DimBridge: Interactive Explanation of Visual Patterns in Dimensionality Reductions with Predicate Logic

Brian Montambault, Gabriel Appleby, Jen Rogers, Camelia D. Brumar, Mingwei Li, Remco Chang

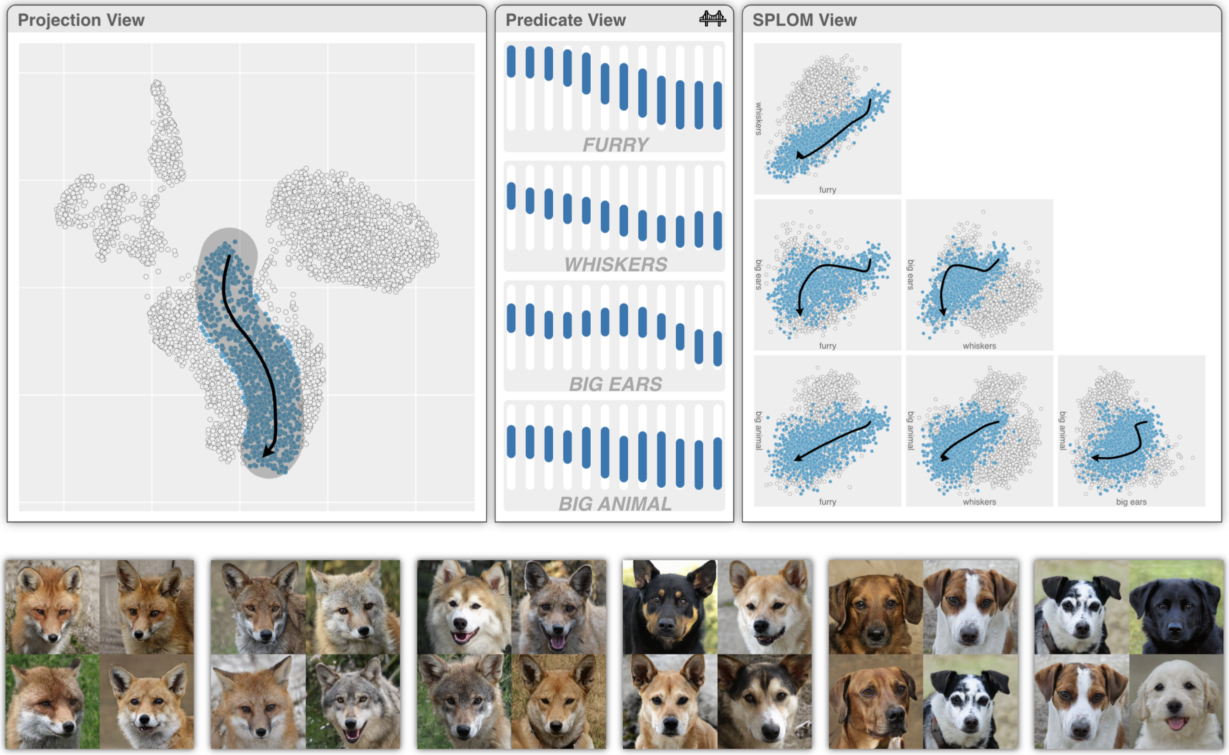

Fig. 1: DimBridge is a system that helps users understand visual patterns in dimensionality reduction-based 2D projections. Within an interface, users can brush perceived patterns in a projection (left), and DimBridge computes *predicates* in response (middle) – compact explanations of the user's selection expressed in terms of the data space. The data dimensions that compose a predicate help place the data in a context well-understood by users, shown as a scatterplot matrix (right).

Abstract— Dimensionality reduction techniques are widely used for visualizing high-dimensional data. However, support for interpreting patterns of dimension reduction results in the context of the original data space is often insufficient. Consequently, users may struggle to extract insights from the projections. In this paper, we introduce DimBridge, a visual analytics tool that allows users to interact with visual patterns in a projection and retrieve corresponding data patterns. DimBridge supports several interactions, allowing users to perform various analyses, from contrasting multiple clusters to explaining complex latent structures. Leveraging first-order predicate logic, DimBridge identifies subspaces in the original dimensions relevant to a queried pattern and provides an interface for users to visualize and interact with them. We demonstrate how DimBridge can help users overcome the challenges associated with interpreting visual patterns in projections.

Index Terms—Predicates, Dimensionality Reduction, Explainable Machine Learning

1 Introduction

The demand for interactive visual exploration techniques for high-dimensional data has significantly grown due to the substantial increase in data generation over the last decade [54]. Among these techniques, dimensionality reduction (DR) is one of the most prominent methods for visualizing high-dimensional data. Well-known DR techniques such as MDS [48], PCA [66], t-SNE [84], and UMAP [59] generate low-dimensional projections of data, where the spatial proximity of points encodes a measure of similarity between data items [25,63]. These techniques offer 2D representations of high-dimensional data, facilitating users in identifying data patterns in the high-dimensional space more readily.

However, despite the utility of DR methods in understanding *what* data items are similar, few provide insight into *why* the data are similar. As a result, users often need to rely on patterns that they perceive in the projection, e.g., clusters of points, spatial density, or distinctive shapes, to infer characteristics of the original high-dimensional data [12,49,73,74]. While some of these patterns may indeed reflect

- *Brian Montambault, Gabriel Appleby, Jen Rogers, Camelia D. Brumar, Mingwei Li, and Remco Chang are with Tufts University. E-mail: {brian.montambault, gabriel.appleby, jennifer.rogers, camelia_daniela.brumar, mingwei.li, remco.chang}@tufts.edu*

underlying data characteristics, they can also be influenced by noise, artifacts stemming from the DR process [47,58,63], or even the propensity of humans to see visual patterns where none exist [22,85,87]. Modern visualization tools and interaction techniques offer some assistance in understanding projections, such as helping users discern DR distortions [5,51,79], examining or contrasting perceived clusters [21,53,56], or generating new plausible data points in the projection [2,20,24].

In this paper, we present DimBridge, building on existing work to offer a new way to explore and understand perceived visual patterns in a DR projection. The workflow for DimBridge is designed for simplicity. As shown in Figure 1, a user highlights a pattern in the projection space, and DimBridge (1) identifies relevant data dimensions and (2) determines value intervals for each dimension to explain the selected visual pattern. The resulting subspace, defined by the selected data dimensions and intervals, is visualized using a scatterplot matrix (SPLOM). This visualization enables a user to observe the selected visual pattern in a reduced context of the original high-dimensional data space – known as *"Mapping Synthesized [Data] to Original Dimensions"* in DR task taxonomies [12,63]. By linking visual patterns in the projection space to the original data space, DimBridge helps users understand the meaning of these patterns in a familiar context.

DimBridge supports several user interactions (select, select and contrast, select and draw) for identifying visual patterns in a projection, including clusters, outliers, and shapes. Once a pattern is selected, DimBridge automatically identifies *predicates* by optimizing a subset of data dimensions and intervals that best explain the visual pattern. In addition to highlighting exactly where the pattern occurs within this subspace, this approach yields a reduced subspace of dimensions that is practical to visualize using a SPLOM and allows users to further contextualize observed patterns within the original data space. DimBridge is equipped with two predicate induction algorithms, a novel Predicate Regression algorithm tailored for speed and scalability, complemented by the PIXAL algorithm for accuracy [61]. Both algorithms allow for a smoothness constraint, which enables DimBridge to support a novel select and draw interaction. Within this framework, predicates serve as a conceptual bridge connecting the low-dimensional (i.e., the projection) and the high-dimensional (i.e., the original data) spaces.

