bid volume of trader 11-20	0.05	0.69	0.37	0.00
Average daily directionality	0.48	0.03	0.40	0.78
Median resting time (milliseconds)	6,069	3,303	2,999	0
Average size of orders	342	301	707	492
Average size of orders SEK	53,023	94,354	481,132	88,459
Average size of orders Euro	4,715	8,424	42,847	7,934

Table 12 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using hierarchical. Last four weeks of data.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	39	33	44	9
Order count	499,777	91,994	208,778	508,582
Trade count	73,699	23,268	16,731	7,478
Order to trade ratio	6.78	3.95	12.48	68.01
Order count (%)	38.22	7.04	15.97	38.90
Trade count (%)	60.91	19.23	13.83	6.18
Liquidity taking trade count	40,269	10,443	4,224	324
Liquidity taking trade count (%)	72.89	18.89	7.64	0.59
Average trader volume in ask side of LOB 0-5	1.65	0.45	1.75	2.49
Average trader volume in bid side of LOB 0-5	1.59	0.76	1.58	2.48
Average absolute imbalance of traders 0-5	0.22	0.38	0.38	0.25
Average ask volume of trader 11-20	0.58	0.23	0.35	0.95
Average bid volume of trader 11-20	0.56	0.30	0.34	0.95
Average daily directionality	0.04	0.42	0.15	0.03
Median resting time (milliseconds)	5,533	3,256	4,308	1,988
Average size of orders	295	380	418	289
Average size of orders SEK	36,272	255,617	526,774	23,126
Average size of orders Euro	3,238	22,816	46,923	2,062

Table 13 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using spectral. Last four weeks of data.

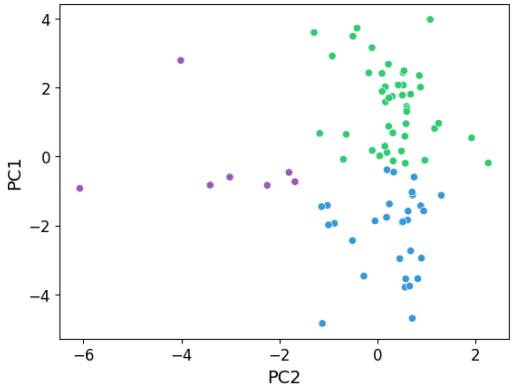
	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	22	13	18	72
Order count	63,383	31,694	27,262	1,186,792
Trade count	34,145	9,539	61,972	6,550
Order to trade ratio	1.86	3.32	0.44	181.19
Order count (%)	4.84	2.42	2.08	90.65
Trade count (%)	30.43	8.50	55.23	5.84
Liquidity taking trade count	6,550	33,406	3,001	12,303
Liquidity taking trade count (%)	11.85	60.45	5.43	22.26
Average trader volume in ask side of LOB 0-5	0.29	0.03	0.71	2.07
Average trader volume in bid side of LOB 0-5	0.87	0.03	0.32	2.01
Average absolute imbalance of traders 0-5	0.41	0.19	0.36	0.72
Average ask volume of trader 11-20	0.01	0.03	0.33	0.72
Average bid volume of trader 11-20	0.02	0.03	0.57	0.71
Average daily directionality	0.44	0.14	0.68	0.03
Median resting time (milliseconds)	3,502	85	1,480	3,407
Average size of orders	301	37	1,021	311
Average size of orders SEK	51,318	810	850,943	139,838
Average size of orders Euro	4,567	72	75,789	12,485

C Results for price

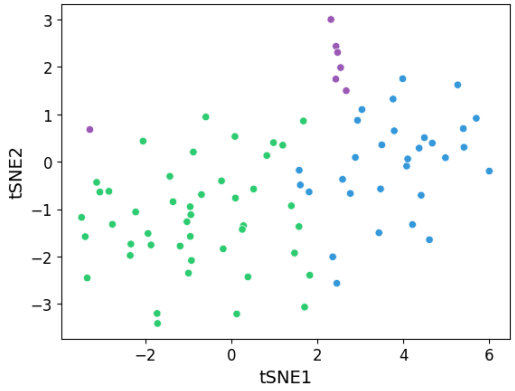
C.1 Three clusters

Table 14 Statistics for traders in clusters 1, 2, and 3 for VOLVO using k-means. Last four weeks of data.

