

MASTERARBEIT | MASTER'S THESIS

Titel | Title

SHIP.AHOI - Visualization of Ultrametric Cluster Hierarchies

verfasst von | submitted by
Alexander Moser BSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of
Master of Science (MSc)

Wien | Vienna, 2025

Studienkennzahl lt. Studienblatt | Degree
programme code as it appears on the
student record sheet:

UA 066 921

Studienrichtung lt. Studienblatt | Degree
programme as it appears on the student
record sheet:

Masterstudium Informatik

Betreut von | Supervisor:

Univ.-Prof. Dipl.-Inform.Univ. Dr. Claudia Plant

Mitbetreut von | Co-Supervisor:

Dr. Anna Beer BSc MSc

Acknowledgements

I want to express my sincere gratitude to my supervisor and co-supervisor for the numerous meetings we have had over the past few months. They made the entire thesis possible and were always helpful and patient.

Additionally, I would like to thank all the contributors from the Visualization and Data Mining and Machine Learning departments for sharing their knowledge with me.

In addition, I would like to express my sincere gratitude to each and every interview partner who generously gave their time to the interviews conducted for this project. I am incredibly grateful to everyone for their willingness to share their perspectives and insights, which were a crucial component of the work.

Finally, I wish to thank my friends, my family, and my girlfriend for their continuous support and encouragement throughout this journey.

Kurzfassung

Das SHIP Framework kombiniert hierarchisches und partitionendes Clustering mit einem flexiblen Ansatz, der einen schnellen Wechsel zwischen verschiedenen Parametern und Lösungen ermöglicht. Diese Masterarbeit stellt SHIP.AHOI (Similarity-Hierarchy-Partition Advanced Human-centered Optimization Interface) vor, ein interaktives Designkonzept, welches Anwender*innen mit verschiedenen Erfahrungsleveln ermöglicht, das SHIP Framework zu verwenden und zu testen. Basierend auf einer iterativen, nutzerzentrierten Designstudie wurde ein Design entwickelt, welches die Anforderungen der Zielgruppen erfüllt. Besonders wurde auf verknüpfte Visualisierungen sowie eine gute Erklärung der Funktionen für Benutzer*innen mit unterschiedlichem Vorwissen gelegt.

Eine durchgeführte Anwenderstudie zeigt, dass das SHIP.AHOI-Design nicht nur die Usability-Ziele erfüllt, sondern auch Anwender*innen dabei hilft, detaillierte Einblicke in die Struktur ihrer Daten zu bekommen. Durch die Integration der Visualisierung in den Clustering-Workflow verwandelt SHIP.AHOI den Prozess von einer Trial-and-Error-Methode in eine explorative Analyse. In diesem sind alle wichtigen Daten und Visualisierungen an einem Ort verfügbar, ohne dass eine einzige Zeile Code geschrieben werden muss.

Abstract

The SHIP framework combines hierarchical and partitional clustering with a flexible approach that enables fast switching between different parameters and solutions. This master's thesis introduces SHIP.AHOI (Similarity-Hierarchy-Partition Advanced Human-centered Optimization Interface), an interactive design concept that allows both users with little experience and experts in the clustering field to use the SHIP framework. Based on a user-centered design study, a design was developed iteratively that meets all requirements of the targeted users. Special focus was placed on linked visualizations, as well as a good explanation of the features for users across different skill levels.

A user study conducted shows that the SHIP.AHOI design not only meets the usability goals set but also helps users gain a deeper understanding of their data. By integrating visualizations into the clustering workflow, SHIP.AHOI changes the process from a trial-and-error method into an exploratory analysis. In this process, all data, visualization, and clustering results are available in one place without having to write a single line of code.

Contents

Acknowledgements	iii
Kurzfassung	v
Abstract	vii
List of Figures	xi
List of Tables	xiii
1. Introduction	1
2. Theoretical Foundations	3
2.1. SHIP Framework	3
2.2. Visualization Techniques for Clustering	5
2.3. Usability and Interactivity in Data Analysis	8
3. Related Work	11
3.1. Existing Approaches	11
3.2. Analysis of Existing Tools	18
3.3. Challenges	19
3.4. Opportunities	19
4. Methodology: Design and Evaluation Studies	21
4.1. Design Study Methodology	21
4.2. User Study	24
5. System Design and Prototyping	25
5.1. Requirements Analysis	25
5.2. Design of the Interactive Clustering Website	29
6. Final Design and Functionalities	37
6.1. Getting Started and Onboarding	37
6.2. Exploring Dataset	40
6.3. Tuning Parameters	42
6.4. Interactive Clustering	46
6.5. Evaluating Clustering solutions	48

Contents

6.6. History	49
6.7. Comparison	50
7. Evaluation and Results	51
7.1. Results of the User Study	52
7.2. Discussion of User Experience	60
8. Discussion	65
9. Conclusion	69
Bibliography	71
A. Appendix	77
A.1. Technical Implementation SHIP.AHOI	77
A.2. User Study Document for Users	78
A.3. System Usability Scale Results Detailed	84

List of Figures

2.1. Regular dendrogram vs stacked trees	7
3.1. Overview of different clustering visualization approaches.	12
3.2. Overview of different hybrid visualization approaches.	17
4.1. Design study steps	22
6.1. Starting page of the application	38
6.2. Example visualization for SHIP explanation	39
6.3. Question-based onboarding	39
6.4. CSV upload configuration options	41
6.5. Heatmap on the Data Explorer page	42
6.6. Sidebar parameter options with tooltip	42
6.7. K-selection page top part	44
6.8. K-selection page lower part	45
6.9. Clustering page plots	46
6.10. Clustering page, ground truth visualization	47
6.11. Cluster Explorer page	48
6.12. History page cards	49
6.13. Comparison page	50
7.1. System Usability Scale results	60
7.2. User Study Task completion times	61
a.1. Basic architecture of the SHIP.AHOI implementation	77
a.2. System Usability Score detailed bar chart and heatmap	84

List of Tables

5.1. Overview of design study participants	26
5.2. Problems reported by participants	27
5.3. User study Tasks	28
7.1. Overview of user study participants	52

1. Introduction

Clustering is a fundamental technique in unsupervised learning, with the goal of grouping similar data points while maximizing dissimilarity between groups. This process is valuable to domain experts across a diverse range of fields, such as astronomy and biology. It enables them to discover structures and patterns in their data. However, this can be rather challenging for large and high-dimensional datasets. To be successful, users must have an in-depth understanding of data characteristics, algorithm behavior, and parameter choices. Furthermore, the selection of fitting parameters is a major challenge for many users. Then the user faces the challenge of interpreting the results and generating insights from them. For this, support such as visualizations are an essential tool in making the process easier and more efficient.

Traditional approaches for visualizing clustering results include the use of static scatterplots and dendrograms. However, these static visualizations only provide limited information and are disconnected from the clustering process. Furthermore, it is challenging to compare different solutions with each other. Creating more advanced visualizations often requires coding skills and are hard to reuse, as clustering results are frequently recalculated from scratch and adjusted.

Existing visualization tools are often domain-specific, lack clustering capabilities, or are complicated to use. This highlights the need for an approach that deeply integrates clustering and interactive visualization. Interactivity is crucial for understanding and engagement [1]. While static scatterplots and dendrograms can provide basic information, they treat visualization as a post-processing step rather than as a core element of the clustering workflow. This separation between algorithms and visualization limits users' ability to explore clustering results and extract insights, as this connection makes the process more transparent and helps users to be more effective [2].

Another critical consideration in clustering workflow is that there is often no single "correct" solution for clustering data. This makes it essential to analyze and compare the multiple possible results. A critical technique for this purpose is to visually inspect the data, both to fine-tune parameters or algorithms and to verify the validity of the solutions.

Consequently, generating just one single clustering solution is rarely enough for compelling data exploration and analysis. Instead, it is often required to create multiple, different solutions and compare them to gain a comprehensive understanding of the data. However, this is computationally expensive, especially for large or high-dimensional datasets.

1. Introduction

A solution to this problem, that this master’s thesis explores, is the framework proposed by Draganov et al. [3]. It introduces the SHIP (Similarity-Hierarchy-Partition) framework, a promising approach to address the issue of quickly exploring different clustering algorithms and parameter settings. By leveraging the hierarchical structure of a selected ultrametric, the framework enables fast transitions between different algorithms and potential partitions. The benefits of rapidly switching between clustering options remain unrealized in existing tools. The goal is to develop an interactive visualization that brings the benefits of the SHIP framework directly to users.

This thesis presents an interactive visualization design that leverages the advantages of the SHIP framework and clustering exploration. It is designed to be easy to use, provide detailed insights and explanations, making it accessible to students and people with basic clustering knowledge, but also to researchers who are interested in the SHIP framework. By doing so, it helps users to understand the SHIP framework better. Additionally, the design seeks to support researchers interested in the SHIP framework, enabling them to experiment with the concepts presented in the paper.

The following work starts by outlining the theoretical foundations necessary to understand the SHIP framework, as well as the fundamentals of clustering and visualization. The Related Work section then reviews existing libraries and tools relevant to the topic. Then, the Methodology: Design and Evaluation Studies chapter presents the design study, following the approach described by Sedlmair et al. [4], as well as the user study. The chapters on System Design and Prototyping and Final Design and Functionalities present the outcomes of the design study, including the final developed design and the crucial design decisions made. The Evaluation and Results chapter focuses on the evaluation results and the evaluation interviews. This is followed by a Discussion section and the Conclusion sections, which summarize the findings of the thesis.

2. Theoretical Foundations

In order to give the reader a solid understanding of why certain parts of SHIP.AHOI are designed as they are, this chapter will first explain the utilized SHIP framework and its benefits. Later the focus shifts to different visualization techniques, which can be used to visualize clustering. Furthermore, the chapter covers how interactivity plays a crucial role in the usability of applications.

2.1. SHIP Framework

The SHIP (Similarity-Hierarchy-Partition) framework is a novel clustering approach. It combines the benefits of hierarchical and partition-based clustering. It offers the possibility to switch between parameters like k -center, k -median, and different partition methods to find a k value nearly instantly after the initial ultrametric tree is constructed in the first preprocessing step [3].

The SHIP framework can be represented by three stages, each corresponding to a parameter selection.

2.1.1. Workflow Overview

The framework begins by constructing an ultrametric tree structure from the input data. This initial construction takes time comparable to traditional clustering algorithms and is performed only once per dataset and ultrametric combination. The result becomes the starting point for all subsequent clustering parameters.

In its first step, the SHIP framework provides selection of tree structures such as the KDTree, XTree, BallTree, CoverTree, and DCTree. Each one is then transformed into a LCA-tree, which define the distance between two points as their lowest common ancestor in the tree. A lowest common ancestor is the lowest node that connects two points in the tree to each other. This construction satisfies the strong triangle inequality and therefore provides a relaxed ultrametric. Hence, the higher the lowest common ancestor between two points is in the tree structure, the more dissimilar they are.

In its first step, the SHIP framework lets the user select a tree-based similarity model (e.g., KDTree, XTree, BallTree, CoverTree, DCTree). Each such structure is converted into a lowest-common-ancestor (LCA) tree by assigning a non-negative value to every internal node; the distance between two points is then defined as the value at the LCA of

2. Theoretical Foundations

their leaves. Because these node values are non-decreasing along any path from the leaves to the root, the induced distance satisfies the strong triangle inequality and therefore is a relaxed ultrametric. Intuitively, the higher (i.e., closer to the root and with a larger node value) the LCA of two points, the more dissimilar the points are.

Further, the DCTree is of special interest. It is based on the DC-distance, which creates a distance function in which DBSCAN, spectral clustering, and k -center clustering are equal. Therefore, offering the SHIP framework even more capabilities and flexibility [5].

The second step of the framework is the selection of a power parameter (z), where $z = 0$ results in a k -center clustering that minimizes maximum distances, $z=1$ generates k -median, which minimizes total distance and $z=2$ generates k -means, which minimizes squared distance. The power parameter can be set to any integer number greater than or equal to zero. The SHIP framework can compute all of these solutions from a single tree, which is created during its initial step [3].

The third and final step is the selection of a partition method. The user does not have to manually define the number of clusters they expect from the clustering. Instead, they can select between different techniques that will automatically determine the best number of clusters for them. Some examples of methods they can choose from are stability-based methods, the elbow method, threshold-based partitioning, and median-of-eblow. However, it is also possible to manually select a number of clusters if desired [3].

2.1.2. Key Advantages

The key advantage is its fast computation once an initial similarity tree is created. Another benefit that derives from the rapid speed of the framework is the possibility to offer fast exploration of different parameters. This is especially important for explorative data analysis, where a lot of solutions are usually generated.

As the SHIP framework generates an ultrametric tree structure, this ultrametric property ensures that the tree can be visualized properly without any ambiguities or inversions. Additionally, the tree provides the opportunity to visualize additional information about the hierarchy directly to the user, allowing for a better understanding of the clustering process and results [3].

2.1.3. Ultrametrics

Ultrametrics are important to the SHIP frameworks first step, where they make it possible to construct the hierarchical structure used to efficiently derive the possible clustering solutions.

Below is the definition of a *relaxed* ultrametric used in SHIP:

Let

$$d : X \times X \longrightarrow \mathbb{R}_{\geq 0}$$

2.2. Visualization Techniques for Clustering

be any function on a set X . We say d is a **relaxed ultrametric** if, for all $x, y, z \in X$, it satisfies:

1. Non-negativity and symmetry:

$$d(x, y) = d(y, x) \geq 0.$$

2. Strong triangle inequality:

$$d(x, z) \leq \max\{d(x, y), d(y, z)\}.$$

Note: The SHIP framework *does not* require $d(x, x) = 0$ or $d(x, y) = 0 \iff x = y$. That is what makes it “relaxed” in the context of the SHIP framework, in contrast to a regular ultrametric.

A selected distance function that satisfies these requirements is then used to construct a lowest common ancestor tree, from which in the following steps the different clustering solutions are efficiently extracted [3].

In the resulting tree, every internal node has a value associated with it, representing the distance at which its child clusters merge. These node values are non-decreasing along every path from the leaves to the root. This property ensures that higher nodes in the tree correspond to larger, more general clusters, while lower nodes with smaller values represent finer, more specific clusters [3].

2.2. Visualization Techniques for Clustering

As mentioned in Chapter 1, the visualization of the clustering also plays a key role in making the clustering more understandable to the user [2, 6]. It is, however, essential to add that in the visualization of clustering, one of the most critical aspects in choosing a fitting visualization is the task the user aims to accomplish. There are no right or wrong visualizations; the choice depends heavily on the user’s task and preferences.

The most common visualization techniques for clustering and hierarchical clustering encountered in the literature are described now [7, 8].

2.2.1. Point-based Clustering Visualizations

The following methods focus on visualizing the data points:

- **Scatterplots (2D/3D):** Scatterplots are one of the most fundamental and widely used visualizations for exploring the structure of datasets. In their simplest form, they plot features in a two or three-dimensional space where each axis is a dimension. However, datasets often have more than three dimensions. Therefore, typically, dimensionality reduction (DR) techniques are used to generate fewer dimensions.

2. Theoretical Foundations

These methods project high-dimensional data into the 2D/3D space while attempting to preserve key structural relationships between points. Prominent techniques are Principal Component Analysis (PCA) [9], t-Distributed Stochastic Neighbor Embedding (t-SNE) [10], or Uniform Manifold Approximation and Projection (UMAP) [11]. All of them work differently, and it is vital to have a selection of techniques as there is no single best DR technique [12].

The user can inspect the low-dimensional representation to find patterns, clusters, and outliers. Points belonging to the same cluster are usually assigned the same color, making the clusters easier to distinguish. An example can be seen in Figure 6.11 on page 48.

Additionally, it is possible to add further information to a scatterplot visualization, like visualizing decision boundaries, as can be seen in Figure 3.1d on page 12.

- **Scatter Plot Matrix:** Displays a grid of pairwise scatter plots for each combination of features; each grid in the feature looks a lot like a scatterplot. However, each of the sub-scatterplots can get really small for high-dimensional datasets.

2.2.2. Hierarchical Clustering Visualization

Some visualizations designed for hierarchical clustering, with a focus on its tree structure, are:

- **Dendrogram:** The most common one, a tree diagram illustrating the hierarchy of merges. Branch heights typically represent the distance (dissimilarity) at which clusters are combined or split. The hierarchical clustering can be cut at a height to get a specific number of clusters. It is also possible to extend the dendrogram visualization to include continuous features, as shown in Figure 2.1 on page 7 [13].
- **Heatmap:** Visualizes the data with points as rows and features as columns. Rows and columns are reordered based on hierarchical clustering, and the cells are colored based on the values. A dendrogram is typically placed alongside the heatmap to show the corresponding cluster memberships [14].
- **Circular Dendrogram / Radial Tree:** Displays the dendrogram in a circular layout, this is often used for visualizing large hierarchies that do not fit a regular dendrogram. The distance (dissimilarity) between clusters is here often not the height like in a regular dendrogram, but the level in the hierarchy [13].
- **Treemap:** They are often found on data dashboards and represent a dense representation of data. They use nested rectangles to show the different clusters and their hierarchical connection. Each rectangle is a cluster or sub-cluster. The size of the rectangle usually represents the number of data points [15].
- **Sunburst Chart:** Similar to treemap, but the hierarchy is shown as radial segments. Each segment corresponds to a node in the hierarchy, with the root in the center.

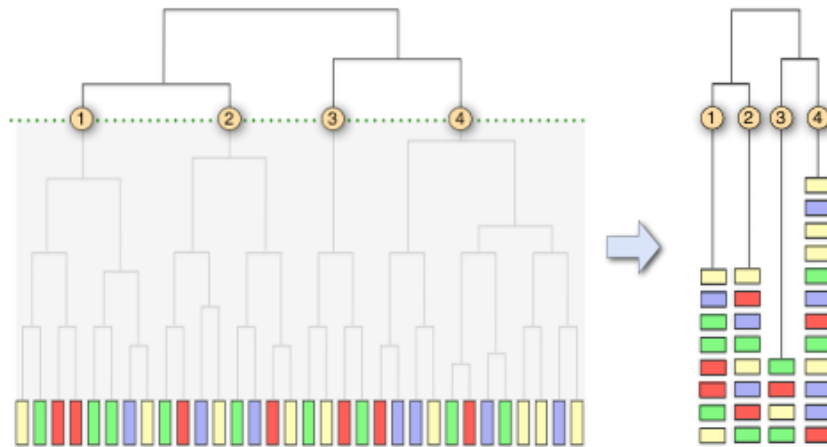


Figure 2.1.: Visualization of a regular dendrogram on the right, while the same dataset on the left as a stacked tree visualization. Source: [18].

Sub-clusters are represented as outer rings surrounding their parent cluster [16]. This type of visualization also allows for interactivity, for example, by centering the diagram around another segment by clicking on it.

- **Icicle plot** The icicle plot represents stacked horizontal layers. Each layer represents one level of clustering. The highest level is the root, and each level below divides the clustering into smaller subclusters. Each cluster is shown as a bar segment, and the length of a segment reflects the number of items in that cluster [17]. Unlike a dendrogram, icicle plots do not represent merge height directly and are therefore more compact and space-efficient. This allows the user to see a lot of information at once.
- **Stacked Trees** The Stacked Tree represents an alternative visualization of a dendrogram it focuses on only keeping the highest levels and intermediate levels are displayed as stacked points. An example of this form of visualization can be seen in Figure 2.1 [18].

2.2.3. Summary-based Clustering Visualizations

Furthermore, it is valuable to explore some different summary-based visualizations. Unlike scatterplots or dendrograms, these techniques do not aim to show individual data points or hierarchical relationships. Instead, they summarize statistical properties or characteristics of clusters. Such visualizations can help users better understand the properties and quality of clusters. Some selected examples are:

- **Boxplots:** Boxplots can be used to show how a feature is distributed across different clusters. They highlight crucial aspects of the data, like median or mean value,

2. Theoretical Foundations

as well as variability using quartiles or percentiles, making it possible to compare feature distributions between clusters.

- **Bar Charts:** Bar charts can be used to display differences in average features across clusters. Furthermore, they are also suitable for visualizing quality metrics such as the Silhouette Score.
- **Cluster magnitude and cardinality:** These visualizations focus on cluster size and internal cohesion. For example, a bar chart can be used to represent the number of points assigned to each cluster (cardinality) or to show the total distance of all points to their respective centroid (magnitude), offering insights into both density and compactness.

2.3. Usability and Interactivity in Data Analysis

Classical data analysis involves analyzing data with established methods. It often follows a hypothesis-driven approach where hypotheses are formulated and then tested using the available data. Exploratory Data Analysis (EDA), however, focuses on approaching the data without a hypothesis and with a focus on exploring the data and finding interesting patterns and insights [19]. EDA connects visualization and data science by utilizing the power of visualization to help understand features, patterns, and how data is connected. Further, it can help find anomalies [20].

