MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

"Lodestar: Supporting exploration and discoveries of unknown star clusters"

verfasst von / submitted by

Johannes Özkan-Preisinger, BSc.

angestrebter akademischer Grad / in partial fulfillment of the requirements for the degree of

Master of Science (MSc)

Wien, 2023 / Vienna 2023

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 935
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Medieninformatik
Betreut von / Supervisor:	Univ.-Prof.Torsten Möller, PhD

Acknowledgements

I am grateful to the following individuals for their invaluable contributions to completing my master's thesis.

My supervisor, Torsten Möller, provided expert guidance and was always organized, responsive, patient, and able to clearly explain complex topics. He has always been an exemplary scientist and encourages and motivates students to give their best and strive for more through his mentorship.

I want to thank Sebastian Ratzenböck, who I worked closely with, for being kind, patient, and helpful and for his passion for astronomy. Without his time, effort, and insightful feedback, this work would not be what it is today. Time is the most valuable resource, and you have extended much of your time to me - Thank you.

I am grateful to the domain experts from the astronomy department. I want to thank Stefan, Laura, Alena, Josefa, Natalie, Cameren, and Fabian for taking the time to evaluate our tool, providing detailed feedback, and making me feel welcome.

Additionally, I would like to thank my two pet cats, Ginger and Paco, for providing a welcome distraction during long working days and working through critical feedback.

Most importantly, I want to express my deep gratitude to my wife, Ezgi, for her unwavering patience, support, and encouragement throughout my master's degree and the writing of this thesis. Without her encouragement, this thesis would not have been possible. I am grateful for her constant presence in my life and for the ways she has helped me complete my degree.

Contents

Abstract

In the search for new star clusters, astronomers are confronted with lots of hard unsupervised machine-learning challenges. Large data sets containing millions to billions of stars, star clusters hiding in a sea of noise making up over 95% of the data, and various sizes and densities are challenges that data mining tools and practitioners alike need to tackle. Despite these challenges, clustering remains a powerful tool for astronomical data analysis. Exploring the solution space and identifying optimal results in clustering methods can be difficult due to the presence of multiple parameters that are difficult to tune manually. Even when parameters are easy to understand, it is often challenging to estimate the impact of parameter changes. As a result, there is a need for a systematic approach to exploring the solution space and identifying optimal outcomes. We propose a novel visualization tool. "Lodestar" is designed to help explore the solution space created by the novel clustering algorithm pipeline SigMA to help understand the algorithm, explore its solution space and find clustering results of interest in the large set of solutions. Based on user test results, this tool is successfully finding suitable clustering solutions and is scheduled to be deployed in the lab of our collaborators to help identify stellar structures in the Milky Way.

Zusammenfassung

Bei der Suche nach neuen Sternhaufen stehen Astronomen vor zahlreichen Herausforderungen im Bereich des unbeaufsichtigten maschinellen Lernens. Die Verarbeitung großer Datensätze mit Millionen bis Milliarden von Sternen sowie Sternhaufen, die sich in einem Meer von Rauschen verstecken das über 95% der Daten ausmacht, stellen sowohl Data-Mining-Tools als auch Anwender vor große Herausforderungen. Verschiedene Größen und Dichten dieser Haufen sind weitere Schwierigkeiten, die es zu bewältigen gilt. Trotz dieser Herausforderungen ist Clustering nach wie vor ein effizientes Werkzeug für die Analyse astronomischer Daten. Die Erkundung des Lösungsraums und die Identifizierung optimaler Ergebnisse in Clustering-Methoden können jedoch schwierig sein, da mehrere Parameter vorhanden sind, die manuell nur schwer abzustimmen sind. Selbst wenn die Parameter einfach zu verstehen sind, ist es oft schwierig, die Auswirkungen von Parameteränderungen abzuschätzen. Daher ist ein systematischer Ansatz zur Erkundung des Lösungsraums und zur Ermittlung optimaler Ergebnisse erforderlich. Um diese Herausforderungen zu bewältigen, schlagen die Autoren ein neuartiges Visualisierungstool namens "Lodestar" vor. Dieses Tool soll dabei helfen, den Lösungsraum, der von der neuartigen Clustering-Algorithmus-Pipeline SigMA geschaffen wurde, zu erkunden, um den Algorithmus besser zu verstehen, seinen Lösungsraum zu erforschen und interessante Clustering-Ergebnisse in der großen Menge von Lösungen zu finden. Basierend auf den Ergebnissen von Benutzertests findet dieses Tool erfolgreich geeignete Clustering-Lösungen und soll im Labor der Astronomen eingesetzt werden, um bei der Identifizierung von Sternstrukturen in der Milchstraße zu helfen.

List of Tables

List of Figures

Chapter 1

Introduction

Stellar clusters, or star clusters, are groups of a few dozen up to a few thousand stars that were formed at the same time. The stars in these clusters are formed in gravitationally collapsing dense gas clouds. According to most recent theories of star formation, the vast majority of stars form in this process [50]. These stellar nurseries not only provide insight into the origin, age, and chemistry of stars but also offer a window into the fundamental processes of star evolution. To uncover star clusters, astronomers employ clustering, an unsupervised machine-learning approach that can automatically detect these groups in complex multi-dimensional data.

Identifying star clusters is typically framed as an unsupervised machine learning task, i.e. clustering problem, due to the lack of trustworthy labels and absence of good simulations. These clusters have to be extracted from a high-dimensional feature space consisting of more than 95% background noise that is composed of stars that have left the cluster. Further, clusters have almost arbitrary shapes, a wide variety of sizes, and densities. Not only does this pose a laborious clustering challenge, but it also makes validating results especially tricky. The variety of methods applied to the Gaia data [57, 42, 76, 1, 68, 32, 27, 29, 67, 46, 62, 14, 41, 66, 44] reflects this challenge very well. However, different clustering algorithms follow different clustering heuristics and objectives that can result in drastically different outcomes. Many clustering techniques come with a set of often unintuitive parameters which introduces an additional level of complexity for practitioners. Especially in high noise and tightly packed cluster environments, results vary widely between individual parameter realizations. Due to the often vast parameter space, an exploratory approach is necessary. Unsupported, this can lead to a long trial-and-error exploration.

In practice, we often see astronomers manually selecting parameters [73] or maximizing internal validation criteria based on cluster compactness and separation [59]. In manual parameter

selection, we find that practitioners are trapped in a slow feedback cycle that consists of parameter selection and visual result validation. Users might even revisit the same result multiple times as parameter details from previous runs are often forgotten. In most extreme cases, this becomes a Markov chain of random walks, in which the parameter values are estimated based solely on previous results. This aimless wandering through the parameter space is not only very inefficient but optimal values might not get detected as the dimensionality grows.

Parameter space exploration for unsupervised methods has been tackled extensively by the visualization community. Main contributions aim to offer tools that facilitate result exploration and generally aim to guide the user through the vast parameter space. Even though general visual clustering frameworks such as Clustrophile 2 [18], Clustrophile 1 [26], Clustervision [48], Uncover [60], Vista [20], DICON [16] offer mechanisms to guide users through result exploration and offer result validation, they do lack domain-specific information for result validation. Many of these tools include general-purpose evaluation techniques such as the Calinski-Harabaz index [15], Silhouette Coefficient [65], Davies-Bouldin index [25], Gap Statistic [72], SDbw validity index [39] that expand on so-called internal validation metrics [53] to aid in interpreting the outcome. However, they fail to incorporate additional quality criteria to validate a clustering performance (e.g., the Hertzsprung-Russel diagram (HRD) or space velocity information, discussed in section 4.2) to optimize the clustering or challenge validation measures. Especially in astronomy, domain experts can use positional and kinematic distribution or HRD information to perform qualitative validation which is a crucial part of the analysis pipeline.

To facilitate the search for stellar clusters in a principled manner, we plan to provide a visualization system that allows domain experts to perform result validation qualitatively by visually inspecting the result in the cluster space and additional feature spaces. To do that, an interface needs to quickly iterate over the possible solutions and provide complementary information about a clustering result, like the mentioned Hertzsprung-Russel diagram. This information is fundamental in making informed decisions quickly and should therefore be present in the visualization tool. With the launch of the GAIA space program [22] and its mission to measure accurate positions and motions of nearly two billion objects in the Milky Way, performance has also become a more immediate problem to tackle. Thus, when we consider usability in the context of astronomy, we also need to consider the data scale. The interface needs to be able to manage large quantities of data and do so quickly. For cluster searches, typical regions of interest contain from 500 000 to 10 million data points. To our knowledge, the previously mentioned general-purpose visual cluster support systems do not scale these data scales[1]. Thus, there is currently a lack of specialized domain-specific visualization tools to help

[1]For some tools no information on their performance was provided. In these cases, we made an estimated

guide experts to satisfying results that can handle millions of data points.

In this work, we put forward a design study intending to help astronomers find stellar clusters in the Milky Way. We present Lodestar, a novel visually assisted workflow for clustering in astronomy, offering an overview of an algorithm solution space and means to explore clustering results with the ability to drill down on individual solutions alongside qualitative validation tools such as the HRD. Using Lodestar, astronomers should be able to quickly analyze and validate possible clustering configurations and potentially discover novel star clusters.

The contributions can be summarized as follows:

- We present a novel visually assisted workflow for finding appropriate hyper-parameters to identify stellar groups in hard[2] clustering scenarios, see subsection 5.1.2, subsection 5.1.4 and subsection 5.1.5.

- Lodestar provides a complete summary of the multi-variate solution space by summarizing the impact of single-variate changes in a simple tree layout for a given parameter, making the solution space easy to understand and reason about a priori. This tree-based solution space summary provides a natural ordering and connections between individual solutions for a given parameter, making the solution space accessible and interpretable even before drilling into individual clustering results, see subsection 5.1.3.

- We introduce a methodology, analysis, and abstraction of data, tasks, and requirements for the star cluster domain, see chapter 4.

- We propose guidelines on precalculation opportunities to handle large data sets to guarantee near-immediate response times, see section 6.8.

In chapter 2, we will introduce the clustering algorithm used and the nature of the input data. We will look at state-of-the-art methods and techniques for visualizing clusters and solution space exploration, and address their research gap in more depth in chapter 3. In chapter 4 we discuss our design methodology, user feedback strategies, task analysis, and list the requirements. In chapter 5, we delve deeper into the user interface of our tool, providing a comprehensive examination of its features and functionality. We also explain the reasoning behind our design choices, highlighting the specific goals and objectives that guided our development process. Furthermore, we provide an in-depth discussion of the implementation details in chapter 6.

guess based on provided data examples and general interface design.

[2]Hard clustering refers to a type of clustering where each data point or object belongs exclusively to one cluster. In contrast to soft clustering (or fuzzy clustering), where each data point can belong to multiple clusters to varying degrees, hard clustering provides a clear and definitive membership for each data point. [7]

This includes a detailed description of the technical methods and techniques used to build and optimize the tool.

We have validated the tool in user tests and collected user feedback. We discuss these results in chapter 7 and go through lessons learned in chapter 8. Finally, we conclude our results in chapter 9.

Chapter 2

Algorithmic and Domain Background

In the following sections, we review stellar groups and their significance. We will discuss the intricate process of identifying and validating clusters in the field of astronomy, wherein we need to overcome several clustering challenges that demand significant domain expertise. Additionally, we discuss the clustering method at the core of this work, its parameters, and the structure of the solution space it produces. This chapter concludes with a summary of the output the tool will generate and how it will aid experts in judging a clustering result.

2.1 Domain Background

Star clusters are groups of stars that originate from the same formation process. Typical sizes of star clusters range between a dozen to thousands of individual stars. They provide valuable insights into galaxy formation, structure evolution, and stellar physics. In understanding the evolutionary process of cluster creation and development, astronomers can infer comprehensive theories of larger-scale systems.

The GAIA space program, launched by the European Space Agency in 2013, has been an incredible success in revolutionizing our understanding of star clusters. GAIA is a space telescope designed to survey the sky to create an unprecedented 3D map of the Milky Way galaxy, including positions, distances, and velocities of about 1.6 billion stars [21, 49, 30]. The highly precise measurements of position, proper motion[1], brightness, and parallaxes, provided by GAIA, have been crucial in identifying and analyzing star clusters, which are groups of stars

[1]The observed changes of a celestial body seen from the sun compared to more distant stars in the background [47], expressed in two measures, the direction of right ascension (α, deg) and the declination (δ, deg).

that are born and evolve together. These measurements have allowed scientists to study the properties and distribution of these clusters in greater detail than ever before, providing new insights into the formation and evolution of stars and galaxies. With the data from GAIA, scientists have been able to identify new clusters, observe the internal dynamics of clusters, and examine the properties of individual stars within clusters. Their common origin manifests itself in two observables: position and velocity. Thus, stars in stellar groups live together and move together [45]. Because we know that all members in the same cluster share similar positions and kinematic properties, we can identify star clusters as regions of increased density in a combined feature space of position and velocity, separated by areas of lower point density.

Therefore, researchers often study star clusters with density-based clustering techniques. Hunt and Reffert [41] points out that density-based methods (e.g., HDBSCAN by McInnes, Healy, and Astels [55] or DBSCAN by Ester, Kriegel, Sander, and Xu [28]) offer very high precision and sensitivity. Results show that HDBSCAN is the most effective algorithm for recovering open clusters in GAIA data sets. We will briefly discuss these two methods.

2.1.1 DBSCAN

DBSCAN is one of the most widely used density-based clustering algorithms, outside and even inside the astronomy community. It is a density-based clustering algorithm that finds clusters as connected components [2] of a graph. In this context, density is estimated via distances between data points. The algorithm differentiates between dense and sparse areas. Dense areas are labeled as clustered, while sparse areas are labeled as unclustered. That means that the algorithm supports the concept of noise.

To perform clustering, this algorithm requires two parameters, a threshold ϵ and a minimum amount of points m_{Pts}. Every distance between two points lower than the threshold ϵ is considered reachable. Every point that is part of a group with at least m_{Pts} points of members and where every member point is reachable to one another is considered a core point of that group. Finally, every point reachable by a core point is considered part of the same cluster.

The challenge of this algorithm is to find an appropriate ϵ threshold and a good value for m_{Pts}. This is a particular issue with the GAIA data sets because the density is highly variable within the data. In the paper by C. Boquan, E. D´Onghia, J. Alves, and A. Adamo [14], a different density-based approach is used, called Shared Nearest Neighbour (SNN). The study employs a two-stage approach, wherein the first stage SNN is run 7000 times with randomly generated parameters. In the second stage, the most repeating stellar groups are identified from

[2] A connected component is a group of nodes in a graph that are all reachable from each other but are not reachable from any other nodes outside of the group.

the first stage. In the paper by A. Castro-Ginard, C. Jordi, X. Luri, F. Julbe, M. Morvan, L. Balaguer-Núñez, and T. Cantat-Gaudin [1], manual tuning of parameters is used on a simulated dataset, the Tycho-Gaia Astrometric[3] Solution (TGAS) dataset[4]. The papers by V. Fürnkranz, S. Meingast, and J. Alves [73], S. Meingast, J. Alves, and A. Rottensteiner [66], and Zari, Brown, Bruijne, Manara, and Zeeuw [76] also employ manual parameter selection for their use of DBSCAN. The examples demonstrate that finding these parameters is non-trivial, and the approaches vary, but generally, a manual step for parameter selection is involved.

2.1.2 HDBSCAN

HDBSCAN is a density-based clustering algorithm based on DBSCAN but adds features such as automatically determining the optimal number of clusters and handling clusters of varying densities more effectively. It uses a hierarchical approach to identify clusters of varying densities, which makes it more effective at identifying groups of different shapes and sizes. This algorithm uses a stability criterion to identify stable clusters, which makes it more robust to noise and outliers. It also is less sensitive to the choice of parameters used for clustering, such as the distance metric and density threshold. Furthermore, it can identify clusters of different sizes and create a hierarchy between them, making it more effective for multi-scale clustering. HDBSCAN can automatically determine the number of groups.

The main contributions of HDBSCAN are a hierarchical clustering method that generates a complete density-based clustering hierarchy, allowing the extraction of a simplified version composed only of the most significant clusters. Additionally, HDBSCAN introduces a new measure of cluster stability to extract a set of stable clusters from possibly different levels of the simplified hierarchy. Subsequently, this method allows practitioners to extract stellar groups of different levels within the hierarchy to maximise stability.

HDBSCAN is applied in astronomy to identify clusters in large datasets of astronomical objects, such as stars and galaxies. In this context, HDBSCAN is used to identify groups of stars in a galaxy or groups of galaxies in large-scale structures like galaxy clusters and superclusters. Practitioners utilize the algorithm to identify structures that would be difficult to detect using other methods, such as visual inspection or traditional clustering algorithms. HDBSCAN was able to identify additional clusters that weren't identified by other methods, demonstrating

[3]Astrometric refers to the branch of astronomy that involves measuring the positions and motions of celestial objects.

[4]The TGAS dataset provides a higher precision astrometric solution than Gaia's first data release, as it combines positional data from the Tycho-2 catalog with the astrometric data from the Gaia satellite. This is, however, no longer true for the second and third Gaia data releases; TGAS is no longer used since Gaia DR2.

the ability of HDBSCAN to detect previously unknown structures in astronomical datasets. According to Hunt and Reffert [41], HDBSCAN does not detect all clusters in astronomical datasets. HDBSCAN found approximately 85-90% of the known stellar groups in the dataset.

2.2 Algorithmic Background

A cluster of stars has over-densities in motion and position features. The Significance Mode Analysis algorithm pipeline [61] (hereafter SigMA) is a clustering tool adjusted to astrometric data and specifically designed to identify star clusters. It aims to locate modes or over-densities in the data set corresponding to star clusters. A mode is an increase in density in relation to its surrounding and, by definition, separated from other modes by a region of low, or a dip in, density. Since statistical fluctuation can also produce random density dips in the density profile, SigMA's primary goal is to identify significantly deep dips, not caused by randomness. To identify these main dips (producing the over-densities as a by-product), SigMA iterates through the data set two times.

In the first iteration, initial clusters are formed by local gradient ascend strategies similar to Mean Shift [23]. During this iteration, SigMA also determines the location of dips between initial clusters (called modal regions). In a second iteration, SigMA loops through each pair of these neighboring modal regions and evaluates a statistical test that checks if the dip size is compatible with random fluctuations; in that case, they are merged. If the dip size cannot be explained by randomness, the clusters are kept separate and are not merged. A schematical depiction can be seen in Figure 2.1.

In this workflow, we have two primary hyperparameters. Experts can adjust each to tune the clustering behavior to their respective needs. For the typical astronomer, the interpretation of these parameters is, however, not straightforward. In practice, users are often found to be stuck in a tedious trial-and-error loop. In the following, we discuss SigMA's parameters and their role in the analysis pipeline. But even when parameters are easily understood, it can be difficult to predict the effect of changing them. This highlights the need for a systematic approach to exploring the solution space and identifying optimal results. Eventually, we aim to provide an overview and means to explore the parameter space spanned by the two parameters, density level and alpha value. We discuss each of them and their effects below.

Density Estimation Level

The SigMA pipeline employs a k nearest neighbor algorithm for the density estimation outlined in steps 2 and 3 of Figure 2.1. When we provide the level for the SigMA algorithm, we control the

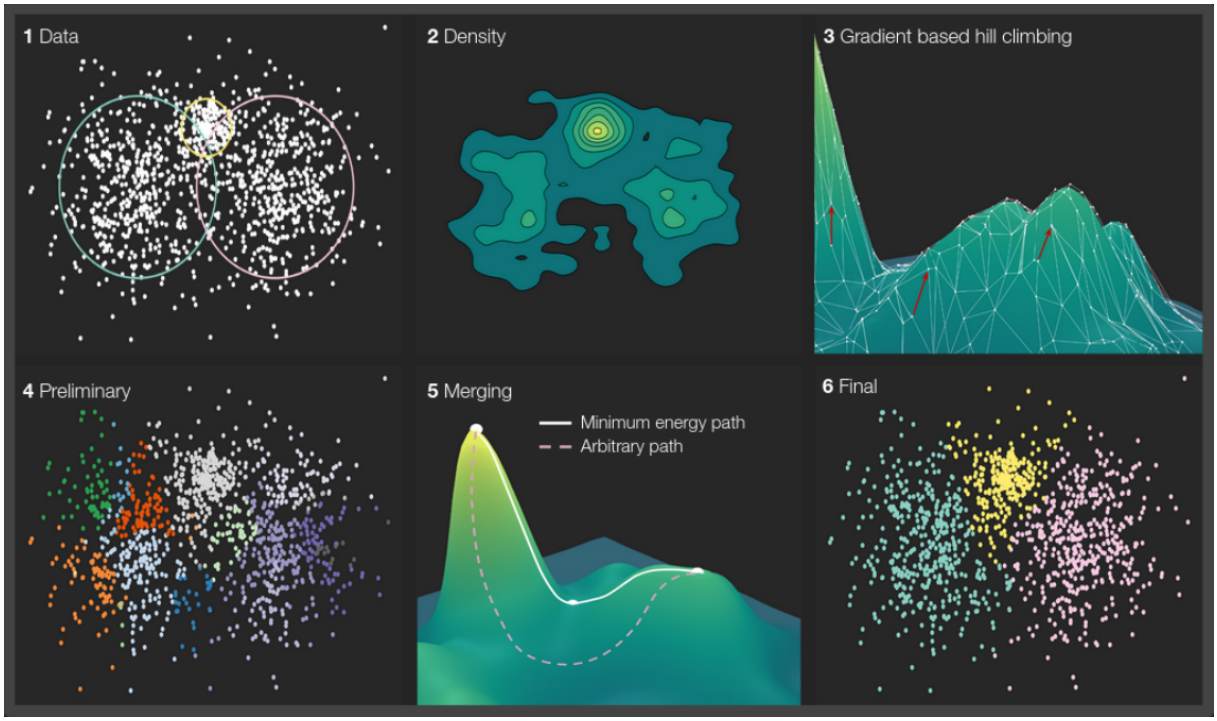


Figure 2.1: The proposed clustering process SigMA is highlighted on a 2D toy data set of three Gaussians with variable covariance matrices and means. (1) The generated toy data set consisting of three bivariate Gaussians is shown in white alongside 2 α confidence ellipses in color. (2) The clustering procedure starts by estimating the density of the input data. (3) Next, a graph-based hill climbing step is performed in which points are propagated along gradient lines towards local peaks. (4) This gradient propagation results in a preliminary segmentation of input samples which typically is far too fine-grained. (5) These segmented regions are iteratively merged with a parent mode if a modality test along the "minimum energy path" detects no significant density dip. (6) The final segmentation retains all three clusters. Used with the authorization of the author Ratzenböck, Großschedl, Möller, Alves, Bomze, and Meingast [61].

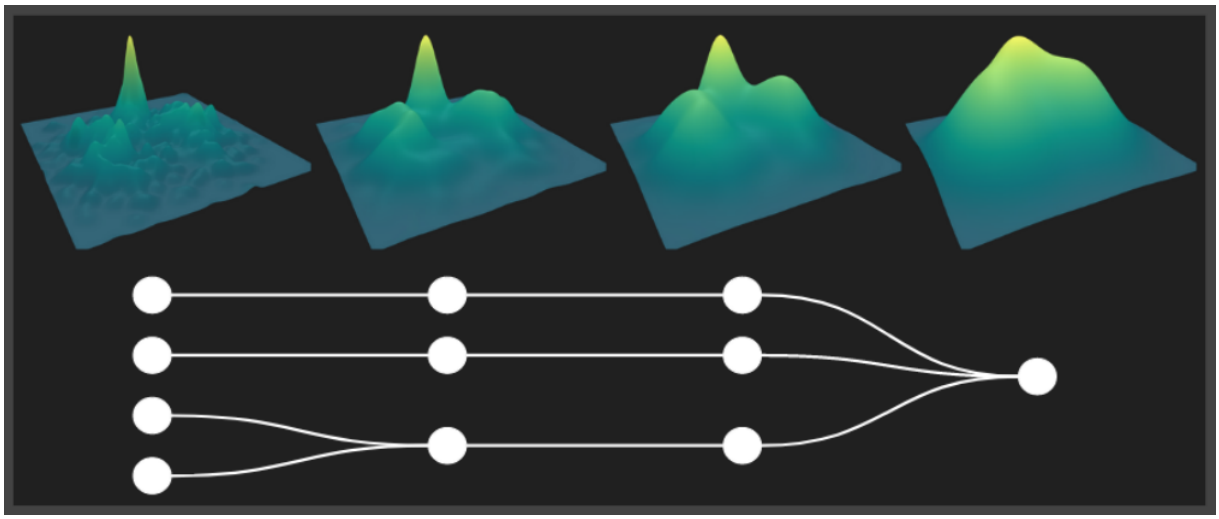


Figure 2.2: Used with the authorization of the author Ratzenböck, Großschedl, Möller, Alves, Bomze, and Meingast [61]. Increasing the level will create larger initial clusters and sacrifice less dense groups. The hierarchical representation of incremental level change is called a scale space, schematically represented as a tree.

number of k-nearest neighbors. The level value determines the size of the neighborhood around each data point used to estimate the density. A lower level value means a smaller neighborhood is used, resulting in a finer and smoother density estimate. On the other hand, a higher level value means that a larger neighborhood is used, which results in a coarser and more lumpy density estimate.

The choice of the level value will depend on the specific data and the desired trade-off between smoothness and level of detail in the density estimate. A lower level value may capture fine-grained structure in the data but may also be more susceptible to noise. On the other hand, a higher level value may be more robust to noise but may miss fine-grained structure in the data.

In practice, the analyst combines multiple labels into a single clustering solution using a consensus clustering approach which can easily ignore important minute details that can get lost by an ensemble approach. Figure 2.2 depicts the smoothing effect.

Alpha

For a given density level, each dip is assigned a p-value corresponding to the likelihood that the dip is associated with random fluctuations in the underlying density of a single modal distribution. If this value becomes smaller than a preselected significance threshold alpha, the null hypothesis of unimodality is rejected, leading to a cluster splitting. As the significance level decreases toward 0, the chance of a cluster splitting diminishes, leaving a single cluster. Thus,

the significance value alpha, as the density level, induces a hierarchy of clusters which can be represented by a tree in which the level represents clusters at a given constant value alpha.

If the p-value associated with the saddle point and adjacent modes pair exceeds this value, i.e., a significant dip is found, we consider the modes separate. In step 5 of Figure 2.1, we can see two paths, the minimum energy path (MEP) and an arbitrary path. The MEP contains a saddle point that we compare to a given alpha value. The effect of both parameters is hard to reason about a priori and has an indirect effect on the result. For users, it can be hard to understand how a change in either of these parameters translates to a clustering result. We attempt to bridge this disconnect by providing a tool to show the effects of these changes.

2.3 Data

The primary data source for this work is the already mentioned Gaia data. The Gaia mission's main contribution is precise measurements of the spatial position and velocity of over 1 billion stars in our Milky Way. These features are the primary input for the clustering algorithm SigMA.

The five dimensions used for the clustering technique consist of a three-dimensional spatial and two-dimensional velocity space. The result is improved if we provide an additional optional 6th dimension for sampling, the radial velocity. It will help in the noise reduction step. Our tool will therefore be able to support all six of the possible input features but will require only five to operate.

For this work, we will feed the SigMA algorithm with a range of predetermined, typical significance threshold values which represent a gradually more and more conservative clustering[5] and a range of density levels which we take from the original publication [61]. Each increment in the density level will introduce the smoothing factor highlighted in Figure 2.2 and provide the initial cluster modes. With every increment in the alpha value, we introduce a different threshold for significant dips in the mode merging step in step 5 of Figure 2.1. Each combination will produce a unique clustering result. We can imagine each alpha value produces a complete scale space (i.e., a hierarchy of clusters); see Figure 2.2. The configured alpha value and density level ranges result from exploratory testing during algorithm development and are considered ranges that produce clustering results of interest. The algorithm developer performed these tests with Jupyter Notebooks[6] on Gaia data. We designed our tool to make this solution space available

[5] We determined statistical significance by using a significance level (alpha) of 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, and 1/128.

[6] https://jupyter.org/

for exploration.

2.3.1 Input

The dataset contains n points $P = \{p_1, p_2, \ldots, p_n\}$, where each point constitutes a star with k measurements $p_i = \{m_1, m_2, \ldots, m_k\}$. The features of interest are position, velocity, and brightness.

Spatial and velocity information can come in different coordinate systems. Lodestar can handle various cluster spaces and takes care of the needs of different user groups. In addition, the algorithm also supports clustering with an additional velocity measurement: the radial velocity and its statistical uncertainty. The calculations with the radial velocity are optional since radial velocity is rarely available for a star - only about 5% of stars in the Gaia data set have this information.

Therefore, the algorithm pipeline requires three spatial input parameters and at least two velocity inputs to pre-compute solutions for that feature space within a range of configured alpha values and density levels. Lodestar will use a default setting of specific alpha value and density level for the initial presentation. Afterward, researchers can change the current values within the range and retrieve the affiliated pre-calculated cluster result.

2.3.2 Output

This tool will leverage the SigMA algorithm pipeline to cluster for each unique pair of alpha value and density level configuration and produce a file containing a column for the label and a column for each feature used for clustering. The label stands for the assigned cluster, and a label of -1 represents a noise label.

The tool computes the files as soon as the user has selected the five plus one features for clustering. Common waiting times for the precalculation steps range between one and five hours for typical data set sizes. This is done purposely and will allow for a quick exploration of the individual results. To support a graphical representation of the set of all level trees, we also store a hierarchical tree for each alpha value increment. Each level in the hierarchy is an increment in the density level range and represents a clustering result. This allows us also quickly to alternate between density level or alpha value selection.

2.4 Validation

We can retrieve the previously mentioned label files on-demand and present groups in a 3D plot for the spatial dimensions and a 2D plot for the velocity dimensions to validate the results.

Users can retrieve additional data for each result to validate the correctness of the clustering. They might choose to select a specific cluster and can request every feature that is contained in the original file, filtered by the label of the selected cluster. Moreover, the selected feature can be viewed as a distribution within that cluster, a histogram plot.

In the subsequent chapter, we will review the current state-of-the-art tools and discuss if they address this specific astronomical clustering challenge adequately. To manage the amount of data and to be able to find unexpected or significant relationships, it is necessary to provide visualization techniques to make this data more accessible to varying stakeholders in the astronomical community.

Chapter 3

Related Work

This chapter overviews currently available visualizations for topological and multi-variant data and compares their advantages and disadvantages. All tools considered should permit parameter space exploration. Even though there are many methods to approach this topic, each of them retains some limitations. Subsequently, we discuss how our tool might improve upon currently available solutions. To help us discuss if a tool meets the requirements of solution exploration in astronomy clustering, we first want to discuss the requirements it should meet (see section 3.1). Afterward, we will review the relevant literature on parameter space exploration (see section 3.2), general clustering tools (see subsection 3.2.1), and the literature that influenced our work the most (see subsection 3.2.2).

