Master Thesis Business Analytics

Clustering companies based on similar financial profiles using XBRL data

Can clustering algorithms be used to gain insights about the data consistency and industry trends.

Author: Dalinio van Kempen (2636471)

Company supervisors: René van der Meij & Ralph Verhelst
First supervisor: Paulo Jorge de Andrade Serra
Second reader: Guszti Eiben

March 7, 2025

Clustering companies using XBRL data

Dalinio van Kempen

Master Thesis Report

Vrije Universiteit Amsterdam

Faculty of Science

Business Analytics

De Boelelaan 1081a

1081 HV Amsterdam

Host Organisation:

SBR Nexus

Hollandse Kade 25

1391 JD Abcoude

March 7, 2025

Preface

This report is the final deliverable of my research project, conducted as part of the Master Project Business Analytics at the Vrije Universiteit Amsterdam. The purpose of this study is to explore how clustering algorithms can be applied to XBRL-based financial data to group companies according to their financial characteristics. By analyzing different clustering techniques, this research aims to provide insight into the handling of data inconsistencies and the detection of financial patterns.

This research was conducted in collaboration with SBR Nexus, specifically within the "Bedrijfsdata" department. SBR Nexus plays a key role in the publication and management of financial data standards in the Netherlands and is responsible for the efficient exchange of business data between government, businesses, and financial institutions.

I would like to express my sincere gratitude to my university supervisor, Prof. Andrade Serra, P.J. de. I really appreciate our insightful meetings and his continuous guidance and valuable feedback throughout this project.

I also extend my thanks to my supervisors at SBR Nexus, René van der Meij and Ralph Verhelst, for their support and practical expertise. I am grateful for the opportunity to conduct this research at SBR Nexus.

Summary

This research investigates the use of clustering algorithms to analyze XBRL-based financial data and group companies based on financial similarities. The main objective is to explore how different clustering techniques handle the challenges of financial data inconsistencies in large datasets. By comparing Hierarchical Clustering, Kmedoids, and Random Forest clustering, this study evaluates how well these methods capture financial patterns and industry trends.

Research Structure & Approach The central research question guiding this study is: How can clustering algorithms be used to group companies with similar financial profiles using XBRL data, and what insights can we gain about data consistency and industry trends across these clusters? A key challenge in financial data clustering is the presence of missing values and inconsistencies across different company reports. To address this, custom distance functions and imputation techniques are implemented to ensure meaningful similarity calculations between companies.

Next, three clustering methods are applied. Hierarchical clustering is tested for its ability to create well-separated groups, while Kmedoids is evaluated for its stability and interpretability. Additionally, Random Forest clustering is explored as a scalable alternative that avoids explicit distance calculations. The results of each method are analyzed using Silhouette scores and visualizations, offering insights into how well financial patterns are captured.

Beyond technical analysis, broader implications of financial clustering are discussed, considering its potential applications in banking, risk assessment, and financial regulation. Finally, challenges such as scalability and data quality issues are addressed, with recommendations for future improvements.

Key Findings One of the most significant findings of this research is the impact of data inconsistencies on clustering outcomes. Many companies fail to report specific financial facts, leading to gaps in datasets that must be handled carefully. A custom distance function was introduced to calculate distances based on only shared features between companies. Without this approach, missing data would have distorted clustering results.

Regarding algorithm performance, Hierarchical Clustering produced well-separated clusters, but can be computationally expensive for large datasets. Kmedoids proved to be interpretable but sensitive to initial medoid selection and outliers, leading to potential variations in outcomes. Random Forest clustering, on the other hand, handled large datasets efficiently but required complete financial data, making it less effective when dealing with missing values.

Another crucial insight is that sector-based classifications alone do not necessarily reflect financial similarity. While companies within the same industry share certain financial structures, their financial ratios and business models can vary widely. This suggests that clustering models should incorporate more nuanced financial indicators beyond simple industry classifications.

Conclusions & Recommendations Clustering techniques offer a powerful way to extract insights from financial data, but their effectiveness depends on data quality and pre-processing strategies. Inconsistencies in XBRL filings remain a major challenge, and improving automated data validation and correction methods would enhance clustering accuracy. Financial institutions and regulators could benefit from refining XBRL standards to reduce variability in reporting.

In practical terms, banks and financial analysts could leverage clustering for risk assessment, fraud detection, and financial benchmarking. However, for real-world applications, future research should explore more strategies to minimize the impact of missing data.

Ultimately, while clustering provides valuable insights into financial patterns, its full potential will only be realized if data reliability challenges are addressed.

Contents

1 Introduction

The evolution of financial reporting has been significantly shaped by advancements in data standardization, particularly with the adoption of the eXtensible Business Reporting Language (XBRL). XBRL is designed to simplify the exchange of financial information by providing a standardized format for reporting, which improves both transparency and efficiency. Research by [7] has already highlighted the potential of XBRL to improve financial reporting processes, emphasizing its importance in the 21st century. Therefore, governments, businesses, and financial institutions around the world increasingly rely on XBRL to ensure the accurate and timely exchange of financial data. Despite its growing adoption, there are still concerns regarding the quality, consistency, and accuracy of the data being reported. Inconsistent tagging practices, errors in financial metrics, and variations in how companies implement XBRL standards present significant challenges to regulators, analysts, and organizations, striving to ensure accurate and efficient exchange of business data.

1.1 Host Organization: SBR Nexus

The host organization for this research is SBR Nexus, an entity that plays a key role in the publication and management of financial data standards in the Netherlands. SBR Nexus is responsible for the efficient exchange of business data between government, businesses, and financial institutions. They work closely with key Dutch banks such as ING, Rabobank, and ABN AMRO to ensure that financial reporting standards are adhered to and that data exchanges are reliable, secure and in compliance with regulatory requirements.

Given SBR Nexus's role as a key player in data standardization, the goal of this research will be to contribute to improving the quality of financial data reporting. By focusing on the consistency of the XBRL data, the findings of this study could lead to more reliable financial reporting processes and help SBR Nexus refine the standards. Furthermore, by exploring the use of big data frameworks for efficient data handling, this research could provide SBR Nexus with the technical capabilities to process larger datasets as more companies adopt XBRL.

1.2 Motivation for study

SBR Nexus, a key player in the Dutch financial reporting ecosystem, publishes various data standards to streamline the sharing of business data across sectors. In collaboration with the government and major financial institutions such as ING, Rabobank and ABN AMRO, SBR Nexus provides data products for annual reports, appraisals, and rental lists, with the goal of ensuring safe, efficient and accurate data exchange. However, ensuring that all companies consistently adhere to XBRL standards is a persistent challenge. The accuracy and reliability of XBRL data directly affect how well stakeholders can assess financial

performance, manage risks, and maintain compliance with regulatory requirements.

Despite the technical benefits of XBRL, one possible challenge is the potential for inconsistencies in how companies report financial data. Even small discrepancies in tagging or data formatting can result in inaccurate financial analyses, making it difficult to draw meaningful comparisons between companies or sectors. Identifying common inaccuracies and inconsistencies in XBRL data, as well as strategies to mitigate these issues, is crucial for the improvement of financial reporting. Furthermore, as the volume of financial data increases, there is a growing need for efficient and scalable methods to process, analyze, and derive insights from these large datasets, but the application of big data technology in the field of XBRL financial reporting also has a strong necessity and feasibility [23].

Given the importance of data quality and consistency in financial reporting, this research seeks to explore how clustering algorithms and big data processing frameworks can be applied to handle inconsistencies in XBRL data and group companies based on their financial profiles. This will not only help improve the reliability of financial reporting, but will also provide insights into broader industry trends, enabling better decision-making for financial institutions and regulators.

1.3 Research Questions

The core research question that guides this study is: How can clustering algorithms be used to group companies with similar financial profiles using XBRL data, and what insights can we gain about data consistency and industry trends across these clusters? This question can be broken down into the following sub-questions:

1. To what extent are there inconsistencies or inaccuracies in XBRL data across different companies?
    - This sub-question aims to identify specific areas where errors or discrepancies occur in financial reporting, such as not reporting certain facts or improper use of XBRL tags.
2. How can big data processing frameworks be leveraged to handle large volumes of XBRL data efficiently and effectively?
    - The focus here is on identifying scalable technologies that can manage and analyze increasing datasets as more companies adopt XBRL reporting, ensuring that the analysis remains efficient and actionable.
3. What patterns or trends can be identified by applying clustering algorithms to the financial data of companies using XBRL?

- This will involve analyzing how clustering methods can group companies with similar financial metrics and exploring the insights these groupings provide about industry performance, risk, and reporting behavior.

1.4 Thesis Outline

The remainder of this thesis is structured as follows. Chapter 2 provides a context description, outlining the background and relevance of the study. This chapter situates the research within the broader landscape of financial reporting and data analytics, highlighting key challenges associated with XBRL data. Chapter 3 presents a review of relevant literature, focusing on XBRL data quality, inconsistencies in financial reporting, and the use of machine learning techniques in financial analysis. This chapter establishes the technical and theoretical foundation for the approaches utilized in this research. Chapter 4 delves into data analysis, describing the process of collecting and preprocessing XBRL data. It outlines the application of big data frameworks to enable scalable and efficient data processing, essential for handling the complexity of XBRL datasets. Chapter 5 explains the methodology and statistical techniques used to implement the clustering algorithms. This chapter lays out the methodological framework that underpins the study. Chapter 6 presents the results of the analysis. It includes findings on the consistency and reliability of XBRL data, insights from clustering analyses, and a closer examination of specific clusters. Chapter 7 offers a discussion of the results, interpreting the key findings in relation to the research questions. It also explores the business conclusions for SBR Nexus and the broader financial reporting ecosystem. Finally, Chapter 8 concludes the thesis by presenting implications associated with the consistency and processing of XBRL data. It also outlines the potential for future research. Through this structure, the thesis aims to systematically address the challenges associated with XBRL data and propose practical solutions that can be applied in the future.

2 Context Description

This section provides an overview of the broader context of XBRL and how it is situated in this research.

The adoption of XBRL has brought a transformative shift to the landscape of financial reporting. XBRL provides a standardized, machine-readable format for reporting financial data, enhancing transparency, accuracy, and comparability across companies and industries. This digital evolution enables stakeholders such as investors, analysts, regulators, and the public to access granular financial data more efficiently [9] [10] [26]. Since the 1990s, the internet revolutionized financial reporting by allowing companies to transition from traditional paper-based reports to online financial disclosures, including XBRL-formatted reports [16]. However, despite the enhanced availability of financial disclosures, challenges remain in seamlessly integrating this information into analytical tools [2].

Initially, XML (eXtensible Markup Language) was proposed as a method for tagging financial data to automate information retrieval. However, the lack of standardization in XML labels limited the benefits of automated web retrieval systems. Companies could create their own tags, making it difficult to compare data across businesses [7]. XBRL, a more specialized markup language, resolves this issue by introducing standardized tags for business reporting, enabling a consistent structure that facilitates automated data extraction and analysis.

Nevertheless, XBRL adoption is not without its challenges. While it facilitates machine-readability and data standardization, the success of XBRL in delivering consistent data depends on how strictly companies adhere to the standard. Variations in tagging practices can still lead to inconsistencies, data inaccuracies and costs, particularly in markets where public information is less robust [10]. Which all undermine the comparability of financial reports. Tohang et al., 2020 [20] highlights another significant challenge: the asymmetry in the transferability of online financial disclosures. This complicates the task of aggregating data from different sources.

The advantages of XBRL extend beyond financial markets. As a language that enables businesses to encode and decode their financial data in a standardized format, XBRL ensures that financial reports are easily accessible and comprehensible by machines, which reduces the risk of human error. This automation can be streamlined in a reporting process, lowering administrative costs and improving the efficiency of data handling in corporate environments [21]. Furthermore, XBRL promotes financial transparency by democratizing access to financial data, allowing a wide range of stakeholders to interpret and use this information for decision-making [13].

3 Literature Review

This section explores the literature regarding: Random Forests (RFs), clustering techniques, and imputation methods to handle missing data and improve financial analysis.

As mentioned earlier, the adoption of XBRL has revolutionized financial data analysis by providing a standardized, machine-readable format for reporting. However, leveraging this structured data to derive actionable insights requires advanced analytical techniques. Integrating machine learning methods, such as Random Forests (RFs) and clustering models, introduces new possibilities for processing and interpreting complex financial datasets.

Random Forests (RFs) are ensemble learning methods that combine multiple decision trees to enhance predictive performance. They are widely recognized as robust tools in pattern recognition, particularly excelling in classification and regression tasks due to their ability to reduce overfitting and improve generalizability [5]. RFs work by aggregating the outputs of individual decision trees, each trained on a bootstrapped subset of the data, and by considering a random subset of features at each split, ensuring diversity among the trees [5]. This characteristic makes RFs particularly robust for analyzing structured datasets like those formatted in XBRL.

Like classification, cluster analysis groups similar data objects into clusters [18]. Clustering analysis is a useful starting point for purposes such as data summarization. A cluster of data objects can be considered as a form of data compression [6].

Despite both their prominences in supervised learning contexts, the application of RFs in unsupervised tasks such as clustering has been less explored. Distance-based RF clustering methods address this gap by deriving meaningful similarity measures between data points from RF proximity metrics [17, 19]. Proximity scores are calculated based on how often data points end up in the same leaf node across all trees in the forest, providing a measure of similarity.

Once a proximity matrix is constructed, it can serve as input for conventional clustering algorithms, such as:

- Hierarchical Clustering, a method that groups data points based on nested relationships, enabling the discovery of multi-level structures within datasets [14].
- Spectral Clustering, A graph-based technique that partitions data using eigenvectors of the similarity matrix, which has been shown to be effective in high-dimensional datasets [15, 12].

Such methods have been useful for analyzing structured financial data and demonstrated how RF proximity metrics can cluster financial reports, uncovering trends and anomalies across industries.

Innovations in Unsupervised RF Techniques Building on the foundational work of Breiman, researchers have developed methods to expand the unsupervised capabilities of RFs.

