Uncover: Toward Interpretable Models for Detecting New Star Cluster Members

Sebastian Ratzenböck, Verena Obermüller, Torsten Möller, *Senior Member, IEEE*, João Alves, and Immanuel M. Bomze

Abstract—In this design study, we present Uncover, an interactive tool aimed at astronomers to find previously unidentified member stars in stellar clusters. We contribute data and task abstraction in the domain of astronomy and provide an approach for the non-trivial challenge of finding a suitable hyper-parameter set for highly flexible novelty detection models. We achieve this by substituting the tedious manual trial and error process, which usually results in finding a small subset of passable models with a five-step workflow approach. We utilize ranges of a priori defined, interpretable summary statistics models have to adhere to. Our goal is to enable astronomers to use their domain expertise to quantify model goodness effectively. We attempt to change the current culture of blindly accepting a machine learning model to one where astronomers build and modify a model based on their expertise. We evaluate the tools' usability and usefulness in a series of interviews with domain experts.

Index Terms—Interpretable models, model selection, novelty detection, star clusters

1 Motivation

Star clusters constitute the elementary building blocks of galaxies [45]. They provide probes for studying fundamental processes such as galaxy structure formation and evolution, stellar physics, and exoplanet evolution [55]. However, what astronomers know about stellar clusters is limited by the discovery process itself. Due to complex interactions with their dusty birthplaces, the tidal forces from the Milky Way, and unavoidable imperfect measurements and missing data, finding and extracting star clusters is challenging. Typically, new star clusters' discoveries consist of small high-confidence samples that minimize misclassification of stars. These high-fidelity samples are usually restricted to the dense cluster centers. However, larger samples would not only dramatically improve the quality of the derived cluster's physical parameters, but they also uncover the so far unseen low-density regions of stellar clusters. These low-density regions contain essential information on cluster formation and evolution [9], [23], [28], [52]. Although there is no conclusive methodology to identify new cluster members, the advent of deep, space-based all-sky surveys makes it a timely topic.

The search for new stars faces the challenges inherent to unsupervised clustering approaches. The absence of labeled data makes finding an optimal clustering result a highly nontrivial task. The two main challenges are hyper-parameter space exploration and result validation. To search for meaningful solutions, users often fall back to a laborious, manual trial-and-error process.

To mitigate the time spent blindly wandering through the hyper-parameter space, interactive tools such as Tuner [72], and Clustrophile 1+2 [13], [21] provide a systematic approach to hyper-parameter space navigation. Conversely, validation depends on the context of the analysis, the users' goals, and expertise. General purpose systems thus often make efforts to increase the interpretability of results beyond so called *internal validation* measures [48] based on cluster compactness and separation. These scores provide proxies for the goodness of a clustering result. However, since clustering results usually cannot be fully validated, internal validation measures should not be used to optimize clustering results.

This situation changes in the case of star clusters. Although no ground truth information is available for individual stars, systems of multiple stars can be validated by domain experts. General purpose visual cluster analysis tools often focus on data exploration and insight generation rather than generating an effective and accurate clustering result. Moreover, to generalize to a broad range of application scenarios tools such as Clustrophile 2 [13] hardly provide any clustering algorithms that can deal with complex feature spaces. Notably, in the search for new member stars of stellar clusters, we already have a set of previously identified members which currently available systems fail to incorporate.

- *Sebastian Ratzenböck is with Data Science Research Network, 1090 Vienna, Austria, and also with the Faculty of Computer Science, 1090 Vienna, Austria. E-mail: sebastian.ratzenboeck@univie.ac.at.*
- *Verena Obermüller is with the Faculty of Computer Science, 1090 Vienna, Austria. E-mail: a11711429@unet.univie.ac.at.*
- *Torsten Möller is with the Faculty of Computer Science, 1090 Vienna, Austria, and also with Data Science Research Network, 1090 Vienna, Austria. E-mail: torsten.moeller@univie.ac.at.*
- *João Alves is with the Department of Astrophysics, 1180 Vienna, Austria, and also with Data Science Research Network, 1090 Vienna, Austria. E-mail: joao.alves@univie.ac.at.*
- *Immanuel M. Bomze is with ISOR/VCOR, A-1090 Vienna, Austria, and also with Data Science Research Network, 1090 Vienna, Austria. E-mail: immanuel.bomze@univie.ac.at.*

Manuscript received 2 June 2021; revised 21 Apr. 2022; accepted 24 Apr. 2022. Date of publication 5 May 2022; date of current version 31 July 2023.
(Corresponding author: Sebastian Ratzenböck.)
Recommended for acceptance by I. Fujishiro.
Digital Object Identifier no. 10.1109/TVCG.2022.3172560

The given set of cluster members provides the chance of employing powerful novelty detection methods in which known stars are used as training samples.

Our goal is to enable astronomers to use their domain expertise to assess the quality of novelty detection models and in the process create interpretable (to astronomers) and accurate star classification models.

Given these design considerations, we present a five-stage workflow approach in which users (1) specify a priori knowledge in terms of constrained summary statistic ranges which influence the training of an ensemble of novelty detection models. Models are (2) clustered into user-defined groups which are subsequently (3) judged on their quality by domain experts. The users' quality assessment then updates the range of valid summary statistics. Subsequently, we support users to study and discover the effect of summary statistics on the shape of the predicted distribution in the context of their qualitative assessment. This gives users the opportunity to update their prior knowledge and influence the filter range (4). The updated statistics influence hyper-parameter restrictions on which a final large ensemble classifier is trained. Finally, the user is able (5) to filter out individual stars based on the prediction frequency across models, to finalize the novelty classifier. The contributions can be summarized in the following:

- We present a novel visually assisted workflow for finding appropriate hyper-parameters for highly flexible one-class support vector machines in the presence of training set contamination and extremely high outlier fractions (see Section 2).
- We introduce an analysis and abstraction of data, tasks, and requirements for the star formation domain (see Section 4).
- We breakdown the star classification process into small interpretable steps. We support users to apply their domain expertise to assess the goodness of trained models, effectively building confidence in the final classifier among domain experts (see Section 5).
- We validate our approach in two scientific use cases that demonstrate the efficiency and effectiveness of the Uncover interface in finding new stars (see Section 9).

2 Algorithmic and Domain Background

Our goal is to enable users to select meaningful models from the vast space of possible star classification solutions. Instead of guiding users through the hyper-parameter space we provide an overview of possible model configurations.

To facilitate model selection, we aim to increase the transparency and interpretability of individual models. To provide trust in selected models, we provide means of validating their outputs. We substitute unintuitive model hyper-parameters with a set of interpretable summary statistics and provide means to study their effect on the model outputs.

In the following, we discuss the necessary expertise to validate star clusters. We highlight and motivate clustering challenges in the context of star clusters more deeply. Subsequently, we discuss one class models and strategies to validate them.

2.1 Domain Background

Star clusters are dense groups of at least a few dozen stars. Although it is widely agreed that most stars form in stellar clusters [45] their exact formation history and subsequent evolution is currently subject to ongoing discussion [44], [76]. This discussion on fundamental star formation principles is fueled by the Gaia mission [29], [30], [31] which provides unprecedented positional and kinematic measurements of over 1.6 billion stars in our Milky Way. Since its public release the richness of the Gaia data has sparked a wave of discoveries of star clusters [10], [12], [16], [51]. By studying their size, age, and chemical compositions, stellar clusters provide valuable insights into galaxy formation, structure evolution, and stellar physics.

The precise study of physical processes and inference of physical model parameters is, however, limited by the discovery process. Star clusters appear as stellar over-densities in the space of position and velocity [42]. Due to physical processes such as complex interactions with the galaxy, imperfect measurements, and missing data, finding and extracting star clusters is challenging. Consequently, discoveries of new star clusters are often accompanied by a small high-confidence sample to avoid a high number of misclassified stars. Thus, when a new star cluster is discovered, domain scientists frequently sacrifice recall for high precision.

To infer physical quantities or test hypotheses on stellar physics and/or Galaxy structure and evolution, a sufficiently large sample of stars is needed. In these situations, a high recall is equally important. To uncover potentially new cluster members, star clusters are often subject to follow-up studies [9], [23], [28], [52]. Even though a set of high-fidelity stars already exists, these follow-up studies usually employ fully unsupervised learning, i.e., in data sets without labels indicating a class. Nevertheless, a common aim is to assign new members to previously identified stellar groups. We actually face a gray area between supervised and unsupervised learning, in statistical jargon between classification and clustering (not in the astronomy sense).

Recently however, novelty (or anomaly) detection approaches have been used to search for new member stars [37], [59]. Specifically, one-class support vector machines (OCSVM) [64] are trained on a set of high-fidelity member stars which are then able to identify unseen members. However, OCSVM classifiers are quite tedious to train. Their high flexibility and the lack of labeled outlier data limits their ability to generalize well on account of the provided training data only. Due to the lack of a clear objective function, domain experts usually fall back to manual trial-and-error processes.

Although no ground truth information is available for individual stars, ensembles of stars can be validated by domain experts. The distribution of stars in the positional and kinematic feature space, alongside their distribution in the Hertzsprung Russell diagram (HRD) provides evidence for or against a "true" star cluster hypothesis.

The HRD shows the evolutionary distribution of stars. It is a scatter plot in which the absolute magnitude of stars, a measure of their brightness, is plotted against the color, a measure of surface temperature, of the stars (see left side of Fig. 10). The position of a star on the HRD depends on a number of factors but notably on its mass, chemical composition, and age. During its life a star follows an evolutionary path

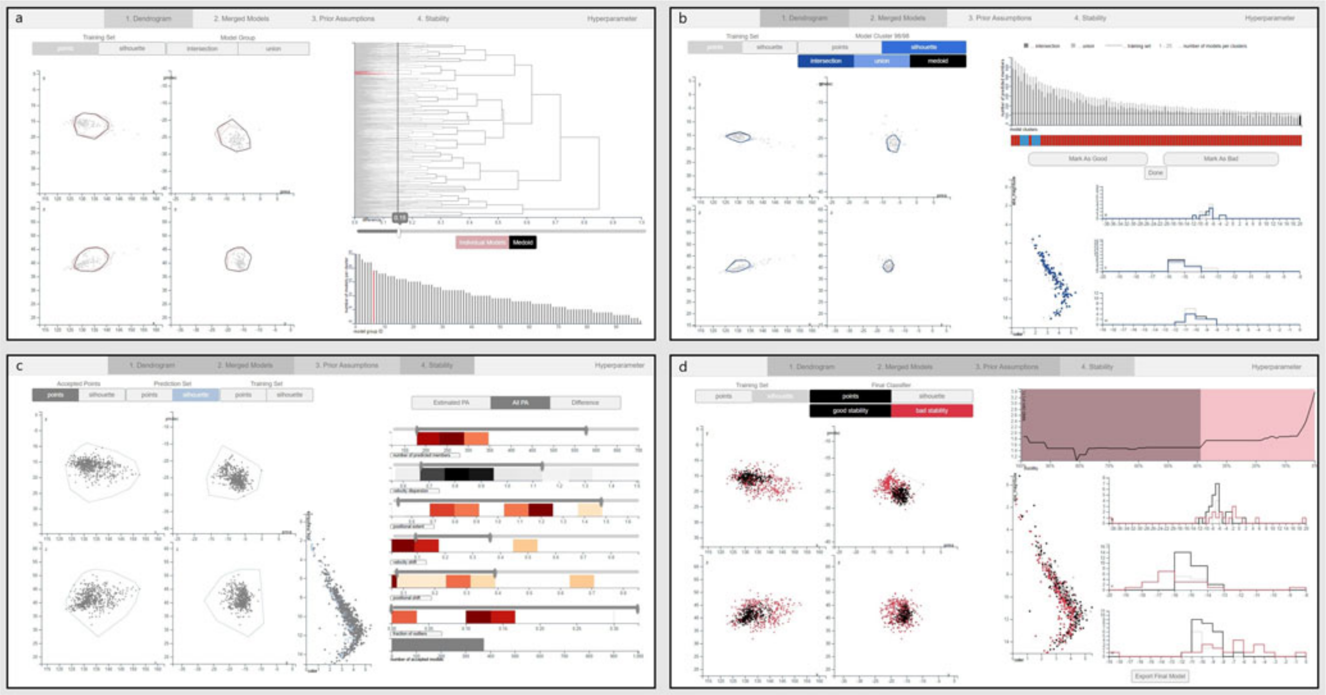


Fig. 1. Uncover Interface. (a) Dendrogram Tab showing the silhouettes of model groups from the selected difference threshold. (b) Model Group Tab showing the distributions of individual model groups. (c) Prior assumption Tab to set accepted ranges for selected summary statistics. (d) Stability Tab showing the final model ensemble and the prediction frequency of inferred members.

through the HRD. Stars in stellar clusters are "born" together, originating from large collapsing molecular clouds, and thus have the same age and chemical composition. Therefore, star cluster members with different masses are found to lie on and around (due to errors in the measurement process) a curve in the 2D plane.