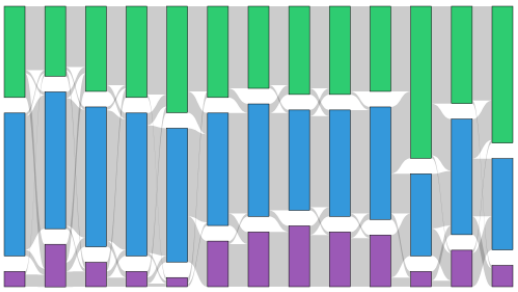
	Cluster 1	Cluster 2	Cluster 3
Number of traders	30	45	7
Order count	941,354	146,406	18,599
Trade count	45,089	72,566	1,082
Order to trade ratio	20.88	2.02	17.19
Order count (%)	85.07	13.23	1.68
Trade count (%)	38.03	61.21	0.91
Liquidity taking trade count	11,172	42,177	592
Liquidity taking trade count (%)	20.72	78.18	1.10
Average trader volume in ask side of LOB 0-5	2.28	1.48	0.14
Average trader volume in bid side of LOB 0-5	2.26	1.32	0.05
Average absolute imbalance of traders 0-5	0.25	0.34	0.23
Average ask volume of trader 11-20	0.83	0.19	0.92
Average bid volume of trader 11-20	0.83	0.11	1.13
Average daily directionality	0.04	0.09	0.27
Median resting time (milliseconds)	1,895	3,965	21,188
Average size of orders	313	295	950
Average size of orders SEK	196,969	104,787	333,843
Average size of orders Euro	17,546	9,357	29,693



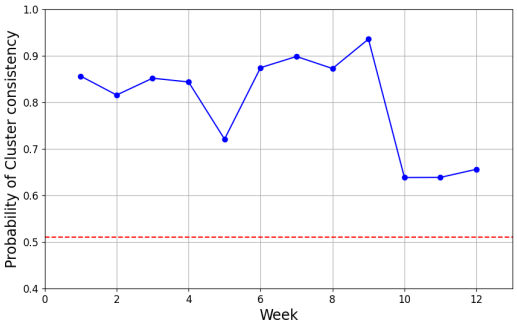
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

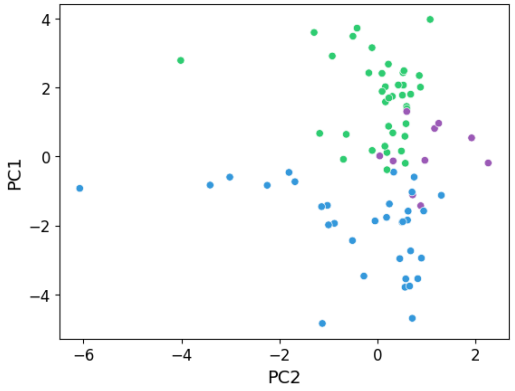


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

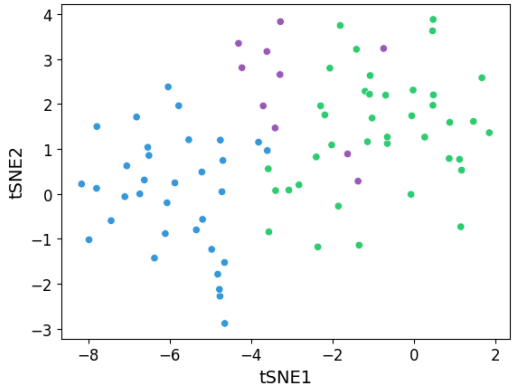
Figure 11: Results of k-means with three clusters.

Table 15 Statistics for traders in clusters 1, 2, and 3 for VOLVO using hierarchical. Last four weeks of data.