2.3.1. Interactive Visualization

While static visualization can be helpful, an interactive visualization is even more helpful and supports the exploration process, particularly for large datasets. Numerous important interaction methods are seen in tools and literature, some of them mentioned in Fekete [21] are:

Parameters:

- **Selecting the parameters** directly, for instance, set the amount of cluster or DBSCAN min distance value.
- **Adjusting methods** like the distance metric, linkage method or certain thresholds.

Comparing:

- **Side-by-Side View**, a common approach to display visualizations next to each other for comparison.
- **Difference Highlighting**, shows the differences between two or more clustering solutions, in one or more visualizations.
- **Comparison Metrics**, these quantitative metrics can measure similarity or quality. If possible, for example, the Adjusted Rand Index (ARI) can be displayed, or if

2.3. Usability and Interactivity in Data Analysis

the ground truth is not available, scores like the Silhouette Score can be calculated. These metrics can also be plotted to make them more visually appealing and to facilitate comparison among different solutions.

Manipulation:

- **Zooming + Panning**, especially useful for large datasets and large diagrams. Allows for better navigation and enlargement of certain parts.
- **Filtering**, dynamically selecting subsets of data based on filter options.
- **Relate**, allows the highlighting of data points between views and visualizations.
- **Reorder**, interactive changing of elements like rows or columns.
- **Details on Demand**, show more information on demand, for example, by hovering over a node in a graph. It can also be used to hide too many details in the main view.

Interactive Labeling or Steering:

Another aspect of how interactive visualization can be achieved is by giving users the ability to define custom relationships between data or reassign data points to clusters that they see fitting. As this is not part of the SHIP framework and is rarely used, we are not focusing on this further in the thesis.

2.3.2. User Guidance

For a tool like SHIP.AHOI, it is also essential to consider various aspects of user guidance, as some of the targeted users may not have in-depth knowledge about clustering and the SHIP framework. Therefore, it is crucial to consider how to guide users effectively while they use the platform. Moreover, it is vital to have a good initial impression, as mentioned by Sadana et al. [22]. Hence, it is also important to explain the most important aspects when the application is first launched.

To effectively guide users within the application, it is essential to explore strategies that help them understand and navigate the features. Therefore, Pohl et al. [23] explored various methods such as tutorials, tooltips, or AI chatbots. The main findings are that nudging can be effective, and tooltips have proven to be an easy and helpful tool. However, AI Chatbots are also preferred by many users.

Another important aspect in guiding the user is to have both progressive disclosure and context awareness. Progressive Disclosure means that complex functionality is introduced gradually, with basic operations being more prominent. This should reduce cognitive load, especially for new users. Contextual help, on the other hand, means that explanation and guidance, for example, should be displayed when needed and hidden when not required [24].

2. Theoretical Foundations

2.3.3. User Experience Heuristics

An important aspect to consider when designing a new system is to leverage lessons learned from previous projects to avoid mistakes and enhance the design. For this reason, Nielsen's 10 heuristics are considered throughout all design steps.

These guidelines state that a tool should use natural and straightforward dialogue, speaking the user's language. Furthermore, it should focus on minimizing the user's memory load and always provide fast feedback on actions, for example, with a loading screen while waiting. Other essential factors to consider are maintaining a consistent interface and wording, as well as clearly marking exits.

Regarding user guidance, it is also essential to provide shortcuts for experienced users. For error handling, it is important to prevent errors in the first place and, if something goes wrong, to provide helpful error messages. Last but not least, it is crucial to provide help and documentation that is easily accessible [25].

3. Related Work

3.1. Existing Approaches

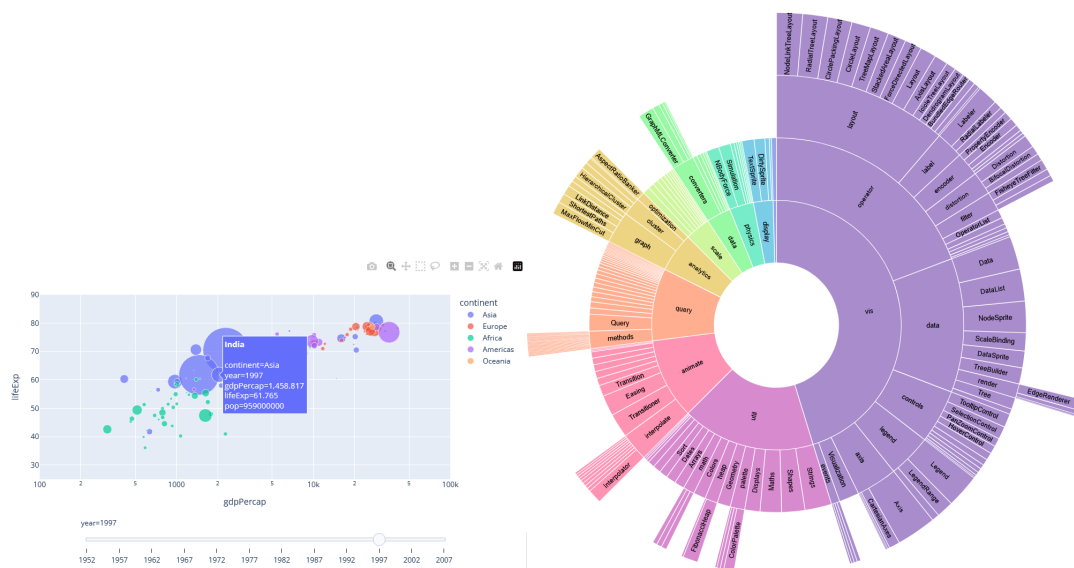
This section investigates two kinds of related works. Firstly, it explores approaches that can be used for visualization and incorporated into the SHIP.AHOI application. These tools and libraries provide clustering visualization capabilities only. In the second part of the chapter, applications and libraries are analyzed that focus not only on visualization but also implement clustering capabilities in some capacity, as they are similar to the planned SHIP.AHOI application.

3.1.1. Visualization Approaches

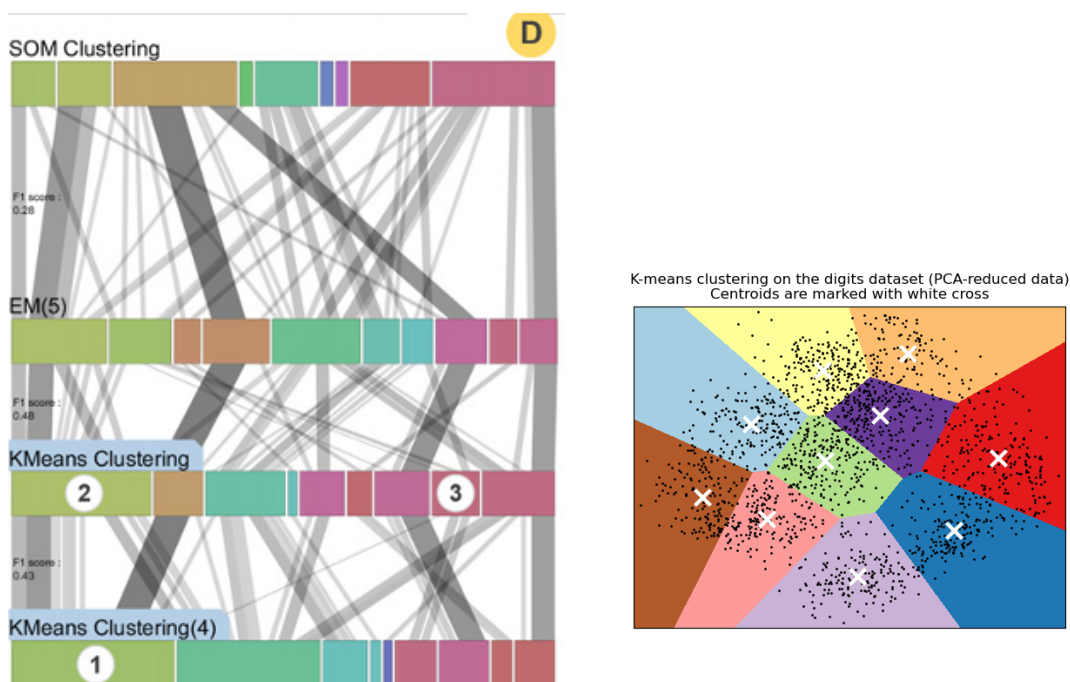
These approaches listed here only provide visualization for clustering results:

- **Plotly**: is a Python library designed for creating visualizations across a wide range of domains, often requiring only a few lines of simple code. It can also create world maps, heatmaps, 3D charts, and interactive visualizations. It offers the user the possibility to add sliders, drop-down menus, click events, and other interactive options. The interactive options are rather limited and include only basic options. An example can be seen in Figure 3.1a on page 12 [30].
- **Streamlit**: is a Python library designed for quickly building interactive web apps with minimal code. It focuses on simplicity and is therefore ideal for prototyping or creating quick dashboards. It integrates well with other data and plotting libraries and offers limited visualization options suited for clustering, like scatterplots, but no native visualization for hierarchical clustering. While Streamlit is well-suited for quick implementations, it is not focused on fine-grained visualization customization or interactive customization [31].
- **Bokeh**: is a Python library that is focused on creating a wide range of different and interactive visualizations. It focuses on efficiently handling large amounts of data and complex datasets. The framework offers numerous adjustment possibilities for its visualizations, including different interactive options. Some examples are hover tools, zooming, panning, and linked brushing across multiple plots. These features make data exploration more intuitive and engaging. Furthermore, Bokeh is easily integrable into web applications without having to leave the Python ecosystem. This makes it suitable for researchers, who need a versatile visualization for their research [32].

3. Related Work



(a) Plotly: Sliders tutorial visualization. Source: [26] (b) D3.js: Sunburst diagram example. Source: [27]



(c) XCluSim enhanced parallel sets view. Source: [28] (d) Matplotlib: Voronoi Diagram with centroids marked. Source: [29]

Figure 3.1.: Overview of different clustering visualization approaches.

- **Matplotlib:** is a flexible and one of the most widely used Python visualization libraries that can be used for clustering visualization. It offers the possibility to create both quick plots and customized visualizations from scratch. Furthermore, the framework integrates seamlessly with many other libraries, and numerous code examples are available online. Its focus is on static visualizations. However, it also supports some basic interactive functions, such as zooming, panning, and moving within plots. This makes the library flexible, satisfying the needs of both novice and advanced users. An example of the application in clustering can be seen in a Voronoi diagram Figure 3.1d on page 12 [33].
- **ECharts:** is a JavaScript library that is developed by the Apache Software Foundation. It offers a wide range of visualizations, including interactive and dynamic ones. The library features include tooltips, zooming, panning, and filtering, enabling users to interact with the data. ECharts also offers a vast collection of templates and ready-to-use examples that can be easily customized, making it accessible for users with less experience. Beyond simple visualization, it supports advanced options such as animations and dynamic updates, which are particularly valuable when working with streaming data or dashboards that require real-time responsiveness. Additionally, a focus is on having working visualizations for mobile devices. ECharts offers a comprehensive feature set for data visualization that can effectively address both standard and advanced visualization needs [34].
- **D3.js:** is a JavaScript library that is similar to ECharts. It offers a comprehensive suite of tools for creating interactive and dynamic visualizations across various domains. It offers flexibility and depth of customization that few other frameworks provide, allowing control over nearly all aspects of a visualization, from elements to interactions. This makes D3.js well-suited for building novel and powerful visualizations, which are not available off-the-shelf in other libraries. Another advantage of the library is that it can also work with large volumes of data, as it can bind data to the Document Object Model. This increases code complexity. However, it makes the code more efficient and flexible than other libraries. To create an easier entry for beginners, the developer created “Observable Plot” a higher-level abstraction that allows users to generate D3.js visualizations with less code [35]. An example of a D3.js visualization can be seen in Figure 3.1b on page 12.
- **VICTOR:** is a cluster comparison application, whose primary focus is to compare different clustering results. It focuses on enabling users to better understand the similarities and differences between various solutions. As input, VICTOR requires two clustering solutions, which it then uses to generate six different visualizations, such as heatmaps, network visualizations, or graphs, along with 10 comparison metrics. These visualizations and metrics enable the identification of patterns and structural differences between the clustering outputs. The application is particularly user-friendly as it is available as a web application. Moreover, the code is available publicly on GitHub [36].

3. Related Work

- **XCluSim**: is a visual analytics tool for comparing different clustering results with a focus on bioinformatics data, where complex and high-dimensional data are common. Unlike VICTOR, which focuses on comparing two clustering outputs at a time, XCluSim is designed to handle comparisons among more than two clustering solutions simultaneously. To achieve this, it offers a diverse set of visualization techniques, including dendrograms and similarity graphs, that highlight the relationships between different methods. One of its most unique features is the “enhanced parallel sets view,” a visualization that allows users to trace how individual data points are allocated across different clustering results. This can be seen in Figure 3.1c on page 12. This provides an interesting approach to see overlaps and divergences between different clustering assignments. It also offers the possibility to explore single clustering results further, with heatmaps for instance, but focuses on comparing different clustering solutions, which can be fundamentally different (eg, k-means, hierarchical clustering, and SOM clustering) [28].

3.1.2. Hybrid Approaches

The following hybrid approaches include both visualizations and clustering capabilities:

- **Points2Regions**: is a Python library that focuses on the visualization of categorical marker data, which is commonly encountered in domains like spatial biology. It not only generates the visualizations, but it also performs the clustering required to visualize the points. The visualizations produced are static. It aims to create a simple and accessible application for researchers by providing extensive assistance throughout the clustering pipeline and offering powerful yet understandable visualizations [37].
- **Clustergrammer**: is an interactive visualization tool that is either available as a web-based application or as a download. It focuses on the visualization of high-dimensional data by creating interactive heatmaps as shown in Figure 3.2b on page 17. The tool also emphasizes interactivity, offering the ability for the user to zoom, reorder rows and columns, and click events that provide more detailed information. While it focuses on the heatmap, this single visualization type is highly optimized, allowing for effective exploration of large and complex datasets. It displays all necessary information in a compact view, making it highly effective; however, it can be quite challenging for novice users to understand. The example use case shown in the paper is for cancer research [38].
- **ClustVis**: is a web-based tool that is usable for free. It allows users to upload datasets or use toy datasets, which are then clustered by the tool after the user inputs the desired number of clusters. ClustVis provides a range of preprocessing options, including normalization and dimensionality reduction, to prepare the data for analysis. The output includes a lower-dimensional scatterplot that provides an overview of the clustering structure as well as a heatmap for more detailed inspection, similar to the one found in Clustergrammer. It also offers the option to

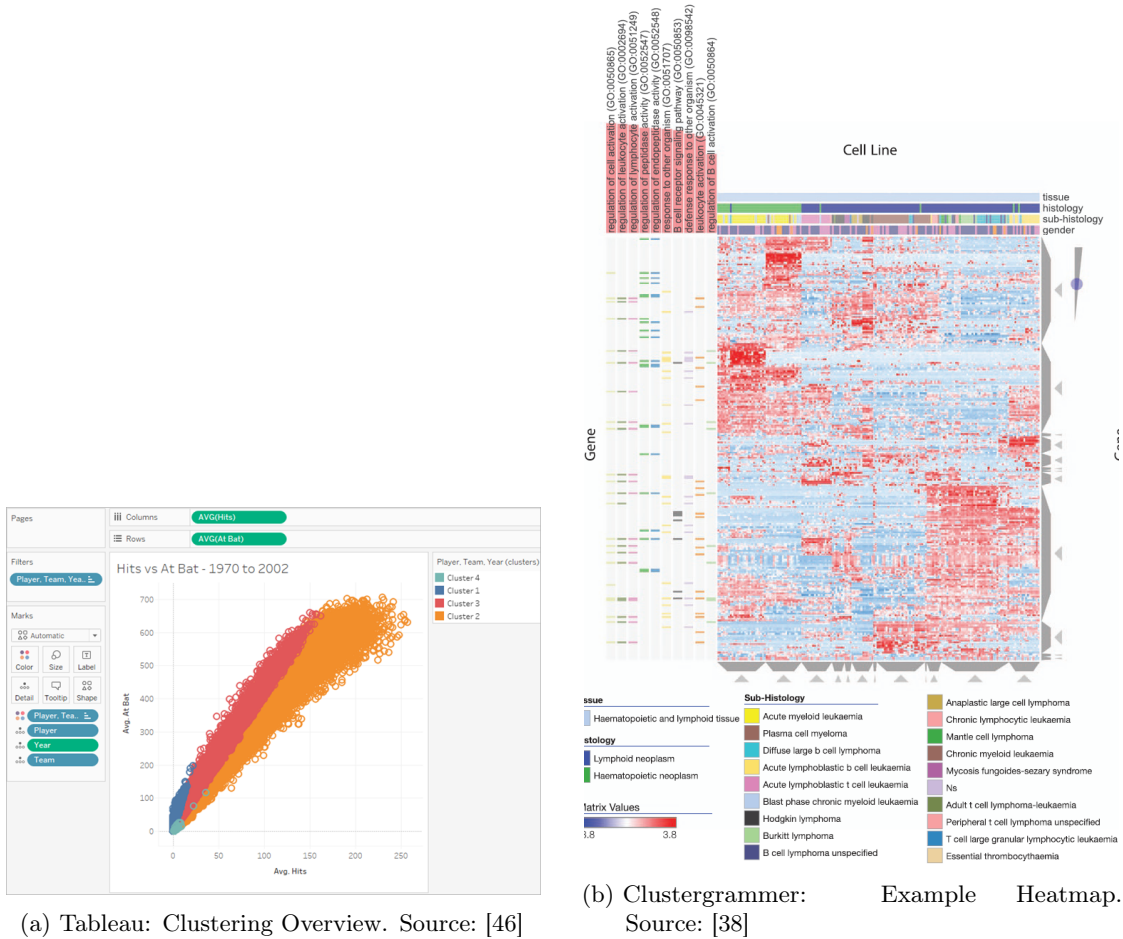
download both the results and intermediate steps. The platform is also designed to be accessible, featuring a tutorial and tooltips that guide users through the process [39].

- **Clustering Visualizer**: is a simple web-based educational tool that showcases the process of clustering algorithms intuitively and interactively. Users can manually place points in an empty two-dimensional space and observe how different algorithms process these inputs step by step. It offers algorithms such as k -means and DBSCAN, and supports setting parameters for all of these algorithms, including k for k -means and min points and epsilon for DBSCAN. This allows users to observe the direct influence of the parameters on the clustering results and process. The strength lies in its educational design, rather than providing real-world applications. It provides a simple environment that makes it easy to grasp the algorithms for all users [40].
- **Orange**: is a data mining tool that focuses on visual programming using workflows, which are composed of modular widgets. Each of these widgets represents a task, such as data import, preprocessing, clustering, or visualization. Users can connect these widgets together to create a customized workflow that fits their needs. For clustering tasks, Orange provides algorithmic modules and visualization components, enabling users to experiment with various methods and parameters quickly. It is available as a freely downloadable application for both Windows and macOS, making it widely accessible. The combination of modular design, interactivity, and integration across the data analysis pipeline makes Orange an interesting application. An example of a workflow is shown in Figure 3.2c on page 17 [41].
- **VizTS**: is an interactive R Shiny application that focuses on clustering and visualization of time series data. The tool guides users through a structured four-step workflow: data import, exploratory analysis, clustering, and interpretation of results. One of its special features is the ability to adjust the temporal granularity of the data, allowing users to explore patterns across different time scales. Furthermore, it provides metrics that support decision-making and visualization of results with a wide range of representations, including map plots, dendrograms, and other graphical outputs suited to temporal data. This makes VizTS particularly useful for domains where time series data are central [42].
- **Clustervision**: is an interactive visual analytics tool that aims to support users in the clustering process by combining automated analysis with visual exploration. The tool automatically generates multiple clustering results and presents them in an overview, allowing users to compare different results. It provides interactive visualization tools, as well as multiple metrics, to users. Among its most notable features are the exploration of feature importance and analyzing cluster composition, as well as an innovative scatterplot design that aggregates nearby points into larger “superpoints” to reduce visual clutter and highlight patterns at a higher level. This makes Clustervision a powerful assistant for clustering analysis. An example of the scatterplot can be seen in Figure 3.2d on page 17 [43].

3. Related Work

- **Clustrophile 2:** is an interactive visual analytics tool that helps users explore different cluster parameters and features. A central feature of Clustrophile 2 is its “Clustering Tour”, which guides the user through various clusterings and iteratively helps them find a matching solution for their use case. The system offers a wide range of different interactive visualizations and multiple metrics in order to help the users. Furthermore, a lot of different parameters can be selected and adjusted. The application strikes a balance between automation and user control, offering an interesting iterative approach to clustering tools [44].
- **Tableau:** is a commercial data visualization and business intelligence platform that includes clustering functionalities alongside its extensive range of visualization tools. It supports the creation of common visualizations that are used in clustering, such as scatterplots and heatmaps. It provides the user with the ability to perform basic clustering, such as using k-means. However, its clustering features are relatively limited in comparison to other tools. Instead, the strength of Tableau lies in its integration with other analytics and visualization environments, where clustering is just a step in many others. It focuses on accessibility and is well-suited for business contexts. An example of this can be seen in Figure 3.2a on page 17 [45, 46].