We aim to provide a visual interface that enables astronomers to use their domain expertise to search for meaningful star classification results.

2.2 Algorithmic Background

In this work, we focus on one-class support vector machines following their recent success in identifying unseen members in star clusters [37], [59]. The OCSVM method is an outlier and novelty detection algorithm which learns a tight and smooth boundary around a target data set. By applying the kernel trick, this boundary is highly flexible and can describe non-linear, arbitrarily shaped boundary regions. However, its extraordinary versatility quickly becomes its greatest drawback, as its performance depends heavily on the choice of input hyper-parameters.

Due to the lack of labeled outlier data, traditional model selection techniques such as cross-validation cannot be applied. Since no second class can restrict model growth, models that encompass the whole feature space would achieve a perfect test score. The optimal hyper-parameter selection for one class models remains an open problem to this day [70].

2.2.1 Summary Statistics Heuristics

To formally quantify the goodness of a classifier, a set of labeled data instances is needed. In the case of one-class models and unsupervised learning algorithms, principled quantitative validation is impossible. Although the OCSVM approach uses a set of training data in an extended sense, the absence of data instances labeled as abnormal may lead to a trivial model including all observations.

Instead, summary statistics such as the Silhouette score [62] offer an automated model selection heuristics. A set of summary statistics and respective predefined ranges provide straight-forward model filters.

In contrast to the hyper-parameters of the classifier (e.g., the bandwidth parameter γ in the kernel function or the relaxation level ν, see below), statistics can be chosen by the domain experts themselves and carry an immediate meaning that can be interpreted by astronomers. Statistics such as velocity dispersion, or the center of mass are metrics already used to quantify star clusters [28], [51]. Such a domain specific model selection heuristics was applied by Ratzenböck *et al.* [59] who initially motivated and described the use of OCSVMs to search for new member stars. Instead of tuning the model hyper-parameters directly, they compiled six "interpretable" summary statistics and selected models based on a priori defined ranges of these statistics. The final star classification model results then from aggregating the prediction of accepted models.

In the limit of sufficient statistics [27] a set of maximum likelihood estimates for the parameters of the data generating model can be determined. This requires, however, a-priori knowledge on the nature of the joint probability distribution function. In reality, we are left with a set of observed data and insufficient but still informative statistics on the unknown population. Due to the unknown complex interaction and physical model uncertainties, the nature of the underlying star clusters distribution is indeterminate. Domain expertise and a high-fidelity training set can be used to create informed summary statistics for model selection.

A major drawback of using summary statistics for model validation is owed to the vague and abstract nature of prior knowledge. For example, instead of specifying explicitly how many stars domain experts predict to find, a common answer would be: "The population is expected to increase only slightly but not by much." Qualitative feedback provides an effective validation alternative over summary statistics that is much less sensitive to vague knowledge.

2.2.2 Qualitative Validation

In qualitative validation, users directly assess the model predictions. The goodness of star cluster models is tied to the distribution of inferred stars in the positional and kinematic features, respective to the training data. Especially the HRD provides means to support this decision.

Although summary statistics provide a fast model filter approach, visual analysis of inferred stars guarantees maximal confidence in the model. However, the manual inspection of up to millions [59] of models is practically infeasible. We aim to combine the best of both worlds by providing an update scheme on a-priori defined summary statistic ranges informed by manually validated models.

2.2.3 A Combined Approach

To enable astronomers to become model builders themselves, we provide domain experts with a variety of potential model candidates for validation. We derive limits to summary statistics from validated models, which provides an automatic model filter for a subsequent exhaustive model search.

The high flexibility of OCSVM models results in a vast space of possible star classification results. Thus, for consistent results we have to properly sample the space of possible solutions. To deal with a large number of model realizations, we adopt a clustering strategy in which similar models are first grouped and then jointly evaluated. A similar strategy can also be found in FluidExplorer [7] where similar frames in a fluid simulation are grouped together.

To account for different star cluster shapes and sizes we cannot impose a strict clustering rule. Instead, our goal is to enable domain experts to summarize models into user-defined groups. A reasonable and interpretable framework to introduce user control is through hierarchical clustering using a complete-linkage criterion [20]. Compared to other popular linkage criteria such as single or average linkage, the complete-linkage criterion provides an easy to grasp conceptual framework for users. Complete-linkage translates the merge threshold domain experts are able to modify into a maximal difference between individual models in a cluster. In addition, models in a group are expected to show characteristic properties, implying a small intra-group variation. Single-linkage, however, can lead to a very high intra cluster variation as it applies a local merge decision, compared to the complete-linkage criterion.

To represent the distance between two models we choose the symmetric difference cardinality (SDC) between inferred sets of stars. The SDC of two sets A and B is the number of elements which appear in either A or B but not in both. To deal with various cluster sizes we normalize the SDC by the union of both sets, a modification which still preserves the metric quality of the difference measure [81]. This metric measures the relative difference between models, that is the fraction of stars by which models differ. It provides an interpretable difference compared to more complex distances such as the Hausdorff distance [61] that is less sensitive to border point fluctuations.

By using a global-to-local [67] approach we essentially cluster the solutions that allows a domain expert to inspect groups of similar models instead of having to validate each model individually. Each model group summarizes a common classifier trait giving users a much more concise overview of the solution space. Instead of qualitatively inspecting models individually domain experts assess resulting model clusters, thus, scaling to thousands of models.

The number of trained models affects the wait time for the initial training phase and the interpretability of the hierarchical model grouping algorithm in subsequent workflow steps. This is contrasted by the need to properly sample the space of possible star classification results. To cover the hyper-parameter space quickly and evenly, we draw samples from the Sobol sequence [2], [69] until convergence. We stop the sampling process if the majority ($>90\%$) of the previous 50 hyper-parameter tuples lack significantly novel models. Model novelty is defined as a normalized SDC of at least 0.05 from previously trained models.

To improve the chance of finding many suitable models we pre-filter models based on initially defined summary statistic ranges based on a priori assumptions. This step limits the models presented to domain experts to plausible solutions.

Models are then trained according to Ratzenböck *et al.* [59] who have initially motivated and described the use of OCSVMs to search for new member stars. We briefly summarize the training steps here. To reduce overfitting, models are trained using five-fold cross validation,[1] admitting only classifiers above a test accuracy of 50% and a maximum standard deviation of 20% across folds. Although cross validation cannot be used to select an optimal model which generalizes well, we can get rid of models that are unable to identify already known members. To reduce the influence of potential contamination by outliers in the training set, bagging is performed. To do so, individual models are trained on a random subset using 80% of the initial training set.

Subsequently, domain experts are tasked to assess the goodness of self-defined model clusters. We derive updated ranges for the initially defined summary statistics from the user choices during the model validation step. Domain experts then have the option to further examine and modify the proposed ranges. Afterwards, a final training step that can be "run overnight" is performed where a much larger number of models are trained. These models have to comply to the user-informed, updated set of summary statistics. In the second, more detailed, model training step we are now able to narrow the hyper-parameter space which we derive from user validated models and the final summary statistic range. Therefore, we reduce the number of samples drawn from hyper-parameter space regions with unfit models while densely sampling from hyper-parameter space

[1] The training data is randomly shuffled before cross validation.

regions with a high acceptance rate. This strategy drastically reduces training time compared to a manual selection of initially vaguely informed summary statistic ranges [59].

3 Related Work

Although OCSVM models require a training step, the lack of labeled outlier data prevents us to quantify the goodness of trained models. Since solutions need a qualitative verification, the process of finding appropriate and effective models is inherently unsupervised.

A large variety of visual tools have been proposed to explore the space of possible classifiers. These tools are often based on visual hyper-parameter space exploration and aim to improve machine learning performance.

Uncover specifically focuses on one-class support vector machines and draws from prior work on optimal hyper-parameter selection.

3.1 Visual Clustering Analysis

A large body of previous work exists on interactive tools to support visual clustering analysis. General purpose tools provide means for exploratory data and cluster analysis. The Hierarchical Clustering Explorer (HCE [68]) is an early example of an interactive visualization tool that improves the users understanding of different clusters. HCE organizes the hierarchical cluster structure as a dendrogram with heatmaps. DICON [11] introduced techniques for comparing clustering results across different algorithms and even data sets. To facilitate cluster analysis DICON uses an icon-based cluster visualization that embeds statistical information into a multi-attribute display. Clustrophile 1+2 [13], [21] is a cluster analysis and exploration tool which guides a user through different choices of clustering hyper-parameters and provides interpretable cluster explanations.

Extensive work has been done on incorporating user feedback into the clustering process. ClusterSculptor [54] enables users to intervene in the clustering processes. Users can iteratively re-organize and interact with clusters using expert knowledge. The system aims to derive clustering rules from these examples. Schreck *et al.* [65] integrate user feedback to influence the result of SOM clusterings of trajectory data. Matchmaker [46] extends ideas from HCE [68] allowing users to modify clusterings by grouping data dimensions. Open-Box Spectral Clustering [66] is an interactive tool that visualizes mathematical quantities involved in 3D spectral clustering. The system provides hyper-parameter value suggestions and immediately reacts to user feedback to increase the quality of image segmentation. Packer *et al.* [56] present a distance-based spatial clustering approach and provide a heuristics computation of input hyper-parameters that supports the search for meaningful cluster results. ReVision [80] allows users to steer hierarchical clustering results by utilizing both public knowledge and private knowledge from users. By reformulating this knowledge into constraints, the data items are hierarchically clustered using an evolutionary Bayesian rose tree.

Conceptually similar research to ours include Geono-Cluster [18] and PK-clustering [58]. Geono-Cluster enables biologists to insert their domain expertise into clustering results. The tool displays the expected clustering results to users based on a small subset of data. The system estimates users' intentions and generates potential clustering results. PK-clustering [58] enables users to input prior knowledge and explore the space of clustering results in the context of the provided prior knowledge. The study of consensus between prior assumptions and cluster results allows users to acquire and update their prior knowledge.

In contrast to previous works we shift the focus from data exploration and insight generation towards effective model generation targeted at a single cluster. We also incorporate previously identified members which currently available systems fail to consider by using a supervised novelty detection approach.

3.2 OCSVM Hyper-Parameter Selection

Optimal hyper-parameter selection for one class models remains an open problem [70]. In the following, we discuss automated as well as visually supported model selection approaches.

3.2.1 Automatic Hyper-Parameter Selection

To mitigate the non-trivial selection process of OCSVM hyper-parameters, automatic hyper-parameter selection approaches have been proposed, which should provide suitable results. Automatic strategies either provide selection heuristics, or focus on producing a set of pseudo-outliers [4], [22], [24], [70], [71], [74]. These artificial outliers are subsequently used as an opposing class to the training data during cross-validation. Heuristics are often limited to specific kernel parametrizations. As RBF kernels bring a high degree of model flexibility most heuristics usually focus on them [26], [34], [43], [75], [78].

Both automatic approaches, however, often assume a problem in which the target class is sufficiently represented while the other class has almost no measurements in comparison [70]. This class imbalance assumption towards the training set is in stark contrast to stellar clustering where the target class is a minority embedded in, and outnumbered by, a background of non-member stars. Furthermore, automatic methods usually provide point estimates for hyper-parameters, providing only a single model to infer new member stars with.