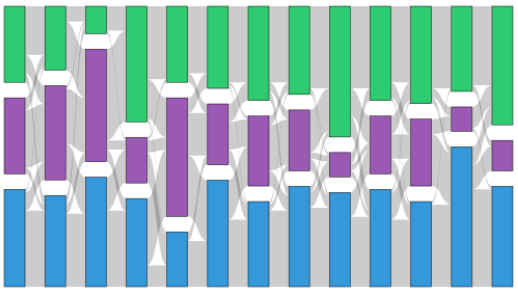
	Cluster 1	Cluster 2	Cluster 3
Number of traders	39	33	10
Order count	69,792	929,747	106,820
Trade count	48,071	35,500	35,166
Order to trade ratio	1.45	26.19	3.04
Order count (%)	6.31	84.03	9.66
Trade count (%)	40.50	29.91	29.59
Liquidity taking trade count	41,230	8,854	3,857
Liquidity taking trade count (%)	76.44	16.41	7.15
Average trader volume in ask side of LOB 0-5	0.16	2.29	2.06
Average trader volume in bid side of LOB 0-5	0.08	2.28	1.86
Average absolute imbalance of traders 0-5	0.21	0.25	0.43
Average ask volume of trader 11-20	0.26	0.83	0.33
Average bid volume of trader 11-20	0.30	0.82	0.29
Average daily directionality	0.14	0.04	0.13
Median resting time (milliseconds)	16,178	1,912	3,437
Average size of orders	413	306	391
Average size of orders SEK	55,516	34,695	368,138
Average size of orders Euro	4,957	3,096	32,819



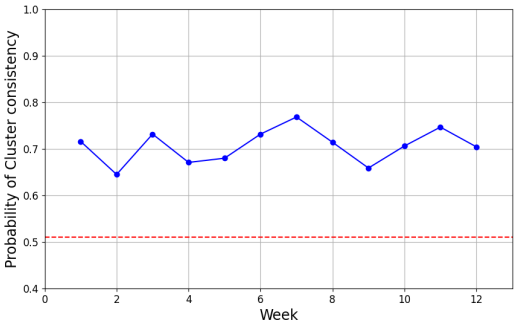
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

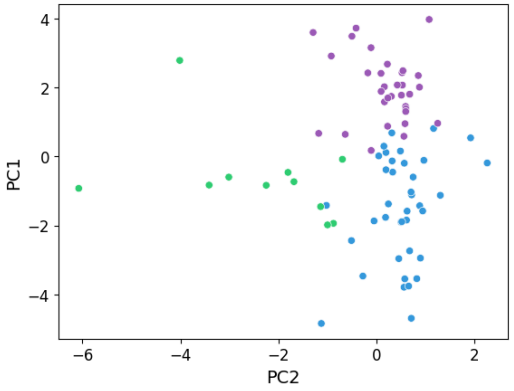


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

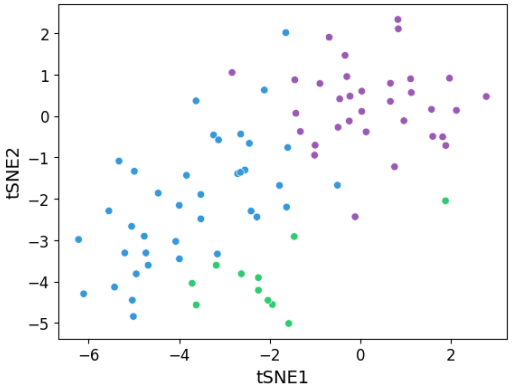
Figure 12: Results of hierarchical with three clusters.

Table 16 Statistics for traders in clusters 1, 2, and 3 for VOLVO using spectral. Last four weeks of data.

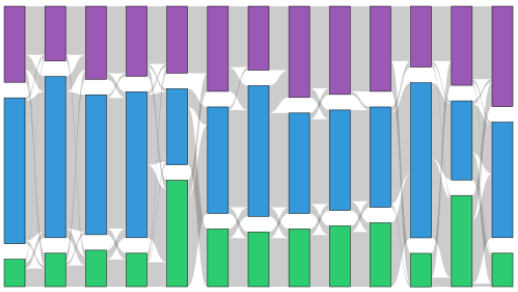
	Cluster 1	Cluster 2	Cluster 3
Number of traders	33	11	38
Order count	50,620	22,916	1,032,823
Trade count	40,434	2,054	76,249
Order to trade ratio	1.25	11.16	13.54
Order count (%)	4.57	2.07	93.36
Trade count (%)	34.08	1.73	64.19
Liquidity taking trade count	37,336	759	15,846
Liquidity taking trade count (%)	69.19	1.41	29.36
Average trader volume in ask side of LOB 0-5	0.19	0.11	2.27
Average trader volume in bid side of LOB 0-5	0.06	0.04	2.24
Average absolute imbalance of traders 0-5	0.21	0.22	0.27
Average ask volume of trader 11-20	0.02	0.75	0.78
Average bid volume of trader 11-20	0.00	0.91	0.77
Average daily directionality	0.16	0.28	0.03
Median resting time (milliseconds)	2,410	21,041	1,995
Average size of orders	351	814	309
Average size of orders SEK	60,351	179,475	184,100
Average size of orders Euro	5,388	15,961	16,415



