Toward Addressing Ambiguous Interactions and Inferring User Intent with Dimension Reduction and Clustering Combinations in Visual Analytics

2

JOHN WENSKOVITCH, MICHELLE DOWLING, and CHRIS NORTH, Virginia Tech

Direct manipulation interactions on projections are often incorporated in visual analytics applications. These interactions enable analysts to provide incremental feedback to the system in a semi-supervised manner, demonstrating relationships that the analyst wishes to find within the data. However, determining the precise intent of the analyst is a challenge. When an analyst interacts with a projection, the inherent ambiguity of interactions can lead to a variety of possible interpretations that the system can infer. Previous work has demonstrated the utility of clusters as an interaction target to address this "With Respect to What" problem in dimension-reduced projections. However, the introduction of clusters introduces interaction inference challenges as well. In this work, we discuss the interaction space for the simultaneous use of semi-supervised dimension reduction and clustering algorithms. We introduce a novel pipeline representation to disambiguate between interactions on observations and clusters, as well as which underlying model is responding to those analyst interactions. We use a prototype visual analytics tool to demonstrate the effects of these ambiguous interactions, their properties, and the insights that an analyst can glean from each.

CCS Concepts: • **Human-centered computing** → **Visualization**; **Visual analytics**; *Visualization design and evaluation methods;*

Additional Key Words and Phrases: Dimension reduction, clustering, interaction, intent, visual analytics

ACM Reference format:
John Wenskovitch, Michelle Dowling, and Chris North. 2024. Toward Addressing Ambiguous Interactions and Inferring User Intent with Dimension Reduction and Clustering Combinations in Visual Analytics. *ACM Trans. Interact. Intell. Syst.* 14, 1, Article 2 (January 2024), 35 pages.
https://doi.org/10.1145/3588565

1 INTRODUCTION

"With Respect to What" was first described as a usability issue with interactive projections by Self et al [98]. In their Andromeda system, analysts are presented with the two-dimensional output of a **Weighted Multi-Dimensional Scaling (WMDS) Dimension Reduction (DR)** computation. By performing direct manipulation interactions on the observations in the projection [92], analysts communicate desired similarity and dissimilarity relationships to the system. This triggers a

Dr. Wenskovitch is now also employed by Pacific Northwest National Laboratory.
Dr. Dowling is currently employed by Pacific Northwest National Laboratory.
Authors' address: J. Wenskovitch, M. Dowling, and C. North, Virginia Tech, 620 Drillfield Drive, Blacksburg, VA 24061; emails: {jw87, dowlingm}@vt.edu, north@cs.vt.edu.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
2160-6455/2024/01-ART2 $15.00
https://doi.org/10.1145/3588565

learning routine that attempts to create such relationships in the projection by altering weights that are associated with each dimension. The process of inferring the intent of an analyst via such direct manipulation interactions and updating the visualization in response is termed *semantic interaction* [36, 38].

The usability issue that emerges from this interaction technique revolves around interpreting analyst intent appropriately. In other words, when the analyst moves an observation to a new position, what is that movement in relation to? Is this relationship assumed or somehow explicitly communicated by the analyst? Possible interpretations for repositioning an observation in the projection include but are not limited to moving the observation away from the source, moving the observation toward a target, and moving the observation with respect to some other observation(s) within the projection. In other words, what did the analyst move, and *with respect to what?*

Such interactive projections are an increasingly popular feature in interactive visual analytics and human-centered **Machine Learning (ML)** applications [6, 8, 37, 69, 72, 79]. As a result, resolving this "With Respect to What" problem is increasingly important to accurately capture the intent of the analyst. In our previous work, we proposed a cluster membership solution to "With Respect to What," utilizing interactive clustering reassignment to communicate similarity relationships in the projection [126, 127]. This use of clustering algorithms is a sensible choice, as implicit clusters often form naturally in dimension-reduced projections that display similarity relationships. Defining these clusters explicitly also enables explicit relationship communication to the system. Indeed, DR and clustering algorithms perform similar functions: DR algorithms simplify a dataset by reducing the number of dimensions to the most important existing or synthetic features, whereas clustering algorithms simplify a dataset by reducing the number of observations through grouping [123].

However, ambiguity in the interpretation of these interactions does still exist after explicit clustering has been introduced, as described in our motivating example in Section 3. In this work, our goal is to detail the interaction space for the simultaneous use of DR and clustering algorithms, particularly in interactive systems that feature a semantic interaction learning component. Additionally, we propose a pipeline representation that can accurately reflect the nuances of these interactions, identifying features such as algorithm order, how the models handle interactions, and how data is manipulated within the system. We further demonstrate the effect of three types of visually ambiguous interactions, showing their unique effects and commenting on the effects of those interactions to the system and the understanding of the analyst. These interactions are demonstrated on a prototype tool that uses WMDS for dimensionality reduction and a weighted k-means variant for clustering. Specifically, we claim the following contributions:

(1) An analysis of the interaction space that exists when incorporating DR and clustering algorithms in the same projection interface, and a summary of the design factors that should be considered by visualization designers.
(2) A system pipeline representation to demonstrate and detail these interactions on both observations and clusters, along with an accompanying set of use cases to demonstrate the flexibility of this representation to disambiguate between a collection of similar interactions.
(3) An exploration of three classes of interactions that are ambiguous within a prototype tool, examining the effects of the interactions on both the underlying models and on the insights that an analyst can recover from them.
(4) A discussion of additional implementation factors external to the "With Respect to What" problem that should be considered by visualization designers.

This work extends our previous paper "With Respect to What" [124] from ACM IUI 2020 with additional content in Sections 5 and 6, providing additional detail to the pipeline representation and more thoroughly exploring the effects of these ambiguous interactions.

2 BACKGROUND

In this section, we provide background information relevant to our contributions. In the first subsection, we briefly survey the use of DR and clustering algorithms in visual analytics systems, both independently and in combination. Following this, we describe the "With Respect to What" problem in the context of interactive DR (and occasionally interactive clustering) visual analytics systems, demonstrating the interactions supported by such systems. We next examine other works that have surveyed direct manipulation interactions in projections, as well as other pipeline representations used in visual analytics. The section concludes with a discussion of metrics to measure the quality of DR and clustering algorithm output.

2.1 DR and Clustering in Visualization

2.1.1 Dimension Reduction. DR algorithms strive to create a low-dimensional representation of high-dimensional data that preserves high-dimensional structures such as outliers and clusters [65]. Surveys of DR algorithms can be found in the literature [41, 44, 46, 73, 132]. Although these representations can be of any number of dimensions smaller than the cardinality of the high-dimensional space, DR algorithms are most often used to reduce the dataset into a two-dimensional projection displayed as a scatterplot or node-link diagram. A number of DR techniques are prevalent in the visual analytics literature. In Andromeda and SIRIUS, WMDS is used to project a dataset into a two-dimensional representation [32]. Force-directed layout algorithms are also common [6, 37, 126], whereas other systems project data using principal component analysis [8] or **t-distributed Stochastic Neighbor Embedding (t-SNE)** [18], among many others.

2.1.2 Clustering. The goal of clustering algorithms is to group sets of observations so that observations in the same group are more similar to each other than to those in other groups. Surveys of clustering algorithms from various perspectives can also be found in the literature [19, 135]. Clustering algorithms can be classified into hierarchical and partitioning families, with the hierarchical family further split into divisive (top-down) and agglomerative (bottom-up) types [123]. The variety of methods for presenting clusters in visualization systems is nearly as broad as the variety of clustering algorithms themselves. Among others, the most common techniques are using color to denote cluster membership [2, 21, 47, 63], encoding clusters by position [20, 71], and enclosing groups of observations with distinct boundaries [71, 126]. It is also common to use dual-encoding [51] to reinforce cluster membership [18, 58].

2.1.3 DR and Clustering. The natural relationship between DR and clustering algorithms has long been recognized. Indeed, Ding and He [28] proved that principal component analysis implicitly performs clustering as well as DR; the principal components are the continuous solutions to the discrete cluster membership indicators for k-means clusters. Similarly, self-organizing maps are a DR technique that can be interpreted as a set of clusters [59].

Observing this relationship, a number of visual analytics systems include both DR and clustering algorithms. These algorithmic combinations come in a variety of visual representations, and the algorithms also process the data in a variety of ways. For example, iVisClustering [63] performs both DR and clustering on the high-dimensional data, implying that a change in the layout does not affect the clustering assignment. In contrast, both "Be the Data" [14] and Castor [126] perform clustering on the output of the DR algorithm, rendering low-dimensional clusters that are dependent on the positioning of observations in the projection. However, "Be the Data" uses color encoding to represent cluster membership, whereas Castor takes the distinct boundaries approach. Reversing this algorithmic order, Ding and Li [29] create a system in which k-means clustering is used first to generate class labels, followed by latent Dirichlet allocation DR for subspace selection [29].

2.2 "With Respect to What"

Given that the "With Respect to What" problem is defined by how the analyst interacts within the projection, it is first important to consider common types of interaction schemes. First, many of these interactions are considered **Visual to Parametric Interactions (V2PI)**, which was defined by Leman et al. [67] and explored by Hu et al [53]. At a high level, this paradigm considers interactions that are performed directly within a projection of the data. Given an interaction, parameters for the underlying projection model are *learned*, resulting in a new projection. As a result, analysts are able to remain within their cognitive zone [48], thereby enhancing analysts' efficiency in performing their analytic tasks.

Bayesian visual analytics as defined by House et al. [52] describes a probabilistic variation of V2PI. However, the deterministic variations are more commonly seen across current visual analytic implementations, including observation-level interaction [39]. Observation-level interaction specifically defines interactions on projected observations of data in which relative pairwise distances between a subset of points is defined by the analyst. From these pairwise distances, new parameters for the underlying distance metric are learned, which in turn are used to produce an updated projection of the data. Note how all of these schemes support incremental formalism [101], which enables analysts to gradually concretize their hypothesis as they investigate the data.

These interaction schemes can be applied in a wide variety of applications. Each method has implications for how the "With Respect to What" problem can or should be solved. For example, using control points within a visualization is a common method for enabling interactive and iterative refinement of the projection [6, 26, 30, 37, 55, 69, 72, 79, 99, 122]. Control points often take the form of analyst-selected and manipulated points within the projection, but these control points can also be represented as anchors on the projection boundaries as well. In either case, the analyst is manipulating a given point with respect to the entire visualization. In other words, this interaction is meant to have an effect on a global scale rather than performing local refinements. This concept is reflected in the fact that a single movement of one control point typically results in all other non-control points moving themselves in relation to the control point's new location.

Rather than using control points, some systems instead use manipulated points to describe desired pairwise distances in a projection [8, 32, 90, 97, 126]. As a result, the information that is communicated through this interaction is a desired set of distances expressed as relative pairwise distances between the moved points, which often reflect similarity/dissimilarity relationships in the data. Using these relative pairwise distances, the system *learns* a new distance metric, typically by updating the parameters of the chosen distance function. The new distance function is then applied to all projected data, not just the interacted points, to produce an updated visualization. The implied "With Respect to What" in such interactions is limited to the points the analyst interacted with; all other points are ignored until the data is reprojected.

There are still other types of interactions that address this "With Respect to What" problem. Podium [118] allows analysts to interactively alter the rank of any item in a table. Podium explicitly defines this interaction to be with respect to the other rows that changed ranks as a result of the interaction. Additionally, ReGroup [3] enables interactive cluster formations in which each additional item that is added to a cluster results in updating a list of suggested items to add to the cluster. Thus, this interaction is with respect to the existing items within the cluster. Andrienko et al. [4] take yet another approach in which cluster definitions can be interactively altered by the analyst, such as merging or splitting clusters. Such interactions are with respect to the involved clusters (i.e., the clusters that are being merged together or which cluster is being split). Other works that provide interactive steerable projections and steerable optimizations enable the direct and indirect manipulation of axes and parameters [12, 42, 57]. These examples demonstrate the variety of manners in which the "With Respect to What" problem can be addressed, indicating the vast design space present in this area.

2.3 Surveys of Projection Interactions

Interactive DR is currently a heavily researched area of visual analytics, and as such, several recent surveys have been published that review various aspects of the space. For example, Sacha et al. [91] present a structured literature review of DR, with portions of their analysis discussing the interactive, semantic interaction-driven topics that are most relevant to this work. Other surveys of the DR literature have been produced by van der Maaten et al. [114], Wismüller et al. [132], and Liu et al. [68].

Still other surveys focus on interactions that underlie the ideas of semantic interaction and interactive model manipulation. For example, Buja et al. [10] present a review of interaction techniques for high-dimensional data visualization, whereas von Landesberger et al. [117] construct an interaction taxonomy to track and analyze user interactions in visual analytics. Similarly, Brehmer and Munzner [7] present a thorough description of DR tasks, whereas Yi et al. [136] reduce relevant DR interactions to a set of low-level interactions.

2.4 Pipelines in Visual Analytics

Pipeline representations are used to visually communicate the flow of data within a visualization system from one processing component to the next. The convention of showing the generation of a visualization by dataflow to the right and interaction handling via flow to the left dates to at least the information visualization pipeline [11], also appearing in the visual analytics task process from Keim et al [56]. These pipelines present a summary representation for the process that transforms raw data into a final visualization at the highest level, making them extremely generalizable.

For more specific use cases, such as interactive visualization systems that support semantic interaction, more detailed pipeline frameworks tailored to their purpose can be created. For example, the V2PI [67] pipeline provides a statistical semi-supervised ML methodology for transforming data into a visualization and also parameterizing a user's interaction into a form that the system can interpret. As a result, a new visualization is created based on the user's interaction. The LAMP framework described by Joia et al. [55] and its extension to iLAMP [31] demonstrate similar approaches toward local affine mappings. More recently, Dowling et al. [33] proposed a bidirectional, multi-model pipeline for semantic interaction applications. In this pipeline structure, the projection is created by processing a sequence of "forward"computations, whereas the interactions are handled by processing a similar but reverse sequence of "inverse" computations. The multi-model capacity of this pipeline also provides an additional feature missing from V2PI, which focuses on transformations that involve only a single model. Norambuena et al. [74] extend this model further, including both low-level and high-level models to capture both local continuous details as well as global discrete structures, supporting narrative visualization with semantic interactions.

With respect to interactive visualizations, specifically focused on projections of document corpora, Endert [36] provide a generic pipeline representation of a model-driven system, further expanded into a multi-model system by Bradel et al [6]. Other systems such as Dis-Function [8], Piecewise Laplacian Projection [79], and the pipeline provided by Mamani et al. [69] create focused pipelines to explain their data processing and visualization creation techniques. Wang et al. [120] further present a survey of many analytical pipeline representations, displaying the broad variety of techniques in visual analytics.

In this work (see Section 5), we adopt a similar pipeline approach to Dowling et al., summarizing models and their interactions specifically in the case of DR and clustering models. In the case of a visualization system that incorporates these processes, each of these algorithms would represent a separate model in the sequence. The order of these models in the sequence can therefore change both the meaning and the behavior of the visualization.