To demonstrate DimBridge's efficacy, we showcase the system on various datasets, present its utility in a case study, and iteratively evaluate its functionality with domain experts. With these examples, we also hope to demonstrate that DimBridge provides a means for users to explain projections using the original data space, solving key challenges in interpretability associated with conventional projection-based high-dimensional data visualization.

In summary, our work makes the following contributions:

- We introduce DimBridge, a system that uses predicate logic to *bridge* the projection and data spaces by simultaneously identifying dimensions and intervals that explain a given pattern, helping users to make sense of 2D projections of high-dimensional data.
- We introduce a new interaction design for selecting visual patterns within DR projections and a new algorithm for generating predicates with smoothness constraints given a selected visual pattern (Predicate Regression).
- We evaluate DimBridge through several showcases with different domain applications, a case study, and evaluations with researchers from materials science and pharmaceutical drug discovery.

2 Related Work

Our work on DimBridge is based on prior work on (1) high-dimensional data visualization, (2) visualization systems that help users make sense of projections, and (3) the use of predicates in the visualization community.

2.1 High-Dimensional Data Visualization

Various techniques for multi-dimensional visualization have been presented over the years. A more thorough discussion of these techniques can be found in the following surveys [28,42,54]. We investigate these approaches to multi-dimensional visualization from the perspective of their level of dimensional reduction, from no reduction to significant reduction, and the techniques' respective affordances.

One of the most common families of methods is multiline graphs [3,43], which plot several to many features either overlaid or stacked vertically over another dimension. This visual technique does not reduce the dimensionality of the data in terms of losing information or combining dimensions. An example of this is the parallel coordinates plot [38,43], which puts each dimension on a separate axis and draws a line connecting these axes for each instance. Parallel coordinates are often used in concert with sophisticated interaction techniques for selecting groups of data items, e.g., angular brushing of dimensions [36], multi-way brushing for high dimensional pattern discovery [71], as well as augmented designs that better convey relationships between dimensions [11]. The multiline approach allows for the visualization of all dimensions but can become cluttered and hard to interpret as the number of dimensions increases.

In the same ethos as multiline approaches, small multiples techniques such as the scatterplot matrix (SPLOM) [23,35], permutation matrices of bar charts [9], and histogram matrices, are techniques used to display and analyze the relationships between multiple variables in a dataset. Small multiples are closely related to the idea of dimension stacking [50], wherein each dimension is broken into histogram-like buckets, with further dimensions recursively partitioned within preceding dimension buckets. Small multiple views allow for a direct comparison across different data dimensions, enhancing our ability to discern patterns and anomalies within the actual data space. Faceting data, or dimensions, across multiple views quickly leads to scalability issues, and thus methods for selecting informative views [89], used in conjunction with visual quality metrics to rank views [10], are common approaches to handle such problems. This is especially the case in high-dimensional data. However, determining the most relevant subspace remains a significant challenge. One approach is to incorporate subspace analysis to help users identify relevant subspaces [82,83]. These have two limitations when it comes to explaining patterns in projections. First, they typically select relevant or important dimensions one at a time, choosing the next dimension greedily. This can result in subspaces that are optimal given the previous subspace, but not optimal given all possible subspaces. Second, the selected dimensions are not bounded by intervals. Further analysis is required to identify the regions of a subspace that correspond with a visual pattern. In DimBridge, we propose a predicate induction algorithm that jointly optimizes both dimension selections and intervals, providing compact and bounded subspaces that better aide the understanding of complex patterns in projections. We discuss this approach in further detail in section 5.

Projections, also known as dimensionality reduction (DR) methods, scale well in terms of the number of samples and dimensions, allowing them to overcome the limitations of many other high-dimensional visualization techniques. Dimensionality reduction techniques attempt to maintain the underlying structure of the original dataset in a lower-dimensional projection. PCA [66], t-SNE [84], and UMAP [59] are three of the most popular dimensionality reduction techniques. t-SNE and UMAP are unique from PCA in that they are non-linear DR techniques [59,84], and beneficial when the data is too complex to adequately project with linear methods. Dimensionality reduction methods allow for detected visual patterns in a lower-dimensional space, facilitating the discovery of high-dimensional patterns. While these methods excel at detecting patterns, making these patterns interpretable to humans remains a challenge.

2.2 Making Sense of Projections

Non-linear DR, increasingly incorporated into visual analytic systems [72], aims to translate high-dimensional similarities, such as the proximity of data items to their nearest neighbors, into spatial closeness within the 2D projection space. Spatial proximity alone, however, lacks the context for *why* points are positioned, e.g., what makes 2 data items similar? Methods exist for bringing in the context of individual data dimensions, e.g., per-dimension aggregations arranged in multiple views [21,79], dimension-specific spatial groupings in the projection [77], or augmenting the projection space with 2D lines [16] or curves [27] that encode large variations in the data space. Another line of work supports users in making better sense of DR hyperparameters [4] rather than patterns in projections or inverting the projection to increase understanding [2,20,24]. There has also been an effort to add features into the DR [30,52]. There is also work that tends to focus on educating users about projections and hyperparameters [18,19] or combining many techniques to help users find a good projection [17,75].

The motivation for better understanding a projection is often driven by the visual patterns found within the scatterplot, such as the meaning behind separate clusters [88]. This objective is achieved by explaining clusters or classes through feature importance [45,55,65,68] which can help users find similar or dissimilar groups [95]. Additionally, there are numerous methods using a projection as a scaffold for interactively selecting clusters and in turn visually comparing clusters [15] or performing a contrastive cluster analysis [31,32]. However, as DR methods are prone to error, cluster analysis methods that convey the distortion induced by DR [44,80] remain important for proper interpretation. Observations made through cluster analysis inform methods for steering DR [91], often performed in response to imperfections found in projections, reflecting an iterative process of (1) visual analysis of projections and clusters and (2) data annotation.

Many previous techniques rely on brushing to help users understand a projection. These techniques vary from traditional brushing (e.g., cross-filtering) over the DR scatterplot [29] to methods that allow users to explore the space through SPLOMs [7,8] or PCPs [36,43,71]. There are also more advanced methods such as [6,57] that allow for multiple dimensions to be brushed at once. More recently, Jeon et al. [44] introduced distortion of points local to brushed area. DimBridge builds upon these earlier methods and introduces a new interaction technique to highlight arbitrary shapes in the DR projection. Based on predicate induction, this technique is able to exclude points that are similar in the projection but not the original dimensions, and include points that are similar in the original dimensions but not in the projection. Additionally, Predicate induction provides compact explanations of patterns found in the projection by identifying a reduced, bounded set of the original data dimensions corresponding to the selected data points.

2.3 Predicates in Visual Analytics

Predicates are frequently used in the visual analytics community to identify important subsets of data in an interpretable way, with a variety of approaches.