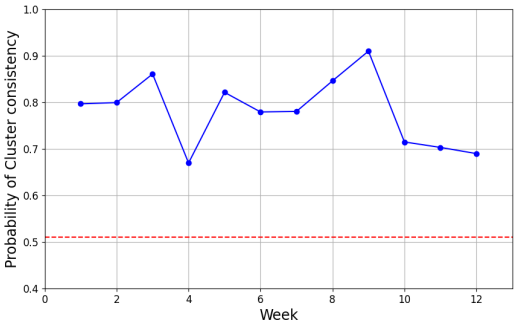
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).



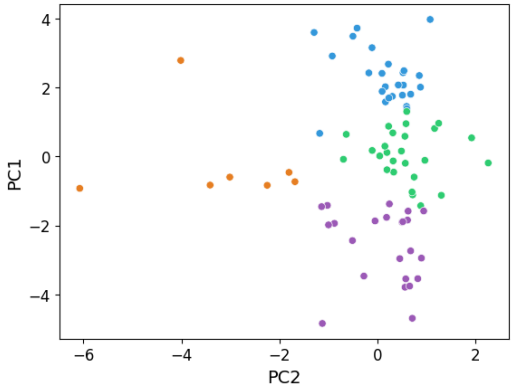
(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

Figure 13: Results of spectral with three clusters.

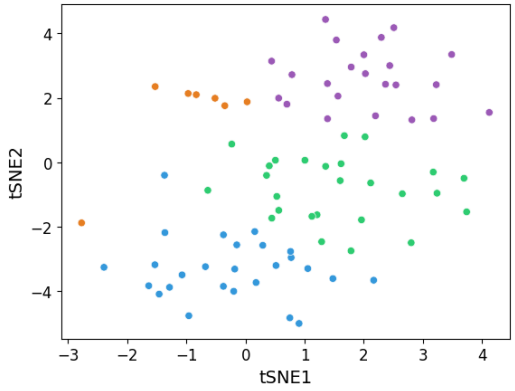
C.2 Four clusters

Table 17 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using k-means. Last four weeks of data.

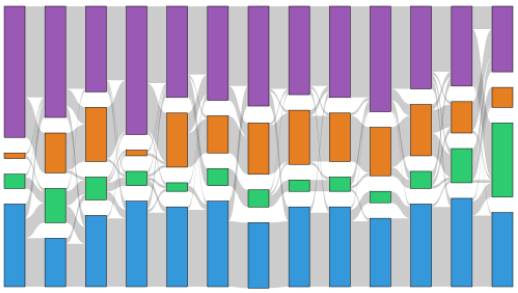
	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	7	23	26	26
Order count	18,599	919,021	131,175	37,564
Trade count	1,082	33,042	49,740	34,873
Order to trade ratio	17.19	27.81	2.64	1.08
Order count (%)	1.68	82.87	11.83	3.39
Trade count (%)	0.91	27.82	41.89	29.37
Liquidity taking trade count	592	7,978	12,749	32,622
Liquidity taking trade count (%)	1.10	14.77	23.60	60.39
Average trader volume in ask side of LOB 0-5	0.14	2.32	1.70	0.23
Average trader volume in bid side of LOB 0-5	0.05	2.30	1.55	0.05
Average absolute imbalance of traders 0-5	0.23	0.25	0.39	0.21
Average ask volume of trader 11-20	0.92	0.84	0.27	0.03
Average bid volume of trader 11-20	1.13	0.83	0.24	0.00
Average daily directionality	0.27	0.04	0.13	0.15
Median resting time (milliseconds)	21,188	1,864	4,074	787
Average size of orders	950	298	365	420
Average size of orders SEK	333,843	25,716	268,420	65,500
Average size of orders Euro	29,693	2,296	23,931	5,848