Notable innovations include:

- Extremely Randomized Trees (ExtraTrees): Geurts et al. (2006) introduced ExtraTrees, which increase the randomness of splits by selecting thresholds randomly instead of optimizing them based on the data. This approach has been shown to enhance computational efficiency and robustness in clustering applications [8].
- Isolation Forests: Liu et al. (2008) proposed Isolation Forests as an anomaly detection method that isolates outliers by recursively partitioning the data. Anomalies, being few and different, require fewer splits to isolate, making this technique ideal for detecting irregularities. [11].

When applied to XBRL, RF-based clustering provides valuable insights into financial behaviors and reporting patterns. These methods can group companies with similar financial characteristics, facilitating cross-sectional analyses and trend identification. By automating these processes, RF clustering enhances efficiency, reduces manual oversight, and ensures consistency across datasets while advancing the goals of transparency and comparability inherent in XBRL.

One main problem of the RF-based clustering model is that it needs a complete dataset to function properly. And missing data is a common problem in financial data analysis. The absence of crucial financial data points can lead to biased insights [22].

There are various approaches to handle missing data, and the most appropriate method depends on the specific circumstances. Common techniques include:

- Imputation: Replacing missing values with estimates, such as mean, regression, or multiple imputation.
- Deletion: Removing incomplete cases, effective when missing data is minimal and does not bias results.
- Modeling: Using statistical models to address missing data, suitable for larger amounts or non-random patterns.

According to the findings of Strike et al. (2001) and Raymond and Roberts (1987), when datasets have a low proportion of missing data, typically less than 10% to 15% of the entire dataset, it is generally acceptable to simply remove the

incomplete entries. Such removal is unlikely to have a significant impact on the overall analytical results. However, when the missing data rate exceeds 15%, a more cautious approach is required to address the issue effectively (Acuna & Rodriguez, 2004; Lin and Tsai, 2021).

Another commonly used method is the mean-mode method. This method assigns numerical missing values with the attribute's mean and replaces a categorical missing value with the most frequently occurring value [4]. This method is the most common method used in research [3]. However, researchers claim that the mean-mode method is the worst possible option, regardless of the amount missing, because it artificially minimizes the dataset's standard deviation [1].

When working with multivariate data, more advanced imputation methods, such as iterative imputation, have shown the potential to yield better results compared to simpler techniques. These methods leverage the relationships between features in the dataset, using the available information in other variables to estimate missing values more accurately.

While iterative imputation methods such as Multiple Imputation by Chained Equations (MICE) have been widely discussed and implemented in various fields[24, 25], there remains a small research gap in their application to specific contexts like financial reporting or other highly structured datasets, such as XBRL.

Overall, XBRL has been instrumental in transforming financial reporting into a digital, transparent, and efficient process, yet significant work remains to standardize its application and ensure that its full potential is realized. This research seeks to further explore these issues by examining the consistency and reliability of XBRL data in the context of data integration and clustering algorithms, which will offer insights into how these tools can help organizations and stakeholders with challenges posed by inconsistent financial reporting.

4 Data Analysis

The data analysis section details the methodology employed for processing and analyzing the dataset. It begins with extraction in Section 4.1. Next, it discusses structuring, cleaning, and imputation, and financial metric calculations in Section 4.2. Finally, data exploration employs visualizations in Section 4.3.

4.1 JSON Data Collection and Extraction

This subsection explains how data was collected and parsed from JSON files, ensuring accessibility and consistency for subsequent processing.

One of the main challenges encountered during this project was the lack of a clean, pre-structured database for financial data. Instead of working with an existing dataset, data gathering was required from publicly available sources. The raw data was scattered across various JSON files on filings.xbrl.org. Manual collection would have been inefficient and prone to errors. This led to the development of an automated process to both gather and clean the data, ensuring it could be used effectively for financial analysis.

This section contains the outline of a process that involves the two main phases: (1) gathering financial data from a website using web scraping techniques, and (2) extracting key financial information from the collected data stored in JSON format. The workflow described here automates both the data collection and processing steps, making it highly efficient for large datasets.

Phase 1: Gathering JSON Files from the website. The first phase of this process involves collecting JSON files from a financial reporting website, called filings.xbrl.org. These files are publicly available, but manual downloading while maintaining a usable structure for numerous companies would be very inefficient. To address this, the script leverages a package called Selenium, a web automation tool, which navigates the website and download the necessary JSON files for each company listed.

Setting up Selenium allows it to simulate a browser and interact with the website just as a human user would. Selenium automates the process of loading the website, navigating to the specific country (in this case, the Netherlands), and applying filters to display relevant companies. Once the script reaches a company's detailed page, it downloads the relevant files and saves it locally.

Phase 2: Extracting and Processing Financial Data from JSON Files. Once the JSON files are downloaded, the second phase of the script focuses on extracting the financial data from each file. The goal is to retrieve the financial figures (e.g., revenue, net income) and store them in a structured format for further analysis. The script iterates through each file in the directory and converts the raw JSON data into Python-readable structures, which makes it easier to

manipulate and process. The JSON files also contain a lot of fields which are not usable for this research, therefore the script only extracts the value fields, which contains the actual numerical data of interest.

After processing the JSON files, the script generates a dataset containing the extracted financial data. However, the dataset remains unclean, unstructured, and not yet ready for immediate use. Significant cleaning, organization, and analysis are required before proceeding to the modeling phase.

4.2 Structuring, cleaning and imputation

Here, the focus is on preparing the dataset by addressing inconsistencies, filling missing values, and organizing the data for analysis.

The first step in the methodology will be the use of a Random Forest model. They work best when features are provided as distinct columns. Pivoting ensures that each financial entry becomes a separate feature (column), making the data more structured for the model. By pivoting, a dataset is created where each column corresponds to a specific financial entry. This makes it easier to interpret feature importance scores generated by the Random Forest model. The resulting pivot table now makes it also easier for cleaning, imputing, and feature engineering, ensuring better pre-processing for the model.

Cleaning was particularly necessary to address instances of duplicate stored values. Annual reports typically include data for both the current and the previous year. However, when a company has multiple annual reports stored in the database, duplicate values may appear. These duplicates must either be aggregated or removed, depending on whether they provide additional or updated information compared to one another.

To ensure the data's usability, specific adjustments were needed to handle non-reported entries of critical financial columns. Because, certain financial metrics were often incomplete or absent. For instance, values like "Current Assets" and "Current Liabilities" were reconstructed using related attributes when missing, ensuring that these foundational elements were accurately represented. Similarly, other financial metrics were imputed through a hierarchy of related columns to capture its most accurate representation or imputed based on logical relationships between related data points. This iterative imputation ensured that even when primary values were missing, alternative sources provided a reliable fallback, preserving the integrity of the dataset for subsequent analysis.

For consistency in the dataset, it was necessary to standardize the representation of time periods. Annual reports typically report time in one of two formats: as a single timestamp or as a time range. This distinction exists to indicate whether a financial value represents a specific point in time (e.g., equity at the beginning of the year) or spans an entire year (e.g., revenue). To facilitate further analysis,

I chose to align these formats by imputing their values to one another.

For example, I paired the revenue reported for the period 01-01-2023 to 01-01-2024 with the equity reported for 01-01-2024. This pairing reflects the logical relationship where the equity at the start of 2024 represents the year-end result of 2023. This approach was applied consistently across all years, ensuring a unified and comprehensive database.

Once the imputation was completed, The dataset's completeness was assessed. Columns with insufficient data, such as custom financial tags created by companies that had very few entries, were removed including other low populated columns. Similarly, some rows were excluded when pairing information was not possible, particularly for entries lacking data from prior years. These rows offered little analytical value and were thus omitted.

After these adjustments, I opted to retain only the time range periods while removing single-timestamp entries. Since both formats now contained identical information due to the imputation process, keeping only one ensured simplicity and consistency in the dataset moving forward. This streamlined approach created a cleaner, more uniform database for subsequent analysis.

To enhance the dataset's analytical depth, several key financial metrics were calculated and used to support financial modeling. A notable addition was the computation of the Altman Z-score, a well-established indicator of financial health and bankruptcy risk. This metric integrates various financial ratios, leveraging relationships between liquidity (working capital), profitability (retained earnings, EBIT), leverage (Equity to total liabilities), and operational efficiency (Revenue to total assets). By calculating the Z-score, the dataset gained a robust tool for assessing the financial stability of entities, enabling insights into their risk profiles and overall economic viability.

$$\begin{aligned}\text{Z-score} = 1.2 \times \frac{\text{Working Capital}}{\text{Total Assets}} + 1.4 \times \frac{\text{Retained Earnings}}{\text{Total Assets}} + \\ 3.3 \times \frac{\text{EBIT}}{\text{Total Assets}} + 0.6 \times \frac{\text{Equity}}{\text{Total Liabilities}} + 1.0 \times \frac{\text{Revenue}}{\text{Total Assets}},\end{aligned} \tag{1}$$

where:

$$\text{Working Capital} = \text{Current Assets} - \text{Current Liabilities}, \tag{2}$$

$$\text{EBIT} = \text{Profit or Loss Before Tax}. \tag{3}$$

In addition to these metrics, essential financial ratios were introduced to further enrich the dataset. These ratios offer valuable perspectives on profitability,

efficiency, and financial stability, forming the foundation for deeper financial analysis.

Metrics such as Return on Equity (ROE) and Return on Assets (ROA) were derived to measure the company's profitability relative to its equity and total assets. Similarly, Profit Margin was calculated to evaluate the efficiency of converting revenue into profit. To assess financial leverage, the Debt-to-Equity ratio was computed, capturing the relationship between a company's liabilities and equity. Additionally, the Current Ratio was included to evaluate liquidity by comparing current assets to current liabilities, providing insight into the company's ability to meet short-term obligations.

$$\text{ROE} = \frac{\text{Profit or Loss}}{\text{Equity}} \times 100, \tag{4}$$

$$\text{ROA} = \frac{\text{Profit or Loss}}{\text{Assets}} \times 100, \tag{5}$$

$$\text{Profit Margin} = \frac{\text{Profit or Loss}}{\text{Revenue}} \times 100, \tag{6}$$

$$\text{Debt-to-Equity} = \frac{\text{Liabilities}}{\text{Equity}}, \tag{7}$$

$$\text{Current Ratio} = \frac{\text{Current Assets}}{\text{Current Liabilities}}. \tag{8}$$

These ratios were designed with safeguards to address scenarios where denominators were missing or zero, ensuring the integrity of the results. By integrating these financial metrics, the dataset gained a robust analytical layer, enhancing its capacity to inform strategic decision-making and financial evaluations.

Despite extensive restructuring, cleaning, and imputation, achieving a fully complete dataset was not possible. This limitation stems from the variability in how companies report their financial data and the specific elements they choose to disclose. This outcome directly addresses the first sub-question: "To what extent are there inconsistencies or inaccuracies in XBRL data across different companies?" This sub-question is aimed at identifying specific areas where discrepancies or errors occur in financial reporting. The significant effort required for structuring and cleaning highlighted these inconsistencies in reporting across companies. Even after implementing imputation and feature engineering to fill gaps and standardize the data where possible, these inconsistencies remained evident, highlighting the challenges posed by the diverse approaches companies take in reporting.

4.3 Data exploration

In this section, we dive deeper into the dataset to explore key financial relationships and trends through data visualizations. Various graphical techniques are employed to uncover patterns, highlight anomalies, and provide a clearer understanding of the financial structures of the companies in the dataset. Visualizations enable the drawing of meaningful conclusions that might be difficult to identify through raw data alone. The focus is primarily on visualizing the relationships between assets, liabilities, and equity, as well as the calculated financial metrics and ratios, offering insights into company size, financial stability, and the potential risks or strengths in their financial positions. Through these visual analyses, a deeper understanding of the data is achieved, facilitating more informed interpretations and decision-making.

Figure 1 below visualizes the relationship between assets, liabilities, and equity across the companies in the dataset, providing insights into their financial structures. On the x-axis, the chart plots total assets, which represent the resources available to each company. The y-axis represents total liabilities, reflecting the financial obligations. The size of each bubble corresponds to the company's equity, which indicates its net worth or the value of ownership after subtracting liabilities from assets.

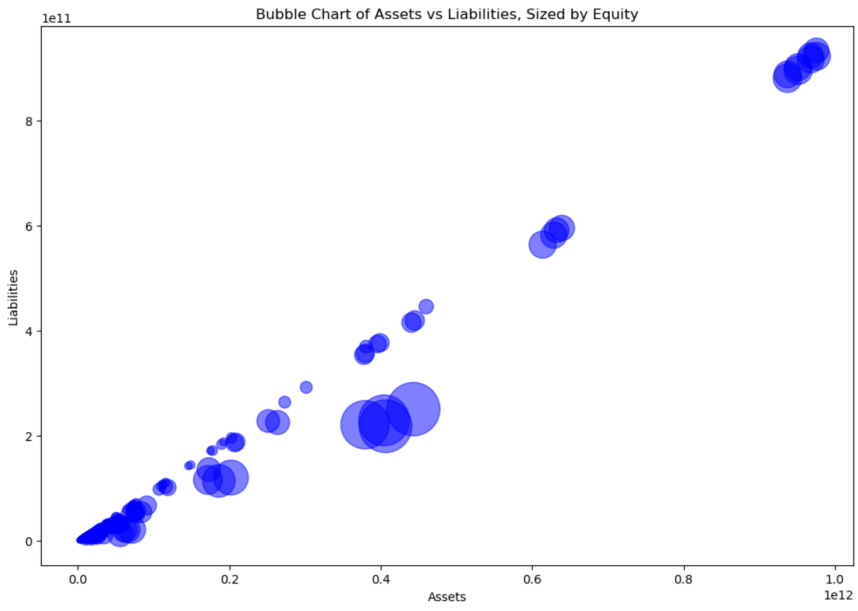

Figure 1: Bubble chart visualizing the relationship between Assets and Liabilities, sized by Equity across the companies in the dataset

By examining Figure 1, patterns and trends in the data emerge. For example, companies with larger assets tend to have higher liabilities, which is expected, but variations from this trend can also be seen. These are outliers, where companies either have disproportionately high liabilities relative to assets or unusually high assets relative to their liabilities. The size of the bubbles offers additional insight into the financial health of each company, with larger bubbles indicating higher equity levels, which suggest a stronger financial position and a more robust net worth. In contrast, smaller bubbles point to companies with narrower equity margins, potentially signaling higher financial risk.