2.5 Quality Metrics for DR

Techniques for evaluating the quality of a DR algorithm or dimensionally reduced projection focus on projection error, searching for and quantifying errors in the low-dimensional projection. With the exception of simple and sparse datasets, these errors are nearly always present due to the loss of information when, say, reducing a dataset from 50 dimensions to 2 dimensions. There are a variety of techniques used to quantify these errors.

The most common method involves distance measures, with the overall goal of ensuring that pairs of observations close to each other in the high-dimensional space appear similarly close in the low-dimensional projection (with a similar relationship for observations that are distant) [24, 25, 60, 75, 94]. Several techniques have been proposed and validated in the literature. Simple measures are often used, such as mean squared error [40] or root mean squared error [86], which measures the average squared difference between the high-dimensional distance D and low-dimensional distance δ between pairs of nodes n_i and n_j. Kruskal's stress [61] performs a similar computation, whereas Sammon mapping [94] includes an additional term to cause small distances to have a higher importance than large distances. Other distance-based techniques include approximate geodesic distances [64, 108] and computing the residual variance, examining the correlation between original and embedded distances [86]. Some metrics are also custom to particular DR approaches. These include techniques such as topographic functions [116] and topographic products [5] for self-organizing maps.

Although distance metrics are more stable with respect to small changes in the low-dimensional projection, rank-based metrics perform better with large changes or linear scaling in the projection [66], also behaving with more stability against outliers in the data [86]. One of the first rank-based metrics to appear in the literature is Spearman's rank correlation [102], which converts high-dimensional distance $D(n_i, n_j)$ and low-dimensional distance $\delta(n_i, n_j)$ into ranks $r(n_i, n_j)$ and $\rho(n_i, n_j)$, respectively. Similarly, continuity measures consider the rank of a high-dimensional observations i and j, $\hat{r}(n_i, n_j)$ according to pairwise distance between the pair of observations [115]. These rank-based metrics are often displayed with co-ranking matrices [25] or Shepard diagrams [100].

Other metrics consider group membership, focusing on the members of clusters within the low-dimensional projection to ensure that they match clusters in the high-dimensional space. This process is relatively straightforward when given labeled data [95, 121], but it is a more complex measure when matching pairs of unlabeled observations. The process of neighborhood loss [86] computes projection error using the k nearest neighbors of each observation in both the high-dimensional space and the low-dimensional embedding. Similarly, mean relative rank error uses rank to penalize very distant observations that intrude and very close observations into the k nearest neighbors of each observation [65]. Trustworthiness metrics such as the local continuity metacriterion [13] also measure the proportion of observations that appear too close together in the projection [115]. Beyond pairwise measures, other techniques use weighted shortest paths to connect many observations [89].

2.6 Quality Metrics for Clustering

Several methods exist for assessing the quality of clustering, each of which has drawbacks resembling the subjective quality of human evaluation [43]. Thus, there is no single best method to evaluate clustering, just as there is not necessarily a best clustering algorithm [82]. Techniques for computationally determining the quality of clustering are often divided into *internal* and *external* validation methods.

Internal validation methods create a scoring scale that seeks high similarity within clusters and low similarity between clusters. Such unsupervised evaluation techniques are once again based on the choice of distance metric and linkage criterion used to ultimately judge the "similarity" of a pair of clusters. One of the first internal validation methods is the Dunn index [35], which computes the ratio between the minimum inter-cluster distance and the maximum intra-cluster distance, with the aim of providing the best score to clusterings with dense, well-separated clusters. A similar approach is found in the Davies-Bouldin index [22], which also includes each observation and centroid in the ratio rather than simply computing the minimum distance between clusters and the maximum distance within a cluster. More recently, the silhouette coefficient [88] was introduced, which compares the average distance of each observation to other members of its assigned cluster and contrasts this with the average distance to observations in an external target cluster, producing a score in the range $[-1, 1]$ for each observation. The mean score of each observation then provides a measure for the quality of clustering across the entire dataset. Although these three techniques are commonly seen, several other methods have been proposed, including the gap statistic [110], the S_Dbw index [50], the PBM index [78], and the Xie-Beni index [133].

External validation methods use an external ground truth to evaluate the quality of the clustering. These supervised evaluation techniques rely on class labels often created by expert humans [82]. For example, the Rand index [85] judges the number of correct decisions made by a clustering algorithm, a ratio of the number of true positives and true negatives with all clustering assignments. The F-measure [70] improves on the Rand index by permitting false positives and false negatives to have differently weighted effects on the score through the introduction of precision and recall measures. Further variations of the Rand index ratio formulation are found in the Dice index [27] and the Fowlkes-Mallows index [45].

A related problem to evaluating cluster quality is simply determining the number of clusters that exist within a dataset. The ideal number of clusters represents a balance between permitting an amount of error in the clustering and limiting the overall number of clusters. In other words, increasing k without bound will eventually result in zero error when each observation is assigned to its own cluster, but a substantially smaller number of clusters may be found that still has a small amount of misclassification error. This tradeoff is the basis behind the elbow method [109], searching for an "elbow" in a plot of number of clusters versus percentage of variance explained. At such an inflection point, adding an additional cluster does not substantially reduce the amount of error in the clustering assignments. Information criteria such as Akaike information criterion [1], Bayesian information criterion [96], and deviance information criterion [103] can be used to further evaluate the quality of introducing another clustering subdivision. This has been implemented in the k-means extension X-means [81]. Other clustering determinations make use of information-theoretic measures such as rate distortion theory [104] and feature rescaling [23].

3 MOTIVATING EXAMPLE AND DESIGN SPACE OVERVIEW

To motivate our discussion of the DR and clustering interaction space, consider the example shown in Figure 1. In this example, an analyst is provided with a dimension-reduced projection of an animal dataset [62], positioned according to their attribute relationships with initially equal weights. The normalized attributes included in the dataset summarize the color, habitat, diet, and behaviors of each animal, all weighted equally within the projection to create this initial view. A clustering algorithm then groups the observations into discrete categories, following the "Dimension Reduction Preprocessing for Clustering" pipeline described in our previous work [123]. After viewing the projection, the analyst wishes to manipulate the projection to continue exploring the data.

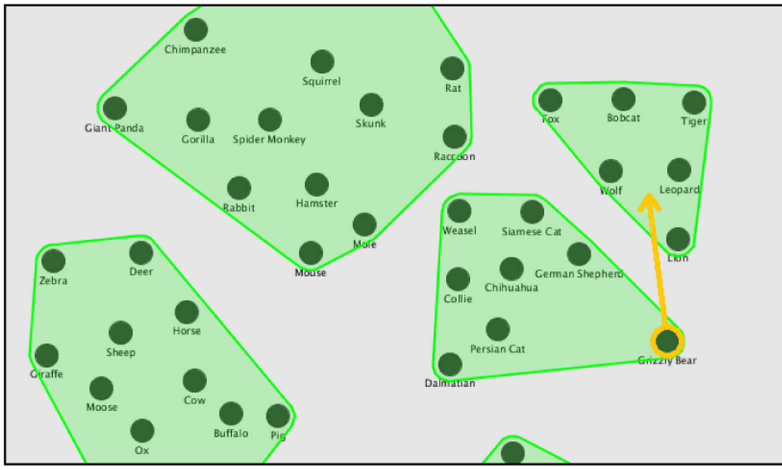


Fig. 1. An analyst repositions the Grizzly Bear observation within the projection, indicated by the orange arrow.

The tool supports steerable projections, permitting the analyst to influence those initially equally weighted dimensions to bias the projection toward a subset of the dimensions. If the analyst decides to explore the diet of these animals, the analyst may choose to reposition the Grizzly Bear observation, removing it from one cluster and placing it into another. With this simple interaction, the analyst could be trying to convey a number of possible intents to the system.

Perhaps the analyst is looking specifically at the relationships between the animals in the projection. For example, the analyst could be trying to convey a relationship about the starting position of the observation ("the Grizzly Bear is not similar to the other animals near the source") or a relationship about the ending position of the observation ("the Grizzly Bear is more similar to the other animals near the destination"). There is also the question of how many observations the analyst considers; the analyst could be trying to communicate a relationship with respect to the closest observation ("the Grizzly Bear is most similar to the Lion"), the closest *n* observations, all observations in a cluster, or all observations in the projection. These types of relationships would be best handled by the DR model.

Alternatively, the analyst may have mapped some semantic meaning onto the cluster groupings in the projection, trying to communicate a membership assignment based on those groups ("the Grizzly Bear is a better fit in the Predators cluster than in the Pets cluster"). Such relationships can incorporate both the source and the target cluster, or perhaps a case where the target is irrelevant ("the Grizzly Bear appears to be an outlier in the Pets cluster and belongs elsewhere") or the source is irrelevant ("the Grizzly Bear is a Predator"). These relationships would be best handled by the clustering algorithm.

The analyst may also be trying to communicate a relationship that includes both observations and clusters. In such cases, the relationship may be relevant to all observations within the cluster ("the Grizzly Bear is more similar to the observations in the target cluster than the source cluster"), or the precise positioning of the observation within the cluster may be important ("the Grizzly Bear belongs in the Predator cluster, but it is not similar to the small Predator (Fox)").

The examples in the preceding paragraphs suggest two primary dimensions to consider when judging the intent of the interaction. First, the interaction could be applied to the observations, the clusters, or both. Second, the interaction could be applied to a variety of cardinalities: the nearest observation, the nearest *n* observations, all observations within a cluster, or all observations in

Table 1. Collection of Example Intents That an Analyst May Wish to Communicate via Repositioning an Observation or a Cluster in a Projection of the Animals Dataset [62]

With Respect to What		**Analyst Repositions**	
		An Observation	**A Cluster**
Nearest 1	Observation	The Grizzly Bear is most similar to the Polar Bear.	The Predators cluster shares few similarities with the Blue Whale observation.
	Cluster	The Grizzly Bear is similar to many of the other members of the Predators cluster.	The Predators cluster is dissimilar from the Large Herbivores cluster.
Nearest *n*	Observations	The Grizzly Bear behaves similarly to animals such as Wolves, Leopards, and Lions.	The Predators cluster shares few similarities with the aquatic animals.
	Clusters	The Grizzly Bear is a large Predatory animal, although still similar to small Predatory animals and scavengers.	The Scavenging Predators cluster is similar to the small actively hunting and large actively hunting predators.
Cluster	Single	The Grizzly Bear belongs in the Scavenging Predators cluster.	The Predators cluster is similar to the Grizzly Bear observation.
	Multiple	The Grizzly Bear is a predator.	The Scavenging Predators cluster is a subset of the overall Predators group.
All of the	Observations	The Grizzly Bear is more similar to the predatory animals on the left than the herbivorous animals on the right.	The Scavenging Predators cluster is more similar to the other carnivorous animals on the left than the herbivorous animals on the right.
	Clusters	The Grizzly Bear is more similar to the predatory animal clusters on the left than the herbivorous animal clusters on the right.	The Scavenging Predators cluster is more similar to the carnivorous animals clusters on the left than the herbivorous animal clusters on the right.

Creating unambiguous interactions to support each of these potential intents remains an open challenge.

the projection. These dimensions are summarized with respect to the Grizzly Bear observation and Predators cluster in Table 1 and are expanded upon in the following sections. However, it is also worth considering further aspects of the interaction and of the visualization itself. We discuss several additional dimensions of this interaction space in the next section.

4 INTERACTIONS ON OBSERVATIONS AND CLUSTERS

This section presents interactions and their potential interpretations when interacting with both DR and clustering algorithms. Given the discussion that follows, we summarize the following factors that should be considered by a designer when mapping the intent of an analyst to an interaction in this space:

- *Interaction target*: An interaction can be applied to the observations, the clusters, or both.
- *Cardinality*: An interaction can be applied to a variety of cardinalities: the nearest observation or *n* observations, all observations within a cluster, or all observations in the projection.
- *With Respect to What*: Is the relationship relative to a target at the source or destination location, or both?
- *Level of thinking*: When performing an interaction, is the analyst is thinking high or low dimensionally? In other words, is the analyst merely altering the coordinates of the observation, or are they considering multiple latent properties of a group of observations?
- *Visual design*: Is the intent of the interaction influenced by the way that observations and clusters are encoded in the visualization?
- *Algorithm order*: Is the DR or the clustering processed first? Or are they simultaneous?

In this section, we first consider the possible interpretations that result when an analyst repositions an observation in the projection. We begin by describing interactions that affect observations, before moving into interactions that affect clusters, and then turn to interactions that affect both. Subsequently, we consider the possible interpretations that result when an analyst repositions a cluster in the projection. This discussion begins by describing cluster movement interactions that communicate a similarity relationship to other observations or clusters, followed by a discussion for interactions unique to the cluster-to-cluster relationship. In some cases, we cite systems that demonstrate interaction properties that we present. In other cases, no such system currently

Table 2. Summary of Interactions by Cardinality, the Importance of the Interaction Source (S), Target (T), or Both (B), and Whether an Analyst Is Typically Thinking High Dimensionally (HD), Low Dimensionally (LD), or Both (B)

	Cardinality	Source/Target	High-D/Low-D
Observation–Observation Similarity			
Move observation toward another observation	1:1	T	LD
Move observation away from another observation	1:1	S	LD
Move observation toward several observations	1:n	T	LD
Move observation away from several observations	1:n	S	LD
Observation–Cluster Similarity			
Move observation toward a cluster	1:1	T	B
Move observation away from a cluster	1:1	S	B
Move observation toward several clusters	1:n	T	B
Move observation away from several clusters	1:n	S	B
Cluster–Observation Similarity			
Move cluster toward an observation	1:1	T	B
Move cluster away from an observation	1:1	S	B
Move cluster toward several observations	1:n	T	B
Move cluster away from several observations	1:n	S	B
Cluster–Cluster Similarity			
Move cluster toward another cluster	1:1	T	B
Move cluster away from another cluster	1:1	S	B
Move cluster toward several clusters	1:n	T	B
Move cluster away from several clusters	1:n	S	B
Observation Change in Membership			
Move observation into cluster	1:1	T	HD
Move observation out of cluster	1:1	S	HD
Move observation between clusters	1:n	B	HD
Move observation external to clusters	1:n	B	HD
Move observation within a cluster	1:1	B	HD
Cluster Change in Membership			
Move cluster into cluster	1:1	T	HD
Move cluster out of cluster	1:1	S	HD
Move cluster between clusters	1:n	B	HD
Move cluster external to clusters	1:n	B	HD
Move cluster within a cluster	1:1	B	HD
Join/Split Clusters			
Join clusters	n:1	T	HD
Split cluster	1:n	T	HD
Create/Remove Clusters			
Create cluster	1	T	HD
Remove cluster	1	S	HD

exists, and so our presented interaction challenges are more speculative. We summarize some of the properties of these interactions in Table 2.