3.1. Existing Approaches



(c) Orange: Example of workflow. Source: [47]

(d) Clustervision: Main screen. Source: [43]

Figure 3.2.: Overview of different hybrid visualization approaches.

3. Related Work

3.2. Analysis of Existing Tools

This analysis of clustering visualization tools reveals that numerous visualization approaches are available, each with its own strengths and limitations. Many visualization libraries could be integrated into the SHIP framework, especially for the visualization approaches to power the SHIP.AHOI application. On the other hand, the hybrid approaches reveal interesting approaches that are further explored and shape the development of SHIP.AHOI.

Regarding visualization approaches, solutions like D3.js and ECharts stand out, as they offer high flexibility and handle large datasets, both of which align with the application's requirements and offer powerful interactive capabilities and flexible visualizations.

Tools such as Plotly or Matplotlib provide flexibility and an easy-to-use setup. However, Plotly, despite its user-friendly interface and simple implementation approach, lacks the fine-grained control over interactive elements that advanced clustering analysis demands. Similarly, Matplotlib focuses on static visualizations with only basic interactive functionalities such as zooming and panning, compared to frameworks like D3.js and ECharts.

Comparative analysis tools like Victor and XCluSim introduce interesting methods for comparing clustering approaches. Their extensive use of different visualizations and metrics is appealing, and insights from these implementations could enhance the visualizations in the SHIP.AHOI tool. Especially XCluSim's ability to trace individual data points across different visualizations and across different clustering solutions is interesting.

Streamlit, on the other hand, demonstrates a particular value in the prototyping phase, as it offers rapid development and diverse capabilities with minimal coding required. This enables quick iteration and testing, making it ideal for design studies. This rapid prototyping capability can help to get more extensive feedback during the early design phases.

The hybrid approaches also reveal rich insights into the landscape of existing clustering tools. Domain-specific solutions, such as Points2Regions and VizTS, demonstrate the power of specialized tools in addressing particular data types and requirements. As VizTS mainly focuses on time-series data, and Points2Regions on categorical marker data in spatial biology.

Web-based solutions such as ClustVis, Clustergrammer, and Clustering Visualizer represent an interesting category of accessible tools. ClustVis combines various preprocessing options with intuitive visualization outputs, making complex clustering more accessible and comprehensible. Clustergrammer optimizes its single visualization form, an interactive heatmap, which can lead to potent analytical capabilities. This approach can also be interpreted as a form of minimalism, as it focuses on a single, yet complex, visualization. Clustering Visualizer stands out as an accessible and easy-to-understand implementation, effectively explaining clustering to novice users and bridging the gap between theoretical

understanding and practical application. This educational focus could be adapted to make the SHIP framework better understandable, especially to inexperienced users.

Advanced systems, such as Clustervision and Clusterphile 2, offer comprehensive analytical capabilities that demonstrate the potential of integrated clustering environments. Clustervisions' generation of multiple solutions simultaneously provides an interesting approach. Clusterphile 2's "Clustering Tour" approach represents a significant advancement in user guidance and onboarding, making it accessible to a diverse range of users.

Approaches like Tableau and Orange illustrate different philosophies in the design of clustering tools. Orange's visual programming approach and its modular widgets demonstrate how to make complex workflows accessible to users and easily adjustable. Further steps enhance the clarity of the clustering workflow for users. Tableau's integration of clustering within a broader business intelligence context shows the importance of considering clustering as part of a larger system rather than an isolated task.

3.3. Challenges

This analysis of existing clustering visualization tools reveals several critical challenges that the SHIP.AHOI framework can solve and improve. Most significantly, there are no existing solutions that provide specialized visualizations for ultrametric trees and utilize their benefits, like in the SHIP framework. In addition, most tools, if they offer clustering options at all, provide far fewer options and parameter selections compared to the SHIP framework. This limitation is problematic for users who need to explore diverse clustering solutions and parameter spaces to identify optimal solutions for complex datasets. Furthermore, most tools do not focus on real-time exploration and are primarily concerned with examining a few clustering solutions in great detail. Moreover, in many approaches, users are forced to work with static results rather than iterative and interactive results. Educational support is another critical gap, as the majority of tools assume significant prior knowledge of clustering concepts and provide minimal guidance for understanding the algorithms or the tool itself. This creates a barrier for less experienced users.

3.4. Opportunities

The chapter reveals numerous opportunities for innovation and the design of new tools. One interesting finding for improving SHIP.AHOI is that, as seen in Clustergrammer, an interactive dendrogram offers lots of possibilities for interaction and metric integration. Tools like Clustervision further highlight how metrics can be made easily accessible to the user. Additionally, a combination of different visualizations, as seen in some explored multi-view approaches, provides an interesting way to display a large amount of data simultaneously. This multi-view approach demonstrates how various aspects of clustering results can be analyzed simultaneously in different visualizations, enabling

3. Related Work

users to identify patterns that might be hidden in single-view approaches. However, approaches like Clustergrammer’s single heatmap also offers a design that focuses on a single visualization. This focused approach demonstrates that the depth of interaction within a single, well-designed visualization can sometimes provide more value than focusing on many poorly connected visualizations. This suggests that a balance needs to be struck between multiple visualizations and minimalism.

Furthermore, tools like Orange demonstrate that breaking down the workflow into steps helps inexperienced users to better understand the application. Additionally, the graduated complexity, as seen in onboarding tours or step-by-step guides, such as those in the Clustering Visualizer, offers promising possibilities to enhance the usability of the application, particularly for the target group of inexperienced users.

4. Methodology: Design and Evaluation Studies

Designing a practical and effective visualization for a complex domain, such as clustering, can be a challenging task. It requires integrating domain expertise, user needs, and visualization expertise within a single project. For this reason, this thesis adopts the Design Study Methodology as described by Sedlmair et al. [4].

A design study is a structured process that uses a user-centered approach to create a design. It relies on iterative prototyping, continuous feedback, and integration of domain knowledge, ensuring that the developed visualization is not only technically correct but also practical and helpful for its target users. This approach prevents the design from becoming disconnected from users and real-world workflows, thereby maintaining a focus on solving real issues. The design study provides a framework that implements this process in a structured way.

The user study methodology also enables the direct integration of domain knowledge into the requirements, ensuring that creativity in the design process is balanced with fast and iterative validation. In addition, the methodology provides the possibility to gain insights into user needs by generalizing the results, thereby improving future design processes.

The following chapters describe this approach and how it helps solve the complexity of creating a visualization design for the clustering domain.

4.1. Design Study Methodology

The Design Study Methodology consists of a 9-step process, which can be further categorized into three top-level categories. The top-level categories are: Precondition, Core, and Analysis. A visualization of the steps and categories can be found in Figure 4.1 on page 22.

9 Step Design Study Methodology

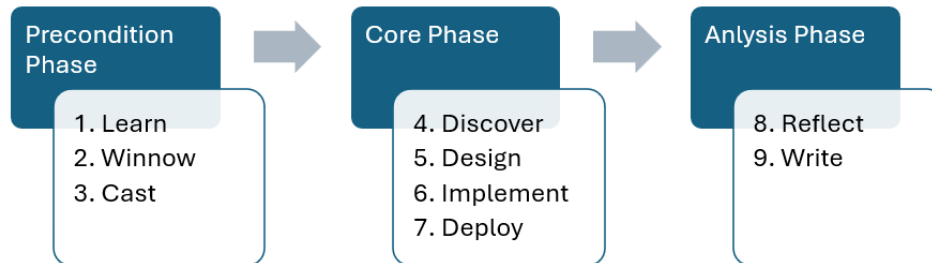


Figure 4.1.: Steps of the design study visualized [4]

4.1.1. Precondition Phase

This phase prepares the developer for the upcoming steps by identifying collaboration partners for the core phase and developing the necessary visualization knowledge.

- **Learn:** Gain knowledge in the visualization domain, which is required to execute the other steps and to design an effective visualization.
- **Winnow:** In this phase, it is important to identify collaborators and talk to them as early as possible. This could be, for example, domain experts who provide further knowledge for the rest of the design study. Also, candidates are first screened for the interviews in the following steps.
- **Cast:** This step is about assigning roles to the collaborators that are found. For example, a “Gatekeeper” is a person who must approve the project, or a “Front-line analyst” is a person who is going to use the developed solution.

4.1.2. Core Phase

This is, as the name suggests, the core phase where the main parts of the requirement analysis as well as the design and implementation of the project, are done.

- **Discover:** A crucial aspect in a design study is to correctly identify the needs, problems, and requirements of the users. Here, as mentioned in the paper, it is important not to focus only on problems but also to identify successful parts of their work as well. Therefore, it is essential to conduct interviews with experts and users to gather requirements. It is also possible to observe people in real-world situations to gain insight into their workflows.

- **Design:** This part involves the generation of different visualization solutions. It is essential to consider a wide range of options here, narrow them down, and discuss ideas and solutions with users. These can be visualized as low-fidelity prototypes (concepts with limited details) and high-fidelity prototypes (more detailed and similar to the final solution). As this process is iterative, it is crucial to incorporate feedback from users/domain experts in each iteration and adjust the prototypes accordingly.
- **Implement:** This step involves developing the final solution using Code. However, it is essential to remain agile in the development, as some ideas may need to be discarded during the implementation. It is a common pitfall to skip feedback at this stage, but user feedback is also vital in this step.
- **Deploy:** This stage involves deploying the tool and gathering feedback on it. Here, it is essential to validate whether the tool helps domain experts.

4.1.3. Analysis Phase

This is the last phase, which involves reflecting on the entire process and writing a paper.

- **Reflect:** In this stage, the project moves from engineering to research, as the researcher analyzes the project and improves existing design guidelines based on the findings.
- **Write** - This stage is about writing a design paper on the process, where the authors note that enough time should be planned for this step.

4.1.4. Adaption

For this design study, four rounds of interviews with a minimum of six participants will be conducted, as well as an additional user study for evaluation. The first round of interviews is designed to gather the requirements from each user, understand their current workflow, and identify the problems they currently face. Then, two rounds are planned in which users provide feedback on iteratively designed and improved low-fidelity prototypes. These prototypes, which are mocked up using different design elements, aim to give a general idea of how the application could look.

After that, in the final round, an advanced state of the implementation is presented to users to give final feedback on the design and small details that could improve the application. In these steps, some parts of the clustering process already work as a showcase, since this is important to find problems in the development early and present the users with working implementations that already solved implementation issues, such as performance issues.

4.2. User Study

In line with Sedlmair et al. [4], the evaluation is set in the Deploy stage. The goal is to find out if the deployed system really helps the targeted users perform their Tasks “better” in some form, or changes the way experts see problems.

For this master thesis, an approach is chosen where part of the testers are selected from the design study and some new participants are recruited, also to test, how people without knowledge of the design study perceive the application and to remove any bias that users might like it simply because some of their ideas from the design study were implemented, or that it was over-engineered for these aspects. The user study consists of Tasks that represent the typical workflow of a clustering expert and should enable users to test the key elements of the application. During each Task, users are frequently asked to verbalize their thoughts. After each task, they should answer short reflective questions regarding task difficulty, the usefulness of features, and their experience. At the end of the study, participants provide qualitative feedback through a series of interview questions that focused on general usability.

In addition to these qualitative insights, the user study also collects quantitative performance measurements, such as task completion time and a standardized usability score. For the latter, the System Usability Scale (SUS) questionnaire was used, in which participants rated ten usability statements on a five-point Likert scale [48].

5. System Design and Prototyping

This chapter presents the findings from the conducted design study, including the interviews. Furthermore, design decisions necessary for the design process are explained.

First, it is important to note which data is available from the SHIP framework to understand the possible visualization options based on the data:

- **Tree:** The hierarchical tree from the SHIP framework. It is generated from the chosen ultrametric and encodes the hierarchical relationships between all data points. This tree can then be efficiently reused to derive clusterings under different power parameters or partitioning strategies. This additional hierarchical information could be used to further visualize the clustering process.
- **Data Points:** Alongside the tree, the framework has the individual data points with their feature values and cluster labels available. This enables a wide range of possible visualizations: points could be plotted in reduced dimensions, colored according to their cluster assignment, or linked interactively with the hierarchical representation. Having access to both the feature values and the assigned clusters allows to visualize detailed distribution of features within and across clusters.

There are no specific visualization requirements from my supervisor. However, we expected that the data points should be visualized in some form. It was also required to explore whether people want the tree to be visualized and to determine, based on feedback, which visualization should be used as part of the design study.

5.1. Requirements Analysis

The most important findings of the precondition phase of the design study are the selected participants and their assigned roles. Additional potential users were contacted. However, some did not respond, and others did not meet the selection criteria. The main requirement was that participants had at least a basic understanding of clustering, for example, by having taken a university course in which clustering was covered as a topic.

5.1.1. Interview Participant Roles

In Table 5.1 on page 26, you can see the selected participants for the design study. The participants include experts who bring experience in the clustering domain, some of

5. System Design and Prototyping

them are also familiar with the SHIP framework. Other participants are users with less experience, who can provide valuable input on their use cases and perspectives on explainability.

ID	Role	Background / Expertise	Role
U1	Post-Doc, linked to SHIP paper	Development of clustering algorithms, supervision of students	Domain Expert, Gate-keeper
U2	Researcher, linked to SHIP paper	Theoretical expertise in clustering methods	Domain Expert, User
U3	PhD Candidate, linked to SHIP paper	Development and application of clustering algorithms	Domain Expert, User
U4	Researcher (Applied Science)	Use of clustering methods in a astronomy domain	User
U5	Student	Basic knowledge in clustering from courses, frontend development experience	User
U6	Student	Clustering courses, software engineering background	User
U7	Student	Clustering courses, practical university experience	User

Table 5.1.: Overview of design study participants

5.1.2. Identified Problems and Challenges

Each of the requirement interviews with the selected participants, shown in Table 5.1, took about 45 minutes. This allowed an in-depth understanding of their most important issues and the problems they are facing. Of course, the well-functioning parts of their current workflows should still be preserved in the design. However, detailed problems were identified to address the pain points they currently encounter. These problems can be seen in Table 5.2 on page 27.

The main problems identified during the interviews were that all participants found it challenging to select the correct parameters for their clustering. However, it remains to be seen whether this issue can be fully addressed in this master’s thesis, as it is a more fundamental clustering problem. Each dataset requires its own set of parameters, and there is often no single correct solution. Furthermore, some central issues are deeply related to visualization (P2, P3, P5, P6, P10, P11) and require special attention in the design study. Other problems, such as P4, P8, P9, and P12, are more focused on clustering algorithms. These can be partly addressed with the SHIP framework. Still, they are challenging to resolve fully, and most of them are “backend” issues that are not affected by the visualization, design, or implementation of SHIP.AHOI. Additionally, the requirement interviews revealed that each user has a diverse set of individual requirements and use

cases. However, many requirements overlap and provide guidance for the implementation.

ID	Problem Description	Involved Participants
P1	Choosing appropriate parameters for clustering algorithms (e.g., k in k-means, eps/minPts in DBSCAN, distance metric) is unintuitive and requires experience or trial and error.	All
P2	Evaluating clustering quality is difficult, especially without ground truth. Visualization helps but is subjective; metrics exist but are often hard to interpret.	U1, U2, U3, U7, U5
P3	Comparing results from different algorithms or parameter settings is challenging.	U4, U2, U5, U7
P4	Scalability is problematic, as many algorithms do not scale well to large datasets or high-dimensional data.	U1, U5, U7
P5	Metrics alone are not sufficient; visualization is also important. Some participants found metrics adequate, but others relied on visuals.	U1, U2, U3
P6	Handling high-dimensional data is challenging; visualization and clustering are difficult, and dimensionality reduction may cause information loss.	U2, U6, U1
P7	Understanding algorithm behavior and results is difficult (e.g., why points cluster together, impact of parameter changes).	U2, U1, U3
P8	Handling noise and outliers can strongly affect clustering results.	U4, U1, U2
P9	Reproducibility is important; making results understandable and reproducible can be difficult.	U1
P10	Overloaded interfaces and poor usability were frequently reported as problems to avoid in the tool.	U2, U5, U6, U7
P11	Displaying large hierarchical structures (e.g., trees) is difficult and can become cluttered with big datasets.	U2, U1, U3, U5, U7
P12	Data not being normalized or requiring preprocessing are common issues.	U4, U1, U5
P13	Understanding how methods work in tools and which ones to choose.	U5, U6, U7

Table 5.2.: Problems reported by participants

5.1.3. Derived Requirements and Tasks

Following the interviews and the formulation of the observed problems, tasks were defined that should be solved in the final design of the application. Furthermore, each task was assigned a priority, indicating its importance to the users. Another important aspect

5. System Design and Prototyping

is that some Tasks were also defined by the Gatekeeper, who expects these tasks to be implemented in the final version. In addition, some optional features were determined. These are considered nice to have, but it still needs to be explored whether participants truly want these features or would prefer alternative approaches.

Task ID	Task Description	Priority	Linked Problems
	Core Functionality		
K1	Load and inspect data from common formats, optionally provide an overview of features/statistics.	High	Gatekeeper
K2	Interactive parameter setting for: Ultrametric construction, hierarchical clustering, partitioning methods.	High	Gatekeeper
K3	Visualize resulting clusters (2D, optional 3D).	High	P2, P3, P5, P6
K4	Enable fast switching and exploration.	High	Gatekeeper
	Data handling		
D1	Save and load configurations for reproducibility.	Medium	P9
D2	Export final clustering results, parameters, and visualizations.	High	P9
	Visualization		
V1	Interactive parameter adjustment with immediate feedback on clustering impact.	High	P1, P7
V2	Evaluate clustering quality visually and via metrics (e.g., Elbow-Plot, ARI).	Medium	P2, P5
V3	Compare different clustering solutions side by side or by fast switching.	Low	P3
V4	Provide guidance/tooltips on algorithms and parameters.	Medium	P1, P7, P10, P13
V5	Suggest parameter ranges and optionally best values.	Medium	P1, P13
	Optional + Advanced Features		
O1	Visualize hierarchy and interact (zoom, select subtrees, highlight points).	Medium	P11
O2	Analyze feature influence on clusters.	Low	P7
O3	Handle noise/outliers, e.g., mark leaf nodes as noise or integrate noise-cleaning methods.	Low	P8
O4	Provide preprocessing options like normalization/scaling.	Low	P12

Table 5.3.: User study Tasks mapped to linked problems

5.2. Design of the Interactive Clustering Website

5.2.1. Low-Fidelity Prototypes: Description and Evaluation

The process of designing the first prototypes can be challenging, as many aspects and requirements have to be taken into account. Furthermore, it is essential to create distinctly different versions, which give the user a more abstract idea of how fundamentally different the visualizations and flows for the application could be, and then work from there to create a solution that fits the needs of all users.

First Round Low-Fidelity Prototypes

Some important factors that are considered in the first iteration of the low-fidelity prototypes are general visualization aspects. However, the requirements interviews also play a big role in shaping these aspects:

- **Simplicity:** In the requirement analysis, it was often mentioned by participants that it is important for them that the developed solution is simple yet powerful and not overloaded with features and options like some existing solutions. Therefore, it is essential to find out which features and data are necessary for users.
- **Page Design:** A critical aspect in the application is the flow in which the pages should be visited and the general layout. The primary objective for the first iteration was to determine whether people prefer a linear setup for parameter configuration, where users transition from one page to another with information tailored to each selection. Alternatively, it was to determine if people prefer a single central page from which more detailed information about specific aspects can be accessed.
- **Visualization Methods:** The question of whether people want to see the hierarchical tree is also a crucial question for the first iteration, and if yes, to maybe get an idea of which visualization they would like. Also, for other visualizations, it is vital to test if people want a scatterplot and or other visualization forms.

The low-fidelity prototypes were designed using the open-source and free software Penpot, in which different designs were created. For many elements, the Streamlit framework was utilized to provide basic components, including menus, scatterplots, and menu bars, ensuring a consistent minimalist design.

The first five prototypes used in the initial iteration are as follows, each with a nickname assigned to distinguish them better. The feedback that the participants of the design study provided are stated directly after each prototype, and the key takeaways are summed up afterwards.