Examples such as the SEER or Scorpion systems leverage predicate logic to describe user-specified patterns [33,34,92], outliers [90], or query results [86]. Systems such as SuRE and Rule Matrix leverage rule-based logic for interpreting complex machine learning results [60,93]. This approach has been applied to both understanding the predictions of machine learning models, for example Anchors [69], as well as their errors, as demonstrated by iSEA [94] and Domino [26]. Predicate logic is leveraged for its interpretability and expressivity. Predicates are human-readable, supporting a user in understanding otherwise complex logic in machine learning and data analysis. Predicates are also expressive – a user can easily modify a predicate to express their domain knowledge.

More generally, a number of systems have used automated generation of interpretable rules to help users understand complex phenomena. For example, by providing human-understandable summaries to better understand database provenance [1]. The Graphiti system follows a comparable approach, generating Boolean-logic rules to explain relationships between user-specified subsets from graphs by the user [78]. The DRIL system combines automatically and interactively generated rules to help analysts gain insights into the characteristics of interesting clusters of data points [14].

While these predicate or rule-based approaches have been applied to a number of common tasks in visual analytics, it has not yet been applied to understanding DR results. DimBridge leverages these strengths of predicates to provide an interpretable *bridge* between the projection and data-visualization spaces.

3 Interpreting DR Results: Design Goal and Tasks

Patterns identified in conventional DR visualizations are challenging to interpret because they lose the underlying semantics of the original dimensions, such as the context for why points are considered similar. Our work addresses these challenges by bridging the projection and underlying data space. Specifically, DimBridge facilitates the interpretation of patterns found in DR results through interactive querying of the projection, enabling retrieval and visualization of relevant subspaces of the original dimensions.

We break down the process of identifying and interpreting patterns into three high-level tasks, developed while working closely with domain experts. These tasks are the foundational requirements used to develop the DimBridge system.

(T1) Identify and query visual patterns in a projection

Dimensionality reduction (DR) transforms high-dimensional data into low-dimensional forms to facilitate visualization and analysis while preserving important structures and relationships. A common approach is to reduce the data to two dimensions and visualize the resulting patterns using a scatterplot. DimBridge should support the user in identifying potential patterns of interest and facilitate direct engagement with the projection, enabling users to explore by selecting patterns that capture their interest. There are multiple patterns that a user can identify within a projection that have unique approaches to selection and interaction. DimBridge must allow the user to identify and select each of the following patterns:

Clusters: Identifying clusters is a key application of DR projections [72], as it allows a way to classify similar and dissimilar points based on attributes. Although it can be intuitive to identify distinct groups of data points, it is not directly clear why they formed or how they differ by visualizing the projection alone.

Outliers: Identifying outliers is closely linked to recognizing clusters, yet determining if outliers in a 2D projection accurately reflect those in the high-dimensional data is challenging based solely on the projection. While outliers may be identified based on distance in the projection, it is not clear how this distance translates to the original dimensions.

Spatial Density: Identifying clusters is not always straightforward, especially with varied point densities [37,81]. Unlike clear-cut clusters, density across a projection can change gradually, complicating the task of selecting a group of points for detailed analysis. For example, points just outside a selected dense area might still belong to that group, reflecting the challenge of making precise selections.

Shapes: DR projections are also used to uncover hidden low-dimensional structures in data [72], such as identifying a low-dimensional manifold that explains the data distribution. These structures often appear as specific shapes within the projection, such as a curve slicing through a group of points.

(T2) Retrieve and visualize relevant subspaces of the original data

Understanding visual patterns in the DR results requires explaining them in the context of the data's original dimensions. Traditional visualization techniques, however, struggle with handling high-dimensional data effectively. Given a visual pattern in the DR results queried by the users, DimBridge must be able to retrieve relevant subspaces of the original data. For this subspace to be visualized effectively, it must have a relatively small number of dimensions.

(T3) Evaluate the visual pattern in context

While the goal is to explain a visual pattern in the original dimensions, distortions introduced by the DR algorithm can result in patterns in the DR results that do not correspond to patterns in the original dimensions. Beyond simply retrieving relevant subspaces, DimBridge must help users evaluate a selected pattern given a retrieved subspace.

4 Example Use Case and System Overview

We illustrate DimBridge's design goals with a simple use case. Figure 1 shows a set of animal images labeled with 14 numeric attributes (e.g.,

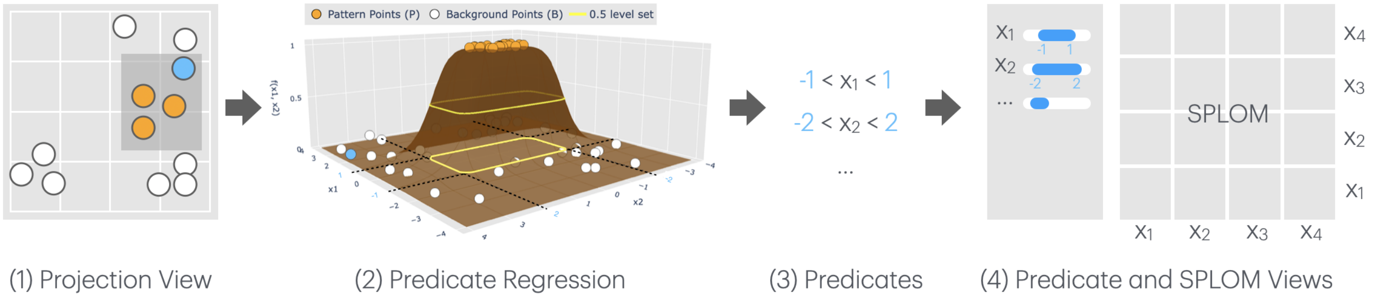

Fig. 2: System Overview. (1) User makes a brush selection in the projection view. Note that the user intends to select the cluster of orange points, but includes another data point (shown in blue) by mistake due to DR distortions. (2) DimBridge fits a multi-dimensional "bump" function (Eq. 5) to encapsulate the user selected points (orange). Notice that the algorithm automatically removes the blue point from consideration. (3) The "bump" function can be described using a reduced set of the original data dimensions and their respective intervals. These, in turn, can be represented as predicates. (4) The predicates are used to generate the predicate and the SPLOM views, which show the intervals and the selected data in the original data dimension, respectively.

furry, whiskers). After applying UMAP for dimensionality reduction and visualizing the results, the user identifies a cluster of data points forming an unusual curved shape in the *Projection View*.

The user investigates this pattern by selecting points at the top of the cluster with a bounding box and dragging it to the bottom, following the cluster's curved shape (T1). DimBridge then identifies the key dimensions and intervals that explain the visual pattern (T2). These dimensions are visualized in a scatterplot matrix in the *SPLOM View*, with the intervals shown in the *Predicate View* (T3). With these visualizations, the user observes:

- The selected shape can be explained using just 4 of the original 14 dimensions.
- The shape most closely correlates with the "furry" and "whiskers" dimensions.
- The top of the shape represents very furry animals with whiskers and big ears. Towards the bottom, the animals become larger, less furry, and have small ears and no whiskers.
- There is a downward shift in the "big ears" interval where the shape curves right, indicating a tipping point where the animals' ears become smaller.
- The shape represents a progression from foxes to wolves, wolf-like dogs, large dogs, and finally smaller puppies, which the user can see in the returned data.