4.1 Observation Interactions with Respect to Observations

As detailed by the left column of the intents in Table 1 and summarized in the previous section, when an analyst repositions an observation, the system must determine what the analyst is moving the observation with respect to. The analyst might be repositioning the observation to move it away from something near the source, toward something near the target, or relative to any other observation in the projection. The analyst might also be repositioning the observation relative to the position of just a single observation or a collection of n observations.

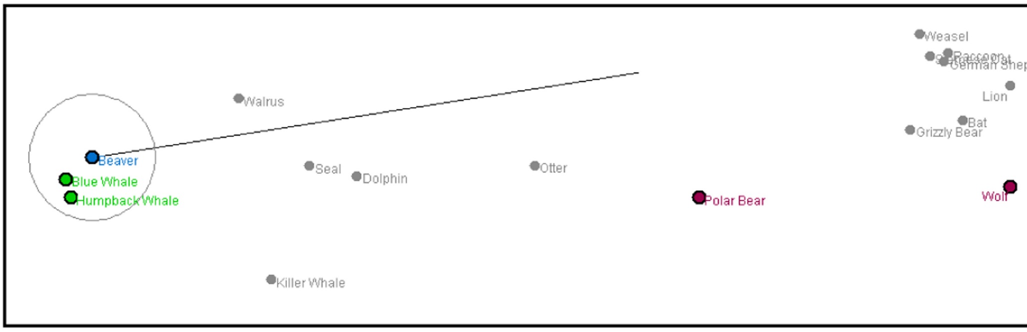


Fig. 2. Selection interactions in Andromeda [97]: nearest neighbor selection at the source (Polar Bear), radius selection at the target (Blue and Humpback Whales), and additional observation selection in other regions (Wolf).

Determining the other component(s) involved in the interaction is an additional challenge, particularly when differentiating between movements with respect to 1, n, or all observations. The most straightforward solution to this challenge is to provide analysts with one or more selection mechanisms. For example, Andromeda [97] implements several methods to permit the analyst to choose all elements necessary for the interaction. At the source of the interaction, the nearest neighbor is selected by default. At the target, all observations within a set radius of the target position of the interaction are selected. Following these default selections, the analyst is permitted to select or deselect any observation in the projection. An example of each of this selection mechanism is shown in Figure 2, in which an analyst has repositioned the Beaver observation closer to the whales, possibly signifying an interest in exploring aquatic-dwelling animal behavior. As the nearest neighbor to the source, the Polar Bear was automatically selected as part of the interaction, as were the Blue Whale and Humpback Whale in the target radius. After this observation was repositioned, the Wolf was also selected to denote dissimiliarity between land-dwelling and water-dwelling animals.

The further possibility exists that the analyst does not wish to alter any underlying models with the interaction they provide. Instead, they may be merely exploring the current projection. Endert et al. [39] define these categories of exploration as exploratory and expressive: *exploratory* interactions provide an analyst with insight into the structure of the data, whereas *expressive* interactions communicate an intent to the system and effect underlying models. For example, Castor treats interactions that do not cross cluster boundaries as exploratory, allowing analysts to investigate relationships between observations without affecting the underlying learning system [126]. This is generally true for drag interactions in other systems that incorporate force-directed layouts, such as ForceSPIRE [37] and StarSPIRE [6].

That said, StarSPIRE allows for explicit interactions by having the analyst overlap the boundaries of two documents. In this case, the system interprets this interaction as the analyst expressing not just document similarity, but their immediate, close relatedness. Thus, StarSPIRE uses this interaction to increase the weight associated with all shares entities between the two documents, resulting in an updated projection that includes new documents discovered through semantic interaction foraging [6].

4.2 Observation Interactions with Respect to Clusters

Repositioning an observation with respect to a cluster leads to a further set of challenges, primarily centered around the means by which cluster information is encoded in the projection. This is due

Table 3. Collection of Example Interactions and Intents That an Analyst Could Communicate via Reclassifying an Observation with Respect to a Cluster in a Projection That Uses Cluster Boundaries

Moving an Observation	**Intent Expressed by the Analyst**
Into a cluster	The Grizzly Bear is a predator.
Out of a cluster	A Grizzly Bear is not a pet.
Between clusters	The Grizzly Bear is better classified as a hunting predator than a scavenging predator.
External to all clusters	The Grizzly Bear is more like the large cats than the wolves, although it belongs to neither group.
Internal to a cluster	The Grizzly Bear is a predator, and it is more like the large predators than the small predators.

to the fact that the visual encoding of clusters leads to different affordances for interaction. An important consideration when introducing both DR and clustering interactions in the same interface is determining the order of these algorithms. If the DR algorithm runs first, then the clustering algorithm is computed on the low-dimensional data. In contrast, if the clustering algorithm runs first, a layout needs to be constructed to appropriately project these clusters into the two-dimensional view necessary for display. It is also possible to perform both computations in the high-dimensional space at the expense of computational efficiency. Our previous work [123] discusses tradeoffs with respect to algorithm order for such systems in more detail. Here, we first consider clusters defined by an explicit border, as in the motivating example from Section 3 and Figure 1. After this discussion, we summarize these interactions with respect to color and cluster hierarchies.

4.2.1 Boundaries. Explicit cluster boundaries in a projection suggest that repositioning an observation into or out of a cluster is communicating a membership assignment to the system, clearly crossing a demarcation line between the content within the cluster and external to the cluster [112]. Such an interaction then can be interpreted in a variety of ways: an observation is being repositioned into a cluster, out of a cluster, between clusters, separate from all clusters, or internal to a cluster. Regardless of the interaction performed, the observation is being repositioned with respect to some cluster at the source or target of the interaction. As a result, a distance between the repositioned observation and the source and/or target clusters is necessary to model the high-dimensional relationship between these entities. A collection of example observation reclassification interactions and their related intents are included in Table 3. Castor presents an example of an explicit cluster boundary system, treating any observation reposition that crosses a cluster boundary as an expressive interaction [126].

4.2.2 Color. If clusters are encoded by a mechanism other than boundaries, such as color, then the natural interactions afforded by the system will change. Color is often used to demonstrate cluster assignments in systems where items belonging to different clusters may be positioned nearby in a projection [2, 18, 58]. In other words, explicit cluster boundaries are more easily interpreted when cluster regions can be easily and accurately expressed by separate, non-overlapping regions. Simply repositioning an observation into a multi-colored grouping of observations will not always be sufficient to communicate a new cluster assignment. Instead, an alternative cluster reassignment mechanism would be preferred. For example, clicking on an observation can cycle through its possible colors and therefore its cluster assignment.

4.2.3 Cluster Hierarchies. If clustering is hierarchical, then the metric learning process becomes more complex. Such a system must evaluate the intended position in the overall hierarchy at which an observation began and where it ended. To use the Animals dataset example, the Grizzly Bear could belong to a Large Predators cluster, a subset of the overall Predators cluster which in turn is a subset of a Carnivores cluster. If the Grizzly Bear is moved from its current position, then in addition to determining if the source of the interaction is important, the system must also determine which level of the hierarchy is the relevant part of the source. Therefore, the learning relationships may also depend upon not only the source and target cluster of the interaction but also the parent clusters and their properties at each endpoint. A recursive computation of cluster properties and weights may be necessary to consider the full hierarchical structure.

4.3 Observation Interactions with Respect to Both Clusters and Observations

In addition to the prior examples, an analyst may wish to communicate both position and membership information simultaneously via an interaction on an observation. Again, consider the interaction in the motivating example from Section 3. The analyst may wish to communicate that the Grizzly Bear belongs in the Predators cluster while also communicating that the Grizzly Bear is more similar to the large predators in the cluster (Lion, Tiger) than it is to the small predators (Fox, Bobcat). In such a case, factors from both of the previous two subsections must be considered.

4.4 Cluster Repositioning Interactions

Much like repositioning an observation with respect to another component of the visualization, relocating a cluster to a new position may need to consider the source and target positions of the cluster, as well as the positional relationships to other observations and clusters. The right column of Table 1 summarizes some cluster intents with respect to both observations and clusters.

However, a significant difference between observations and clusters is the space used by each interaction target in the projection—observations require a single point, whereas clusters require a broader space. As a result, a visualization designer should consider how to compute the location and value of a cluster in these interactions. Such computations could consider a simple centroid of the cluster, or potentially a weighted centroid based on the position of each observation within the cluster. Further, distances between a cluster and a second component of the visualization could be computed in a variety of ways, including single linkage, average linkage, and complete linkage [106]. Given that a cluster has a complex value that could be determined in a variety of ways, determining how to update underlying weights based upon the result of an interaction is also a complex decision.

There is further ambiguity with respect to drag interactions on clusters, particularly in the case where cluster membership is encoded by boundaries. Consider the case in which one cluster is dragged into another—is the analyst intending to join the clusters, or to express a hierarchical relationship between those clusters? It is also possible to use a drag interaction to reposition the boundary of a cluster without relocating the observations that it encloses. Such an interaction can be used by an analyst to correct for misclassifications, encapsulating additional observations within the cluster by performing an interaction to shift the boundary. This interaction should be interpreted by the system with a very different meaning, as the analyst is again performing an expressive interaction to reclassify observations.

4.5 Cluster-Specific Interactions

In addition to the repositioning relationships that can be implemented for clusters, a further set of cluster-specific interactions are possible to implement through ambiguous operations upon a projection. For example, consider the interaction in which an analyst drags one cluster toward another until their boundaries slightly overlap. One potential interpretation for this interaction is that the analyst is again providing a similarity relationship, indicating that these clusters are similar. However, the analyst could be intending that the clusters be merged into a single, larger cluster.

Such ambiguity within a single interaction computation and across interaction computations is not limited to joining clusters. For example, consider a sequence of interactions in which an analyst drags some nodes to the left side of a cluster and others to the right. In the Castor approach [126], such an interaction is interpreted as exploratory, and as such is not handled by either interaction computation. However, the analyst could also be indicating an intent to split this cluster into two smaller clusters. In this case, both clustering and DR interaction computations are necessary: the clustering to create the new clusters and update assignments, and the DR to examine the dissimilarities between the groups that the analyst formed.

Further, an analyst could also attempt to create a new cluster by repositioning a set of observations into a single region of the projection. Again, both clustering and DR computations are necessary to judge this intention, as the analyst is creating a new cluster while also communicating similarity relationships among the collection of observations that are grouped together. Other interactions such as removing clusters, growing or shrinking the size of a cluster, and increasing or decreasing the importance of a cluster can be implemented through ambiguous interactions that must be interpreted to understand analyst intent.

5 AN INTERACTION-BASED PIPELINE REPRESENTATION

To represent and disambiguate the nuances of these interactions, we propose a pipeline representation that extends the work by Dowling et al. [33]. In this pipeline representation, computations to create visualizations and respond to interactions are encapsulated within independent models. Each of these models includes an algorithmic component for each of those computations. Data flows clockwise through the pipeline representation, initially creating a visualization by transforming the initial data through the sequence of computations to produce a visualization (top of the pipeline, left to right). When an analyst performs an interaction, the computational flow continues clockwise through a sequence of computations (bottom of the pipeline, right to left) that may or may not execute depending on the interaction. These computations provide some update to an underlying weight vector, which is then converted into a new visualization in response to the interaction through a follow-up execution of the visualization creation direction.

The goal of this pipeline representation is to represent the plethora of ways that ambiguous interactions may be interpreted and responded to by a system, while simultaneously displaying the data transformations involved in presenting the grouped and reduced data to an analyst. Representing these interactions using a pipeline representation is useful for system developers who implement support for these interactions and for designers who must convey affordances and visual guidance to express the availability and effect of these interactions. This pipeline also serves the purpose of communicating to analyst how a system will interpret one or more of their actions.

From this pipeline representation, we incorporate the following modifications for clarity of communicating an interaction:

- *Nomenclature*: We update the "forward" computations to instead be "projection" computations, and we update the "inverse" computations to instead be "interaction" computations, clarifying the purpose of each computation.
- *Edges*: We annotate the edges that connect the data, models, and visualizations to demonstrate the inputs and outputs that are updated by each model.
- *Model computations*: If a model projection or interaction computation does not need to execute, we gray out the computation to make it clear that the computation is not relevant to the interaction in question.

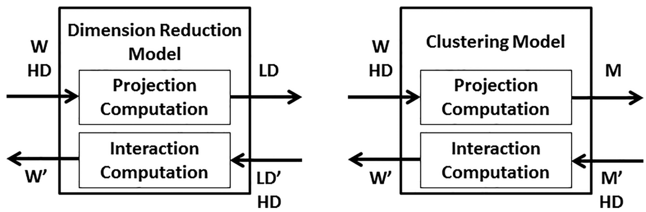


Fig. 3. A representation of the DR and clustering models described in this section, here shown as independent processes before they are combined into the pipelines throughout this section. In this figure and the subsequent pipeline figures, W = dimension weights, HD = high-dimensional data, LD = low-dimensional coordinates, and M = cluster membership.

- *Clusters*: As we are considering updates to both observations and clusters, a mechanism must exist in the pipeline representation to disambiguate between these possible inputs and outputs (here, we simply add a glyph to the model).

5.1 Models

As detailed by the left column of the intents and interactions in Table 1 and described in the previous section, when an analyst repositions an observation, the system must determine what the analyst is moving the observation with respect to. The analyst might be repositioning the observation to move it away from something near the source, toward something near the target, or relative to any other observation in the projection. The analyst might also be repositioning the observation relative to the position of just a single observation or a collection of *n* observations.

As the DR model is responsible for the positioning (and repositioning) of observations, it is most natural to use the DR model to interpret the intent of the analyst in these cases. The projection computation of the DR model (Figure 3(a)) takes dimension weights (*weights*) and high-dimensional data (*hd_data*) as input and generates low-dimensional data (*ld_positions*) as output. In contrast, the interaction computation of this model considers relative positions of observations that the user has repositioned in the low-dimensional space (*ld_positions′*), and will generate new weights (*weights′*) that will cause such relationships to appear in a subsequent projection when applied to the immutable high-dimensional data. Such an approach is used by Andromeda [97, 98]. These relationships can be expressed as follows:

$$\begin{aligned} ld\_positions &= DR\_PROJECT\ (\ weights,\ hd\_data\ ) \\ weights' &= DR\_INTERACT\ (\ hd\_data,\ ld\_positions'\ ). \end{aligned}$$

Repositioning an observation with respect to a cluster leads to a further set of challenges, primarily centered around the means by which cluster information is encoded in the projection. This is due to the fact that the visual encoding of clusters leads to different affordances for interaction. For example, explicit cluster boundaries in a projection suggest that repositioning an observation into or out of a cluster is communicating a membership assignment to the system. An interaction then could be interpreted in a variety of ways such as those listed in the example interactions and intents for reclassifying an observation with respect to a cluster in Table 3.