3.1 Requirements for Clustering in Astronomy

Star clusters can vary greatly in shape, density, and size, and the background noise is not uniform. This poses a difficult challenge for clustering methods, as they must be able to adapt to the complexity of the data. The vast amount of data and complex morphologies of star groups make it difficult to find the optimal algorithm parameters. To overcome this, it is necessary to explore the space of all solutions within a given parameter space. Each set of parameters can result in vastly different clustering results, making the clustering process dependent on input parameters and the results not robust. A single set of parameters may not reveal the optimal solution, and a combination of multiple solutions may be required. To address this, we will compare the results of different parameter sets, select specific clusters, and track their development with changing parameters. This will allow us to study the characteristics and details of the selection to determine if a group is relevant. In conclusion, the user needs to be able to explore the different solutions that an algorithm can provide.

Clustering without ground truth is often done through empirical research, relying on observation and measurement. To ensure accurate results, users must validate each outcome and determine its relevance. To assist in this process, we must utilize visualization techniques that are both guided and intuitive, allowing for the analysis of the solution space generated by the clustering algorithm. These techniques should be easy to use and provide users with feedback on how parameter changes affect the clustering outcome. Domain-specific information, such as radial velocity histograms or HRDs, can also be used to evaluate the quality of a cluster and aid in decision-making.

As pointed out in chapter 2, managing large quantities of data is not only an algorithmic problem but also a visualization problem. Both can incur waiting times that are unacceptable to the user and use up more memory than is available or sustainable. If the literature sources discuss information on run-time and memory complexity, we will discuss potential limitations. This is important because a visualization tool could use additional algorithms to prepare the data for presentation, and its run-time performance could be worse than the actual clustering algorithm. In this case, the run-time of the clustering algorithm, instead of being a limiting factor, would become a negligible factor.

While evaluating a clustering result, the spatial and velocity features provide important visual feedback for judging the quality of a clustering result. Astronomers tend to visualize space in a 3D plot. This allows domain experts to identify structures hidden by other, possibly larger structures. Zooming and panning will additionally support the search for hidden structures. Based on these facts, we identify a need for a 3D representation of the space properties – aided by actions such as zooming and panning to confirm the correctness of a clustering result.

Any tool or visualization used for clustering in astronomy must be able to handle noise effectively, as the GAIA data sets often contain a significant number of field stars that do not contribute meaningful information to the clustering process. These stars are not members of any cluster and are considered noise. Furthermore, it can result in the identification of many small stellar groups, and the number of groups is not fixed. Therefore, the tool must be able to scale to handle tens and even hundreds of clusters, not only from a computational perspective but also from a visual one. This large amount of data can lead to many visual objects, which can cause cluttering, performance issues, and information overload. It's important to keep this in mind when designing the tool.

In the following sections, we aim to review the literature on visual parameter space analysis tools. We want to highlight a research gap in large-scale, complex clustering challenges. Because every domain has its challenges, we are looking at the following visualization tools from the

perspective of an astronomy data scientist. A comprehensive tool is needed to effectively analyze the multi-variate solution space in astronomy, including features such as HRD, velocity, and 3D scatter plots. An effective visualization should enable efficient navigation of the parameter space by highlighting regions of interest and providing interactive tools for further exploration. Additionally, the visualization should be able to handle large datasets, potentially ranging from 500,000 to 10 million data points, while maintaining a runtime performance of less than $O(n^2)$.

3.2 Parameter Space Exploration

Research on multidimensional data and multiple inputs, such as Vismon by Booshehrian, Möller, Peterman, and Munzner [8], has inspired research on parameter space exploration and optimization guided by expert assessment. The challenge of parameter space exploration is an ongoing topic. A subdomain of this research is computational steering addressed by Bruckner and Möller [12] and with the tool World-Lines by Waser, Fuchs, Ribičič, Schindler, Blöschl, and Gröller [74]. World-Lines allows making optimizations to an ongoing simulation while it is running. This research on parameter space exploration has advanced, as exemplified by the development of tools such as FluidExplorer by Bruckner and Möller [12]. FluidExplorer is a work that focuses on simulation output optimization based on temporal behavior. In Tuner by T. Torsney-Weir, A. Saad, T. Moller, H.C. Hege, B. Weber, J.M. Verbavatz, and S. Bergner [69], the goal was parameter-finding in image segmentation. This tool is designated to aid developers in finding optimal parameters for a wide array of algorithms. Tuner uses a ground truth for algorithm optimization. The work by Brochu, Brochu, and Freitas [9] requires no ground truth or objective function. It proposes a new method for procedurally designing animations using a Bayesian interactive optimization approach for researchers and practitioners in computer graphics and animation. Paramorama by Pretorius, Bray, Carpenter, and Ruddle [58] does not require any ground truth as well and lets users define areas of interest and provides tools, such as tagging, to refine and validate their results. It is based on the layout algorithm and focuses on image analysis algorithms. ParaGlide by Bergner, Sedlmair, Möller, Abdolyousefi, and Saad [6] provides visual aids to assist users in making decisions by dividing large multidimensional data sets, such as those with eight dimensions, into smaller, more manageable groups. This allows researchers to partition the parameter space and more easily explore individual subspaces, leading to a deeper understanding of the data and improved decision-making.

3.2.1 General Clustering Approaches

The work discussed by Chen and Liu [20] provides a visual framework named VISTA, developed with the understanding that human interaction is an important factor in clustering. It is designed with the concept in mind that clustering is not finished without human interaction, understanding of the result, and acceptance of the pattern applied. The researchers discuss how visual representations can help to detect trends, spot outliers, and visualize detected clusters and additionally emphasize the importance of run-time performance. Further, it is implied that human-computer interaction usually takes more time than automated algorithms but that the interaction enables users to produce improved results and higher confidence in these results.

Their work provides a visual framework developed to work with a body of different clustering algorithms. It presents the processed data in a way that is independent of the underlying algorithm, but users can still contribute to the clustering result and refine it. The visual framework can work with the results of different algorithms and attempts to help the domain users to interact with the results and refine them.

The downside of this approach is that the visualization is disjoint from the applied algorithm, and the only way to reason about the algorithm is to interact with its clustering result. Each adjustment incurs a restart of the complete clustering process. This produces some effort to perform exploration of the solution space and makes it hard to compare the two results. We also find that in their work, there is no mention of how this tool can identify or handle noise. While the flexibility such a generic tool offers is very helpful in comparing the capabilities and run-time performances of different algorithms, it might restrict domain users from understanding or following the inner workings of a specific algorithm and critically questioning their outcomes. Such generic frameworks often lack domain-specific knowledge and do not serve domain-specific needs. These are crucial components in steering, monitoring, and refining the results. In addition to these concerns, it is also unclear how the tool performs with data sets larger than 50.000 items, such tests have not been mentioned nor demonstrated.

Similar to that, Clustervision by Kwon, Eysenbach, Verma, Ng, Filippi, Stewart, and Perer [48] is a visual analytics tool that helps sample the space of possible solutions by considering multiple different clustering techniques and various input parameters to provide means to explore these solutions. Its goal is to provide a domain-independent comparison of clustering results using quality criteria for ranking. The tool ranks different clustering results by sorting according to five statistical quality metrics, Calinski-Harabaz index [15], Silhouette Coefficient [65], Davies-Bouldin index [25], the S_{Dbw} Validity index [39], and Gap Statistic [72]. It selects the top 3 highest-ranking results in each metric ranking and presents the user with the top 15 items. This

tool allows the user to compare different results by exploring them with multiple visual tools, such as comparing different clustering results and inspection of clusters. Techniques that help to reflect on the clustering result and should help to improve the overall clustering outcome. This work uses meta clustering, an approach where the algorithms perform many different clustering runs. Users can compare these results based on their specific requirement [17]. We think that the investigation of the solution space is well supported but lacks visual aids to reason about the results from a domain-specific point of view. Since there is a higher emphasis on quality ranking and less emphasis on the solution space exploration itself, we believe that our work is different in this case. Their work does not demonstrate that it can work on large data sets, and due to its domain independence, the visualization does not support domain-specific information.

DICON [16] is a tool designed to interpret, evaluate, and compare multidimensional data sets to visualize and reason about the quality of clusters. It offers a treemap-like icon representation of the multidimensional cluster, allowing for easy evaluation and comparison by looking at the embedded statistical information within the icon's shape. Additionally, DICON allows for clustering results to be refined via user interactions by manipulating the icon's shape and influencing the clustering result. However, this tool does not allow for comparing different clustering results and lacks domain-specific visualizations for validation. This work did not perform tests on data sets commonly used in astronomy. Furthermore, no information is provided on how the tool deals with noise.

Cluster Sculptor [13] is interactive, designed to help users apply their domain-specific knowledge by iteratively updating the cluster labels of a data set in a semi-automatic way. It does so by employing a semi-automatic update mechanism for the cluster labels. A user manually selects a few representative data items, also referred to as *seeds*, used to refine the overall clusters. This technique is employed to minimize manual user input in cluster refinement. Both the updates on the labels and the representation of the clustering results are done on a 2D projection of the multidimensional data by employing the t-Distributed Stochastic Neighbor Embedding technique [54].

While the feature of semi-automatic cluster refinement could help refine clustering results in astronomy, like VISTA, Cluster Sculptor is disjoint from the actual clustering algorithm. It offers user interactions simply on the provided labels. This introduces the same issues that we identified within VISTA. The exploration of different results with varying input parameters is currently not supported, and we did not identify visual aids to judge the quality of clusters. In addition, Cluster Sculptor has a runtime complexity of $O(n^2)$.

The work of Cavallo and Demiralp [18] builds upon the research conducted by Demiralp [26]

by introducing Clustrophile, an interactive tool for guided exploratory cluster analysis. This tool addresses a variety of tasks, including the ability to show variation within clusters, allowing for quick iteration over parameters, promoting multiscale exploration, compactly representing clustering instances, facilitating interpretable naming, supporting reasoning about clusters and clustering instances, guiding users in clustering analysis, supporting the analysis of large datasets, and promoting reproducibility. Clustrophile utilizes multiple validation metrics to perform quantitative validation and allows for visual exploration of the clustering space through the use of an undirected decision graph. It also incorporates user feedback and preferences in ranking the cluster results. The tool focuses on user-guided exploration, providing textual information and links to external resources for additional information. Additionally, the tool offers automated suggestions based on the current dataset, user preferences, and previous computations, such as recommending similar results to the user based on the total number of likes. To handle large-scale data sets, Clustrophile employs techniques such as caching and precomputations in the computation phase and uses panning, zooming, and visual grouping with the help of convex hulls to combat visual clutter and enable users to handle large-scale data sets. The tool also allows for the logging of user interactions and the ability to undo or redo them, enabling users to continue a session at any time and reproduce complex interactions. Many features directly influenced our tool, especially the approach and focus of a user-guided tour through the solution space are central to our work.

Despite its advantages, the authors did not test the tool on data sets with data entries up to several thousand or several hundred thousand members, and the visual layout presented in Figure 3.1 shows that the space left for cluster visualization can quickly become cluttered while analyzing large data sets. The tool does not offer different perspectives on the clustering result, for example, an array of 2D scatter plots or a 3D plot, to leverage spatial information and detect groups covered by other, larger clusters. Though the underlying DBSCAN algorithm supports the concept of noise, it is not addressed in the visualization. The option to remove noise, for example, would be a necessity in astronomical data. In this sense, the tool is missing domain-specific hints and language. It is harder to get started and apply it to domain-specific problems.

In addition, we have drawn from the work of Gerber, Bremer, Pascucci, and Whitaker [33] and Gortler, Schulz, Weiskopf, and Deussen [37] for inspiration on clustering and how to visualize it.

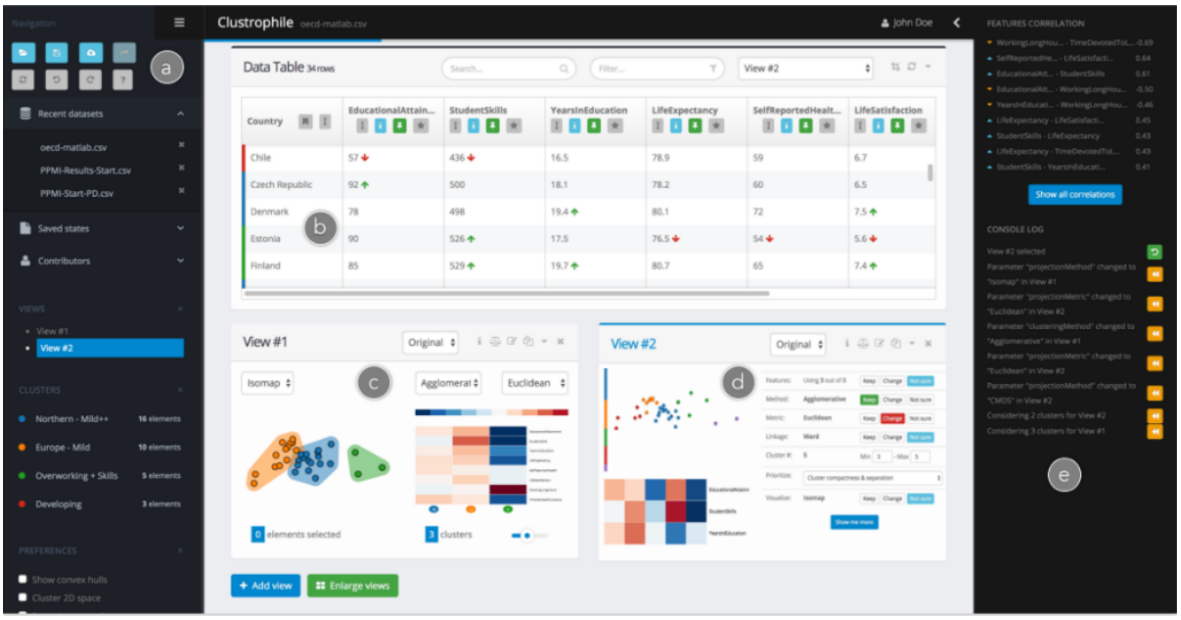


Figure 3.1: Main screen of Clustrophile [18]

3.2.2 Specialized Approaches

Connecting related heterogeneous datasets has become a priority in astronomy as data volume and diversity increase rapidly. There is a growing open-source and collaborative environment in astronomy. An example of this is Glue [64], a platform that hosts a large variety of data and numerous open-source modular software. The Glue environment, by Beaumont, Goodman, and Greenfield [4], allows users to load multiple datasets and "glue" the related attributes together from different data types.

TOPCAT[1] [71] supported by STILTS [70] is a flexible and highly configurable graphical interface for tabular data. It combines many of the advantages of a general-purpose tool and a specialized tool. Since it is predominantly used in astronomy, has many astronomy-specific features, and focuses on tabular data, it can serve many of the astronomy domain-specific validation techniques. It does support exploring the cluster-specific distribution of selected features, exploring clusters in a 3D or 2D plot with selectable columns and can provide quick response times for large volumes of data. Unfortunately, this tool cannot perform clustering or labeling by itself and has no means to explore a multivariate solution space. But we highly appreciate this tool as a validation tool of its own to improve the development of our work and to cross-validate results from a different perspective. Because our tool supports the download of a selected result and the ease with which you can upload that file in TOPCAT, we think it is a great tool to be used in combination with Lodestar.

[1]TOPCAT: Tool for Operations on Catalogues And Tables http://www.star.bris.ac.uk/~mbt/topcat/

During our review process, we identified a lack of appropriate visualization tools that meet all criteria. Consequently, we propose our solution to tackle these problems.

Chapter 4

Methodology

The iterative design process, schematically illustrated in Figure 4.1, follows a task analysis (see section 4.2), an initial brainstorming and low-fidelity prototyping phase using the Five Design Sheet Methodology (FDSM, see section 4.4), a high-fidelity prototyping and feedback step (see section 4.5), and a final implementation phase with a usability questionnaire (see section 4.6).

The main focus of this thesis was to create a visualization tool tailored to the needs of the astronomy community, with which domain experts can interpret and explore the results of a novel algorithm pipeline - SigMA [61].

To design an effective tool, we began by defining its main tasks. Well-stated tasks are an essential part of a data visualization tool's design and implementation. They provide clarity and direction, which reduces ambiguity and enhances accountability for the work that needs to be done. These tasks were created in a collaborative effort involving the stakeholders, such as the author of SigMA, experts from the astronomy department, and a visual design expert.

First, we focused on identifying user's goals to determine what they needed to achieve with the tool. We then addressed tasks related to data integration and management, including how to handle data import and manage data types and formats. Next, we defined operations for data transformation, such as filtering, sorting, and aggregating data. This was followed by designing data representation, allowing users to select the type of graph or chart, colors and labels.

We also defined tasks related to user interaction, such as zooming, panning, and selecting. Furthermore, we discussed operations such as annotating visualizations and creating exports of the clustered data.

In terms of scalability and performance, we ensured that our tasks don't prohibit our tool to perform well with large datasets. To allow for customization, we made sure that users could interact with the visualization according to astronomers needs.

Usability and user experience were also paramount, so we sought to enhance these aspects

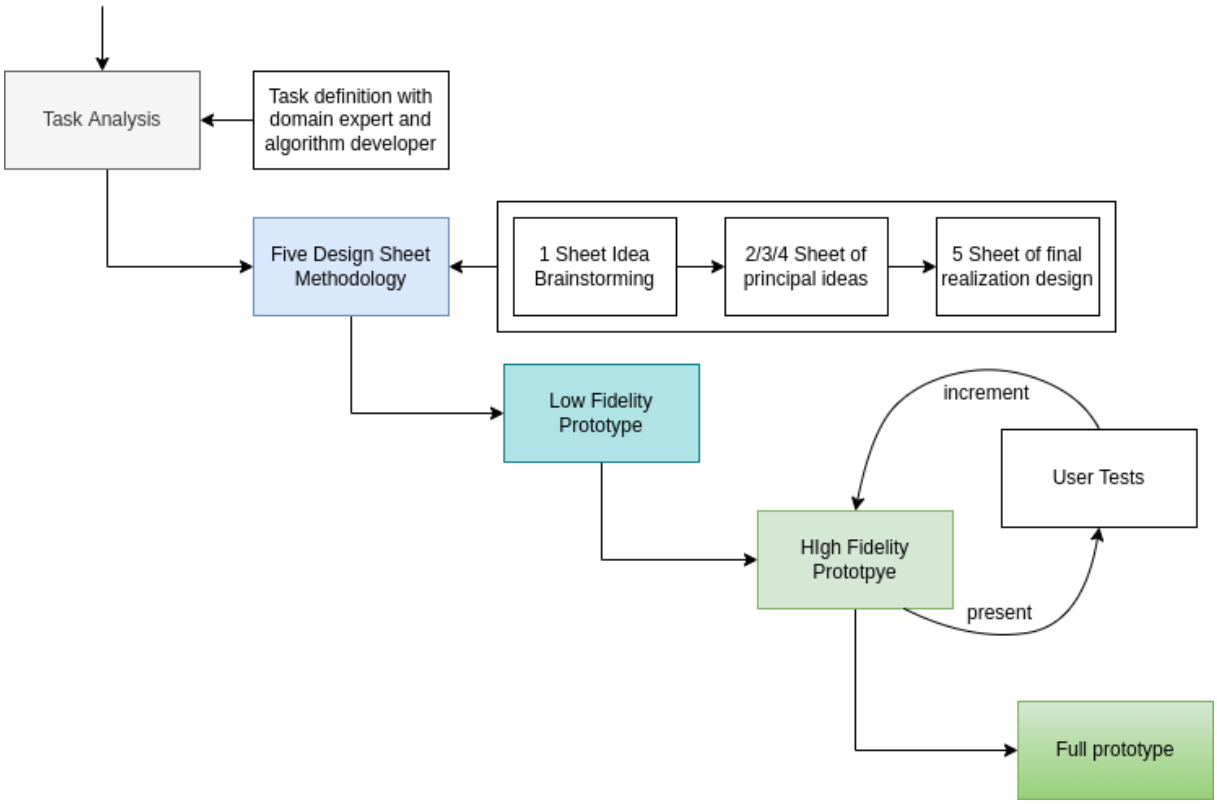


Figure 4.1: Depicted is the overall methodology broken down into distinct steps. One performed after another. Each step in itself contained several sub-steps. The depiction includes the most important.

through verbal verification [1] of the tasks. Finally, we planned for maintaining and updating the tool as our stakeholders needs evolved. These tasks were clearly stated, prioritized, and communicated.

As a result, we have defined the tasks discussed in more detail in section 4.3. Afterwards, we applied the five-design sheet methodology to bring them to life. We worked closely with domain experts, a visualization expert, and the creator of the SigMA algorithm to ensure that our low-fidelity prototype accurately reflected their needs and expectations. After presenting the prototype for review, we met with the expert group to gather their feedback and used it to refine the design. We then created a high-fidelity prototype, which served as the basis for individual user interviews. By carefully considering the comments and suggestions of each participant, we were able to identify the most pressing requests and use them to improve the tool further. Throughout the process, we remained dedicated to collaborating with experts and incorporating their insights to create a product that meets the needs of our target audience.

In each step shown in Figure 4.1, we worked closely with domain experts. As seen in the schematic overview, we consulted experts to verify that each step was complete and that we aligned with their expectations. In the high-fidelity phase, we used an iterative refinement cycle, after which the domain experts served as quality gatekeepers. Their feedback served directly as input for the next cycle. Therefore, we constantly exchanged with the target audience and the SigMA creator. This domain-oriented and iterative approach ensured that our design was flexible and could adapt to changing circumstances. As soon as the design or implementation diverged from expectations or the expectations changed, we were able to react immediately. In the early stages of this work, the algorithm pipeline was still in development, and our tool had to evolve with it. In the later stages, we used valuable interview feedback to introduce unexpected features and make changes to the visualization.

4.1 Test User Group and Domain Experts

During this work, we collaborated closely with a selected group of astronomers. They not only provided important feedback during the five design sheet methodology but also during the development of the tool. For the low-fidelity tests, the group consisted of six people, four women, and two men. During the high-fidelity tests, the group consisted of eight members, four men, and four women, all of whom were students or research personnel at the Department of Astrophysics at the University of Vienna. One member of this group was the developer of the algorithm described in chapter 2 and was constantly involved. It was a group of academics at

[1] In this method, we spoke out loud our thought process as we performed a task.

different stages in their academic career. Notably, one member had a color vision deficiency which provided valuable insights for the design of the tool and for further discussion, and we derived helpful lessons from that fact. Due to their domain know-how, they served as domain experts and represented the target audience.

The SigMA creator is an expert in physical data analysis and visualization. The visualization expert has a background in Visualization, Computer Graphics, Image Processing, and Data Science.

4.2 Task Analysis

Analyzing user needs is the most challenging part of creating a proper visualization tool. It helps to understand our target audience and is required since users can have vastly different expectations depending on their profession and educational background. Therefore it is paramount to properly study the user's needs and tasks before starting the first concrete drafts. The book by Munzner [56] provided great insight into the topic of task analysis and guided us to help answer the following questions:

- Who is the target audience? Who will either consume or produce data?
- Why are we creating this tool? Which questions does it answer?
- What choices or switches will the tool offer a user to influence the outcome?
- Which actions should be allowed? What will be their results?

4.2.1 Who?

Defining a target audience was very helpful. It provided the necessary direction for decisions. We should base our decisions on the person that will use our tool and drive design choices. Therefore a proper definition of a target audience was necessary for a thorough task analysis. It helped to observe the target audience in action, to understand how they perform their tasks and achieve their goals.

Our target audience was the astronomy department at the University of Vienna. This tool provides an entry point to explore and understand the algorithm discussed in Chapter 2, and our users explore the solution space this algorithm produces. In collaboration with domain experts, we defined several necessary views for the user to perform their tasks.

It turned out that two overarching tasks will drive the development of the application and should be supported. Firstly, a potential user should be able to get familiar with the purpose

of the algorithm. Secondly, the user should be able to change these parameters in the tool to explore the potential solution space. We discussed these parameters in depth in section 2.2. The changes users make should propagate through all the controls and views with which the user interacts. Since we worked with stellar data, the application should handle large amounts of data and thousands of stars and still perform well. To validate if the clustering solutions are satisfying, an astronomer would like to receive feedback immediately, so run-time performance was a high priority. Furthermore, astronomers wanted to explore the distribution of different features and how the clustering looks in space, velocity, or on an Hertzsprung-Russell diagram. Because they often use the results for further processing in other tools, they also wanted to export the clustering result into a file and download it.

Further, the users do not have a complete understanding of the algorithm. The tool must communicate the presented data well, with additional text information, an introductory presentation, or a tutorial.

4.2.2 Why?

If your tool cannot provide value to users, then why should it even exist? Therefore it was necessary to outline the purpose of our tool and which problem it solves.

We have summarized the purpose as such:

> Provide astronomers with a tool to explore and find new stellar clusters within the clustering results of a novel clustering algorithm.

Because we are dealing with a recently published algorithm pipeline, new visualization methods might be necessary. Therefore, a careful and abstract analysis of the tasks was mandatory. Guided by that, we identified the abstract tasks listed in Section 4.3.

4.2.3 What?

When it was clear why and for whom we wanted to create a tool and which tasks it should be able to perform, it was necessary to outline what kind of choices and actions were required. The recommendation is to do this analysis on three different levels:

1. How is our tool used to analyze? Is data consumed or additional data created?
2. What kind of search is involved?
3. What kind of query?

Level 1: How is the tool used to analyze? Is data consumed or additional data created?

To answer this, we needed to address if users only consume data or if they produce data during a user session. Given a parameter configuration, astronomers will primarily use the tool to read the algorithm output - a label allocated to each star representing the assigned cluster. The only exception to this is the customization of the cluster name. Because it can be hard to track them solely by their numerical ID, users wanted to be able to assign an alphanumerical name. Our tool adds additional columns to the original data for the assigned labels and user customization. The modified file is available for download in the application.

Level 2: What kind of search is involved?

We designed the tool to introduce a novel algorithm and allow users to get familiar with this algorithm by exploring the different solutions created. No search is required, but we intended to provide enough information to help the users search for what they need. We wanted to accomplish this by immediately showing the effects of changing one of the two parameters and displaying these results from multiple angles.

Level 3: What kind of query?

Astronomers have criteria by which they would like to judge a clustering. It turned out that they would like to view the complete clustering at once, pick out a specific cluster and look at the cluster's detailed information, such as the Hertzsprung-Russell diagram, velocity histograms, and the clustering in a space and a velocity plot. In addition, the domain users wanted to explore the relationship of features in a scatter plot. Each axis needed to be configurable to compare different features. To support the exploratory nature of this tool, we also intended to plot the development of a cluster with changing parameters. We used development graphs for that purpose (see chapter 5).

4.2.4 Which Actions?

As discussed in section 2.3, we can have different variations of possible input data for the algorithm, meaning that there is no predefined name or positional information about the required features within the tabular data set. Users wanted to choose them on demand from the available columns in the data set. Additionally, they also wanted to explore clusters and their information in isolation. Therefore we also support inspecting a selected stellar group or a subset of all groups. For that reason, we allow the users to disable a range of clusters. Finally, we have also learned

that astronomers want to judge a clustering in the context of the overall data set - including noise. Therefore, we added the action of toggling the visibility of the surrounding noise on or off.

4.3 Tasks

Following the methodology outlined in section 4.2, we met with the algorithm developer and visualization experts to discuss the application's workflow, schematically depicted in Figure 4.2, and derived abstract tasks that addressed the needs of the target group and provided solutions for algorithm introduction, solution space exploration, and cluster validation. We then consulted with representatives from the astronomy department and added sub-tasks to provide more detailed descriptions of the requirements for each task. The definition of sub-tasks helped to clarify the goals of each task and eliminate confusion. Some tasks were modified or adapted during the course of this work; these changes and their rationale will be discussed in detail in chapter 7.

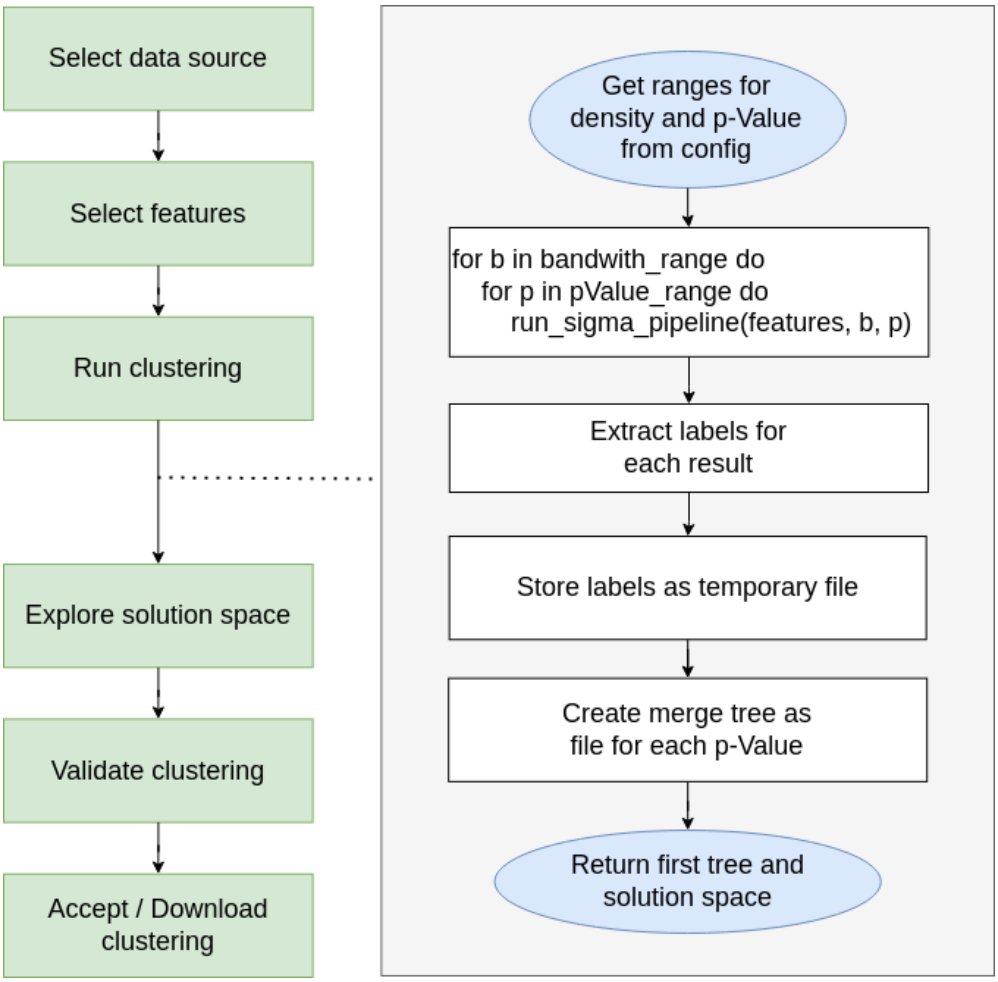


Figure 4.2: Schematical depiction of the primary workflow of Lodestar.