1. **Multiple Page Prototype:** This prototype consists of 5 pages, one for each step of the SHIP algorithm and two additional ones. The first page is the Data Upload page, where the user can upload data or use sample datasets. The user is guided through the pages in order, but is also free to use a navbar to switch. On the second

5. System Design and Prototyping

page, the user selects a similarity metric and is presented with a graph view of it. On the third page, the power parameter is selected, and the different trees for each power are displayed. On the fourth page, a partition method is selected, and support graphs like Elbow plots are provided based on the chosen parameter. On the last page, the results are shown in a scatterplot along with the metrics.

Feedback: Participants felt that the design is too linear and hinders exploration on many different pages. However, the page-by-page structure was seen as easy to follow. It was praised for clarity, but seen as not practical, especially for iterative work.

2. **Mode Prototype:** This prototype features two modes, which can be selected in the sidebar: the “Novice” mode and the “Expert” mode. In Novice mode, users answer some questions that determine the clustering parameters, and the resulting clusters are then displayed in a scatterplot. The advanced “Expert” mode offers a basic sidebar for file upload and mode selection. On the main page for the “Expert” mode, the parameters can be selected at the top of the page, and a scatterplot is displayed. From the navigation bar, some more in-depth pages are accessible, which offer visualizations and statistics on specific topics. A visualization comparison mode allows you to adjust the scatterplot axes to highlight selected features. Another page is a Hierarchy page, where the hierarchy is visualized as a graph from top to bottom. Then, a Data Overview page displays basic statistics, including the mean, count, and standard deviation of features, as well as a correlation heatmap. Another page shows the feature importance that distinguishes between clusters and indicates which ones are similar within a cluster. The last page is a “Drill-Down” page, where you can select a resulting cluster and split it further.

Feedback: Participants considered this prototype the most successful overall. They appreciated the accessibility of the guided *Novice Mode* while valuing the flexibility of the *Expert Mode*. However, they noted that the Novice Mode was too simplistic and should be expanded to provide more meaningful guidance. Furthermore, participants liked the wide range of different supporting pages available in this prototype.

3. **Side by Side Prototype:** This prototype is more minimalist and features a single page, where there is a file upload or toy dataset selection at the top of the page. The page is divided in the middle, with all options mirrored on both sides — including the visualizations. However, you can make independent selections on each side, which changes the visualizations for each. There are options to change the tree parameters (tree, power, partition method) of the SHIP framework. Below is a radial graph that displays the hierarchy. Below that is a scatterplot, and at the bottom are the metrics.

Feedback: User appreciated the powerful visual comparison and described it as highly useful for evaluating alternative settings. However, the users felt that the prototype had an overwhelming interface and too much focus on this single use case. Many suggested that the design would be a good page embedded into another

prototype.

4. **Sidebar Prototype:** This prototype consists of one page, with a detailed sidebar. In the sidebar, you can upload files, see a preview of the data, select all the SHIP parameters, and choose the dimensionality of the scatterplot and dimensionality reduction method. On the page itself, there is a dendrogram-inspired graph. Below that is the scatterplot next to a partition method support graph (for example, an Elbow plot). At the bottom, you can expand a visualization for the feature importance and the metrics. For this prototype, some different forms of tree visualizations were also shown to the user (Graph, Graph left to right, Sunburst, Treemap).

Feedback: The prototype was praised for its efficient use of space and continuous visibility of controls. Participants felt that it offered numerous interactive visualizations that aid the workflow. Furthermore, they mentioned that they prefer the tree visualization to be there and choose either a dendrogram or a sunburst diagram as a tree visualization. Users further noted that the sidebar could become overloaded and should focus on essential operations and being well-organized.

5. **Minimalism Prototype:** This single-page prototype features a data upload element at the top of the page, with the SHIP parameter configuration arranged vertically below it. Below that is a scatterplot and the metrics. At the bottom, you can also expand a button to show more cluster details (mean values, count of features). This prototype, as its name suggests, focuses primarily on providing the most necessary elements and gathering feedback from users.

Feedback: Participants were split in their opinions. Some valued the clarity and minimalism, praising it for being easy to understand. However, most felt it was too stripped-down and missing key features, such as the hierarchy visualizations and supporting plots.

• Key Takeaways

- The “Sidebar Prototype” and “Mode Prototype” stood out as clear favorites.
- Tree visualizations were strongly preferred and seen as essential.
- Interactivity was emphasized as a key requirement across prototypes.
- Participants preferred subpages over long scrolling for navigation.
- Users wanted some form of beginner mode to aid inexperienced users, but in-depth information for experienced users.
- Participants preferred having the parameter selection placed easily available, for example, in a sidebar.

Second Round Low-Fidelity Prototypes

For the second low-fidelity phase, four different designs were created. A key aspect in their design was to focus on what the previous “Mode Prototype” and “Sidebar Prototype” did well, improve on that, and also try to combine them to some extent. Additionally, some new ideas were introduced for participants to give feedback on. The feedback is again after the prototype description and the takeaways at the end.

1. **Multi-Page Prototype:** This prototype focuses on implementing a sidebar with many options. In the sidebar, you can upload files, change parameters, set visualization options, and also export or save the data. The main page is strongly inspired by the “Sidebar Prototype”, which offers a compact view with both a tree visualization and a scatterplot. Furthermore, two additional pages were added to show more details, accessible via a navigation bar at the top of the page. The first page provides an overview of the dataset, including statistics such as mean values and some example entries. The second page focuses on clustering, where different scores are displayed, along with a heatmap of feature correlations, the size of the clusters, and additional statistics for each cluster, including mean values for the features.

Feedback: The “Multi Page Prototype” failed to impress the users, as it was received as lacking information and useful additional pages. However, the clean layout on the main page was praised. Participants mentioned that they prefer data visualization over tables or plain text and find the in-depth pages of this prototype to be lacking and not supportive of their workflows.

2. **No Mode Comparison Prototype:** This prototype features the same sidebar as the previous one, but without the “Novice” and “expert” modes. The main page features a tree visualization, a scatter plot, and a summary of the metrics. As before, the prototype features a navigation menu at the top and several advanced pages. The first additional page is a Data Overview, similar to the one implemented in the “Mode Prototype”, but now also including visualizations for feature distributions and missing values. The Clustering Performance page is also equipped with a wide range of visualizations, including bar charts for cluster size and cohesion quality. Mean values are also displayed in a dedicated visualization. The “Side by Side Prototype” now exists as its own page, and an additional page has been introduced to display supporting visualizations for different partition methods, for example, an elbow plot for the elbow method. These visualizations also highlight the recommended value for each partition method.

Feedback: Some of the users favored the Prototype, and people liked the different visualizations across all the provided pages, as it made understanding the data and clustering easier for them. The sidebar was also perceived as useful once again. Users liked the scatterplot. However, they mentioned that they would like more interactive possibilities with it, like tooltips. Furthermore, people suggested that they would like a dedicated page for the exploration of partition numbers.

3. **Sunburst Prototype:** This prototype focuses on having a simple sidebar where you can upload the dataset, choose different pre-processing options, select visualization options, and load previous runs from a drop-down menu. It also reintroduces the mode selection, now renamed to “Advanced mode” and “Basic mode” instead of “Expert mode” and “Novice mode”, as suggested by some users in the last interviews. In “Advanced mode,” the parameters for clustering are placed at the top of the page. Below them is a large sunburst diagram where you can hover over segments, which then highlight according to the points in the scatterplot underneath. Below the scatterplot, there is a supporting visualization for partition choice and metrics. “Basic mode” is similar but has no additional pages and removes history and configurations from the sidebar. The other pages are accessible again via the top menu. One page for cluster analysis with statistics and a heatmap, and another page focused on showing the performance and speed of completed runs. This is displayed using bar charts and tables, which also highlight the scores of each run for easier comparison.

Feedback: This prototype had some interesting feedback, as the users gave mixed and negative feedback on the sunburst diagram in contrast to the last round. They felt that it was unifying and ineffective in showing the hierarchy compared to other options. The user modes in this prototype, however, were seen as valuable and positive. Furthermore, people liked the possibility of comparing different runs with each other.

4. **History Prototype:** This prototype features the same sidebar with mode selection as the “Sunburst Prototype”. The only difference is that the drop-down for the run history has been replaced small visualization for each of the previous clustering runs. This should help the user organize previous runs more easily and provide a better UI for selecting them. The “basic mode” again features questions to find the ideal clustering, and the output is a scatterplot along with an elbow plot that explains why a particular partition was chosen. In “Advanced mode”, the main page again shows the parameters at the top, followed by a scatterplot and a tree visualization, with the metrics displayed at the bottom. The other pages, accessible via the top menu, include a Visualization Comparison page, which allows manual axis selection for scatterplots, as well as a Data Overview page showing a correlation heatmap between features and general dataset statistics. A “Drill Down” page is also available, where a single cluster can be further analyzed. A new addition is the Clustering page, which shows previous runs with their scores as well as a small scatterplot. From this page, runs can also be reloaded or deleted from the history.

Feedback: The prototype was seen as the favorite of some users, as they liked the modes and especially the history feature. The history feature was seen as useful, and the visualization was seen as good. However, it lacked key statistics, such as cluster amount. The drill-down menu received mixed feedback, as people liked it but did not really find it helpful. Some areas for improvement were mentioned, such as the history page not showing enough information to compare runs effectively.

5. System Design and Prototyping

• Key Takeaways

- People liked the interactions with visualizations like the tree and dendrogram and appreciated features such as hovering.
- People liked the history feature as a core element of the application, and most also liked the “Beginner” and “Advanced” mode separation to help new users.
- Interviewees liked having visualization options for the tree (e.g., keeping sunburst as an option) but not as the default visualization.
- Focus should be placed on not fragmenting the pages too much and on ensuring that the additional pages are truly useful, as well as improving the placement of controls in some prototypes.
- Runtime and performance statistics were seen as non-essential features by nearly all users.

5.2.2. High-Fidelity Prototype

The feedback from the second round highly influenced the design of the implementation. The following important design decisions were made after the first two rounds of interviews:

- **Advanced and Basic Mode:** Based on the feedback received in the first two rounds of interviews, which indicated that these modes would be a nice addition, the idea was slightly reworked. A better way to onboard new users, while still offering them the same set of features as advanced users without confusing them, was to create a proper onboarding process and a starting page. The Starting page should provide information about the SHIP algorithm and explain how the website works. The onboarding, on the other hand, should guide new users through their first clustering run by helping them pick parameters based on questions, as in the prototypes. Tooltips should also assist users of all backgrounds to better understand metrics and parameter selection on the pages.
- **Visualizations:** Regarding the scatterplot, it was discussed whether the scatterplot on the main page of the clustering should offer, in addition to the 2D visualization, a 3D scatterplot. However, most users found it unnecessary, and furthermore Sedlmair et al. [12] states that it rarely helps users and often leads to worse results, so it should be avoided. Therefore, only a 2D visualization is available for the scatterplot. Additionally, interactivity is crucial for the scatterplot, as hovering over a point in the tree highlights the corresponding point in the scatterplot, allowing the user to understand the connection between them better and receive additional information when needed.
- **Detailed pages:** According to the feedback, the Data overview and Cluster overview pages will be implemented with a focus on using visualizations to help users understand the data. Furthermore, a page will be dedicated to partition

5.2. Design of the Interactive Clustering Website

methods and the ideal selection of the number of clusters. The “Drill Down” page will not be implemented as there was not enough need for it.

- **History:** The application will use a history system. It will be possible to load previously used parameters from the sidebar and load runs. However, it is essential to display key information directly, such as parameters and cluster amount, on the History page to make it easy to use and find the correct runs. This History page will feature a comparison tool, enabling users to compare different runs easily. The comparison should use metrics to show which option is the best. Still, it should also offer a scatterplot to visually compare runs, as people found it essential to also have a visualization in the comparison.
- **Main Page:** The main Clustering page will focus on a minimalist design with the sidebar from the “No Mode Prototype,” which features an extensive sidebar with the parameter options. On the page itself, a tree visualization and a scatterplot will be displayed, allowing for maximum space for these visualizations. This enables users to focus on them and have sufficient screen space to view all the details, even on small screens. Customization options will be available to either split the screen horizontally for the visualization or stack them vertically, also allowing for the use of wide-screen monitors if available. Furthermore, there will be an option to switch between different tree visualizations and options.

This decision decisions were implemented, and the resulting high-fidelity prototype was presented to the design decision participants for some feedback.

Summary of Interview Feedback : Overall, in the feedback, participants liked the clean, modern look, the connections between visualizations, and the use of tooltips to assist users. However, some improvements were also mentioned:

- **What worked well:** Interactivity between views; clarity of tooltips; overall look and feel; clear separation into subpages.
- **Where users struggled:** The hierarchy graph became confusing at deeper levels; the main page felt overloaded; metrics on the main page distracted from exploration.
- **Requests and suggestions:** Add the disco metric; provide more and better tooltips; make sidebar groups collapsible and better organized; expand the Starting page with more guidance; persist runs in a database; improve *bidirectional* linking so selecting points in the scatterplot also highlights them in the hierarchy.

Resulting Changes in the Prototype Based on this feedback, the implementation introduced these changes to fix all the problems users and to improve the design further.

- **Bidirectional Linking:** Selecting a point in the scatterplot now highlights the corresponding node in the tree visualization. This integrates into the existing logic,

5. System Design and Prototyping

which highlights points in the scatterplot when hovering over them in the tree visualization.

- **Starting Page and Onboarding:** The Starting Page was completely reworked with more in-depth explanations and interactive examples to encourage exploration.
- **Sidebar Organization:** Sidebar elements are now collapsible and better grouped, while also focusing on easy reachability of basic operations.
- **Tooltips & Terminology:** Tooltips were improved and added throughout the interface, and coloring and consistency were improved.
- **Metrics Placement:** Metrics previously shown on the main page were relocated to the Cluster Explorer page to keep the Clustering page focused on exploration.
- **Additional Metric:** The disco metric was added to provide a density-focused metric.

6. Final Design and Functionalities

This chapter gives a detailed overview of the most important functionalities of the final design of the application. It describes which elements support each functionality and provide screenshots from the implementation.

The implementation used for the screenshots utilized a JavaScript frontend with D3.js to visualize the various visualizations and the web pages. The data is taken from a backend that implements the SHIP framework and various other supportive features. They are connected via an API, and a database is also used to store clustering results. More details can be found in Section A.1.

The provided screenshots and design already implement all the improvements that are discussed in the design study and user study Section 7.2.7.

6.1. Getting Started and Onboarding

The functionality that allows users to get started quickly, along with an optional guided onboarding, is one of the most crucial aspects of the design. It mainly consists of the Starting page, but also involves Data Upload and Question pages.

On the Starting page, the user receives detailed information about the SHIP framework and how the SHIP.AHOI pages work. This is achieved by first explaining the fundamentals of the SHIP framework, which can be seen in Figure 6.1 on page 38. In (1), there is a short summary of the SHIP framework, and in (2), a direct link to the paper for more information. (3) is a visualization showing all steps of the SHIP framework to give users a quick and better understanding of how the framework works. Even if they choose to skip reading the paper and the introductory text, this visualization provides a quick overview.

Below, more detailed information about the benefits of the framework is provided. Furthermore, information is shown in detail about each of the three steps of the SHIP framework. Based on the design study and user study experience, it is essential to break down the complex framework into these steps, as this makes it easier for users to understand. Furthermore, the explanation of each step is aided by visualizations and examples of different parameters. An example for this can be seen in Figure 6.2 on page 39, where (4) shows an explanation for the various tree types and how an RTree works. The goal is that the interactive element encourages users to explore these visualizations, which explain how the parameters work and provide examples, so they are better prepared for the Clustering page.

6. Final Design and Functionalities

1

Welcome to SHIP.AHOI

SHIP.AHOI is an interactive tool for exploratory data analysis, built upon the powerful SHIP (Similarity-Hierarchy-Partition) framework. It allows you to generate and explore a vast landscape of clustering solutions with unprecedented speed. Instead of running one algorithm at a time, SHIP computes a foundational similarity model (an ultrametric tree) of your data. From this single tree, you can nearly instantly derive countless hierarchical clusterings and partitions, making it easy to discover the most meaningful patterns in your dataset.

2

Read the Research Paper

arXiv:2502.14018 - Learn more about the SHIP framework →

3



3. A Partition is extracted to form the final clusters.

Figure 6.1.: Starting page of the application

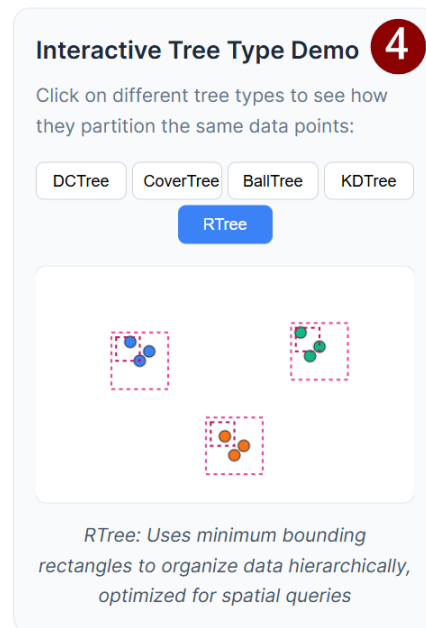


Figure 6.2.: Visualization of Tree Types for the first step of the SHIP framework

How do you want to determine the number of clusters?

This directly sets your partition method parameter.

I know the exact number (K)

I have a specific number of clusters in mind

K: Use when you know your data structure

Find optimal using Elbow method

Automatically find the "elbow" point in clustering quality

Elbow: Use when you want automatic, interpretable results

5

Find most stable clustering

Choose the number that gives most consistent results

Stability: Use when you need robust, reliable clusters

Question 1 of 3

Figure 6.3.: Visualization of the first question in the guided onboarding process

6. Final Design and Functionalities

The next step for the user is to upload a file. However, this is described in more detail in Section 6.2. On the Starting page, the user can further select a guided onboarding option at the bottom, where they receive a series of questions to aid the parameter choice for their clustering after uploading the file. In the following sections, the guided onboarding and Starting page are collectively referred to as the onboarding process. An example of the questions for the guided onboarding can be seen in Figure 6.3 on page 39, where (5) shows the answer to determine the best number of clusters. The goal of this is to help inexperienced users get started with the tool.

6.2. Exploring Dataset

The capability to upload and explore data is available during the onboarding process, where you can upload files and configure them. Data exploration is primarily conducted on the Data Explorer page, where users can obtain information about the dataset.

For the data upload, the user can either select to upload a CSV file or choose from simple datasets like blob or moons. Alternatively, the application includes several real-world datasets, such as the Digits Dataset and the Iris Dataset. For the CSV file upload, users have customization options for using the data. This can be seen in Figure 6.4 on page 41. In (1), the user can see which file is currently uploaded, and in (2), they can see the first five rows of the data. In (3), the user can customize for each feature if it should be used as a feature, be ignored, or if the column is the ground truth. Furthermore, they can normalize features if they want to.

The Data Explorer page shows basic information about the dataset as well as some statistics, such as mean values, min/max values, and missing values. Below, there is a feature distribution visualization. Below that is a correlation heatmap (4), which illustrates the correlation between the features. This can be seen in Figure 6.5 on page 42. Under the heatmap, the page shows the first rows of the data. In the sidebar, the user can select which features they want to see for the statistics displayed on the page. This is especially important for high-dimensional datasets, where there are too many dimensions to visualize the entire correlation matrix at once.