Using the visual encoding of the clusters in this manner means that a clustering model (Figure 3(b)) is now necessary to interpret these interactions. Clusters in the data are identified and visualized with the projection computation of the model, once again using a set of weights (*weights*) and the immutable high-dimensional data (*hd_data*) to assign observations to clusters (*memberships*). As with the DR model, the purpose of the interaction computation in the clustering model is to learn a set of weights (*weights′*) that reflect the intent of an analyst when they perform a reassignment interaction, moving observations between clusters (*memberships'*). This set of weight can then be applied to the high-dimensional data to influence the clustering, resulting in an updated cluster membership assignment for each observation as the projection computation runs again. These concepts are represented by the following equations:

$$memberships = CLUSTER\_PROJECT\ (\ weights,\ hd\_data\ )$$
$$weights' = CLUSTER\_INTERACT\ (\ hd\_data,\ memberships'\ ).$$

5.2 Model Composition for Interactions with Observations

The models' relative locations in the system pipeline are relevant to both the projection and the interaction. As discussed in our previous work [123], running the DR model before the clustering model in the projection direction implies showing the clusters that exist in the low-dimensional space. In other words, rather than using the high-dimensional data as input to the clustering model projection, we instead use the current low-dimensional positions (*ld_positions*). If the projection displays low-dimensional clusters, then a cluster reassignment can be interpreted as informing the system that the high-dimensional interpretation of groups in the data does not match the low-dimensional classification. In such a pipeline, the analyst is reasoning in high-dimensional space, and thus the high-dimensional data should be considered when interpreting the interaction. The combined projection direction then becomes the following:

$$ld\_positions = DR\_PROJECT\ (\ weights,\ hd\_data\ )$$
$$memberships = CLUSTER\_PROJECT\ (\ weights,\ ld\_positions\ ).$$

Still, a distance between the repositioned observation and each of the source and target clusters is necessary to model the high-dimensional relationship between these entities. The precise inputs and outputs that are supported by the interaction computations are dependent upon the intent that should be inferred from the interactions. In the case where crossing a cluster boundary should be interpreted as a reassignment and the position of an observation within a cluster is relevant, both interaction computations are necessary in the pipeline. The weight vector is thus transformed by both models:

$$weights' = CLUSTER\_INTERACT\ (\ hd\_data,\ memberships'\ )$$
$$weights'' = DR\_INTERACT\ (\ hd\_data,\ ld\_positions'\ ).$$

A pipeline that supports this relationship is provided in Figure 4. This pipeline has both an overarching projection direction to generate the desired visualization and an overarching interaction direction to interpret analyst intent, with each of these computation directions spanning the DR and clustering models.

In contrast, a pipeline that follows the "Clustering Preprocessing for Dimension Reduction" pattern [123] implies a layout of cluster centroids of any dimensionality. In such a projection, repositioning an observation from one cluster to another implies that the high-dimensional classification of the observation does not meet the expectation of the analyst during their current exploration. Perhaps in this case only the interaction computation of the clustering model is needed to resolve the change in cluster membership, and no consideration is given to the position of the observation within the target cluster. In this case, the interaction computation of the DR model is not required. Such pipelines are provided in Figure 5, with the unneeded computation grayed out in the figure and removed from the following equation:

$$weights' = CLUSTER\_INTERACT\ (\ hd\_data,\ memberships'\ ).$$

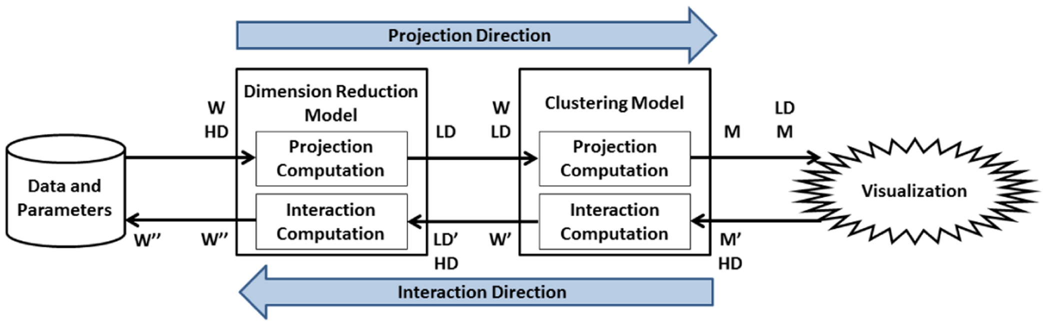
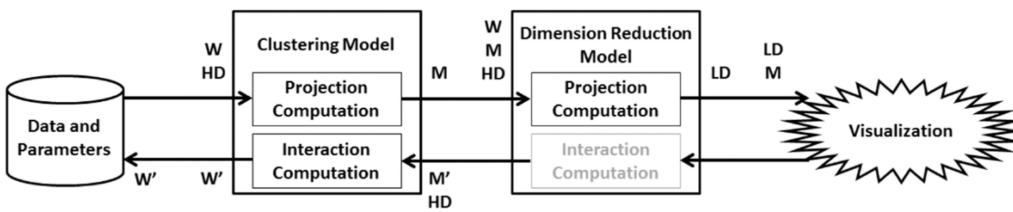
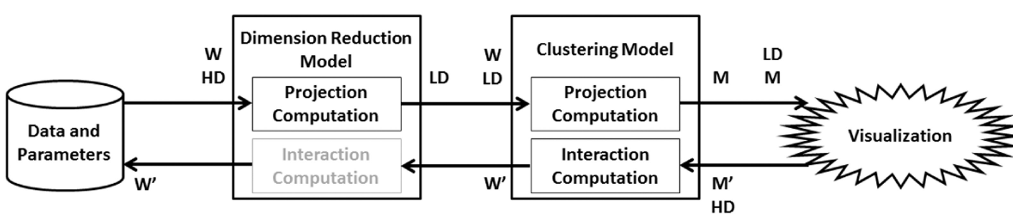


Fig. 4. A representation of dataflow using a clustering model to interpret a change in cluster membership, followed by learning distances with a DR model, to learn a new set of weights and hence a new projection. All other pipeline figures also have these overarching projection and interaction directions. We do not include them on the other pipeline figures because the concept remains the same; we only highlight the idea visually here.



(a) A representation of using a clustering model alone to interpret a change in cluster membership in a pipeline that begins the projection direction with the clustering model



(b) An alternative representation of dataflow using a clustering model to interpret a change in cluster membership to learn a new projection

Fig. 5. Even when the projection direction of two pipelines begins with two different models, the interactions that are supported by the pipeline may be the same.

If clusters are encoded by a mechanism other than boundaries, such as color, then the natural interactions afforded by the system will change. For example, if clicking on an observation will cycle through its possible colors and therefore its cluster assignment, the clustering model is solely responsible for learning from the interaction. The position of the DR algorithm is therefore dependent upon the method of projection, either identical to that shown in Figure 5(a) if the clustering model is first to execute, or as shown in Figure 5(b) if the DR model is first to execute. However, it is also possible to take the pipeline from Figure 5(a), maintain the order of the models in the projection direction, but exchange the interactions so that the pipeline supports layout manipulation interactions rather than cluster membership reassignment operations. In this case, the DR model

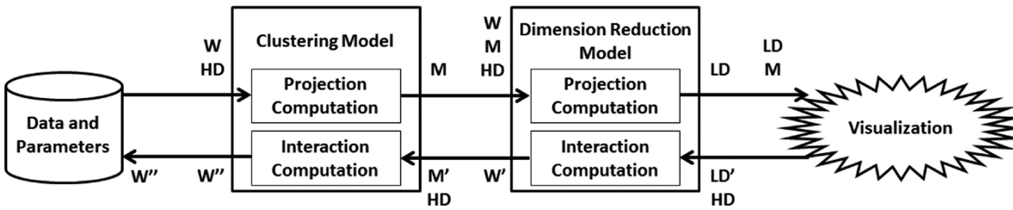


Fig. 6. A representation of dataflow using a clustering model first to interpret a change in cluster membership to learn a new projection.

handles the interaction, not the clustering model:

$$weights' = DR\_INTERACT\ (\ hd\_data,\ ld\_positions'\ ).$$

Consider the case that an analyst is communicating both position and membership information simultaneously via an interaction, such as the interaction in the motivating example from Section 3 in which the analyst may wish to communicate that the Grizzly Bear belongs in the Predators cluster, while simultaneously communicating that the Grizzly Bear is more similar to the large predators in the cluster than it is to the small predators. In such a case, both the DR and clustering models are required to interpret the interaction. The precise pipeline to handle such an interaction depends upon the projection meaning. If the DR output is used to inform low-dimension clustering assignments, then the overall pipeline remains similar to that in Figure 4. Alternatively, if the clustering output is used to position cluster centroids via DR, then a similar pipeline can be utilized but with the model order swapped, as in Figure 6.

5.3 Model Composition for Interactions with Clusters

Much like repositioning an observation with respect to another component of the visualization, repositioning a cluster to a new position is most naturally handled by the DR model. The right column of Table 1 summarizes some cluster intents and interactions with respect to both observations and clusters. In general, the DR model is most suited to measuring and responding to interactions that cause distance changes between clusters and other visualization components.

In the case where cluster membership is encoded by boundaries, a drag interaction remains ambiguous. Such an interaction could be implemented so that all observations contained within the cluster are also repositioned with the cluster. For example, consider the interaction in which an analyst drags one cluster toward another until their boundaries overlap. One potential interpretation for this interaction is that the analyst is again providing a similarity relationship, indicating that these clusters are similar. Because the analyst is communicating a similarity relationship, this interaction would be handled by the DR model, with the interaction computation of the clustering model skipped since no clustering reassignments have occurred. Such an example is provided in Figure 7. To demonstrate that an interaction computation is processing a cluster rather than an observation, we use the ⚄ glyph:

$$\begin{aligned} ld\_positions &= DR\_PROJECT\ (\ weights,\ hd\_data\ ) \\ memberships &= CLUSTER\_PROJECT\ (\ weights,\ ld\_positions\ ) \\ weights' &= DR\_INTERACT\ (\ hd\_data,\ ld\_positions\text{⚄}'\ ). \end{aligned}$$

However, it is also possible to use a drag interaction to reposition the boundary of a cluster without relocating the observations that it encloses. Such an interaction could be used by an analyst to correct for misclassifications, encapsulating additional observations within the cluster by shifting the boundary. This interaction should be interpreted by the system with a very different meaning,

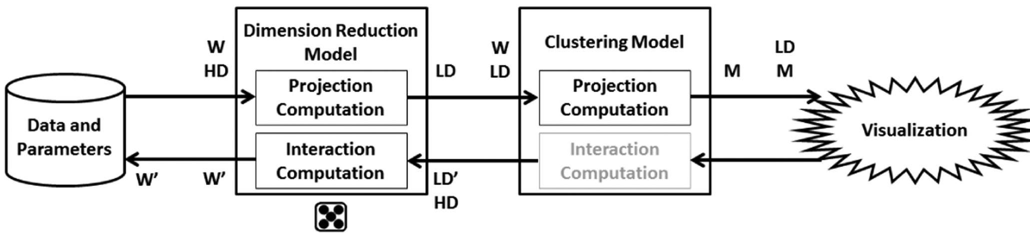


Fig. 7. A representation of dataflow for an interaction that repositions a cluster to another location in the projection, only requiring the interaction computation of the DR model.

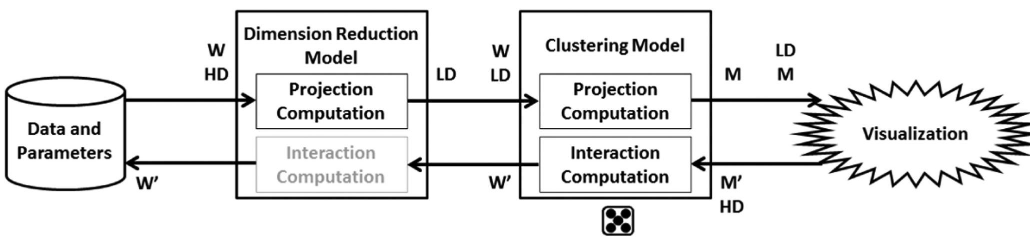


Fig. 8. A representation of dataflow for an interaction that repositions cluster boundaries to encapsulate new observations, only requiring the interaction computation of the clustering model.

as the analyst is again performing an expressive interaction to reclassify observations. The cluster membership reclassifications are then handled by the clustering model, as shown in Figure 8.

$$
\begin{aligned}
ld\_positions &= DR\_PROJECT\ (\ weights,\ hd\_data\ )\\
memberships &= CLUSTER\_PROJECT\ (\ weights,\ ld\_positions\ )\\
weights' &= CLUSTER\_INTERACT\ (\ hd\_data,\ memberships'\ )
\end{aligned}
$$

However, the analyst could be intending that the clusters be merged into a single, larger cluster. Alternatively, if the analyst drags one cluster fully into another, they may be demonstrating an intended hierarchical relationship between these two clusters. Both of these potential interactions are naturally handled by the clustering model, necessitating that the interaction computation of the clustering model knows how to determine which of these analyst intents is best matched by the interaction. Indeed, the interaction computations of the DR and clustering models require some internal communication and negotiation to jointly determine the intent of the analyst, which could be conveyed with additional edges in the interaction direction of the pipeline.

Resolving ambiguity in cluster interactions is complex, as all of the analyst intent is interpreted by the clustering model's interaction computation. As such, providing additional visual feedback to communicate the interpreted intent can help to resolve any resulting issues from these interactions. We further explore this difficulty with ambiguous interactions in the next section.

6 OPERATIONALIZING PIPELINES AND INTERACTIONS

Combining the pipelines from the previous section along with the interaction properties discussed in Section 4 yields a methodology for displaying the structure of a system, the methods by which data are processed by interactions, the effect that these interactions have on the underlying models, and on the insights that an analyst could glean from performing those interactions. This section begins by revisiting the motivational example from Section 3 to explore the effects of three ambiguous intents that underlie that interaction within a prototype visual analytics tool:

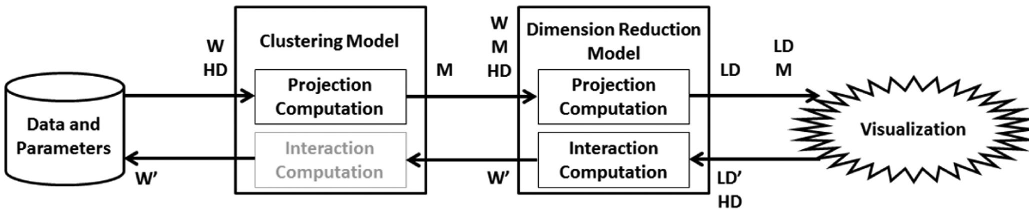


Fig. 9. A representation of using a DR model alone to interpret a change in observation position in a pipeline that begins the projection direction with the clustering model.