To support this workflow, DimBridge models the user's interaction as a set of continuous brushes. In Figure 1, the user's drawn shape is discretized into 12 brushes, shown as 12 intervals for each dimension in the Predicate View. DimBridge uses a novel algorithm, Predicate Regression, to identify the dimensions and intervals that best explain these brushes by minimizing a differentiable loss function (see Fig. 2 (2)) and enforcing a **smoothness constraint** to ensure consistency between the 12 brushes.

In section 5 we describe our predicate induction engine. Section 6 covers the design of the visualization interface and interaction techniques. We demonstrate DimBridge's utility with three examples and a use case in drug discovery.

5 Predicate Induction Engine

The central motivation of this work is to support users' interpretation of patterns found in the DR results in the context of the original data dimensions. The predicate induction engine facilitates this by generating predicates that best explain, in the original dimensions, a visual pattern identified by the user.

5.1 Bridging DR and Data Patterns with Predicates

DimBridge uses first-order predicate logic as a "bridge" between patterns in the DR results and relevant subspaces of the original dimensions. This allows users to capitalize on the strengths of both spaces: patterns can be identified visually in the DR results, and their semantics can be recovered in the original dimensions.

A first-order predicate, Φ, is defined as a conjunction of one or more clauses, each consisting of a data dimension and a minimum and maximum value. The clause ϕ_j is defined for the j-th dimension, with minimum and maximum values denoted ϕ_j^{min} and ϕ_j^{max}. A predicate contains all data points with values that fall within the intervals defined by each clause. Alternatively, a predicate can be thought of as a function that takes a data point, x_i, as input and outputs a binary label: 1 if the data point falls within the intervals defined by each clause, and 0 otherwise:

$$\Phi(x_i) = \mathbb{1}[\phi_j^{min} \leq x_{ij} \leq \phi_j^{max},\ \forall \phi_j \in \Phi] \quad (1)$$

5.2 Generating Predicates from User Interactions

The predicate induction engine enables DimBridge to leverage predicates as a bridge between patterns in the DR results and original dimensions by generating predicates that closely match the user's selection. Unlike existing approaches that identify a subspace of relevant dimensions, DimBridge seeks a compact description of the selected pattern in the original dimensions. Our predicate induction approach jointly identifies both dimensions and intervals that describe the selected pattern.

To achieve this, we first derive a set of background points, B, and pattern points, P, from a user's interaction. Second, we define the combined dataset $X = B \cup P$ and associated binary labels, $Y = \{\mathbb{1}[x_i \in P] | x_i \in X\}$. Finally, we use a predicate induction algorithm to identify predicates that contain a set of data points, represented by the binary labels $\Phi(X) = \{\Phi(x_i) | x_i \in X\}$, that closely match the data points selected by the user, represented by Y.

In our implementation of DimBridge, we consider two predicate induction algorithms. The first is the recursive predicate induction (RPI) algorithm used by the PIXAL system [61]. This algorithm constructs predicates from the ground up, starting with a set of single clause predicates that are iteratively refined. A predicate is scored at each step and continuously refined until the score stops improving. For DimBridge, we score predicates using the F1 score calculated between $\Phi(X)$ and Y, balancing false positives (points that are in the predicate but not the user's selection) and false negatives (points that are in the user's selection but not the predicate). This results in multiple, overlapping predicates, each representing a candidate subspace. While this may be suitable for offline applications, this is too slow for an interactive system. The second is a novel algorithm, Predicate Regression that works by approximating high dimensional bounding cuboids via a differentiable proxy function.

5.2.1 Predicate Regression

Rather than generating and evaluating predicates iteratively, the Predicate Regression algorithm defines a differentiable loss function based on a reparameterization of first-order predicates, which it then minimizes to find the "best" predicate given the user-defined X and Y.

When a predicate clause fully covers the data extent along a dimension, it always returns true. Therefore, even if a predicate defines a subspace including only the clause dimensions, we can expand it to all dimensions with the interval being strictly greater than the data extent:

$$\Phi(x_i) = \mathbb{1}[\phi_j^{min} \leq x_{ij} \leq \phi_j^{max},\ \forall j = 1 \dots M] \tag{2}$$

with $\phi_j^{max} \geq \max\limits_i\{x_{ij}\}$ and $\phi_j^{min} \leq \min\limits_i\{x_{ij}\}$ when $\phi_j \notin \Phi$.

This expansion allows us to optimize all predicate clauses simultaneously. Moreover, we can define a predicate with the parameters $\boldsymbol{\mu}$ and $\mathbf{r}$, denoting a midpoint and range for each dimension (j) respectively:

$$\mu_j = \frac{\phi_j^{max} + \phi_j^{min}}{2},\ r_j = \frac{\phi_j^{max} - \phi_j^{min}}{2} \tag{3}$$

This reparameterization allows us to consider whether a data point is contained by a predicate not only as a binary label, but a continuous probability:

$$Pr(\Phi(x_i) = 1 | \mathbf{r}, \boldsymbol{\mu}, b) := \frac{1}{1 + \sum_{j=1}^{M} |\frac{1}{r_j} \cdot (x_{ij} - \mu_j)|^b} \tag{4}$$

Generating a Predicate for a Single Brush: Geometrically, the probability gives a rounded bump function (see Fig. 2 (2)) of x_i centering at $\boldsymbol{\mu}$, where b is a fixed parameter controlling the steepness of the bump. Moreover, the 0.5 level set is enclosed by the predicate bounding box, $\prod_{j=1}^{M}[\mu_j - r_j, \mu_j + r_j]$, along each data feature dimension j. To find the optimal predicate via optimizing the parameters, we rewrite $1/r_j$ as a_j and view the probability as a differentiable function of a_j and μ_j:

$$f(x_i | \mathbf{a}, \boldsymbol{\mu}, b) = \frac{1}{1 + \sum_{j=1}^{M} |a_j \cdot (x_{ij} - \mu_j)|^b} \tag{5}$$

Given an X and Y defined by a user's selection, the loss function, binary cross entropy (BCE), is defined as:

$$\mathcal{L}_{bce}(\mathbf{a}, \boldsymbol{\mu} | X, Y) = \frac{1}{N} \sum_{i=1}^{N} y_i \log(f(x_i)) + (1 - y_i) \log(1 - f(x_i))$$

Recall that a_j is the inverse of the range r_j. Enlarging the range beyond data extent (i.e. forcing $a_j \rightarrow 0$) effectively eliminates the corresponding clause in the predicate conjunction. Therefore, selection of features in the predicate can be achieved through an L_1 regularization, $||\mathbf{a}||_1$. Eventually, minimizing loss functions over $\mathbf{a}$ and $\boldsymbol{\mu}$ gives a predicate in the form $\prod_{j=1}^{M}[\mu_j^* - 1/a_j^*, \mu_j^* + 1/a_j^*]$, where

$$\mathbf{a}^*, \boldsymbol{\mu}^* = \underset{\mathbf{a}, \boldsymbol{\mu}}{\arg\min}\, \mathcal{L}_{bce}(\mathbf{a}, \boldsymbol{\mu}) + \gamma_1 \cdot ||\mathbf{a}||_1 \tag{6}$$

and γ_1 controls the strength of the predicate sparsity.