Task 1 : Provide an introductory part showcasing the algorithm and examples of the parameter influence on the output. Due to the complexity of the algorithm, novice users should get an initial impression and "feel" of the inputs and outputs of the algorithm. This should curb the initial steep learning curve.

Task 2 : Provide an overview and allow for quick iteration over parameters: A user should be able to quickly get an overview of the clustering solution space and be able to navigate it efficiently.

1. Provide an overview of different runs (different parameter sets).
2. Guide users through possible solution "groups"[2] of the algorithm. Provide means of summarizing and navigating through the clustering solution space.

Task 3 : Study the clustering output of the algorithm and provide multi-scale exploration: The resulting clusters constitute a hierarchy from coarse to fine structures. The user should be able to explore this hierarchy at multiple levels.

1. Explore the created clusters.
2. Study the cluster sizes (referring to the number of members in the cluster). The sizes provide an overview for details on demand. These clusters can then be visually validated.
3. Find clusters contained within other clusters.
   (a) Zoom in and out into regions of interest.
4. Compare the results of two different values for the parameter alpha.
5. Identify significant clusters.
   (a) The algorithm may mark a cluster or extreme value as significant.
   (b) Highlighted significant clusters or extremes will be marked as such.
   (c) The marked cluster will remain highlighted until the significance attribute disappears. This might happen if the result of the algorithm changes due to user interaction.

Task 4 : Extract stellar cluster from surrounding noise.

1. Highlight the extracted cluster in position, velocity, and Hertzsprung–Russell diagram for validation.

[2] A solution group refers to multiple related solutions. Solutions can be grouped by alpha or density value.

4.4 Low-Fidelity Prototyping

This work attempted to approach working solutions by using the Five Design-Sheet methodologies, introduced by Roberts, Headleand, and Ritsos [63]. We selected this method because it provides a formalized and structured approach to low-fidelity prototyping, a means of finding solutions with stakeholders with low financial or resource impact. It allowed us to explore, filter, and refine ideas before answering technical questions. Thus, we developed ideas without any technical restrictions.

Based on the results from the task abstraction, we started to work on a low-fidelity prototype. Because we wanted to find solutions for a novel algorithm and to approach a prototype in iterations, we decided to use pen and paper [51]. Only for a few, rather involved, workflows, we used Adobe XD [3]. It allowed us to experiment with different 3D visual encodings. This tool also proved valuable in working and testing navigation workflows. To brainstorm about specific tasks, we researched currently available solutions for similar problems. As Munzner [56, p.12] stated:

> First, there is the space for all possible solutions, including potential solutions that nobody has ever thought of before. Next, there is the set of possibilities that are known to you, the vis designer. Of course, this set might be small if you are a novice designer who is not aware of the full array of methods that have been proposed in the past. If you're in that situation, one of the goals of this book is to enlarge the set of methods that you know about.

We created the first drafts of the low-fidelity prototype with pen and paper. We applied the five design sheet methodologies for each task defined in chapter 4. This approach allowed for rapid prototyping. For each, we created the brainstorming sheet to quickly collect ideas, filter them and remove duplication, categorize the sketches, and identify groups or patterns. We combined those ideas and created solutions or workflows for the application overall.

We selected multiple solutions from the brainstorming sheets with the help of the algorithm developer, a visualization expert, and three domain representatives. Afterward, we created initial designs with the initial design sheets. We discussed each and improved them until we narrowed the possible solutions down to a single combined solution. For this solution, we created the last sheet, the realization design. The final solution was presented to two visualization experts and refined into the final pen-and-paper design–the basis for the high-fidelity prototype.

The final version before refinement is illustrated in Figure Figure 4.3, and the final version

[3] Adobe XD is a UI/UX design & collaboration tool https://www.adobe.com/products/xd.html

after the last refinement in Figure Figure 4.4.

With the final version of our pen and paper prototype, we confronted our test user group with multiple workflows. We stepped through each workflow supported by the designs and explained which action we would perform or what function we would provide. We documented the feedback we received and used it for creating the high-fidelity prototype, which we will discuss in detail in section 4.5.

The final paper design in Figure 4.4 shows strategies for tackling predefined tasks. It allows for an introduction section where users can step through the features. It includes a clustering view with a scale space tree that encodes the solutions space for a specific alpha value. The design provides an option to modify that alpha value and subsequently change the tree. It is also possible to choose another density level within that tree. Both changes affect the overall clustering. Changing either will produce a unique pair of alpha and density levels and result in a unique clustering. A user can inspect details by clicking on that specific cluster. In addition, the design also includes the option to disable multiple stellar groups in bulk by creating a rectangle exclusion around the nodes in the level-set tree view. All clusters within that rectangle are excluded from the plots in all other views.

4.5 High-Fidelity Prototyping

During the design, the implementation, and the user test phase of this tool, we pursued improvement in incremental steps [52] and presented each increment to either a domain expert or the algorithm developer himself. This approach allowed us to react quickly to changing circumstances. During the user test phase, between each interview, we collected feedback from the users, both from the feedback form (see subsection 4.5.3) and the interviews (see subsection 4.5.2). We created a priority list in consultation with domain and visualization experts and implemented key feature requests and feedback between interview sessions. This method allowed us to apply the changes that mattered to the users and to focus on high-priority parts of the tool. There are two reasons for this decision:

- If we consider the Pareto Principle in software development [36], [34], [75], then around 20% of the changes will account for 80% of the effect
- To spend our time intelligently and potentially even reduce it, the domain and visualization experts should be deeply involved in this final stretch to focus on only the changes that matter.

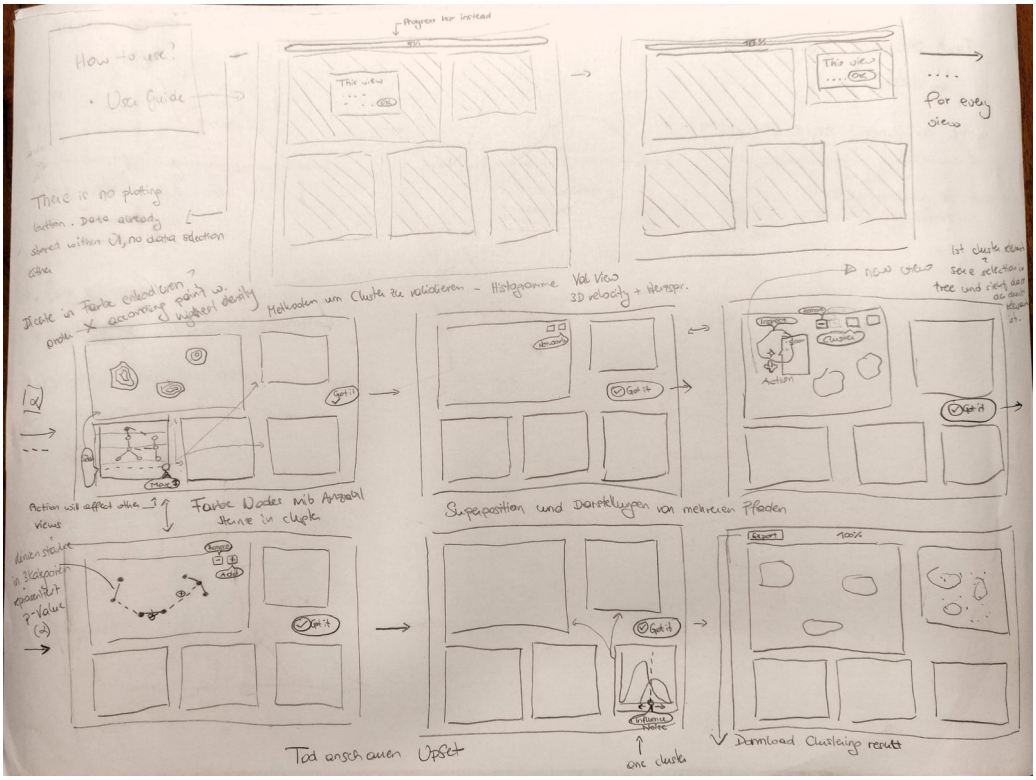


Figure 4.3: The pen and paper design solution before the final solution review

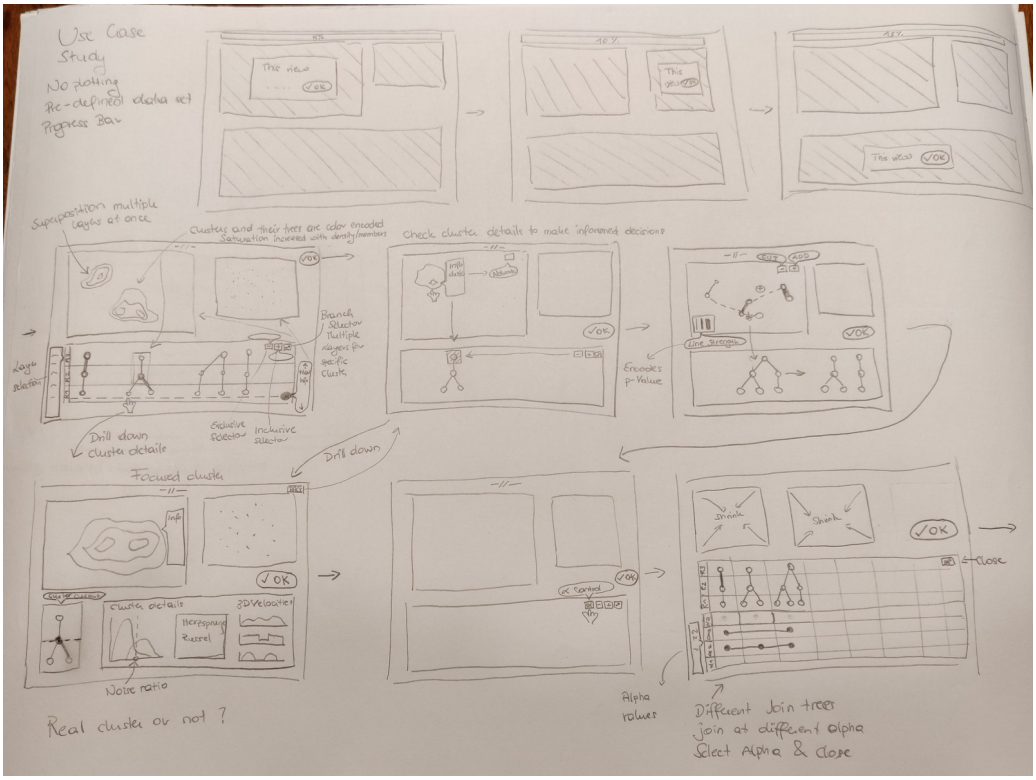


Figure 4.4: The final pen and paper design after the review. This version provides more details about cluster inspection and introduces a strategy to modify the alpha value. Herein we removed the noise extraction view and replaced it with velocity histograms and HRD in the cluster details.

In subsection 8.1.3, we will also reflect on this approach and discuss if it yielded any advantages.

Before creating the high-fidelity prototype, we conducted further research on effective visualization strategies, particularly for features such as color coding (see subsection 5.1.7) for various elements. We wanted to demonstrate how altering our two main parameters impacts the overall clustering process, including how changes in density affect cluster creation or elimination and how changes in the alpha value impact the clustering.

4.5.1 Prototypical Implementation

The high-fidelity prototype helped see the pen and paper designs in action and gather information from the users while they used the tool. We implemented it according to the task analysis performed in section 4.2 and based on the final version of the low fidelity prototype, created during the low fidelity phase section 4.4. The expert test user group used this prototype to perform previously defined mini-tasks. To verify its usability, we decided to use the System Usability Scale [10], hereafter SUS, questionnaire. To derive topics for future research, we also asked our participants to fill out a feedback form. To gain additional insights into their behavior and motivations, we decided to run each interview session in person.

4.5.2 Personal Interviews

Personal interviews allow the interviewer to make observations on the behavior of the particular interviewee. Significantly, judged in this case by the interviewer, behaviors or remarks were transcribed and then digitized - directly after the interview occurred. While the interview occurred, the interviewer aimed to make notes on the following criteria:

- Did they experience any problems while using the tool?
- Did they need to ask specific questions to continue?
- Were they able to navigate with ease?
- Was information missing?
- Did they provide additional feedback that wasn't within the feedback form?
- Did they experience bugs or performance issues due to their behavior?

4.5.3 Feedback

This work also included a feedback form to evaluate potential improvements for the tool. The users completed it after our test user group finished the SUS questionnaire. The feedback form consisted of the following questions:

- Can you add a new resource to the resource pool?
- Are you able to traverse the solution space, all solutions between different alpha and delta settings?
- Were you able to extract or identify a cluster from that solution space?
- Was it easy to identify areas of interest in the solution space?
- Was it possible to inspect the details of a specific cluster and change its name?
- Did you find it easy to export the result of your interaction with the tool?
- Are you satisfied with the result?
- What would you improve?
- What can be improved to traverse the possible options?
- What can be improved to compare a cluster to other clusters?
- Do you find the user interface intuitive?
- What could help you to explore the parameter space more efficiently?
- What functionality is missing?

We designed those questions to not only include specific inquiries related to this tool but asked for general feedback.

4.6 System Usability Scale Questionnaire

To assess the usability of our implementation, we administered the SUS [11] questionnaire to participants. The SUS is a widely-used, industry-standard tool known for its brevity and ability to elicit meaningful responses quickly. In line with the findings of Galesic and Bosnjak [31], we believe using a shorter questionnaire can improve response quality. To interpret the results, we followed the percentile ranking method proposed by Bangor, Kortum, and Miller [3] and

converted the scores to a letter grade system for ease of presentation and interpretation. Additionally, we used an adjective-based rating scale to provide a complementary perspective on the scores. We hope to gain a more nuanced understanding of the results and communicate them effectively using both evaluation techniques.

To gather accurate feedback, we asked each domain expert in our test user group to complete the questionnaire immediately after using the tool. This allowed us to capture their impressions and reactions while they were still fresh.

Chapter 5

Design

In this chapter, we will discuss the interface of Lodestar (see section 5.1), and we will conclude with the design rationales (see section 5.2).

5.1 Interface

Due to the nature and scale of GAIA data, there are many possible cluster solutions, each depending on a specific parameter configuration. Density-based clustering, in particular, is known to be highly sensitive to the input parameters and can result in a vast range of different cluster shapes and densities, as highlighted, for example, in a study by C. Boquan, E. D´Onghia, J. Alves, and A. Adamo [14], which have searched for star clusters in the Orion complex. Our goal is to simplify the process of exploring these solutions by providing an easy-to-use tool that allows for efficient exploration of the parameter configuration space.

To accomplish this, we adhere to the Schneiderman Mantra, which emphasizes the importance of providing an overview first, allowing users to zoom and filter, and providing details on demand. Our tool provides an interpretable overview of all possible clustering results, enabling users to focus on regions of interest and exclude irrelevant data quickly. Users can filter the data to gain a deeper understanding of specific clusters and acquire additional information, such as the relationship between the luminosity of each star and its classification.

The ability to zoom in on specific regions and access details on demand allows for quick validation of the overall results. The ultimate aim of our tool is to facilitate a comprehensive exploration of the solution space and make it easy for users to understand the full extent of the data, hone in on specific areas of interest, and validate their findings with detailed information.

Our interface consists of multiple screens (see Figure 5.1) to help users resolve the predefined tasks from section 4.3. We will discuss the features of our tool while stepping through the main

workflow and relate each to a task, where appropriate.

When a user reaches the web interface of our tool, it responds with the input form for the clustering algorithm (see Figure 5.1 (1)). We present a call to action, allowing a user to start with the clustering promptly, and we consider this as the pre-clustering step. After the algorithm finishes, we enter the post-clustering phase. Aside from that, a user has the option to switch to another main navigation page, complementing its primary function with a tutorial section (see Figure 5.3a (b)) and a resource management (see Figure 5.3a (c)) page. The tutorial addresses Task 1, and it aims to answer specific questions and helps to understand how the main parameters or the individual views work. The resource manager offers an input field wherein a user can provide a link to upload an external resource by clicking on the "Add resource" button.

The input form asks a user to select a source file containing a region the user wants to study the cluster behavior in and the features used for clustering (see Figure 5.1 (2)). To run SigMA, a user clicks the run button and, when the clustering is complete, is forwarded to the clustering summary (see Figure 5.1 (3)) to explore the available solution space. The tool supports exploring that space with three modes of operation, each providing a different perspective on the available data.

The first mode, hereafter referred to as "Solution Space Mode" (see Figure 5.1 (3)), provides the user with the means to explore the solution space by a change in either density level or alpha value in the available parameter space. This view contains a cluster development tree for each parameter (see Figure 5.6 and Figure 5.7) and provides action buttons to change the current selection that subsequently causes a change in either tree. Since we have a much higher variance in cluster solutions in the density level tree, we have reserved a larger space for it as for the alpha tree. These trees, the parameter heat map, and the stability view are different perspectives on the parameters space and help to address Task 2 and Task 3. A change in either view will lead to a change in the presented clustering, reflected in each validation view (depicted in Figure 5.11. We will talk more about views in subsection 5.2.1). The validation views contain a position, velocity, and Hertzsprung–Russell scatter plot and aid a user in verifying a clustering result. These views help a user to address Task 4.

To access details on demand, a user triggers the forward button to activate the "Alpha Mode" (see Figure 5.1 (4)). While switching, the current alpha value is stored and remains constant in this mode, allowing the user to focus on changes in the density level. When activated, the "Alpha Value Tree," "Steering Wheel," and "Stability" views are hidden to give users more space to explore the density level tree and validate results with the top views. Both modes provide actions to inspect and validate individual clusters in the "Cluster Validation Mode" (see

Figure 5.1 (5)). At any time, a user may download the current clustering (see Figure 5.1 (6)). We will discuss these modes in subsection 5.1.2.

The "Modes of Operation" philosophy emphasizes the importance of providing a quick overview of the solution space while simplifying the data analysis process by dividing the tool into different modes of operation. This includes providing a high-level overview of the data, such as the clustering dashboard and the parameter trees, as well as modes for more in-depth exploration, such as filtering and drilling down into specific subsets of the data. We follow this philosophy not only by providing multiple modes but follow the same approach to structure each mode individually (see Figure 5.2). Additionally, we designed the tool to reduce complexity by providing a clear and concise explanation and context for the data and the analysis methods used. The tool's layout remains consistent across all modes to minimize cognitive load and increase efficiency. By maintaining a familiar structure, users can focus their mental energy on the task at hand rather than navigating the interface. The goal of this philosophy is to allow users to quickly understand the scope of the solution space, help navigate it, understand the effect a change in density level and alpha value has on the outcome, and provide insights and findings of the results while also providing the flexibility to explore the data in more depth if desired.

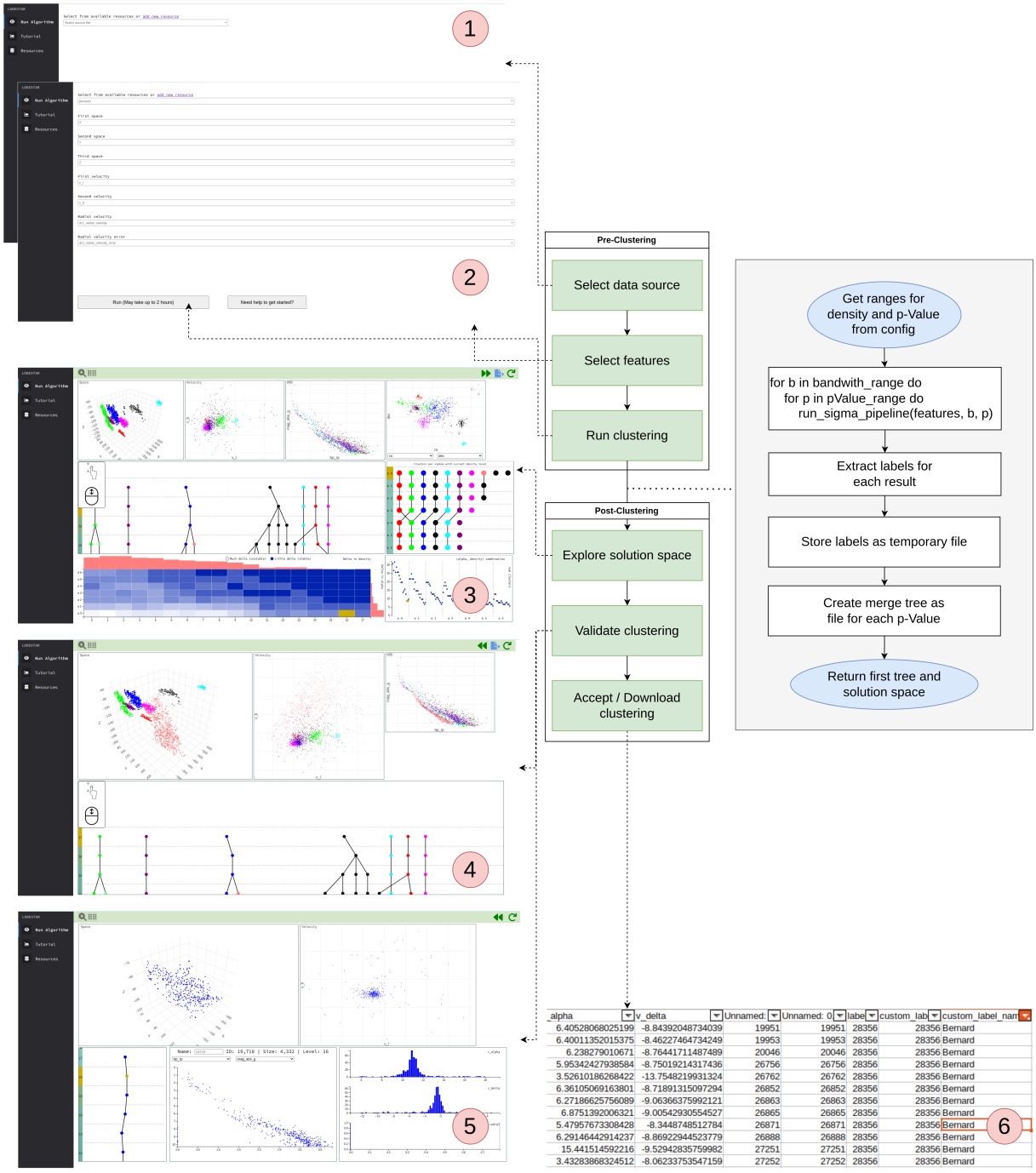
5.1.1 Navigation

Our tool incorporates navigation with few elements (see Figure 5.3). Users should reach their goals by interacting with the visual encodings and can drill down on demand. Therefore only the most high-level concepts are available in the navigation. A user that navigates to our web interface arrives at the default page seen in Figure 5.3a, which is solely concerned with the algorithm execution.

Although the solution space exploration takes place on a single page, we have two steps separating the pre-clustering and the post-clustering step. In pre-clustering, we choose the data source and the features for clustering. After the algorithm is finished, the post-clustering step (see Figure 5.4a) will open, and we may interact with the algorithm output. By interacting with the visualization, we view the data from different modes of operation (see subsection 5.1.2).

5.1.2 Modes of Operation

Figure 5.4a, Figure 5.4b and Figure 5.4c showcase the three modes the tool supports. We developed those modes to allow the user to drill down and focus on different levels of complexity. At first, the tool introduces the user to the solution space. It presents the clustering for the



alpha	v_delta	Unnamed:	Unnamed: 0	labe	custom_lab	custom_label_nam
6.405280680251990	-8.843920487340390	19951	19951	28356	28356	Bernard
6.400113520153750	-8.462274647342490	19953	19953	28356	28356	Bernard
6.238279010671	-8.764417114874890	20046	20046	28356	28356	Bernard
5.953424279385840	-8.750192143174360	26756	26756	28356	28356	Bernard
3.526101862684220	-13.75482199313240	26762	26762	28356	28356	Bernard
6.361050691638010	-8.718913150972940	26852	26852	28356	28356	Bernard
6.271866257560890	-9.063663759921210	26863	26863	28356	28356	Bernard
6.875139200632100	-9.005429305545270	26865	26865	28356	28356	Bernard
5.479576736308280	-8.344874851278400	26871	26871	28356	28356	Bernard
6.291464429142370	-8.869229445237790	26888	26888	28356	28356	Bernard
15.44151459221600	-9.529428357599820	27251	27251	28356	28356	Bernard
3.432838683245120	-8.062337535471590	27252	27252	28356	28356	Bernard

Figure 5.1: (1) Source file selection. (2) Clustering feature selection. (3) Solution Space Mode. (4) Alpha Mode. (5) Validation Mode. (6) Result file export.

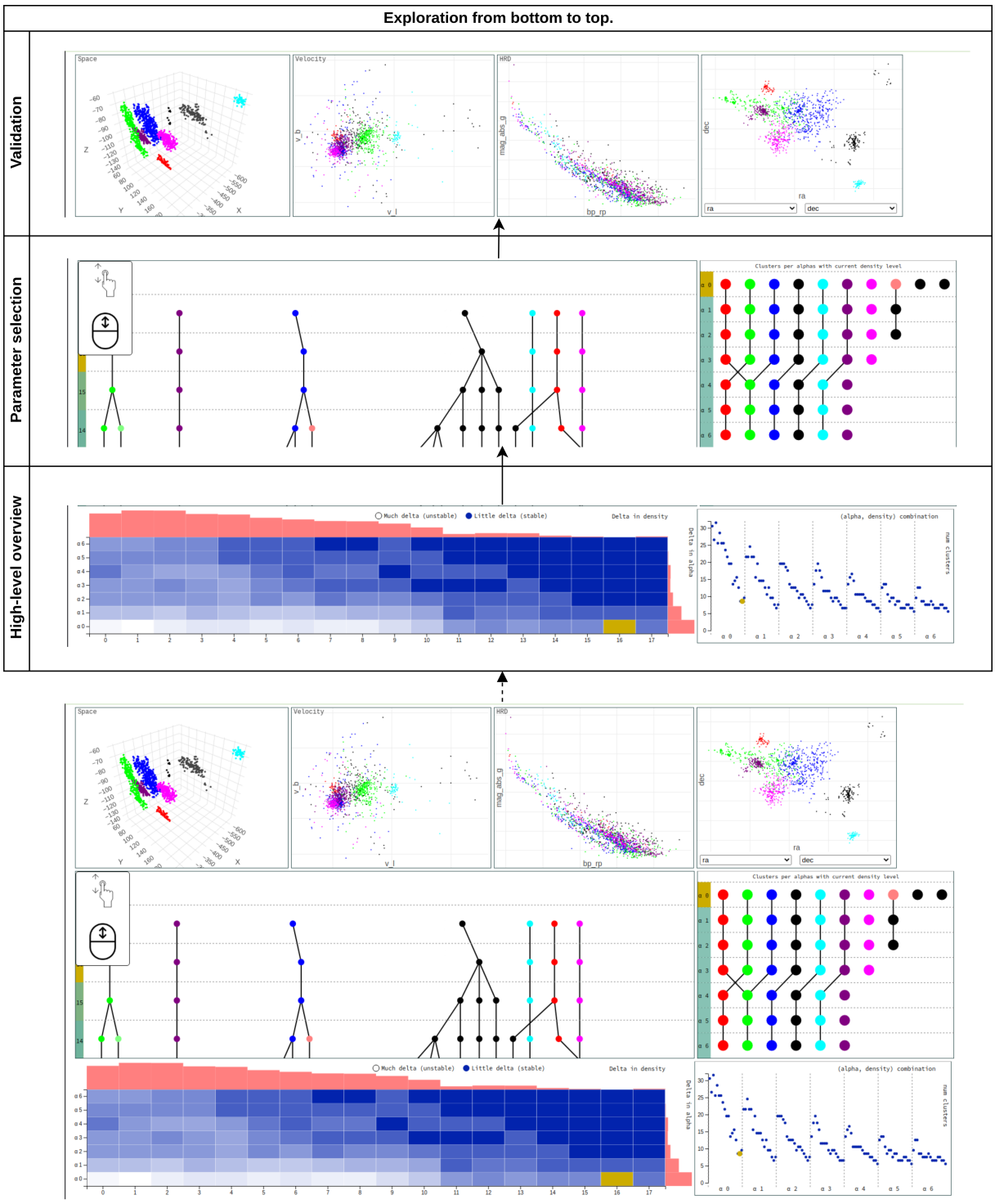
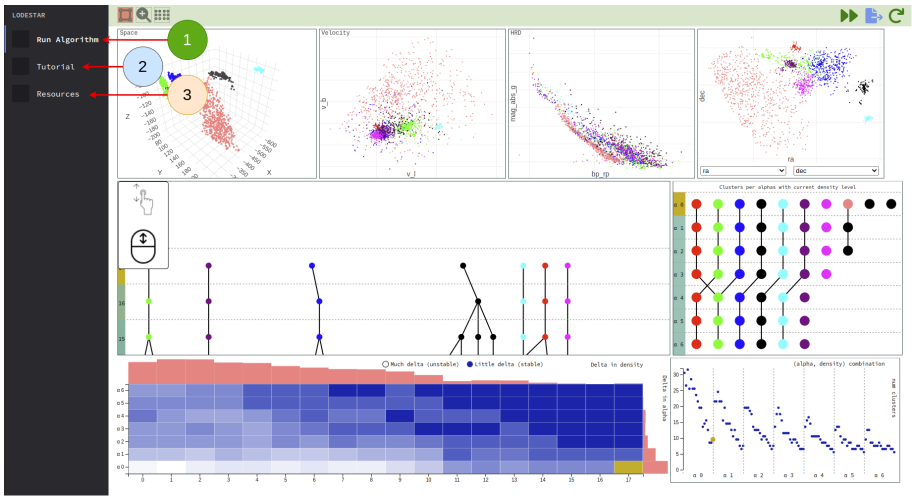
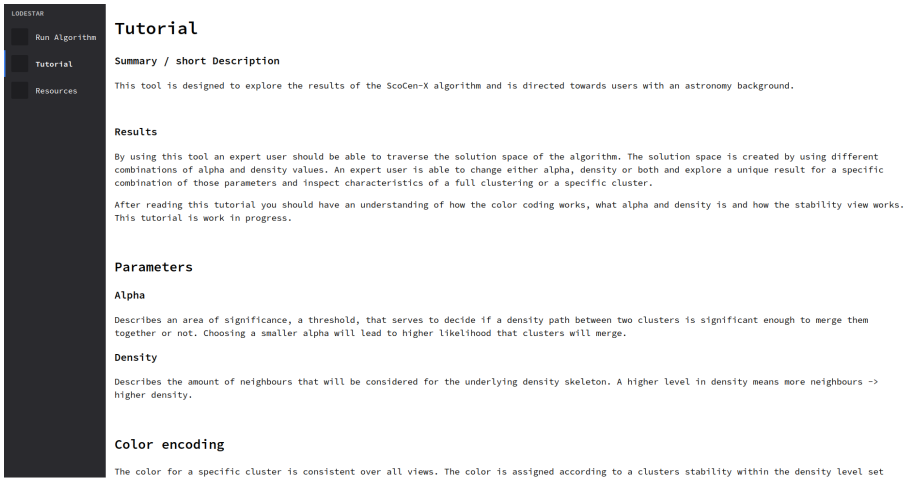


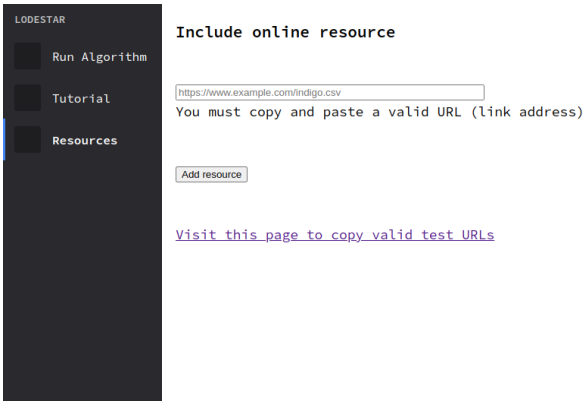
Figure 5.2: As an example of the drill-down principle within a mode, this depiction shows the "Solution Space Mode" to highlight the hierarchical nature of each mode.