What can be seen in the radar charts below, is that it provides a comparative visualization of the financial performance across multiple companies or entries between companies using key financial ratios and the Altman Z-score. Each axis represents a distinct financial metric: Return on Equity (ROE), Return on Assets (ROA), Profit Margin, Debt-to-Equity Ratio, Current Ratio, and the Altman Z-score. The plotted areas for each entry allow for an intuitive comparison of these metrics, highlighting variations in financial stability, profitability, and risk.

Larger areas in the radar chart generally signify stronger financial performance. For example, a higher ROE or ROA indicates efficient use of equity or assets to generate profits. Similarly, the Altman Z-score reflects the company's financial health and bankruptcy risk, with higher scores implying lower risk. The Profit Margin axis reveals how effectively revenue is translated into profit, while the Debt-to-Equity Ratio axis evaluates leverage, where lower values are often favorable. Finally, the Current Ratio assesses liquidity, indicating the company's ability to meet short-term obligations.

Figure 2 shows notable differences among the entries(years), suggesting variations in financial conditions over time. For instance, some entries have higher profitability or overall financial health while others metrics like leverage (Debt-to-Equity) or liquidity (Current Ratio) stayed relatively the same. This radar chart serves as a summary of financial performance, offering a clear, visual comparison of key metrics across multiple periods.

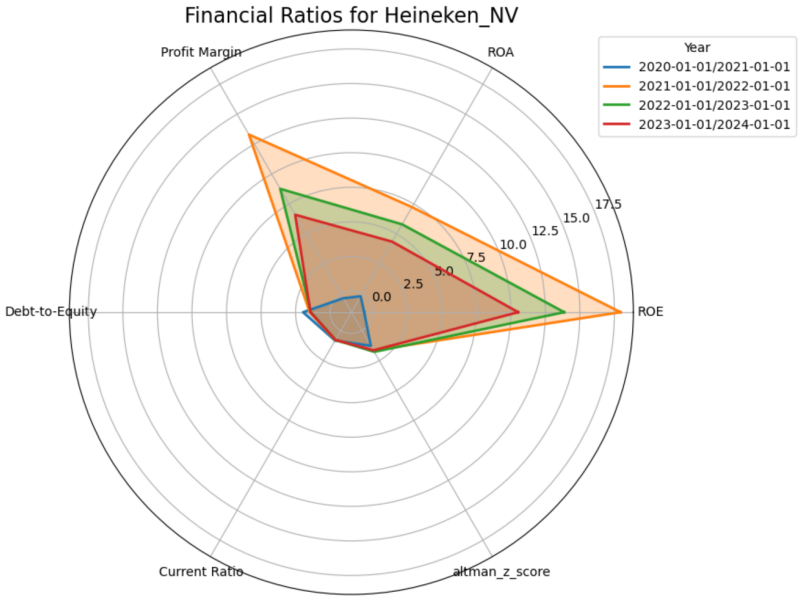

Figure 2: Radar chart of Heineken NV visualizing the financial performance between years using key financial ratios and the Altman Z-score

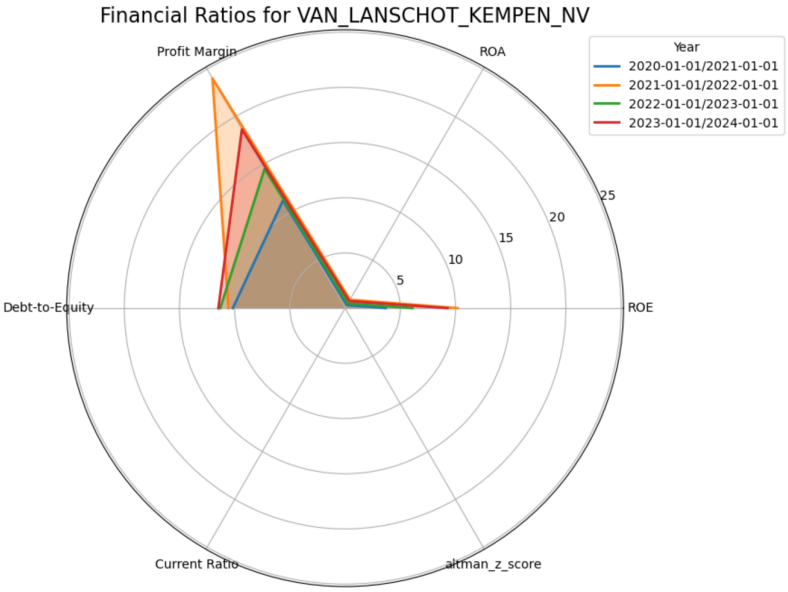

Figure 3: Radar chart of Van Lanschot Kempen NV visualizing the financial performance between years using key financial ratios and the Altman Z-score

Figure 3 visualizes the financial ratios and Altman Z-score for Van Lanschot Kempen NV across four entries. However, a key observation is the missing Altman Z-score and Current Ratio for all entries, which leaves two critical metrics unrepresented in the comparison. Upon further inspection, the inability to calculate the Altman Z-score stems from missing working capital data. The working capital, a vital component of the Z-score formula, could not be derived due to the absence of specific details in the dataset. While total assets and total liabilities were reported, the lack of separation between current and noncurrent components prevented the calculation of working capital. The same holds for the Current Ratio where Current Assets and Current Liabilities were both needed for the calculation.

The chart still offers valuable insights into the other financial ratios. A notable trend is the consistently high Profit Margin, suggesting effective cost control and profitability. However, ROA shows negligible values, indicating possible inefficiencies in asset utilization. The Debt-to-Equity remains moderate, implying balanced leverage.

This data exploration again highlights that there are inconsistencies in reporting across companies, which can lead to incomplete financial analyses. Which directly addresses the first sub-question again: "To what extent are there inconsistencies or inaccuracies in XBRL data across different companies?" The missing Altman Z-score underscores how reporting discrepancies, such as a lack of detailed breakdowns in financial data, can hinder the ability to calculate critical metrics and conduct comprehensive analyses. This emphasizes the need for standardized reporting practices to enhance data usability and comparability across entities.

5 Methodology

This section outlines the approaches taken to cluster and analyze the dataset. The methodology describes three distinct versions of clustering, each utilizing different pre-processing techniques to handle missing values and generate meaningful groupings. Section 5.1 focuses on hierarchical clustering, where we employ a custom pairwise distance function to handle missing data effectively, preserving the integrity of the dataset. In Section 5.2, we explore KMedoids clustering, which leverages the same custom distance function but uses actual data points as cluster centers, making it more robust to outliers. Section 5.3 introduces a machine learning-based approach, combining Random Forest imputation with Agglomerative Clustering to uncover latent relationships in the data, providing a powerful alternative for handling missing values and complex structures. Each method is evaluated based on its strengths, limitations, and the way it handles missing data, ultimately offering a comprehensive approach to clustering.

5.1 Hierarchical Clustering

The first approach applies hierarchical clustering to a dataset containing missing values. Hierarchical clustering builds a hierarchy of clusters by iteratively merging or splitting groups based on a distance metric. This method is particularly valuable due to its flexibility in defining cluster granularity and its ability to produce a visual representation of relationships via dendrograms.

A key challenge in clustering is handling missing values, as they can distort distance calculations and affect the validity of results. Instead of discarding incomplete data or imputing missing values, this approach employs a custom pairwise distance function to compute distances while ignoring dimensions with missing values. This ensures that data points are compared only across shared attributes, preserving as much information as possible while minimizing bias.

5.1.1 Handling Missing Values

The choice to use a custom distance function is motivated by the need to minimize assumptions about the data. By restricting distance calculations to non-missing dimensions, the analysis avoids introducing biases that could arise from imputation or complete-case analysis. However, ignoring missing values altogether is not without drawbacks, as more sophisticated imputation techniques exist in the literature that could potentially yield more robust results. This limitation is discussed further in Section 5.1.4.

The distance function is defined as follows:

$$d(\mathbf{x}, \mathbf{y}) = \begin{cases} \sqrt{\frac{\sum_{i \in S}(x_i - y_i)^2}{|S|}}, & \text{if } S \neq \emptyset \\ \infty, & \text{if } S = \emptyset \end{cases} \tag{9}$$

where:

- $S = \{i \mid \neg\text{isna}(x_i) \wedge \neg\text{isna}(y_i)\}$,
- x_i, y_i are the values of $\mathbf{x}$ and $\mathbf{y}$ in dimension i,
- $\text{isna}(x_i)$ checks if x_i is missing,
- $|S|$ is the number of shared dimensions, ensuring normalization by common fields.

This formulation ensures that only shared dimensions are considered when calculating the Euclidean distance metric. This is used to measure dissimilarity between data points. The metric is well-suited for numerical data and ensures consistency with the chosen clustering method. If no shared dimensions exist ($S = \emptyset$), the distance is set to infinity, effectively disqualifying the pair from being grouped together in a cluster. This approach minimizes assumptions about the data while maximizing the amount of information retained. However, a potential downside is that data points with fewer shared attributes may appear artificially closer simply due to fewer dimensions being considered. By normalizing the distance by the number of shared fields, this issue is mitigated, ensuring a fairer comparison between data points.

5.1.2 Preprocessing and Scaling

Before computing the distance matrix, the data is standardized to ensure comparability across features. Standardization rescales the data so that each feature has a mean of 0 and a standard deviation of 1:

$$z = \frac{x - \mu}{\sigma} \tag{10}$$

where:

- z is the standardized value,
- x is the original feature value,
- μ and σ are the mean and standard deviation of the feature, respectively.

This step is crucial for distance-based methods, ensuring that all features contribute equally to the distance calculation, regardless of their original scale.

5.1.3 Clustering Approach

The hierarchical clustering algorithm is applied to the distance matrix derived from the custom distance function. The clustering process consists of the following steps:

1. **Distance Matrix Calculation:** The pairwise distances between all data points are computed using Equation (9). This results in a condensed distance matrix, a one-dimensional array containing the upper triangular portion of the full pairwise distance matrix.

2. **Linkage Calculation:** A linkage method determines how clusters are merged based on the distances. In this approach, Ward's linkage is used, which minimizes the variance of the clusters being merged. This ensures that the resulting clusters are compact and relatively homogeneous, which is particularly beneficial when working with numerical datasets. Mathematically, for two clusters A and B, Ward's linkage updates the total within-cluster variance as:

$$\Delta E = \frac{|A||B|}{|A| + |B|} \|\bar{x}_A - \bar{x}_B\|^2 \tag{11}$$

   where $\bar{x}_A$ and $\bar{x}_B$ are the centroids of clusters A and B, and $\|\cdot\|^2$ represents squared Euclidean distance.

3. **Dendrogram Construction:** The hierarchical structure is visualized using a dendrogram. The dendrogram provides a visual summary of the clustering process, including the order in which data points are merged and the distances at which clusters are formed. Each node represents a cluster, and the height of a node corresponds to the distance (or dissimilarity) at which clusters were merged.

5.1.4 Advantages and Limitations

This approach is well suited for data sets with missing values, offering a balance between simplicity and robustness. The method's strengths lie in its ability to handle missing values naturally without discarding information. It's interpretability using a dendrogram, providing a clear visual representation of the clustering process, making it easier to understand the relationships between data points and clusters. Furthermore, it needs no pre-specification of Clusters.

However, hierarchical clustering also has a few limitations that need to be considered. One of the main challenges is its computational complexity. For large datasets, hierarchical clustering can become computationally expensive, primarily due to the $\mathcal{O}(n^2)$ complexity associated with calculating pairwise distances between data points. This can lead to significant processing time, especially as the dataset grows in size. Another limitation is its sensitivity to the linkage choice. The outcome of hierarchical clustering is heavily influenced by the linkage method used, whether it's single, complete, or average linkage. Selecting the appropriate linkage method is crucial and requires careful consideration of the specific characteristics of the dataset, as different methods can lead to varying results.

5.2 KMedoids Clustering

The second approach applies the KMedoids clustering algorithm, which is well-suited for datasets with missing values when combined with a custom distance matrix. KMedoids differs from k-means by selecting actual data points (medoids) as cluster centers instead of centroids, making it more robust to outliers. A medoid is a data point that minimizes the sum of distances to all other points within its cluster.

5.2.1 Handling Missing Values and Scaling

Given its dependence on pairwise distances, this method naturally integrates the custom distance function introduced in Section 5.1, ensuring that missing values are handled consistently across clustering approaches. This allows the clustering to be based solely on shared attributes, reducing bias from imputation. The data is standardized before distance calculation to ensure comparability across features, as in Equation (10).

5.2.2 KMedoids Clustering Approach

Unlike hierarchical clustering, which iteratively merges clusters, KMedoids partitions the data into k clusters by minimizing intra-cluster dissimilarity. The clustering process consists of the following steps:

1. **Distance Matrix Calculation:** The pairwise distances between all data points are computed using the custom function from Section **??**, ensuring compatibility with the handling of missing values.

2. **Initialization:** The algorithm begins by randomly selecting k data points as initial medoids. These medoids serve as the starting representatives for the clusters.

3. **Assignment:** Each data point is assigned to the cluster of its nearest medoid based on the precomputed distance matrix.

4. **Medoid Update:** Within each cluster, the algorithm searches for a new medoid that minimizes the total dissimilarity to all other points in the cluster. The point that achieves this becomes the new medoid.

5. **Iteration:** Steps 3 and 4 are repeated until medoids stabilize or a predefined number of iterations is reached.

The objective function minimizes the total within-cluster dissimilarity:

$$\sum_{j=1}^{k} \sum_{i \in C_j} d(\mathbf{x}_i, \mathbf{m}_j) \tag{12}$$

where:

- C_j is the set of points in cluster j,
- $\mathbf{m}_j$ is the medoid of cluster j,
- $d(\mathbf{x}_i, \mathbf{m}_j)$ is the pairwise distance.