Generating Predicates for Contrasting Two or Multiple Continuous Brushes: A key requirement for a predicate induction algorithm in the interactive nature of DimBridge is the consistency and smoothness of results from consecutive interactions. For example, one may expect smoothly changing clauses when fine-tuning, contrasting queries, or brushing curves over a region. For these cases, a constraint can be added to the objective function that encourages smoothness in the resulting predicates. The draw interaction results in a discretized sequence of selections derived from the user's gesture. Given a sequence X_t, Y_t for $t = 1...T$, optimizing

$$\sum_{t=1}^{T} \mathcal{L}_{bce}(\mathbf{a_t}, \boldsymbol{\mu_t} | X_t, Y_t) \tag{7}$$

gives a sequence of predicates independent to one another. To compare two regions, we let $T = 2$; to brush curves over a region, we let $T = 12$. To encourage consistency and continuity between consecutive predicates, a smoothness loss function is introduced:

$$\mathcal{L}_{smooth} = \sum_{t=2}^{T} \gamma_a \cdot ||\mathbf{a_t} - \mathbf{a_{t-1}}||^2 + \gamma_\mu \cdot ||\boldsymbol{\mu_t} - \boldsymbol{\mu_{t-1}}||^2 \tag{8}$$

where γ_a and γ_μ control the strength of consistency and continuity between consecutive predicates.

In entirety, the loss function becomes

$$\mathcal{L} = \sum_{t=1}^{T} \mathcal{L}_{bce}(\mathbf{a_t}, \boldsymbol{\mu_t} | X_t, Y_t) + \sum_{t=1}^{T} \gamma_1 \cdot ||\mathbf{a_t}||_1 \tag{9}$$

$$+ \sum_{t=2}^{T} \gamma_a \cdot ||\mathbf{a_t} - \mathbf{a_{t-1}}||^2 + \sum_{t=2}^{T} \gamma_\mu \cdot ||\boldsymbol{\mu_t} - \boldsymbol{\mu_{t-1}}||^2 \tag{10}$$

6 Visual Interface

The DimBridge interface interface comprises three coordinated views: the *Projection View*, the *Predicate View*, and the *SPLOM View*. These views enable users to query patterns in the DR results, view and modify predicates, visualize relevant subspaces of the original dimensions, and evaluate the queried patterns in context. By coordinating these views, DimBridge allows for comprehensive exploration and understanding of patterns in high-dimensional data, linking DR results to original dimensions through interactive visualizations.

6.1 Projection View

The DR results are visualized with a scatterplot in the Projection View. A user can interact directly with the scatterplot through multiple brushing techniques to explore perceived patterns.

Select: A user can select a group of data points (P) distinct from the rest of the dataset (B) using the bounding box or lasso. The Predicate Induction Engine will then generate predicates to distinguish P from B in the original dimensions. This selection helps identify unique **clusters** based on shape or position, global **outliers** spatially distant from the rest of the data, and regions with varying **spatial density**.

Select and Contrast: Instead of comparing a cluster to the entire dataset, a user can use the bounding box or lasso to select another group of data points (B) for comparison. This interaction helps explain differences between two clusters, local outliers and their neighbors, or variations in spatial density within a specific region.

Select and Draw: If a group of data points forms a distinct, continuous **shape**, a user can use the bounding box or lasso to select the starting point. Then, they can drag to the shape's endpoint, selecting points along the way. The Predicate Induction Engine will then generate predicates to distinguish the selected points (P) at multiple intervals along the shape.

6.2 Predicate View

The Predicate View bridges patterns in the DR results with the original dimensions, highlighting points in both the Projection and SPLOM Views. Each dimension in a predicate generated by the induction engine is listed along with its associated interval. Users can modify predicates by adjusting the intervals or adding/removing dimensions to see the impact on the other views. Intervals can be represented in several ways based on user interaction:

Select: Each dimension is displayed with a horizontal bar representing its full range of values. A highlighted segment indicates the predicate's defined interval (Fig. 3). Users can modify the predicate by dragging the endpoints of the highlighted segment (Fig. 7, 8).

Select and Contrast: Instead of one interval, two intervals (one for each selection) are displayed for each dimension (Fig. 4).

Select and Draw: A user's drawn gesture is converted into discrete selections, displaying multiple intervals vertically to save space (Fig. 1).

6.3 SPLOM View

The SPLOM View visualizes subspaces of the original dimensions in a scatterplot matrix, focusing on dimensions relevant to the predicates. Only including dimensions involved in the predicates reduces clutter and illustrates the relationship between these dimensions and the data points within the predicate.

6.4 Visualizing Predicate Accuracy

The Predicate Induction Engine aims to match a user's selection as closely as possible, but distortions from the DR algorithm can cause mismatches. These mismatches are illustrated using color in the Projection and SPLOM Views. Similar to prior work [44], this provides transparency on whether a cluster can be explained by a first-order predicate in the original dimensions.

Points in the Projection and SPLOM Views are color-coded based on their relation to the predicate. True positives (points in both the user's selection and the predicate) are shown in purple. False positives (points in the predicate but not in the selection) are shown in red, and false negatives (points in the selection but not in the predicate) are shown in blue. True negatives (points in neither the selection nor the predicate) are shown in grey.

6.5 Implementation and Performance

Our implementation of DimBridge consists of a web-based front-end and a Flask server[1]. The front-end visualization is implemented in Observable, rendering the scatterplots in the projection view and the SPLOM using WebGL.

The server-side Predicate Regression algorithm is implemented using PyTorch, which allows for GPU acceleration and distributed computing for improved performance. In our exploration with expert users, we maintained interactivity (typically less than 3 seconds) with datasets up to 2,703 dimensions and 22,000 instances on an Nvidia T4 GPU. However, the latency can be reduced with additional GPU support or by reducing the number of iterations in the optimization (Equations 6 and 8).

7 Showcases

In this section, we provide three showcases using DimBridge to analyze image data, motion-capture data, and scientific data. Each of these showcases highlights user tasks in exploring and making sense of high-dimensional data.

7.1 Understanding Model-Generated Images

We use DimBridge to analyze the output of a generative vision model, demonstrating the range of patterns in DR projections and how DimBridge helps make sense of them. We consider the StyleGAN3 [46] image generation model, which generates images of animals. We define 14 textual phrases as attributes describing visual appearances (e.g., "furry," "size") and behaviors (e.g., "excited," "suspicious looking"). Using CLIP [67], we score each attribute against a generated image. This set of continuously-valued attributes forms our high-dimensional space, alongside the generated images, to verify findings.

Explaining a Cluster: DimBridge demonstrates cluster analysis in a UMAP projection, highlighting discernible groups within the data, as seen in the scatterplot (Fig. 3). Upon brushing one of the clusters, DimBridge derives the predicate that best distinguishes those data items from the rest of the dataset. The subsequent predicate shows cheetahs as uniquely furry and spotted within the dataset (Fig. 3.3-4).

There are a few false positives (colored in red), such as animals within the predicate but outside the brushed region. These are mostly cheetahs, indicating the brushed cluster was slightly imprecise.