6.2. Exploring Dataset

wine-clustering.csv

10.83 KB

178 rows 13 columns

First row contains headers

Data Preview (First 5 rows)

ALCOHOL	MALIC_ACID	ASH	ASH_ALCANITY	MAGNESIUM	TOTAL_PHENOLS	FLAVANOIDS	NONFLAVANOIDS
14,23	1,71	2,43	15,6	127	2,8	3,06	0,28
13,2	1,78	2,14	11,2	100	2,65	2,76	0,26
13,16	2,36	2,67	18,6	101	2,8	3,24	0,3
14,37	1,95	2,5	16,8	113	3,85	3,49	0,24
13,24	2,59	2,87	21	118	2,8	2,69	0,39

Configure Columns

Use Dataset as is →

3

Configure Your Data Columns

Select which columns to use as features and how to handle them

COLUMN NAME	SAMPLE VALUES	DATA TYPE	MISSING VALUES	USAGE	NORMALIZE
Alcohol	14.23, 13.2, 13.16	Number	✓	Ignore	☐ Normalize
Malic_Acid	1.71, 1.78, 2.36	Number	✓	Feature	☒ Normalize
Ash	2.43, 2.14, 2.67	Number	✓	Feature	☒ Normalize

Figure 6.4.: CSV upload configuration options

6. Final Design and Functionalities

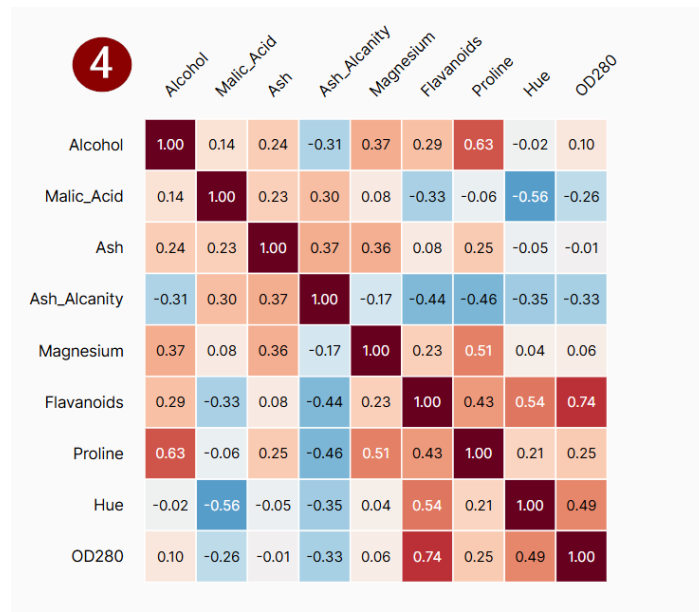


Figure 6.5.: Heatmap on the Data Explorer page

6.3. Tuning Parameters

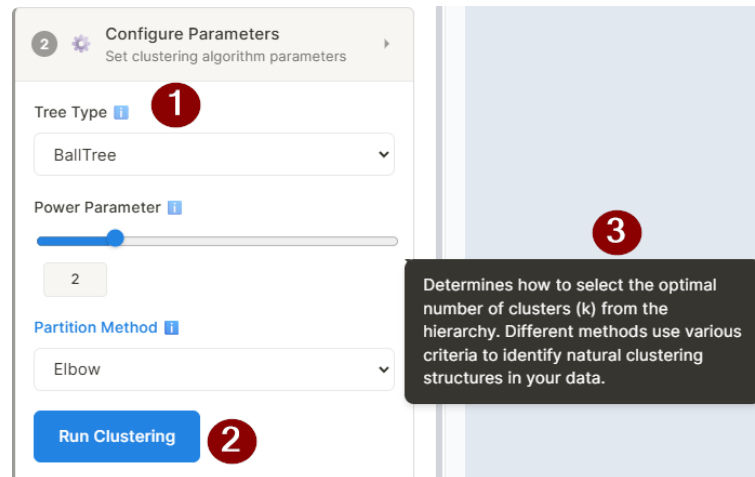


Figure 6.6.: Sidebar parameter options with tooltip

Adjusting the parameters is a crucial step within the SHIP framework. On the Clustering page, this can be done using the sidebar. In addition, the K-selection page provides tools to determine the most suitable number of clusters. This chapter therefore, explains in detail how the sidebar supports clustering and how the K-selection page assists in selecting the optimal number of clusters.

6.3.1. Sidebar

The sidebar on the Clustering page features the option to change the dataset and to configure the parameters for the SHIP framework. The selection for the parameters can be seen in Figure 6.6 on page 42. At (1), you can see the different parameter options to select from, and the “Run Clustering” button can be seen at (2). Furthermore, one of the tooltips for the partition methods can be seen in (3), which serves as a quick reminder for users who are unfamiliar with the parameters.

6.3.2. K-Selection

The k-selection page helps users decide the number of clusters that is best for the dataset if the automatic solution (for example elbow method) is not sufficient. The user on the page manually sets a range of values for the number of clusters he wants to explore, which is done using the sidebar. The page then focuses on finding the best number of clusters in this range. To assist users, the page visualizes, as seen in Figure 6.7 on page 44, at the top of the page (1), the recommended values for various metrics. Below, in (3), the user can view more detailed information about a specific metric and its corresponding value for each number of clusters. The user can adjust which metric to see in (3) by selecting the desired metric in (2).

The lower part of the page can be seen in Figure 6.8 on page 45. In (5), a scatterplot offers a quick view of the clustering using different k values. For this scatterplot, the user can select a value in the drop-down menu in (4) or click into the chart in (3) to view the scatterplot at the selected value. Furthermore, at the bottom of the page (6), the user sees all the metrics for the currently selected number of clusters.

6. Final Design and Functionalities

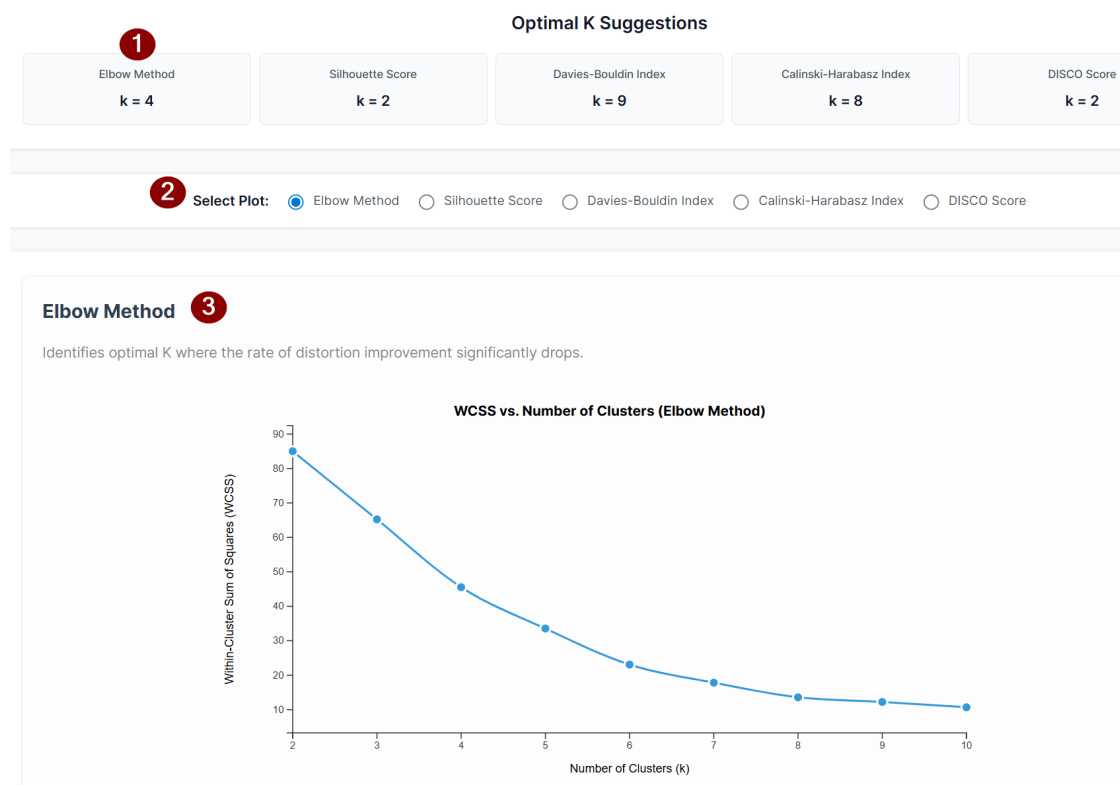


Figure 6.7.: K-selection page optimal k suggestions and metric visualization

6.3. Tuning Parameters

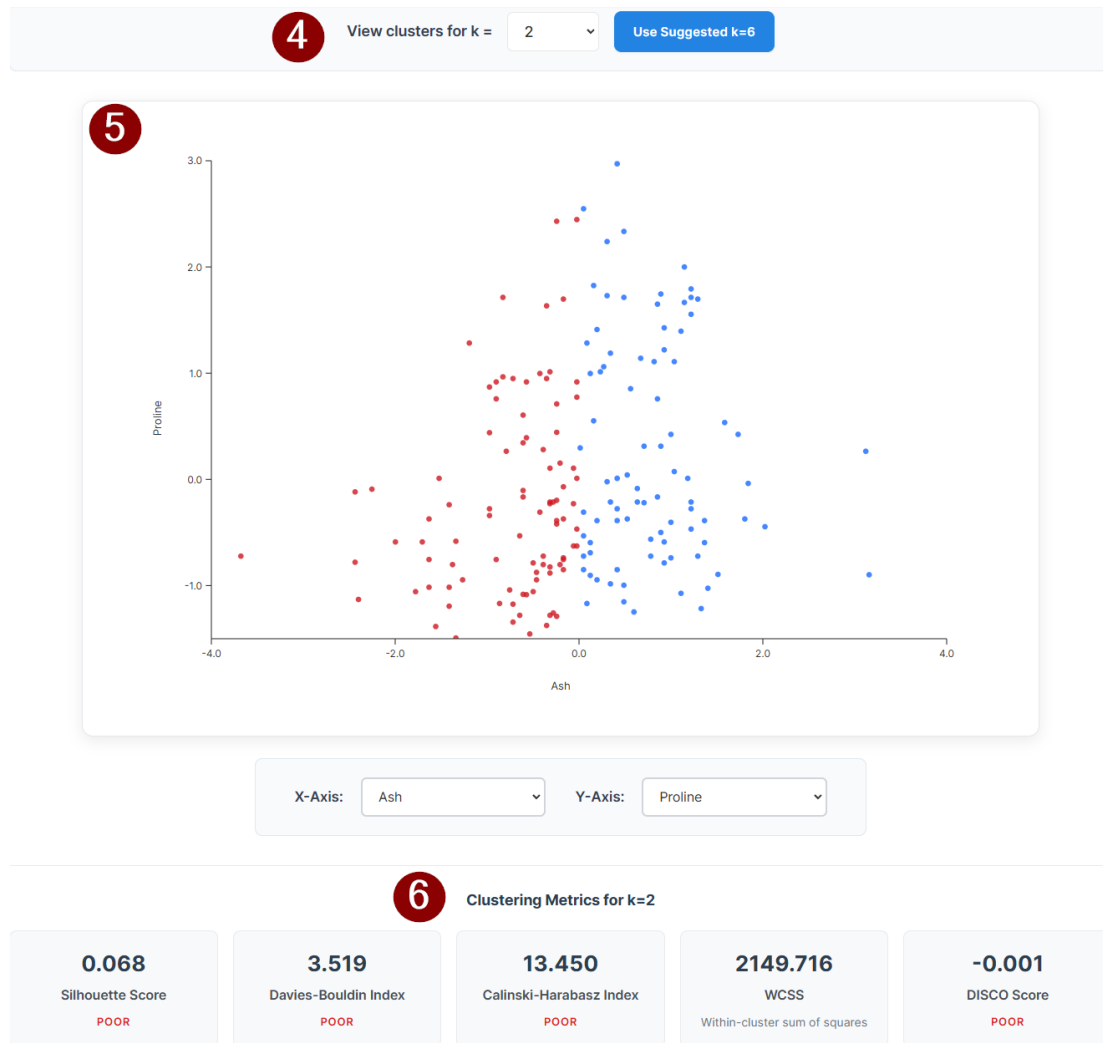


Figure 6.8.: K-selection page scatterplot and metrics overview

6.4. Interactive Clustering

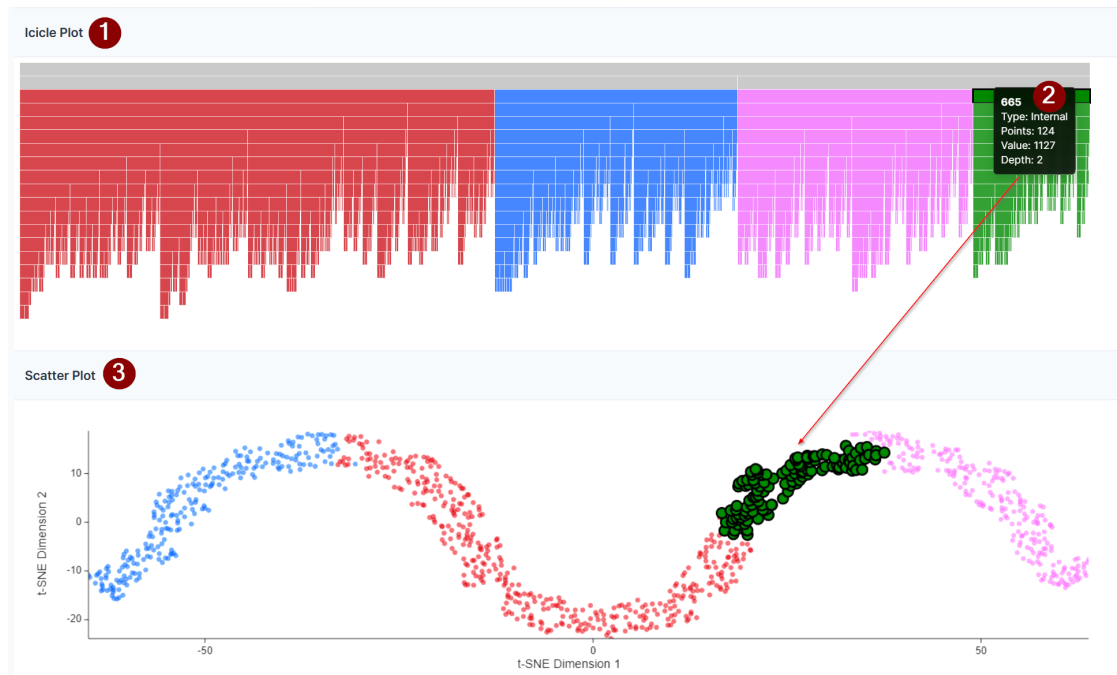


Figure 6.9.: Connected icicle plot and scatterplot on Clustering page

The central capability of the application is the interactive clustering visualization, which can be found on the Clustering page. It offers an interactive tree visualization and scatterplot.

The main Clustering page, shown in Figure 6.9, consists of three elements. The tree visualization can be seen in (1), which can be displayed either as an icicle plot (as in the figure) or as a graph representation. In this visualization, the bars represent clusters, while the vertical levels indicate their progressive subdivision into smaller groups, with each subcluster placed beneath its parent. The color of each block shows its cluster assignment. Grey blocks represent multiple clusters, while colored blocks correspond to a single cluster. Colors are consistent between the tree visualization (1) and the scatterplot (3).

The scatterplot, as seen in (3), shows t-SNE for dimensionality reduction. Multiple interactive options link the scatterplot and the tree visualization. For example, hovering over a block in the tree highlights the corresponding points in the scatterplot, as seen in (2). Conversely, clicking on a point in the scatterplot reveals its position in the tree visualization. Additionally, hovering over either visualization displays tooltips with information such as the number of points in the cluster, as illustrated in (2).

The graph (dendrogram) view, which resembles a dendrogram, is shown in Figure 6.10 on page 47 (4) and (5). This figure showcases the comparison view, which enables the

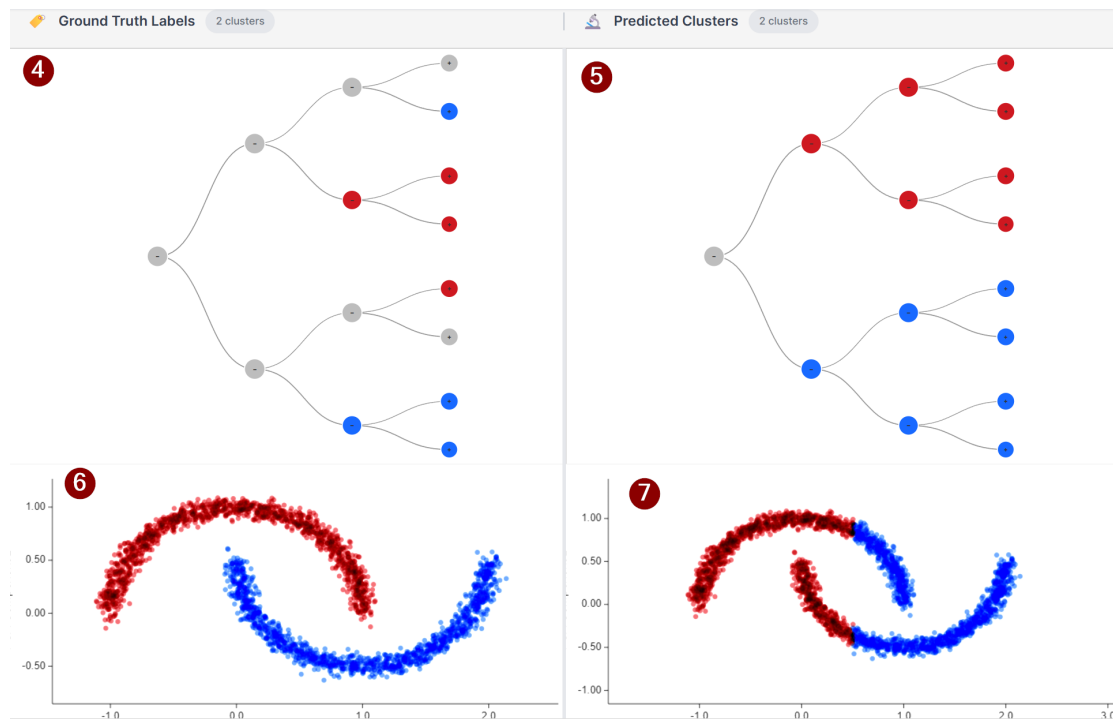


Figure 6.10.: Connected dendrogram view and scatterplot in comparison view to display the ground truth (left) with predicted clusters (right)

6. Final Design and Functionalities

comparison of predicted clusters (5 and 7) with the ground truth, if available (4 and 6). This functionality is useful with toy datasets, where reference solutions are often provided.

6.5. Evaluating Clustering solutions

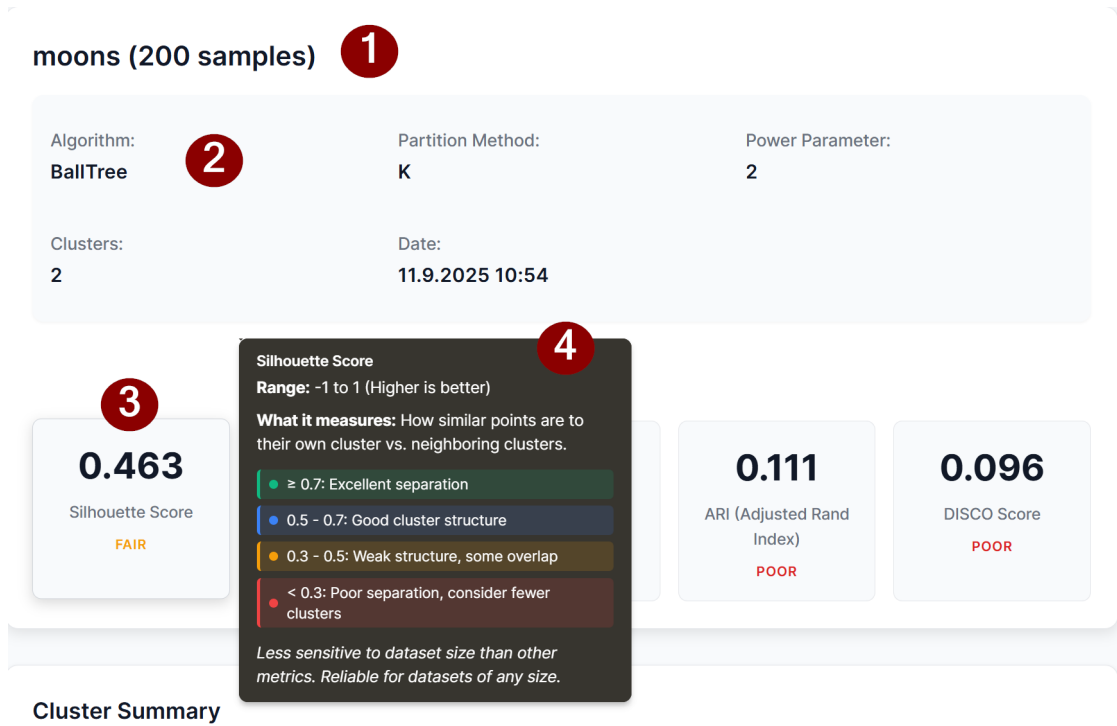


Figure 6.11.: Top section of the Cluster Explorer that shows general information, scores and tooltip for Silhouette Score

Another necessary functionality of the application is that, after creating a clustering using the Clustering page, it is sometimes essential to further assess the quality of the clustering beyond the visual inspection of the scatterplot. For this purpose, the Clustering Explorer page provides additional visualizations and information.

The Clustering Explorer page shows the used dataset, as seen at (1) in Figure 6.11. Furthermore, a brief overview of the most important parameters of the clustering can be seen at (2), as well as various clustering metrics at (3). The clustering evaluation relies on several well-established metrics: Silhouette Score, Davies–Bouldin Index, and Calinski–Harabasz Index, which measure the compactness and separation of clusters. If the ground truth is available, the Adjusted Rand Index (ARI) is also included to assess how well the clustering matches the real solution. In addition, we use the DISCO metric, a density-based clustering metric that evaluates the quality of the clustering.