(a) (1) Clicking this element will open (a), and a user will either see an already active session or has the option to start a new clustering session. By clicking (2) a user will see (b) and by clicking (3) a user will be directed to (c).



(b) A tutorial about the visual encodings of the tool and parameters of the algorithm.



(c) Data source management. The sources will be available during algorithm execution.

Figure 5.3

lower bound of predefined ranges of alpha values and density levels, the "Solution Space Mode." We have obtained these ranges from the work of Ratzenböck, Großschedl, Möller, Alves, Bomze, and Meingast [61]. The tool allows users to pick different values among the ranges to explore different results or pick the current alpha value selection and explore that clustering selection in more detail by adjusting only density levels in the "Alpha Mode." In both modes, a user may proceed to the Validation Mode and look at individual clusters and their specific properties to validate single clusters for their goodness.

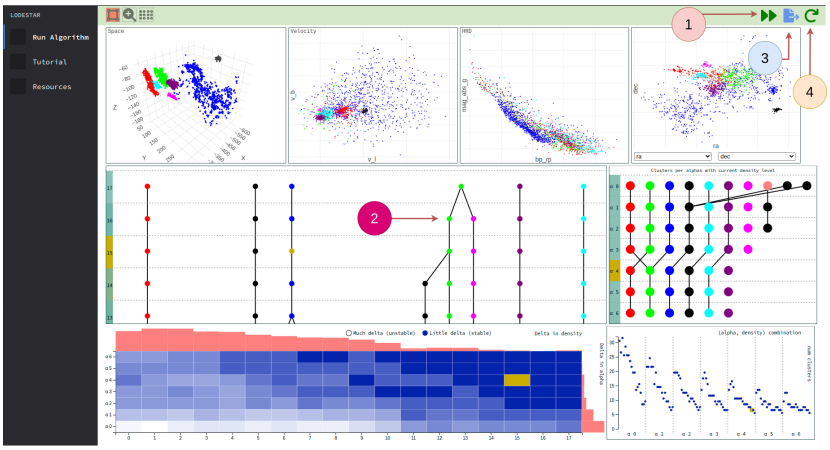
5.1.3 Scale Space and Alpha Tree

The "Scale Space Tree" (also referred to as the density level tree) addresses Task 2 and Task 3. This view provides an overview of the solution space in the density level variable. It visualizes the change of the clustering solution by varying the density filter coarseness. The outcome is a tree with numbered rows where each row is an entry in our density level range, i.e., a row represents one clustering solution. Each node in the tree is a cluster, and the edges represent merge paths between the nodes. Node/cluster identity in both the "Alpha Tree" and "Scale Space Tree" and validation scatter plots is consistently encoded via color throughout respective views. Colors appearing most frequently across all rows are considered the most stable.

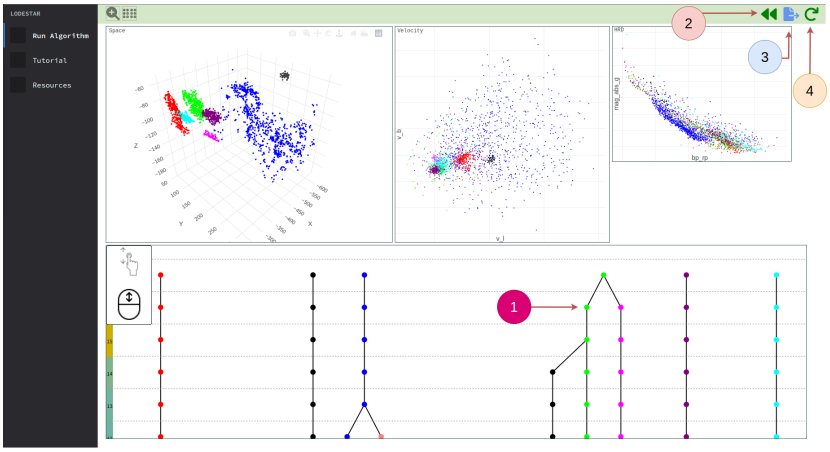
This view allows users to select a specific row number for an entry in the density level range. Doing so will change the active clustering solution and update the clustering validation views (see subsection 5.1.6), "Alpha Tree," "Steering Wheel" (see subsection 5.1.4), and the "Stability" view (subsection 5.1.5).

Additionally, users can interact with the nodes in the tree. Hovering allows users to see a summary, a mouse click will highlight the cluster in all other views, and a double-click will open the "Validation Mode," thus enabling users to drill into various aspects of a stellar group. Due to the large range of density levels, we introduce additional actions like zooming and scrolling to navigate the density tree and zoom into areas of interest. Figure 5.6 illustrates the density tree.

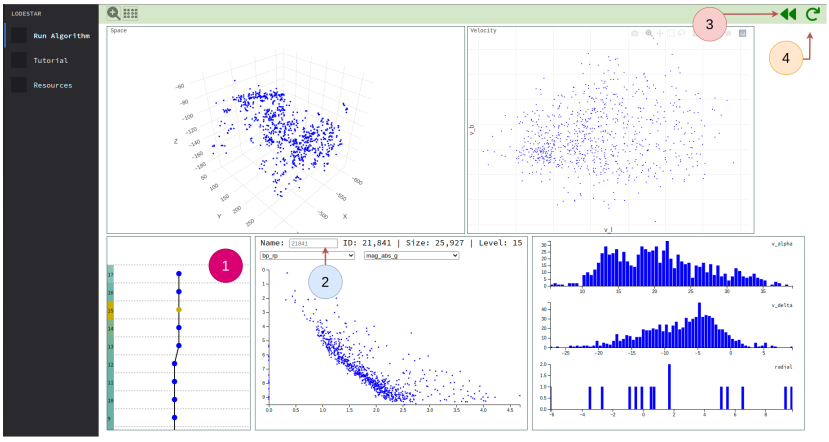
Similarly to the "Scale Space Tree" view, we use the "Alpha Tree" view to explore the development of clusters for the current density level selection, see Figure 5.7. If a change to the alpha value occurs by selecting another row, then all other views will reflect that change. By clicking on the numbered row in the left-hand column, a user can explore the alpha value and analyze what effect that change has on the current clustering selection. Similar to the "Scale Space Tree," it addresses Task 2 and Task 3. In combination, they represent our parameter range. Consequently, changing one tree will also alter the other, resulting in a linked relationship



(a)

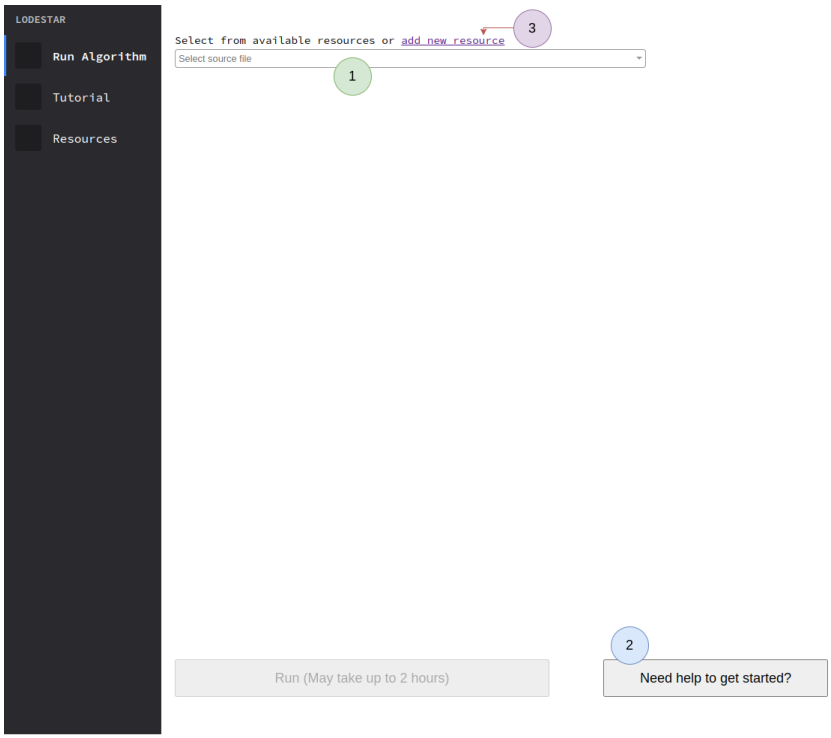


(b)

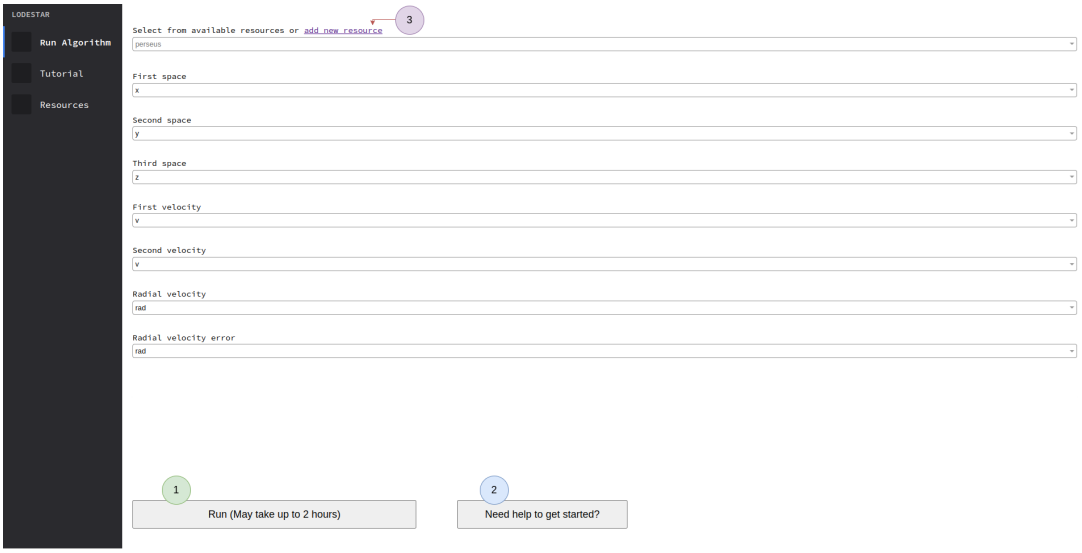


(c)

Figure 5.4: (a) State of the tool when the algorithm is finished. By using the navigation button (a)(1), the tool will switch to mode (b). Clicking on either (a)(2) or (b)(1) will highlight that cluster. Double-clicking on (a)(2) or (b)(1) leads to (c). The tree depicted in (c)(1) is filtered to show only the development of the current cluster. (c)(2) changes the current cluster's name. Either (c)(3) or (b)(2) will lead to the previous mode. Either (a)(3) or (b)(3) will export the current clustering to a file. (a)(4) or (b)(4) or (c)(4) will start a new session.

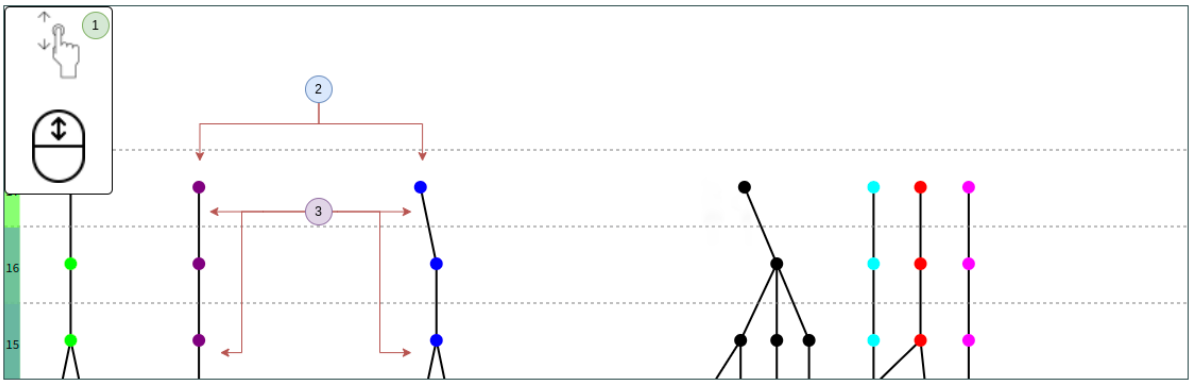


(a)

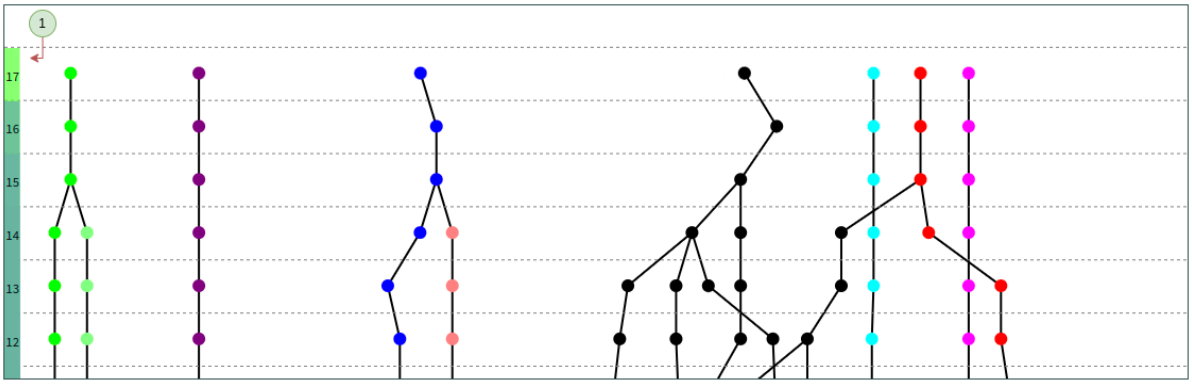


(b)

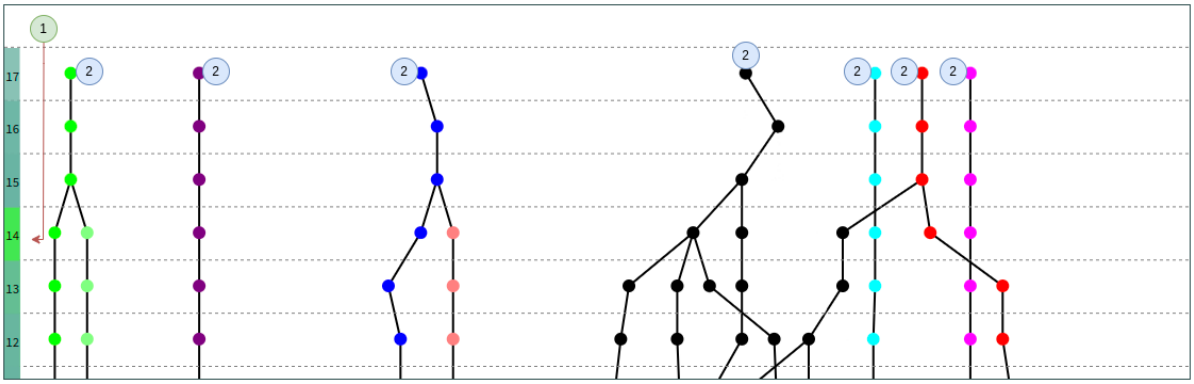
Figure 5.5: (a) Landing page of the tool. In (a)(1) a resource is selected. This resource will be used by the algorithm. After the resource is selected (b) will appear and the seven parameters used by the algorithm are selected. The possible options are collected from the selected resource and represent the column headers. With (b)(1) the algorithm is executed with the current selection. With either (a)(2) or (b)(2) a user opens the tutorial section of the tool. With either (a)(3) or (b)(3) a user switches to resource management to upload additional resources.



(a)



(b)



(c)

Figure 5.6: (a) Initial state of the tree view with an active tooltip indicating a panning and zooming option in (a)(1). (a)(2) shows stable clusters and their development as a sub-tree in (a)(3). (b) shows the same view - the trees moved up because of the panning action. Interacting with this view will make the tooltip disappear. (b)(1) highlights the currently selected density level. When a user clicks the row, the level is active, as seen in (c)(1). Clusters allow interaction with one click or double-click on (c)(2). A click will highlight this cluster in every view, and a double click will navigate the user to the cluster details Figure 5.4c.

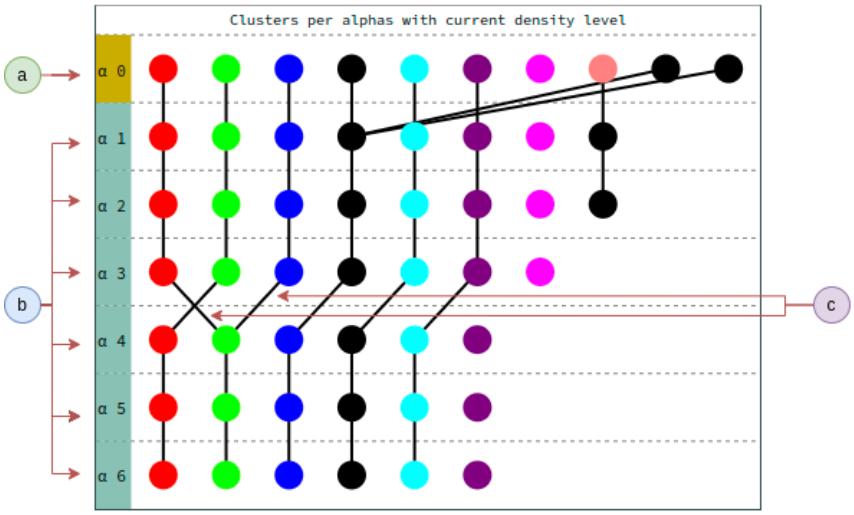


Figure 5.7: (a) Yellow is the current alpha selection. (b) If you click on any other row, you change the alpha value and update all other views automatically. (c) The lines between the circles indicate merge paths. When the value transitions from $\alpha 3$ to $\alpha 4$, the red and blue clusters will merge with the green cluster.

between the trees. This interdependence creates a linked tree scenario where any modification in one tree will inevitably affect the layout of the other. These interlinked trees reflect the interconnected nature of the parameters, highlighting the complex interplay between the two that determines the outcome. Combining the trees creates a more complex structure, a collection of trees that intersect each other. We refer to it as an intersecting tree collection. The overall structure shows the relationship between the parameters. The resulting nodes are an intersection of both selections.

Thus, a user can track the clustering solution with varying density levels and alpha values. Both views showcase stable clusters as constant colored nodes that persist for a long time. This view enables the user to gain an overview of the entire solution space in both variables, make preliminary decisions on clustering solutions, and see changes in the clustering by changing either variable. This arrangement enables users to reason about the solution space before looking into individual clusters and validating them. This a priori reasoning capability makes identifying good solutions more efficient and speeds up the decision-making process. Both views offer actions to drill into specific solutions or clusters on demand.

5.1.4 Steering Wheel

Tree views offer a useful way for users to assess the relevance of cluster solutions quickly. The views, however, only depict limited portions of the parameter space, and a full representation

of the intersecting tree collection is not practical at this level of detail. To address this, the goal of this view is to provide a comprehensive summary of the entire parameter space, highlighting the stability of regions in the feature space, which is crucial to a successful solution.

The "Steering Wheel" view (shown in Figure 5.9) serves as the central control for quickly navigating between alpha value and density selections, addressing Task 2. It presents a heat map with sampled alpha values listed on the vertical axis and sampled density levels on the horizontal axis. By clicking on a rectangular cell within the heat map, the user simultaneously changes the current selection of alpha value and density level. Such a change updates the current density level and alpha value and therefore causes an update in all other views. The color intensity of each cell is determined by the solution stability, which is computed by calculating the mean of the symmetric differences between the current cell and its horizontal and vertical neighbors, as represented by the following mathematical notation:

$$\frac{1}{4}\Big(|C_{\text{up}} - C| + |C_{\text{down}} - C| + |C_{\text{left}} - C| + |C_{\text{right}} - C|\Big)$$

C refers to a cluster solution (i.e., a given alpha value and density level), while C_{up}, C_{down}, C_{left} and C_{right} refer to the neighboring cells, see Figure 5.8. The symmetric difference between two cells is defined as the difference in their respective cluster label sets.

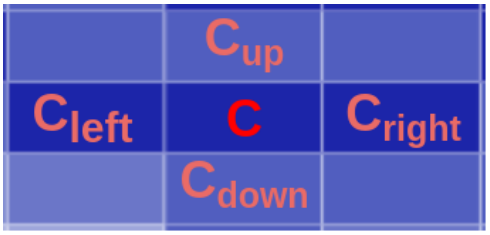


Figure 5.8: Neighboring cells in the "Steering Wheel" heatmap.

Additionally, we use a similar approach to calculate the bar charts on the top and right-hand sides of the heat map. Each bar on the top represents the variation within the column, and each bar on the right chart the variation within the row representing the total variation in clusters across a given density level and alpha value, respectively.

By combining the stability encoding of the heatmap and the bar charts, we provide visual clues to separate volatile from stable areas, thus helping in making decisions on good clusters based on stability. This view is beneficial in listing and quickly exploring all possible clustering solutions and in filtering those solutions based on the stability feedback provided by the heat map and the two histograms on the top and right-hand sides.

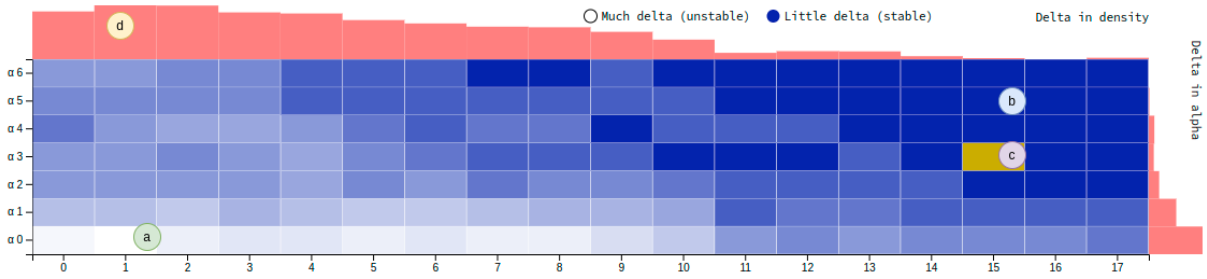


Figure 5.9: A heat map depicting the stability range with changing alpha and density values. (a) White indicates an unstable selection - a high rate of change compared to the neighbors. (b) Dark blue indicates a stable selection - a low rate of change compared to the neighbors. (c) Yellow highlights the currently selected density and alpha value. (d) Light red is used for representing the bars of a histogram. Each bar's height is derived from the overall change within that column or row.

5.1.5 Stability View

The "Stability" view (shown in Figure 5.10) is similar to the "Steering Wheel" view, which helps find stable solutions in the entire solution space. When a user operates in the "Solution Space Mode," this view is in the lower right corner of the screen. It is a read-only scatter plot that highlights the currently active clustering solution. The solutions are grouped by alpha value, and in each alpha section, the density level is increasing. This view allows users to observe the dispersion of solutions within an alpha value and changing density level and shows how many clusters this solution produces.

This view addresses Task 2 and provides a bird's-eye view of all solutions, highlighting the change in the number of clusters with changing parameter tuples. In contrast to the density tree visualization (discussed in subsection 5.1.3), this view allows for easy detection of stable areas across the entire solution space (instead of focusing on slices in the tree views), enabling the user to make informed decisions when choosing the appropriate parameters.

Stability around a parameter tuple is often a good indicator of a desirable outcome. We assign each parameter tuple an index and map it along the x-axis, while the y-axis ranges from 0 to the maximum number of clusters for any combination. Therefore, in contrast to the "Steering Wheel," this view focuses solely on the total amount of distinct clusters in that result. Furthermore, points are grouped by constant alpha values and arranged in each alpha group by increasing density level. Plateaus or steep regions in each subgroup highlight either stable or volatile alpha selections, respectively. This added level of detail can help users identify not only the number of clusters but also the stability of the resulting clusters as the parameters vary.

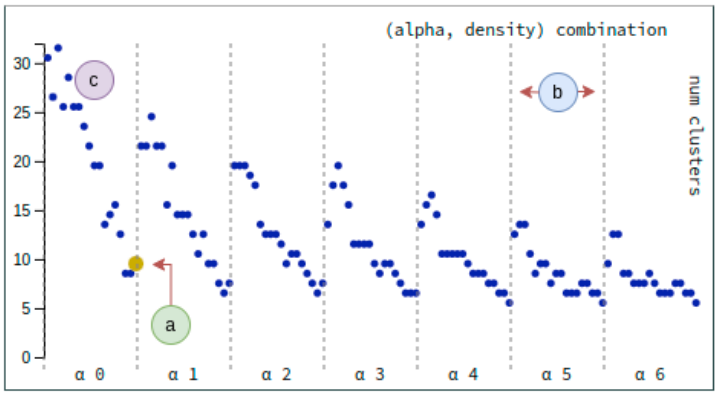


Figure 5.10: (a) Yellow is the current alpha value, density level pair. (b) The stroked line separates changes in alpha value while (c) the blue dots represent individual pairs.

5.1.6 Clustering Validation

Users can explore and evaluate clustering solutions in various ways using the cluster evaluation views seen in Figure 5.11. These views, which include a 3D space, 2D velocity, HRD, and configurable 2D scatter plot, allow for studying different properties, shapes, and distributions within clusters. In the "Cluster Validation Mode", we provide velocity histograms for further analysis. Together, they address Task 4 by presenting the clustering solution in space, velocity, HRD, and other feature pairs the user selects. The colors of each star in the plots represent the assigned cluster. Clicking on a node in the tree visualizations, mentioned in subsection 5.1.3, highlights the current cluster (as seen in Figure 5.12). By toggling the noise button (seen in Figure 5.13), a user can view the entire data set, including stars classified as noise. Users can pan and zoom in on each evaluation view and optionally download a picture of the current state.

These interactions enable experts to validate the quality of the clustering solution by examining the proximity and shapes of stellar groups in spatial and velocity dimensions, supported by an HRD and velocity histogram.

5.1.7 The Use of Color

A well-designed color palette is essential for the efficient and intuitive use of a visualization tool. It is important to maintain consistent color assignments and meanings across all tool views. To ensure that the astronomy community easily understands our tool, we draw inspiration from existing solutions that are based on commonly used techniques such as color consistency across views, separating system colors from colors for data entities, and using colors with a high perceptional difference [24]. The color palette shown in Figure 5.14 is generated using a color

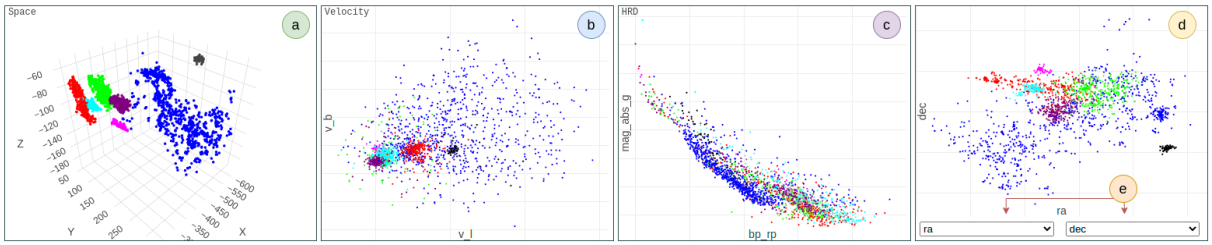


Figure 5.11: Figure depicts how clusters from a selected clustering solution group in either 3D space (a), velocity (b), amongst an Hertzsprung-Russell diagram (c) and in the fourth view (d), any two dimensions the user selects (e).

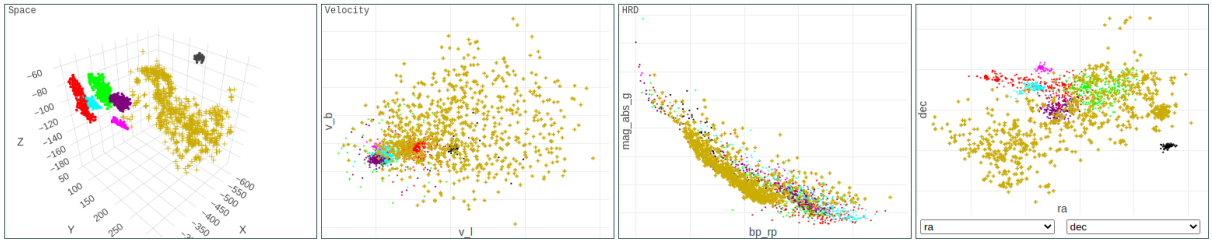


Figure 5.12: Evaluation views with the highlighted cluster.