Even in the case of optimal model hyper-parameters, one-class algorithms exhibit poor performance [71], which we can combat by using non-optimal learners in an ensemble approach. Bagging estimators improve the performance and robustness of the prediction [36]. Additionally, point estimates cannot adapt to specific user expectations and introduce errors in the case of noisy training data. Since residual contamination in the training sample from non-member stars is expected, we have to consider that OCSVM classifiers can be sensitive to contamination from outlier data [39], [47]. In this case, the OCSVM classifiers tend to skew toward the anomalies. Amer *et al.* [1] propose to mitigate the influence of outliers by altering the OCSVM objective function introducing training sample weights. Instead of tweaking the objective function, Ghafoori *et al.* [33] introduce a pre-processing step which removes anomalies from the training set and simultaneously tries to estimate suitable hyper-parameters. Both approaches, however, need some form of outlier

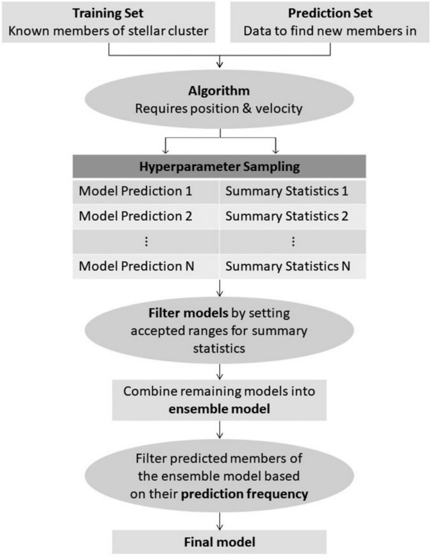

Training Set
Known members of stellar cluster
Prediction Set
Data to find new members in
Algorithm
Requires position & velocity
Hyperparameter Sampling
Model Prediction 1
Summary Statistics 1
Model Prediction 2
Summary Statistics 2
Model Prediction N
Summary Statistics N
Filter models by setting accepted ranges for summary statistics
Combine remaining models into ensemble model
Filter predicted members of the ensemble model based on their prediction frequency
Final model

Fig. 2. Schematic data flow of Uncover.

estimate, be it either through the distance to the data centroid [1] or via a k-NN density estimate [33] implying that outliers occur towards the border, or in low density regions of the training set. While this assumption is sufficient for many applications, we cannot generalize this to star clusters where contamination depends greatly on the training set selection method.

3.2.2 Visual OCSVM Hyper-Parameter Estimation

A different and more user-centered approach to find a suitable model was presented by Xie *et al.* [79] in which the OCSVM classifier is trained in an active learning scenario. User feedback on uncertain samples near the decision boundary updates the decision boundary.

Although active learning is able to adapt to specific user expectations, it fails in the context of star clusters. Data instances can rarely, if ever, be assessed on an individual basis. Conversely, however, it is very much possible for domain experts to discern a genuine star cluster from an incoherent system of stars.

4 Data and Task Analysis

We now discuss the data and tasks, and a derived workflow to support the search of new star cluster members. The data flow and workflow are schematically depicted in Figs. 2 and 3, respectively.

4.1 Data

The main data source is the aforementioned Gaia data set [29], [30], [31], a tabular data set containing measurements of over 1.6 billion stars in our Milky Way. Features relevant for this analysis constitute continuous, real-valued measurements of position and velocity, and color and absolute magnitude information which are used for model fitting, and validation, respectively. Users input two separate data sources, a training set and a prediction set. The latter is used to infer cluster membership with trained models. We note here that the full 3D kinematic information is available only for a small subset of stars in the Gaia data set. As discussed in Ratzenböck *et al.* [59], during training a reduced 2D velocity space is used, called *proper motion* space. Stars that have the full 3D kinematic information are used to validate models.

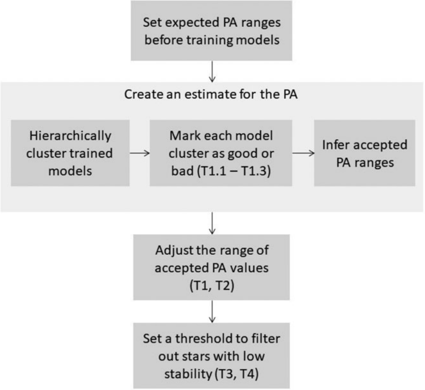

Set expected PA ranges before training models
Create an estimate for the PA
Hierarchically cluster trained models
Mark each model cluster as good or bad (T1.1 – T1.3)
Infer accepted PA ranges
Adjust the range of accepted PA values (T1, T2)
Set a threshold to filter out stars with low stability (T3, T4)

Fig. 3. Schematic workflow of the tool.

To speed up the inference process it is advised to provide a small subset of stars in the positional vicinity of the training set where new stars are assumed to lie in. Usually, both the training and prediction set are subsets of the Gaia catalogue. In principle, these two data sources can originate from different star catalogues as long as the feature set is identical.[2]

4.1.1 Model Abstraction

The OCSVM model can be abstracted as a basic deterministic input-output model converting input tuples to outputs.

Given the input hyper-parameters γ, ν, and $\frac{c_x}{c_v}$ and the training set, OCSVM constructs a decision surface that aims to maximize the separation between the training data and the origin. The resulting model is a decision hyper surface enclosing the training data in the input space which constitutes a binary function that classifies new data as in- or outliers. The hyper-parameter γ is related to the RBF kernel and controls the region of influence of support vectors. The variable ν provides an upper bound on the fraction of outliers and at the same time a lower bound on the fraction of support vectors used to construct the decision surface. The hyper-parameter $\frac{c_x}{c_v}$ provides a scaling relationship between positional and proper motion features [59]. Both subspaces are weighted equally when $\frac{c_x}{c_v} = 1$ in which case the variance in both feature spaces is the same.

The kernelized nature of OCSVMs provides an extremely flexible model that adapts well to arbitrary cluster shapes observed in star clusters. In extreme cases, a strongly concave shape is observed resulting from projection effects due to the lack of radial velocities.

Among the outputs are a Boolean member classification for each star in the prediction set and a set of six informative summary statistics derived from the predicted members.

4.2 Summary Statistics

Here we make use of the following summary statistics defined by Ratzenböck *et al.* [59]:

[2] In case two different source catalogues are used, special care must be taken to correctly consider differences in statistical and systematic errors between them.

The "number of predicted stream members" is the amount of cluster members a trained model infers from the given prediction set.

The statistics "positional extent" and "velocity dispersion" measure the mean deviation from inferred cluster members from the training set centroid in position and proper motion space, respectively.

The relative position or systematic shift of inferred stars compared to the training set in these two subspaces is characterized by "positional shift" and "velocity shift". These statistics characterize the distance between the centroids of training and inferred stars.

Lastly, "fraction of outliers" utilizes information of stars in the training set and inferred stars that have radial velocity measurements. Models that show significantly different 3D velocities than the training set are considered outliers. This statistic measures the fraction of inferred stars with radial velocities that are outside the 3σ region of training set stars in marginal 3D velocity distributions.

The authors referred to these summary statistics as *prior assumptions* (PA) which we use synonymously in the following sections.

4.3 Task Analysis

We aim to enable astronomers to update vague prior knowledge on the number, location, and movement of unidentified stars, altogether six summary statistics. The assessment of the goodness of multiple models should thereby provide the necessary information to reduce the uncertainty in these summary statistics.

To facilitate this transition the user has to be able to validate and influence the model selection process down to the individual classifier. With this characterization in mind, we carry out a task analysis. To facilitate comparison to other works, we try to provide abstract reasoning *why* a task is performed [5].

T1 Verify/Validate a trained model via its predicted members. To validate models, summary statistics usually provide too little information to inform a confident decision. Instead, domain experts use qualitative judgement to assess the goodness of models, requiring the following. First, users have to be able to assess the distribution of predicted stars in the space of *position and velocity* (*T1.1*) and compare them to the training set. Second, the distribution of stars in the *HRD* provides additional evidence for or against a valid star cluster (*T1.2*).

T2 Identify suitable summary statistics ranges. Ranges on summary statistics provide a filter criterion during the full training process (see Section 2.2.3) to automatically remove unfit models. We derive updated ranges for each of the six statistics from the users' qualitative model assessment. However, to provide insight into these filters, users have to be able to study and discover their effect on the shape of the predicted distribution (*T2.1*). Users should also be able to explore and analyze the distribution of assessed models in the context of summary statistics (*T2.2*). This gives them the opportunity to update and substantiate their prior knowledge. Finally, users must be able to apply their updated knowledge and interactively refine filter ranges on summary statistics (*T2.3*).

T3 Explore the effect of stability filters on the inferred stars. Stability is the prediction frequency of stars across the model ensemble. Stars with high stability are thus inferred by most of the models and vice-versa. Ratzenböck *et al.* [59] have shown that removing stars with low stability values removes disproportionately more contaminant stars than genuine cluster members, effectively cleaning the sample. We aim to facilitate the exploration of different stability thresholds to study the effects on the ensemble model prediction. Using their domain expertise, users should thereby be able to select a meaningful stability threshold.

T4 Present the inferred cluster members of the final ensemble model. To validate the final ensemble model we present the distribution of training and inferred stars in the space of position and velocity, in combination with the HRD. In case domain experts see the final model as unfit, users can go back to previous workflow steps and intervene accordingly.

T5 Summarize the model ensemble in terms of their hyper-parameters at different workflow steps. To provide a transparent view on the OCSVM algorithm, users have to be able to inspect the distribution at any time. To understand the model selection effect on the hyper-parameters, we present the distribution of hyper-parameters of models that domain experts deemed fit in comparison to the initially trained, unfiltered models.

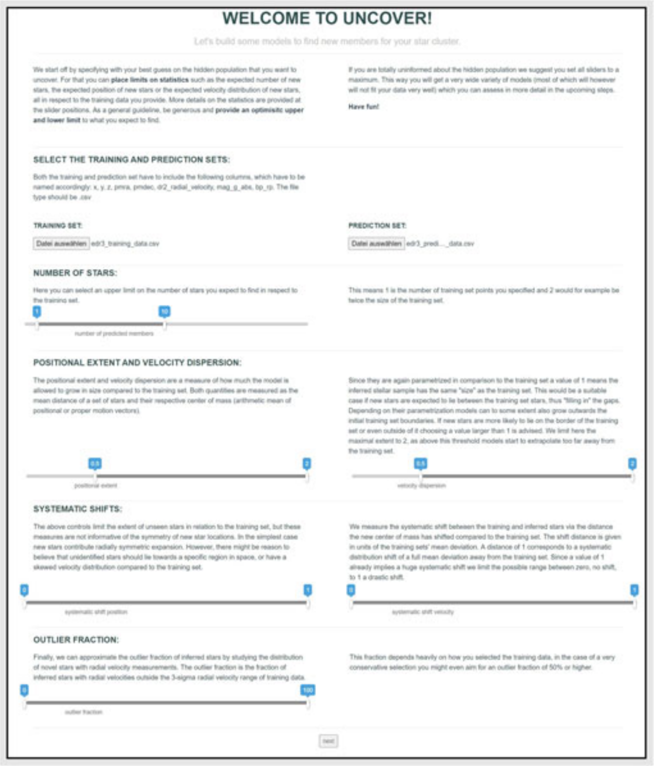
5 Uncover Interface

We now discuss the design of the tool starting with the general layout, followed by descriptions of the individual tabs and visualization components.

5.1 Layout

The prototype comprises six different views in total, one for each workflow step as well as an additional view for showing information on the hyper-parameters. At the top of each view is a tab-bar, which enables the user to navigate between the different workflow steps and the hyper-parameter view. The tabs are arranged in order of the workflow steps, see Fig. 1 for an overview of the interface from the second to the last workflow steps. The first workflow phase is shown in Fig. 4.

For each of the five tabs, the same general layout (see supplemental material Fig. 2, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TVCG.2022.3172560) is used to create a consistent interface throughout the tool. If users can already anticipate where certain information will be presented, users can more quickly adapt to a new view and therefore reduce mental overhead [63]. We divide the interface into two equal sized sections, the *scatterplot matrix* and *update* section. The update section in the right half adapts to each workflow step. It contains interaction components which facilitate cluster selection, model navigation and assessment, updating and refining prior knowledge (*T2*), and stability threshold exploration and selection (*T3*). The left section provides a reduced scatterplot matrix, which shows the position and proper motion dimensions separately. This is used to display the multi-dimensional data set (*T1*, *T4*). Depending on the respective workflow step, different data aspects and models are highlighted. This can be the training set, different model groups, medoid models, the



WELCOME TO UNCOVER!