Unlike hierarchical clustering, where linkage choice affects results, KMedoids relies directly on the custom distance function for cluster formation.

5.2.3 Advantages and Limitations

KMedoids clustering offers several advantages that make it a compelling choice for certain data analysis tasks. One of its primary strengths is its robustness to outliers. Unlike centroid-based methods such as k-means, which can be highly sensitive to outliers, KMedoids uses actual data points as medoids. This approach significantly reduces the impact of outliers, leading to more reliable clustering results in the presence of noisy data. Additionally, KMedoids utilizes the same distance function as hierarchical clustering, avoiding potential biases that may arise from imputing missing values. Another key benefit is its interpretability. Since medoids are representative data points, they provide a clear and intuitive way to interpret the clusters, aiding in understanding the structure of the data.

However, KMedoids does come with its limitations. One notable challenge is its computational complexity. Similar to hierarchical clustering, constructing the distance matrix in KMedoids can be computationally expensive, especially when dealing with large datasets, as the time complexity scales with $\mathcal{O}(n^2)$. Furthermore, KMedoids requires the predefinition of k, the number of clusters, which may necessitate additional exploratory analysis or domain expertise to determine the optimal number of clusters. Lastly, KMedoids suffers from random initialization, as the final clustering result can depend on the initial selection of medoids. To ensure stable and reliable results, multiple runs are often necessary. Despite these trade-offs, KMedoids provides a strong alternative to hierarchical clustering, particularly when a predefined number of clusters is desired and robustness to outliers is important.

5.3 Random Forest Clustering with Imputed Dataset

The third approach involves clustering a dataset where missing values have been imputed using a robust iterative imputation method. The clustering process is guided by a proximity matrix derived from a Random Forest Classifier, capturing relationships between companies. This method combines the power of machine learning-based imputation with the interpretability of proximity-based clustering.

5.3.1 Handling Missing Values

In this version, missing values are addressed through an iterative imputation model. The dataset is imputed using an IterativeImputer with a RandomForestRegressor as the estimator. The imputation process works iteratively, estimating missing values based on features with a correlation above a certain threshold (e.g. $|r| > 0.7$). For a given target column x, the algorithm identifies features that are highly correlated with x, and uses them to predict missing values using a Random Forest model. The imputer iteratively updates missing values until convergence. By using highly correlated features to estimate missing values, it retains more of the dataset's structure compared to simpler methods like mean or median imputation.

5.3.2 Clustering Approach

The clustering process begins with the construction of a proximity matrix, a matrix that represents the similarities between different companies in the dataset, derived from a RandomForestClassifier. This step serves as the foundation for the subsequent clustering analysis, leveraging the Random Forest's ability to capture intricate relationships in the data. The overall process can be broken down into the following steps:

Step 1: Training the Random Forest After imputation to handle missing values, a RandomForestClassifier is trained on the complete numerical dataset. Random Forest is an ensemble learning method that operates by creating a collection of decision trees, each trained on a random subset of the dataset and features. This randomness introduces diversity among the trees, which enhances the model's ability to capture complex, non-linear relationships. Importantly, the Random Forest algorithm assigns each company to a terminal node, or leaf, in every tree.

Step 2: Constructing the Proximity Matrix To quantify the similarity between companies, the Random Forest's structure is leveraged to construct a proximity matrix. This process involves two key steps:

1. **Leaf Index Extraction**: As each company is passed through every tree in the forest, the index of the leaf node where the company lands is recorded. If two companies land in the same leaf of a tree, they are considered similar

with respect to that tree. This step effectively uses the Random Forest as a mechanism for grouping companies based on their feature similarities.

2. **Proximity Calculation**: For each pair of companies i and j, the number of trees where both companies share a leaf is counted. This count is then normalized by the total number of trees T in the forest, resulting in a proximity value:

$$\text{Proximity}(i, j) = \frac{\text{Number of shared leaves}(i, j)}{T}. \tag{13}$$

   A higher proximity value indicates greater similarity between the two companies.

The proximity matrix is a $n \times n$ matrix, where n is the total number of companies, and it provides a robust, non-parametric representation of the relationships in the data.

Step 3: Converting to a Distance Matrix To enable clustering, the proximity matrix is transformed into a distance matrix, where the distance between two companies is defined as:

$$\text{Distance}(i, j) = 1 - \text{Proximity}(i, j). \tag{14}$$

This conversion ensures that higher proximities correspond to shorter distances, aligning with the requirements of clustering algorithms.

Step 4: Agglomerative Clustering With the distance matrix in place, Agglomerative Clustering is applied to group the companies into clusters. Agglomerative clustering is a hierarchical clustering method that starts with each company as an individual cluster and iteratively merges the two closest clusters until a predefined number of clusters is reached. Here the same linkage criterion is used as in the Hierarchical clustering approach, namely the Ward linkage, which minimizes the variance within clusters during each merge. This results in compact, cohesive clusters.

The resulting clusters represent groups of companies with internal similarity as determined by the Random Forest's decision trees. This approach combines machine learning with hierarchical clustering, creating a pipeline where the Random Forest acts as the first step to uncover latent relationships, and agglomerative clustering builds upon these relationships to form interpretable groupings.

5.3.3 Advantages and Limitations

This methodology offers several notable strengths that contribute to its effectiveness in various data analysis tasks. One of the key advantages is its ability to handle non-linear relationships. The Random Forest algorithm excels in capturing complex, non-linear interactions between features, which are then reflected in the proximity matrix. This makes it particularly useful for datasets where relationships between variables cannot be captured by simpler linear models. Additionally, the model demonstrates robustness to outliers. Due to its ensemble nature, where multiple decision trees are used, the impact of outliers is minimized. Each individual tree can handle irregularities independently, preventing any one outlier from disproportionately affecting the overall model performance. Another strength is the seamless integration with clustering. The proximity matrix, which is generated by Random Forest, serves as a natural input for agglomerative clustering. This allows for a smooth transition from machine learning to clustering, enabling a cohesive and efficient analysis pipeline.

However, there are some limitations to consider. One major drawback is the dependence on imputation quality. The Random Forest model requires a complete dataset, so any missing values must be imputed beforehand. If the imputation step is done poorly, errors can propagate through the entire process, ultimately affecting the quality of the clustering results. Additionally, the performance of both the Random Forest and the clustering process is highly sensitive to the choice of hyperparameters. With numerous parameters that need fine-tuning, optimizing this model can be challenging, and achieving the best performance requires careful attention to detail.

5.4 Summary of methods

- **Hierarchical Clustering:** Uses a custom distance function directly on a datasets with missing values. The custom pairwise distance function computes distances only on shared dimensions between data points, avoiding bias from imputation or data exclusion. The Ward linkage method was used to create interpretable clusters visualized through a dendrogram.

- **Kmedoids:** This approach applies the same custom distance function as in Hierarchical Clustering. Medoids are iteratively updated to minimize intra-cluster distances until the medoids stabilize or the algorithm reaches the maximum number of iterations. This method is robust to outliers.

- **Random Forest Clustering:** This method handles missing values with an imputation method. A Random Forest model generates a proximity matrix by measuring how often companies share leaf nodes across trees. This matrix is converted to a distance matrix and clustered using Agglomerative Clustering with Ward linkage, capturing non-linear relationships in the data.

6 Results

This section presents the results obtained from the three different clustering approaches: Hierarchical clustering with a custom distance function (Section 6.1), KMedoids clustering with a custom distance function (Section 6.2), and Random Forest clustering (Section 6.3).

A key challenge in validating the clustering results was the absence of a predefined ground truth or validation set. Unlike supervised learning, where model performance can be evaluated against labeled data, clustering is mostly an unsupervised technique that does not inherently provide a measure of correctness. This makes it difficult to determine whether the identified clusters truly represent meaningful groupings in the underlying data. To address this challenge, cluster quality was assessed using a combination of visualizations and validation metrics, including cluster cohesion and separation.

Each clustering approach uses the same companies and features as input. The features used are the following:

Financial Metrics From Annual Report	
Assets	NoncurrentAssets
BasicEarningsLossPerShare	NoncurrentLiabilities
CashAndCashEquivalents	ProfitLoss
CashFlowsFromFinancingActivities	ProfitLossBeforeTax
CashFlowsFromInvestingActivities	PropertyPlantAndEquipment
CashFlowsFromOperatingActivities	RetainedEarnings
ComprehensiveIncome	Revenue
CurrentAssets	SharePremium
CurrentLiabilities	Inventories
CurrentTaxLiabilities	IssuedCapital
DeferredTaxAssets	Liabilities
DeferredTaxLiabilities	Equity
DilutedEarningsLossPerShare	EquityAndLiabilities
IncomeTaxExpense	
Manually Created Financial Metrics	
Altman Z Score	Profit Margin
Current Ratio	Return on Assets (ROA)
Debt-to-Equity	Return on Equity (ROE)
Working Capital	

Table 1: List of Financial Metrics used for the models

Number of Clusters choice: In this study, the number of clusters was set at 15 for all clustering methods to facilitate a fair comparison between them. This number was chosen because it strikes a balance between achieving sufficient separation between clusters and retaining the ability to derive meaningful

insights within each group. Although increasing the number of clusters could improve the Silhouette score and capture more subtle variations in the data, it could also fragment natural financial groupings, making it more difficult to extract actionable insights. Financial patterns across industries typically follow broader trends, and excessive clustering could artificially divide companies, obscuring genuine structural differences, making the results harder to interpret.

By choosing 15 clusters, the analysis ensures that companies with similar financial characteristics are grouped together, preserving both the robustness of the clustering process and the interpretability of the results.

6.1 Hierarchical Clustering

In this section we discuss the results of Hierarchical clustering. Hierarchical clustering was applied using a pairwise distance function that accommodates missing values. The resulting dendrogram (Figure 4) visualizes how the clusters merged at different distance thresholds.

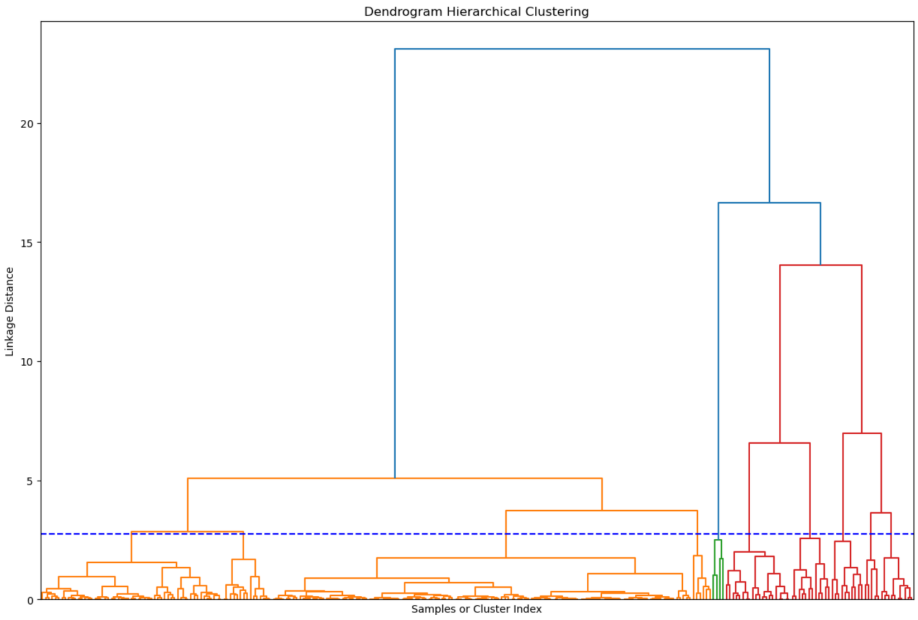

Figure 4: Dendrogram of hierarchical clustering. The colored branches represent distinct clusters identified in the hierarchical clustering process. Each color corresponds to a different group

A dendrogram shows the height at which branches merge, indicating the similarity between clusters:

- **X-axis**: Represents individual companies or clusters.
- **Y-axis**: Represents the distance between clusters. The height at which two clusters merge is proportional to their similarity, where a higher distance means that they are less similar.

From Figure 4, it can be observed that companies on the left side (orange) merge at a shorter distance, suggesting higher similarity, whereas those on the right side (green and red) merge at larger distances, indicating greater dissimilarity. Tracking company-specific splits in the dendrogram provides insights into how financial characteristics drive cluster formation.

6.1.1 Cluster Assignments

To analyze the clustering structure, the dataset was partitioned first into 10 clusters based on a distance threshold of $t = 3$. Table 2 summarizes the cluster assignments.

In the introduction of this section it was mentioned that the number of clusters was set at 15 for all clustering methods to facilitate a fair comparison. However, when applying hierarchical clustering, a direct approach with 15 clusters resulted in a very dominant cluster. Therefore another approach was taken, which will be explained in more detail later in this section. This method led to a more balanced distribution while preserving meaningful financial groupings.

Cluster	1	2	3	4	5	6	7	8	9	10
# Companies	56	15	350	6	4	20	12	10	5	10

Table 2: The number of companies per cluster using Hierarchical clustering

As mentioned, one of the key challenges was the validation of the clustering results without the absence of a predefined ground truth or validation set. Therefore, a three-dimensional visualization (Figure 5) was generated based on Assets, Equity, and Liabilities, offering a simplified view of how companies are distributed across clusters.