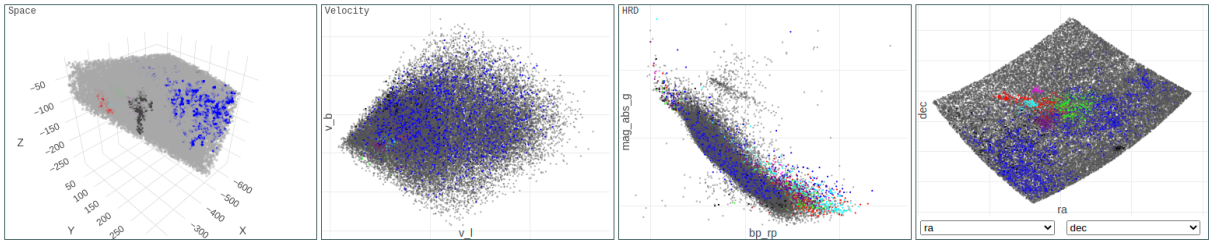


Figure 5.13: Evaluation views with noise activated.

scale tool[1] to maximize perceptional differences. We used the generator to ensure that the colors in the palette are easily distinguishable from one another.

The tool assigns these colors, from left to right, to a set of clusters sorted by their stability. In this context, stability refers to the number of levels in the scale space tree where a particular cluster appears. Once the pool of available colors is exhausted, they are assigned black. This means that the most stable groups are significant enough to carry color. By clicking on a cluster, a user activates the highlight feature, and its color changes to the highlight color (see Figure 5.14 right highlight color). The markers for the selected cluster change from circles to crosses Figure 5.15. Therefore the user identifies and focuses on the stellar group and tracks it through all views.

[1]https://mokole.com/palette.html

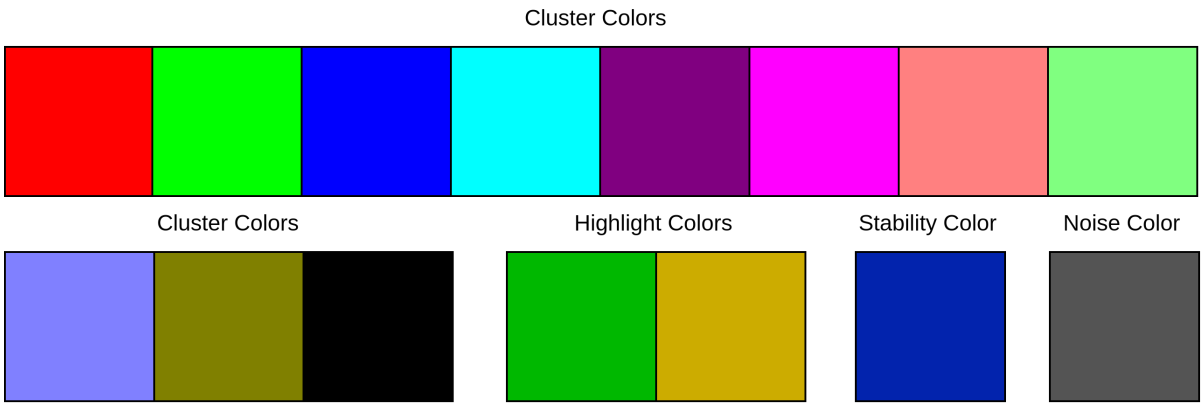


Figure 5.14: Colour palette, ordered by significance and assigned to clusters that are ordered by significance. The image also shows the selection for highlighting colors and the color used for the two stability plots.

(a) Default marker for a star within a cluster.

(b) Default marker for a star within a highlighted cluster.

Figure 5.15: Both symbols are symbols from the plotly symbol list[2].

5.1.8 Noise Toggle

Multiple test users requested a means by which noise can be enabled or disabled. In the default setting, all plots will only draw the points of labeled clusters - noise is ignored. When a user enables the noise toggle, our tool adjoins the noise points to each plot. Colored in grey, they are distinctly different from the other points and can be visually separated.

5.1.9 Result File Export

The result file download feature in our interface allows users to download the results of their analysis for further processing in other tools such as TOPCAT. This feature is designed to be easily accessible within the visual interface and provides users with a convenient way to export their results for further analysis or visualization. The interface displays the download option in every mode as a button on the top right corner, allowing users to initiate the download process with a single click. Upon completion, the result file is saved to the user's local system and ready for use in other tools. This feature enables users to take full advantage of the results of their analysis and use them in other tools for even more in-depth analysis and visualization.

5.1.10 Source File Upload

The resource file upload feature allows users to upload new data sets to the system. This enables users to easily add new data sets to the application, which can then be processed, analyzed, and visualized. The resource file upload feature is designed to be user-friendly and intuitive by providing a text field to enter a URL to an online data source and informing the user about the upload process via status feedback. This allows users to quickly and easily upload new data sets and incorporate them into their analysis. The visual interface provides confirmation when the upload is complete, allowing users to continue their work with the newly added data sets quickly.

5.2 Rationale

In this chapter, we will discuss the reasons and motivations behind the design decisions and visual encodings we have chosen.

5.2.1 Tile-Based Interface

We have opted for a tile-based interface to flexibly add or remove tiles based on design decisions that may arise in the future and to fit multiple visual encodings into a single dashboard. Furthermore, we wanted to scale each tile according to the available screen space and the neighboring tiles. We have removed the borders of the tiles at a later stage due to feedback from our visual experts. While doing that, we removed the action buttons from each tile. These buttons are now global and reside in an action bar at the top of the dashboard. This change has helped us save space and communicates that each action affects the entire data. To reduce cognitive load [19], we wanted to offer a single dashboard - all the related views are composed and prepared to serve a single task, exploring the solution space. When users decide they want more information on a specific result, they can drill down by inspecting a cluster.

The tile-based solution helped separate parts of the data into distinct plots. Each of these tiles is interconnected, and a change in one will affect the others.

5.2.2 Graph-Based View for Algorithm Hyper-Parameters

Graph-based visualizations are ideal for representing relationships between entities hierarchically and for scalable and dynamic data [5]. They allowed us to introduce hierarchy in both horizontal and vertical directions. In our case, especially the development of the entities along the vertical axis helped to highlight the changes an increment of a parameter introduces. We have chosen

to use this encoding to convey how changing one parameter will affect the cluster solution. The graphs served to observe the evolution of the clustering result with increasing or decreasing parameter values. Users can select either by clicking on a level or a specific cluster. The graph-based overview helps to immediately identify which stellar groups are eliminated or created and which merge. We could have also encoded the same information as a treemap which is also used to visualize hierarchies. Treemaps have the downside in that information is encoded as areas which is a worse representation than length. It's also not straightforward how to encode information such as stability into the areas of the treemap. Instead of stability, the cluster size could also be encoded in the rectangle size. But sometimes, two subclusters are larger than their parent, which results in problems with visual encoding. Another interesting quantity and candidate for showing in a tree map is cluster stability. But again, it is not straightforward to encode the stability of a cluster in a treemap. Whereas it can more easily be interpreted in a tree. In a tree, the stability can be visualized with the number of levels the cluster exists in.

A treemap could leverage color (e.g., saturation) to encode stability. But using color to encode both stability and containment could be confusing. Moreover, colors are again not as effective as length. We were looking for a more dynamic visualization focused on highlighting the evolution of the clustering. Separating the parameters in separate graphs also helped to reduce cognitive load and directly communicated the effect of a change. Both views will immediately affect the other—representing the connectedness between the parameters.

5.2.3 Scatterplots for Features, Velocity, Space, and HRD

Scatterplots appear throughout clustering visualization tools [71], [48], [26], [18] and are a typical chart type for visualizing clustering results, that effectively highlights groups and memberships. We have chosen the scatter plot for visualizing clustering in space, velocity, and for any two user-configured features. For space, we used a three-dimensional scatter plot due to user feedback in the low-fidelity prototyping phase. 3D visualizations can present several challenges, such as increased complexity and difficulty in interpretation, limitations in the human visual system's ability to perceive 3D information, the potential for clutter and occlusion, and difficulty in interaction and explorations [56]. But the domain experts were very familiar with this kind of representation, especially because they used tools like TOPCAT, which supports a 3D rendering of the data set and according to the feedback we received, it is part of an astronomer's decision-making to evaluate clusters. To mitigate some of the shortcomings, we have added several interactions to the plot, like panning, zooming, rotation, and highlighting. Panning, zooming, and rotating allow users to modify a chart and see the data from multiple perspectives. Through

this, we overcame some of the problems cluttering and occlusion can introduce. With the highlighting action and the tooltip, we added additional information on demand to reduce visual complexity and help users track specific clusters.

5.2.4 Heatmap to Hint at Areas of Interest

Heatmaps are especially useful in uncovering relationships within multidimensional data [38] and provide an excellent overview of bi-variate data. They allow for the comparison of multiple data points in a single image, using color to indicate the relative value of a data point. Making it easy to identify patterns and trends in the data. One of the main advantages of heatmaps is that they can display large amounts of data in a small space, making it easy to compare data points and identify patterns.

This visual encoding was added to detect stable subspaces in the overall parameter space and is ideal for quickly traversing that space. While each graph-based scale space tree provides a compact overview of the clustering effect by adjusting one parameter, the heatmap gives us a useful summary of the whole parameter space. The work of Healy [40] provided us with many alternative visual encodings that we will shortly discuss.

Other visual encodings, such as contour plots, density plots, and choropleth maps, also use color to indicate the value of a data point. However, they are not as well suited for displaying data in a 2-dimensional grid format as heatmaps are. Contour plots are better for displaying data on a continuous surface, density plots are good for showing the distribution of data points and their concentration, and choropleth maps display data on a map.

5.2.5 Number of Clusters for an Alpha Value and Density Level Combination

The "Stability" view was added late in the high-fidelity prototyping phase and contains a scatter plot. In this scatter plot, the horizontal axis comprises different combinations of alpha value and density level. For each parameter tuple, we show the number of clusters created along the vertical axis. Like the heatmap, we have added this view to support a user in finding areas of interest. Located next to the heatmap, it will help users to navigate large solution spaces.

5.2.6 Using Histograms for Distribution of Features

When a user selects a cluster for closer inspection, we use histograms to encode the distribution of a cluster feature. They provide an overview of the frequency distribution [43] and aid in visual cluster validation. As we previously highlighted, stellar groups share similar spatial and velocity properties. Therefore the distribution of an (e.g., velocity) feature can provide information

about the goodness of a clustering solution. We have added this feature due to feedback from the astronomy user group in the low-fidelity prototype.

Chapter 6

Implementation

The main concern of good software design is the separation of concerns. Therefore we have separated our software solutions into two parts. The front-end application is written in Typescript and contained within the Vue.js framework. The back-end is developed with the Flask framework and written in Python. We divided both parts again into smaller modules, or areas of concern, independent modules in the front-end and back-end applications.

6.1 Rapid Setup and Startup

This work has benefited from a rapid tool installment and startup. We achieved this by first setting up the infrastructure and creating the whole setup by containerizing the back-end and front-end into isolated environments. We have used Docker[1] for that purpose. All startup instructions were attached in a Dockerfile and executed by a bash script to allow the user to start and stop the complete application with simple single-line commands.

6.2 Separation of Concerns

Because we had two concerns in our application, the management of data (loading of data, preparation, and sanitizing) and the visualization, we opted to separate them into two standalone applications. In consideration of our rapid startup principle section 6.1, we selected the Vue.js[2] framework for the visualization part and the Flask[3] framework for the data management part. Both frameworks provide simple servers that either react to change automatically or allow graceful shutdown and quick startup. Both frameworks use simple and readable scripting

[1] https://www.docker.com/

[2] https://vuejs.org/

[3] Flask web framework https://flask.palletsprojects.com/en/2.1.x/

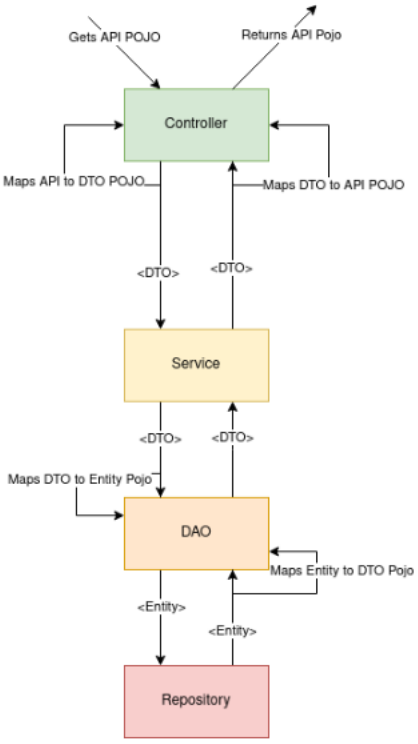


Figure 6.1: Basic depiction of the layer's pattern applied to our tool in the back-end

languages. We achieved communication between those two applications via RESTful [4] communication.

6.3 The Pattern of Layers

Working in increments requires the software to stay flexible. To acquire this flexibility, this work used the layer's pattern [2], as demonstrated in Figure 6.1. We applied the layer's pattern to separate different levels of abstraction or responsibilities in individual layers, with an obvious control flow. Higher levels of abstraction only call for lower levels of abstraction or access services within the same level of abstraction. We separated these layers by clearly defined interfaces. This separation makes the software more structured and allows components or entire layers to be exchanged or modified. We enforced this pattern in the back-end and the front-end application. Since we knew the algorithm pipeline before implementation, the back-end application was less prone to change. The presentation side of the data was in a more prototypical state and much more predisposed to spontaneous changes. Due to the separation of concerns, see section 6.2, this didn't prove to be a problem. We treat the data management part and visualization parts differently.

[4] https://en.wikipedia.org/wiki/Representational_state_transfer

6.3.1 The Publish-Subscribe Pattern

Due to a similar need for flexibility in the front-end and back-end, we added the publish-subscribe pattern [2] to our front-end. The intention behind this pattern was that there should be a single place in the tool responsible for storing the current state of the data. We have a set of consumers, our views, which depend on this data. With every change we introduce to this central store, we want each consumer to know about this change. Each will decide if they propagate this change and convert it into a modification in the visualization. During the design phase, see section 4.4, this pattern proved to help treat each view separately. Both the type and the number of views overall could change during the course of the implementation. It also became clear that a change in data would need to update potentially multiple consumers at once. For that reason and to reduce duplication, we introduced the publish-subscribe pattern.

6.4 The Single Responsibility Principle

We developed each view and service, in the back-end and front-end, with the single responsibility principle in mind. Like the patterns before, this served us to enhance the flexibility of the software and to react quickly to changing user demands or changes in the tool due to other unexpected circumstances. It states that every service or class should only have a single responsibility within the software.

6.5 Architecture

Figure 6.2 depicts the basic architecture of the application. The complete solution is running in two docker containers. We start both containers with a simple docker-compose[5] file. The docker-compose file contains configuration to access the container specifications of the back-end and front-end applications and create separate containers for them. We control actions in the docker-compose file by simple bash scripts. The scripts help us install all the prerequisites for the tool and the start, shutdown, and purging of the docker containers. Initially, we created this setup to support quick feedback loops and short development iterations. In Figure 6.3, you can see the folder structure of the complete solution. The complete solution is publicly available[6].

In the following two sections, see section 6.6 and section 6.7, we will go into more detail about the visualization, the front-end, and the data management implementation, the back-end.

[5] https://docs.docker.com/compose/

[6] Online tool repository https://github.com/ratzenboe/topoclust

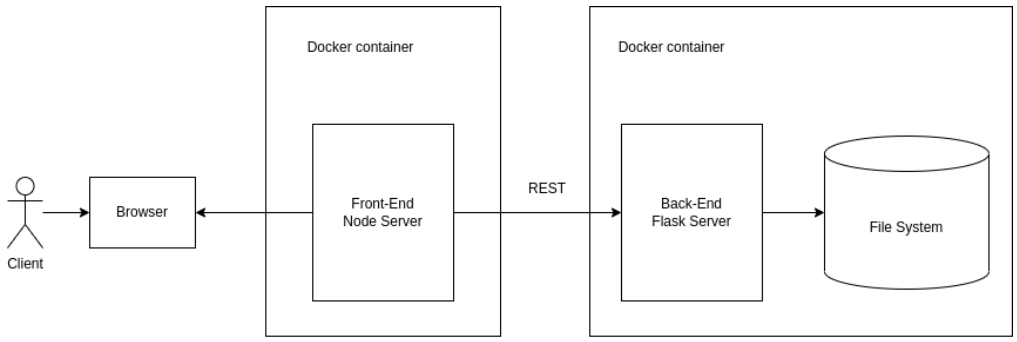


Figure 6.2: Layout of the overall architecture

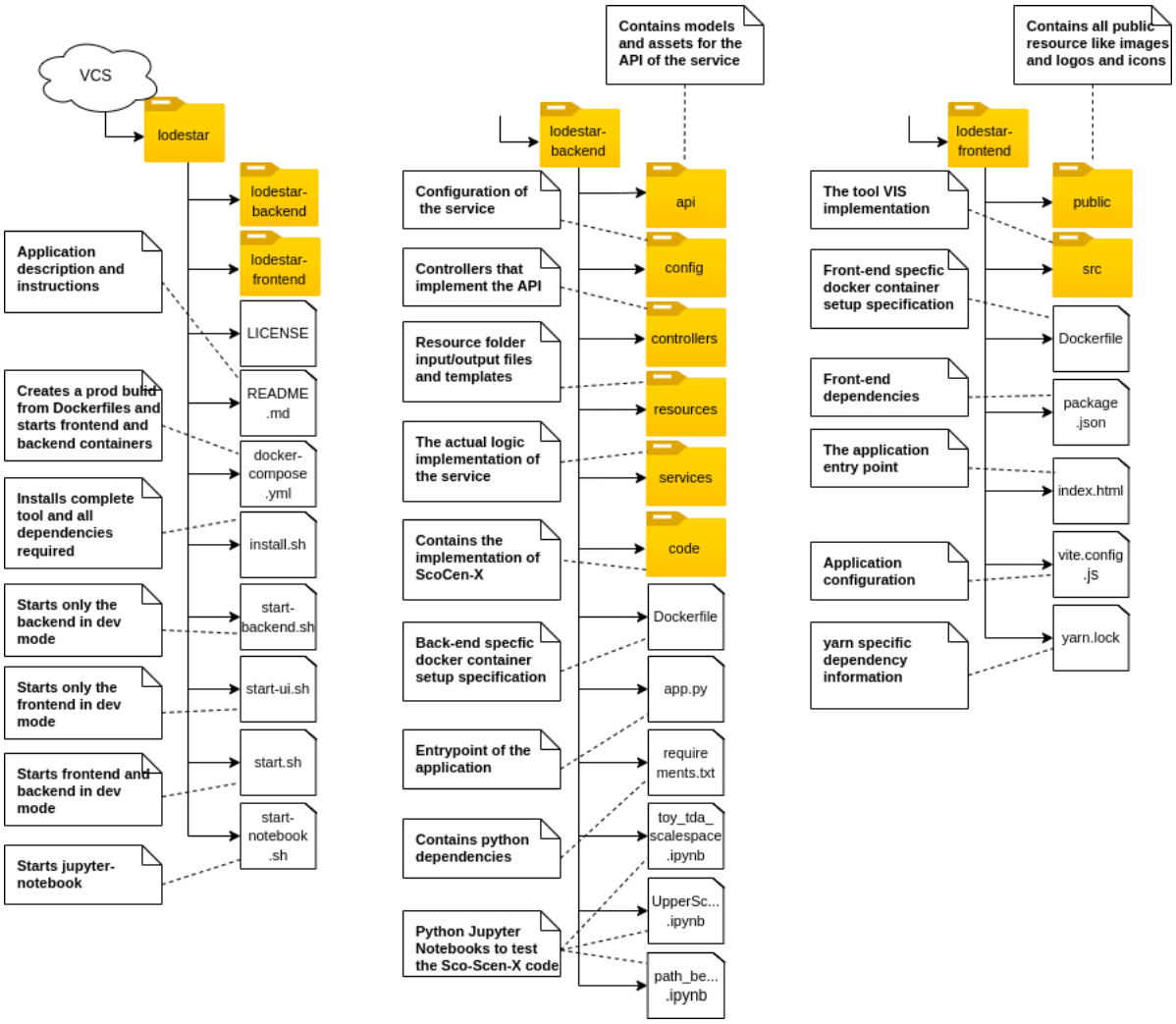


Figure 6.3: The folder structure of the application

6.6 Front-End

This work used Vue.js as the front-end framework and Typescript[7] as the scripting language. We leveraged the VueX store[8] to store current states and to propagate changes to all the views that subscribed to this data. To design each dashboard view, we made use of two graphing libraries, D3.js[9] and plotly.js[10]. We manage the dependencies with and build the application by using yarn[11].

This setup allowed for rapid prototyping because Vue.js offers a lot of standard components that can be reused and changed. While prototyping, it is advantageous that the HTML, Javascript, and CSS code is structured within the same file. By introducing back-end mocks in the early stages of development, we have eliminated long waiting times, either by the algorithm itself or by some back-end calculations. This further decreased iteration duration and allowed for quick changes. We removed those mocks later to integrate the front-end with the back-end application successfully. The back-end server mocks supported us in creating the initial layout and views.

6.6.1 Update Views

We introduced the publish-subscriber pattern to notify each view about significant changes to the data. Each is, in essence, of the same generic type **View**, and each is a publisher and a possible subscriber to data. Because not every view is interested in view updates, or updates in the central data store, we selectively subscribe to a narrow slice of the overall data. This proved to be beneficial to the overall performance of the application.

The responsibility of the front-end application is to host our set of visualization techniques and views. It will be responsible for requesting data from the back-end application via REST and communicating updates to all other views related to the changes. It will also hold additional data unrelated to the algorithm clustering to support visualization features.

6.7 Back-End

The main concern of the back-end application is the hosting of scientific data. It will be responsible for making that data available via a defined REST interface and is ignorant of specific implementations of the clients. It will host a variety of REST endpoints to request additional

[7] https://developer.mozilla.org/en-US/docs/Web/JavaScript
[8] VueX store https://vuex.vuejs.org/guide/
[9] D3.js - Data-Driven Documents https://d3js.org/
[10] Plotly Graphing Libraries https://plotly.com/javascript/
[11] https://yarnpkg.com/

transformations of the provided data, and the back-end will also perform precalculations for data that is static for a specific data set and the provided repository.

In this context, a repository is a directory containing multiple files. Each file will adhere to the .csv file format and contains multivariate data in a row-column relationship. The back end is implemented with Python[12] and leverages the Flask web framework to create a web server that operates on these repositories.

6.7.1 Features Provided by the REST API

The back-end web server supports multiple REST endpoints to receive specific data slices calculated and stored. To have a good overview of the currently supported endpoints, we used the apispec[13], flask-apispec[14] and apispec-webframeworks[15] libraries to document the currently available endpoints via swagger-ui[16]. The result of this documentation can be seen in Figure 6.4, and the endpoints are grouped by domain. Each endpoint is versioned, so they remain flexible in the future and support different versions of the same endpoint. When the application runs, you can access this information by navigating to the subdomain /swagger-ui/.

6.8 Performance and Preprocessing

The application utilizes a strategy to enhance the runtime performance of solution space exploration by storing the clustering results for a given alpha value and density level range in temporary files. By storing the clustering results in temporary files (see Figure 6.5), the application can quickly access and utilize the results for future solution space explorations without having to perform the calculations again. This approach improves the exploration performance but results in a longer initial algorithm runtime. However, this is not a significant issue for astronomers. During the prototyping phase, we determined that they prefer longer initial calculations and fast solution exploration over the reverse.

Subsequently, to further improve runtime performance, the application also stores the current modifications and selections in session files. This means we can separate the initial run on some data from the actual inspection and modification of that data. Since our tool keeps track of changes in session files, the application can quickly access and resume the current session without repeating the previous steps. Overall, caching the level files, sessions, and density trees greatly

[12] Python programming language https://www.python.org/
[13] https://pypi.org/project/apispec/
[14] https://pypi.org/project/flask-apispec/
[15] https://pypi.org/project/apispec-webframeworks/
[16] https://swagger.io/tools/swagger-ui/

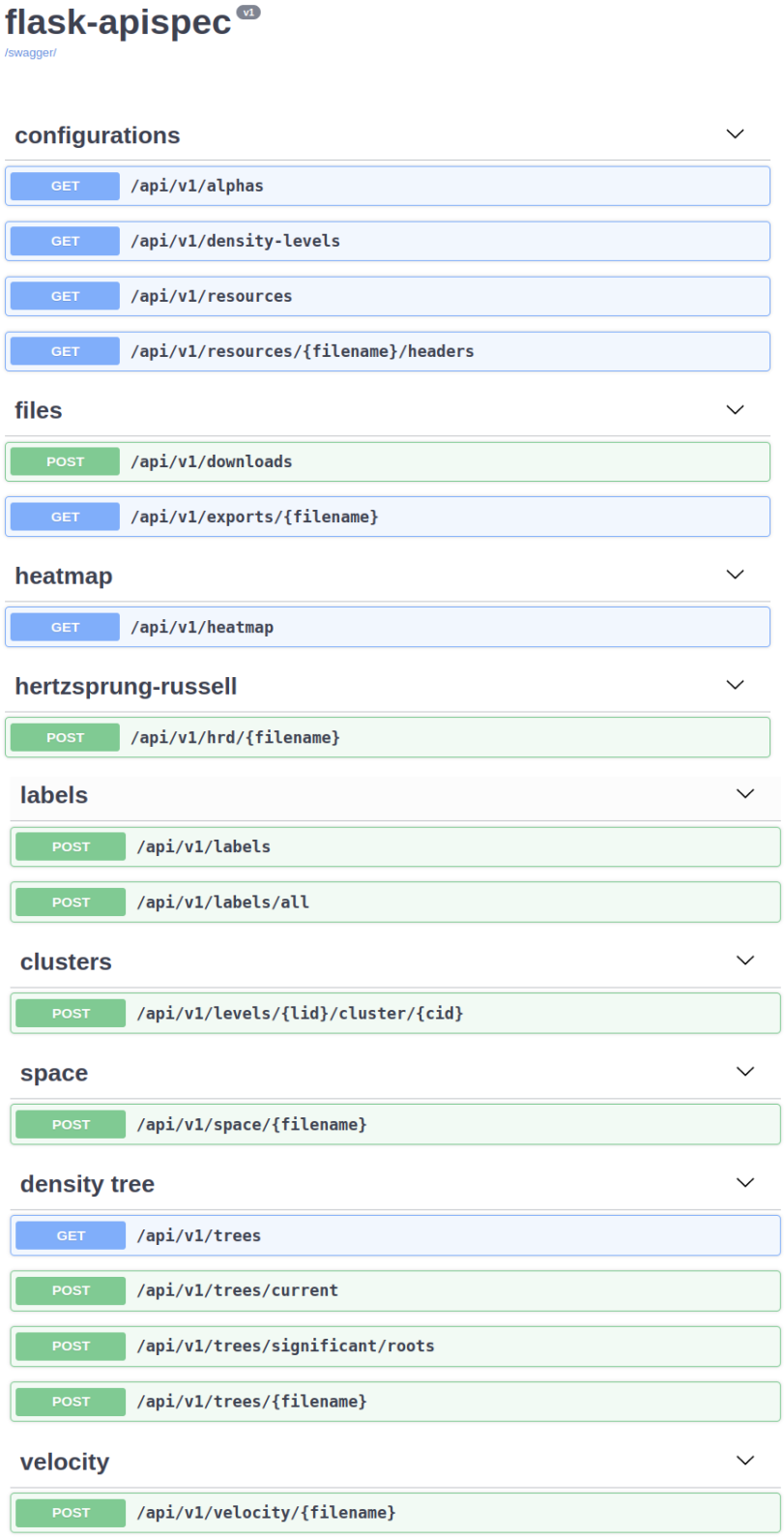
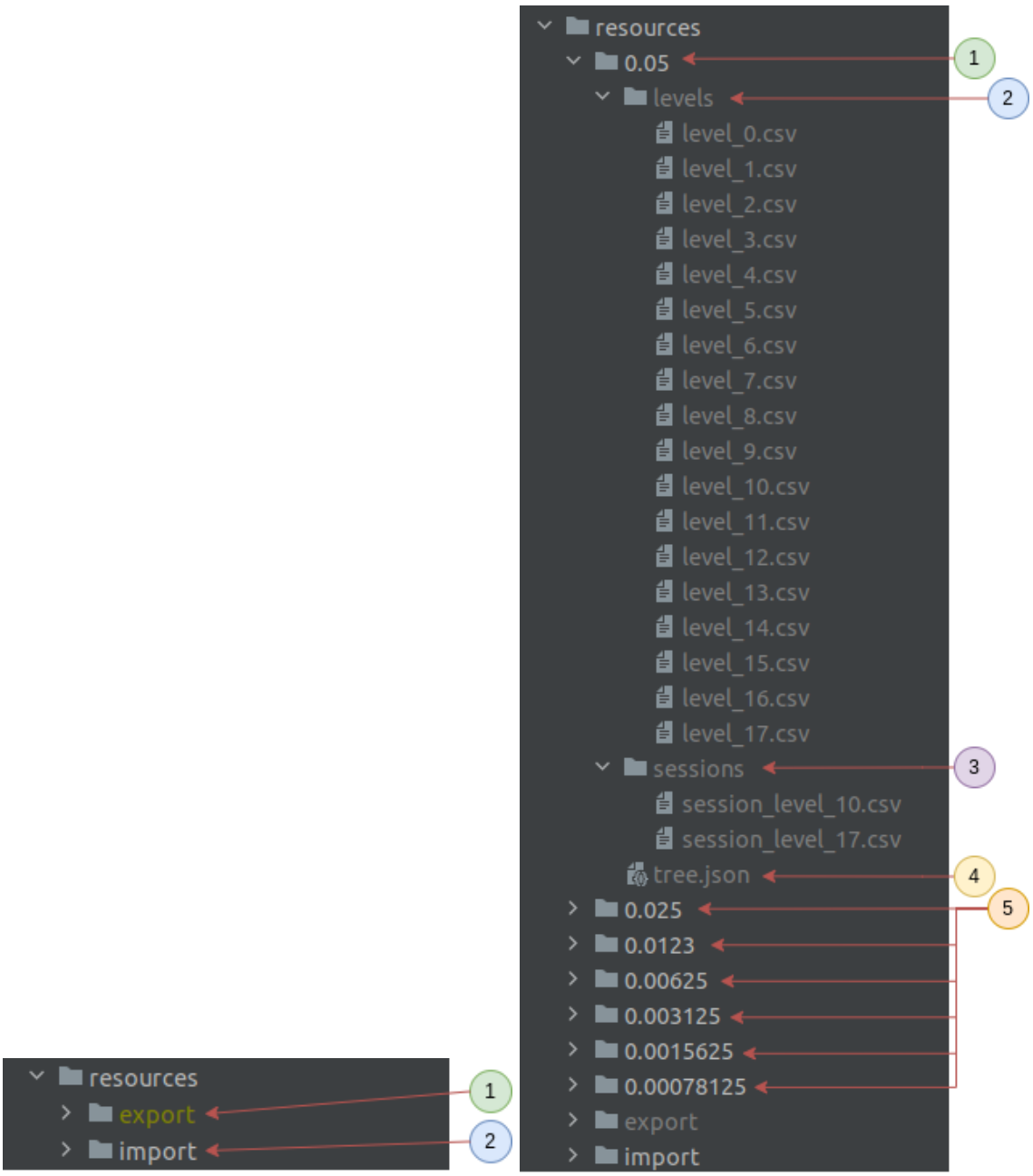


Figure 6.4: Endpoints supported by the back-end Python server used by the front-end to generate the graphics and features.

improves performance.