Let's build some models to find new members for your star cluster.

We start off by specifying with your best guess on the hidden population that you want to uncover. For that you can **place limits on statistics** such as the expected number of new stars, the expected position of new stars or the expected velocity distribution of new stars, all in respect to the training data you provide. More details on the statistics are provided at the slider positions. As a general guideline, be generous and **provide an optimisitc upper and lower limit** to what you expect to find.

If you are totally uninformed about the hidden population we suggest you set all sliders to a maximum. This way you will get a very wide variety of models (most of which will however will not fit your data very well) which you can assess in more detail in the upcoming steps.

Have fun!

SELECT THE TRAINING AND PREDICTION SETS:

Both the training and prediction set have to include the following columns, which have to be named accordingly: x, y, z, pmra, pmdec, dr2_radial_velocity, mag_g_abs, bp_rp. The file type should be .csv

TRAINING SET: Datei auswählen edr3_training_data.csv

PREDICTION SET: Datei auswählen edr3_predi..._data.csv

NUMBER OF STARS:

Here you can select an upper limit on the number of stars you expect to find in respect to the training set.

This means 1 is the number of training set points you specified and 2 would for example be twice the size of the training set.

number of predicted members: 1 – 10

POSITIONAL EXTENT AND VELOCITY DISPERSION:

The positional extent and velocity dispersion are a measure of how much the model is allowed to grow in size compared to the training set. Both quantities are measured as the mean distance of a set of stars and their respective center of mass (arithmetic mean of positional or proper motion vectors).

Since they are again parametrized in comparison to the training set a value of 1 means the inferred stellar sample has the same "size" as the training set. This would be a suitable case if new stars are expected to lie between the training set stars, thus "filling in" the gaps. Depending on their parametrization models can to some extent also grow outwards the initial training set boundaries. If new stars are more likely to lie on the border of the training set or even outside of it choosing a value larger than 1 is advised. We limit here the maximal extent to 2, as above this threshold models start to extrapolate too far away from the training set.

positional extent: 0.5 – 2

velocity dispersion: 0.5 – 2

SYSTEMATIC SHIFTS:

The above controls limit the extent of unseen stars in relation to the training set, but these measures are not informative of the symmetry of new star locations. In the simplest case new stars contribute radially symmetric expansion. However, there might be reason to believe that unidentified stars should lie towards a specific region in space, or have a skewed velocity distribution compared to the training set.

We measure the systematic shift between the training and inferred stars via the distance the new center of mass has shifted compared to the training set. The shift distance is given in units of the training sets' mean deviation. A distance of 1 corresponds to a systematic distribution shift of a full mean deviation away from the training set. Since a value of 1 already implies a huge systematic shift we limit the possible range between zero, no shift, to 1 a drastic shift.

systematic shift position: 0 – 1

systematic shift velocity: 0 – 1

OUTLIER FRACTION:

Finally, we can approximate the outlier fraction of inferred stars by studying the distribution of novel stars with radial velocity measurements. The outlier fraction is the fraction of inferred stars with radial velocities outside the 3-sigma radial velocity range of training data.

This fraction depends heavily on how you selected the training data, in the case of a very conservative selection you might even aim for an outlier fraction of 50% or higher.

outlier fraction: 0 – 100

Fig. 4. First workflow step of Uncover.

models at the minimum and maximum of each summary statistic range, or the final ensemble model. We describe the scatterplot matrix component in more detail when it first appears on the "Dendrogram Tab" in Section 5.2.2.

5.2 Visualization Components

In this section, the chosen visualizations as well as their intended function for carrying out the corresponding workflow step are discussed in more detail.

5.2.1 Dendrogram Tab

Based on training set characteristics, the number of hyperparameter tuples needed to properly cover the space of possible star classification results can be in the hundreds or even thousands. However, users cannot be tasked to assess the quality of each individual model. Instead, we support users to choose groups of similar models that can be assessed together instead of individually.

To summarize possible model clustering configurations the update section of this view, shown in Fig. 9, features a dendrogram. The dendrogram provides an overview of the clustering hierarchy of models resulting from a complete-linkage agglomerated clustering approach. At each step, the two model clusters with the smallest relative difference in predicted points are combined into the same cluster. This difference value is shown on the x-axis of the dendrogram plot. To be able to perceive structure in the dendrogram towards smaller distances, its lines become progressively thinner from 1 to 0 to avoid visual overlaps. The slider can be used to set a threshold for the difference, where merging will stop, so that models with a difference greater than the selected value will remain in separate groups. The bar chart below the dendrogram shows the number of models in each group resulting from the current threshold.

5.2.2 Scatterplot Matrix

The scatterplot matrix, seen in the left half of the view (see Fig. 1a), shows the model groups resulting from the current cut along with the training set representing the baseline. We aim to provide an overview of the clustering results and thus facilitate a comparison between the resulting groups of models. We choose two summary operands for model groups; the union and the intersection of points inferred by individual models in a group.

The intersection provides a summary of common model features across a group. By comparing the intersection and union of stars inferred by group members we provide an estimate of within-group variation that is easy to understand. The further the two group summaries diverge, the less the models in a group form a coherent cluster. In such cases, a better clustering result can be achieved by reducing the difference threshold.

We choose to summarize models as silhouettes in the scatterplot matrix which shows the maximal extent region of the predicted distribution in each projection. It acts as a visual simplification of a model in the form of a convex hull around the predicted points. Compared to scatter points, silhouettes allow users to easily compare multiple model groups. In this scenario, indicating group identity is non-trivial in scatter points. Not only is the use of color limited to roughly six to seven groups [53] but a large amount of points are also part of multiple groups which drastically increases the amount of unique visual encodings required. Therefore, since examining the stars inferred by individual model groups and assessing their goodness is not the purpose of this workflow step, but of the following one, we omit the display of scatter points here.

To assess a group of models in detail, in order to determine whether they form a meaningful unit, domain experts can explicitly display the convex hull of each model in a given group. Additionally, users can highlight the group medoid, the representative model of the group. It provides an opportunity to identify group characteristics like a certain set of stars that this model group has in common. A comparison with the remaining models should provide further insight into the model variation within the group. By studying the group medoid and the overlap and variation between silhouette shapes, users can determine an appropriate threshold.

5.2.3 Model Group Tab

In this workflow step, users are tasked to assess the goodness of model groups defined in the previous step.

The scatterplot matrix view displays the model groups one after the other. The user can choose to plot the training set in the same scatterplot matrix to compare it to the currently shown model group. Depending on the use case, the distribution of inferred member stars in positional and proper motion space in relation to the training can give strong indications towards a good and bad model, see Fig. 5.

To validate the models, positional and kinematic information is provided in the scatterplot matrix (*T1.1*) and an

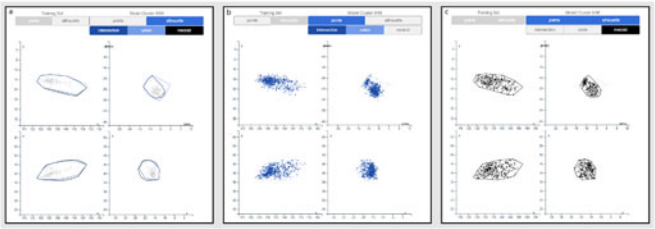


Fig. 5. Different selections in the scatterplot matrix during the model group validation step. On the left (a) three group summaries are highlighted; the union and intersection of predicted members, as well as the group medoid are shown in the form of silhouettes. The middle view (b) shows predicted members as scatter points. In the right view (c) a combination of both model group summaries – points and silhouettes – are used. The training data are displayed as gray scatter points in all three views.

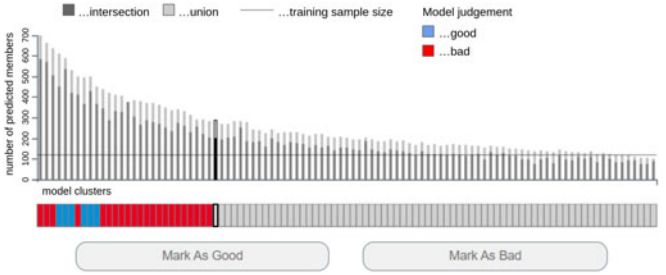


Fig. 6. Visualization components for assessing the model groups. The stacked bar chart shows the number of predicted members resulting from the union and intersection of models in each group. The blue and red cells indicate a good or bad marking of the corresponding model group, respectively.

HRD is provided in the update section (*T1.2*). To assess a model, domain experts can verify whether predicted members are distributed in a narrow line in the HRD according to the training set or not. To leverage the kinematic information from the inferred stars for model validation, we provide two kinematic views, see Fig. 1b for more details. First, the proper motion information used for training is displayed in the scatterplot matrix. Second, the Cartesian velocity distribution is displayed in three histograms next to the HRD, see Fig. 10. To verify that the predicted member stars constitute a stellar cluster the Cartesian velocity distribution should roughly follow a normal distribution and not deviate significantly from the training set [42].

For both the training set and the model group, the user can switch between viewing individual data points, which are classified as members, or the silhouette thereof by clicking the buttons labeled accordingly. For each model group, the user can choose to view the union of all inferred members or only the stars that are predicted members across all models in the group. This selection can be done via the buttons labeled “union” or “intersection” above the scatterplot matrix, respectively, see Fig. 5. Additionally, to facilitate the judgement of a group of models, the medoid can be selected as a model representative. Compared to the union and intersection of stars via a model group, the medoid represents an individual model in which characteristic model details become more apparent.

The number of predicted points for both the union and intersection of the models in each group is visualized using a stacked bar chart in the update section. Since users aim to find additional star cluster members, this is the most important summary statistic which provides an overview across model groups.

The number of inferred stars is considered to strongly correlate with model goodness. Depending on the level of prior knowledge, domain experts might be interested in specific ranges of inferred member sizes. Therefore, we sort the bar chart in descending order by union size to support different levels of attention during the users’ workflow. This allows users to string together groups that require more attention during the validation process, followed by groups that require less consideration. This attention bias applies, for example, to models that find about the same number or even fewer members compared to the training set. These models typically require less validation effort, as their member size alone indicates a lack of new discoveries. To facilitate a comparison with the training set a horizontal dashed line is drawn indicating its size.

Once the user has come to a decision regarding the suitability of the currently shown model group, the corresponding button in the update section shown in Fig. 6 can be clicked to either mark it as “good” or as “bad”. Afterward, the next model group is shown. The bar between the buttons and the bar chart highlights the progress and gives an overview of the model group assessment. Blue and red indicate a good or bad model group, respectively. Model groups which have not been assessed yet are colored in gray. When all the model groups have been evaluated, the button labeled “Done” can be clicked to generate the estimate for the accepted Prior Assumption (PA) ranges.

5.2.4 Prior Assumption Tab

The third workflow step, seen in Fig. 1c, supports the analysis and possible adjustment of the PA ranges which result from the previous step.

Each of the six PAs and the corresponding derived ranges are visualized with the help of *scented widgets* [77]. The *widget* is made up of two sliders, one for the minimum and one for the maximum of each PA range. These are positioned on top of the *visual scent* in the form of a bar, which shows the distribution of PA values from all models as a heatmap. The darker the luminance of a cell, the more models have a PA value in the matching range. See Fig. 7 for a detailed view of the PA range interface.

The heatmap provides a means to analyze the distribution of assessed models in the context of summary statistics (*T2.2*). Users can explore correlations between a selected PA and the remaining PAs. By clicking on a cell of a given PA, all the models whose PA value lies in the selected range will be highlighted in the remaining five heatmaps. To visually separate the distribution of models from a selected heatmap cell in the other summary statistics we choose a red colormap, as can be seen in the update section of Fig. 1c. This interaction supports users to find model trends and correlations.