Another important aspect of the usability of the page is explainability. Therefore, as

seen in (3), each metric is classified for a quick interpretation to help inexperienced users determine whether the score is good or bad. Furthermore, a tooltip for each metric is available when hovering, as seen in (4), which gives even more detail on how each of the metrics works. The page provides further information about clustering statistics, including cluster size and compactness. Finally, the page also provides information about the importance of each feature for the clustering results. The feature importance is calculated by using the in-cluster variance and between-cluster variance of features.

6.6. History

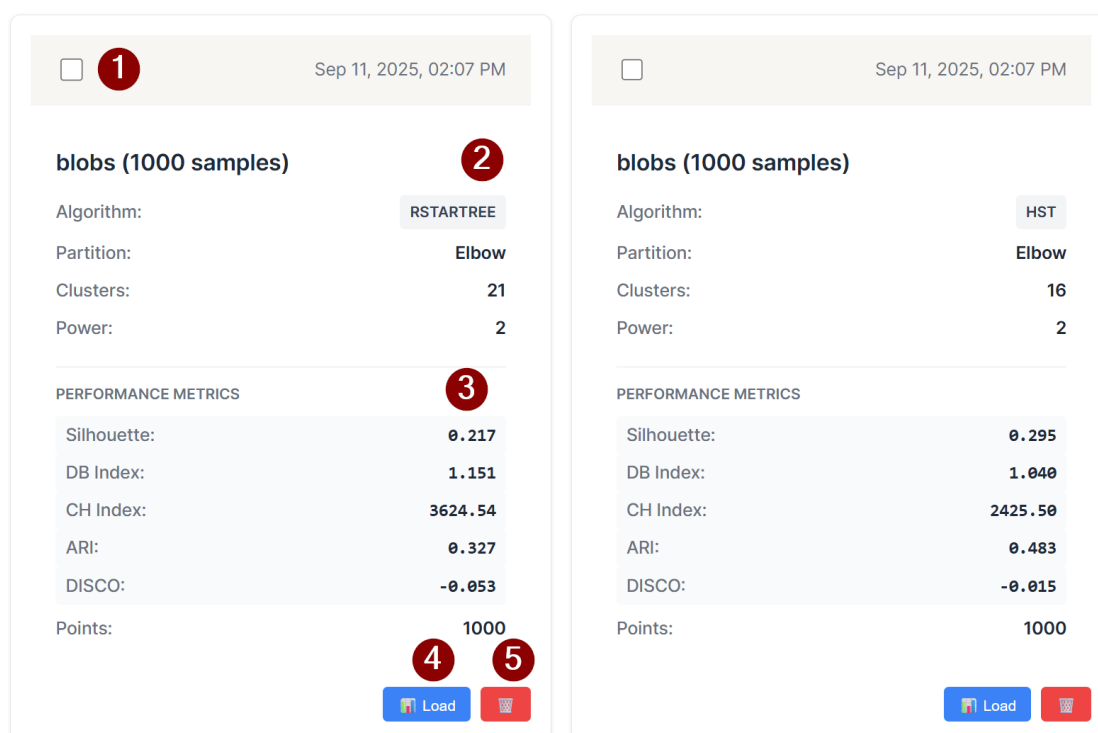


Figure 6.12.: History page shows these cards for each of the clustering runs

The history functionality is important for users to quickly switch between different runs. The options to switch to historic runs are either on the sidebar or on a dedicated History page, where more details are shown.

The History page can be seen in Figure 6.12, which shows the cards that represent the runs on the page. These can be filtered and compared using the toggle at (1). Information about the dataset and parameters used can be seen in (2). Furthermore, as a quick overview, the different metrics can be seen in (3). The button (4) allows the user to load this run and switch to the Clustering page, while (5) allows the user to delete the run from the history.

6.7. Comparison



Figure 6.13.: Comparison page visualizations and metrics

The Comparison page offers the crucial feature of comparing different runs to each other. It can be accessed from the History page by selecting up to 5 different runs.

At the top of the page, the Comparison page offers the possibility to visually compare metrics of the runs with each other, and at the bottom, there is also a table to compare all the metrics and parameters with each other. From there, it is also possible to load a selected run. In between these two elements are the clustering cards, which can be seen in Figure 6.13, where (1) shows the parameters for each card. The scatterplot at (2) allows the user to visually and easily compare different clustering results with each other. Furthermore, it is easy to compare metrics as seen in (3), as the application automatically highlights in green the best value for each metric to help the user get a better understanding of the different clustering runs.

7. Evaluation and Results

This chapter and its evaluation of the design represent the final iterative step of the design process. It represents the empirical validation of the SHIP.AHOI visualization tool. The primary objective of the user study is to determine if the implementation addressed the requirements defined in Section 5.1.3. Furthermore, it evaluates if the design meets the usability requirements and also the needs of both novice and expert users.

To evaluate the design, the users study features eight participants with varying levels of clustering experience. The user study combines quantitative metrics and qualitative feedback to assess the tool's performance and user experience. A table of participants can be found in Table 7.1 on page 52.

Another important factor is that some participants were chosen from the design study (Participants 1, 3, and 5), while also new participants were introduced. This should provide information on whether users who participated in the user study like the final design and are satisfied with the design process. Additionally, it should demonstrate whether new users can understand the SHIP framework from scratch and work effectively with the tool. The participants were selected based on the same criteria as in the design study, which required that all participants have basic clustering knowledge already.

The goal of the user study is to validate the tool's effectiveness and also to validate the design decisions made in the design study. The evaluation consists of different elements. Each participant was asked to complete a series of six Tasks designed to cover all major functionalities of SHIP.AHOI. These Tasks ranged from basic data exploration to advanced clustering comparisons. During the session, we collected both quantitative data (Task completion times and SUS scores) and qualitative feedback (verbal comments and observations). Additionally, participants completed the System Usability Scale (SUS) questionnaire to provide standardized usability metrics that could be compared across people. The complete Tasks can be found in Section A.2

The next chapter includes the results of the conducted user study. First, the summaries of all the interviews are presented and are then analyzed at the end of the chapter.

7. Evaluation and Results

ID	Role	Background / Expertise	Design Study Participant
P1	PhD Candidate, linked to SHIP paper	Development of clustering algorithms, supervision of students	Yes
P2	PhD Candidate	Advanced clustering user, PhD in visualization	No
P3	Student	Clustering courses, practical university experience	Yes
P4	Support Engineer	Basic knowledge of clustering from Bachelor at University	No
P5	Student	Basic knowledge in clustering from courses, frontend development experience	Yes
P6	Developer	Clustering courses, software engineering background	No
P7	Student	Clustering courses, focus on algorithm development	No
P8	Teacher and Domain Expert	Basic clustering knowledge, domain knowledge in Biology	No

Table 7.1.: Overview of user study participants

7.1. Results of the User Study

In the following subsections, summaries of the interviews conducted for the user study are presented.

7.1.1. Interview Participant 1

Participant 1 is an author of the SHIP paper and has extensive experience with clustering, including several papers on the topic, making the feedback particularly valuable. They assigned a SUS score of 95. Task times were average for Tasks 1-4. The time for Task 6 was longer because of a bug that required recalculating the clustering.

They explored multiple visualization options and used the Data and Clustering Explorer frequently. The correlation heatmap on the Data Explorer page helped them confirm known relationships in the wine dataset between the features alcohol, color, and proline. In the digit datasets, they focused on trying different dimensionality reduction techniques in order to separate the numbers better. They noted that some digits separate well. However, digits like 0 and 7 always appear similarly grouped. During the comparison, they preferred the visualizations over the metrics.

Key Features Appreciated

- **System Cohesion:** The user liked the onboarding process, as it explained the framework well and also provided helpful examples. The interviewee also mentioned that the texts used were straightforward to understand.
- **Interactivity:** The most valuable feature for them was the ability to interactively explore the dendrogram/icicle plot and see the connection between them in the scatterplot.
- **Data Overview:** The initial data overview, especially the correlation matrix, was beneficial for feature exploration.
- **History for Comparison:** The History page was valuable for comparing different runs and identifying the best results.

Suggestions for Improvement

- **Manual Cluster Splitting:** Suggested that a feature to split an existing cluster manually would be useful.
- **Visualization Adjustments:** The Participant wanted to adjust scatterplot axes (compress/stretch them).
- **Icicle Plot Interaction:** Recommended that the icicle plot have more collapsible levels to manage its detail better, as it can be overwhelming at first.
- **Some Inconsistencies:** Noted some minor inconsistencies with tooltips (sometimes missing, for example, in the Cluster Explorer) and the data upload placement of the CSV element.

7.1.2. Interview Participant 2

Participant 2 is a PHD candidate focusing on the area of visualizations and assigned a SUS score of 82.5. They spend the most time in the user study and devoted much time, especially on Tasks 2 and 3.

The user set a special focus on closely checking how each metric aligns with the visible structure of the clustering results. For example, they questioned the Davies-Bouldin score, where compactness was not reflected in the scatterplot. At first, the user was confused by the icicle plot, but after some time using it, it became a effective tool for exploring the hierarchy. A small problem occurred when they loaded runs instead of comparing them in the last Task. In general, the user attempted to utilize all available visualization options and parameters to achieve a satisfactory result. It was also mentioned that the tooltips were particularly helpful in using the tool, making it easier for them to use the application.

7. Evaluation and Results

Key Features Appreciated

- **Icicle Plot:** User specifically mentioned the icicle plot as a unique and “very cool” feature that they had not encountered before and found useful.
- **Interactivity:** The instant feedback from hovering and clicking across linked visualizations was appreciated and helpful.
- **Sidebar Workflow:** The step-by-step labeling in the sidebar provided clear and effective guidance through the clustering process.
- **Transparency of Data:** They appreciated the visual transparency for overlapping data points in the scatterplot visualization.

Suggestions for Improvement

- **Performance:** Waiting time for t-SNE and UMAP was mentioned to be a painpoint, as the user wanted faster access to it.
- **Missing comparison options:** The user suggested that looking at ground truth and clustering at the same time does not work and would be useful.
- **Pin Highlighting:** The user requested the ability to pin and select multiple points in the tree visualization simultaneously. So that all the corresponding points would be highlighted in the scatterplot at the same time.

7.1.3. Interview Participant 3

Participant 3, is a student with advanced clustering knowledge from different courses. They reported a perfect SUS score of 100 and completed most Tasks quickly in comparison to the other participants.

They systematically explored all the various parameters, like tree types, and used all pages extensively. They successfully identified correlations between features and uneven cluster size distributions using the visualizations on the Cluster Explorer page. In Task 3, they initially loaded the wrong dataset but corrected it by themselves. The K-selection plot was seen as complex to interpret because the scores were similar between different cluster values. The user effectively utilized the tree visualization and had no issues connecting it to the scatterplot. The user had a special focus on identifying outliers and interpreting why they are outliers. This outlier detection was mainly done with the scatterplot.

Key Features Appreciated

- **Guided Onboarding:** The user specifically mentioned this was beneficial in making the tool more accessible to potential users.

- **Connections:** The participant mentioned that they liked the connection between the tree visualization and scatterplot. They also appreciated that samples were shown for the digits dataset in the scatterplot.
- **History Comparison:** The user appreciated this view and mentioned that it was essential to see all the scores in comparison. They considered it one of the best features for finding a good clustering.
- **Visual Appearance:** The user specifically mentioned that they liked the visual appearance of the application and that this was important for their overall sense of usability.

Suggestions for Improvement

- **Icicle Plot:** The user was initially confused about how to interpret the icicle plot, especially in relation to the ground truth. They also noted that the disappearance of small nodes was a problem.
- **History Comparison:** The user had some problems with the interface on the History page, as they wanted to load rather than compare runs. They also noted that colors should be more consistent in comparisons.
- **Comparison page** The user found cluster colors inconsistent across runs in the comparison view, hindering visual comparison.

7.1.4. Interview Participant 4

Participant 4 has a bachelor's degree in computer science. They are not focused on clustering, but know the basics about it. They rated the tool with a SUS score of 97.5 and described it as beginner-friendly.

They accidentally skipped the onboarding questions as they pressed the wrong button to get started. However, they navigated the app well by using the information from the Starting page, as well as the tooltips and explanations provided throughout the application. On the K-selection page, the user focused on optimizing the scores and heavily relied on the suggested k values. They preferred UMAP/t-SNE over PCA for cluster inspection. Regarding the tree visualizations, they favored the dendrogram for inspecting small clusters and to understand their meaning. They found the comparison feature hidden and wished for interactive scatterplots on the Comparison page, as well as consistent colors across runs.

Key Features Appreciated

- **Communication:** The user specifically mentioned that it is helpful that the application communicates how to use it with the onboarding and tooltips. This

7. Evaluation and Results

also includes the “beginner-friendly design”, as they felt they could not make any mistakes and had the feeling that the application would always find a way back to a working setting.

- **Interactive Elements:** The user valued hover highlighting and the connection between the different visualizations as useful.
- **Structure:** The user mentioned that it is useful that information and insights are clearly separated across pages.

Suggestions for Improvement

- **Distance Metrics:** The user mentioned that, in general, they would like to have more information about distance metrics.
- **Export:** The user mentioned that some additional export options, such as an export menu, would be useful.
- **History Comparison Visibility:** The user mentioned that they would make the feature more visible, as the comparison is a bit hidden in the history, and they did not find out about it until the last Task. They also noted that interactive scatterplots in the comparison would be a nice-to-have addition.

7.1.5. Interview Participant 5

Participant 5, who has experience with clustering from multiple university courses, gave the application a SUS score of 92.5.

They did not spend much time on the Starting page and had problems finding the correct button for the guided onboarding. The icicle plot felt overwhelming to them at first. They correctly observed that the used wine dataset is not normalized using the Data Explorer page. They also implied correctly that this negatively affects the clustering results. They appreciated the digit previews, but were unsure about the information that was shown when hovering over data points in the scatterplot. They were uncertain if the information label refers to the ground truth or the clustering label. The user focused on using the different scores for comparing different runs. Overall, they praised the number of different features and logical organization.

Key Features Appreciated

- **Functionality:** The user mentioned that it is nice to have a tool that can do many things at once and mentioned it could be useful for university assignments.
- **Visualization Connections:** The user mentioned that the visual feedback across the visualizations was beneficial and well done.

- **Robust Design:** The user felt that the system was stable and well-built, and they could not break it by accident.
- **Step Workflow:** The user liked that the workflow was clearly marked by steps for the clustering, which helped them to quickly understand the clustering process.

Suggestions for Improvement

- **Comparison View:** The interviewee mentioned that the comparison is too hidden in the History page.
- **Summary Parameters:** The user indicated that they would like a summary of parameters used after the guided onboarding to have an overview.
- **Onboarding:** The user suggested removing the onboarding from the navigation bar to prevent people from doing the onboarding again by accident, as this resets the current clustering progress.

7.1.6. Interview Participant 6

Participant 6 has taken several university courses about clustering and has some practical experience with clustering. The user reported a SUS score of 82.5.

They read the Starting page thoroughly and interacted with the provided visualizations. The ultrametric explanation helped, but remained partly unclear without more background information. They utilized the correlation heatmap effectively and observed that the tree type has a strong impact on the metrics, further noting that ball trees perform well for the dataset. They initially struggled to find the option to switch datasets and to understand conflicting recommendations for the number of clusters from different metrics on the K-selection page. They understood that both provided tree visualizations. However, they preferred the icicle plot because it provided dense information. They noted that the dendrogram view is easier to understand. In Task 5, they mentioned that separating some digits, like for example 4 and 7, is complicated. In the comparison, they mainly relied on the highlighted metric values to decide on the better clustering solution.

Key Features Appreciated

- **Guidance:** The user found the initial Starting page and guided onboarding helpful and liked that it explained the application well.
- **UI:** The user really liked the intuitive user interface.
- **Visualization:** The user appreciated the visualization of the icicle plot and the interactivity option it provided.

7. Evaluation and Results

- **Comparison:** The user valued that it was possible to compare runs easily and said that especially the metrics helped them in comparing.

Suggestions for Improvement

- **Metrics on K-selection page:** The interviewee mentioned that the metrics were hard to understand on the K-selection page, as they did not understand why certain metrics, for example, the elbow plot, recommended certain k-values. Furthermore, they were confused why k=10 was hardly recommended by the metrics, which represented the ground truth in the digits dataset.
- **Data Upload:** The user wished to have the data upload location more prominent, so it would not be missed so easily.
- **Hard-to-find Features:** The user mentioned that they sometimes had difficulties finding certain features (for example, ground truth comparison), but said that after some time with the tool, it became easier.

7.1.7. Interview Participant 7

Participant 7 has basic knowledge of clustering from a few related courses. They rated the tool with a SUS score of 85.

They skipped most of the Starting page, with the comment “Who reads instructions?”. However, the user used tooltips to orient and completed early Tasks quickly by locating controls efficiently. The user struggled to understand the concept of the correlation heatmap. On the K-selection page, they relied mainly on the scatterplot and recommended values to select a fitting value, rather than deep metric interpretation. They preferred the dendrogram over the icicle plot, and the user also hid the scatterplot to focus on the tree visualization several times. In the final Task, the Participant focused mainly on metrics to select the best run.

Key Features Appreciated

- **Interactivity:** The user found the visualizations and the connections between them, especially the scatterplot and tree visualization, helpful and unique.
- **UI:** The user liked the style and user interface of the application.
- **All-in-One Place:** The user appreciated that information was available on demand and mentioned that there was a good balance between features and usability for the domain.

Suggestions for Improvement

- **Feature Distribution:** The user mentioned that the feature distribution on the Data Explorer page should have adjustable axes.
- **Vector Graphics:** The user suggested adding an option to export vector graphics, like SVG, instead of PNG.
- **Correlation Matrix:** The user did not understand in detail how the correlation matrix worked and was overwhelmed.

7.1.8. Participant 8

Participant 8 has a bachelor's degree in education and teaches subjects like biology, computer science, and digital foundations. The user rated the application with a SUS score of 80.

The user mentioned that the animated onboarding was helpful. However, they found the rest of the Starting page to be overloaded and too detailed. Furthermore, it was noted that the button to start the clustering should be placed more prominently, and some preview visualization of the datasets would be helpful. Moreover, the user found the correlation heatmap useful, but noted that it was a bit small and a bit unclear to them. Tooltips for the metrics were praised, but feature importance values were also too small. On the K-selection page, the user followed the suggested number of clusters ($k=4$). However, they noted that it was not intuitive, as the real number of clusters should be 10. For the tree visualization, the Participant used both visualizations, preferring the dendrogram as more understandable and comparing it to a phylogenetic tree. PCA worked best for the user, and during the comparison stage, the Participant focused heavily on the provided metrics and explanations.

Key Features Appreciated

- **Interactivity:** The user found the visualizations and the connections between them useful, as they helped them understand the data better
- **Dendrogram:** The dendrogram was familiar to the user and offered an easy-to-understand view of the hierarchical structure.
- **All-in-One Place:** The user praised that the Clustering page offered the most important visualizations and options in one place.
- **Sidebar** The Participant mentioned that the sidebar had powerful options and used the space efficiently.

7. Evaluation and Results

Suggestions for Improvement

- **Explainability:** The user requested more detailed information for the heatmap and some more in-depth explanation for each page on top of the page.
- **Onboarding:** The user mentioned that they would prefer a more compact Starting page and a quicker way to start clustering.
- **Colors:** It was mentioned that the Colors in the scatterplot could be more different from each other, as some looked similar.

7.2. Discussion of User Experience

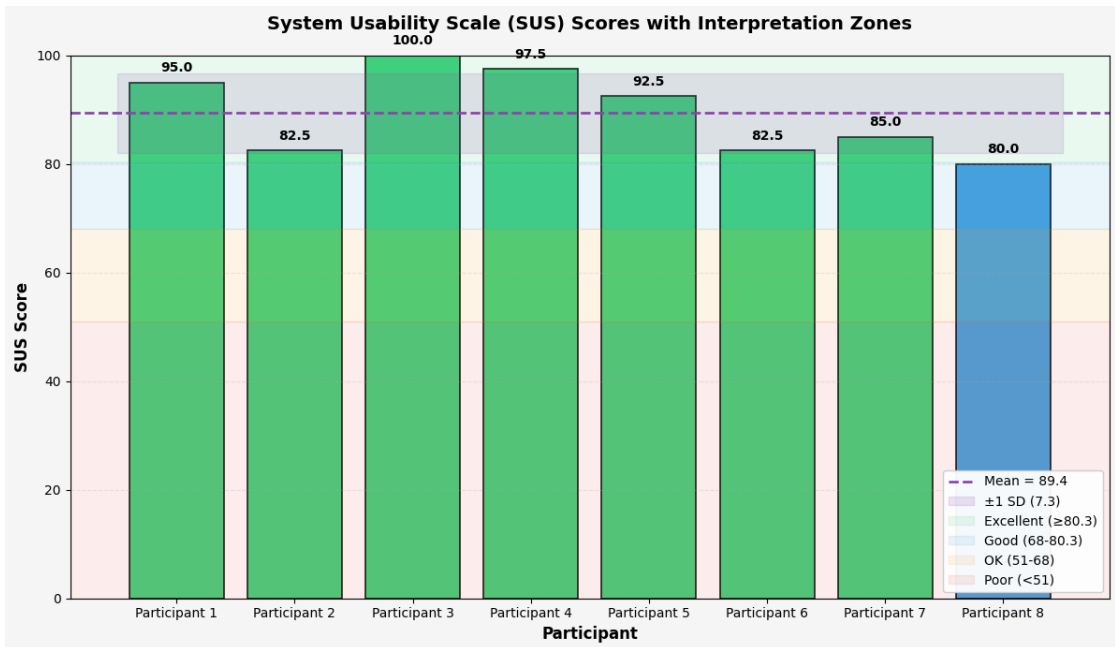


Figure 7.1.: System Usability Scale results, each Participant is visualized using one bar. The zones highlight acceptability metrics for SUS tests based on an empirical study by Bangor et al. [49] for better interpretability.