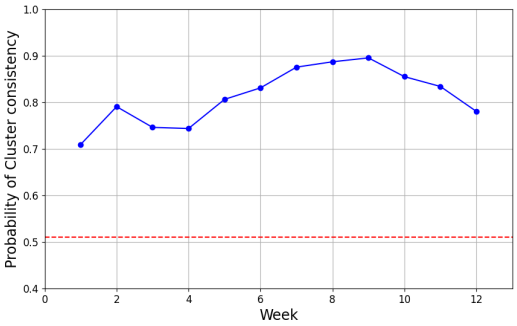
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

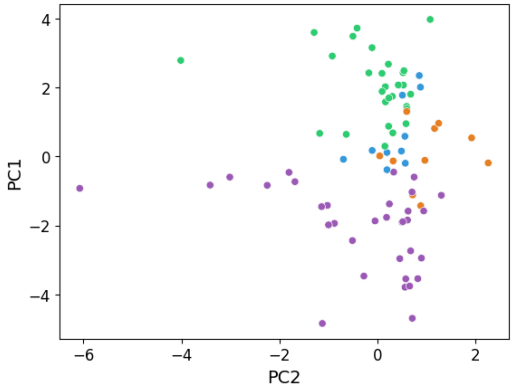


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

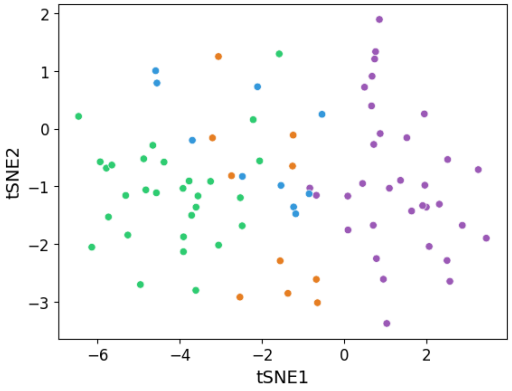
Figure 14: Results of k-means with four clusters.

Table 18 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using hierarchical. Last four weeks of data.

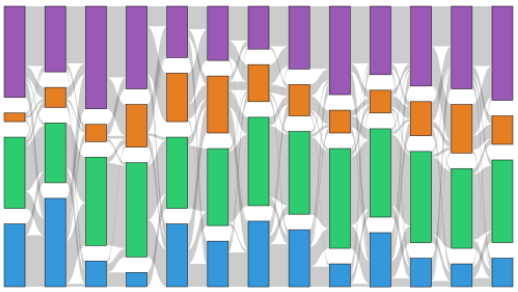
	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	33	10	10	29
Order count	929,747	26,534	106,820	43,258
Trade count	35,500	23,194	35,166	24,877
Order to trade ratio	26.19	1.14	3.04	1.74
Order count (%)	83.83	2.39	9.63	3.90
Trade count (%)	29.93	19.55	29.64	20.97
Liquidity taking trade count	8,854	20,365	3,857	20,865
Liquidity taking trade count (%)	16.40	37.73	7.14	38.66
Average trader volume in ask side of LOB 0-5	2.29	0.05	2.06	0.24
Average trader volume in bid side of LOB 0-5	2.28	0.06	1.86	0.09
Average absolute imbalance of traders 0-5	0.25	0.19	0.43	0.22
Average ask volume of trader 11-20	0.83	0.01	0.33	0.41
Average bid volume of trader 11-20	0.82	0.00	0.29	0.49
Average daily directionality	0.04	0.10	0.13	0.20
Median resting time (milliseconds)	1,912	14,583	3,437	16,925
Average size of orders	306	72	391	621
Average size of orders SEK	34,695	3,729	368,138	10,3800
Average size of orders Euro	3,096	333	32,819	9,268



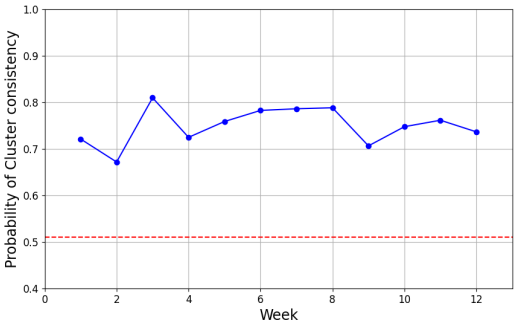
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).