When first opening the tab, the initial slider position shows the estimate for the accepted PA ranges created in the previous step. For each PA, the sliders are placed according to the minimum and maximum PA value of the models that were marked as good. Additionally, by clicking on the buttons

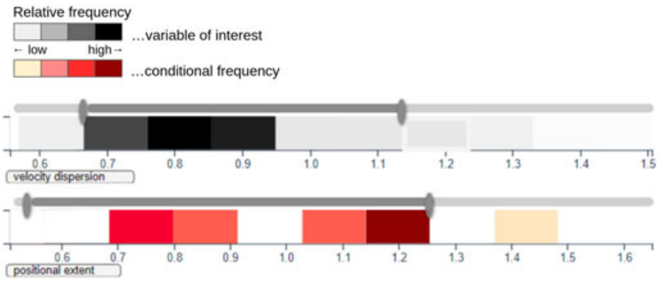


Fig. 7. Scented widget for setting the accepted PA range. The heatmap visualizes the distribution of PA values among either all trained models, all models marked as good, or the difference between the two. Clicking on a heatmap-cell will update the remaining heatmaps to show the models in the selected range in a red colormap.

above the heatmaps users can either study the distribution of all initially trained models, under "All PA," the distribution of models assessed as good by the user, under "Estimated PA," and the models judged as bad, under "Difference".

By studying the correlation between models' summary statistics and the distribution of "good" and "bad" models, users can substantiate their prior knowledge and interactively refine PA filter ranges giving them the opportunity to (*T2.3*) precisely control the properties of the final model.

The slider positions in the heatmaps correspond to models shown in the scatterplot matrix. We provide a what-if-analysis where users can isolate the effects of a single PA and study its influence on the inferred stars. At each slider position, stars inferred by models which adhere to the selected filter criterion are shown in the scatterplot matrix. The minimum slider position corresponds to a minimum set of stars that these models can identify. A sensible choice is to require models to at least identify large parts of the training set. The maximum slider represents stars that can be detected by models up to the selected PA value. To illustrate the effect of the whole slider range, we exclusively show stars that can be detected beyond the minimum slider value. Stars associated with the minimum slider position are colored in light blue whereas stars associated with the maximum slider position are highlighted in a darker shade of blue. Light gray points in the background indicate stars outside the maximum slider position which are not inferred by selected models. When no PA is selected, stars inferred by models which adhere to the slider range filters are highlighted in gray in the scatterplot matrix.

By interactively changing the slider position for one or multiple PAs users can study the influence of various summary statistics on the shape and distribution of inferred stars in position, velocity, and the HRD, as well as the correlations between the model behavior and a given summary statistics in more detail (*T.2.1*). This interaction provides additional information for users to update their prior belief and refine given filter ranges (*T2.3*).

The bar visualization at the very bottom of the right half encodes the number of models out of the initially trained ensemble that pass the PA range filter. Thus, it informs the user how restrictive their current ranges are setup. The bar length is updated whenever slider positions are changed.

In this step outlier models can motivate an alternative workflow. As discussed, Uncover is not aimed at providing means for exploratory data analysis, but rather for effective model building. Thus, identifying and characterizing outlier models is not an important task for the user. Especially outlier models which are classified as "bad" require no further investigation on the users' end. Hence, outlier models are not explicitly marked as such in the tool to avoid drawing unnecessary attention to them. However, if an outlier model is considered "good," a user may find few appropriate models in the initially trained model ensemble. In an effort to increase the diversity of "good" models, domain experts might want to restart the training process. This can be done by returning to the first workflow step and modifying initial summary statistic ranges. A sensible choice is to center updated ranges around those of given outlier models. Their respective summary statistics can be analyzed in the heatmap view, see Fig. 7.

5.2.5 Stability Tab

The last step of the workflow is dedicated to the final ensemble model and the stability of its predicted members. The final ensemble model is the result of combining the predictions of the models that fulfil the PA restrictions set up in the previous workflow steps. The final predicted distribution of the stellar cluster in question is shown in the scatterplot matrix, the HRD, as well as in the histograms displaying the Cartesian velocity, as shown in Fig. 1d. These views also show the training set to facilitate comparison (*T1*) and allow the user to verify that the final ensemble model creates a suitable prediction.

To switch between viewing the points and the silhouettes, the buttons on top of the scatterplot matrix can be used. However, in the case of the HRD, showing the silhouette of a distribution is not always useful. Stars in different stages of stellar evolution typically occupy distinct subregions of the diagram [30], so a predicted distribution that comprises stars in varying evolutionary phases could form separate clusters with large gaps between them in the HRD. Drawing a silhouette encompassing all the points would then result in a shape that is too coarse and does not reflect the underlying distribution in a useful manner.

The threshold for the stability can be set with the help of a *scented widget* [77] which features a line chart showing the stability in percent and the median absolute deviation (MAD) of predicted members from the expected 3D velocity. The right side of the brush on the line chart can be moved to set the minimum stability for the final classifier. This also updates the presentation of the predicted distribution in the remaining plots: All points with accepted stability are colored black, while points that will be filtered out because their stability is too low are shown in red, an example of this can be seen in Fig. 1d. If the silhouette-button is selected, the silhouette resulting from the points with acceptable stability is colored black while the silhouette encompassing all predicted members is shown in red.

5.2.6 Hyper-Parameter Tab

The previous tab aims to provide supplementary information, available online, on the hyper-parameters of the accepted models. Even though the aim of the tool is to relieve the user of having to work directly with the hyper-parameters, information on them should still be available to

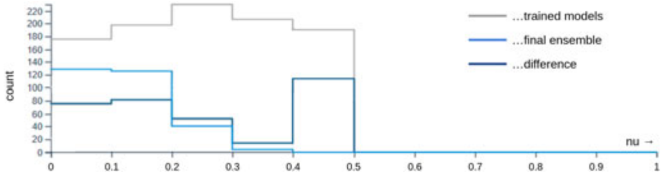


Fig. 8. Histogram of one of the three hyper-parameters. The dark gray line corresponds to all trained models, the bright blue line to all accepted models. The difference between these two is shown by the the dark blue line. The light gray lines show the accepted models from previous settings.

give the user the possibility to get a better understanding of them. For each of the three hyper-parameters [59] γ, ν and $\frac{c_x}{c_v}$, there is one histogram showing the number of accepted models as well as the number of trained models for each possible hyper-parameter value as can be seen in Fig. 8.

6 Design Rationale

In this section, the motivations behind the chosen visual encodings are discussed regarding the data and task abstraction.

6.0.0.1 Why a tab-based interface? An alternative to tabs would be to present the necessary visualizations for carrying out the different workflow steps on a single page. This would remove the need to switch between different views and therefore impose a lower cognitive load on the user [14]. However, the number of required visualizations would not fit onto a single screen in a reasonable size without requiring to scroll the page. The different workflow steps were therefore separated into individual tabs to ensure that the visualizations for each step fit appropriately onto a single screen. To reduce the cognitive load caused by the user, each tab functions as a self-contained unit. This means that every tab contains all necessary visualizations to fully carry out the associated tasks and does not require the user to remember information shown in previous tabs.

To further reduce the mental load when opening a new tab the interface layout (see supplemental material Fig. 2), available online, remains the same throughout the tool. Especially the scatter plot matrix and the actual data displayed on the left-hand side stay the same across the entire tool.

Although the tool supports a linear workflow once the initial model ensemble is trained, the user can decide to go back to any workflow step and modify their decisions. The first step is not part of this tab interface since it amounts to starting the tool up again from the beginning, which requires another time-consuming training step.

6.0.0.2 Why scatterplot matrices? The 3D position and proper motion of the stars are always shown using scatterplot matrices, since this is the standard way of visualizing stellar clusters in the field of astronomy. Other visualization methods for multi-dimensional data were initially considered but found to be unsuitable in this context. Parallel coordinates [40] would be an alternative to scatterplot matrices; However, they are not commonly used in astronomy and would therefore not be very intuitive for the target audience. Additionally, scatterplots can act as 2D projections of the underlying real-world objects described by the data, which reside in a 3D space, and are therefore much more straightforward to interpret. 3D scatterplots were also considered, but showing the data in 3D can result in a variety of problems [53]. The large number of points that need to be presented would make the use of 3D especially challenging, since this would lead to a significant amount of occlusion and thus make it hard to get a full view of the distribution.

Showing the apparent motion of stars as an oriented line anchored at their sky position is a common visual encoding used in astronomy, e.g., de Zeeuw *et al.* [19] famously showcase three co-moving groups in the nearby Scorpius-Centaurus OB association. The instantaneous velocity of a star is encoded as a small arrow whose origin is at its position. The length of the arrow encodes speed while the angle channel represents the direction of movement.

However, this hybrid visualization presents the following problems. First, available velocity information is limited to proper motion data which may suffer from drastic projection effects. Large stellar populations such as the Meingast 1 stream [51] show significant distortions in proper motion which can lead users to misguided decisions. Second, trained OCSVM models are bound by given training data. Thus, inferred stars will largely have similar velocities which eliminates random background noise that can cause a visual pop-out effect. Third, not only is the angle channel less accurately perceived as the positional channel [49] but it also lacks an absolute scale. Due to variable star cluster positions and projection effects, changes in angle do not carry an unambiguous meaning.

Users have to judge star clusters by considering their positional and kinematic distribution of its members where especially the search for outliers constitutes an essential task. These tasks benefit from the more effective spatial position channel compared to the less accurately perceived angle channel [38]. Combined with discussed projection effect issues we thus refrain from adding velocity information into the positional scatter plot via the angle channel.

6.0.0.3 Why a reduced scatterplot matrix? The reduced scatterplot matrix, which shows the position and proper motion dimensions separately, was designed to use the available screen space more efficiently. Since it consists of fewer panels than the full scatterplot matrix, it would have the advantage of displaying the individual scatterplots in a bigger size. Both versions were presented to astronomy experts in the course of iterative prototyping.

6.0.0.4 Why silhouettes in addition to points? An integral part of each step in the workflow is to compare different distributions of stars. This can mean comparing model groups or the final ensemble model to the training set to see if they are a good match or examining the models with the smallest and largest permitted value of each PA to see how much they differ. To facilitate this comparison, the silhouettes of the distributions can provide a summary of their overall shape that is easier to interpret [17].

6.0.0.5 Why scented widgets with heatmaps? An integral part of the workflow is to set accepted PA ranges that result in a suitable final classifier. To facilitate this task, supplementary information, available online, is necessary to help the user make an informed decision about how to best constrain the PA. The corresponding sliders were therefore implemented as *scented widgets* [77], which feature additional visualizations in the form of heatmaps to show the number of models for each PA value. Histograms were considered as an alternative to heatmaps. These would enable the user to read the exact number of models in each bin more accurately. But this comes at the cost of taking up more screen space, since the histograms would need to be shown in an appropriate size to discern the exact length of a bar. However, in this context, communicating the exact number of models in each bin of the PA range is not the goal. Instead, the user should get an idea of the overall distribution of PA values to see if many models are concentrated around a certain range and then set the sliders accordingly. This can be accomplished adequately with the help of heatmaps; therefore histograms would only provide a level of detail that is not necessary in this context at the cost of taking up more screen space.

6.0.0.6 Why histograms and scatter plots to show kinematics? Kinematic information is used during both training and validation. As discussed, due to largely missing radial velocity measurements, models are trained with two instead of three velocity features. Although star clusters are approximately normally distributed in Cartesian velocity space, the observed 2D velocities, i.e., proper motions, are subject to sometimes drastic projection effects. The observed, potentially highly concave shapes contribute to the difficulties of traditional clustering approaches.

Since very few stars have radial velocity measurements and, thus, 3D velocity information, stellar kinematics is commonly displayed in proper motions space. Typically, proper motion information is displayed in scatter plots as discussed above.

Stars that have 3D velocities are used as model validation. Models that show significantly different 3D velocities than the training set are removed. This information is quantified in the PA "fraction of outliers" which measures the fraction of inferred stars with radial velocities that are outside the 3σ region in marginal 3D velocity distributions of the training set. To validate models qualitatively, domain experts are tasked to compare the training set distribution against the distribution of inferred star cluster members. To compare the velocity distributions, two design alternatives were considered, scatter plots and histograms.