The interviews provided many valuable insights from the various users who had utilized the tool. A lot of insights were gathered on how users actually interact with it. In general, the SUS feedback provided by the participants was positive. The mean value across all interview partners and questions is 89.4. Using the scale provided by Bangor et al. [49], all except one Participant rated the application as “Excellent”. To explain the SUS score further in detail, nearly all participants indicated they would definitely use the application, and almost all participants said that the system is not unnecessarily complex. Regarding whether the system is easy to use and whether technical assistance would be

7.2. Discussion of User Experience

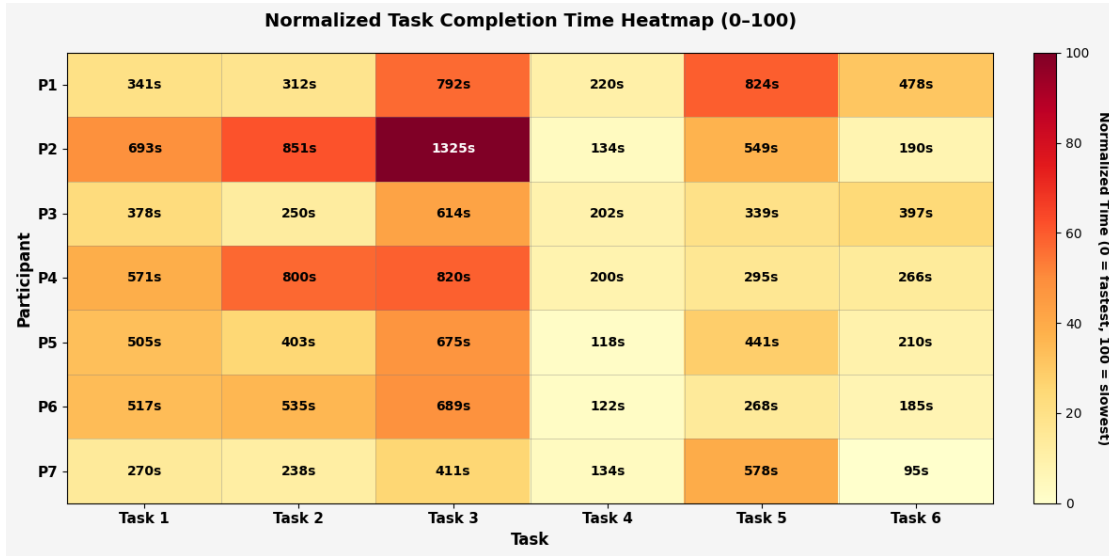


Figure 7.2.: User Study Task completion time Heatmap, per User, per Task

required, the score was also excellent at around 90 percent. The detailed SUS scores can be found in Figure 7.1, while time for each tasks can be found in Figure 7.2.

Another notable finding was that almost all the participants considered the system components to be well integrated into each other. This suggests that the tool not only offers the required functionality for effective data exploration but also ensures seamless interaction between elements. The majority of the interview partners also indicated that the system showed no inconsistencies. By contrast, the question “I would imagine that most people would learn to use this system very quickly” received a slightly lower, average score of 87.5 percent. This outcome is understandable given the complexity of the clustering domain and the powerful application. Most participants also agreed that the application was not cumbersome to use.

The lowest scores were observed for the final two questions. The question concerning users’ confidence when operating the system achieved an average of 75 percent, the worst of all questions. The question of whether substantial learning would be required before use received 78 percent. These results can also be attributed to the demanding nature of the domain. Many participants reported having only limited prior experience with clustering, and some had not worked with it recently. It is therefore reasonable that participants feel a certain level of insecurity when applying clustering and perceive the system as more challenging to master, despite its overall usability.

Overall, the SUS scores are high, with each question receiving more than 75 percent of the possible points, indicating strong user satisfaction across all aspects of the system. Additional findings from the interviews are now organized around their central topic and analyzed.

7. Evaluation and Results

7.2.1. Interactivity

A crucial aspect for the success of the application is the interactivity and focus on visualizations. The visualizations were well-received by users, who appreciate that data is visualized on pages such as the Cluster Explorer and Data Explorer. However, most feedback focused on the interaction between the different visualizations as the best part of the application, especially on the Clustering page.

The part that most interviewees mentioned regarding the interactivity was the Clustering page, where the scatterplot and tree visualization are deeply connected with each other. On the Clustering page, especially the hovering in the tree, which highlights the corresponding points in the scatterplot, was mentioned as making it easier for people to understand how the clustering works. Other interactive features, like clicking on points in the scatterplot and a preview of the data points, were also seen as positive and helpful.

Some users mentioned having even more interactive tools, such as holding Shift to highlight multiple points in the tree at once, and various other wishes. Here, it is essential to strike a balance between offering numerous features and avoiding overloading the application, so that new users are not overwhelmed or unable to find what they need. Therefore, adding a few minor features could be considered. Since the current implementation has received positive feedback, no major additions are necessary.

7.2.2. Onboarding and Tooltips

The feedback on the onboarding process and the Starting page was mostly positive throughout the interviews. Participants praised the Starting page and onboarding for helping them understand the framework in clear, easy-to-understand language. The tooltips were also seen as helpful in providing additional information about metrics and how they work. Also, the tooltips for the parameter selection were seen as particularly helpful.

Interestingly, Participant 7 skipped the Starting page, proceeded directly to the guided onboarding, explored the tool, and relied solely on tooltips to understand the various parts. This is an interesting finding, as perhaps many students who are in a hurry might also skip the Starting page. Still, it can be seen as a success that users can understand the application and how it works without needing to read the Starting page in depth.

However, interviewees correctly pointed out that, for example, in the Cluster Explorer, some of the statistics could be explained in more detail to give both new and experienced users a better understanding of the clustering.

7.2.3. Icicle plot vs Dendrogram

Another interesting aspect observed in the interviews was the preference for icicle plots versus dendrograms for the tree visualization. Two aspects were of special interest in that

regard. Firstly, the familiarity aspect: most participants were unfamiliar with icicle plots and initially felt overwhelmed by them, as they did not fully understand how they work. They preferred the dendrogram view to get started. Nevertheless, with more time using the icicle plot and having to find splits in the data, participants began to understand its benefits and preferred it for certain Tasks.

A point they liked about the icicle plot was the higher information density, as it allows users to see most of the data at once. However, it is impossible to see some small details, such as small clusters, as the blocks become too small. For that, participants preferred the dendrogram, which was limited by not showing the entire hierarchy at once and requiring more clicking to find the desired parts.

This split in the user base regarding visualization preference is not a failure or a negative aspect. Instead, it demonstrates that the platform effectively appeals to diverse user preferences and levels of expertise.

7.2.4. Metrics

An interesting observation from the interviews was the varying use of metrics across the system by different users. From the observations, it was clear that there are groups of users who focus mainly on visualization and others who focus mainly on metrics, some also use both. However, different users at different skill levels work differently here. From the interviews, people with less experience in clustering tended to focus more on the metrics than experienced users. That is no surprise, as just looking at a number to determine the quality of a clustering is an easy way to settle on quality. In contrast, domain experts look more closely into the clustering results, which is easier with visualizations, and they also know about the limitations and focus of certain metrics.

A good example of a user with more experience, who used the metrics selectively and analyzed them, was Participant 2. However, the divide between people preferring visualizations versus metrics is no problem. The goal of the application is to appeal to different target groups, which is effective if both groups find both metrics and visualizations in all the important parts of the application.

7.2.5. Insights

Another interesting observation from the interviews was how people gather insights about the data and the clustering. As mentioned before, metrics play a role here, but some visualizations were seen as important by users. A good example of this is the correlation matrix on the Data Explorer page, which provided many insights to users. Another example is the feature importance on the Clustering page, which also led to valuable insights in the interviews. Of course, the Clustering page also provided a lot of insights, as it is the center of the application. However, the K-selection page was not really mentioned as being a source of insight.

7. Evaluation and Results

7.2.6. User Interface

The UI was also seen as quite positive, as indicated by the SUS scores and the feedback provided. Some minor inconveniences were mentioned, for example the suggestion to move the CSV upload to the top of the Data Upload page instead of the bottom. Furthermore, some issues with finding the correct elements were observed, mainly on the History page and also on the main Clustering page. Since the Clustering page offers many features, a balance must be found between options and customizations on the one hand and simplicity on the other. However, some area for small improvement exists.

7.2.7. Improvements

Some areas for improvement were identified during the interviews and are addressed in the final design. A minor point that was mentioned was that the statistics and correlation heatmap on the Cluster Explorer page should include tooltips. This has been added along with further tooltips. Additionally, the comparison view on the Clustering page is improved and reworked to offer a side-by-side view instead of switching between the visualization of the ground truth versus the clustering. This allows users to better compare their current clustering to the ground truth.

Another improvement is to add better explanations to the icicle plot, as many users mentioned that they see it as more powerful and valuable than the dendrogram, but do not fully understand it. Therefore, an explanation of how the icicle plot works was added to the Starting page, along with visualizations. For explanation, the same dataset is shown in both an icicle plot and a dendrogram to demonstrate better how they work with a familiar comparison.

The feedback to highlight multiple points in the tree visualization at the same time by hovering has been implemented. If the user holds shift and clicks, they can now select multiple elements in the tree visualization for highlighting in the scatterplot. Improving the speed of dimensionality reduction, unfortunately, is not possible without sacrificing the quality of the results.

Furthermore, the color palette for the scatterplots for the final design has been improved and uses the Gasbey color palette [50]. This qualitative color palette allows for 20+ colors, while the colors are still as distinct as possible.

8. Discussion

As we refer back to the introduction of the thesis in Chapter 1, the goal was to create a visualization for the SHIP framework using an iterative, user-centered approach. The tool should also be targeting both students and domain experts. It should be user-friendly and provide detailed insights along with explanations. For this, it is helpful to look back at Chapter 7, where we can see that the partners in the user study found the tool both accessible and user-friendly. This is supported by the SUS questionnaire, which awarded the tool approximately 89 percent of the possible points. Users also praised the tool for the advanced explanations, which go beyond basic implementations, by using interactive visualizations and multiple forms of help, such as tooltips. All of this was mentioned by both domain experts and students, covering the target groups defined in the beginning.

Looking at the formulated requirements in Section 5.1.3, the core functionality is fully available in the final design. It is possible to load and inspect data and also look at basic statistics. Parameter settings for all SHIP parameters are available, as well as a 2D scatterplot visualization, which users preferred over a 3D representation. The design also enables rapid exploration of various solutions. Regarding data handling, it is possible to load and save configurations on the History page and to export results from the application as labels or visualizations.

Regarding the visualization requirements, SHIP.AHOI provides fast parameter switching with immediate feedback, either with results or a short loading bar. It is possible to evaluate clusters in various visualizations and metrics, and they can also be compared directly on the Comparison page. The goal of providing guidance for the tool is addressed through extensive guidance and tooltips, both at the start of the application and throughout the workflow. The goal of suggesting parameters was implemented in a basic form. Since it is often challenging to suggest parameter ranges, we provide solutions that work best for most use cases but can be freely adjusted. Regarding optional features, all of the optional and advanced features have been successfully incorporated into the design. The hierarchy tree is visualized, as was decided by user-driven feedback in the design study, and it is possible to interact with either the icicle plot or the dendrogram view in various ways. Feature importance is also shown, together with the detailed information in the Data Explorer and Cluster Explorer. Some of the partition methods cover noise and outliers and display them in various ways. Manual marking of nodes as noise, however, is not possible, as it was not requested much by users and would complicate the application significantly by adding manual overrides to the clustering. Preprocessing options are available for CSV uploads—for example, normalization or missing data handling.

8. Discussion

In conclusion, we can say that all the tasks, including the optional ones, were implemented, as well as many additional features that were developed in the design study.

The thesis successfully achieved its goal of creating a visualization for the SHIP framework, implementing its tasks, and developing a detailed clustering framework that both novice and advanced clustering users can utilize. Compared to existing tools, SHIP.AHOI stands out with its extensive set of features that require no coding skills and are easily available in one place. No other tool integrates the SHIP framework or a comparable framework into its design and implementation, and therefore, they miss out on the unique speed benefits. Furthermore, the design study managed to gain significant insights into how users with different expertise levels approach clustering and what tools they need to have an application that suits their needs. The main finding here was that experienced users focused more on visualizations and used advanced diagrams like the icicle plot, while novice users focused more on metrics and simpler visualizations. This highlights that tools for visualizations also need to consider the diverse preferences of user groups, rather than simply representing as much information as possible.

Further insights of the design study are that features like the “Drill Down page”, initially seen as useful, received mixed feedback and were therefore cut. Conversely, the History page emerged as an important aspect of the application, based solely on user requests. This demonstrates that the design study effectively translated user needs into a useful and usable design.

Further from these findings, several implications emerge:

- **Interactivity:** Users consistently mentioned that interactivity with visualization is a key aspect when working with clustering data, and impacts the usability and offers further insights.
- **Layered complexity:** Inexperienced users benefit from guided onboarding and tooltips, while experts require more advanced exploration options. Both groups can be served if advanced features are put into various optional pages and assistance, such as tooltips, are provided.
- **Explanations:** Visualizations like the icicle plot remain underutilized without enough explanations. Future clustering tools should also prioritize pedagogy as a central component in designing the application.

Compared to existing clustering tools such as Orange, Tableau, or XCluSim, SHIP.AHOI stands out for integrating quick parameter exploration and various visualizations into a single platform. It is designed to combine the benefits of the tools discussed in the related work section in Chapter 3 and build up on that. Clustergrammer demonstrated the power of a single visualization; these insights shaped the design of the Starting page. Furthermore, Clustering Visualizer and Clusterphile 2 highlighted the importance of explainability and the educational focus of the application. The splitting up of complex workflows, as seen in Orange, was also essential to consider in the design of SHIP.AHOI. Clustervision provided additional insights into the run history and the combination of

different visualizations that were also used.

Building on these insights, the design focused on applying the heuristics found in Section 2.3.3 as well as the user guidance discussed in Section 2.3.2. In addition, Section 2.2 provided the necessary background for this thesis on hierarchical visualizations, such as the icicle plot used here and on visualizations like the scatterplot.

9. Conclusion

In conclusion, this master thesis contributes to the area of data exploration and clustering visualization by introducing SHIP.AHOI. SHIP.AHOI is a visualization design that utilizes the benefits of the SHIP framework. The work addresses a current gap in clustering visualization tools, which do not support the quick generation of multiple clustering solutions and also integrates visualizations deeply into their clustering workflow.

This thesis developed a visualization design that integrates parameter exploration, multiple connected visualizations, and numerous in-depth information pages into a cohesive and user-friendly interface. The design study process used to develop the design involved iterative prototyping and extensive user feedback for each iteration of the prototypes. This approach ensured that the final prototype not only meets the technical requirements but also considers the real-world practical needs and workflows of its potential users.

A main contribution of this work is its focus on both interactive visualizations and usability. The tool utilizes linked visualizations, including icicle plots, dendrograms, and scatterplots. They enable users to understand their data, explore the hierarchical structure, and compare different solutions. Additionally, features such as guided onboarding and tooltips are designed to support inexperienced users while still providing powerful tools to experienced users. By integrating these elements into the tool, the thesis illustrates how visualization can transform clustering into an efficient and iterative process.

A conducted user study further validates the final design. The user study revealed that the design enhances understanding and exploration for both novice and expert users.

Despite these contributions, the work acknowledges limitations and offers possibilities for future work. A limitation is that only two tree visualizations are available. Future work could explore new visualization techniques specifically targeting ultrametric clusterings to find a more intuitive and informative visualization of the tree structure that is also dense in information. Limitations are that the design focuses solely on the SHIP framework and is therefore limited in the choice of clustering algorithms. Furthermore, the design's handling of noise and outliers is limited and could be further expanded.

In summary, this thesis not only provides a functional and innovative visualization design for the SHIP framework but also serves as a good example for designing accessible data analysis tools that offer numerous features. It also makes the SHIP framework more approachable for users, enabling them to leverage its benefits without requiring coding.

Bibliography

- [1] M. Canizares, A. Gibson, J. Willis, and D. Lovell, “Towards meaning-making with interactive visualisations,” in *Proceedings of the 2023 ASCILITE Conference: People, Partnerships and Pedagogies*. Australasian Society for Computers in Learning in Tertiary Education, 2023, pp. 45–55.
- [2] J. Bae, T. Helldin, M. Riveiro, S. Nowaczyk, M.-R. Bouguelia, and G. Falkman, “Interactive clustering: A comprehensive review,” *ACM Comput. Surv.*, vol. 53, no. 1, Feb. 2020. [Online]. Available: <https://doi.org/10.1145/3340960>
- [3] A. Draganov, P. Weber, R. S. M. Jørgensen, A. Beer, C. Plant, and I. Assent, “I want ’em all (at once) – ultrametric cluster hierarchies,” 2025. [Online]. Available: <https://arxiv.org/abs/2502.14018>
- [4] M. Sedlmair, M. Meyer, and T. Munzner, “Design study methodology: Reflections from the trenches and the stacks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2431–2440, Dec. 2012. [Online]. Available: <https://doi.org/10.1109/TVCG.2012.213>
- [5] A. Beer, A. Draganov, E. Hohma, P. Jahn, C. M. Frey, and I. Assent, “Connecting the dots – density-connectivity distance unifies dbscan, k-center and spectral clustering,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD ’23. New York, NY, USA: Association for Computing Machinery, 2023, p. 80–92. [Online]. Available: <https://doi.org/10.1145/3580305.3599283>
- [6] E. Ventocilla and M. Riveiro, “A comparative user study of visualization techniques for cluster analysis of multidimensional data sets,” *Information Visualization*, vol. 19, no. 4, pp. 318–338, 2020. [Online]. Available: <https://doi.org/10.1177/1473871620922166>
- [7] G. A. Pavlopoulos, T. G. Soldatos, A. Barbosa-Silva, and R. Schneider, “A reference guide for tree analysis and visualization,” *BioData Min*, vol. 3, p. 1, 2010, place: LONDON.
- [8] K. Rink, “Best Practices for Visualizing Your Cluster Results | Towards Data Science — towardsdatascience.com,” Available: <https://towardsdatascience.com/best-practices-for-visualizing-your-cluster-results-20a3baac7426/>, 2022, [Accessed: 07-09-2025].
- [9] S. Mishra, U. Sarkar, S. Taraphder, S. Datta, D. Swain, R. Saikhom, S. Panda, and