As a baseline for comparison, we chose four dimensions and plotted the user's selection in the SPLOM (Fig. 3-4). Plotting these dimensions hardly contextualizes the user's selection against the dataset, whereas the algorithmically defined predicates allow reasoning about *why* this cluster is discriminative, highlighted in the scatterplot views (Fig. 3-3).

Contrasting Two Regions in Context: Beyond comparing a cluster with the overall dataset, users can also compare one cluster to another. In Fig. 4, DimBridge highlights the differences between kitten and puppy clusters in the DR projection, distinguishing them by features such as "whiskers" and "big ears" and noting they are both categorized as smaller than other animals in the dataset.

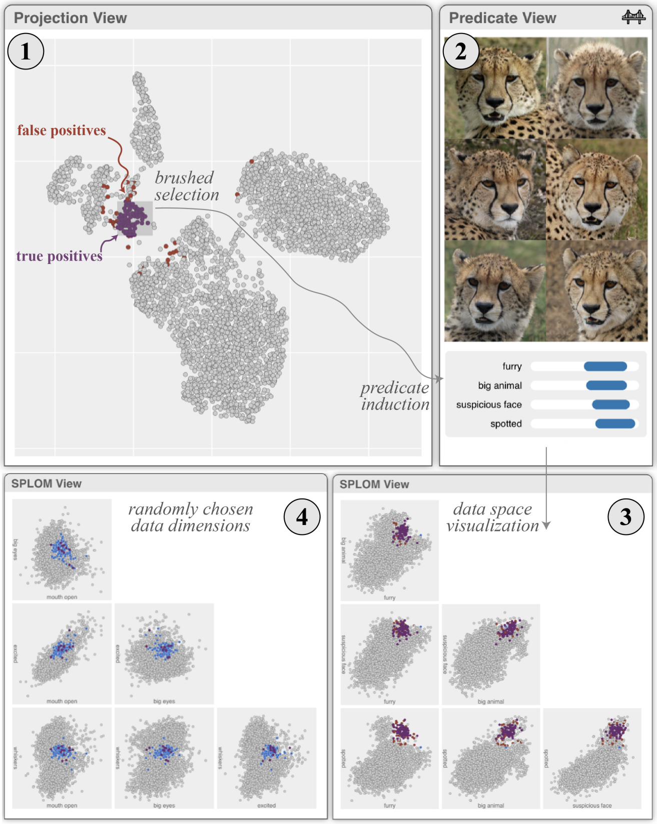

Fig. 3: DimBridge allows one to better understand a potential cluster within the output space of a generative vision model. Upon performing a brush in the scatterplot (1), DimBridge finds a predicate comprised of 4 attributes (2) that, combined, help distinguish cheetahs from other animals (3), e.g. a big animal with spotted features. In comparison, highlighting brushed data points in randomly chosen four features (4) does not help in distinguishing key features of cheetahs from other animals.

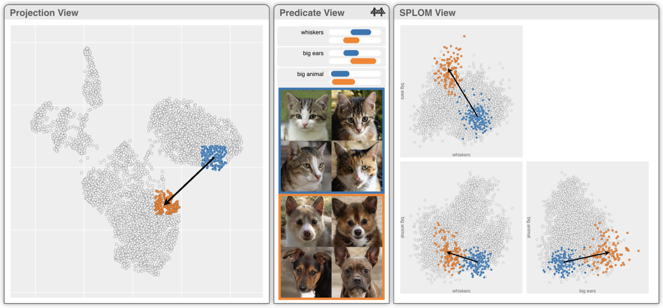

Fig. 4: DimBridge allows one to contrast one region of the dimensionality reduction plot from another. Upon brushing two regions, DimBridge finds a predicate that explains the two regions from the rest of the data points. DimBridge finds that while both kittens (blue) and puppies (orange) are not big animals, kittens have whiskers, and puppies have bigger ears.

7.2 Understanding Progression in Motion Captures

Biomechanical data, such as cyclic motion recordings, are crucial in orthopedic, rehabilitation, and sports research [41]. The Multivariate Gait Dataset [39–41, 76] captures human walking motion through 6 joint angles (left, right × ankle, knee, hip) over 101 timestamps and 10 repetitions, involving 10 subjects under 3 bracing conditions (unbraced, knee brace, ankle brace). For illustration, we selected 2 subjects from the dataset.

[1] The front-end and server source code are available on Observable (`https://observablehq.com/@tiga1231/dimbridge`) and Github (`https://github.com/tiga1231/dim-bridge`).

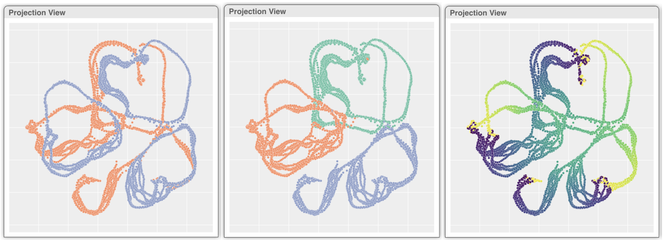

Fig. 5: DR plot of the Motion Capture dataset. **Left:** color by subject. **Middle:** color by bracing conditions. **Right:** color by time.

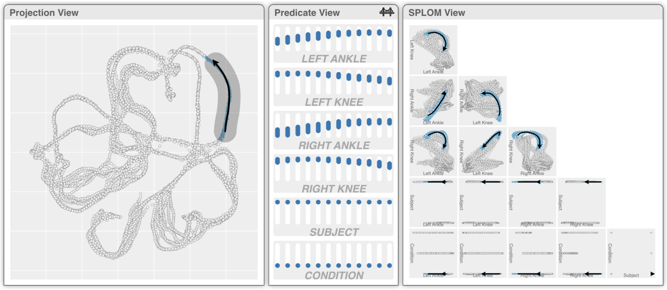

Fig. 6: DimBridge shows that the curve following the flow of time in the figure captures only one subject and condition, and the segment represents a period with increased angles on left and right ankles and slightly decreased angles on left and right knees.

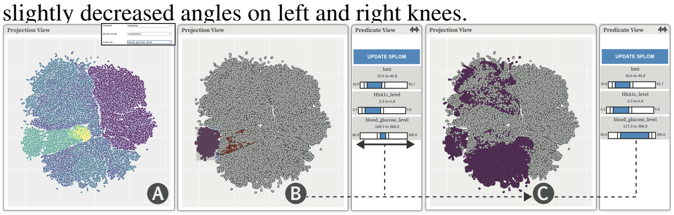

Fig. 7: A system screenshot showing the projection and predicate components: (A) Users start by coloring the projection according to a feature, here Blood Glucose level. (B) A domain practitioner explores a data subset, creating a predicate based on their selection. (C) They then adjust the Blood Glucose range to a meaningful one, observing changes in the point distribution.

The DR plot is generated via UMAP using six angular features, excluding timestamps, repetitions, subject IDs, and bracing conditions. In the DR projection scatterplot, (Fig. 5), we identified three groups of loops corresponding to the three bracing conditions, each containing two overlapping repetition bundles. The difference between the two bundles is due to the subject, as shown in Fig. 5. Coloring the DR plot by attributes aids in understanding data one attribute at a time, but analyzing multiple continuous attributes simultaneously is challenging. Displaying all six joint angles in a SPLOM creates cognitive load due to the numerous subplots. However, DimBridge generates predicates that highlight a few key views, significantly reducing the visual workload in exploratory data analysis.