As discussed above, other designs such as parallel coordinates are unfamiliar to the domain experts and were judged as confusing. Domain experts noted that both design alternatives facilitate the comparison between distributions. Due to the low number of stars, however, users noted that histograms make it easier to reason on the distribution shape. Especially determining if the data are approximately normally distributed, and thus providing means of validating a model, was perceived to be easier with histograms.

Thus, three histograms showing the Cartesian, marginal velocity distributions are provided alongside the HRD to support model validation. We add them to the Model Group Tab and Stability Tab, see Sections 5.2.3 and 5.2.5, respectively. In the PA Tab, see Section 5.2.4, the velocity histograms are not included as model validation plays a secondary role in this workflow step. Additionally, the summary statistic "fraction of outliers," whose influence the user can interactively explore already supports a quantitative evaluation of 3D velocities.

6.0.0.7 Why histograms for showing hyper-parameters? The distribution of hyper-parameters for the accepted models could also be presented using the same heatmaps as before. But an additional task is to provide an overview on the OCSVM hyper-parameter distribution at any time. This helps domain experts to gain insights on the effects that model selection via summary statistics has on the model hyper-parameters themselves (*T.5*). Therefore, the chosen visualization type should support displaying multiple distributions at once. When using heatmaps, this can be achieved by juxtaposing several heatmaps to show different distributions [35]. However, length can be judged more accurately than color [53], which would be an advantage of histograms. Instead of juxtaposing several histograms, another option is to add a line corresponding to each distribution that needs to be presented on the same plot as shown in Fig. 8. Superimposing the distributions in this manner also allows for easier comparison between the heights of different bins [41].

7 Implementation

The front-end visualization components and interactions are implemented in JavaScript and use d3.js and vue.js. Additional data processing for building the dendrogram and calculating the PA for the trained models has been separated from the front-end and is implemented using python and the web framework Flask.

We made use of the libsvm [15] OCSVM implementation available in scikit-learn [57] library and the Sobol sampling sequence implemented in SciPy [73]. The software is publicly available to foster open science and reproducibility.[3]

8 Evaluation

In the following, we discuss both formative and summative evaluation steps we performed in the course of this design study.

8.1 Formative Evaluations

During progressing from initial paper prototypes to the final implementation, the tool was repeatedly presented to experts in data visualization, statistics, as well as astronomy and subsequently underwent changes based on their feedback.

In the first stage of the design process, the paper prototypes were reviewed in the group of co-authors featuring a visualization expert, a domain expert, and an applied mathematician and statistician. The feedback sessions were held bi-weekly and lasted for 3 months.

After arriving at a final design, the paper prototype was implemented as an interactive wireframe tool which was used during the second review stage. This interactive prototype was then tested and discussed in two interview session with astronomy experts who had no previous involvement in the design process. The two domain experts had different levels of prior knowledge about the underlying algorithm. One test user already had substantial experience using the algorithm and could therefore confidently navigate through the views of the prototype. The domain expert noted that the proposed design would alleviate many challenges she was facing when searching for new member stars. The second user was less familiar with the inner workings of the algorithm, but with the help of additional explanations it

[3] https://github.com/ratzenboe/uncover-tool

was possible to correctly interpret the visualizations and carry out the associated tasks. These interviews suggest that additional documentation for the final tool would be helpful. The user tests also resulted in a number of feature requests, which were taken into consideration when creating the implementation of the final prototype. For a more detailed description of the prototyping process see Section 2 of the supplemental material, available online.

8.2 Summative Evaluations

To evaluate the usability of the tool, the final implementation was tested by nine domain experts in astronomy. Three of the participants were experts in the field of stellar clusters while the remaining six test users classified their knowledge as intermediary level knowledge of the subject. All test subjects had previous experience in validating stellar clusters via the HRD and 3D velocities. Six of the participants had no previous experience using the algorithm, the other three test users had worked with the algorithm at least once and were familiar with the basic properties of it. One of the users had already tested the interactive prototype in a formative test, the remaining users were new to the tool.

Each test user was given 60 minutes to test the tool. Every session started with a brief introduction to provide some information on the aim of the test as well as the algorithm itself. The participants were then asked to use the tool and instructed to "think-aloud" while doing so. Additional explanations for the individual steps were provided upon request.

All users tested the tool with the same training set as well as a subset of the Gaia DR2 catalogue as the prediction set and were tasked with finding new member stars for the given training data. Since the main purpose of these tests was to assess the usability of the tool and creating a suitable prediction for a stellar cluster might take more refinement than was possible during the given time, the resulting outputs were not checked for their correctness.

The last 20 minutes of each test were reserved for filling out the SUS-questionnaire [6] as well as conducting a short interview. The resulting SUS-score was 78.06 with a standard deviation of 7.89, which would indicate acceptable usability [3].

Participants, who were inexperienced with the underlying algorithm, mentioned that providing more information and explanations as part of the tool would be helpful. Specifically, the statistical foundations of the PA and stability were deemed as hard to interpret without additional explanations. The dendrogram was considered the least intuitive visualization component by test users regardless of their experience level with the algorithm. All users requested extra explanations but after its purpose and use was explained, the information it provides was deemed very helpful by all participants, for more details see supplemental material Section 1, available online. The intended purpose of the remaining views was more straightforward to understand without requiring supplementary clarifications, available online. The overall workflow and sequence of steps was judged as well thought-out. They fully cover the necessary functionality for the required data analysis according to all test users. All participants considered the tool a helpful addition to the algorithm and stated that they would prefer it rather than working directly with the algorithm. This suggests that our main goal for the tool was fulfilled. One test user, who had made use of the algorithm before, also expressed interest in using the tool for their future work.

9 Scientific Use Cases

In this section, we showcase the efficiency and effectiveness of the Uncover interface in finding new stars to a given stellar cluster in a case study and use case, respectively. More details on the interactive session discussed in the case study can be found in screenshots throughout this paper and in the accompanying video.

9.1 Case Study: Searching for New ρ Oph Members

Recently, Grasser *et al.* [37] have detected over 100 new member stars for the ρ Oph cluster using an ensemble of OCSVM models. Following model selection ideas from Ratzenöck *et al.* [59] the authors had to limit the result space via prior assumption ranges that models have to adhere to. Since the ρ Oph cluster has been thoroughly investigated in multiple earlier studies [8], [25], [60] their search for new members was highly uninformed. Due to the lack of substantial prior knowledge Grasser *et al.* had to resort to randomly sampling different prior assumption ranges and analyze the results manually. In the following we repeat this study, using the same training and prediction set, and showcase a more efficient workflow using Uncover.

The target user is an astronomer who aims to find additional sources in the ρ Oph cluster. Upon starting the tool, the user specifies her prior knowledge on the yet unidentified stellar population via range sliders, see Fig. 4. Since the ρ Oph cluster has been studied extensively in the past, she suspects to find new members predominantly outside the currently known cluster region. She limits "positional extend" and "velocity dispersion" to $0.5 - 2$ and "number of predicted members" to $1 - 10$ times the training set size. Having no specific prior knowledge on limiting other summary statistics she leaves the remaining sliders at their initial position (see Fig. 4). On clicking close she arrives at the next workflow step.

In the *Dendrogram Tab* the astronomer aims to group multiple models into a meaningful unit. She explores various difference thresholds in the dendrogram via the slider interface. Looking for a sensible clustering of models she clicks on the leftmost bar in the bar chart to inspect its individual group members in the scatterplot matrix, see Fig. 9. She notes that models within each group – represented as silhouettes – highlight very different characteristics of the ρ Oph cluster. To find more meaningful model groups which capture a single model characteristic she gradually decreases the differences threshold while inspecting silhouettes of corresponding group members. At the normalized SDC threshold of 0.15 she stops her search (see Fig. 1a); not only does she find low variation between individual silhouettes and the group medoid, but also different model groups seem to capture different aspects of the ρ Oph cluster.

In the *Model Group Tab* the astronomer is tasked with assessing the goodness of previously defined model groups. She analyzes the distribution of inferred members in the

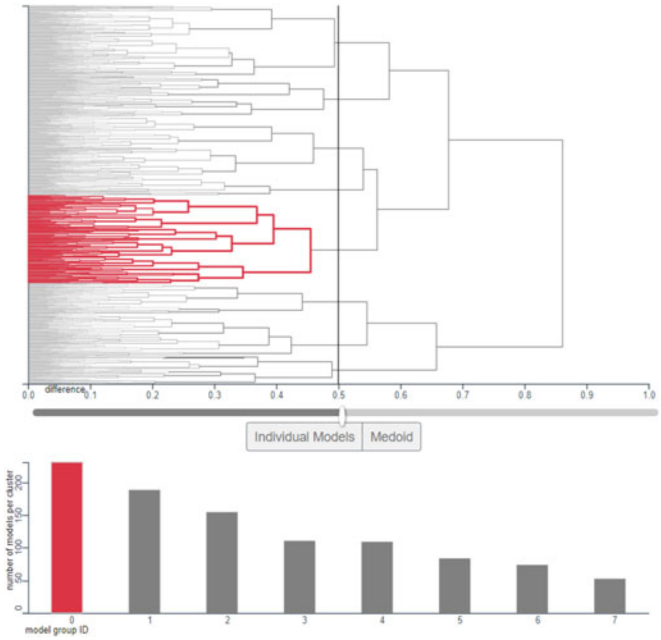


Fig. 9. Dendrogram with corresponding slider and bar chart which shows the number of model groups and their respective size.

space of position and proper motions as well as the HRD and three velocity histograms. She finds that the first three models show a large scatter in the HRD and velocities, see Fig. 10. Co-evolving stellar groups show a distinguished narrow and well-defined sequence in the HRD as well as a roughly Gaussian distributed 3D velocity. Thus, such increased scatter indicates a large contamination fraction in the sample. Consequently, she marks these groups as "bad".

The next few models are the most interesting ones. They show a second population near the training set, highlighted in gray, in position and kinematic space, but the HRD and 3D velocities indicate a good model. The astronomer conjectures that she just uncovered a second stellar population right next to ρ Oph cluster (see Fig. 5), which Grasser *et al.* [37] recently discovered. She assesses models featuring this second population as "good". Model groups thereafter do not capture the adjacent population and are thus rejected by the astronomer.

In the *PA tab* the user observes that her initially defined ranges on summary statistics have been updated based on the distribution of accepted models. All updated ranges are rather concentrated towards larger values. By interactively changing the minimum and maximum position of the range slides she learns that the PA "velocity dispersion" and "velocity shift" have a stark influence on the second population as well as the distribution of inferred stars in the HRD. Models with both a lower velocity dispersion and shift are not able to infer stars from the second population. By clicking through the heatmap bins (see Fig. 7) the astronomer finds that a very large fraction of outliers does not correlate with a large "fraction of outliers" scores indicating that these ranges are a good selection. To study the influence of the positional extent statistic on the inferred stars, the astronomer clicks on the corresponding heatmap row. The scatterplot matrix now highlights in dark blue possible stars that can be inferred at the maximum slider position. She increases the slider position and sees a large increase in scatter in position and the HRD. She conjectures that this selection criterion correlates with an increasingly contaminated sample. Thus, she reduces the maximal slider again to exclude likely non-cluster members. By clicking on the next tab she arrives at the *Stability Tab*.

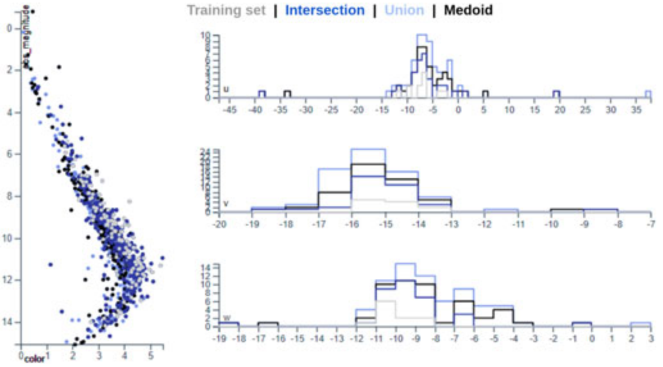


Fig. 10. Histograms showing the Cartesian velocity distribution and HRD from the Model Group Tab.