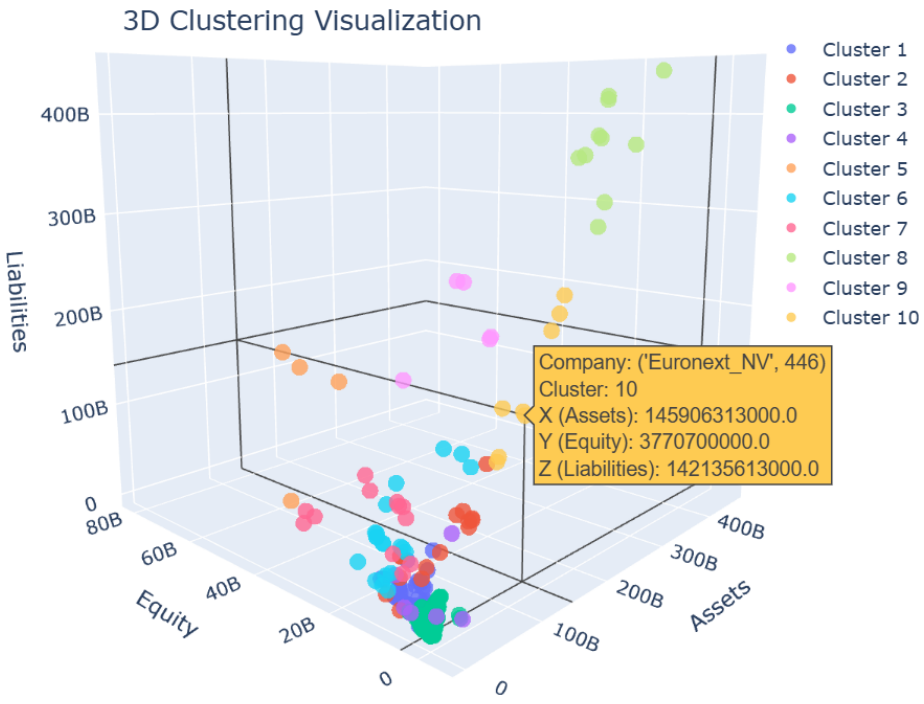

Figure 5: 3D scatter plot of hierarchical clustering results based on Assets, Equity and Liabilities

Although this representation is useful for visualizing cluster separability in a limited feature space, it does not capture the full 34-dimensional structure of the data. In addition, the hierarchical clustering model revealed a significant challenge: the presence of a large dominant cluster containing approximately 75% of the companies. As shown in Table 2, this dominant cluster (3) contained 350 companies, while the remaining companies were distributed among much smaller clusters.

This imbalance suggests two things:

- The majority of the companies are highly similar
- The clustering algorithm struggles to distinguish meaningful subgroups due to high-dimensionality effects or outliers.

The first case is highly unlikely because this dataset consists of listed companies which are very different from each other, so to further investigate the second case, there are two possible approaches. The first approach to address this challenge is dimensionality reduction, such as Principal Component Analysis (PCA). PCA could help by transforming the original 34 features into a smaller set of orthogonal components that capture the most significant variance in the

data. However, PCA requires complete data, and the presence of missing values prevents its direct application.

Another solution is a sub-clustering approach. This was applied exclusively to the largest cluster (Cluster 3). The goal was to identify potential hidden structures within this large grouping that the initial clustering step failed to capture. Reapplying the same clustering techniques to this subset of companies. Resulted in the following dendrogram shown in Figure 6.

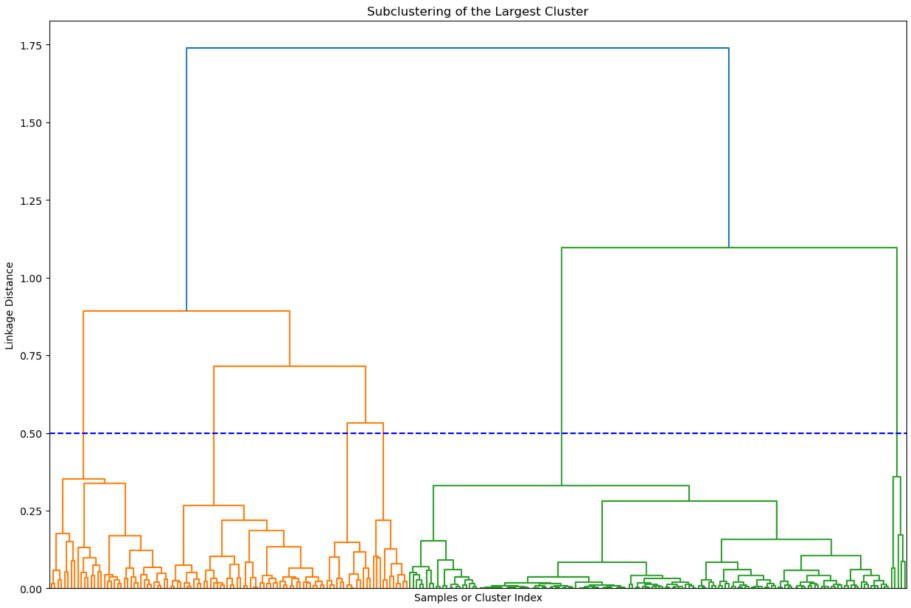

Figure 6: Dendrogram of hierarchical sub-clustering. The colored branches represent distinct sub-clusters identified in the hierarchical sub-clustering process. Each color corresponds to a different group

The assignment of clusters was divided into 6 clusters based on a distance threshold of $t = 0.5$. Which could be seen in Table 3

Cluster	1	2	3	4	5	6
# Companies	36	51	14	11	233	5

Table 3: The number of companies per sub-cluster using Hierarchical clustering

The results of the sub-clustering indicate that the initial large cluster (cluster 3) was not truly homogeneous; rather, the high-dimensional nature of the data likely caused the clustering algorithm to group dissimilar companies together. Further validation was performed using 3D visualizations based on key financial metrics. These visualizations (Figures 7 and 8) illustrate how the sub-clusters separate more clearly now.

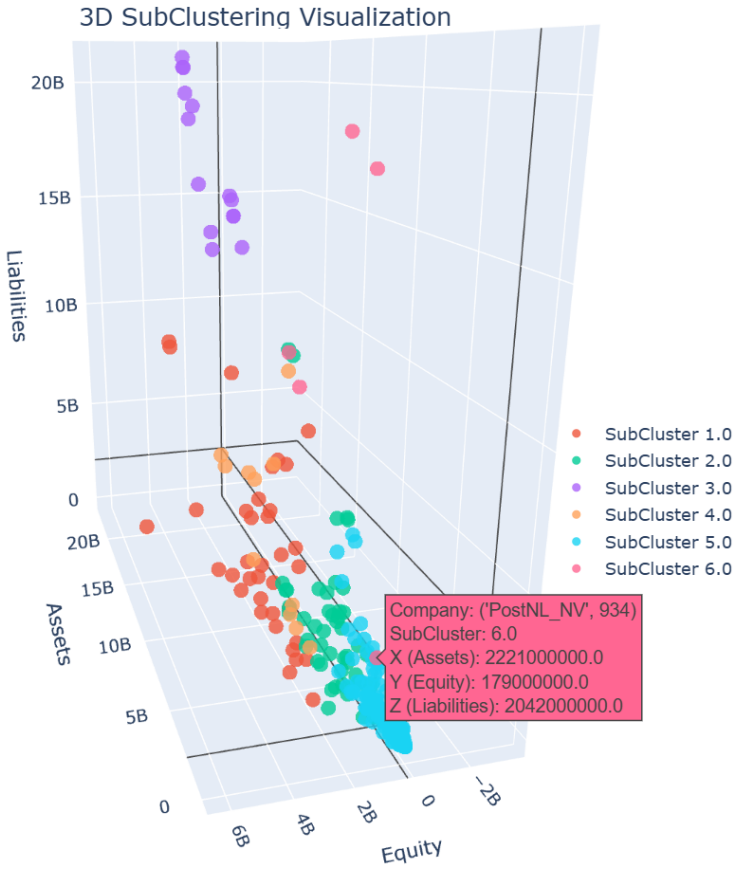

Figure 7: 3D scatter plot of hierarchical sub-clustering results based on Assets, Equity and Liabilities

Figure 7 shows the sub-cluster distribution based on Assets, Equity, and Liabilities. Initially, the clustering may seem unexpected, with a company such as PostNL appearing in cluster 6 despite being very different in these attributes from the other companies in cluster 6.

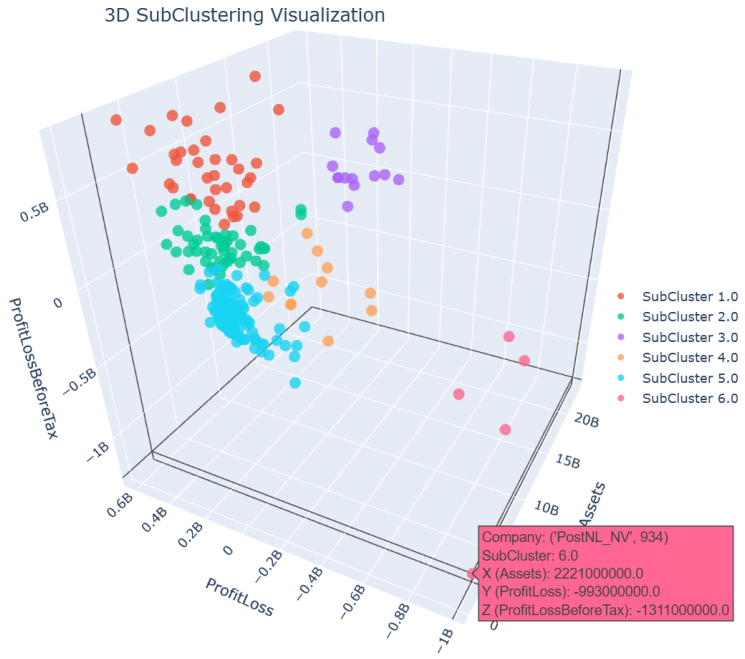

Figure 8: 3D scatter plot of sub-cluster results results based on Assets, ProfitLoss, ProfitLossBeforeTax

However, when visualizing the same sub-clusters with Assets, ProfitLoss, and ProfitLossBeforeTax (Figure 8), it becomes evident why certain companies such as PostNL could be assigned to cluster 6. Companies that seemed different based on one set of financial metrics could exhibit strong similarities in terms of other measures.

This demonstrates that the hierarchical clustering model effectively captured meaningful structures in the dataset, even when they were not immediately apparent in the first 3D visualization.

6.1.2 Silhouette score and Cluster Cohesion

In order to quantitatively assess the quality of the clustering results, the Silhouette score was calculated. This score provides a measure of how well-separated the clusters are and how cohesive the individual clusters are. It combines two main components:

- Intra-cluster cohesion: This measures how close the points within a single cluster are to each other. A higher value indicates that the points within the cluster are more similar to each other.

- Inter-cluster separation: This measures how distinct the clusters are from one another. A higher value indicates that the clusters are well-separated from one another.

The Silhouette score is calculated for each data point, with values ranging from -1 to 1:
- A score close to 1 indicates that the data points are well-clustered, meaning they are both close to each other within their own cluster and far from other clusters.
- A score close to 0 suggests that the data points are on the border between two clusters, indicating overlap or ambiguity in the cluster assignment.
- A score close to -1 implies that the data points may have been assigned to the wrong clusters, as they are closer to other clusters than their own.

For the initial hierarchical clustering, the calculated Silhouette score was S = 0.662. This score indicates that the clusters are reasonably well-separated, though some overlap still exists. This is not unexpected in high-dimensional financial datasets, where companies can share multiple characteristics, making cluster boundaries less distinct. The score suggests that the initial clustering captured meaningful structures within the data while leaving room for refinement.

To refine the clustering results, a sub-clustering approach was applied to further segment the largest cluster. The Silhouette score for sub-clustering was S = 0.587, indicating that while the refined clusters maintain structure, the increased granularity led to closer proximity between certain sub-groups. This result suggests that sub-clustering was effective in capturing additional variations within the dominant cluster, allowing for a more detailed segmentation of companies with similar financial profiles.

When combining the results of the original clustering and sub-clustering, the overall Silhouette score for the entire dataset was S = 0.474. While lower than the individual clustering scores, this is expected as finer sub-clusters naturally reduce the global silhouette measure. However, this does not directly imply a decline in clustering quality; rather, it reflects the introduction of more precise groupings that better capture the underlying financial structures. The results demonstrate that the sub-clustering step helped refine the initial segmentation, ensuring that companies with similar financial characteristics are grouped in a more meaningful way. Furthermore, this level of separability would not have been achievable if the initial clustering step had been restricted to a fixed number of 15 clusters. The sub-clustering approach allowed for more nuanced groupings within the broader structure, revealing additional patterns that a predefined cluster count might have overlooked.

In conclusion, the Silhouette Scores for both the initial hierarchical clustering (S = 0.662) and the sub-clustering (S = 0.587), as well as the combined score (S = 0.474), indicate that the clustering process successfully identified underlying patterns within the dataset. While the overall silhouette measure decreased, this reflects an increase in cluster granularity rather than a loss of structure. The sub-clustering approach ultimately enhanced the interpretability of the clusters, allowing for a more detailed analysis of financial groupings. The final cluster assignments can be seen in Table 4.

Cluster	1	2	3_1	3_2	3_3	3_4	3_5	3_6	4	5	6	7	8	9	10
# Companies	56	15	36	51	14	11	233	5	6	4	20	12	10	5	10

Table 4: The number of companies per cluster after sub-clustering using Hierarchical clustering

6.1.3 Summary of Findings

Hierarchical clustering successfully grouped companies based on financial metrics, but challenges such as high dimensionality and a large dominant cluster complicated the analysis. The use of sub-clustering provided additional insights, revealing meaningful subgroup structures. The key findings are:

- The dendrograms reveal distinct merging patterns, highlighting companies with similar financial structures.
- The 3D visualizations provide an intuitive understanding of cluster separation, but do not capture all 34 features, meaning that clusters appearing mixed in this view might actually be well separated in a higher-dimensional space.
- More balanced cluster distribution: Instead of a single dominant cluster, the companies were now divided into five sub-clusters of varying sizes, with the largest sub-cluster containing 233 companies a significant reduction from the previous 350 companies.
- The final Silhouette score (0.474) suggests average clustering effectiveness, with still room for improvement.

Overall, this approach offers an interpretable method for clustering companies but may struggle with even higher-dimensional feature interactions, potentially benefiting from alternative clustering techniques.