In summary, the application uses a combination of temporary file storage, session files, and caching to enhance the runtime performance of solution space exploration while accepting a long initial calculation step.



(a) Before the algorithm has run

(b) After the algorithm has run

Figure 6.5: (a)(1) Is the export folder containing any downloaded labeled clustering files. (a)(2) is the import folder containing all available resource files. If a new one is uploaded, it will be stored there. (b)(1) is a folder for a specific alpha value. (b)(2) is the folder that contains the calculations for each density level at the current alpha value. Currently, 18 density levels. (b)(3) is the sessions folder containing the level files the user has interacted with. Only the columns that can be changed are stored within. (b)(4) is the cached density tree for the current alpha value containing all density levels as a network. (b)(5) are the folders for the remaining alpha values, configured within the back-end and dynamically generated, the same structure as (b)(1).

Chapter 7

Evaluation

In this chapter, we present the results of our evaluation of the proposed system. The evaluation is divided into two main parts: a formative evaluation and a summative evaluation. The formative evaluation was conducted to gather feedback and improve the system during the development process. In contrast, the summative evaluation was conducted to assess the overall performance of the final system. The formative evaluation section describes the methods for gathering feedback, such as user interviews, surveys, and usability testing. The results of the formative evaluation are presented and discussed in terms of the changes and improvements made to the system based on the feedback received.

In the summative evaluation section, we discuss the results of the SUS questionnaire regarding the system's effectiveness, efficiency, usability, and satisfaction.

Overall, this chapter provides a comprehensive evaluation of the proposed system, highlighting its strengths, and we will use the results of this evaluation to guide future work and improve the system.

7.1 Formative Evaluation

Throughout the design process, we have followed a structured approach of task analysis, low-fidelity prototyping, and high-fidelity prototyping. During the low-fidelity prototyping phase, we regularly met with two visualization experts and the algorithm pipeline developer to create the first version of a pen-and-paper low-fidelity prototype. This step was accompanied by interactive wireframes (AdobeXD) to experiment with navigation alternatives and novel 3D representations. The prototype was presented to three female astronomy researchers, one Ph.D. student, and two Master's students. The collected feedback was the basis for a second version of the low-fidelity prototype, presented approximately two weeks after the first session. The

same group of astronomers and an additional male Ph.D. challenged this prototype as well. The combined feedback eventually led to a high-fidelity prototype.

During the high-fidelity prototyping phase, eight members of the astronomy community gave feedback, four men and four women. Four have already provided feedback during the low-fidelity prototyping phase, and another three have joined during the high-fidelity prototyping. The eighth person was the algorithm developer. We conducted personal interviews with each participant. Each session concluded with a SUS questionnaire and feedback form. During the interview, we also made notes on user behavior and on issues and questions that arose during the meeting. We collected these notes with pen and paper and digitized them afterwards. The results of the SUS questionnaire can be seen in section 7.2. We have used the interval to perform iterative improvement of the overall system. The feedback form served as a source to create items for improvement. After we prioritized them, we implemented high-priority entries between interview sessions.

Overall we have received very positive feedback, and participants were looking forward to using the system. The feedback form and personal feedback provided great insights into the behaviors and needs of the test users. We have used this information to improve the system.

In the following subsections, we will go through the results of the low fidelity user tests subsection 7.1.1, feedback forms subsection 7.1.2 and the results of the personal interviews subsection 7.1.3 and how these results influenced our design decisions.

7.1.1 Low-Fidelity User Test Results

The low-fidelity prototype user tests provided valuable feedback. We discovered issues that otherwise would likely have surfaced during implementation. By receiving this feedback early, we could address them before high-fidelity prototyping. Not only did this save resources, but improved development speed. The list of all improvements is long, so we decided to list representative examples in the subsequent paragraphs. While we presented the pen-and-paper prototype to the test group, we received multiple feature requests. Three out of eight participants who tested our validation mode requested additional velocity histograms to study the respective distributions in that cluster. Four participants wanted an HRD plot to support cluster validation. Additionally, due to changes in the algorithm pipeline, the noise slider encoding (see Figure 7.1a and Figure 7.1b) became obsolete. Subsequently, we removed the noise slider in Figure 7.1c, the high-fidelity prototype, and added a Hertzsprung-Russel diagram instead. The evolution of the cluster details can be seen in Figure 7.1. We applied a similar change to the exploration mode, adding an HDR and a configurable feature scatterplot for an entire clustering solution.

Its evolution is depicted in Figure 7.2.

Since we can distinguish only a small range of colors comfortably, we realized that our original color palette would not suffice. Our prototype would assign one of these colors from the color palette to a sub-tree in the solution space. In case there are too many, we run out of colors. In a collaborative effort, we found a solution, discussed in more detail in subsection 5.1.7, and we can see an example of the current color assignment in Figure 7.2c.

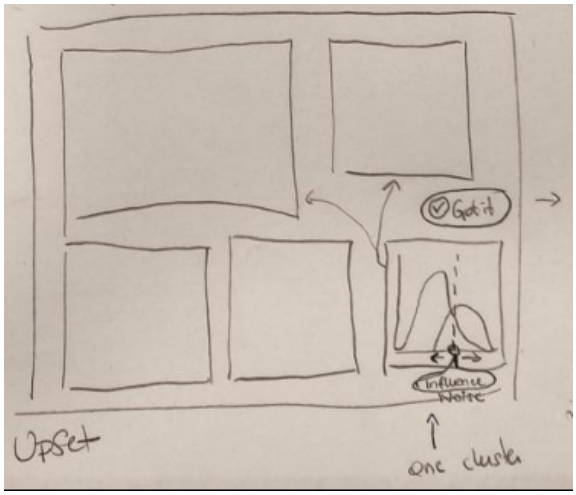
For subsequent processing and evaluation in other tools, it is beneficial to name clusters with alphanumeric sequences instead of numbers. Two users pointed this out during our version two prototype review. The final implementation is depicted in Figure 7.3. When a user exports a result file, the tool will display that name in the affected rows (seen in Figure 7.1b).

Most notably, during the prototyping phase, we identified the topics we should address in a tutorial section. In the conversations, we observed that some views and the input parameters were unclear and needed more introduction.

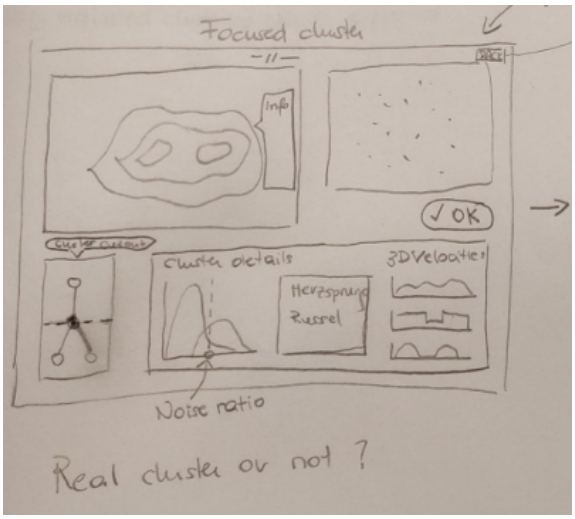
7.1.2 High-Fidelity Prototyping: Feedback Form Results

The feedback form provided valuable insights into specific issues of the tool. It included feedback on topics we didn't discuss during personal interviews and vice versa. From the test user group, the most requested addition was a zoom function in the evaluation views, which was promptly implemented. Two users asked for an improved explanation of density level and alpha value, leading to an expanded tutorial section for better understanding. Additionally, we enhanced the tool by displaying cluster information when hovering over a cluster, as per two requests received. Another user suggested improving color coding as colors were not distinct enough, which influenced the color coding implementation we applied in subsection 5.1.7. To continue an active cluster exploration session, one person requested to keep the session active unless the tab or browser is closed, which was added to the tool's functionality. A stability view was also added in response to a comment from one user to address the non-linearity of parameter changes and make it easier to understand their effect.

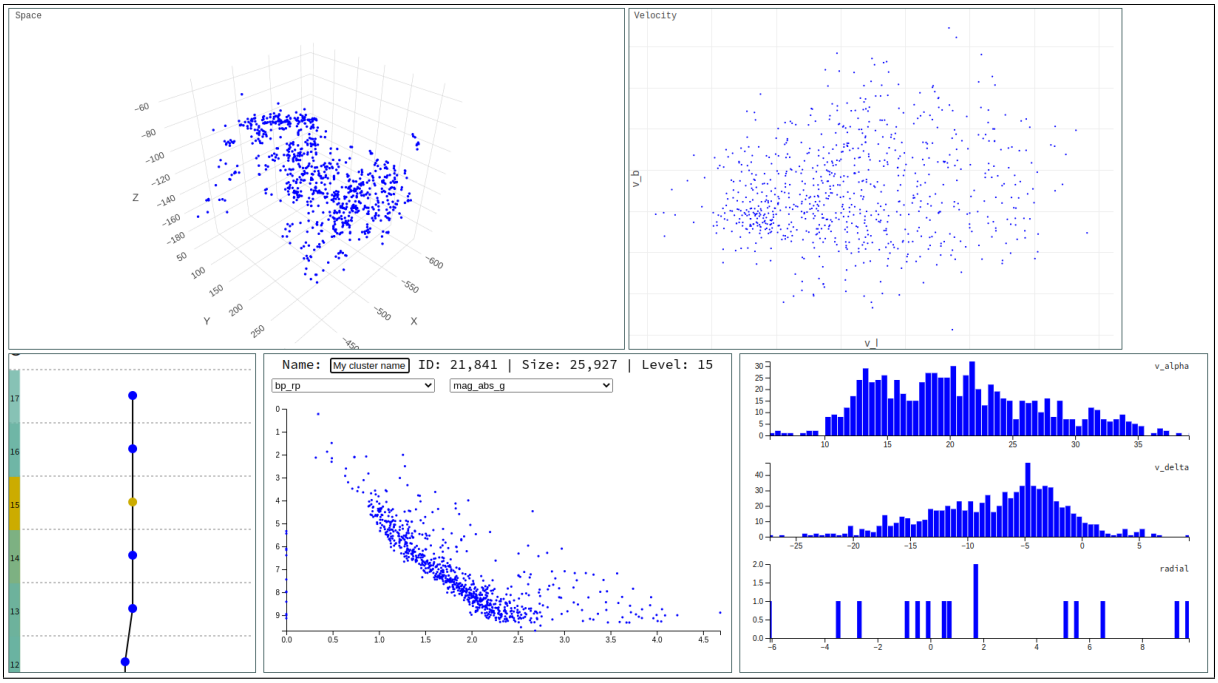
While we addressed most requests, there were also change requests to inspire future work. Two participants wanted the ability to plot and compare the clustering results of two unique pairs of alpha value and density level. One participant wanted to exclude some clusters from the clustering results. Another participant suggested adding expressions to one axis to plot multiple columns on one axis. Lastly, we received a suggestion to differentiate between more clusters by using an encoding that combines colors with signs, allowing for a large set of unique identifiers.



(a) Validation mode for low-fidelity prototype version 1

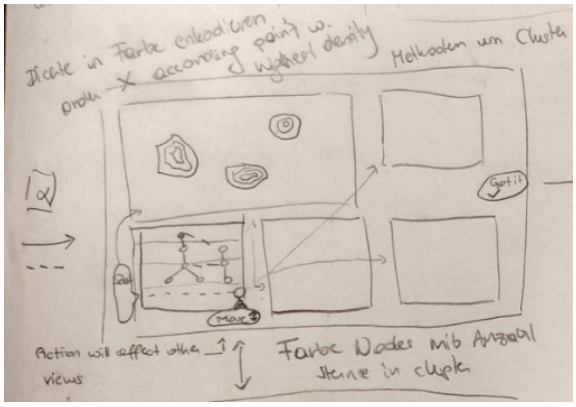


(b) Validation mode for low-fidelity prototype version 2

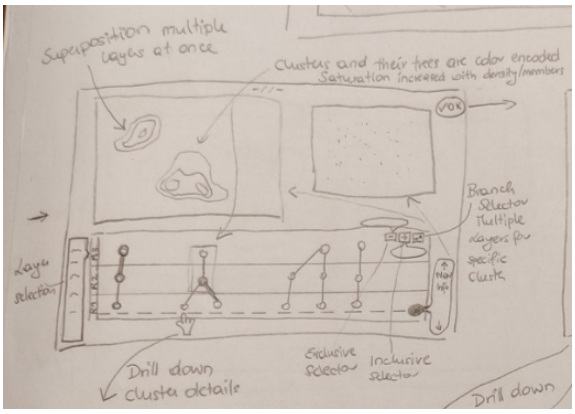


(c) Validation mode for high-fidelity prototype

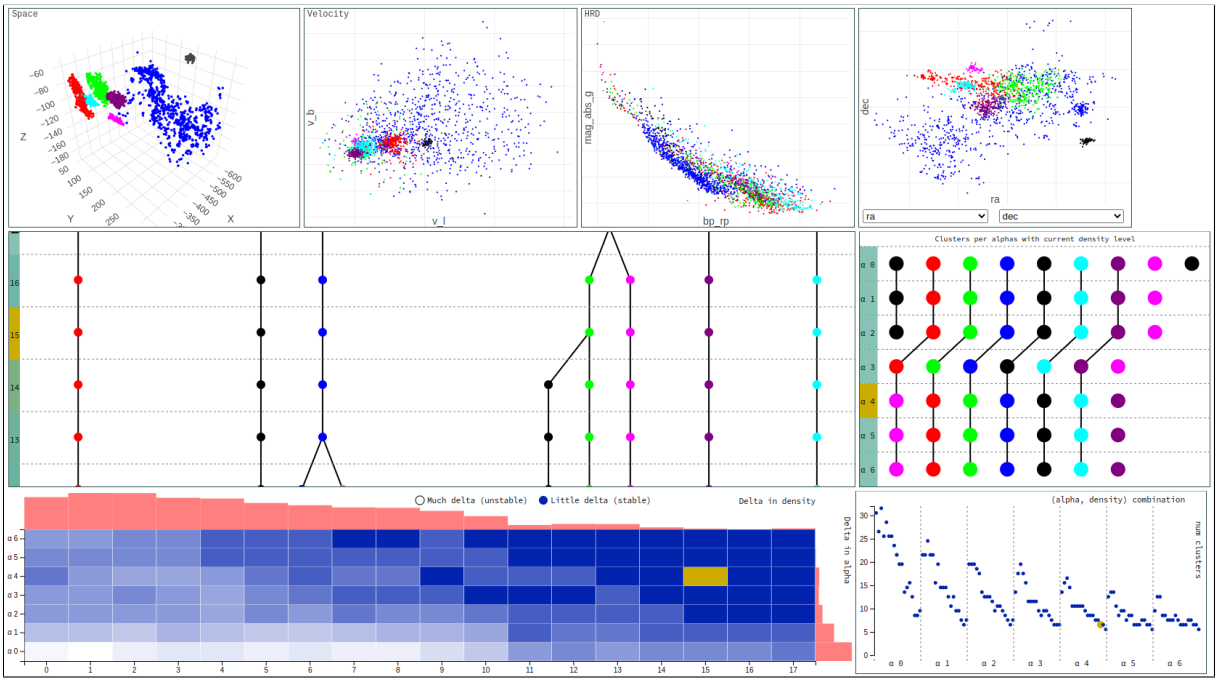
Figure 7.1: Evolution of the "Validation Mode" interface design in prototyping.



(a) Exploration mode for low-fidelity prototype version 1

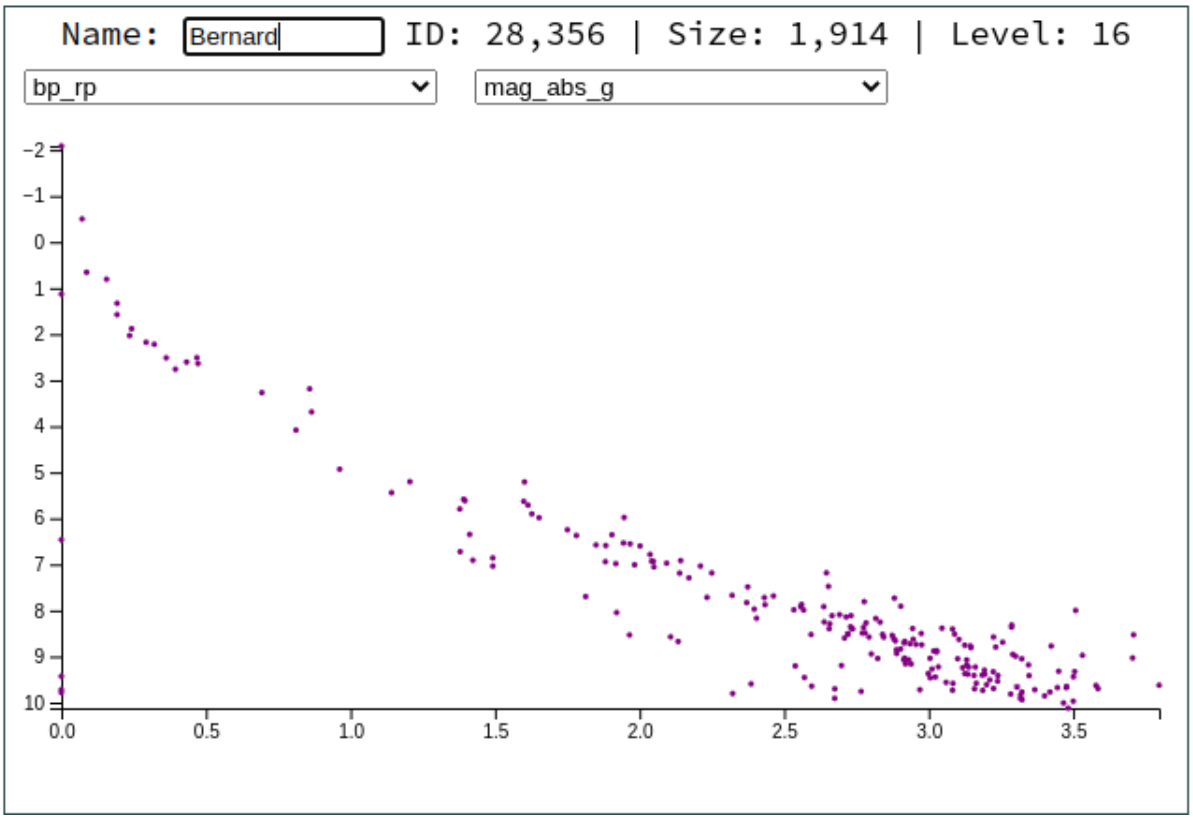


(b) Exploration mode for low-fidelity prototype version 2

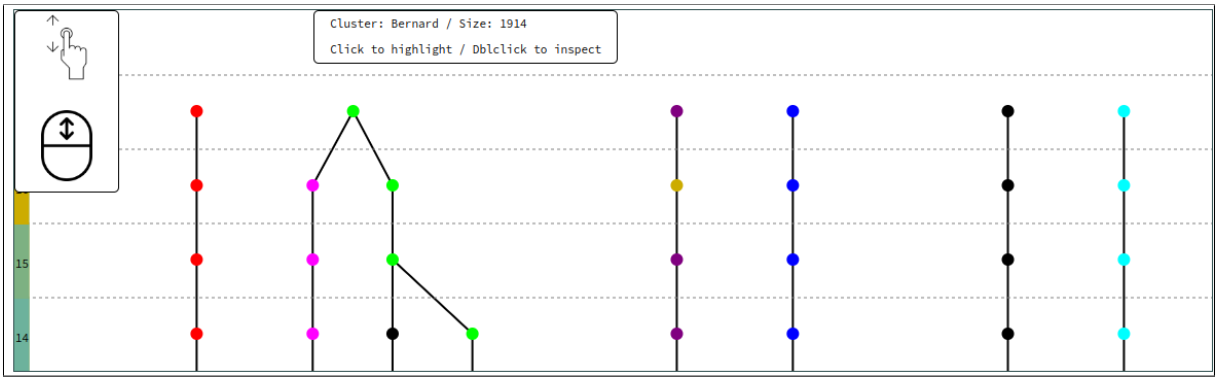


(c) Exploration mode for high-fidelity prototype

Figure 7.2: Evolution of the "Exploration Mode" interface design in prototyping.



(a) Cluster validation mode allows naming a cluster.



(b) This name is stored and will appear in the exploration mode.

alpha	v_delta	Unnamed:	Unnamed: 0	labe	custom_lab	custom_label_nam
6.40528068025199	-8.84392048734039	19951	19951	28356	28356	Bernard
6.40011352015375	-8.46227464734249	19953	19953	28356	28356	Bernard
6.238279010671	-8.76441711487489	20046	20046	28356	28356	Bernard
5.95342427938584	-8.75019214317436	26756	26756	28356	28356	Bernard
3.52610186268422	-13.7548219931324	26762	26762	28356	28356	Bernard
6.36105069163801	-8.71891315097294	26852	26852	28356	28356	Bernard
6.27186625756089	-9.06366375992121	26863	26863	28356	28356	Bernard
6.8751392006321	-9.00542930554527	26865	26865	28356	28356	Bernard
5.47957673308428	-8.3448748512784	26871	26871	28356	28356	Bernard
6.29146442914237	-8.86922944523779	26888	26888	28356	28356	Bernard
15.441514592216	-9.52942835759982	27251	27251	28356	28356	Bernard
3.43283868324512	-8.06233753547159	27252	27252	28356	28356	Bernard

(c) When the file export is executed, the name is added to the affected rows.

Figure 7.3: Workflow of the cluster naming feature.

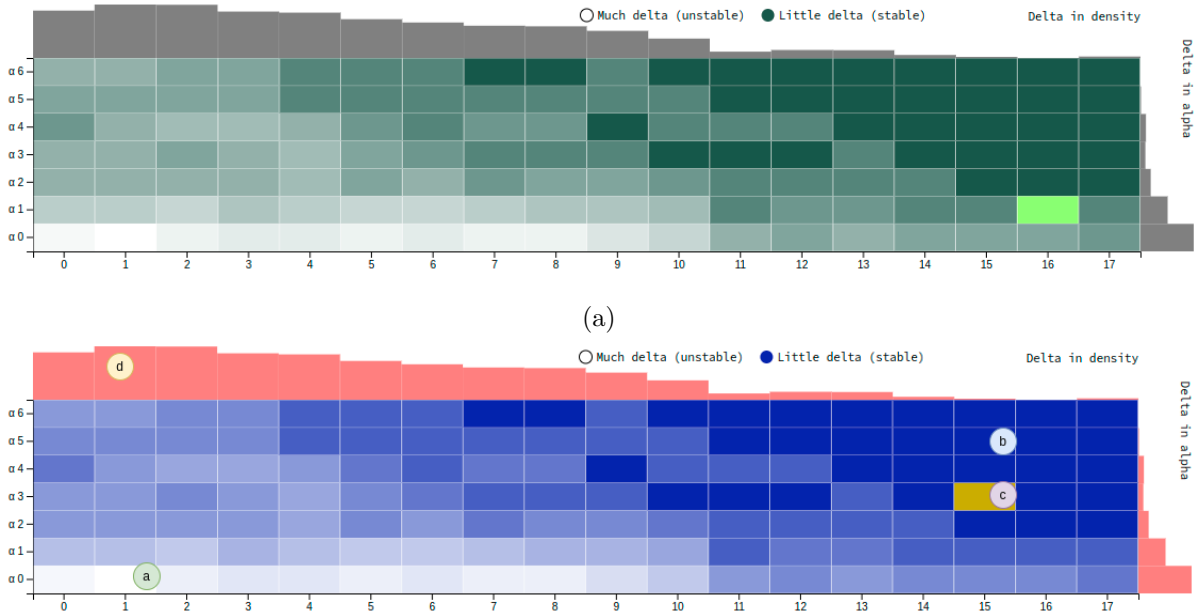


Figure 7.4: (a) The view before we got the feedback. Grey and green can easily be confused with each other. (b) Usage of colors that can easily be distinguished.

7.1.3 High-Fidelity Prototyping: Personal Interviews Results

We have gained valuable insights from the feedback forms, SUS questionnaires, and private participant interviews. One important lesson we learned was the importance of accessibility for color-blind users. Our tool's heat map, which encoded stability as a range between white and dark green and used gray bars in histograms, was confusing for a participant who was partly color-blind. We modified the heat map to accommodate color-blind users better, as seen in the Figure 7.4b.

Personal conversations can provide valuable insights when combined with objective measurements such as the SUS. Between user test sessions, we noticed that we introduced more features in this period than in previous development periods. Additionally, the number of improvements between each interview decreased in subsequent periods. By comparing the feedback for each user test, we found that implementing those features correlated with less feedback, suggesting that our incremental approach led us to focus on the right problem areas, resulting in fewer feature requests and an increase in overall verbalized satisfaction with the tool. The SUS scores, however, remained stable.

Question	$\overline{x}$	σ
I think I would like to use this system frequently	4	0.58
I found the system unnecessarily complex	1.71	0.49
I thought the system was easy to use	4.29	0.49
I think that I would need the support of a technical person to be able to use this system	2.14	1.07
I found the various functions in this system were well integrated	4.29	0.76
I thought there was too much inconsistency in this system	1.57	0.79
I would imagine that most people would learn to use this system very quickly	4	0.82
I found the system very cumbersome to use	1.71	0.58
I felt very confident using the system	3.71	0.49
I needed to learn a lot of things before I could get going with this system	2.43	0.53

Table 7.1: SUS questionnaire with the mean and standard deviation result per question. The three results that deviate most from an optimal rating are highlighted in orange.

7.2 Summative Evaluation

With a final score of 76.79 and a standard deviation of 7.60, we consider the tool as "Good." Equivalently it would receive a B grade. To interpret the score, we have used the work of Bangor, Kortum, and Miller [3]. It explains the letter grading interpretations and a table to assign a human-readable version to the score.

Interestingly, two test users gave the tool a rating of 87.5, which would refer to the adjective "Excellent," while the ratings of all other users were well within the range of a "Good" rating.

We will look at the average rating of each question to reason about their meaning in chapter 9. Table 7.1 is a table with all the questions listed and their average value. The most noteworthy three questions and their average results are highlighted in orange. In this context, noteworthy refers to the questions that deviated the most from an optimal rating. We only focus on these three of the ten questions to find effective ways to improve the overall rating in possible future work.

Chapter 8

Lessons Learned and Future Work

In this chapter, we will discuss the lessons learned in section 8.1. In section 8.2 we look at key takeaways and outline potential areas for future work.

8.1 Lessons Learned

In this section, we want to highlight the key lessons learned from the research such as the importance of user feedback, the benefits of an iterative development process, and the need to consider performance when developing a system.

8.1.1 Personal Collaboration over Standardized Questionnaires

Especially during the low-fidelity prototyping phase, we found that personal interaction is more valuable to get good feedback than standardized tests. While standardized tests have the advantage of being standardized, comparable, and structured, the unstructured and spontaneous nature of a personal interview provides creative or unintended feedback. These interviews developed into much more of a workshop where solutions were discussed and found together. All the individual behaviors were registered and noted to record any information that a questionnaire might not capture. For the final user tests and future work in general, we decided on a mixture of personal interviews and standardized feedback questionnaires.

8.1.2 User Test Group in Iterations

Because we used each iteration to address the user feedback, we initially hoped to see a positive trend with every subsequent SUS result. But plotted on a bar chart (see Figure 8.1), the scores don't improve over time. But even if the results showed a positive trend, it would not qualify as proof that the tool's usability actually increased. After an iteration ended, we performed a

SUS questionnaire with a single user instead of a user group. This poses two problems. First, scoring preferences, expectations, experience, age, gender, or impairments, among other reasons, can affect a person's score. Therefore, one would normally look for large user test groups to reduce the effects of outliers. But we were limited by the low number of potential users we had access to. Second, knowing the former, we should have asked the same person in subsequent iterations to identify a positive trend, which could cause other problems. Better yet, we should have concluded each iteration with a SUS questionnaire targeted at a new test user group, with the same size and composition but composed of people that did not use the tool before.

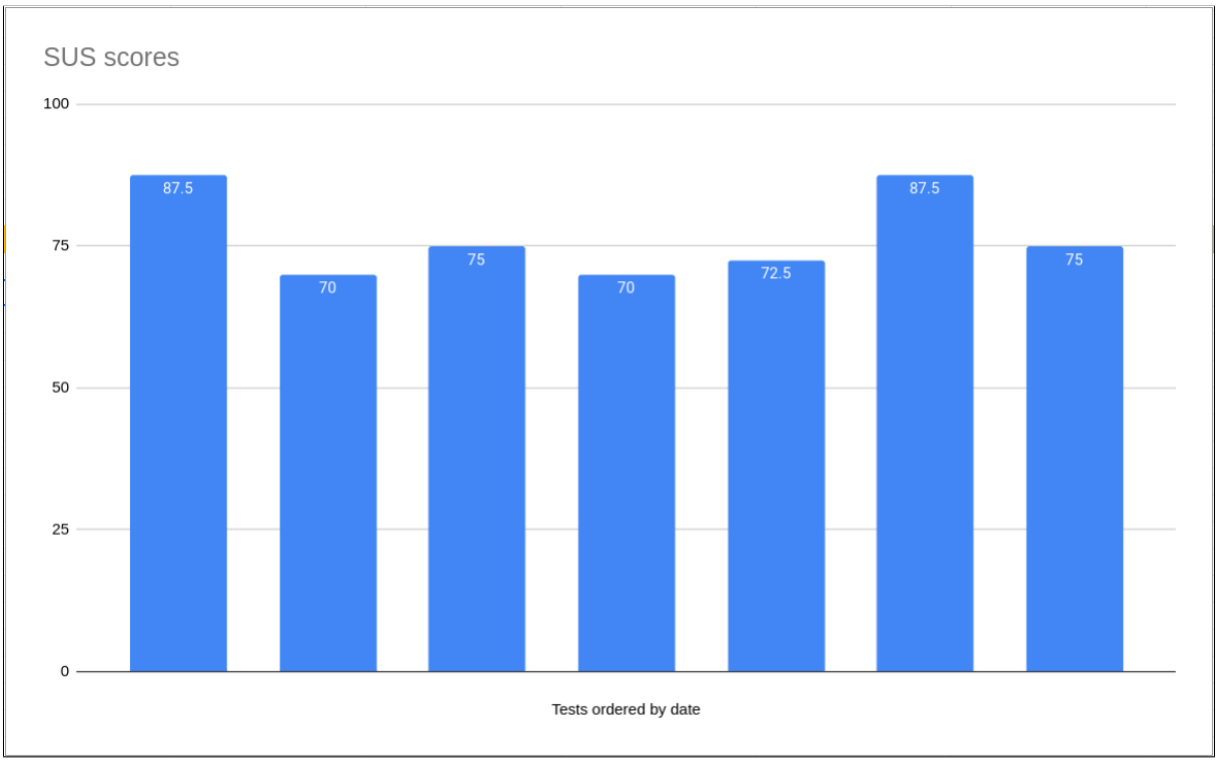


Figure 8.1: SUS scores, ordered by date.