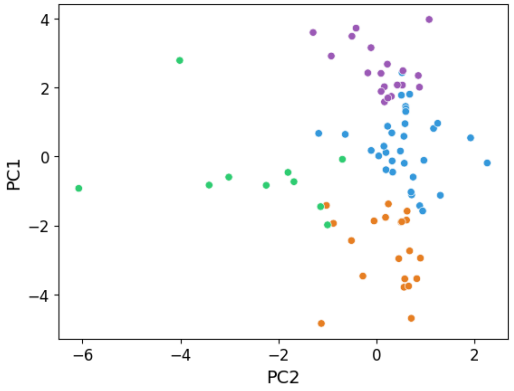


(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

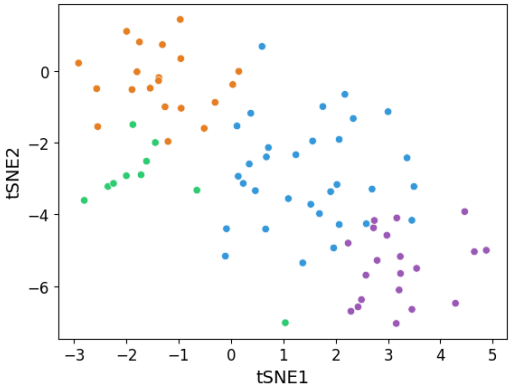
Figure 15: Results of hierarchical with four clusters.

Table 19 Statistics for traders in clusters 1, 2, 3 and 4 for VOLVO using spectral. Last four weeks of data.

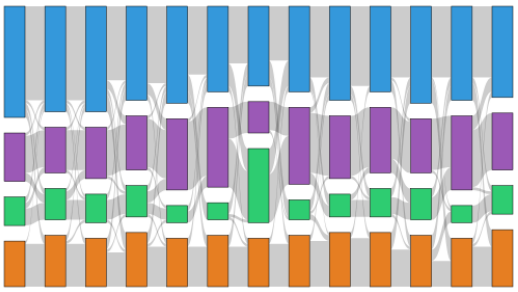
	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Number of traders	20	20	10	32
Order count	917,310	16,243	19,029	153,777
Trade count	31,387	19,596	1,321	66,433
Order to trade ratio	29.23	0.83	14.41	2.32
Order count (%)	82.94	1.47	1.72	13.90
Trade count (%)	26.36	16.45	1.11	55.78
Liquidity taking trade count	7,124	18,867	701	27,249
Liquidity taking trade count (%)	13.20	34.95	1.30	50.47
Average trader volume in ask side of LOB 0-5	2.32	0.00	0.13	1.51
Average trader volume in bid side of LOB 0-5	2.31	0.01	0.05	1.33
Average absolute imbalance of traders 0-5	0.25	0.16	0.23	0.37
Average ask volume of trader 11-20	0.84	0.00	0.90	0.24
Average bid volume of trader 11-20	0.83	0.00	1.10	0.21
Average daily directionality	0.04	0.19	0.27	0.13
Median resting time (milliseconds)	1,859	12,129	21,172	3,893
Average size of orders	296	58	932	422
Average size of orders SEK	51,318	810	850,943	139,838
Average size of orders Euro	4,567	72	75,789	12,485



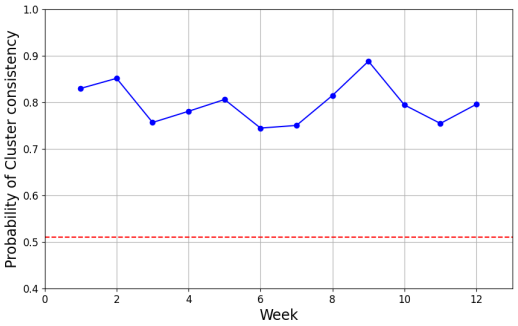
(a) Cluster visualization with PCA for week 13.



(b) Cluster visualization with t-SNE for week 13.



(c) Cluster transition over 13 clustering exercises (each column represents 4 weeks of trading data, i.e., weeks 1-4, 2-5, etc.).



(d) Stability of clusters for VOLVO across the thirteen clustering exercises.

Figure 16: Results of spectral with four clusters