6.2 KMedoids Clustering

In this section we discuss the results of Kmedoids clustering. Kmedoids clustering was applied as a second method to group companies based on financial characteristics. The clustering process resulted in 15 clusters, as summarized in Table 5.

Cluster	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
# Companies	18	35	44	33	8	45	33	31	33	82	26	30	26	30	28

Table 5: The number of companies per cluster using Kmedoids clustering

The Kmedoids clustering resulted in a more balanced distribution of companies across clusters. The largest cluster (9) contained 82 companies, which is significantly smaller compared to hierarchical clustering's largest group. Due to the more balanced distribution, no sub-clustering was applied for this approach.

6.2.1 3D Visualization of Cluster Distribution

To develop an intuitive understanding of the clustering structure, 3D scatter plots were generated using the same feature sets as in the hierarchical clustering analysis. These visualizations provide insight into how well the clusters are separated and whether certain financial dimensions contribute more effectively to distinguishing between groups.

The first 3D scatter plot (Figure 9) illustrates the clustering results of Kmedoids based on assets, equity, and liabilities. Here, clusters 11 (orange) and 7 (lime green) dominate the graph with their large values and show a small degree of overlap. This suggests that these financial dimensions alone may not fully differentiate company groups.

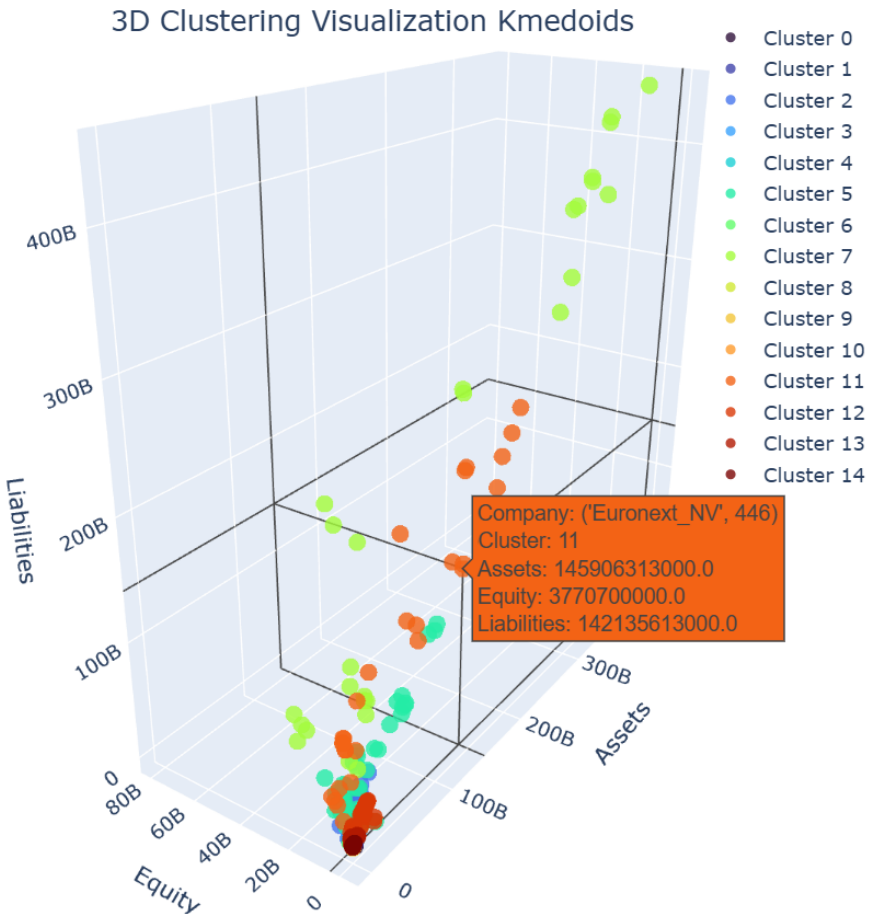

Figure 9: 3D scatter plot of Kmedoids clustering results based on Assets, Equity, and Liabilities

The second scatter plot (Figure 10) presents the same clustering results but based on Assets, ProfitLoss, and ProfitLossBeforeTax. This alternative view highlights more structure that was less apparent in the first visualization. Notably, clusters 11 (orange) and 7 (lime green) are now more distinct from each other, suggesting that profitability metrics contribute more effectively to separating these groups compared to the previous features.

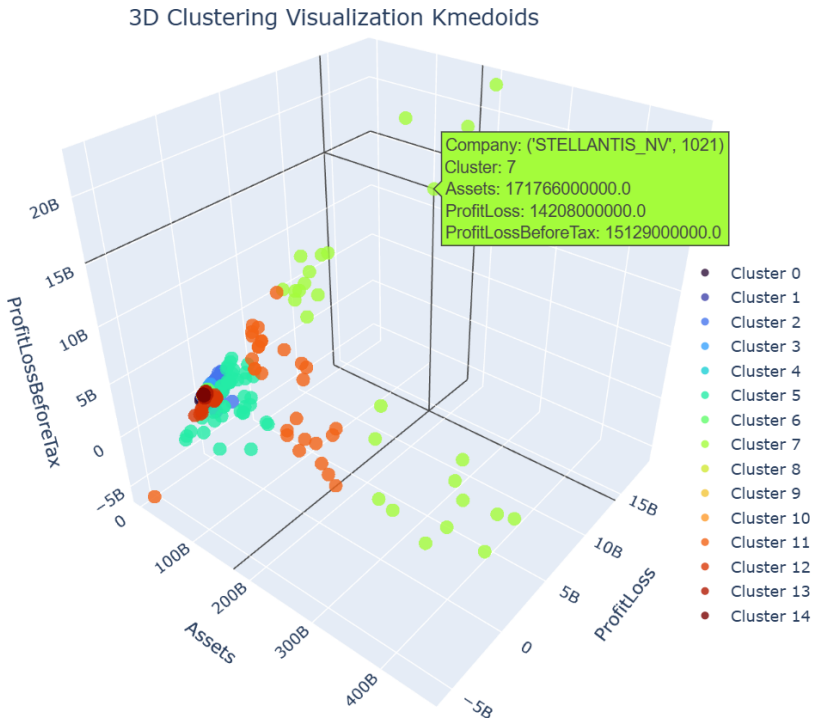

Figure 10: 3D scatter plot of Kmedoids clustering results based on Assets, ProfitLoss, and ProfitLossBeforeTax

A key observation from these figures, particularly in comparison to **Figure 5** (which visualized hierarchical clustering results), is that Kmedoids struggles more with clustering outlier companies. In **Figures 9 and 10**, companies in cluster 7 (lime green) appear highly dispersed across the feature space. This suggests that while these firms were assigned to the same cluster, their financial profiles are not as similar as one might expect, reducing intra-cluster cohesion.

In contrast, hierarchical clustering provided better separation for these outlier companies, placing them in more distinct clusters rather than forcing them into broad, heterogeneous groups. This suggests that hierarchical clustering may have been more effective in identifying meaningful financial structures, at least in terms of handling edge cases.

Additionally, the presence of extreme-value companies makes it difficult to assess the separation of other clusters in the visualizations. The dominant influence of these high-value outliers can obscure finer distinctions at a more granular level. To address this, the next set of plots zooms in on the lower-value clusters while hiding those with extreme values, allowing for a clearer examination of the clustering performance within the main distribution.

6.2.2 Zoomed-In Views for Better Interpretability

To better assess cluster separability, the following figures present the same Kmedoids clustering results with high-value clusters removed. This adjustment makes it easier to observe how well-separated the remaining clusters are in each feature space.

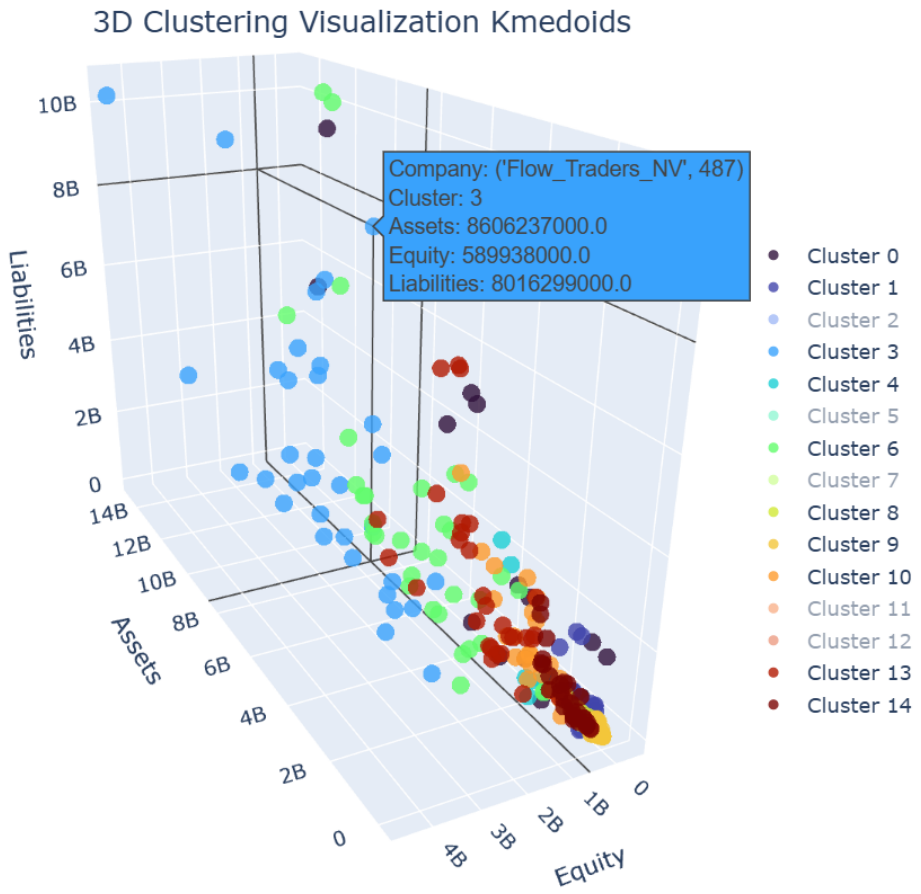

Figure 11: Zoomed-in 3D scatter plot of Kmedoids clustering results based on Assets, Equity, and Liabilities, with extreme-value clusters removed

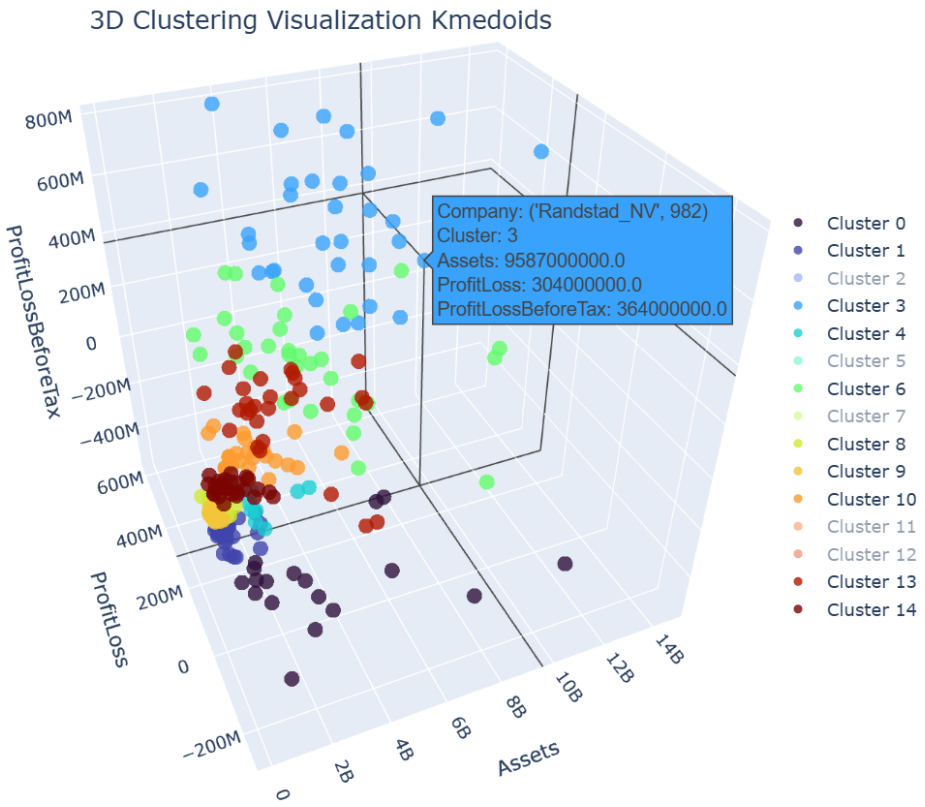

Figure 12: Zoomed-in 3D scatter plot of Kmedoids clustering results based on Assets, ProfitLoss, and ProfitLossBeforeTax, with extreme-value clusters removed

After zooming in, a few key insights emerge: In the Assets, ProfitLoss and ProfitLossBeforeTax space, cluster separation is much clearer, with minimal overlap between clusters. While in the Assets, Equity and Liabilities space, separation is less pronounced, suggesting that the model still struggles with good separation for clustering. Showing that some degree of overlap remains between clusters, particularly for companies with similar capital structures.

6.2.3 Silhouette score and Cluster Cohesion

To quantitatively assess the clustering quality, the Silhouette score was computed. The score measures both intra-cluster cohesion (how similar companies within the same cluster are) and inter-cluster separation (how distinct each cluster is from the others).
For Kmedoids clustering, the computed Silhouette score was $S = 0.135$. This is significantly lower than the hierarchical clustering score (0.465), indicating

that the Kmedoids clusters are less well separated. The lower score suggests the presence of substantial overlap between clusters, making it harder to draw clear distinctions between groups.
Possible reasons for this lower score include:

- The high dimensionality of the dataset, making it difficult to form well-separated clusters.
- The nature of Kmedoids, which relies on medoid selection rather than distance-based hierarchical merging, leading to potential misclassifications in high-dimensional space.
- The random nature of Kmedoids, which relies on random/manual initial medoid selection rather than placing each company in their own cluster.