The astronomer explores the influence of various stability thresholds by brushing the line graph representing the 3D velocity dispersion on the right-hand side. She observes a sudden drop in scatter around the training sources in the HRD at about 85% at which coincides with a rapid drop in the 3D velocity dispersion (see Fig. 1d). At this threshold both populations seem to be perfectly separated. Not only does the second population, colored in red, show an older age indicated by a shift in the HRD, its 3D velocity distribution is also slightly shifted compared to the training set.

A comparison with the results reported by Grasser *et al.*[4] yields a 93.3% recall and a relative percentage difference in detected stars of only 3.8%.[5] These findings highly coincide with their validated study results [37], which the user was able to replicate with ease in a single session using Uncover.

Finally, she exports the final model with a click on the "Export Final Classifier" button.

9.2 Use Case: Finding New Corona-Australis Members

Uncover was used to discover previously unknown members of the Corona-Australis cluster. Due to its proximity and young age Corona-Australis is an important laboratory for studying the star formation process. We chose Corona-Australis specifically, as our collaborators at the Astronomical Institute are interested in finding the most complete sample of the star cluster for follow-up studies. The stellar content of Corona-Australis has recently been studied by Galli *et al.* [32] who identified 313 members, 262 of which were new. Although this star census seemed comprehensive, our collaborators were curious to see if any additional stars could be identified using the Uncover interface.

We defined the training set by taking bonafide samples from the Galli *et al.* [32] catalog with radial velocities v_r between $-8 < v_r < 5$ km s^{-1} yielding 21 stars. As discussed in Ratzenböck *et al.* [59], consistent radial velocity

[4] The catalog is publicly available at: https://cdsarc.cds.unistra.fr/viz-bin/cat/J/A+A/652/A2

[5] The results are compared by applying a quality filter in accordance to Grasser *et al.* [37].

measurements increase the likelihood of being true cluster members, reducing the chance of contamination in the model output. To increase the number of training examples, we combine Gaia radial velocity measurements with radial velocities from the APOGEE-2 [50] survey adding another 29 stars to the training set. The prediction set includes Gaia EDR3 [31] stars in a 50 pc radius around Corona-Australis.

Our collaborators were primarily interested in finding additional high-fidelity members of Corona-Australis. Therefore, they carefully selected models during the *Model Group Tab* which minimize contamination in the HRD and 3D velocity axes. During the *PA Tab* they were able to discover that models with high values of the summary statistic "fraction of outliers" correlated with large values in all the remaining summary statistics, except for "positional extent". A final extraction was reached after slightly decreasing the maximum slider of "fraction of outliers" until some remaining contaminating sources in the HRD disappeared.

We find 66 potential new members and, thus, increase the number of known members by over 20%, which will improve downstream analyzes.

10 Lessons Learned and Future Work

During the user tests carried out as part of the design process, it became evident that simplifying prototypes too much can be misleading and results in users making incorrect assumptions about the tool. For example, based on a paper prototype, it is easy to underestimate the size of the dendrogram and the resulting number of model clusters or the number of points shown in the scatterplot matrices. In these situations, it was necessary to ask follow-up questions to clear up any misunderstandings. It also became evident that not all visualization types are intuitive enough to be understood without supplementary explanations, available online. Specifically, test users were unfamiliar with the dendrogram, which was difficult to interpret correctly. Once the dendrogram's purpose was explained, the information it provides was deemed very helpful by all participants. Additionally, users unfamiliar with the algorithm's inner workings also requested more explanations of certain context-specific phrases such as stability or prior assumptions. It would therefore be worth considering adding short explanations at the appropriate places. However, while we would love to make this tool a self-contained software this goes beyond the scope of this paper which presents a proof-of-concept implementation. Including an interactive tutorial and some explanatory texts would be a very important first step in commercializing this tool.

The user tests also revealed that the preferred visualization types are the ones that are already commonly used in the field of astronomy. For example, to show the predicted points in 5D, there would be several visualization options aside from scatterplot matrices. But since scatterplot matrices are the standard way of visualizing stars' positions and velocity, employing this visualization type will make the tool more intuitive and easier to use for its intended audience. Therefore, it is more beneficial to the tool's usability to focus on creating an ideal workflow instead of trying to come up with new visualization types.

It also became evident that the features requested by the test users do not always comply with what would be best from a data visualization perspective. For instance, it was requested to apply a continuous colormap to the scatter points in the last view to encode the stability. However, accurately judging whether two points have the same saturation is difficult [53]. To avoid these deficiencies, we used two different hues to color the points with accepted and rejected stability.

11 Conclusion

In this design study, we presented Uncover, a visual workflow that aims to increase the interpretability and accuracy of new detection models. We provided a transparent and interactive workflow that supports astronomers searching for new members of stellar clusters. By enabling astronomers to use their domain expertise to assess a models' goodness, we shifted their workflow from blindly using machine learning algorithms to building validated and powerful models.

Our workflow and design study provides general guidelines for interactive and interpretable model generation in unsupervised scenarios, where qualitative model validation is possible. These guidelines are the following:

First, provide an overview of possible model solutions. In this design study, we summarize the vast space of possible model configurations using a hierarchical clustering approach. We support users to control the granularity of model groups and the means of validating them. Users are able to identify interesting model trends and, thus, summarize the solution space into interesting sub-groups.

Second, we aim to increase trust in the selected model groups by validating their outputs. We support the assessment of user-defined model groups by providing domain specific validation tools such as the Hertzsprung–Russell diagram or 3D kinematic information.

Third, we substitute the unintuitive model hyper-parameters with a set of interpretable summary statistics. We incorporated the domain experts' a priori knowledge on yet unidentified cluster members as a filter criterion for models. Upon startup, users are tasked to specify their prior knowledge via interpretable summary statistics. To support users to update and substantiate their initial, potentially vague prior knowledge we provide the following: We determine updated ranges from users' qualitative model assessment for these summary statistics. Users are able to explore correlations between summary statistics via linked heatmaps. Further, we support What-If analyzes where users can study the effect of individual summary statistics on model outputs. Users could update their prior knowledge and, thus, tailor model filters to their needs, effectively becoming model builders in the process.

In a usability study with nine domain experts and two use cases, we observe users efficiently building effective and high-performance novelty detection models which support our claims. Given these results, we see great potential in extending the presented workflow to any unsupervised algorithm beyond OCSVMs. However, due to their high flexibility and ability to incorporate a training set OCSVMs present serious advantages when efficient model building is the primary goal.

Due to its success, Uncover is going to be deployed in the lab of our astronomical collaborators. The goal is to give bachelor students the opportunity to explore and expand the stellar content of the local Milky Way cluster. These students can now perform the same tasks efficiently and effectively previously executed by experienced researchers. In doing so, they provide valuable new probes for the study of star formation processes, the formation of galaxies, and their structural evolution.

Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful feedback. This work has used data from the European Space Agency (ESA) mission *Gaia* (https://www.cosmos.esa.int/gaia), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, https://www.cosmos.esa.int/web/gaia/dpac/consortium). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement.

References

[1] M. Amer, M. Goldstein, and S. Abdennadher, "Enhancing one-class support vector machines for unsupervised anomaly detection," in *Proc. ACM SIGKDD Workshop Outlier Detection Description*, 2013, pp. 8–15.

[2] I. A. Antonov and V. M. Saleev, "An economic method of computing LP$_\tau$-sequences," *USSR Comput. Math. Math. Phys.*, vol. 19, no. 1, pp. 252–256, 1979.

[3] A. Bangor, P. Kortum, and J. Miller, "Determining what individual SUS scores mean: Adding an adjective rating scale," *J. Usability Stud.*, vol. 4, no. 3, pp. 114–123, May 2009.

[4] A. Bánhalmi, A. Kocsor, and R. Busa-Fekete, "Counter-example generation-based one-class classification," in *Proc. Eur. Conf. Mach. Learn.*, 2007, pp. 543–550.

[5] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2376–2385, Dec. 2013.

[6] J. Brooke, "SUS: A 'quick' and 'dirty' usability scale," in *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Eds., New York, NY, USA: Taylor and Francis, Jun. 1996, pp. 189–194.

[7] S. Bruckner and T. Möller, "Result-driven exploration of simulation parameter spaces for visual effects design," *IEEE Trans. Vis. Comput. Graphics*, vol. 16, no. 6, pp. 1468–1476, Nov./Dec. 2010.

[8] H. Cánovas *et al.*, "Census of ρ ophiuchi candidate members from gaia data release 2," *Astron. Astrophys.*, vol. 626, Jun. 2019, Art. no. A80.

[9] T. Cantat-Gaudin *et al.*, "A gaia DR2 view of the open cluster population in the milky way," *Astron. Astrophys.*, vol. 618, Oct. 2018, Art. no. A93.

[10] T. Cantat-Gaudin *et al.*, "Gaia DR2 unravels incompleteness of nearby cluster population: New open clusters in the direction of perseus," *Astron. Astrophys.*, vol. 624, Apr. 2019, Art. no. A126.

[11] N. Cao, D. Gotz, J. Sun, and H. Qu, "DICON: Interactive visual analysis of multidimensional clusters," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 12, pp. 2581–2590, Dec. 2011.

[12] A. Castro-Ginard, "Hunting for open clusters in gaia DR2: 582 new open clusters in the galactic disc," *Astron. Astrophys.*, vol. 635, Mar. 2020, Art. no. A45.

[13] M. Cavallo and Ç. Demiralp, "Clustrophile 2: Guided visual clustering analysis," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 267–276, Jan. 2019.

[14] P. Chandler and J. Sweller, "Cognitive load theory and the format of instruction," *Cogn. Instruct.*, vol. 8, no. 4, pp. 293–332, 1991.

[15] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2 no. 3, May 2011, Art. no. 27.

[16] B. Chen, E. D'Onghia, J. Alves, and A. Adamo, "Discovery of new stellar groups in the orion complex. Towards a robust unsupervised approach," *Astron. Astrophys.*, vol. 643, Nov. 2020, Art. no. A114.

[17] W. S. Cleveland and R. McGill, "The many faces of a scatterplot," *J. Amer. Statist. Assoc.*, vol. 79, no. 388, pp. 807–822, 1984.

[18] S. Das, B. Saket, B. C. Kwon, and A. Endert, "Geono-cluster: Interactive visual cluster analysis for biologists," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 12, pp. 4401–4411, Dec. 2021.

[19] P. T. de Zeeuw, R. Hoogerwerf, J. H. J. de Bruijne, A. G. A. Brown, and A. Blaauw, "A hipparcos census of the nearby OB associations," *Astronomical J.*, vol. 117, no. 1, pp. 354–399, Jan. 1999.

[20] D. Defays, "An efficient algorithm for a complete link method," *Comput. J.*, vol. 20, no. 4, pp. 364–366, Jan. 1977.

[21] Ç. Demiralp, "Clustrophile: A tool for visual clustering analysis," in *Proc. KDD Workshop Interactive Data Exploration Analytics*, 2016, pp. 37–45.

[22] H. Deng and R. Xu, "Model selection for anomaly detection in wireless ad hoc networks," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, 2007, pp. 540–546.

[23] C. Ducourant *et al.*, "Proper motion survey and kinematic analysis of the ρ ophiuchi embedded cluster," *Astron. Astrophys.*, vol. 597, Jan. 2017, Art. no. A90.

[24] C. Désir, S. Bernard, C. Petitjean, and L. Heutte, "One class random forests," *Pattern Recognit.*, vol. 46, no. 12, pp. 3490–3506, 2013.

[25] T. L. Esplin and K. L. Luhman, "A survey for new stars and brown dwarfs in the ophiuchus star-forming complex," *Astronomical J.*, vol. 159, no. 6, Jun. 2020, Art. no. 282.

[26] P. F. Evangelista, M. J. Embrechts, and B. K. Szymanski, "Some properties of the Gaussian kernel for one class learning," in *Proc. Int. Conf. Artif. Neural Netw.*, 2007, pp. 269–278.

[27] R. A. Fisher, "On the mathematical foundations of theoretical statistics," *Philos. Trans. Roy. Soc. London. Ser. A.*, vol. 222, no. 594–604, pp. 309–368, Jan. 1922.