(1) *Only the clustering model is relevant*: "The Grizzly Bear is a Predator." Figure 5(a) reflects this interaction.
(2) *Only the DR model is relevant*: "The Grizzly Bear is similar to the Lion." Figure 9 reflects this interaction.
(3) *Both models are used*: "The Grizzly Bear is a Predator and is more similar to the Lion than it is to the Fox." Figure 6 reflects this interaction.

Using our categories from Section 4, we can then summarize the properties of these interactions:

- *Interaction target*: The interaction is performed on an observation, the Grizzly Bear.
- *Cardinality*: This property of the interaction varies. Interaction #1 is performed considering one cluster, Interaction #2 is performed considering only one other observation, and Interaction #3 is performed considering a cluster and two other observations.
- *With Respect to What*: In all three interactions, the source cluster and observations are irrelevant. Only the target cluster and/or observations noted previously are considered.
- *Level of thinking*: The analyst is considering many dimensions that constitute their intent in their interaction rather than focusing on projection coordinates.
- *Visual design*: Clear cluster boundaries need to be crossed to perform each of the interactions. The remainder of the interaction is left visually ambiguous for this exercise.
- *Algorithm order*: The tool that is used to demonstrate these interactions performs clustering prior to the DR.

In the next subsection, we provide a brief overview of the Pollux tool that is used for this demonstration. Following that, we explore both the model updates that result and the insights gleaned from each interaction above in the Animals dataset [62], followed by a similar analysis on a U.S. State Census dataset [113]. We recorded measures before and after the interaction of the projection quality using Kruskal's stress [61] and of the clustering quality using the Dunn index [35]. As both are distance-based measures, the logged values provide a measure of projection quality (Kruskal) and a measure of cluster quality (Dunn) that are both affected by the weight vector update that occurs in response to the interaction. These values further demonstrate the effects of the weight vector update that may be more subtle in the visualization, particularly with respect to the projection layout (with convex hull boundaries, changes to the cluster membership assignments are more obvious). We also note insights that an analyst could gain about the data from the new projection that resulted from performing the interaction.

6.1 Pollux: Clustering First

Pollux (the original version is described in more detail in previous work [127]) is a prototype tool that was created to explore the design space for combining DR and clustering algorithms when the clustering model is the first to execute in the projection direction. This relatively minor change from a DR-first pipeline leads to substantial differences in the way that data is processed and visualized, resulting in a multi-level layout (i.e., a set of projections within a projection) to increase the efficiency of the layout computation while also getting an accurate collection of clusters. In the projection direction, the current dimension weights and the high-dimensional data are first processed by the clustering model using a weighted k-means algorithm to produce a set of cluster membership assignments. These groups of observations are then communicated to the projection computation of the DR model to be placed in the projection. To do so, we first project the high-dimensional coordinates of the cluster centroids into the projection, followed by projecting the individual observations into groups around their respective centroids.

Computationally, the Pollux tool operates by passing a data object between modularized implementations of clustering and dimensionality reduction algorithms, as well as a visualization frontend. The data exchange follows the approach indicated by the pipeline figures from the previous section, in which the projection computations use a set of weights stored in the data object to either cluster or project the raw high-dimensional data, whereas the interaction computations update those weights as needed in response to user interactions in the visualization. We chose to use a weighted k-means variant as a clustering algorithm and WMDS as a dimensionality reduction algorithm because of those interaction computations; our previous work has established mathematical inversions of these projection and clustering algorithms to appropriately respond to user interactions through those weight updates [32, 34, 97, 126]. Other clustering algorithms (e.g., DBSCAN, Gaussian mixture modeling with expectation-maximization) and dimensionality reduction algorithms (e.g., UMAP, t-SNE) could be substituted into the process if paired with an interaction computation.

With Pollux, there are five different types of edges that can be considered, each of which can have a different strength applied to them to deform the projection. In the case of the implementation used here, edges that connect the cluster centroids are given a strength of 1.0, giving them the greatest amount of influence in the overall layout. Edges that connect a node to its cluster centroid are given a strength of 0.7, and edges that connect a pair of nodes that both belong to the same cluster are given a strength of 0.5. This creates the effect of compact, non-overlapping clusters. Edges that connect a node to other cluster centroids and edges that connect a node to nodes in other clusters are not included in the instance of Pollux used in this demonstration. There are arguments both for and against including such edges in the graph. One benefit of including these edges is if, for example, Cluster B is positioned to the right of Cluster A, and a node in Cluster A also shares many properties of other nodes in Cluster B, then that node will be drawn to the right side of Cluster A. This further glimpse of similarity in the projection provides additional information to an analyst, but with a drawback of the additional computational complexity required to handle the influence of these edges on the graph. A primary argument for using this cluster-first pipeline in the first place is to use the hierarchical structure to generate a more efficient projection that does not require a completely connected graph.

In the original version of Pollux, only interactions in which an observation was moved into a new cluster were considered relevant to updating the underlying weight vectors and models. We kept this implementation for the cluster-only interactions and added the learning module for Andromeda to handle the DR-only interactions (see Section 3.2 of Self et al. [97] for algorithmic details). These modules were executed in sequence for the interactions that used both algorithms. For this demonstration, we maintain the visual ambiguity to demonstrate the potentially unexpected effects that could occur without visual guidance and feedback, but we do implement interaction responses that address the intent of the interactions noted earlier. After the necessary interaction computations execute in the pipeline, the model updates are applied to all data in the projection. As a result, the knowledge communicated by the interaction and learned by the models could result in reclassifying an untouched observation into a new cluster, structurally rearranging the layout

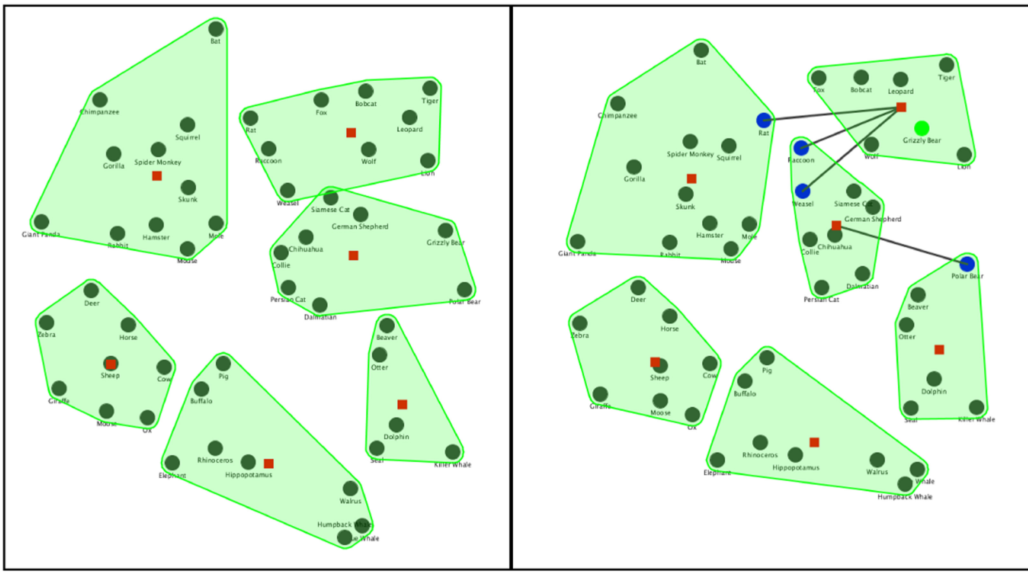


Fig. 10. The intent "The Grizzly Bear is a Predator" only uses the interaction computation of the clustering model. The associated interaction produced several cluster reassignments relevant to the Predator cluster but few overall layout changes. The green Grizzly Bear node indicates the observation that the analyst moved, and the blue nodes indicate other observations that moved to other clusters, with lines pointing back to the centroid (red square) of the cluster to which they were previously assigned.

of a cluster locally, and/or there being little evident change in the projection. We implemented the three pipelines with interaction directions that correspond to cluster-only (see Figure 5(a)), DR-only (see Figure 9), and both algorithms (see Figure 6) support into the Pollux system, selecting the interaction used with a constant variable that is loaded as tool launches.

6.2 Interactions on the Animals Dataset

The Animals dataset [62] used in this example is a high-dimensional, quantitative dataset that contains 49 animals and 85 attributes, with the attributes expressing information about each animal's color, habitat, diet, and behavior, among other properties. We have often used this dataset in previous studies because of its general knowledge applicability. The dataset is normalized with 0–100 ranges for each attribute.

The Pollux projection direction initially created six clusters to encapsulate the animals within this dataset. Using the three interactions pipelines previously detailed, we performed each of the three interactions noted at the beginning of the section within the tool. Each of the three actions was approximately identical, with the user dragging the Grizzly Bear into the Predators cluster near the Lion; the interaction was just interpreted different by the system in each of the three cases.

Cluster-Only Interaction. This interaction corresponds to the intent "The Grizzly Bear is a Predator," and the state of the visualization before and after the interaction is shown in Figure 10. As a result of this interaction and the updates to the underlying weight vector, the layout measure of Kruskal's stress in the projection was reduced by 10.6%, whereas the Dunn index clustering measure was reduced by 4.6%, yielding a more accurate projection with more compact clusters. As shown in the figure, the interaction resulted in the ejection of several of the smaller carnivorous animals (Rat, Raccoon, Weasel) from the cluster, as well as the Polar Bear being reassigned to a cluster with many of the aquatic animals. The three attributes that the models learned to be most important to this interaction and underlying intent are Hunter, Domestic, and Fierce, each of which are traits that are similar in larger predatory animals. The system does appear to respond well to the intent of the interaction.

DR-Only Interaction. This interaction corresponds to the intent "The Grizzly Bear is similar to the Lion." Positioning the Grizzly Bear near the Lion in this implementation resulted in a reduction of Kruskal's stress by 2.2% but an increase in the Dunn index measure by 1.8%. This interaction only removed the Rat from the Predator cluster, in contrast to the previous interaction that also removed the Raccoon and Weasel. The resulting broader cluster explains the increase, since the Dunn index relies on cluster compactness for part of its measure. The three most important attributes learned by the models in this case were Meatteeth, Furry, and Fierce, all clear attributes that are shared by the Grizzly Bear and Lion. The system again appears to respond well to the intent of the interaction, focusing on slightly different attributes that are shared by two animals rather than the Grizzly Bear and the full Predator cluster.

Both Algorithms Interaction. This interaction corresponds to the intent "The Grizzly Bear is a Predator and is more similar to the Lion than it is to the Fox." The result of this interaction reduced both measures (Kruskal 5.0%, Dunn 6.3%), although not to the same degree as the cluster-only interaction. As with that first interaction, the Rat, Raccoon, and Weasel were all removed from the Predator cluster, but this time the Fox was removed as well. The Polar Bear also joined the Grizzly Bear in the Predator cluster. The three most important attributes here were Big, Paws, and Hunter, reflecting similarities between the Grizzly Bear and the other remaining animals of the Predator cluster, as well as excluding the smaller carnivorous animals from the cluster due to the importance of the Big attribute.

6.3 The States Dataset

Similar to the previous Animals data, this States dataset is both high-dimensional and quantitative. It contains 35 dimensions that describe population, ethnicity, gender, age, education, and employment measures of the 48 contiguous U.S. states [113]. The dataset was not initially normalized, so we processed the raw data to yield z-score normalized distributions for each attribute.

The Pollux projection direction again initially created six clusters to encapsulate the states. We again perform interactions that affect the clustering model, the DR model, and both within the tool. Each of the three actions was again approximately identical, with the user dragging the Pennsylvania node into the small cluster of high-population states near New York. Pennsylvania was selected for this interaction rather than other high-population states (e.g., Florida, Illinois) because of its geographical proximity to New York for a DR-only similar measure.

Cluster-Only Interaction. This interaction corresponds to the intent "Pennsylvania is a high-population state." As a result of this interaction, both Florida and Illinois were also drawn into the high-population states cluster. The updates to the underlying weight vector and hence to the layout and clustering led to an increase in both Kruskal's stress (4.2%) and the Dunn index (4.1%), both of which are apparent effects of the inflated size of the High-Population cluster (Figure 11); evidently these states do not share many properties aside from their populations. The three attributes most important to this interaction learned by the models are Population Density, Income Per Capita, and High School Graduation Rate.

DR-Only Interaction. This interaction corresponds to the intent "Pennsylvania is similar to New York." This interaction was unique in that it did not cause any other cluster reassignments among the other states, only producing minor updates to the layout in other parts of the projection. This indicates that the properties shared by Pennsylvania and New York are uncommon within the rest of the dataset. The High-Population cluster did inflate again, resulting in another increase to

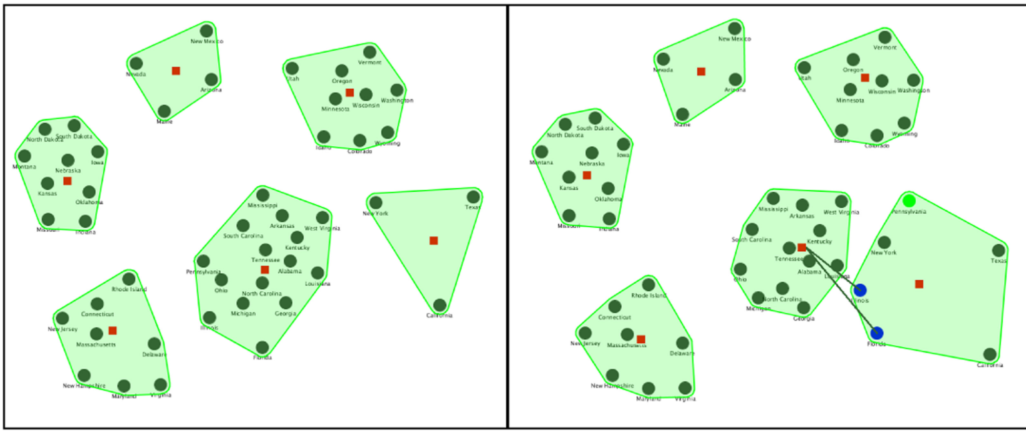


Fig. 11. The intent "Pennsylvania is a high-population state" only uses the interaction computation of the clustering model. The associated interaction produced several cluster reassignments relevant to the high-population states cluster but few overall layout changes, although the High-Population cluster is noticeably inflated. The green Pennsylvania node indicates the observation that the analyst moved, and the blue nodes indicate observations that moved to other clusters, with lines pointing back to the centroid (red square) of the cluster to which they were previously assigned.

the Dunn index (1.7%), but the global Kruskal's stress decreased slightly (0.3%) as a result of the other minor layout updates. The three most important attributes learned by the models from this interaction are High School Graduation Rate, Area, and Population (not Population Density, as in the previous interaction).