8.1.3 Incremental Development

Especially towards the end, during the user test phase, this approach proved to be productive. Most bugs, but especially noteworthy, most improvements were made before the next interview. The written feedback proved especially valuable in combination with the notes we made in the personal interviews.

8.1.4 Performance

As the tool was being implemented, it became evident that strategies were required to address its inefficiency and improve its slow execution time. Several bottlenecks greatly affected the

runtime of the overall tool. D3.js proved to be a bottleneck while rendering the results from the algorithm, particularly when we used <svg> DOM elements to render our results. Switching from <svg> to <canvas> elements and slightly changing the rendering code, we improved the situation considerably. Another improvement in the performance was possible by switching from D3.js to plotly.js. Plotly proved to be a scalable and fast library for simple scatter plots. And it provides many additional useful features out of the box.

We could have prevented this issue since the performance was a known challenge working with astronomy data. We conjecture that the work would have progressed faster if we had defined non-functional requirements [35] and by making them part of the methodology. In future work, we proactively want to add non-functional requirements to the literature review. Thus, having the information to identify threats to the non-functional requirements much earlier.

8.2 Future Work

In this section, we will reflect on the challenges faced during the research and discuss how these experiences have informed our understanding of the problem at hand and how these challenges could be addressed in future work.

Table 7.1 can help us understand which aspects of the tool could still be improved. We have highlighted the three noteworthy scores in orange. The question "I think that I would need the support of a technical person to be able to use this system" and its rating suggests that users could still benefit from an onboarding section or that the tutorial page still needs improvement. The same could be true for "I felt very confident using the system" and might suggest that the views in the tool are not clear enough and could benefit from additional labeling or helpful tooltips. In combination with the third question, "I needed to learn a lot of things before I could get going with this system," we suspect that the tool's functions are not clear enough and should be improved. This could improve by combining a tutorial run, introducing the most important steps, with clear labels for each view, and clearly stating the intention at the beginning of the page.

In the following, we have defined concrete areas of improvement.

8.2.1 Session Concept

This strategy will also allow for extending the tool with a user session feature in the future. The generated files, and the produced session files, could be stored long-term and be linked to a specific user. This feature would enable users to resume a session even if the browser or tab was previously closed, and it would provide the capability to handle multiple algorithm executions

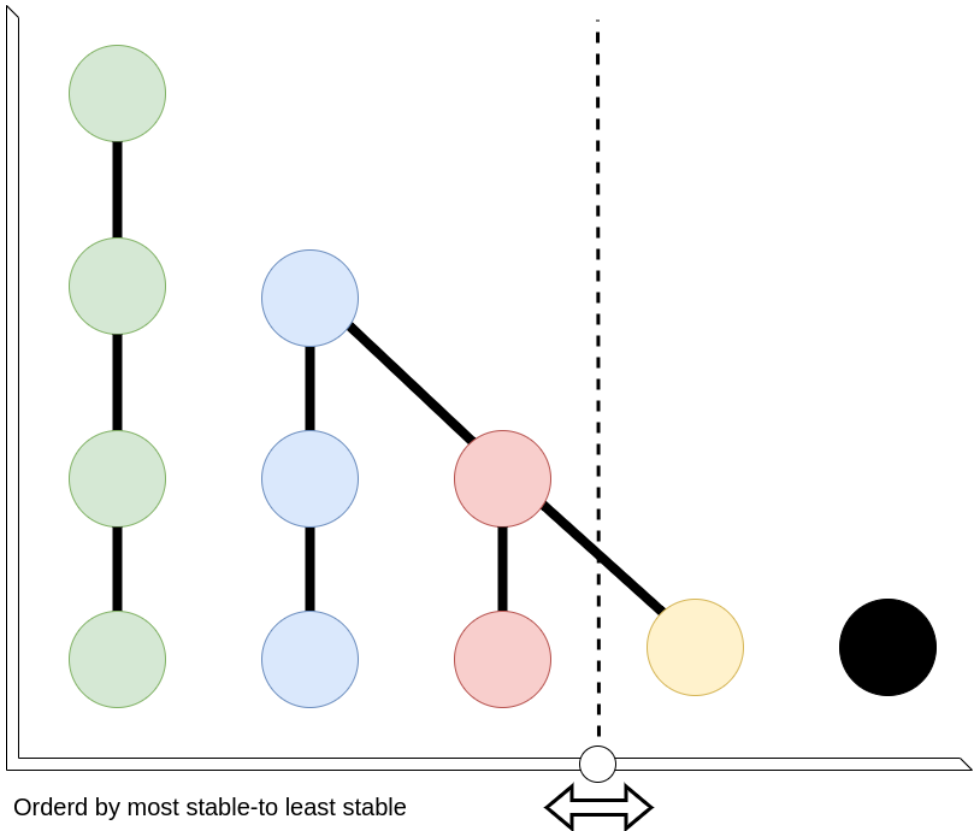


Figure 8.2: Example of a density bandwidth exploration tree. Each sub-tree is ordered from most stable to least. A slider is located at the bottom, controlling the excluded clusters.

concurrently. A single run is time-intensive, and allowing users to store and revisit them at a point in the future would increase usability and user satisfaction.

8.2.2 Sorted Graph

In the following, we discuss possible avenues to improve the graph-based cluster development view, as the current design can result in overlapping merge paths and does not provide means to filter short-lived sub-trees. The sub-trees could be sorted by their stability value, leading to a graph where the most stable clusters are on the left-hand side and the least on the right-hand side. By introducing a range slider, users could clip the sub-trees and decide which sub-trees to ignore. We believe this change would help users to explore the solution space more effectively and rapidly remove clusters from the solution space. A schematic example is seen in Figure 8.2.

8.2.3 Open Source Libraries

Our findings suggest that the clustering community would highly benefit from a framework that supports generic approaches and domain-specific approaches by aggregating those generic modules into a domain-specific application.

One way to tackle an issue like that could be an online artifact repository managed by an open-source clustering community with libraries of generic back-end and front-end modules. In this scenario, all modules must satisfy a globally defined interface pattern and can be registered and unregistered in the back-end and the front-end application via publish-subscribe. We could imagine something like that as a toolbox or an open-ended box of Lego pieces. We would use those to build specific structures to tackle complex domains, such as the astronomy domain. That way, we could combine the advantages of building generic clustering algorithms and visualization strategies and enable developers to leverage those generic algorithms and visualization tools and combine them into a domain-specific clustering visualization. How to approach this exactly would be a topic for further research.

8.2.4 Two Parameter Tuples Side-By-Side Comparison

As the work of Kwon, Eysenbach, Verma, Ng, Filippi, Stewart, and Perer [48] already suggests, allowing domain users to explore the effects of parameter changes on the clustering result and comparing the different results can lead to a better understanding of the results. Considering that, we assume that a domain-specific visualization and in-depth exploration of the solution space could aid potential users in understanding the effects of the input parameters on the clustering outcome and help them to make informed interactions with the result. By making manual adjustments, users should be able to improve clustering results and aid them in exploring the solution space of a specific clustering algorithm.

Chapter 9

Conclusion

With Lodestar, we have proposed a novel approach for understanding and identifying stellar groups in hard clustering scenarios. Our approach includes general guidelines for understanding the algorithm and model parameters and an intuitive navigation system to aid in exploring the extensive solution space of the SigMA algorithm pipeline [61]. By working closely with domain experts, we leveraged their expertise to design a visualization system focused on astronomers.

One of the primary benefits of our tool is that it enables the provision of a solution space that is easy to comprehend and analyze beforehand. This tree-based solution space summary provides a natural ordering and connections between individual solutions. Supported by an overview of solution stability across the entire solution space, we offer a visually assisted approach to find areas of interest, making the solution space accessible and interpretable before drilling into individual clustering results.

With the density level and alpha scale space trees, we provide a novel approach to reveal the effect of a single parameter change on the clustering solution while simultaneously exposing the overall solution space. The tree-based design allows for a simple representation that conceptually maps the cluster evolution to nodes and edges of a tree, showing which clusters will merge with the parameter change.

By providing domain-specific validation views, such as a 3D spatial, 2D velocity, HRD, and a configurable feature scatter plot. In addition to the 2D velocity scatter plot, our solution also has histograms for each of the velocity features. That also includes the radial velocity. In total that amounts to three velocity histograms. These views support users in assessing the goodness of a clustering solution and building confidence in the outcome.

We implemented our tool to have a generic and reusable back-end, reusable for other front-end projects. We created the front-end application like a central store that communicates updates to multiple independent views. We integrated both into a docker setup, making it

faster to set up and simplifying for us to introduce change. This allowed us to react quickly to requests after the interviews and to tackle these requests in iterations. Additionally, our solution can handle large data sets with near-immediate response times, using extensive pre-calculation and caching techniques.

In individual interview sessions, we have tasked eight domain experts to perform five use cases. We observed that without much intervention, they efficiently traversed the solution space and validated clusters. We complemented each interview with a SUS questionnaire and feedback form to collect scores and improvement suggestions. With a final score of 76.79 and very positive feedback, we see great potential in helping astronomers detect novel clusters.

In summary, this work has contributed to the field of Astronomy and Data Visualization by providing a new approach that facilitates identifying stellar groups in hard clustering scenarios. The proposed method is effective and efficient and has the potential to be applied to various real-world scenarios. We hope that our research will inspire further studies and improvements in the field.

Bibliography

[1] A. Castro-Ginard, C. Jordi, X. Luri, F. Julbe, M. Morvan, L. Balaguer-Núñez, and T. Cantat-Gaudin. "A new method for unveiling open clusters in Gaia - New nearby open clusters confirmed by DR2." In: *Astronomy & Astrophysics* 618 (2018), A59. DOI: 10.1051/0004-6361/201833390.

[2] P. Avgeriou and U. Zdun. "Architectural Patterns Revisited - A Pattern Language." In: *Proceedings of 10th European Conference on Pattern Languages of Programs (EuroPlop 2005)*. 2005, pp. 1–39.

[3] A. Bangor, P. Kortum, and J. Miller. "Determining what individual SUS scores mean: Adding an adjective rating scale." In: *Journal of Usability Studies* 4.3 (2009), pp. 114–123.

[4] C. Beaumont, A. Goodman, and P. Greenfield. "Hackable User Interfaces In Astronomy with Glue." In: *Astronomical Data Analysis Software an Systems XXIV (ADASS XXIV)*. Ed. by A. R. Taylor and E. Rosolowsky. Vol. 495. Astronomical Society of the Pacific Conference Series. Sept. 2015, p. 101.

[5] F. Beck, M. Burch, S. Diehl, and D. Weiskopf. "A Taxonomy and Survey of Dynamic Graph Visualization." In: *Computer Graphics Forum* 36.1 (2017), pp. 133–159. DOI: https://doi.org/10.1111/cgf.12791.

[6] S. Bergner, M. Sedlmair, T. Möller, S. Abdolyousefi, and A. Saad. "Paraglide: Interactive parameter space partitioning for computer simulations." In: *IEEE Transactions on Visualization and Computer Graphics* 19.9 (2013), pp. 1499–1512.

[7] P. Berkhin. "A survey of clustering data mining techniques." In: *Grouping multidimensional data: Recent advances in clustering* (2006), pp. 25–71.

[8] M. Booshehrian, T. Möller, R. Peterman, and T. Munzner. "Vismon: Facilitating analysis of trade-offs, uncertainty, and sensitivity in fisheries management decision making." In: *Computer Graphics Forum*. Vol. 31. 3pt3. Wiley Online Library. 2012, pp. 1235–1244.

[9] E. Brochu, T. Brochu, and N. D. Freitas. "A Bayesian interactive optimization approach to procedural animation design." In: *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 2010, pp. 103–112.

[10] J. Brooke. "SUS: A Quick and Dirty Usability Scale." In: *Usability Evaluation in Industry* (1996).

[11] J. Brooke. "SUS: a retrospective." In: *Journal of Usability Studies* 8.2 (2013), pp. 29–40.

[12] S. Bruckner and T. Möller. "Result-Driven Exploration of Simulation Parameter Spaces for Visual Effects Design." In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1468–1476. DOI: 10.1109/TVCG.2010.190.

[13] P. Bruneau, P. Pinheiro, B. Broeksema, and B. Otjacques. "Cluster Sculptor, an interactive visual clustering system." In: *Neurocomputing* 150 (2015). Special Issue on Information Processing and Machine Learning for Applications of Engineering Solving Complex Machine Learning Problems with Ensemble Methods Visual Analytics using Multidimensional Projections, pp. 627–644. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2014.09.062.

[14] C. Boquan, E. D´Onghia, J. Alves, and A. Adamo. "Discovery of new stellar groups in the Orion complex - Towards a robust unsupervised approach." In: *Astronomy & Astrophysics* 643 (2020), A114. DOI: 10.1051/0004-6361/201935955.

[15] T. Caliński and J. Harabasz. "A dendrite method for cluster analysis." In: *Communications in Statistics* 3.1 (1974), pp. 1–27. DOI: 10.1080/03610927408827101.

[16] N. Cao, D. Gotz, J. Sun, and H. Qu. "DICON: Interactive Visual Analysis of Multidimensional Clusters." In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 2581–2590. DOI: 10.1109/TVCG.2011.188.

[17] R. Caruana, M. Elhawary, N. Nguyen, and C. Smith. "Meta Clustering." In: *Sixth International Conference on Data Mining (ICDM'06)*. 2006, pp. 107–118. DOI: 10.1109/ICDM.2006.103.

[18] M. Cavallo and Ç. Demiralp. "Clustrophile 2: Guided Visual Clustering Analysis." In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), pp. 267–276. DOI: 10.1109/TVCG.2018.2864477.

[19] P. Chandler and J. Sweller. "Cognitive Load Theory and the Format of Instruction." In: *Cognition and Instruction* 8.4 (1991), pp. 293–332. DOI: 10.1207/s1532690xci0804_2.

[20] K. Chen and L. Liu. "A visual framework invites human into the clustering process." In: *15th International Conference on Scientific and Statistical Database Management, 2003.* 2003, pp. 97–106. DOI: 10.1109/SSDM.2003.1214971.

[21] G. Collaboration, A. G. A. Brown, A. Vallenari, T. Prusti, J. H. J. de Bruijne, F. Mignard, R. Drimmel, C. Babusiaux, C. A. L. Bailer-Jones, U. Bastian, M. Biermann, D. W. Evans, L. Eyer, F. Jansen, C. Jordi, D. Katz, S. A. Klioner, U. Lammers, L. Lindegren, X. Luri, W. O'Mullane, C. Panem, D. Pourbaix, S. Randich, P. Sartoretti, H. I. Siddiqui, C. Soubiran, V. Valette, F. van Leeuwen, N. A. Walton, C. Aerts, F. Arenou, M. Cropper, E. Høg, M. G. Lattanzi, E. K. Grebel, A. D. Holland, C. Huc, X. Passot, M. Perryman, L. Bramante, C. Cacciari, J. Castañeda, L. Chaoul, N. Cheek, F. De Angeli, C. Fabricius, R. Guerra, J. Hernández, A. Jean-Antoine-Piccolo, E. Masana, R. Messineo, N. Mowlavi, K. Nienartowicz, D. Ordóñez-Blanco, P. Panuzzo, J. Portell, P. J. Richards, M. Riello, G. M. Seabroke, P. Tanga, F. Thévenin, J. Torra, S. G. Els, G. Gracia-Abril, G. Comoretto, M. Garcia-Reinaldos, T. Lock, E. Mercier, M. Altmann, R. Andrae, T. L. Astraatmadja, I. Bellas-Velidis, K. Benson, J. Berthier, R. Blomme, G. Busso, B. Carry, A. Cellino, G. Clementini, S. Cowell, O. Creevey, J. Cuypers, M. Davidson, J. De Ridder, A. de Torres, L. Delchambre, A. Dell'Oro, C. Ducourant, Y. Frémat, M. Garcıa-Torres, E. Gosset, J.-L. Halbwachs, N. C. Hambly, D. L. Harrison, M. Hauser, D. Hestroffer, S. T. Hodgkin, H. E. Huckle, A. Hutton, G. Jasniewicz, S. Jordan, M. Kontizas, A. J. Korn, A. C. Lanzafame, M. Manteiga, A. Moitinho, K. Muinonen, J. Osinde, E. Pancino, T. Pauwels, J.-M. Petit, A. Recio-Blanco, A. C. Robin, L. M. Sarro, C. Siopis, M. Smith, K. W. Smith, A. Sozzetti, W. Thuillot, W. van Reeven, Y. Viala, U. Abbas, A. Abreu Aramburu, S. Accart, J. J. Aguado, P. M. Allan, W. Allasia, G. Altavilla, M. A. Álvarez, J. Alves, R. I. Anderson, A. H. Andrei, E. Anglada Varela, E. Antiche, T. Antoja, S. Antón, B. Arcay, N. Bach, S. G. Baker, L. Balaguer-Núñez, C. Barache, C. Barata, A. Barbier, F. Barblan, D. Barrado y Navascués, M. Barros, M. A. Barstow, U. Becciani, M. Bellazzini, A. Bello Garcıa, V. Belokurov, P. Bendjoya, A. Berihuete, L. Bianchi, O. Bienaymé, F. Billebaud, N. Blagorodnova, S. Blanco-Cuaresma, T. Boch, A. Bombrun, R. Borrachero, S. Bouquillon, G. Bourda, H. Bouy, A. Bragaglia, M. A. Breddels, N. Brouillet, T. Brüsemeister, B. Bucciarelli, P. Burgess, R. Burgon, A. Burlacu, D. Busonero, R. Buzzi, E. Caffau, J. Cambras, H. Campbell, R. Cancelliere, T. Cantat-Gaudin, T. Carlucci, J. M. Carrasco, M. Castellani, P. Charlot, J. Charnas, A. Chiavassa, M. Clotet, G. Cocozza, R. S. Collins, G. Costigan, F. Crifo, N. J. G. Cross, M. Crosta, C. Crowley, C. Dafonte, Y. Damerdji, A. Dapergolas, P. David, M. David, P. De Cat, F. de Felice, P. de Laverny, F. De Luise, R.

De March, D. de Martino, R. de Souza, J. Debosscher, E. del Pozo, M. Delbo, A. Delgado, H. E. Delgado, P. Di Matteo, S. Diakite, E. Distefano, C. Dolding, S. Dos Anjos, P. Drazinos, J. Duran, Y. Dzigan, B. Edvardsson, H. Enke, N. W. Evans, G. Eynard Bontemps, C. Fabre, M. Fabrizio, S. Faigler, A. J. Falcão, M. Farràs Casas, L. Federici, G. Fedorets, J. Fernández-Hernández, P. Fernique, A. Fienga, F. Figueras, F. Filippi, K. Findeisen, A. Fonti, M. Fouesneau, E. Fraile, M. Fraser, J. Fuchs, M. Gai, S. Galleti, L. Galluccio, D. Garabato, F. Garcıa-Sedano, A. Garofalo, N. Garralda, P. Gavras, J. Gerssen, R. Geyer, G. Gilmore, S. Girona, G. Giuffrida, M. Gomes, A. González-Marcos, J. González-Núñez, J. J. González-Vidal, M. Granvik, A. Guerrier, P. Guillout, J. Guiraud, A. Gúrpide, R. Gutiérrez-Sánchez, L. P. Guy, R. Haigron, D. Hatzidimitriou, M. Haywood, U. Heiter, A. Helmi, D. Hobbs, W. Hofmann, B. Holl, G. Holland, J. A. S. Hunt, A. Hypki, V. Icardi, M. Irwin, G. Jevardat de Fombelle, P. Jofré, P. G. Jonker, A. Jorissen, F. Julbe, A. Karampelas, A. Kochoska, R. Kohley, K. Kolenberg, E. Kontizas, S. E. Koposov, G. Kordopatis, P. Koubsky, A. Krone-Martins, M. Kudryashova, I. Kull, R. K. Bachchan, F. Lacoste-Seris, A. F. Lanza, J.-B. Lavigne, C. Le Poncin-Lafitte, Y. Lebreton, T. Lebzelter, S. Leccia, N. Leclerc, I. Lecoeur-Taibi, V. Lemaitre, H. Lenhardt, F. Leroux, S. Liao, E. Licata, H. E. P. Lindstrøm, T. A. Lister, E. Livanou, A. Lobel, W. Löffler, M. López, D. Lorenz, I. MacDonald, T. Magalhães Fernandes, S. Managau, R. G. Mann, G. Mantelet, O. Marchal, J. M. Marchant, M. Marconi, S. Marinoni, P. M. Marrese, G. Marschalkó, D. J. Marshall, J. M. Martın-Fleitas, M. Martino, N. Mary, G. Matijevič, T. Mazeh, P. J. McMillan, S. Messina, D. Michalik, N. R. Millar, B. M. H. Miranda, D. Molina, R. Molinaro, M. Molinaro, L. Molnár, M. Moniez, P. Montegriffo, R. Mor, A. Mora, R. Morbidelli, T. Morel, S. Morgenthaler, D. Morris, A. F. Mulone, T. Muraveva, I. Musella, J. Narbonne, G. Nelemans, L. Nicastro, L. Noval, C. Ordénovic, J. Ordieres-Meré, P. Osborne, C. Pagani, I. Pagano, F. Pailler, H. Palacin, L. Palaversa, P. Parsons, M. Pecoraro, R. Pedrosa, H. Pentikäinen, B. Pichon, A. M. Piersimoni, F.-X. Pineau, E. Plachy, G. Plum, E. Poujoulet, A. Prša, L. Pulone, S. Ragaini, S. Rago, N. Rambaux, M. Ramos-Lerate, P. Ranalli, G. Rauw, A. Read, S. Regibo, C. Reylé, R. A. Ribeiro, L. Rimoldini, V. Ripepi, A. Riva, G. Rixon, M. Roelens, M. Romero-Gómez, N. Rowell, F. Royer, L. Ruiz-Dern, G. Sadowski, T. Sagristà Sellés, J. Sahlmann, J. Salgado, E. Salguero, M. Sarasso, H. Savietto, M. Schultheis, E. Sciacca, M. Segol, J. C. Segovia, D. Segransan, I.-C. Shih, R. Smareglia, R. L. Smart, E. Solano, F. Solitro, R. Sordo, S. Soria Nieto, J. Souchay, A. Spagna, F. Spoto, U. Stampa, I. A. Steele, H. Steidelmüller, C. A. Stephenson, H. Stoev, F. F. Suess, M. Süveges, J. Surdej, L. Szabados, E. Szegedi-Elek, D. Tapiador, F. Taris, G. Tauran,

M. B. Taylor, R. Teixeira, D. Terrett, B. Tingley, S. C. Trager, C. Turon, A. Ulla, E. Utrilla, G. Valentini, A. van Elteren, E. Van Hemelryck, M. van Leeuwen, M. Varadi, A. Vecchiato, J. Veljanoski, T. Via, D. Vicente, S. Vogt, H. Voss, V. Votruba, S. Voutsinas, G. Walmsley, M. Weiler, K. Weingrill, T. Wevers, Ł. Wyrzykowski, A. Yoldas, M. Žerjal, S. Zucker, C. Zurbach, T. Zwitter, A. Alecu, M. Allen, C. Allende Prieto, A. Amorim, G. Anglada-Escudé, V. Arsenijevic, S. Azaz, P. Balm, M. Beck, H.-H. Bernstein, L. Bigot, A. Bijaoui, C. Blasco, M. Bonfigli, G. Bono, S. Boudreault, A. Bressan, S. Brown, P.-M. Brunet, P. Bunclark, R. Buonanno, A. G. Butkevich, C. Carret, C. Carrion, L. Chemin, F. Chéreau, L. Corcione, E. Darmigny, K. S. de Boer, P. de Teodoro, P. T. de Zeeuw, C. Delle Luche, C. D. Domingues, P. Dubath, F. Fodor, B. Frézouls, A. Fries, D. Fustes, D. Fyfe, E. Gallardo, J. Gallegos, D. Gardiol, M. Gebran, A. Gomboc, A. Gómez, E. Grux, A. Gueguen, A. Heyrovsky, J. Hoar, G. Iannicola, Y. Isasi Parache, A.-M. Janotto, E. Joliet, A. Jonckheere, R. Keil, D.-W. Kim, P. Klagyivik, J. Klar, J. Knude, O. Kochukhov, I. Kolka, J. Kos, A. Kutka, V. Lainey, D. LeBouquin, C. Liu, D. Loreggia, V. V. Makarov, M. G. Marseille, C. Martayan, O. Martinez-Rubi, B. Massart, F. Meynadier, S. Mignot, U. Munari, A.-T. Nguyen, T. Nordlander, P. Ocvirk, K. S. O'Flaherty, A. Olias Sanz, P. Ortiz, J. Osorio, D. Oszkiewicz, A. Ouzounis, M. Palmer, P. Park, E. Pasquato, C. Peltzer, J. Peralta, F. Péturaud, T. Pieniluoma, E. Pigozzi, J. Poels, G. Prat, T. Prod'homme, F. Raison, J. M. Rebordao, D. Risquez, B. Rocca-Volmerange, S. Rosen, M. I. Ruiz-Fuertes, F. Russo, S. Sembay, I. Serraller Vizcaino, A. Short, A. Siebert, H. Silva, D. Sinachopoulos, E. Slezak, M. Soffel, D. Sosnowska, V. Straižys, M. ter Linden, D. Terrell, S. Theil, C. Tiede, L. Troisi, P. Tsalmantza, D. Tur, M. Vaccari, F. Vachier, P. Valles, W. Van Hamme, L. Veltz, J. Virtanen, J.-M. Wallut, R. Wichmann, M. I. Wilkinson, H. Ziaeepour, and S. Zschocke. "Gaia Data Release 1. Summary of the astrometric, photometric, and survey properties." In: *Astronomy & Astrophysics* 595, A2 (Nov. 2016), A2. DOI: 10.1051/0004-6361/201629512.