Bibliography

- M. Laishram, "Principal component analysis," *International Journal of Livestock Research*, p. 1, Jan. 2017.
- [10] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: <http://jmlr.org/papers/v9/vandermaaten08a.html>
- [11] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2020. [Online]. Available: <https://arxiv.org/abs/1802.03426>
- [12] M. Sedlmair, T. Munzner, and M. Tory, "Empirical guidance on scatterplot and dimension reduction technique choices," *IEEE transactions on visualization and computer graphics*, vol. 19, no. 12, pp. 2634–2643, 2013.
- [13] L. M. C. Cabezas, R. Izbicki, and R. B. Stern, "Hierarchical clustering: visualization, feature importance and model selection," 2023. [Online]. Available: <https://arxiv.org/abs/2112.01372>
- [14] Geeksforgeeks, "Creating Heatmaps with Hierarchical Clustering - GeeksforGeeks — [www.geeksforgeeks.org](https://www.geeksforgeeks.org/creating-heatmaps-with-hierarchical-clustering/)," Available: <https://www.geeksforgeeks.org/creating-heatmaps-with-hierarchical-clustering/>, 2023, [Accessed: 20-04-2025].
- [15] P. Laubheimer, "Treemaps: Data visualization of complex hierarchies," Sep 2019. [Online]. Available: <https://www.nngroup.com/articles/treemaps/>
- [16] FasterCapital, "Visualization techniques: Sunburst charts: Illuminating data layers with sunburst charts - fastercapital," Apr 2025. [Online]. Available: <https://fastercapital.com/content/Visualization-Techniques--Sunburst-Charts---Illuminating-Data-Layers-with-Sunburst-Charts.html>
- [17] J. B. Kruskal and J. M. Landwehr, "Icicle plots: Better displays for hierarchical clustering," *The American statistician*, vol. 37, no. 2, pp. 162–168, 1983.
- [18] G. Bisson and R. Blanch, "Improving visualization of large hierarchical clustering," Sep. 2012, pp. 220–228.
- [19] A. Agarwal, S. Prabha, and R. Yadav, "Exploratory data analysis for banking and finance: Unveiling insights and patterns," 2024. [Online]. Available: <https://arxiv.org/abs/2407.11976>
- [20] Geeksforgeeks, "What is Exploratory Data Analysis? - GeeksforGeeks — [www.geeksforgeeks.org](https://www.geeksforgeeks.org/what-is-exploratory-data-analysis/?ref=rp)," Available: <https://www.geeksforgeeks.org/what-is-exploratory-data-analysis/?ref=rp>, 2025, [Accessed: 20-04-2025].
- [21] J.-D. Fekete, "Interaction for visualization," Available: <https://www.cs171.org/2015/assets/slides/08-Interaction.pdf>, Feb. 2015, lecture presented in CS171: Visualization, Harvard University, Spring 2015. [Online]. Available: <https://www.cs171.org/2015/assets/slides/08-Interaction.pdf>

- [22] R. Sadana, M. Agnihotri, and J. Stasko, “Touching data: A discoverability-based evaluation of a visualization interface for tablet computers,” 2018.
- [23] M. Pohl, B. Potzmann, C. Stoiber, and W. Aigner, “Inconspicuous guides: A qualitative study to understand the degree of intrusiveness in visualization onboarding,” in *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 1: IVAPP, INSTICC*. SciTePress, 2025, pp. 706–717.
- [24] A. Springer and S. Whittaker, “Progressive disclosure: Designing for effective transparency,” 2018. [Online]. Available: <https://arxiv.org/abs/1811.02164>
- [25] J. Nielsen, “Enhancing the explanatory power of usability heuristics,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 1994, pp. 152–158.
- [26] Plotly-Developers, “Plotly python graphing library sliders,” Available: <https://plotly.com/python/sliders>, n.d., [Accessed: 05-09-2025].
- [27] D3js-Developers, “Sunburst — d3,” Available: <https://observablehq.com/@d3/sunburst/2>, n.d., [Accessed: 21-04-2025].
- [28] S. L’Yi, B. Ko, D. Shin, Y.-J. Cho, J. Lee, B. Kim, and J. Seo, “XCluSim: A visual analytics tool for interactively comparing multiple clustering results of bioinformatics data,” *BMC Bioinformatics*, vol. 16, p. S5, 2015, place: LONDON.
- [29] S. learn Developers, “A demo of K-Means clustering on the handwritten digits data — scikit-learn.org,” Available: https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_digits.html, n.d., [Accessed: 21-04-2025].
- [30] Plotly-Developers, “Plotly python graphing library,” Available: <https://plotly.com/python/>, n.d., [Accessed: 05-09-2025].
- [31] Streamlit-Developers, “Streamlit Docs — docs.streamlit.io,” Available: <https://docs.streamlit.io/>, n.d., [Accessed: 05-09-2025].
- [32] Bokeh-Developers, “Bokeh documentation — docs.bokeh.org,” Available: <https://docs.bokeh.org/en/latest/index.html>, n.d., [Accessed: 21-04-2025].
- [33] Matplotlib-Developers, “Interactive figures 2014; Matplotlib 3.10.1 documentation — matplotlib.org,” Available: <https://matplotlib.org/stable/users/explain/figure/interactive.html>, n.d., [Accessed: 21-04-2025].
- [34] Apache-Developers, “Apache ECharts — echarts.apache.org,” Available: <https://echarts.apache.org/en/index.html>, n.d., [Accessed: 21-04-2025].
- [35] D3js-Developers, “D3 by Observable | The JavaScript library for bespoke data visualization — d3js.org,” Available: <https://d3js.org/>, n.d., [Accessed: 21-04-2025].

Bibliography

- [36] E. Karatzas, M. Gkonta, J. Hotova, F. A. Baltoumas, P. I. Kontou, C. J. Bobotsis, P. G. Bagos, and G. A. Pavlopoulos, “VICTOR: A visual analytics web application for comparing cluster sets,” *Computers in Biology and Medicine*, vol. 135, p. 104557, Aug. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0010482521003516>
- [37] A. Andersson, A. Behanova, C. Avenel, C. Wählby, and F. Malmberg, “Points2regions: Fast interactive clustering of in situ transcriptomics data,” Dec. 2022.
- [38] N. F. Fernandez, G. W. Gundersen, A. Rahman, M. L. Grimes, K. Rikova, P. Hornbeck, and A. Ma’ayan, “Clustergrammer, a web-based heatmap visualization and analysis tool for high-dimensional biological data,” *Scientific Data*, vol. 4, no. 1, p. 170151, 2017. [Online]. Available: <https://doi.org/10.1038/sdata.2017.151>
- [39] T. Metsalu and J. Vilo, “ClustVis: a web tool for visualizing clustering of multivariate data using Principal Component Analysis and heatmap,” *Nucleic Acids Research*, vol. 43, no. W1, pp. W566–W570, Jul. 2015. [Online]. Available: <https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkv468>
- [40] C. Visualizer-Developers, “Clustering Algorithm Visualizer — clustering-visualizer.web.app,” Available: <https://clustering-visualizer.web.app/>, n.d., [Accessed: 21-04-2025].
- [41] U. o. L. Bioinformatics Laboratory, “Orange Data Mining — orangedatamining.com,” Available: <https://orangedatamining.com>, n.d., [Accessed: 21-04-2025].
- [42] A. Sulistio, G. I-Vy, and L. S. L. Evelyn, “Vizts – visualisation of time series clustering,” Singapore Management University, Research Paper, n.d., iSSS608 Group 05. [Online]. Available: https://wiki.smu.edu.sg/18191iss608g1/img_auth.php/0/02/ISSS608_Group05_Research_Paper.pdf
- [43] B. C. Kwon, B. Eysenbach, J. Verma, K. Ng, C. De Filippi, W. F. Stewart, and A. Perer, “Clustervision: Visual supervision of unsupervised clustering,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 142–151, 2018.
- [44] M. Cavallo and C. Demiralp, “Clustrophile 2: Guided visual clustering analysis,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 267–276, 2019.
- [45] Tableau , “Tableau official website,” Available: <https://www.tableau.com/>, 2025, [Accessed: 03-09-2025].
- [46] Tableau, “Clustering in tableau,” Available: <https://help.tableau.com/current/pro/desktop/en-us/clustering.htm>, 2025, [Accessed: 03-09-2025].
- [47] U. o. L. Bioinformatics Laboratory, “Visual programming,” Available: <https://orangedatamining.com/home/visual-programming/>, n.d., [Accessed: 21-04-2025].

- [48] J. Brooke, “Sus: A quick and dirty usability scale,” *Usability Eval. Ind.*, vol. 189, Nov. 1995.
- [49] A. Bangor, P. T. Kortum, and J. T. Miller, “An empirical evaluation of the system usability scale,” *International Journal of Human–Computer Interaction*, vol. 24, no. 6, pp. 574–594, 2008. [Online]. Available: <https://doi.org/10.1080/10447310802205776>
- [50] C. Glasbey, G. van der Heijden, V. F. Toh, and A. Gray, “Colour displays for categorical images,” *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, vol. 32, no. 4, pp. 304–309, 2007.

A. Appendix

A.1. Technical Implementation SHIP.AHOI

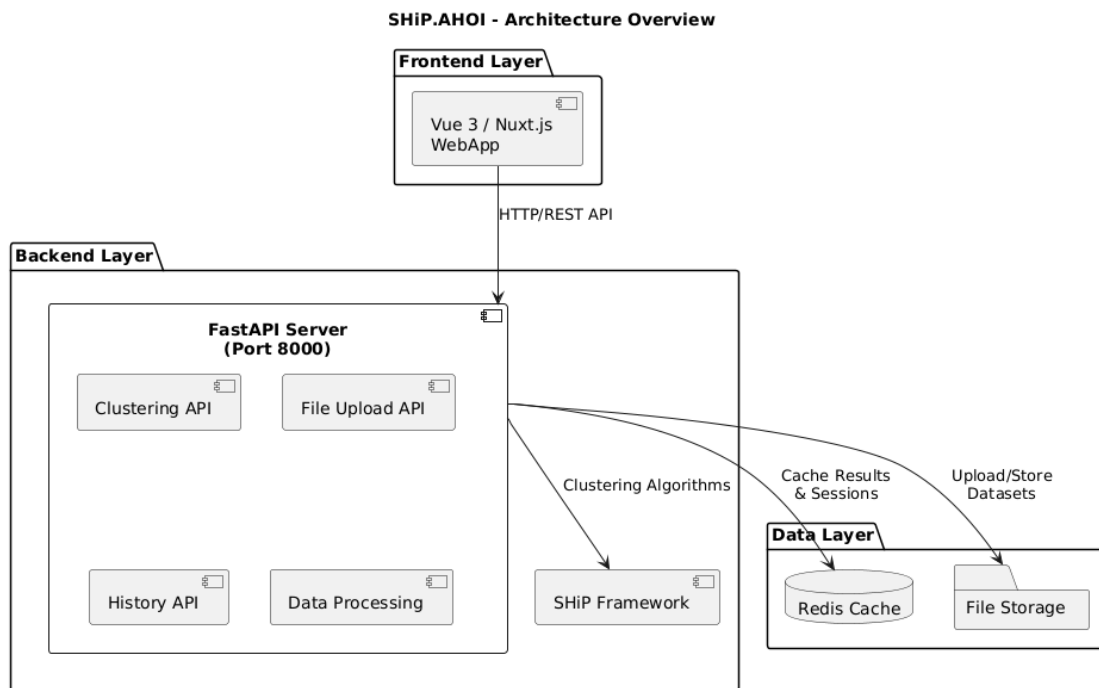


Figure a.1.: Basic architecture of the SHIP.AHOI implementation

A technical overview for the prototype is provided in Figure a.1. The frontend was developed with TypeScript using the Vue.js and Nuxt.js libraries. Furthermore, D3.js was used for the highly interactive visualizations, as it offers the most flexibility and customization among the libraries analyzed in Section 3.1.1. This flexibility is required for the tree visualization on the Clustering page and the scatterplot to implement all the interactive elements while maintaining good performance for large datasets.

The backend, on the other hand, handles the actual clustering using the SHIP framework, as well as many other utilities. In general, the frontend communicates with the backend via API calls to a central API class. The backend provides information such as the history of runs, statistics about datasets, or clustering results. An important aspect is that the backend also calculates the colors for the clusters. This is necessary to ensure that colors

A. Appendix

remain consistent across all parts of the application, for example, between the tree and scatterplot visualizations. Calculating the colors in one place provides a single source of truth, which is important.

Furthermore, the dimensionality reduction mappings are also calculated in the backend. This is done lazily, so clustering and dimensionality reduction are not calculated simultaneously. The frontend queries the backend at regular intervals to check if the calculation is finished. Once ready, the backend returns the data to the frontend, which then seamlessly integrates the UMAP and t-SNE results into the application. This approach reduces waiting time for users, as calculations for UMAP and t-SNE can take longer than clustering, and making users wait would hinder the user experience.

Another important design decision was to run all tasks in separate processes within the Python backend. This ensures that tasks can be cancelled directly from the frontend.

An example of this is that the clustering process can be stopped at any time, which aborts the process and instantly frees all resources. Without processes, but only threads, it would not be possible to terminate a clustering run while it is executing; it could only be stopped before or after the SHIP clustering.

The backend also uses a database, for which Redis is used. Redis dynamically decides whether data is stored in memory or on disk, abstracting this decision away from the application. The database stores both the history and clustering results, making them quickly accessible from the History page. Additionally, the database retains the SHIP trees between runs. This allows the ship object, which was built based on the ultrametric, to be reused if only the power parameter or the partitioning method changes, thereby leveraging the benefits of the SHIP framework.

The deployment of the application is straightforward, as a `docker-compose.yml` file is provided in the project. The user can start the application by running:

```
docker run --rm -v /var/run/docker.sock:/var/run/docker.sock \
stalex7/ship-bootstrap:latest
```

(with Docker installed). Once executed, the application will be available at `http://localhost:3000`.

A.2. User Study Document for Users

In the following section, you can find the Tasks for the Users Study and the System Usability Scale, which was taken from Brooke [48]:

You are a user exploring a new data analysis and clustering platform. Your goal is to understand and segment a dataset to uncover patterns and explainable groupings. As part of your journey, you'll simulate realistic steps: loading data, preprocessing, clustering, and evaluating the results. For this you are going to use the following two datasets today:
Datasets:

1. **Wine Dataset CSV (First Part):** Comprises 178 wine samples from 3 cultivars, described by 13 chemical features (e.g., alcohol, acidity). Clustering is used to recover wine types from continuous-valued attributes and explore feature influence.
2. **Digits Dataset (Second Part):** Contains 1,797 images of handwritten digits (0–9), each as an 8×8 grayscale image flattened into 64 features (each pixel one feature). Clustering aims to group similar digit images.

Note:

1. When applying dimensionality reduction techniques, it is advisable to use the first and second components from the same method for the x and y axes, respectively. Mixing components from different methods (e.g., t-SNE component 1 with UMAP component 2) is not recommended."
2. There are often no right or wrong solutions in clustering. Do not be afraid of doing something "wrong".
3. Feel free to click around into visualization and also use hovering there are a lot of different features in the application

Task 1: Load and Explore Data Characteristics

Scenario:

You've just opened the platform and want to get a sense of how it handles loading and inspecting data. You'll begin with the guided onboarding process.

Instruction:

- Start with reading the information on the main page to understand how the SHIP framework works.
- Begin the **guided onboarding flow** on the main page.
- Load the csv file "**wine-clustering**" (at the bottom of the page, you can use the file as is) and continue with the process
- Answer the guided questions to help choose the fitting parameters (feel free to choose anything you like).
- When on the Clustering page, select Button **Run Clustering** on the sidebar in the parameter section to execute your first clustering
- Feel free to explore the results before continuing.

Task 2: Explore Dataset and Clustering in Depth

Scenario:

You want to dive deeper and understand what drives the clustering. Are some features more important? Are there any surprising patterns?

Instruction:

- Go to the `/Data Explorer` and `/Clustering Explorer` pages (using the navbar on top of the website) using your current clustering and investigate the different metrics and visualizations.
- Examine advanced stats/visualizations which are available on the sites
- **Think out loud:** Note anything unexpected or newly discovered, about dataset/features or clustering.

Task 3: Configure and Adjust Initial Clustering Setup

Scenario:

You now want to create your first clustering configuration to explore structure in another dataset you work with. You're curious about how many clusters might be ideal.

Instruction:

- On the `/clustering` page, change the dataset using the sidebar to "**Digits (Small)**".
- Set parameters on `/clustering` page then.
 - TreeType: **CoverTree**
 - Power = **4**
 - Partition method = **Elbow**
- Explore the scatterplot and change axis options (e.g., switch between different dimensionality reduction methods PCA,UMAP,TSNE). Also, make sure to use the hover function to see an example of the digits. If you want you can also display the ground truth in the visualization options on the sidebar.
- Go to `/k-selection` page (navbar), apply a recommended k -range, investigate scatterplot and metrics.
- **Think out loud:** Try to interpret which k value you find the most fitting for the dataset using the available information on the page and visualizations.
- *Hint:* You can click into metrics to adjust the scatterplot on `/k-selection` page.

Task 4: Produce and Export Clustering Artifacts

Scenario:

You've found a good clustering setup and now want to save your work to share it with other people.

Instruction:

- On the `/clustering` page, finalize your current clustering with the `k` value you found in the last task.
- Export your current clustering on the bottom of the sidebar:
 - Cluster labels as **CSV**
 - Visualizations as **PNG**
- Check the exported files to ensure they work.
- *Hint:* The last column in the CSV file is the cluster assignment.

Task 5: Explain Hierarchy, Merges, and Detailed Insights

Scenario:

You want to better understand the cluster structure. You're exploring how smaller clusters merge and what those merges mean.

Instruction:

- Go to the hierarchical view (**dendrogram** or **icicle plot**, you can switch between them in the sidebar and use the preferred one, on the top half of the `/clustering` page.
- Use the Digits Dataset with parameters:
 - TreeType: **DCTree**
 - Power = **2**
 - $K = 10$
- Find a merge point where two clusters become one.
- Use the linked scatterplot to highlight the clusters.
- **Think out loud:** Try to interpret what the merge means and what data points are involved. (Use either the dendrogram or icicle plot – or both.)
- **Think out loud:** Try to find insights besides the merge that are available in the icicle plot / dendrogram that are interesting

Task 6: Compare and Evaluate Multiple Configurations

Scenario:

You've tried a few different clustering setups and want to compare them. Which one is best, and why?

Instruction:

- Go to the `/history` page.
- Load the initial clustering from Task 3 and compare it to a newer setup (or different setups you tried).
- Use:
 - Metrics (like ARI, Silhouette Score, etc)
 - Visualizations: scatterplot comparison
- Choose a preferred configuration and reload it on the `/clustering` page.

General Feedback

- Please fill out the SUS (System Usability Scale) questionnaire for the overall system.

System Usability Scale (SUS)

Taken from [48]

For each of the following statements, please mark how strongly you agree or disagree.

Scale: 1 = Strongly Disagree 5 = Strongly Agree

1. I think that I would like to use this system frequently.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

2. I found the system unnecessarily complex.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

3. I thought the system was easy to use.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

4. I think that I would need the support of a technical person to use this system.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

5. I found the various functions in this system were well integrated.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

6. I thought there was too much inconsistency in this system.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

7. I would imagine that most people would learn to use this system very quickly.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

8. I found the system very cumbersome to use.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

9. I felt very confident using the system.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

10. I needed to learn a lot of things before I could get going with this system.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

A. Appendix

A.3. System Usability Scale Results Detailed

Figure a.2 shows the detailed System Usability Scale results for each user. The questions can be found above in Section A.2

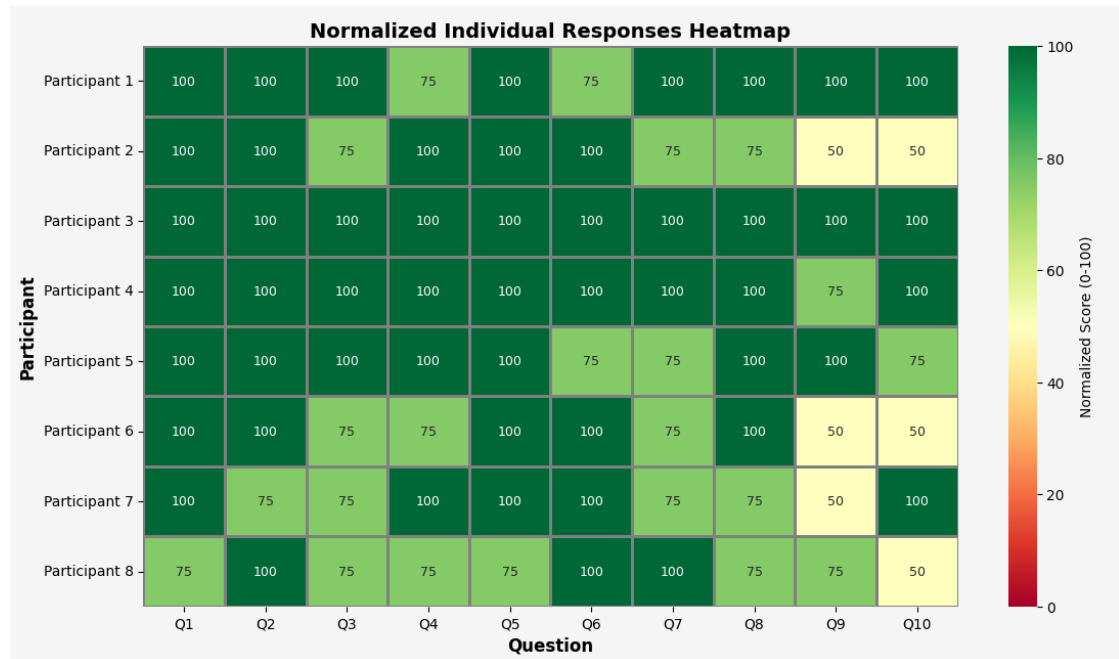


Figure a.2.: Normalized System Usability Scale score for each question and heatmap, where the best answer is 100 percent and worst is 0 percent and, 20 percent steps in between for better visualization.