Both Algorithms Interaction. This interaction corresponds to the intent "Pennsylvania is a high-population state and is more similar to New York than it is to California." Adding the intended measure of similarity between Pennsylvania and New York resulted in the High-Population cluster excluding one high-population state but bringing in two others. As with the cluster-only interaction, this intent resulted in Illinois being pulled into the High-Population cluster, but Florida was not. Instead, Ohio and Michigan joined the cluster. The reason for this is evident from the important attributes learned by the models, which in addition to Population Density include Proportion of Farm Land and Age 65+. Florida notably differs from the other high-population states with less farmland and higher proportion of elderly residents, leading to its exclusion from the cluster. The cluster did inflate again, resulting in an increase to the Dunn index of 3.0%, as well as a moderate increase to Kruskal's stress of 1.5%.

7 DISCUSSION

In this section, we reflect on the results of our interaction demonstrations from the previous section, as well as discuss several additional considerations that a system designer should factor in when designing a system that incorporates an interactive projection with ambiguous interactions. In addition, we discuss bigger-picture considerations of this work, both with respect to the future of semantic interaction and intent inference, as well as to the relationship between **Human-Computer Interaction (HCI)** and ML. Further, we discuss the limitations of this work and present future research opportunities in this interaction design space.

7.1 Reflections on the Interactions

The interactions that were provided in the previous section were purposefully left visually ambiguous. As seen in the changes to the DR and clustering metrics in the preceding examples, each of the seemingly identical interactions that were performed in Sections 6.2 and 6.3 produced different values in addition to different visual outputs. These values were driven by the individual pipelines that modeled those interactions, with the activation and deactivation of varying interaction computation(s) leading to the learning of unique weight vectors, emphasizing different attributes in the data in response to the interactions.

In the Animals dataset example, the second interaction that used only the interaction computation of the DR model was unique in increasing the Dunn index measure for cluster quality, indicative of clusters that did not fit the data as well as the default projection. In contrast, the other two interactions resulted in decreases to both the Dunn index and to Kruskal's stress. In addition to unique visual output in both the layout and the clustering assignments, standard metrics of quality differ as the interaction computations are activated and deactivated.

However, an increase in either (or both) of these metrics is not necessary a bad result. For example, the first interaction in the States data example (only the clustering model had an activated interaction computation) resulted in increases to both metrics. However, this is explainable from an inspection of the data, seeing that although the states California, Texas, New York, Florida, Pennsylvania, and Illinois all share a high-population property, there is little else in the data that is a common feature of all of these states. Here, an analyst can see this conclusion visually with the inflated cluster, but the underlying models also capture that result. Indeed, the visualization and the quality metrics can be used in a complementary fashion when necessary to gain additional insight about the effects of an interaction. A metric alone is simply a scalar value that will not be sufficient to communicate the nuanced effects of the interaction, but it can be used to support what an analyst is seeing in the visualization as a result of their interaction.

7.2 Visualizing Guidance and Feedback

As an analyst performs interactions in a projection, some of the ambiguity can be removed by providing the analyst with visual guidance demonstrating how the system will interpret the interaction. Such features are similar to those seen in Explainable AI systems [49], as they reveal details of the underlying model state. Some tools with interactive projections have already implemented such feedback techniques. For example, StarSPIRE [6] includes a feature within the documents to highlight words judged to be important by the underlying models. This feature is used by analysts both to determine the overall importance of a document within the projections and to locate the important phrases and sections of a document [122]. In contrast, early versions of Andromeda [97] used a dynamic-length slider to indicate the weights applied to dimensions, whereas later versions demonstrated changes in system-computed similarities [119]. Related techniques seen in visual interfaces include changing the color of observations or colors (also seen in Andromeda) and drawing boundaries around tentatively recognized clusters. Resolving the ambiguity in these interactions is certainly complex. As such, providing additional visual feedback to communicate the system's interpretation of the user's intent can help to resolve any resulting issues from these interactions. Techniques such as visual scent [15–17, 83, 131] can be used to convey the effects of an interaction to the analyst so that the analyst has the ability to correct their interaction before the system begins to respond to the interaction.

7.3 Shared or Separate Weight Vectors

Much of the discussion in the preceding sections has glossed over the changes made to the system parameters after handling an interaction. In single-model systems like Andromeda, the weight update is straightforward, as there is only one model learning weights. In multi-model systems such as StarSPIRE, the weight update becomes moderately more complex, as a relevance threshold needs to be learned in addition to set of term weights. Further increasing in complexity, SIRIUS [32]

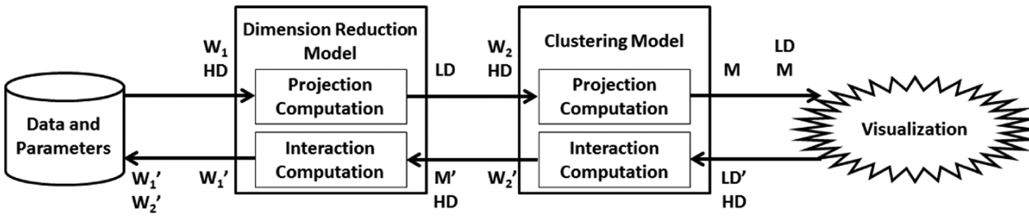


Fig. 12. A representation of dataflow using two independent algorithms, each of which maintains a separate weight vector.

uses two separate weight vectors, one for the observation projection and one for the attribute projection, each of which are computed separately but with some dependency between them as interactions are performed.

Thinking more generally about a visualization system that incorporates both DR and clustering algorithms, a designer should consider whether each model should maintain its own weight vector or if a weight vector should be shared between models. The tasks supported and pipeline selected both play a large role in this decision. For example, Castor [126] follows the "Dimension Reduction Preprocessing for Clustering" projection pattern, with the cluster assignments naturally following from the low-dimension positions of the observations. In such a case, it is natural to support a shared weight vector between the models. In contrast, a system like iVisClustering [63], which computes DR and clustering separately on the high-dimensional data without interaction between the two (the "Independent Algorithms" pattern [123]), naturally supports separate weight vectors. Such a system without dependence between the inputs and outputs of algorithms and separate weight vectors is provided in Figure 12, demonstrated by the following equations:

$$
\begin{aligned}
ld\_positions &= DR\_PROJECT\ (\ weights_1,\ hd\_data\ ) \\
memberships &= \ CLUSTER\_PROJECT\ (\ weights_2,\ hd\_data\ ) \\
weights_2' &= \ CLUSTER\_INTERACT\ (\ hd\_data,\ memberships'\ ) \\
weights_1' &= DR\_INTERACT\ (\ hd\_data,\ ld\_positions'\ ).
\end{aligned}
$$

7.4 Toward Resolving Semantic Interaction Ambiguity

Semantic interaction aims to improve the quality of user interactions by enabling an analyst to directly manipulate a projection rather than attempt to finesse the parameters of the underlying mathematical model(s) [33, 37, 38]. However, this work demonstrates that the variety of possible meanings and intents of an analyst's interactions can be difficult to capture in a single tool. In other words, interactions such as repositioning an observation are inherently ambiguous; this is the "With Respect to What" usability challenge [98]. Introducing clusters can make some interactions easier by introducing a hard target but also introduces added ambiguity (e.g., has the analyst moved an observation into a cluster, or was their goal to move the observation closer to some of the observations within the cluster?) [129].

Resolving this ambiguity is critical to the future of semantic interaction. The role that the analyst is embodying at the current step of their analysis process can alter their interaction possibilities as well [125], leading to a further dimension to consider when inferring user intent from interactions. Future tools that make use of immersive spaces likewise have a need to infer the purpose of even more complex interactions in supporting user sensemaking [107]. As such, several techniques have been introduced to provide feedback to the analyst regarding how the system will interpret their interaction [53, 105]. For example, Figure 2 displays the selection interactions in Andromeda, including nearest neighbor selection, radius selection, and additional observation selection. Pollux also limits the interaction space to reclassifying observations and manipulating their position within clusters [127]. However, limiting the interaction space can prevent analysts from learning more about their data from forbidden interactions.

To truly allow for free-form interactivity and data manipulation in systems, there is an inherent tradeoff between creating complex interactions that are precise but difficult for analysts to remember and perform and creating simple interactions that are ambiguous but easy for analysts. Precise interactions can include components such as a double-click to indicate the importance of the source of the target of the update, multi-touch to denote the cardinality of the interaction, and presenting visual feedback to the analyst before the interaction is handled by the system [17, 83, 131]. More ambiguous interactions could learn from a small training set and/or the interaction history to match the intent of a user to the interactions that they perform, such as found in ActiveInk [87]. Such a training set could be generated by an elicitation study, understanding precisely how analysts wish to perform these interactions.

7.5 Complementary HCI and ML Perspectives

This work makes use of terminology from the ML community ("metric learning") as well as from the HCI community ("inferring user intent"). This choice is not accidental or unintentional, as the research presented here exists at the intersection of both fields: we include discussions of both the semi-supervised training of ML algorithms as well as design and interaction considerations for interactive visualization tools. Perhaps the clearest example of such symmetry within this work comes from the overlapping ideas of inferring the intent of a user and mapping that intent to a learned metric in the system.

From the HCI perspective, we provide discussions of design considerations, suggestions for responding to interaction ambiguity, and two representative interactive interfaces. For ML, we focus upon some of the underlying mathematics, reference a number of implementations of interactive systems that can learn from analyst interactions, and discuss learning issues within this design space. Both of these perspectives address separate but complementary facets of the same problem.

These perspectives are at the core of resolving the "Two Black Boxes" problem [128], which considers the challenge of extracting human cognition from the black box of the brain and presenting it to a machine intelligence to complement the Explainable AI challenge of extracting information from the black box of the machine. With such considerations in mind, bias can be mitigated and trust can be built between human and machine co-learners.

7.6 User Modeling, Provenance, and the Role of Individual Differences

The interaction implementations that we demonstrated in the previous section assumed a cold start, with no advance knowledge of the user's tendencies or behaviors, as well as no advance knowledge of previous analyses that had been conducted on the data. Designing systems that respond to not only the actions of a user but also to their tendencies and behaviors is an active area of research [77, 130]. Surveys of the application of provenance data from Xu et al. [134] and Ragan et al. [84] demonstrate that tracking the interactions, states, and data associated with the exploration of the user can be analyzed to inform a system about that user. Methods for learning about a user vary from simply tracking mouse click [9, 76] to more complex techniques such as eye gaze [111] and fMRI measurements [80]. Understanding the unique behaviors and capabilities of each user can assist with accurately personalizing interactions and their responses to the expressed intent.

7.7 Limitations

Although we overview the challenges of the "With Respect to What" problem as it pertains to DR and clustering algorithms, we make no claim regarding the completeness of our survey of interactions. For example, an additional portion of the interaction that can be considered is its speed. Perhaps a quick interaction can be used to indicate a cluster reassignment only, whereas a slower interaction can be interpreted as more carefully positioning the final location of the observation, indicating a positional similarity interaction. Extending to a future extreme, a system can be designed with speech recognition support to permit a user to explain their intent in natural language while performing an interaction. The creativity of visualization designers is nearly boundless, and we fully expect that future designers can extend this work.

We also note that our interaction evaluation from the previous section was limited in the quantity of both datasets and types of interactions. In this work, we sought only to briefly explore and demonstrate the effects for ambiguous interactions. A rigorous user study involving 20 to 30 participants would yield much stronger results and better conclusions about what analysts can gain from using these interactions. Further, we did not explore visualization tradeoffs in these tools. For example, the participants' insights might have been different had we rendered cluster membership with nodes of various colors rather than convex hulls. We plan to perform such a study on the effect of cluster encoding on interactive clustering in the future; until then, some work in the literature has touched on the non-interactive portion of this area [54, 93]. Finally, we limited the interactions that were permitted in our tools to a small set of what is possible. We do plan to undertake the elicitation study and expand on the details of this interaction space in future work.

The choice of algorithms included in our examples in Section 6 was driven by our past research, to conveniently use previous implementations of interaction computations for WMDS and weighted k-means, in which we identified the challenges that result due to ambiguous interactions in the visual space. As a result, our focus on these algorithms ignores the complexities that can result from ambiguous interactions in more complex projection spaces such as t-SNE and UMAP and with more advanced density- or model-based clustering approaches. We leave a thorough exploration of these matters for future work.

7.8 Future Work

This work is focused on a novel space at the intersection of DR, clustering, and semantic interaction. Although a number of systems exist at the intersection of two of these three fields, only a few can be found at the intersection of all three [126, 127]. As a result, some of our discussion of potential interactions is speculative rather than demonstrative. We plan a future elicitation study to determine how analysts will naturally wish to perform these interactions, as well as future implementations that provide such interactions. Semantic interaction is a relatively new field that is still developing, and it presents several opportunities for future system and interaction development, including the potential for a generic toolkit or system for semantic interaction.

8 CONCLUSION

This work models the complexity and ambiguity inherent in the interaction space of DR and clustering algorithms in interactive projections. We framed this discussion in the context of the "With Respect to What" problem, a research challenge in visual analytics identified by Self et al [98]. Through our discussion, we identified several factors necessary to consider for such interactions: thinking in high- or low-dimensional space, interaction with observations or clusters, interaction with source and destination, and cardinality of interaction. We presented a new pipeline representation that incorporates both a projection direction to generate the visualization and an interaction direction to handle the interaction and interpret the intent of the analyst. We demonstrated the utility of this pipeline to disambiguate a series of potential DR and clustering algorithm projection and interaction computations, including both interactions on observations and interactions on models. We demonstrated the utility of several of these interactions within a visual analytics prototype tool, examining the changes to both insight and the underlying model as a result of the interactions. Finally, we discussed additional considerations related to the implementation of such systems, as well as supplementary interactions and visual metaphors that further assist in communication and exploration.

ACKNOWLEDGMENTS

The members of the Virginia Tech InfoVis and BaVA (Bayesian Visual Analytics) groups provided an excellent sounding board for the ideas contained herein. We also thank the reviewers for their insightful comments in improving the presentation of this work.