Understanding Progression Over a Curve: Brushing over a segment in the direction of time flow, the predicate induction algorithm identifies it as belonging to a single subject under a specific bracing condition (Fig. 6). It summarizes the progression as an increase in left and right ankle angles with a slight decrease in left and right knee angles. The SPLOM view confirms that this summary distinguishes the brushed segment from the rest of the data.

7.3 Examining Populations within a Diabetes Study

Healthcare remains one of the fastest-growing industries in the U.S. [13] rapidly modernizing and producing data at an increasing rate [70]. New techniques are needed to make sense of this influx of data. In this use case, we examine a diabetes dataset with eight features and a binary class label. The features are gender, age, hypertension, heart disease, smoking history, body mass index, HbA1c level, and blood glucose level [62].

Specifying a Known Population: We demonstrate how a practitioner with domain knowledge can use DimBridge to manually adjust predicate ranges to find specific populations within the projection. The user starts by adding relevant features to the projection and SPLOM in the data-space visualization. In this case, the user is interested in BMI, HbA1c level, blood glucose level, and age. They can color the projection by blood glucose values (Fig. 7-A). Brushing a subset of interest in the lower left of the projection, they can generate predicates informed by the value range of this feature (Fig. 7-B).

Interactive Predicate Refinement: The user can then adjust the blood glucose level ranges in the predicate view to focus on particular states of the condition (Fig.7-B). By manipulating these ranges, the system dynamically updates the projection view, highlighting the data points that fall within the newly specified thresholds. This process is visualized in Fig.7-C, where the adjusted range causes a redistribution of the highlighted points, providing immediate visual feedback.

The ability to update these ranges is not only important for identifying distinct diabetic populations but also for exploring the complex interplay between glucose levels and other biomarkers. Once ranges are updated, users can click the "Update SPLOM" button to examine pairwise relationships between blood glucose levels and other variables like BMI, HbA1c, and age. This refined analysis can reveal subtle correlations or patterns that might be missed with a static approach.

8 Case Study: Investigating Properties of the Mn1-xGexTe Alloy

The case study demonstrates the use of DimBridge in materials science to understand the structure of alloys that are hard to model with traditional methods. Our collaborators applied advanced sampling techniques [64] and density functional theory (DFT), a method to calculate material properties based on electron behavior, to explore the Mn1-xGexTe alloy structure. This resulted in about 500 theoretical models with different combinations of manganese (Mn) and germanium (Ge).

8.1 Theoretical Models and Empirical Validation: Current Analysis Methods

Characterizing theoretical models is crucial for understanding how atoms bond and arrange themselves and the local structure, which directly relates to the material properties. For non-periodic materials like alloys, this characterization combines theoretical and empirical methods to simulate atom distributions. The standard analysis involves comparing computational predictions with experimental data.

In this case study, empirical data validating the theoretical models were obtained from the Spallation Neutron Source at Oak Ridge National Lab. Researchers analyze data using computational simulations, graphical representations, and empirical data comparisons to understand and compare the local atomic structure of different alloy compositions to experimental data.

Signals in the data, referred to as "peaks", are discussed in subsequent sections. Although generating lower-dimensional representations is straightforward, linking these to the underlying science driving the clustering remains challenging in our collaborators' current workflow.

8.2 Analyzing the Mn1-xGexTe Alloy with Dimbridge

Our collaborators used DimBridge as an exploratory analysis tool to understand the shared characteristics of computationally generated theoretical models for the Mn1-xGexTe alloy that align with experimental data from Oak Ridge National Lab. They specifically asked: **(1)** How does changing the proportion of manganese (Mn) affect the alloy's structure as it shifts from a slanted, diamond-like shape to a straight-edged, cube-like shape? **(2)** Are the shapes formed by Mn atoms consistent across the alloy space? For instance, do shapes with 6 out of 12 manganese atoms in their outer layer look the same across different Mn-Ge mixtures?

One collaborator, a graduate student, studies how varying the Mn and Ge ratio in alloy samples affects crystal structures to understand the relationship between local atomic coordination, global crystal structure, and material properties. Familiar with the data, she used DimBridge to generate hypotheses, find areas of interest, and refine her understanding and investigative strategies for subsequent data runs. During our collaborator's use of DimBridge, it became clear that the system is not just a tool for bridging dimensional spaces but also a catalyst for testing hypotheses and identifying areas for exploration.

Her exploration progressed through three distinct iterations, each using a revised dataset tailored to a new investigative strategy. All iterations used datasets from the five subsets but differed in the properties of these subsets. Subsets varied in Mn to Ge combinations: $Mn_{0.125}Ge_{0.875}Te$, $Mn_{0.2}Ge_{0.8}Te$, $Mn_{0.25}Ge_{0.75}Te$, $Mn_{0.3}Ge_{0.7}Te$, $Mn_{0.375}Ge_{0.625}Te$.

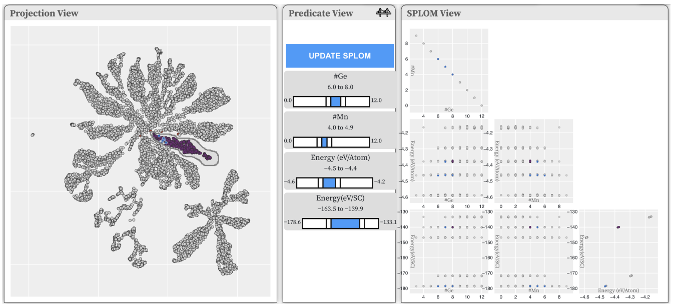

Fig. 8: UI of DimBridge with cluster selected, showing the division of points into clean groups in the SPLOM. This indicates to our collaborator the strong influence Mn has on the clustering of data.

8.2.1 Phase one: Investigating Mn's Influence on Separation

Initial exploration involved a broad examination of all five subsets. The graduate student experimented with various hyperparameters to assess their impact on data separation, including adjusting and removing attributes to test the significance of the Mn coordination shell.

After removing attributes, she would recolor the projection to confirm expected separations, such as distinct clusters within each subset. She then brushed clusters of interest to check for correlations between attributes or to verify if the clusters were due to uninteresting factors. For example, some clusters were separated by the proportion of Mn and Ge, producing results that were not informative for her research, as shown in Figure 8.

8.2.2 Phase two: Investigating Temperature Dependencies

Her findings showed that the number of Mn atoms significantly influenced data separation in lower-dimensional spaces, prompting her to adjust her analysis to focus on the relationships between different attributes and her experimental data. She then investigated how temperature affects each alloy combination (Fig. 9), focusing on attributes related to the Pair Distribution Function (PDF) plots. These plots helped her link local atomic structures, seen as peaks in the PDF, to the properties of various alloys. By examining the projection's top curve to observe value changes within interesting loop shapes, she confirmed that with increasing temperature, the data points tend to converge (Fig. 10-A).