Despite the relatively low Silhouette Score, Kmedoids clustering still provides useful insights by forming more balanced cluster distributions and reducing the dominance of a single large group.

6.2.4 Summary of Findings

The Kmedoids clustering approach provided an alternative partitioning method, producing a more evenly distributed set of groups compared to hierarchical clustering. However, several key observations highlight both its strengths and limitations:

- Kmedoids avoided the dominance of a single large cluster, leading to a more balanced distribution of companies across groups and no need for sub-clustering.
- The 3D visualizations revealed that while some clusters are well-separated, others exhibit overlap, indicating that the interactions between financial features are complex and not fully captured in low-dimensional space.
- The relatively low Silhouette score (0.135) suggests that while Kmedoids identified meaningful groupings, cluster separation remains weak, likely due to the high dimensionality of the dataset and the challenges of partitioning-based clustering in financial data.
- The approach struggled particularly with outlier companies and cluster cohesion. Companies in certain clusters (e.g., Cluster 7) were highly dispersed across the feature space, reducing intra-cluster similarity. Where hierarchical clustering appeared to provide a better separation of extreme-value companies, suggesting that it may be more effective for datasets with highly variable financial structures.

Overall, while Kmedoids clustering provides a useful comparative approach, but its effectiveness is limited by the dataset's complexity.

6.3 Random Forest Clustering

This last section discusses the result of the Random Forest model. Unlike traditional clustering algorithms that rely on predefined distance metrics, this approach leverages the Random Forest's ability to measure data similarity based on how frequently observations appear in the same leaf nodes across multiple decision trees. The resulting similarity matrix was then used for clustering, yielding 15 distinct clusters.

The distribution of companies across clusters is summarized in Table 6. Unlike hierarchical and Kmedoids clustering, this method did not require handling missing values separately, as the dataset was fully imputed before model training.

Cluster	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
# Companies	12	35	34	46	28	30	40	65	62	17	55	30	31	9	8

Table 6: The number of companies per cluster using Random Forest clustering

Here the clusters also appear more evenly distributed compared to hierarchical clustering but still exhibit some variation in size. The largest cluster (7) contains 65 companies, while the smallest (14) contains only 8. Because there was no real dominant cluster for this model, no sub-clustering was performed.

6.3.1 3D Visualization of Cluster Distribution

To gain deeper insight into the clustering structure, 3D scatter plots were generated using key financial features. These visualizations help assess the degree of cluster separation and evaluate how clustering assignments change under varying financial perspectives.

The first 3D scatter plot (Figure 13) illustrates the Random Forest clustering results based on Assets, Equity, and Liabilities. Compared to the Kmedoids clustering in the previous section, there are notable differences in how companies are assigned to clusters. Specifically, several companies that were previously part of Cluster 8 (lime green) are now grouped into Cluster 11 (orange). This reassignment primarily affects companies positioned higher along the asset and liabilities dimensions. While the overall cluster structure has changed, a slight degree of overlap between Clusters 8 and 11 remains.

Figure 13: 3D scatter plot of Random Forest clustering results based on Assets, Equity, and Liabilities

The second scatter plot (Figure 14) visualizes the clustering results based on Assets, Profit/Loss, and Profit/Loss Before Tax. This perspective helps explain why certain companies were reassigned between clusters. Cluster 8 primarily consists of companies with lower asset values but higher profitability, while Cluster 11 contains companies with high asset values but lower profitability, including some that are operating at a loss. This suggests that the revised clustering structure provides a more intuitive separation compared to the Kmedoids approach for these specific features. However, compared to hierarchical clustering, both Clusters 8 and 11 appear to be elongated rather than distinctly separated into different groups. This does not necessarily indicate a flaw in the clustering process, as the assignments are influenced by multiple financial features beyond those visualized here, but it is good to keep this in mind.

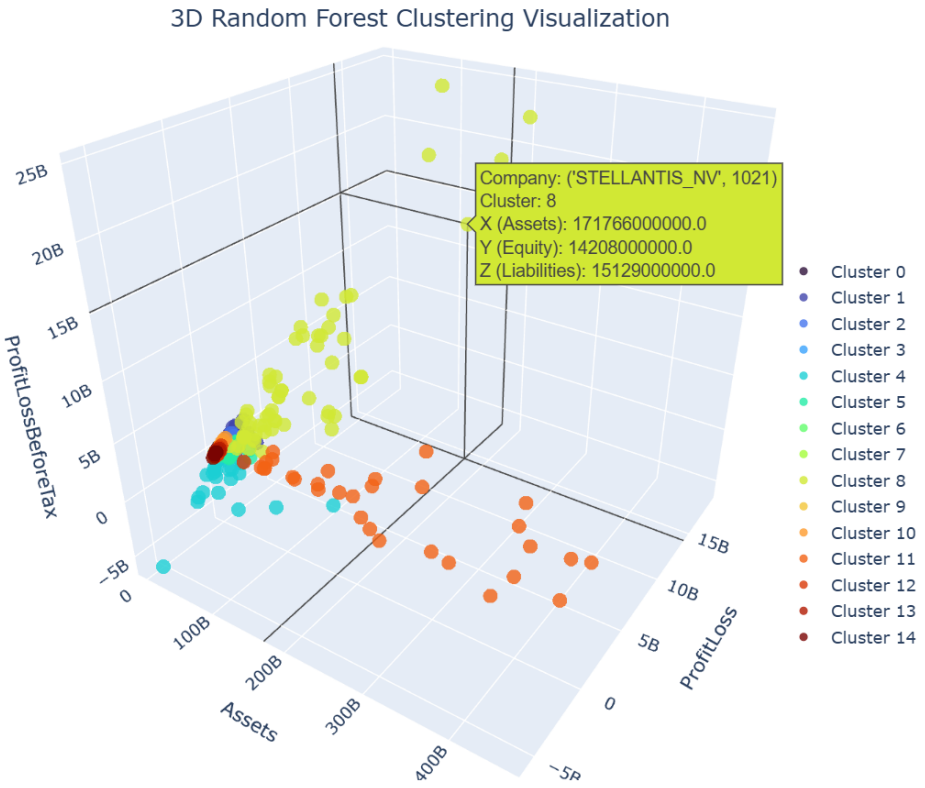

Figure 14: 3D scatter plot of Random Forest clustering results based on Assets, ProfitLoss, and ProfitLossBeforeTax

While these 3D visualizations are valuable in assessing cluster distribution, they are inherently limited by the fact that they only display a subset of the financial features used for clustering. Additionally, the presence of companies with extremely high financial values dominates the scale, making it difficult to interpret the structure of clusters with smaller values. To address this limitation, an alternative visualization technique is explored in the next section.

6.3.2 3D Visualization Using t-SNE

Unlike the other clustering methods, Random Forest clustering enables the use of t-Distributed Stochastic Neighbor Embedding (t-SNE) for visualizing high-dimensional data in a lower-dimensional space, this is because the dataset for this model contains imputed data to handle the missing data. t-SNE is a non-linear dimensionality reduction technique that preserves local structure, making it particularly useful for assessing cluster cohesion and separation in complex datasets.

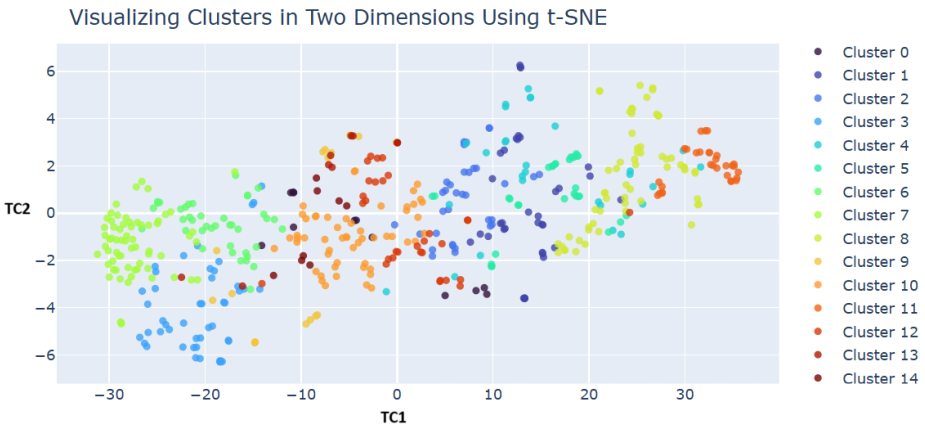

Figure 15: 2D t-SNE visualization of Random Forest clustering results

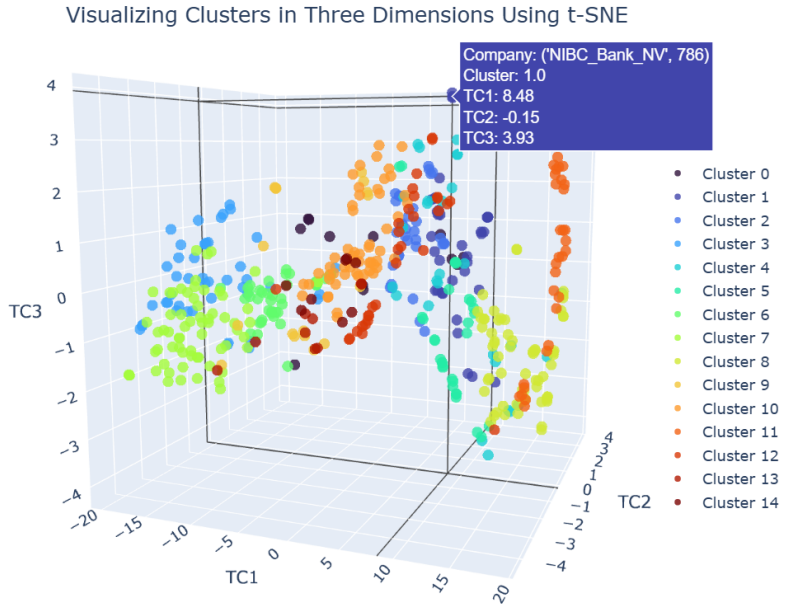

Figure 16: 3D t-SNE visualization of Random Forest clustering results.

As shown in Figures 15 and 16, the t-SNE projections reveal the underlying structure of the Random Forest clusters. While certain clusters form clearly distinct groupings, others exhibit partial overlap, suggesting that some financial feature interactions remain challenging to separate even after dimensionality reduction. Nevertheless, t-SNE provides a more holistic view of the clustering landscape by incorporating the entire feature set rather than just three selected

financial metrics. This makes it a powerful tool for evaluating the effectiveness of clustering approaches in high-dimensional financial data.

6.3.3 Silhouette score and Cluster Cohesion

To evaluate clustering quality, the Silhouette score was calculated using the proximity-based similarity matrix derived from the Random Forest model. This approach measures intra-cluster cohesion and inter-cluster separation in a way that reflects the underlying tree-based distance metric.
For this clustering approach, the computed Silhouette score was $S = 0.065$, which is lower than both hierarchical clustering ($S = 0.465$) and Kmedoids ($S = 0.135$). This suggests that while the clusters provide meaningful groupings, there is substantial overlap, indicating that the clustering structure may not be as well-defined as in the previous methods.
Potential reasons for the lower Silhouette score include:

- The complex nature of financial data, where company characteristics do not always form clearly distinct groups.
- The high dimensionality of the dataset, which can introduce noise and make it harder for clustering algorithms to form well-separated groups.
- Clustering could be influenced if the imputation of the missing data was not that good.
- The reliance on the proximity matrix, which, while powerful, may not always reflect the best clustering boundaries for this dataset.

6.3.4 Summary of Findings

Despite the relatively low Silhouette Score, the Random Forest clustering method provides an alternative perspective on company segmentation by capturing non-linear interactions between features, which traditional distance-based methods may overlook. The key observations from this method include:

- The clustering structure resulted in relatively well-balanced groups, though some clusters were notably larger or smaller than others.
- The use of t-SNE allowed for an alternative visualization of cluster separability, highlighting areas of both distinct grouping and overlap.
- The Silhouette score (0.065) suggests that the clusters, while meaningful, exhibit significant proximity-based overlap.
- Unlike hierarchical and Kmedoids clustering, this approach does not rely on traditional distance metrics, making it more robust to nonlinear relationships.

Overall, Random Forest clustering provides a valuable comparison by incorporating tree-based proximity measures, though its lower Silhouette score indicates room for improvement in cluster separation.

7 Conclusion

The primary objective of this study was to explore how clustering algorithms can be applied to XBRL-based financial data to group companies with similar financial profiles. Using three distinct clustering approaches, hierarchical clustering, Kmedoids, and Random Forest clustering has this research demonstrated how different methodologies capture patterns in financial reporting and revealed the importance of complete XBRL data.

7.1 Key Findings

7.1.1 Inconsistencies and Challenges in XBRL Data

One of the major challenges encountered during this study was the presence of missing or inconsistent data within XBRL filings. Hierarchical and Kmedoids clustering therefore required a custom distance function to handle missing values, whereas the Random Forest approach leveraged an imputation model to accommodate missing data.

These inconsistencies highlight potential inaccuracies in financial reporting, including:

- Companies failing to report specific financial facts, leading to gaps in datasets. Like the features CurrentAssets and NoncurrentAssets. This limited the possibility of calculating new ratios and features for certain companies.
- Variability in the application of XBRL taxonomies, where similar financial metrics were tagged differently by different companies.
- Structural differences in reports that complicated direct comparisons between entities.

These findings suggest that while XBRL aims to standardize financial disclosures, real-world implementation inconsistencies persist, necessitating careful preprocessing before large-scale analysis. If these complications can be removed or reduced, than the preprocessing becomes less heavy of a job.