[22] G. Collaboration, T. Prusti, J. H. J. de Bruijne, A. G. A. Brown, A. Vallenari, C. Babusiaux, C. A. L. Bailer-Jones, U. Bastian, M. Biermann, D. W. Evans, L. Eyer, F. Jansen, C. Jordi, S. A. Klioner, U. Lammers, L. Lindegren, X. Luri, F. Mignard, D. J. Milligan, C. Panem, V. Poinsignon, D. Pourbaix, S. Randich, G. Sarri, P. Sartoretti, H. I. Siddiqui, C. Soubiran, V. Valette, F. van Leeuwen, N. A. Walton, C. Aerts, F. Arenou, M. Cropper, R. Drimmel, E. Høg, D. Katz, M. G. Lattanzi, W. O´Mullane, E. K. Grebel, A. D. Holland, C. Huc, X. Passot, L. Bramante, C. Cacciari, J. Castañeda, L. Chaoul, N. Cheek, F. De Angeli, C. Fabricius, R. Guerra, J. Hernández, A. Jean-Antoine-Piccolo, E. Masana,

R. Messineo, N. Mowlavi, K. Nienartowicz, D. Ordóñez-Blanco, P. Panuzzo, J. Portell, P. J. Richards, M. Riello, G. M. Seabroke, P. Tanga, F. Thévenin, J. Torra, S. G. Els, G. Gracia-Abril, G. Comoretto, M. Garcia-Reinaldos, T. Lock, E. Mercier, M. Altmann, R. Andrae, T. L. Astraatmadja, I. Bellas-Velidis, K. Benson, J. Berthier, R. Blomme, G. Busso, B. Carry, A. Cellino, G. Clementini, S. Cowell, O. Creevey, J. Cuypers, M. Davidson, J. De Ridder, A. de Torres, L. Delchambre, A. Dell´Oro, C. Ducourant, Y. Frémat, M. Garcıa-Torres, E. Gosset, J.-L. Halbwachs, N. C. Hambly, D. L. Harrison, M. Hauser, D. Hestroffer, S. T. Hodgkin, H. E. Huckle, A. Hutton, G. Jasniewicz, S. Jordan, M. Kontizas, A. J. Korn, A. C. Lanzafame, M. Manteiga, A. Moitinho, K. Muinonen, J. Osinde, E. Pancino, T. Pauwels, J.-M. Petit, A. Recio-Blanco, A. C. Robin, L. M. Sarro, C. Siopis, M. Smith, K. W. Smith, A. Sozzetti, W. Thuillot, W. van Reeven, Y. Viala, U. Abbas, A. Abreu Aramburu, S. Accart, J. J. Aguado, P. M. Allan, W. Allasia, G. Altavilla, M. A. Álvarez, J. Alves, R. I. Anderson, A. H. Andrei, E. Anglada Varela, E. Antiche, T. Antoja, S. Antón, B. Arcay, A. Atzei, L. Ayache, N. Bach, S. G. Baker, L. Balaguer-Núñez, C. Barache, C. Barata, A. Barbier, F. Barblan, M. Baroni, D. Barrado y Navascués, M. Barros, M. A. Barstow, U. Becciani, M. Bellazzini, G. Bellei, A. Bello Garcıa, V. Belokurov, P. Bendjoya, A. Berihuete, L. Bianchi, O. Bienaymé, F. Billebaud, N. Blagorodnova, S. Blanco-Cuaresma, T. Boch, A. Bombrun, R. Borrachero, S. Bouquillon, G. Bourda, H. Bouy, A. Bragaglia, M. A. Breddels, N. Brouillet, T. Brüsemeister, B. Bucciarelli, F. Budnik, P. Burgess, R. Burgon, A. Burlacu, D. Busonero, R. Buzzi, E. Caffau, J. Cambras, H. Campbell, R. Cancelliere, T. Cantat-Gaudin, T. Carlucci, J. M. Carrasco, M. Castellani, P. Charlot, J. Charnas, P. Charvet, F. Chassat, A. Chiavassa, M. Clotet, G. Cocozza, R. S. Collins, P. Collins, G. Costigan, F. Crifo, N. J. G. Cross, M. Crosta, C. Crowley, C. Dafonte, Y. Damerdji, A. Dapergolas, P. David, M. David, P. De Cat, F. de Felice, P. de Laverny, F. De Luise, R. De March, D. de Martino, R. de Souza, J. Debosscher, E. del Pozo, M. Delbo, A. Delgado, H. E. Delgado, F. di Marco, P. Di Matteo, S. Diakite, E. Distefano, C. Dolding, S. Dos Anjos, P. Drazinos, J. Durán, Y. Dzigan, E. Ecale, B. Edvardsson, H. Enke, M. Erdmann, D. Escolar, M. Espina, N. W. Evans, G. Eynard Bontemps, C. Fabre, M. Fabrizio, S. Faigler, A. J. Falcão, M. Farràs Casas, F. Faye, L. Federici, G. Fedorets, J. Fernández-Hernández, P. Fernique, A. Fienga, F. Figueras, F. Filippi, K. Findeisen, A. Fonti, M. Fouesneau, E. Fraile, M. Fraser, J. Fuchs, R. Furnell, M. Gai, S. Galleti, L. Galluccio, D. Garabato, F. Garcıa-Sedano, P. Garé, A. Garofalo, N. Garralda, P. Gavras, J. Gerssen, R. Geyer, G. Gilmore, S. Girona, G. Giuffrida, M. Gomes, A. González-Marcos, J. González-Núñez, J. J. González-Vidal, M. Granvik, A. Guerrier,

P. Guillout, J. Guiraud, A. Gúrpide, R. Gutiérrez-Sánchez, L. P. Guy, R. Haigron, D. Hatzidimitriou, M. Haywood, U. Heiter, A. Helmi, D. Hobbs, W. Hofmann, B. Holl, G. Holland, J. A. S. Hunt, A. Hypki, V. Icardi, M. Irwin, G. Jevardat de Fombelle, P. Jofré, P. G. Jonker, A. Jorissen, F. Julbe, A. Karampelas, A. Kochoska, R. Kohley, K. Kolenberg, E. Kontizas, S. E. Koposov, G. Kordopatis, P. Koubsky, A. Kowalczyk, A. Krone-Martins, M. Kudryashova, I. Kull, R. K. Bachchan, F. Lacoste-Seris, A. F. Lanza, J.-B. Lavigne, C. Le Poncin-Lafitte, Y. Lebreton, T. Lebzelter, S. Leccia, N. Leclerc, I. Lecoeur-Taibi, V. Lemaitre, H. Lenhardt, F. Leroux, S. Liao, E. Licata, H. E. P. Lindstrøm, T. A. Lister, E. Livanou, A. Lobel, W. Löffler, M. López, A. Lopez-Lozano, D. Lorenz, T. Loureiro, I. MacDonald, T. Magalhães Fernandes, S. Managau, R. G. Mann, G. Mantelet, O. Marchal, J. M. Marchant, M. Marconi, J. Marie, S. Marinoni, P. M. Marrese, G. Marschalkó, D. J. Marshall, J. M. Martın-Fleitas, M. Martino, N. Mary, G. Matijevic, T. Mazeh, P. J. McMillan, S. Messina, A. Mestre, D. Michalik, N. R. Millar, B. M. H. Miranda, D. Molina, R. Molinaro, M. Molinaro, L. Molnár, M. Moniez, P. Montegriffo, D. Monteiro, R. Mor, A. Mora, R. Morbidelli, T. Morel, S. Morgenthaler, T. Morley, D. Morris, A. F. Mulone, T. Muraveva, I. Musella, J. Narbonne, G. Nelemans, L. Nicastro, L. Noval, C. Ordénovic, J. Ordieres-Meré, P. Osborne, C. Pagani, I. Pagano, F. Pailler, H. Palacin, L. Palaversa, P. Parsons, T. Paulsen, M. Pecoraro, R. Pedrosa, H. Pentikäinen, J. Pereira, B. Pichon, A. M. Piersimoni, F.-X. Pineau, E. Plachy, G. Plum, E. Poujoulet, A. Prsa, L. Pulone, S. Ragaini, S. Rago, N. Rambaux, M. Ramos-Lerate, P. Ranalli, G. Rauw, A. Read, S. Regibo, F. Renk, C. Reylé, R. A. Ribeiro, L. Rimoldini, V. Ripepi, A. Riva, G. Rixon, M. Roelens, M. Romero-Gómez, N. Rowell, F. Royer, A. Rudolph, L. Ruiz-Dern, G. Sadowski, T. Sagristà Sellés, J. Sahlmann, J. Salgado, E. Salguero, M. Sarasso, H. Savietto, A. Schnorhk, M. Schultheis, E. Sciacca, M. Segol, J. C. Segovia, D. Segransan, E. Serpell, I.-C. Shih, R. Smareglia, R. L. Smart, C. Smith, E. Solano, F. Solitro, R. Sordo, S. Soria Nieto, J. Souchay, A. Spagna, F. Spoto, U. Stampa, I. A. Steele, H. Steidelmüller, C. A. Stephenson, H. Stoev, F. F. Suess, M. Süveges, J. Surdej, L. Szabados, E. Szegedi-Elek, D. Tapiador, F. Taris, G. Tauran, M. B. Taylor, R. Teixeira, D. Terrett, B. Tingley, S. C. Trager, C. Turon, A. Ulla, E. Utrilla, G. Valentini, A. van Elteren, E. Van Hemelryck, M. van Leeuwen, M. Varadi, A. Vecchiato, J. Veljanoski, T. Via, D. Vicente, S. Vogt, H. Voss, V. Votruba, S. Voutsinas, G. Walmsley, M. Weiler, K. Weingrill, D. Werner, T. Wevers, G. Whitehead, L. Wyrzykowski, A. Yoldas, M. Zerjal, S. Zucker, C. Zurbach, T. Zwitter, A. Alecu, M. Allen, C. Allende Prieto, A. Amorim, G. Anglada-Escudé, V. Arsenijevic, S. Azaz, P. Balm, M. Beck, H.-H. Bernstein, L. Bigot, A. Bijaoui, C. Blasco,

M. Bonfigli, G. Bono, S. Boudreault, A. Bressan, S. Brown, P.-M. Brunet, P. Bunclark, R. Buonanno, A. G. Butkevich, C. Carret, C. Carrion, L. Chemin, F. Chéreau, L. Corcione, E. Darmigny, K. S. de Boer, P. de Teodoro, P. T. de Zeeuw, C. Delle Luche, C. D. Domingues, P. Dubath, F. Fodor, B. Frézouls, A. Fries, D. Fustes, D. Fyfe, E. Gallardo, J. Gallegos, D. Gardiol, M. Gebran, A. Gomboc, A. Gómez, E. Grux, A. Gueguen, A. Heyrovsky, J. Hoar, G. Iannicola, Y. Isasi Parache, A.-M. Janotto, E. Joliet, A. Jonckheere, R. Keil, D.-W. Kim, P. Klagyivik, J. Klar, J. Knude, O. Kochukhov, I. Kolka, J. Kos, A. Kutka, V. Lainey, D. LeBouquin, C. Liu, D. Loreggia, V. V. Makarov, M. G. Marseille, C. Martayan, O. Martinez-Rubi, B. Massart, F. Meynadier, S. Mignot, U. Munari, A.-T. Nguyen, T. Nordlander, P. Ocvirk, K. S. O´Flaherty, A. Olias Sanz, P. Ortiz, J. Osorio, D. Oszkiewicz, A. Ouzounis, M. Palmer, P. Park, E. Pasquato, C. Peltzer, J. Peralta, F. Péturaud, T. Pieniluoma, E. Pigozzi, J. Poels, G. Prat, T. Prod´homme, F. Raison, J. M. Rebordao, D. Risquez, B. Rocca-Volmerange, S. Rosen, M. I. Ruiz-Fuertes, F. Russo, S. Sembay, I. Serraller Vizcaino, A. Short, A. Siebert, H. Silva, D. Sinachopoulos, E. Slezak, M. Soffel, D. Sosnowska, V. Straizys, M. ter Linden, D. Terrell, S. Theil, C. Tiede, L. Troisi, P. Tsalmantza, D. Tur, M. Vaccari, F. Vachier, P. Valles, W. Van Hamme, L. Veltz, J. Virtanen, J.-M. Wallut, R. Wichmann, M. I. Wilkinson, H. Ziaeepour, and S. Zschocke. "The Gaia mission." In: *Astronomy & Astrophysics* 595 (2016). DOI: 10.1051/0004-6361/201629272.

[23] D. Comaniciu and P. Meer. "Mean shift: A robust approach toward feature space analysis." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.5 (2002), pp. 603–619.

[24] R. Communications. *Astro User Experience Design System*. 2021. (Visited on 02/19/2021).

[25] D. Davies and D. Bouldin. "A Cluster Separation Measure." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1.2 (1979), pp. 224–227. DOI: 10.1109/TPAMI.1979.4766909.

[26] Ç. Demiralp. *Clustrophile: A Tool for Visual Clustering Analysis*. 2017. DOI: 10.48550/ARXIV.1710.02173.

[27] E. Zari, A.G.A. Brown, and P.T. de Zeeuw. "Structure, kinematics, and ages of the young stellar populations in the Orion region." In: *Astronomy & Astrophysics* 628 (2019), A123. DOI: 10.1051/0004-6361/201935781.

[28] M. Ester, H. Kriegel, J. Sander, and X. Xu. "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise." In: *Proceedings of the Second Interna-*

tional Conference on Knowledge Discovery and Data Mining. KDD'96. Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[29] F. Damiani, L. Prisinzano, I. Pillitteri, G. Micela, and S. Sciortino. "Stellar population of Sco OB2 revealed by Gaia DR2 data." In: *Astronomy & Astrophysics* 623 (2019), A112. DOI: 10.1051/0004-6361/201833994.

[30] Gaia Collaboration, A. G. A. Brown, A. Vallenari, T. Prusti, J. H. J. de Bruijne, C. Babusiaux, M. Biermann, O. L. Creevey, D. W. Evans, L. Eyer, A. Hutton, F. Jansen, C. Jordi, S. A. Klioner, U. Lammers, L. Lindegren, X. Luri, F. Mignard, C. Panem, D. Pourbaix, S. Randich, P. Sartoretti, C. Soubiran, N. A. Walton, F. Arenou, C. A. L. Bailer-Jones, U. Bastian, M. Cropper, R. Drimmel, D. Katz, M. G. Lattanzi, F. van Leeuwen, J. Bakker, C. Cacciari, J. Castañeda, F. De Angeli, C. Ducourant, C. Fabricius, M. Fouesneau, Y. Frémat, R. Guerra, A. Guerrier, J. Guiraud, A. Jean-Antoine Piccolo, E. Masana, R. Messineo, N. Mowlavi, C. Nicolas, K. Nienartowicz, F. Pailler, P. Panuzzo, F. Riclet, W. Roux, G. M. Seabroke, R. Sordo, P. Tanga, F. Thévenin, G. Gracia-Abril, J. Portell, D. Teyssier, M. Altmann, R. Andrae, I. Bellas-Velidis, K. Benson, J. Berthier, R. Blomme, E. Brugaletta, P. W. Burgess, G. Busso, B. Carry, A. Cellino, N. Cheek, G. Clementini, Y. Damerdji, M. Davidson, L. Delchambre, A. Dell´Oro, J. Fernández-Hernández, L. Galluccio, P. Garcıa-Lario, M. Garcia-Reinaldos, J. González-Núñez, E. Gosset, R. Haigron, J.-L. Halbwachs, N. C. Hambly, D. L. Harrison, D. Hatzidimitriou, U. Heiter, J. Hernández, D. Hestroffer, S. T. Hodgkin, B. Holl, K. Janßen, G. Jevardat de Fombelle, S. Jordan, A. Krone-Martins, A. C. Lanzafame, W. Löffler, A. Lorca, M. Manteiga, O. Marchal, P.M. Marrese, A. Moitinho, A. Mora, K. Muinonen, P. Osborne, E. Pancino, T. Pauwels, J.-M. Petit, A. Recio-Blanco, P. J. Richards, M. Riello, L. Rimoldini, A. C. Robin, T. Roegiers, J. Rybizki, L. M. Sarro, C. Siopis, M. Smith, A. Sozzetti, A. Ulla, E. Utrilla, M. van Leeuwen, W. van Reeven, U. Abbas, A. Abreu Aramburu, S. Accart, C. Aerts, J. J. Aguado, M. Ajaj, G. Altavilla, M. A. Álvarez, J. Álvarez Cid-Fuentes, J. Alves, R. I. Anderson, E. Anglada Varela, T. Antoja, M. Audard, D. Baines, S. G. Baker, L. Balaguer-Núñez, E. Balbinot, Z. Balog, C. Barache, D. Barbato, M. Barros, M. A. Barstow, S. Bartolomé, J.-L. Bassilana, N. Bauchet, A. Baudesson-Stella, U. Becciani, M. Bellazzini, M. Bernet, S. Bertone, L. Bianchi, S. Blanco-Cuaresma, T. Boch, A. Bombrun, D. Bossini, S. Bouquillon, A. Bragaglia, L. Bramante, E. Breedt, A. Bressan, N. Brouillet, B. Bucciarelli, A. Burlacu, D. Busonero, A. G. Butkevich, R. Buzzi, E. Caffau, R. Cancelliere, H. Cánovas, T. Cantat-Gaudin, R. Carballo, T. Carlucci, M. I. Carnerero, J. M. Carrasco, L. Casamiquela, M. Castellani, A. Castro-Ginard, P. Castro Sampol, L. Chaoul, P. Charlot, L. Chemin, A.

Chiavassa, M.-R. L. Cioni, G. Comoretto, W. J. Cooper, T. Cornez, S. Cowell, F. Crifo, M. Crosta, C. Crowley, C. Dafonte, A. Dapergolas, M. David, P. David, P. de Laverny, F. De Luise, R. De March, J. De Ridder, R. de Souza, P. de Teodoro, A. de Torres, E. F. del Peloso, E. del Pozo, M. Delbo, A. Delgado, H. E. Delgado, J.-B. Delisle, P. Di Matteo, S. Diakite, C. Diener, E. Distefano, C. Dolding, D. Eappachen, B. Edvardsson, H. Enke, P. Esquej, C. Fabre, M. Fabrizio, S. Faigler, G. Fedorets, P. Fernique, A. Fienga, F. Figueras, C. Fouron, F. Fragkoudi, E. Fraile, F. Franke, M. Gai, D. Garabato, A. Garcia-Gutierrez, M. Garcıa-Torres, A. Garofalo, P. Gavras, E. Gerlach, R. Geyer, P. Giacobbe, G. Gilmore, S. Girona, G. Giuffrida, R. Gomel, A. Gomez, I. Gonzalez-Santamaria, J. J. González-Vidal, M. Granvik, R. Gutiérrez-Sánchez, L. P. Guy, M. Hauser, M. Haywood, A. Helmi, S. L. Hidalgo, T. Hilger, N. Hladczuk, D. Hobbs, G. Holland, H. E. Huckle, G. Jasniewicz, P. G. Jonker, J. Juaristi Campillo, F. Julbe, L. Karbevska, P. Kervella, S. Khanna, A. Kochoska, M. Kontizas, G. Kordopatis, A. J. Korn, Z. Kostrzewa-Rutkowska, K. Kruszy´nska, S. Lambert, A. F. Lanza, Y. Lasne, J.-F. Le Campion, Y. Le Fustec, Y. Lebreton, T. Lebzelter, S. Leccia, N. Leclerc, I. Lecoeur-Taibi, S. Liao, E. Licata, E. P. Lindstrøm, T. A. Lister, E. Livanou, A. Lobel, P. Madrero Pardo, S. Managau, R. G. Mann, J. M. Marchant, M. Marconi, M. M. S. Marcos Santos, S. Marinoni, F. Marocco, D. J. Marshall, L. Martin Polo, J. M. Martın-Fleitas, A. Masip, D. Massari, A. Mastrobuono-Battisti, T. Mazeh, P. J. McMillan, S. Messina, D. Michalik, N. R. Millar, A. Mints, D. Molina, R. Molinaro, L. Molnár, P. Montegriffo, R. Mor, R. Morbidelli, T. Morel, D. Morris, A. F. Mulone, D. Munoz, T. Muraveva, C. P. Murphy, I. Musella, L. Noval, C. Ordénovic, G. Orrù, J. Osinde, C. Pagani, I. Pagano, L. Palaversa, P. A. Palicio, A. Panahi, M. Pawlak, X. Peñalosa Esteller, A. Penttilä, A. M. Piersimoni, F.-X. Pineau, E. Plachy, G. Plum, E. Poggio, E. Poretti, E. Poujoulet, A. Prsa, L. Pulone, E. Racero, S. Ragaini, M. Rainer, C. M. Raiteri, N. Rambaux, P. Ramos, M. Ramos-Lerate, P. Re Fiorentin, S. Regibo, C. Reylé, V. Ripepi, A. Riva, G. Rixon, N. Robichon, C. Robin, M. Roelens, L. Rohrbasser, M. Romero-Gómez, N. Rowell, F. Royer, K. A. Rybicki, G. Sadowski, A. Sagristà Sellés, J. Sahlmann, J. Salgado, E. Salguero, N. Samaras, V. Sanchez Gimenez, N. Sanna, R. Santoveña, M. Sarasso, M. Schultheis, E. Sciacca, M. Segol, J. C. Segovia, D. Ségransan, D. Semeux, S. Shahaf, H. I. Siddiqui, A. Siebert, L. Siltala, E. Slezak, R. L. Smart, E. Solano, F. Solitro, D. Souami, J. Souchay, A. Spagna, F. Spoto, I. A. Steele, H. Steidelmüller, C. A. Stephenson, M. Süveges, L. Szabados, E. Szegedi-Elek, F. Taris, G. Tauran, M. B. Taylor, R. Teixeira, W. Thuillot, N. Tonello, F. Torra, J. Torra, C. Turon, N. Unger, M. Vaillant, E. van Dillen, O. Vanel, A. Vecchiato, Y. Viala, D. Vicente, S.

Voutsinas, M. Weiler, T. Wevers, L. Wyrzykowski, A. Yoldas, P. Yvard, H. Zhao, J. Zorec, S. Zucker, C. Zurbach, and T. Zwitter. "Gaia early data release 3-summary of the contents and survey properties." In: *Astronomy & Astrophysics* 649 (2021).

[31] M. Galesic and M. Bosnjak. "Effects of questionnaire length on participation and indicators of response quality in a web survey." In: *Public opinion quarterly* 73.2 (2009), pp. 349–360.

[32] P. Galli, H. Bouy, J. Olivares, N. Miret-Roig, L. Sarro, D. Barrado, A. Berihuete, E. Bertin, and J. Cuillandre. "Chamaeleon DANCe-Revisiting the stellar populations of Chamaeleon I and Chamaeleon II with Gaia-DR2 data." In: *Astronomy & Astrophysics* 646 (2021), A46.

[33] S. Gerber, P. Bremer, V. Pascucci, and R. Whitaker. "Visual Exploration of High Dimensional Scalar Functions." In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1271–1280. DOI: 10.1109/TVCG.2010.213.

[34] M. Gittens, Y. Kim, and D. Godwin. "The vital few versus the trivial many: examining the Pareto principle for software." In: *29th Annual International Computer Software and Applications Conference (COMPSAC'05)*. Vol. 1. 2005, 179–185 Vol. 2. DOI: 10.1109/COMPSAC.2005.153.

[35] M. Glinz. "On Non-Functional Requirements." In: *15th IEEE International Requirements Engineering Conference (RE 2007)*. 2007, pp. 21–26. DOI: 10.1109/RE.2007.45.

[36] M. Goeminne and T. Mens. "Evidence for the Pareto principle in open source software activity." In: *the Joint Porceedings of the 1st International workshop on Model Driven Software Maintenance and 5th International Workshop on Software Quality and Maintainability*. Citeseer. 2011, pp. 74–82.

[37] J. Gortler, C. Schulz, D. Weiskopf, and O. Deussen. "Bubble Treemaps for Uncertainty Visualization." In: *IEEE Transactions on Visualization and Computer Graphics* 24.01 (Jan. 2018), pp. 719–728. ISSN: 1941-0506. DOI: 10.1109/TVCG.2017.2743959.

[38] Z. Gu, R. Eils, and M. Schlesner. "Complex heatmaps reveal patterns and correlations in multidimensional genomic data." In: *Bioinformatics* 32.18 (May 2016), pp. 2847–2849. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw313.

[39] M. Halkidi and M. Vazirgiannis. "Clustering validity assessment: finding the optimal partitioning of a data set." In: *Proceedings 2001 IEEE International Conference on Data Mining*. 2001, pp. 187–194. DOI: 10.1109/ICDM.2001.989517.

[40] K. Healy. *Data Visualization: A Practical Introduction*. Princeton University Press, 2018.

[41] E. Hunt and S. Reffert. "Improving the open cluster census." In: *Astronomy & Astrophysics* 646 (Feb. 2021), A104. ISSN: 1432-0746. DOI: 10.1051/0004-6361/202039341.

[42] I. Kushniruk, T. Schirmer, and T. Bensby. "Kinematic structures of the solar neighbourhood revealed by Gaia DR1/TGAS and RAVE." In: *Astronomy & Astrophysics* 608 (2017), A73. DOI: 10.1051/0004-6361/201731147.

[43] Y. Ioannidis. "The history of histograms (abridged)." In: *Proceedings 2003 VLDB Conference*. Elsevier. 2003, pp. 19–30.

[44] J. Olivares, H. Bouy, L.M. Sarro, E. Moraux, A. Berihuete, P.A.B. Galli, and N. Miret-Roig. "Miec: A Bayesian hierarchical model for the analysis of nearby young open clusters." In: *Astronomy & Astrophysics* 649 (2021), A159. DOI: 10.1051/0004-6361/202140282.

[45] H. Kamdar, C. Conroy, Y. Ting, A. Bonaca, M. Smith, and A. Brown. "Stars that Move Together Were Born Together." In: *The Astrophysical Journal Letters* 884.2 (Oct. 2019), p. L42. DOI: 10.3847/2041-8213/ab4997.

[46] M. Kounkel and K. Covey. "Untangling the Galaxy. I. Local Structure and Star Formation History of the Milky Way." In: *The Astronomical Journal* 158.3 (Aug. 2019), p. 122. DOI: 10.3847/1538-3881/ab339a.

[47] K. Kuhn and T. Koupelis. *In quest of the universe*. Jones & Bartlett Learning, 2007, p. 369.

[48] B. Kwon, B. Eysenbach, J. Verma, K. Ng, C. D. Filippi, W. Stewart, and A. Perer. "Clustervision: Visual Supervision of Unsupervised Clustering." In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), pp. 142–151. DOI: 10.1109/TVCG.2017.2745085.

[49] L. Lindegren, J. Hernández, A. Bombrun, S. Klioner, U. Bastian, M. Ramos-Lerate, A. de Torres, H. Steidelmüller, C. Stephenson, D. Hobbs, U. Lammers, M. Biermann, R. Geyer, T. Hilger, D. Michalik, U. Stampa, P.J. McMillan, J. Castañeda, M. Clotet, G. Comoretto, M. Davidson, C. Fabricius, G. Gracia, N.C. Hambly, A. Hutton, A. Mora, J. Portell, F. van Leeuwen, U. Abbas, A. Abreu, M. Altmann, A. Andrei, E. Anglada, L. Balaguer-Núñez, C. Barache, U. Becciani, S. Bertone, L. Bianchi, S. Bouquillon, G. Bourda, T. Brüsemeister, B. Bucciarelli, D. Busonero, R. Buzzi, R. Cancelliere, T. Carlucci, P. Charlot, N. Cheek, M. Crosta, C. Crowley, J. de Bruijne, F. de Felice, R. Drimmel, P. Esquej, A. Fienga, E. Fraile, M. Gai, N. Garralda, J.J. González-Vidal, R. Guerra, M. Hauser, W. Hofmann, B. Holl, S. Jordan, M.G. Lattanzi, H. Lenhardt, S. Liao, E. Licata, T. Lister, W. Löffler, J. Marchant, J.-M. Martin-Fleitas, R. Messineo, F. Mignard, R. Morbidelli, E. Poggio, A.

Riva, N. Rowell, E. Salguero, M. Sarasso, E. Sciacca, H. Siddiqui, R.L. Smart, A. Spagna, I. Steele, F. Taris, J. Torra, A. van Elteren, W. van Reeven, and A. Vecchiato. "Gaia Data Release 2 - The astrometric solution." In: *Astronomy & Astrophysics* 616 (2018), A2. DOI: 10.1051/0004-6361/201832727.

[50] C. Lada and E. Lada. "Embedded clusters in molecular clouds." In: *Annual Review of Astronomy and Astrophysics* 41.1 (2003), pp. 57–115.

[51] A. Lancaster. "Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces." In: *IEEE Transactions on Professional Communication* 47.4 (2004), pp. 335–336. DOI: 10.1109/TPC.2004.837973.

[52] C. Larman and V. Basili. "Iterative and incremental developments. A brief history." In: *Computer* 36.6 (2003), pp. 47–56. DOI: 10.1109/MC.2003.1204375.

[53] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu. "Understanding of internal clustering validation measures." In: *2010 IEEE International Conference on Data Mining*. IEEE. 2010, pp. 911–916.

[54] L. V. der Maaten and G. Hinton. "Visualizing data using t-SNE." In: *Journal of Machine Learning Research* (2008).

[55] L. McInnes, J. Healy, and S. Astels. "hdbscan: Hierarchical density based clustering." In: *Journal of Open Source Software* 2.11 (2017), p. 205. DOI: 10.21105/joss.00205.

[56] T. Munzner. *Visualization Analysis and Design*. AK Peters Visualization Series. CRC Press, 2015. ISBN: 9781498759717.

[57] S. Oh, A. Price-Whelan, D. Hogg, T. Morton, and D. Spergel. "Comoving stars in Gaia DR1: an abundance of very wide separation comoving pairs." In: *The Astronomical Journal* 153.6 (2017), p. 257.

[58] A. Pretorius, M. Bray, A. Carpenter, and R. Ruddle. "Visualization of parameter space for image analysis." In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 2402–2411.

[59] R. Chen, L. Yang, S. Goodison, and Y. Sun. "Deep-learning approach to identifying cancer subtypes using high-dimensional genomic data." In: *Bioinformatics* 36.5 (Oct. 2019), pp. 1476–1483. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btz769.

[60] S. Ratzenbock, V. Obermuller, T. Moller, J. Alves, and I. Bomze. "Uncover: Toward Interpretable Models for Detecting New Star Cluster Members." In: *IEEE Transactions on Visualization and Computer Graphics* (2022). DOI: 10.1109/TVCG.2022.3172560.

[61] S. Ratzenböck, J. E. Großschedl, T. Möller, J. Alves, I. Bomze, and S. Meingast. *Significance Mode Analysis (SigMA) for hierarchical structures. An application to the Sco-Cen OB association*. 2022. arXiv: 2211.14225 [astro-ph.GA].

[62] S. Ratzenböck, S. Meingast, J. Alves, T. Möller, and I. Bomze. "Extended stellar systems in the solar neighborhood-IV. Meingast 1: the most massive stellar stream in the solar neighborhood." In: *Astronomy & Astrophysics* 639 (2020), A64.

[63] J. Roberts, C. Headleand, and P. Ritsos. "Sketching Designs Using the Five Design-Sheet Methodology." In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 419–428. DOI: 10.1109/TVCG.2015.2467271.

[64] T. Robitaille, C. Beaumont, P. Qian, M. Borkin, and A. Goodman. *glueviz v0.13.1: multi-dimensional data exploration*. Version 0.13.1. Feb. 2017. DOI: 10.5281/zenodo.1237692.

[65] P. Rousseeuw. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis." In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53–65. ISSN: 0377-0427. DOI: https://doi.org/10.1016/0377-0427(87)90125-7.

[66] S. Meingast, J. Alves, and A. Rottensteiner. "Extended stellar systems in the solar neighborhood - V. Discovery of coronae of nearby star clusters." In: *Astronomy & Astrophysics* 645 (2021), A84. DOI: 10.1051/0004-6361/202038610.

[67] S. Meingast, J. Alves, and V. Fürnkranz. "Extended stellar systems in the solar neighborhood - II. Discovery of a nearby 120° stellar stream in Gaia DR2." In: *Astronomy & Astrophysics* 622 (2019), p. L13. DOI: 10.1051/0004-6361/201834950.

[68] T. Cantat-Gaudin, C. Jordi, A. Vallenari, A. Bragaglia, L. Balaguer-Núñez, C. Soubiran, D. Bossini, A. Moitinho, A. Castro-Ginard, A. Krone-Martins, L. Casamiquela, R. Sordo, and R. Carrera. "A Gaia DR2 view of the open cluster population in the Milky Way." In: *Astronomy & Astrophysics* 618 (2018), A93. DOI: 10.1051/0004-6361/201833476.

[69] T. Torsney-Weir, A. Saad, T. Moller, H.C. Hege, B. Weber, J.M. Verbavatz, and S. Bergner. "Tuner: Principled Parameter Finding for Image Segmentation Algorithms Using Visual Response Surface Exploration." In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 1892–1901. DOI: 10.1109/TVCG.2011.248.

[70] M. Taylor. "STILTS - A Package for Command-Line Processing of Tabular Data." In: *Astronomical Data Analysis Software and Systems XV*. Ed. by C. Gabriel, C. Arviset, D. Ponz, and S. Enrique. Vol. 351. Astronomical Society of the Pacific Conference Series. July 2006, p. 666.

[71] M. Taylor. "TOPCAT & STIL: Starlink Table/VOTable Processing Software." In: *Astronomical Data Analysis Software and Systems XIV*. Ed. by P. Shopbell, M. Britton, and R. Ebert. Vol. 347. Astronomical Society of the Pacific Conference Series. Dec. 2005, p. 29.

[72] R. Tibshirani, G. Walther, and T. Hastie. "Estimating the number of clusters in a data set via the gap statistic." In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.2 (2001), pp. 411–423.

[73] V. Fürnkranz, S. Meingast, and J. Alves. "Extended stellar systems in the solar neighborhood - III. Like ships in the night: the Coma Berenices neighbor moving group." In: *Astronomy & Astrophysics* 624 (2019), p. L11. DOI: 10.1051/0004-6361/201935293.

[74] J. Waser, R. Fuchs, H. Ribičič, B. Schindler, G. Blöschl, and E. Gröller. "World lines." In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1458–1467.

[75] K. Yamashita, S. McIntosh, Y. Kamei, A. Hassan, and N. Ubayashi. "Revisiting the Applicability of the Pareto Principle to Core Development Teams in Open Source Software Projects." In: *Proceedings of the 14th International Workshop on Principles of Software Evolution*. IWPSE 2015. Bergamo, Italy: Association for Computing Machinery, 2015, pp. 46–55. ISBN: 9781450338165. DOI: 10.1145/2804360.2804366.

[76] E. Zari, A. Brown, J. de Bruijne, C. Manara, and P. de Zeeuw. "Mapping young stellar populations toward Orion with Gaia DR1." In: *Astronomy & Astrophysics* 608 (2017), A148.