REFERENCES

[1] H. Akaike. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control* 19, 6 (1974), 716–723.
[2] B. Alper, N. Riche, G. Ramos, and M. Czerwinski. 2011. Design study of LineSets, a novel set visualization technique. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (Dec. 2011), 2259–2267. https://doi.org/10.1109/TVCG.2011.186
[3] Saleema Amershi, James Fogarty, and Daniel Weld. 2012. Regroup: Interactive machine learning for on-demand group creation in social networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, New York, NY, 21–30.
[4] G. Andrienko, N. Andrienko, S. Rinzivillo, M. Nanni, D. Pedreschi, and F. Giannotti. 2009. Interactive visual clustering of large collections of trajectories. In *Proceedings of the 2009 IEEE Symposium on Visual Analytics Science and Technology*. 3–10. https://doi.org/10.1109/VAST.2009.5332584
[5] Hans-Ulrich Bauer and Klaus R. Pawelzik. 1992. Quantifying the neighborhood preservation of self-organizing feature maps. *IEEE Transactions on Neural Networks* 3, 4 (1992), 570–579.
[6] L. Bradel, C. North, L. House, and S. Leman. 2014. Multi-model semantic interaction for text analytics. In *Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology (VAST'14)*. 163–172. https://doi.org/10.1109/VAST.2014.7042492
[7] Matthew Brehmer, Michael Sedlmair, Stephen Ingram, and Tamara Munzner. 2014. Visualizing dimensionally-reduced data: Interviews with analysts and a characterization of task sequences. In *Proceedings of the 5th Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV'14)*. ACM, New York, NY, 1–8. https://doi.org/10.1145/2669557.2669559
[8] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang. 2012. Dis-function: Learning distance functions interactively. In *Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST'12)*. 83–92. https://doi.org/10.1109/VAST.2012.6400486
[9] Eli T. Brown, Alvitta Ottley, Helen Zhao, Quan Lin, Richard Souvenir, Alex Endert, and Remco Chang. 2014. Finding Waldo: Learning about users from their interactions. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (Dec. 2014), 1663–1672. https://doi.org/10.1109/TVCG.2014.2346575
[10] Andreas Buja, Dianne Cook, and Deborah F. Swayne. 1996. Interactive high-dimensional data visualization. *Journal of Computational and Graphical Statistics* 5, 1 (1996), 78–99.
[11] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman. 1999. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann.
[12] A. Chatzimparmpas, R. M. Martins, and A. Kerren. 2020. t-viSNE: Interactive assessment and interpretation of t-SNE projections. *IEEE Transactions on Visualization and Computer Graphics* 26, 8 (2020), 2696–2714. https://doi.org/10.1109/TVCG.2020.2986996
[13] Lisha Chen and Andreas Buja. 2009. Local multidimensional scaling for nonlinear dimension reduction, graph drawing, and proximity analysis. *Journal of the American Statistical Association* 104, 485 (2009), 209–219. https://doi.org/10.1198/jasa.2009.0111
[14] X. Chen, J. Z. Self, L. House, J. Wenskovitch, M. Sun, N. Wycoff, J. R. Evia, S. Leman, and C. North. 2018. Be the data: Embodied visual analytics. *IEEE Transactions on Learning Technologies* 11, 1 (Jan. 2018), 81–95. https://doi.org/10.1109/TLT.2017.2757481

[15] Ed H. Chi, Lichan Hong, Michelle Gumbrecht, and Stuart K. Card. 2005. ScentHighlights: Highlighting conceptually-related sentences during reading. In *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI'05)*. ACM, New York, NY, 272–274. https://doi.org/10.1145/1040830.1040895

[16] E. H. Chi, L. Hong, J. Heiser, and S. K. Card. 2006. Scentindex: Conceptually reorganizing subject indexes for reading. In *Proceedings of the 2006 IEEE Symposium on Visual Analytics Science and Technology*. 159–166. https://doi.org/10.1109/VAST.2006.261418

[17] Ed H. Chi, Lichan Hong, Julie Heiser, Stuart K. Card, and Michelle Gumbrecht. 2007. ScentIndex and ScentHighlights: Productive reading techniques for conceptually reorganizing subject indexes and highlighting passages. *Information Visualization* 6, 1 (2007), 32–47.

[18] J. Choo, C. Lee, C. K. Reddy, and H. Park. 2013. UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec. 2013), 1992–2001. https://doi.org/10.1109/TVCG.2013.212

[19] Jason Chuang and Daniel J. Hsu. 2014. Human-centered interactive clustering for data analysis. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS'14)*.

[20] Jason Chuang, Christopher D. Manning, and Jeffrey Heer. 2012. Termite: Visualization techniques for assessing textual topic models. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI'12)*. ACM, New York, NY, 74–77. https://doi.org/10.1145/2254556.2254572

[21] C. Collins, G. Penn, and S. Carpendale. 2009. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (Nov. 2009), 1009–1016. https://doi.org/10.1109/TVCG.2009.122

[22] D. L. Davies and D. W. Bouldin. 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1, 2 (April 1979), 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

[23] Renato Cordeiro de Amorim and Christian Hennig. 2015. Recovering the number of clusters in data sets with noise features using feature rescaling factors. *Information Sciences* 324 (2015), 126–145.

[24] P. Demartines and J. Hérault. 1993. Vector quantization and projection neural network. In *New Trends in Neural Computation*, José Mira, Joan Cabestany, and Alberto Prieto (Eds.). Springer, Berlin, Germany, 328–333.

[25] P. Demartines and J. Herault. 1997. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks* 8, 1 (1997), 148–154.

[26] Marie Desjardins, James MacGlashan, and Julia Ferraioli. 2007. Interactive visual clustering. In *Proceedings of the 12th International Conference on Intelligent User Interfaces*. ACM, New York, NY, 361–364.

[27] Lee R. Dice. 1945. Measures of the amount of ecologic association between species. *Ecology* 26, 3 (1945), 297–302.

[28] Chris Ding and Xiaofeng He. 2004. K-means clustering via principal component analysis. In *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*. ACM, New York, NY, 29–36. https://doi.org/10.1145/1015330.1015408

[29] Chris Ding and Tao Li. 2007. Adaptive dimension reduction using discriminant analysis and K-means clustering. In *Proceedings of the 24th International Conference on Machine Learning (ICML'07)*. ACM, New York, NY, 521–528. https://doi.org/10.1145/1273496.1273562

[30] E. P. dos Santos Amorim, E. V. Brazil, J. Daniels, P. Joia, L. G. Nonato, and M. C. Sousa. 2012. iLAMP: Exploring high-dimensional spacing through backward multidimensional projection. In *Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST'12)*. 53–62. https://doi.org/10.1109/VAST.2012.6400489

[31] E. P. dos Santos Amorim, E. V. Brazil, J. Daniels, P. Joia, L. G. Nonato, and M. C. Sousa. 2012. iLAMP: Exploring high-dimensional spacing through backward multidimensional projection. In *Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST'12)*. 53–62. https://doi.org/10.1109/VAST.2012.6400489

[32] Michelle Dowling, John Wenskovitch, J. T. Fry, Scotland Leman, Leanna House, and Chris North. 2019. SIRIUS: Dual, symmetric, interactive dimension reductions. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan. 2019), 172–182. https://doi.org/10.1109/TVCG.2018.2865047

[33] Michelle Dowling, John Wenskovitch, Peter Hauck, Adam Binford, Nicholas Polys, and Chris North. 2018. A bidirectional pipeline for semantic interaction. In *2018 IEEE Workshop on Machine Learning from User Interaction for Visualization and Analytics (MLUI) (VIS'18)*. 1–11. https://doi.org/10.1109/MLUI52768.2018.10075562

[34] Michelle Dowling, Nathan Wycoff, Brian Mayer, John Wenskovitch, Scotland Leman, Leanna House, Nicholas Polys, Chris North, and Peter Hauck. 2019. Interactive visual analytics for sensemaking with big text. *Big Data Research* 16 (2019), 49–58. https://doi.org/10.1016/j.bdr.2019.04.003

[35] Joseph C. Dunn. 1974. Well-separated clusters and optimal fuzzy partitions. *Journal of Cybernetics* 4, 1 (1974), 95–104.

[36] Alex Endert. 2014. Semantic interaction for visual analytics: Toward coupling cognition and computation. *IEEE Computer Graphics and Applications* 34, 4 (July 2014), 8–15. https://doi.org/10.1109/MCG.2014.73

[37] Alex Endert, Patrick Fiaux, and Chris North. 2012. Semantic interaction for sensemaking: Inferring analytical reasoning for model steering. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2879–2888.

[38] Alex Endert, Patrick Fiaux, and Chris North. 2012. Semantic interaction for visual text analytics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'12)*. ACM, New York, NY, 473–482. https://doi.org/10.1145/2207676.2207741

[39] A. Endert, C. Han, D. Maiti, L. House, S. Leman, and C. North. 2011. Observation-level interaction with statistical models for visual analytics. In *Proceedings of the 2011 IEEE Conference on Visual Analytics Science and Technology (VAST'11)*. 121–130. https://doi.org/10.1109/VAST.2011.6102449

[40] Mateus Espadoto, Gabriel Appleby, Ashley Suh, Dylan Cashman, Mingwei Li, Carlos Scheidegger, Erik W. Anderson, Remco Chang, and Alexandru C. Telea. 2023. UnProjection: Leveraging inverse-projections for visual analytics of high-dimensional data. *IEEE Transactions on Visualization and Computer Graphics* 29, 2 (2023), 1559–1572. https://doi.org/10.1109/TVCG.2021.3125576

[41] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea. 2021. Toward a quantitative survey of dimension reduction techniques. *IEEE Transactions on Visualization and Computer Graphics* 27, 3 (2021), 2153–2173. https://doi.org/10.1109/TVCG.2019.2944182

[42] R. Faust, D. Glickenstein, and C. Scheidegger. 2019. DimReader: Axis lines that explain non-linear projections. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 481–490. https://doi.org/10.1109/TVCG.2018.2865194

[43] Ronen Feldman and James Sanger. 2007. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press.

[44] I. K. Fodor. 2002. *A Survey of Dimension Reduction Techniques*. Technical Report. Office of Scientific and Technical Information, U.S. Department of Energy, Washington, DC. https://doi.org/10.2172/15002155

[45] Edward B. Fowlkes and Colin L. Mallows. 1983. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association* 78, 383 (1983), 553–569.

[46] S. L. France and J. D. Carroll. 2011. Two-way multidimensional scaling: A review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 41, 5 (Sept. 2011), 644–661. https://doi.org/10.1109/TSMCC.2010.2078502

[47] E. R. Gansner, Y. Hu, and S. Kobourov. 2010. GMap: Visualizing graphs and clusters as maps. In *Proceedings of the 2010 IEEE Pacific Visualization Symposium (PacificVis'10)*. 201–208. https://doi.org/10.1109/PACIFICVIS.2010.5429590

[48] Tera Marie Green, William Ribarsky, and Brian Fisher. 2009. Building and applying a human cognition model for visual analytics. *Information Visualization* 8, 1 (2009), 1–13.

[49] David Gunning. 2017. *Explainable Artificial Intelligence (XAI)*. DARPA.

[50] M. Halkidi and M. Vazirgiannis. 2001. Clustering validity assessment: Finding the optimal partitioning of a data set. In *Proceedings of the 2001 IEEE International Conference on Data Mining*. 187–194. https://doi.org/10.1109/ICDM.2001.989517

[51] Rex Hartson and Pardha S. Pyla. 2012. *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*. Elsevier.

[52] Leanna House, Scotland Leman, and Chao Han. 2015. Bayesian visual analytics: BaVA. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 8, 1 (2015), 1–13.

[53] Xinran Hu, Lauren Bradel, Dipayan Maiti, Leanna House, and Chris North. 2013. Semantics of directly manipulating spatializations. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2052–2059.

[54] R. Jianu, A. Rusu, Y. Hu, and D. Taggart. 2014. How to display group information on node-link diagrams: An evaluation. *IEEE Transactions on Visualization and Computer Graphics* 20, 11 (Nov. 2014), 1530–1541. https://doi.org/10.1109/TVCG.2014.2315995

[55] Paulo Joia, Danilo Coimbra, Jose A. Cuminato, Fernando V. Paulovich, and Luis G. Nonato. 2011. Local affine multidimensional projection. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (2011), 2563–2571.

[56] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon. 2008. Visual analytics: Definition, process, and challenges. In *Information Visualization*. Lecture Notes in Computer Science, Vol. 4950. Springer, 154–175. https://doi.org/10.1007/978-3-540-70956-5_7

[57] H. Kim, J. Choo, H. Park, and A. Endert. 2016. InterAxis: Steering scatterplot axes via observation-level interaction. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (Jan. 2016), 131–140. https://doi.org/10.1109/TVCG.2015.2467615

[58] M. Kim, K. Kang, D. Park, J. Choo, and N. Elmqvist. 2017. TopicLens: Efficient multi-level visual topic exploration of large-scale document collections. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (Jan. 2017), 151–160. https://doi.org/10.1109/TVCG.2016.2598445

[59] T. Kohonen. 1990. The self-organizing map. *Proceedings of the IEEE* 78, 9 (Sept. 1990), 1464–1480. https://doi.org/10.1109/5.58325

[60] Mark A. Kramer. 1991. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal* 37, 2 (1991), 233–243. https://doi.org/10.1002/aic.690370209

[61] Joseph B. Kruskal. 1964. Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis. *Psychometrika* 29, 1 (1964), 1–27.

[62] Christoph H. Lampert, Hannes Nickisch, Stefan Harmeling, and Jens Weidmann. 2009. *Animals with Attributes: A Dataset for Attribute Based Classification*. Institute of Science and Technology Austria.

[63] Hanseung Lee, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. 2012. iVisClustering: An interactive visual document clustering via topic modeling. *Computer Graphics Forum* 31, 3pt3 (2012), 1155–1164. https://doi.org/10.1111/j.1467-8659.2012.03108.x

[64] John Aldo Lee, Amaury Lendasse, and Michel Verleysen. 2002. Curvilinear distance analysis versus Isomap. In *Proceedings of the 10th European Symposium on Artificial Neural Networks (ESANN'02)*. 185–192.

[65] John A. Lee and Michel Verleysen. 2007. *Nonlinear Dimensionality Reduction*. Springer Science & Business Media.

[66] John A. Lee and Michel Verleysen. 2009. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing* 72, 7-9 (2009), 1431–1443.

[67] Scotland C. Leman, Leanna House, Dipayan Maiti, Alex Endert, and Chris North. 2013. Visual to parametric interaction (V2PI). *PLoS One* 8, 3 (2013), e50474.

[68] S. Liu, D. Maljovec, B. Wang, P. Bremer, and V. Pascucci. 2017. Visualizing high-dimensional data: Advances in the past decade. *IEEE Transactions on Visualization and Computer Graphics* 23, 3 (March 2017), 1249–1268. https://doi.org/10.1109/TVCG.2016.2640960

[69] G. M. H. Mamani, F. M. Fatore, L. G. Nonato, and F. V. Paulovich. 2013. User-driven feature space transformation. *Computer Graphics Forum* 32, 3pt3 (2013), 291–299. https://doi.org/10.1111/cgf.12116

[70] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. 2010. Introduction to information retrieval. *Natural Language Engineering* 16, 1 (2010), 100–103.

[71] Tauno Metsalu and Jaak Vilo. 2015. ClustVis: A web tool for visualizing clustering of multivariate data using principal component analysis and heatmap. *Nucleic Acids Research* 43, W1 (2015), W566–W570.