7.1.2 Scalability and Big Data Processing

Efficiently processing the large volume of XBRL financial data necessitates the use of scalable data processing frameworks. While the current data gathering process is not the most efficient, this issue can be resolved once a database is available with preprocessed data ready for use. Additionally, this study found that although clustering methods effectively identified distinct groupings, they can become computationally demanding as dataset size increases, particularly since hierarchical clustering and Kmedoids clustering scale with $\mathcal{O}(n^2)$. This scalability challenge makes these approaches less practical for really large

datasets. In contrast, Random Forest clustering proved to be more efficient, as it constructs similarity matrices without relying on explicit distance calculations, thereby mitigating the scaling issue during the modeling phase.

These findings indicate that:

- Methods like hierarchical clustering and Kmedoids, while insightful, may not be practical for very large datasets due to their computational complexity.
- Random Forest clustering methods provide an alternative for handling high-dimensional financial data, but at the cost of needing complete data.

7.1.3 Financial Clusters

The application of clustering algorithms to XBRL financial data revealed meaningful patterns that correspond to industry trends and financial performance characteristics:

- **Hierarchical clustering** produced well-separated clusters with a moderate Silhouette score (0.465), suggesting that companies exhibit distinct financial structures that this method can effectively capture. The results indicate that hierarchical clustering is well-suited for identifying clear-cut financial groupings, reinforcing the notion that structural differences exist within corporate financial profiles.
- **Kmedoids clustering** created stable and interpretable financial groupings, albeit with a lower Silhouette score (0.135). While the separation between clusters was not as strong, the method still highlighted meaningful financial similarities among companies, making it a viable approach for industry segmentation. One drawback of this method in this research is its inherent randomness in selecting initial medoids, which can lead to variations in clustering outcomes. While this unpredictability is not ideal, it can be advantageous when one wants to designate specific companies as starting points for clustering, allowing for a more controlled analysis of financial segmentation.
- **Random Forest clustering** uncovered more nuanced and complex relationships between companies, reflected in its lower Silhouette score (0.065). While the weaker cluster separation suggests that financial profiles may not always form distinct boundaries, this method excels at capturing nonlinear financial similarities. The t-SNE visualization further demonstrated how companies distribute across financial dimensions, providing an intuitive way to explore these intricate relationships.

Despite varying levels of cluster separation, each method provided a unique lens through which to analyze financial patterns, contributing to a more comprehensive understanding of how companies align based on their financial data.

7.2 Business conclusions

The results of this study offer several practical applications for financial institutions, particularly banks, in leveraging clustering techniques for improved financial decision-making.

Banks can integrate these clustering techniques into AI-driven credit assessment models to enhance fairness in evaluating companies. This approach is particularly beneficial for small and medium-sized enterprises (SMEs) that lack an extensive credit history. By placing companies in clusters based on financial similarities rather than relying solely on historical credit data, financial institutions can perform more accurate and context-aware risk assessments.

Clustering methods can also be used to determine how far each company deviates from the central point of its respective cluster. Companies that fall significantly further from the center may be experiencing financial distress or structure their reports differently compared to their peers. This anomaly detection capability can serve as an early warning system for financial instability.

From a banking perspective, identifying outlier companies is critical for risk management. Banks can closely monitor these firms, as they may represent higher credit risks. Additionally, the centroids of clusters can serve as benchmarks for assessing financial norms within specific industries or company types. By comparing a company's financial profile to the cluster it belongs to, banks can refine their credit scoring methodologies and gain deeper insights into sector-specific financial behaviors.

SBR Nexus can use these insights to enhance standardization, for example, by incorporating improved validation rules in consultation with software providers. By ensuring more consistent and structured XBRL filings, financial clustering techniques can become more reliable, ultimately benefiting banks, regulators, and businesses that rely on high-quality financial data.

In conclusion, this research demonstrates that clustering algorithms offer a powerful tool for analyzing financial data from XBRL filings, enabling the identification of meaningful company groupings and industry trends. However, challenges related to data quality, scalability, and interpretability must be addressed to fully realize the potential of XBRL-driven financial analytics.

8 Discussion

This section outlines the limitations of the study and suggests directions for further research.

One of the main difficulties stems from the substantial diversity in company financials, with large discrepancies in assets and equity levels. Some companies report extremely high financial figures, while others operate on a much smaller scale. This variation complicates the clustering process, as traditional distance-based methods may struggle to balance such differences, potentially leading to bias toward larger firms.

Furthermore, although clustering is an unsupervised learning approach, a validation set could have been useful for assessing the reliability of results. Since no ground truth labels exist, external validation techniques such as comparing clusters to known financial categories or benchmarking them against established classification systems could help evaluate model performance. Initially, the SBI codes (which classify companies by sector) seemed like a logical benchmark for validation. However, their effectiveness was limited for two reasons: (1) the dataset lacked sufficient sectoral diversity, and (2) companies within the same industry can have vastly different financial structures due to differences in size, business model, or strategic focus. Instead of using sector classification as a validation metric, incorporating it as a feature in the clustering model could improve the segmentation process by providing additional financial context.

Another key consideration is the choice of distance metric used for clustering. While the selected methods provided meaningful groupings, different distance measures such as Mahalanobis distance (which accounts for correlations between financial variables) or cosine similarity (which emphasizes relative proportions rather than absolute values) could influence the cluster formation in another way. Although different distance metrics have been explored. The current custom metric seemed the most appropriate for this problem.

A notable limitation of this research is the potential biases introduced by imputation. While imputation improved dataset completeness, it may have distorted financial patterns. The chosen imputation strategy, based on correlating features, aimed to create realistic estimates, but the true values are likely to differ. Additionally, financial data can exhibit significant year-to-year fluctuations, where companies report substantial changes in key metrics. The imputation approach may not always capture external shocks, such as the financial impact of COVID-19, unless explicitly trained to account for such events. Therefore, potential biases remain in this study.

Additionally, the research was conducted using the IFRS taxonomy, whereas companies in the Netherlands primarily report under a different taxonomy (NT and FT). While this does not prevent reproducibility, adapting the methodology

to the Dutch taxonomy would require modifications to ensure compatibility. A notable advantage of the Dutch taxonomy is that certain fields are mandatory for banks, ensuring that financial institutions systematically report specific data points. If clustering were applied exclusively to these key fields, it could improve both the completeness and reliability of the dataset, reducing the influence of missing values and enhancing consistency in financial clustering.

For future research, several avenues could be explored:

- **Improving Data Consistency**: XBRL reporting inconsistencies remain a major challenge. Future research could explore automated methods for detecting and correcting tagging errors, missing values, and reporting discrepancies to improve data quality.
- **Feature Importance in Clustering**: Rather than treating all financial variables equally, integrating feature importance methods (such as using a Random Forest model to rank the most relevant financial features) could refine clustering results by giving more weight to key financial indicators.
- **Alternative Clustering Methods**: This study focused on hierarchical clustering, Kmedoids, and Random Forest clustering. Future research could explore deep clustering approaches, such as autoencoder-based clustering or self-organizing maps, which might better capture non-linear financial relationships. Additionally, density-based methods like DBSCAN could be tested to identify financial outliers and irregular patterns in corporate reporting.
- **Semi-Supervised Approaches**: While this study was purely unsupervised, future work could investigate semi-supervised learning techniques where partial labels (e.g., known company classifications, financial risk ratings, or regulatory categories) are incorporated into the clustering process. This could improve model interpretability and provide a more structured approach to financial segmentation.
- **Enhanced Imputation Strategies**: While imputation was necessary to handle missing data, alternative imputation techniques could be tested to evaluate their impact on clustering results. If missing data remains an issue in future datasets, more advanced strategies such as domain-specific imputations or generative models could further refine clustering performance.

By addressing these challenges and refining clustering methodologies, future research can improve the accuracy, interpretability, and real-world applicability of financial clustering models.

References

[1] Alan C. Acock. "Working with Missing Values". In: *Journal of Marriage and Family* 67.4 (2005), pp. 1012–1028. DOI: 10.1111/j.1741-3737.2005.00191.x. URL: http://dx.doi.org/10.1111/j.1741-3737.2005.00191.x.

[2] Bruce Mwiya et al. "Examining the effects of electronic service quality on online banking customer satisfaction: Evidence from Zambia". In: *Cogent Business & Management* 9 (2022), p. 2143017. DOI: 10.1080/23311975.2022.2143017.

[3] Gustavo E. A. P. A. Batista and Maria Carolina Monard. "An analysis of four missing data treatment methods for supervised learning". In: *Applied Artificial Intelligence* 17.5-6 (2003), pp. 519–533. DOI: 10.1080/713827181.

[4] Fernando Berzal and Nicolfás Matín. "Data mining: concepts and techniques by Jiawei Han and Micheline Kamber". In: *SIGMOD Rec.* 31.2 (June 2002), pp. 66–68. ISSN: 0163-5808. DOI: 10.1145/565117.565130. URL: https://doi.org/10.1145/565117.565130.

[5] Leo Breiman. "Random Forests". In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: https://doi.org/10.1023/A:1010933404324.

[6] Fan Cai, Nhien-An Le-Khac, and Tahar Kechadi. "Clustering Approaches for Financial Data Analysis: a Survey". In: (Sept. 2016). DOI: 10.48550/arXiv.1609.08520.

[7] Roger Debreceny and Glen L. Gray. "The production and use of semantically rich accounting reports on the Internet: XML and XBRL". In: *International Journal of Accounting Information Systems* 2 (2001), pp. 47–74. ISSN: 1467-0895. DOI: 10.1016/S1467-0895(00)00012-9.

[8] Pierre Geurts, Damien Ernst, and Louis Wehenkel. "Extremely randomized trees". In: *Machine Learning* 63.1 (2006), pp. 3–42. ISSN: 1573-0565. DOI: 10.1007/s10994-006-6226-1. URL: https://doi.org/10.1007/s10994-006-6226-1.

[9] Jib Seung Hwang, Choon Seong Leem, and Hyung Joon Moon. "A Study on Relationships among Accounting Transparency, Accounting Information Transparency, and XBRL". In: 1 (2008), pp. 502–509. DOI: 10.1109/ICCIT.2008.221.

[10] Chunhui Liu et al. "The impact of XBRL adoption in PR China". In: *Decision Support Systems* 59 (2014), pp. 242–249. ISSN: 0167-9236. DOI: https://doi.org/10.1016/j.dss.2013.12.003. URL: https://www.sciencedirect.com/science/article/pii/S0167923613002923.

[11] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation Forest". In: *2008 Eighth IEEE International Conference on Data Mining*. 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17.

[12] Ulrike von Luxburg. "A tutorial on spectral clustering". In: *Statistics and Computing* 17.4 (2007), pp. 395–416. ISSN: 1573-1375. DOI: 10.1007/s11222-007-9033-z. URL: https://doi.org/10.1007/s11222-007-9033-z.

[13] Narcisa Roxana Mosteanu and Alessio Faccia. "Digital Systems and New Challenges of Financial Management – FinTech, XBRL, Blockchain and Cryptocurrencies". English. In: *Quality – Access to Success* 21.174 (Feb. 2020), pp. 159–166. ISSN: 1582-2559.

[14] Fionn Murtagh and Pedro Contreras. "Algorithms for hierarchical clustering: an overview". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.1 (2012), pp. 86–97. DOI: 10.1002/widm.53.

[15] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. "On spectral clustering: Analysis and an algorithm". In: *Advances in Neural Information Processing Systems*. Vol. 14. 2001, pp. 849–856.

[16] Ahmed Mahdi Sahi et al. "A systematic literature review of financial disclosure by financial institutions". In: *Cogent Business Management* 9.1 (2022), p. 2135210. ISSN: 2331-1975. DOI: 10.1080/23311975.2022.2135210.

[17] Tao Shi and Steve Horvath. "Unsupervised Learning with Random Forest Predictors". In: *Journal of Computational and Graphical Statistics* 15.1 (2006), pp. 118–138. DOI: 10.1198/106186006X94072.

[18] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison Wesley, 2006, pp. 150–172.

[19] Kai Ming Ting et al. "Overcoming Key Weaknesses of Distance-based Neighbourhood Methods using a Data Dependent Dissimilarity Measure". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 1205–1214. ISBN: 9781450342322. DOI: 10.1145/2939672.2939779. URL: https://doi.org/10.1145/2939672.2939779.

[20] Valentina Tohang, Amelia Limijaya, and Marcus Chitrahadi. "An Analysis of the Impact of XBRL Filings Towards Information Asymmetry in Indonesia". In: (2020), pp. 330–335. DOI: 10.1109/ICIMTech50083.2020.9211114.

[21] Roslee Uyob, Ram Al Jaffri Saad, and Aidi Ahmi. "A review of the study on the impacts of the extensible business reporting language (XBRL)". In: *International Journal of Scientific & Technology Research* 8.9 (2019), pp. 2320–2329.

[22] G. Weckman W. Young and W. Holland. "A survey of methodologies for the treatment of missing values within datasets: limitations and benefits". In: *Theoretical Issues in Ergonomics Science* 12.1 (2011), pp. 15–43. DOI: 10.1080/14639220903470205.

[23] Yong Wen. "Research on the Improvement Path of Financial Report Based on XBRL". In: *Journal of Physics: Conference Series* 1756 (Feb. 2021), p. 012018. DOI: 10.1088/1742-6596/1756/1/012018.

[24] Ian R. White, Patrick Royston, and Angela M. Wood. "Multiple imputation using chained equations: Issues and guidance for practice". In: *Statistics in Medicine* 30.4 (2011), pp. 377–399. DOI: 10.1002/sim.4067. URL: http://dx.doi.org/10.1002/sim.4067.

[25] Jesper N Wulff and Linda Ejlskov Jeppesen. "Multiple imputation by chained equations in praxis: guidelines and review". In: *Electronic Journal of Business Research Methods* 15.1 (2017), pp. 41–56.

[26] Ju-Chun Yen and Tawei Wang. "The Association between XBRL Adoption and Market Reactions to Earnings Surprises". In: *Journal of Information Systems* 29.3 (Sept. 2015), pp. 51–71. ISSN: 0888-7985. DOI: 10.2308/isys-51039. eprint: https://publications.aaahq.org/jis/article-pdf/29/3/51/8180/isys-51039.pdf. URL: https://doi.org/10.2308/isys-51039.