[28] J. Gagné *et al.*, "BANYAN. XI. The BANYAN Σ multivariate Bayesian algorithm to identify members of young associations with 150 pc," *Astrophysical J.*, vol. 856, no. 1, Mar. 2018, Art. no. 23.

[29] "Gaia Collaboration, The Gaia mission," *Astron. Astrophys.*, vol. 595, Nov. 2016, Art. no. A1.

[30] Gaia Collaboration, "Gaia data Release 2. Summary of the contents and survey properties," *Astron. Astrophys.*, vol. 616, Aug. 2018, Art. no. A1.

[31] Gaia Collaboration, "Gaia early data release 3: Summary of the contents and survey properties," *Astron. Astrophys.*, vol. 649, May 2021, Art. no. A1.

[32] P. A. B. Galli *et al.*, "Corona-australis DANCe I. Revisiting the census of stars with Gaia-DR2 data," *Astron. Astrophys.*, vol. 634, Feb. 2020, Art. no. A98.

[33] Z. Ghafoori, S. M. Erfani, S. Rajasegarar, J. C. Bezdek, S. Karunasekera, and C. Leckie, "Efficient unsupervised parameter estimation for one-class support vector machines," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 5057–5070, Oct. 2018.

[34] Z. Ghafoori, S. Rajasegarar, S. M. Erfani, S. Karunasekera, and C. A. Leckie, "Unsupervised parameter estimation for one-class support vector machines," in *Proc. 20th Pacific-Asia Conf. Adv. Knowl. Discov. Data Mining*, 2016, pp. 183–195.

[35] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts, "Visual comparison for information visualization," *Inf. Visual.*, vol. 10, no. 4, pp. 289–309, 2011.

[36] Y. Grandvalet, "Bagging equalizes influence," *Mach. Learn.*, vol. 55, no. 3, pp. 251–270, 2004.

[37] N. Grasser *et al.*, "The ρ ophiuchi region revisited with gaia EDR3. Two young populations, new members, and old impostors," *Astron. Astrophys.*, vol. 652, Aug. 2021, Art. no. A2.

[38] J. Heer and M. Bostock, "Crowdsourcing graphical perception: Using mechanical turk to assess visualization design," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2010, pp. 203–212.

[39] H. Hoffmann, "Kernel PCA for novelty detection," *Pattern Recognit.*, vol. 40, no. 3, pp. 863–874, 2007.

[40] A. Inselberg and B. Dimsdale, "Parallel coordinates: A tool for visualizing multi-dimensional geometry," in *Proc. 1st IEEE Conf. Visual.*, 1990, pp. 361–378.

[41] W. Javed, B. McDonnel, and N. Elmqvist, "Graphical perception of multiple time series," *IEEE Trans. Vis. Comput. Graphics*, vol. 16, no. 6, pp. 927–934, Nov./Dec. 2010.

[42] H. Kamdar, C. Conroy, Y.-S. Ting, A. Bonaca, M. C. Smith, and A. G. A. Brown, "Stars that move together were born together," *Astrophysical J.*, vol. 884, no. 2, Oct. 2019, Art. no. L42.

[43] S. Khazai, S. Homayouni, A. Safari, and B. Mojaradi, "Anomaly detection in hyperspectral images based on an adaptive support vector method," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 646–650, Jul. 2011.

[44] M. A. Kuhn, L. A. Hillenbrand, A. Sills, E. D. Feigelson, and K. V. Getman, "Kinematics in young star clusters and associations with gaia DR2," *Astrophysical J.*, vol. 870, no. 1, Jan. 2019, Art. no. 32.

[45] C. J. Lada and E. A. Lada, "Embedded clusters in molecular clouds," *Annu. Rev. Astron. Astrophys.*, vol. 41, no. 1, pp. 57–115, Jan. 2003.

[46] A. Lex, M. Streit, C. Partl, K. Kashofer, and D. Schmalstieg, "Comparative analysis of multidimensional, quantitative data," *IEEE Trans. Vis. Comput. Graphics*, vol. 16, no. 6, pp. 1027–1035, Nov./Dec. 2010.

[47] W. Liu, G. Hua, and J. R. Smith, "Unsupervised one-class learning for automatic outlier removal," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3826–3833.

[48] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *Proc. IEEE Int. Conf. Data Mining*, 2010, pp. 911–916.

[49] J. Mackinlay, "Automating the design of graphical presentations of relational information," *ACM Trans. Graph.*, vol. 5, no. 2, pp. 110–141, Apr. 1986.

[50] S. R. Majewski *et al.*, "The apache point observatory galactic evolution experiment (APOGEE)," *Astronomical J.*, vol. 154, no. 3, Sep. 2017, Art. no. 94.

[51] 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," *Astron. Astrophys.*, vol. 622, Feb. 2019, Art. no. L13.

[52] S. Meingast, J. Alves, and A. Rottensteiner, "Extended stellar systems in the solar neighborhood. V. Discovery of coronae of nearby star clusters," *Astron. Astrophys.*, vol. 645, Jan. 2021, Art. no. A84.

[53] T. Munzner, *Visualization Analysis and Design*. Boca Raton, FL, USA: CRC Press, 2014.

[54] E. J. Nam, Y. Han, K. Mueller, A. Zelenyuk, and D. Imre, "ClusterSculptor: A visual analytics tool for high-dimensional data," in *Proc. IEEE Symp. Vis. Analytics Sci. Technol.*, 2007, pp. 75–82.

[55] E. R. Newton *et al.*, "TESS hunt for young and maturing exoplanets (THYME). IV. Three small planets orbiting a 120 myr old star in the PiscesEridanus stream," *Astronomical J.*, vol. 161, no. 2, Jan. 2021, Art. no. 65.

[56] E. Packer, P. Bak, M. Nikkilä, V. Polishchuk, and H. J. Ship, "Visual analytics for spatial clustering: Using a heuristic approach for guided exploration," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2179–2188, Dec. 2013.

[57] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.

[58] A. Pister, P. Buono, J. D. Fekete, C. Plaisant, and P. Valdivia, "Integrating prior knowledge in mixed-initiative social network clustering," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 2, pp. 1775–1785, Feb. 2021.

[59] 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," *Astron. Astrophys.*, vol. 639, Jul. 2020, Art. no. A64.

[60] E. Rigliaco *et al.*, The Gaia-ESO survey: Dynamical analysis of the L1688 region in ophiuchus," *Astron. Astrophys.*, vol. 588, Apr. 2016, Art. no. A123.

[61] R. Rockafellar and R. J. B. Wets, *Variational Analysis, vol. 317 of Grundlehren der mathematischen Wissenschaften*. Berlin, Germany: Springer, 1998.

[62] P. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, no. 1, pp. 53–65, Nov. 1987.

[63] J. Scarr, A. Cockburn, and C. Gutwin, "Supporting and exploiting spatial memory in user interfaces," *Found. Trends Hum.- Comput. Interaction*, vol. 6, no. 1, pp. 1–84, Dec. 2013.

[64] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[65] T. Schreck, J. Bernard, T. Tekusova, and J. Kohlhammer, "Visual cluster analysis of trajectory data with interactive kohonen maps," in *Proc. IEEE Symp. Vis. Analytics Sci. Technol.*, 2008, pp. 3–10.

[66] T. Schultz and G. L. Kindlmann, "Open-box spectral clustering: Applications to medical image analysis," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2100–2108, Dec. 2013.

[67] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller, "Visual parameter space analysis: A conceptual framework," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 2161–2170, Dec. 2014.

[68] J. Seo and B. Shneiderman, "Interactively exploring hierarchical clustering results [gene identification]," *Computer*, vol. 35, no. 7, pp. 80–86, Jul. 2002.

[69] I. M. Sobol, "Uniformly distributed sequences with an additional uniform property," *USSR Comput. Math. Math. Phys.*, vol. 16, no. 5, pp. 236–242, 1976.

[70] D. M. J. Tax and R. P. W. Duin, "Uniform object generation for optimizing one-class classifiers," *J. Mach. Learn. Res.*, vol. 2, pp. 155–173, Mar. 2002.

[71] D. M. J. Tax and K.-R. Muller, "A consistency-based model selection for one-class classification," in *Proc. 17th Int. Conf. Pattern Recognit.*, 2004, pp. 363–366.

[72] T. Torsney-Weir *et al.*, "Tuner: Principled parameter finding for image segmentation algorithms using visual response surface exploration," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 12, pp. 1892–1901, Dec. 2011.

[73] P. Virtanen *et al.*, "SciPy 1.0: Fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[74] S. Wang, Q. Liu, E. Zhu, F. Porikli, and J. Yin, "Hyperparameter selection of one-class support vector machine by self-adaptive data shifting," *Pattern Recognit.*, vol. 74, pp. 198–211, Feb. 2018.

[75] S. Wang, J. Yu, E. Lapira, and J. Lee, "A modified support vector data description based novelty detection approach for machinery components," *Appl. Soft Comput.*, vol. 13, no. 2, pp. 1193–1205, 2013.

[76] J. L. Ward, J. M. D. Kruijssen, and H.-W. Rix, "Not all stars form in clusters - Gaia-DR2 uncovers the origin of OB associations," *Monthly Notices Roy. Astronomical Soc.*, vol. 495, no. 1, pp. 663–685, Jun. 2020.

[77] W. Willett, J. Heer, and M. Agrawala, "Scented widgets: Improving navigation cues with embedded visualizations," *IEEE Trans. Vis. Comput. Graphics*, vol. 13, no. 6, pp. 1129–1136, Nov./Dec. 2007.

[78] Y. Xiao, H. Wang, and W. Xu, "Parameter selection of gaussian kernel for one-class SVM," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 941–953, May 2015.

[79] C. Xie, W. Xu, and K. Mueller, "A visual analytics framework for the detection of anomalous call stack trees in high performance computing applications," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 215–224, Jan. 2019.

[80] W. Yang, X. Wang, J. Lu, W. Dou, and S. Liu, "Interactive steering of hierarchical clustering," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 10, pp. 3953–3967, Oct. 2021.

[81] P. N. Yianilos, "Normalized forms for two common metrics," NEC Research Institute, Princeton, NJ, USA, Tech. Rep. 91-082-9027-1, 1991.

Sebastian Ratzenböck received the BSc and MSc degrees in technical physics from the Vienna University of Technology, Austria, in 2015 and 2018, respectively. He is currently working toward the PhD degree with the Research Network Data Science, University of Vienna, Austia, since 2018. His research interests include machine learning algorithms and visual tools facilitating large-scale astronomical data analysis.

Verena Obermüller received the BSc degree in computer science from the University of Vienna, Austria, in 2021. She is currently working toward the MSc degree in quantitative genetics and genome analysis with the University of Edinburgh, U.K.

Torsten Möller (Senior Member, IEEE) received the Vordiplom (BSc) degree in mathematical computer science from the Humboldt University of Berlin, Germany, and the PhD degree in computer and information science from Ohio State University, in 1999. He is a professor of computer science with the University of Vienna, Austria, since 2013. Between 1999 and 2012 he served as a computing science faculty member with Simon Fraser University, Canada. He is a senior member of ACM, and a member of Eurographics. His research interests include algorithms and tools for analyzing and displaying data with principles rooted in computer graphics, human-computer interaction, signal processing, data science, and visualization.

João Alves is a full professor of stellar astrophysics with the University of Vienna, Austria, where he researches on the origins of stars and planets. Before moving to Vienna, in 2010, he was the director of the Max-Planck / CSIC Calar Alto Observatory in Southern Spain, and before that, a fellow and staff with the European Southern Observatory (ESO) in Munich. He is now using ESA's Gaia satellite and the large near-infrared survey VISIONS using ESO's survey telescope in the Atacama desert in Chile, to reconstruct the 3D motion of the interstellar gas of the Milky Way. He is the Astronomy and Astrophysics Letters editor-in-chief.

Immanuel M. Bomze holds a chair (full professor) of applied mathematics and statistics with the University of Vienna where he co-founded the Vienna Center of Operations Research (OR) and serves as its co-director. He also served 2018 – 2020 as the President of EURO (Association of European OR Societies) and is currently the editor-in-chief of the *EURO Journal of Computational Optimization*. His research interests are in the areas of OR, nonlinear optimization and data science.