[72] Vladimir Molchanov and Lars Linsen. 2014. Interactive design of multidimensional data projection layout. In *EuroVis—Short Papers*, N. Elmqvist, M. Hlawitschka, and J. Kennedy (Eds.). Eurographics Association. https://doi.org/10.2312/eurovisshort.20141152

[73] L. G. Nonato and M. Aupetit. 2019. Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics* 25, 8 (2019), 2650–2673. https://doi.org/10.1109/TVCG.2018.2846735

[74] Brian Keith Norambuena, Tanushree Mitra, and Chris North. 2023. Mixed multi-model semantic interaction for graph-based narrative visualizations. In *Proceedings of the 28th International Conference on Intelligent User Interfaces (IUI'23)*, Association for Computing Machinery, New York, NY, 866–888. https://doi.org/10.1145/3581641.3584076

[75] Erkki Oja. 1991. Data compression, feature extraction, and autoassociation in feedforward neural networks. *Artificial Neural Networks* 1 (1991), 737–745.

[76] Alvitta Ottley, Roman Garnett, and Ran Wan. 2019. Follow the clicks: Learning and anticipating mouse interactions during exploratory data analysis. *Computer Graphics Forum* 38, 3 (2019), 41–52. https://doi.org/10.1111/cgf.13670

[77] Alvitta Ottley, Huahai Yang, and Remco Chang. 2015. Personality as a predictor of user strategy: How locus of control affects search strategies on tree visualizations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*. ACM, New York, NY, 3251–3254. https://doi.org/10.1145/2702123.2702590

[78] Malay K. Pakhira, Sanghamitra Bandyopadhyay, and Ujjwal Maulik. 2004. Validity index for crisp and fuzzy clusters. *Pattern Recognition* 37, 3 (2004), 487–501.

[79] F. V. Paulovich, D. M. Eler, J. Poco, C. P. Botha, R. Minghim, and L. G. Nonato. 2011. Piecewise Laplacian-based projection for interactive data exploration and organization. *Computer Graphics Forum* 30, 3 (2011), 1091–1100. https://doi.org/10.1111/j.1467-8659.2011.01958.x

[80] Evan M. M. Peck, Beste F. Yuksel, Alvitta Ottley, Robert J. K. Jacob, and Remco Chang. 2013. Using fNIRS brain sensing to evaluate information visualization interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 473–482.

[81] Dan Pelleg, Andrew W. Moore, et al. 2000. X-means: Extending K-means with efficient estimation of the number of clusters. In *Proceedings of the 17th International Conference on Machine Learning (ICML'00)*, Vol. 1. 727–734.

[82] Darius Pfitzner, Richard Leibbrandt, and David Powers. 2009. Characterization and evaluation of similarity measures for pairs of clusterings. *Knowledge and Information Systems* 19, 3 (2009), 361.

[83] Peter Pirolli. 2003. A theory of information scent. *Human-Computer Interaction* 1 (2003), 213–217.

[84] Eric D. Ragan, Alex Endert, Jibonananda Sanyal, and Jian Chen. 2016. Characterizing provenance in visualization and data analysis: An organizational framework of provenance types and purposes. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (Jan. 2016), 31–40. https://doi.org/10.1109/TVCG.2015.2467551

[85] William M. Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66, 336 (1971), 846–850.

[86] Bastian Rieck and Heike Leitte. 2017. Agreement analysis of quality measures for dimensionality reduction. In *Topological Methods in Data Analysis and Visualization IV*, Hamish Carr, Christoph Garth, and Tino Weinkauf (Eds.). Springer International Publishing, Cham, Switzerland, 103–117.

[87] Hugo Romat, Nathalie Henry Riche, Ken Hinckley, Bongshin Lee, Caroline Appert, Emmanuel Pietriga, and Christopher Collins. 2019. ActiveInk: (Th)Inking with data. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI'19)*. ACM, New York, NY, Article 42, 13 pages. https://doi.org/10.1145/3290605.3300272

[88] Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20 (1987), 53–65.

[89] Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 5500 (2000), 2323–2326.

[90] Tuukka Ruotsalo, Jaakko Peltonen, Manuel Eugster, Dorota Głowacka, Ksenia Konyushkova, Kumaripaba Athukorala, Ilkka Kosunen, et al. 2013. Directing exploratory search with interactive intent modeling. In *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*. ACM, New York, NY, 1759–1764.

[91] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim. 2017. Visual interaction with dimensionality reduction: A structured literature analysis. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (Jan. 2017), 241–250. https://doi.org/10.1109/TVCG.2016.2598495

[92] Bahador Saket, Samuel Huron, Charles Perin, and Alex Endert. 2020. Investigating direct manipulation of graphical encodings as a method for user interaction. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2020), 482–491. https://doi.org/10.1109/TVCG.2019.2934534

[93] B. Saket, P. Simonetto, S. Kobourov, and K. Börner. 2014. Node, node-link, and node-link-group diagrams: An evaluation. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (Dec. 2014), 2231–2240. https://doi.org/10.1109/TVCG.2014.2346422

[94] J. W. Sammon. 1969. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers* C-18, 5 (1969), 401–409.

[95] K. Saul Lawrence and T. Roweis Sam. 2002. *Think Globally, Fit Locally: Unsupervised Learning of Nonlinear Manifolds*. Technical Report CIS-02-18. University of Pennsylvania, Philadelphia, PA.

[96] Gideon Schwarz. 1978. Estimating the dimension of a model. *Annals of Statistics* 6, 2 (1978), 461–464. https://doi.org/10.1214/aos/1176344136

[97] Jessica Zeitz Self, Michelle Dowling, John Wenskovitch, Ian Crandell, Ming Wang, Leanna House, Scotland Leman, and Chris North. 2018. Observation-level and parametric interaction for high-dimensional data analysis. *ACM Transactions on Interactive Intelligent Systems* 8, 2 (June 2018), Article 15, 36 pages. https://doi.org/10.1145/3158230

[98] Jessica Zeitz Self, Radha Krishnan Vinayagam, J. T. Fry, and Chris North. 2016. Bridging the gap between user intention and model parameters for human-in-the-loop data analytics. In *Proceedings of the Workshop on Human-in-the-Loop Data Analytics (HILDA'16)*. ACM, New York, NY, Article 3, 6 pages. https://doi.org/10.1145/2939502.2939505

[99] John Sharko, Georges Grinstein, and Kenneth A. Marx. 2008. Vectorized RadViz and its application to multiple cluster datasets. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1444–1427.

[100] Roger N. Shepard. 1962. The analysis of proximities: Multidimensional scaling with an unknown distance function. I. *Psychometrika* 27, 2 (1962), 125–140.

[101] Frank M. Shipman and Catherine C. Marshall. 1999. Formality considered harmful: Experiences, emerging themes, and directions on the use of formal representations in interactive systems. *Computer Supported Cooperative Work* 8, 4 (1999), 333–352.

[102] Charles Spearman. 1904. The proof and measurement of association between two things. *American Journal of Psychology* 15, 1 (1904), 72–101.

[103] David J. Spiegelhalter, Nicola G. Best, Bradley P. Carlin, and Angelika Van Der Linde. 2002. Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64, 4 (2002), 583–639. https://doi.org/10.1111/1467-9868.00353

[104] Catherine A. Sugar and Gareth M. James. 2003. Finding the number of clusters in a dataset: An information-theoretic approach. *Journal of the American Statistical Association* 98, 463 (2003), 750–763.

[105] Ashley Suh, Ab Mosca, Shannon Robinson, Quinn Pham, Dylan Cashman, Alvitta Ottley, and Remco Chang. 2022. Inferential tasks as an evaluation technique for visualization. In *EuroVis 2022—Short Papers*, Marco Agus, Wolfgang Aigner, and Thomas Hoellt (Eds.). Eurographics Association. https://doi.org/10.2312/evs.20221086

[106] Gabor J. Szekely and Maria L. Rizzo. 2005. Hierarchical clustering via joint between-within distances: Extending ward's minimum variance method. *Journal of Classification* 22, 2 (2005), 151–183.

[107] Ibrahim A. Tahmid, Lee Lisle, Kylie Davidson, Chris North, and Doug A. Bowman. 2022. Evaluating the benefits of explicit and semi-automated clusters for immersive sensemaking. In *Proceedings of the 2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR'22)*. 479–488. https://doi.org/10.1109/ISMAR55827.2022.00064

[108] Joshua B. Tenenbaum, Vin De Silva, and John C. Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 5500 (2000), 2319–2323.
[109] Robert L. Thorndike. 1953. Who belongs in the family? *Psychometrika* 18, 4 (1953), 267–276. https://doi.org/10.1007/BF02289263
[110] Robert Tibshirani, Guenther Walther, and Trevor Hastie. 2001. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63, 2 (2001), 411–423.
[111] Dereck Toker, Sébastien Lallé, and Cristina Conati. 2017. Pupillometry and head distance to the screen to predict skill acquisition during information visualization tasks. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces (IUI'17)*. ACM, New York, NY, 221–231. https://doi.org/10.1145/3025171.3025187
[112] Barbara Tversky. 2005. *Visuospatial Reasoning*. Cambridge University Press.
[113] United States Census Bureau. 2019. State Data Center Program. Retrieved April 25, 2023 from https://www.census.gov/about/partners/sdc.html.
[114] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. 2009. Dimensionality reduction: A comparative review. *Journal of Machine Learning Research* 10, 66-71 (2009), 13.
[115] Jarkko Venna and Samuel Kaski. 2006. Visualizing gene interaction graphs with local multidimensional scaling. In *Proceedings of the European Symposium on Artificial Neural Networks (ESANN'06)*. 557–562.
[116] T. Villmann, R. Der, M. Herrmann, and T. M. Martinetz. 1997. Topology preservation in self-organizing feature maps: Exact definition and measurement. *IEEE Transactions on Neural Networks* 8, 2 (1997), 256–266.
[117] Tatiana von Landesberger, Sebastian Fiebig, Sebastian Bremm, Arjan Kuijper, and Dieter W. Fellner. 2014. Interaction taxonomy for tracking of user actions in visual analytics applications. In *Handbook of Human Centric Visualization*. Springer, 653–670.
[118] Emily Wall, Subhajit Das, Ravish Chawla, Bharath Kalidindi, Eli T. Brown, and Alex Endert. 2018. Podium: Ranking data using mixed-initiative visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 288–297.
[119] Ming Wang, John Wenskovitch, Leanna House, Nicholas Polys, and Chris North. 2021. Bridging cognitive gaps between user and model in interactive dimension reduction. *Visual Informatics* 5, 2 (2021), 13–25. https://doi.org/10.1016/j.visinf.2021.03.002
[120] Xu-Meng Wang, Tian-Ye Zhang, Yu-Xin Ma, Jing Xia, and Wei Chen. 2016. A survey of visual analytic pipelines. *Journal of Computer Science and Technology* 31, 4 (2016), 787–804. https://doi.org/10.1007/s11390-016-1663-1
[121] Kilian Q. Weinberger, Fei Sha, and Lawrence K. Saul. 2004. Learning a kernel matrix for nonlinear dimensionality reduction. In *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*. ACM, New York, NY, 106–113. https://doi.org/10.1145/1015330.1015345
[122] J. Wenskovitch, L. Bradel, M. Dowling, L. House, and C. North. 2018. The effect of semantic interaction on foraging in text analysis. In *Proceedings of the 2018 IEEE Conference on Visual Analytics Science and Technology (VAST'18)*. 13–24. https://doi.org/10.1109/VAST.2018.8802424
[123] John Wenskovitch, Ian Crandell, Naren Ramakrishnan, Leanna House, Scotland Leman, and Chris North. 2018. Towards a systematic combination of dimension reduction and clustering in visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 24, 01 (Jan. 2018), 131–141. https://doi.org/10.1109/TVCG.2017.2745258
[124] John Wenskovitch, Michelle Dowling, and Chris North. 2020. With respect to what: Simultaneous interaction with dimension reduction and clustering projections. In *Proceedings of the 25th International Conference on Intelligent User Interfaces (IUI'20)*. ACM, New York, NY, 177–188. https://doi.org/10.1145/3377325.3377516
[125] John Wenskovitch, Corey Fallon, Kate Miller, and Aritra Dasgupta. 2021. Beyond visual analytics: Human-machine teaming for AI-driven sensemaking. In *Proceedings of the IEEE VIS Workshop on Trust and Expertise in Visual Analytics (TREX'21)*. 40–44. https://doi.org/10.1109/TREX53765.2021.00012
[126] John Wenskovitch and Chris North. 2017. Observation-level interaction with clustering and dimension reduction algorithms. In *Proceedings of the 2nd Workshop on Human-in-the-Loop Data Analytics (HILDA'17)*. ACM, New York, NY, Article 14, 6 pages. https://doi.org/10.1145/3077257.3077259
[127] John Wenskovitch and Chris North. 2019. Pollux: Interactive cluster-first projections of high-dimensional data. In *Proceedings of the 2019 IEEE Visualization in Data Science (VDS'19)*. 38–47. https://doi.org/10.1109/VDS48975.2019.8973381
[128] John Wenskovitch and Chris North. 2020. Interactive artificial intelligence: Designing for the "Two Black Boxes" problem. *Computer* 53, 8 (2020), 29–39. https://doi.org/10.1109/MC.2020.2996416
[129] John Wenskovitch and Chris North. 2021. An examination of grouping and spatial organization tasks for high-dimensional data exploration. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2 2021), 1742–1752. https://doi.org/10.1109/TVCG.2020.3028890
[130] John Wenskovitch, Michelle Zhou, Christopher Collins, Remco Chang, Michelle Dowling, Alex Endert, and Kai Xu. 2020. Putting the 'I' in interaction: Interactive interfaces personalized to individuals. *IEEE Computer Graphics and Applications* 40, 3 (2020), 73–82. https://doi.org/10.1109/MCG.2020.2982465

[131] W. Willett, J. Heer, and M. Agrawala. 2007. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov. 2007), 1129–1136. https://doi.org/10.1109/TVCG.2007.70589

[132] Axel Wismüller, Michel Verleysen, Michael Aupetit, and John Aldo Lee. 2010. Recent advances in nonlinear dimensionality reduction, manifold and topological learning. In *Proceedings of the 18th European Symposium on Artificial Neural Networks (ESANN'10)*.

[133] Xuanli Lisa Xie and Gerardo Beni. 1991. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 08 (Aug. 1991), 841–847. https://doi.org/10.1109/34.85677

[134] Kai Xu, Alvitta Ottley, Conny Walchshofer, Marc Streit, Remco Chang, and John Wenskovitch. 2020. Survey on the analysis of user interactions and visualization provenance. *Computer Graphics Forum* 39, 3 (2020), 757–783. https://doi.org/10.1111/cgf.14035

[135] Rui Xu and D. Wunsch. 2005. Survey of clustering algorithms. *IEEE Transactions on Neural Networks* 16, 3 (May 2005), 645–678. https://doi.org/10.1109/TNN.2005.845141

[136] J. S. Yi, Y. A. Kang, and J. Stasko. 2007. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov. 2007), 1224–1231. https://doi.org/10.1109/TVCG.2007.70515

Received 25 August 2020; revised 26 March 2023; accepted 2 March 2023