She also made note of the small, isolated worm-like clusters outside of the larger shape. Drawing a shape over one of the small clusters, she saw that these small clusters are composed of a single temperature (Fig. 10-B). She noted, "something is definitely happening here! For the [$Mn_{0.3}Ge_{0.7}Te$] datasets, it looks like it's making worms grouped by temperature for a bunch of different POSCARs. I assume that's the case for the [$Mn_{0.25}Ge_{0.75}Te$], and [$Mn_{0.2}Ge_{0.}Te$], worms as well." Color coding the projection by combination, she confirmed her assumption on the data subsets.

8.2.3 Phase Three: Investigating Subsets of Interest

In the next stage of exploration, our collaborator created a dataset focusing on the outputs of her existing model rather than temperature dependencies, aligning it more closely with her experiments. This smaller, targeted dataset highlights subsets that previously formed

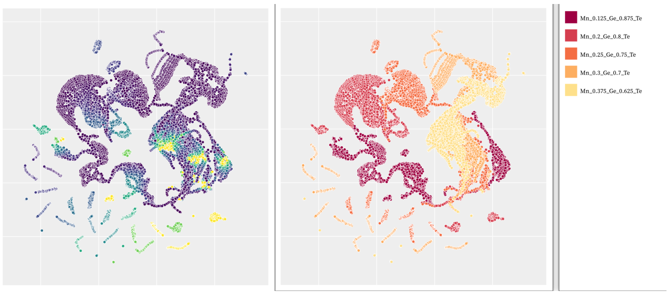

Fig. 9: Projection view showing temperature dependencies color-coded by attributes. The left is color-coded by temperature and the right is color-coded by the data subset indicating the varying combinations of Mn and Ge.

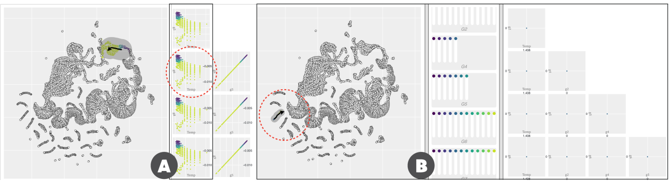

Fig. 10: (A) DimBridge explains a trail within the alloy temperature prediction dataset. The SPLOM in the data view shows curves merging as the temperature increases. (B) The view of the data subset for the drawn shape selection indicates that the selected cluster contains only one temperature value.

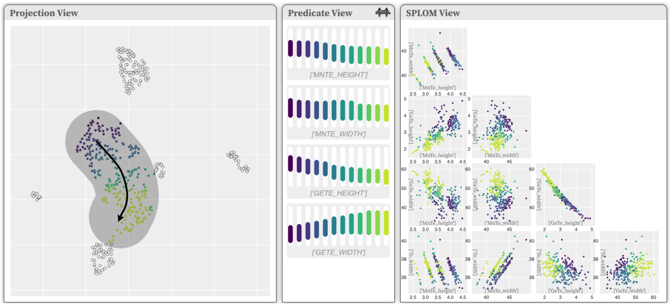

Fig. 11: DimBridge views showing the attributes of the drawn shape selection of the larger cluster. In the SPLOM, you can see the correlation between Te height and Mn height.

unique, worm-like clusters. Building on her analysis suggesting Mn influences Te positions, she colored the projection by $MnGe_r$.

After identifying an interesting cluster, she examined its predicate ranges and potential relationships. Observing correlations, she drew a shape down the larger cluster (Fig. 11), making a key observation: the height and shape of MnTe peaks significantly impact Te peak locations and heights, showing a stronger correlation with MnTe peaks than with GeTe/Te peaks. This consistency suggests that manganese (Mn) plays a pivotal role in altering Te atom positions, while germanium (Ge) has little effect. "I really love seeing this because it's showing me how influential the Mn is in the PDF patterns."

Revisiting the questions posed in the previous section: **How does changing the proportion of manganese (Mn) affect the way the alloy's structure shifts from a slanted, diamond-like shape to a straight-edged, cube-like shape?** Our collaborator found that Mn significantly influences the alloy's structure, showing a clear correlation between the intensities of Mn-Te and Te-Te peaks across the alloy space. This helps understand how Ge atoms cause local distortions, especially in low Mn concentrations, where Ge shifts the structure towards a more rhombohedral form. **Are the shapes formed by manganese (Mn) atoms similar across the alloy space?** In the initial exploration, projections showed clustering based on the number of Mn atoms in the coordination shell, regardless of composition. This suggests that the proximity of Mn atoms within their shells is more crucial than the overall elemental mix.

8.3 Observations

During our collaborator's interaction with DimBridge, we observed several notable aspects. Her process was highly iterative, moving from broad to focused analysis, often switching between datasets. The adaptability of DimBridge was a significant asset, allowing her to switch datasets, add and remove columns, and get immediate feedback on selections. The ability to dynamically update projections and data spaces enabled rapid iteration on hypotheses and analysis direction, even accounting for unexpected clusters defined by unusual attributes.

As well, our collaborator found DimBridge's functionality to draw a shape was extremely valuable for understanding changes in properties between one data subset to another, especially during the second phase of the analysis of temperature dependencies as seen in Figure 10-A.

9 Discussions and Limitations

DimBridge is designed to support the user in making sense of visual patterns in dimensionality reduction projections. In this section, we reflect on emergent insights, implications of DimBridge's value beyond its original design goals, limitations, and opportunities for future work.

9.1 DimBridge Design Considerations: Why SPLOM?

We use a SPLOM due to its efficacy for visualizing patterns in high dimensional space. This could involve the characterization of subspaces or the relationships between attribute values, which are both important tasks for our collaborators. Additionally, the SPLOM view allows a user to visualize the predicate value ranges of the relevant attributes in all possible pairings. For these reasons, we chose SPLOM over alternative high dimensional techniques, detailed in 2.1. However, future work will explore other methods of visualization to make DimBridge more flexible for a wider variety of datasets and tasks.

9.2 The Value of Flexibility

Along with creating a flexible analysis environment, it is also important to consider how we can enhance DimBridge to be more adaptable in terms of its composition and features. Below, we outline several opportunities for doing so.

Beyond Projection Visualizations: Not all data and tasks benefit from the exploration of a projection. The projection view can be substituted for other methods of visualizing an overview or summary of the dataset, depending on the analysis tasks. Considering an example in the context of material science, imagine researchers are seeking to develop a new alloy with high strength and corrosion resistance for aerospace applications. They use a connectivity matrix to explore potential candidates, where the matrix includes a variety of known alloys, each characterized by their mechanical and chemical properties. Selecting cells or submatrices are entry points for understanding the properties of nearest neighbors of a selected cell, or defining ranges of desired properties to highlight relevant cells in the matrix. Or in a more general context, one could also imagine a scenario where the projection view might contain geospatial data, and patterns of location could be explored across relevant dimensions. Future work will explore the use of predicates as a bridge between high-dimensional data and alternative low-dimensional representations, beyond just projections.

Beyond Axis-Aligned Dimensions: DimBridge assumes the original (axis-aligned) data dimensions in the data-space visualization as they are the most interpretable. However, this design requirement can be lifted for advanced users who can understand complex data dimensions. Although we illustrated DimBridge using the original data dimensions in Section 8, we observe that the predicate induction could also have been performed using the principal components with the SPLOM showing the data with principal components as the data dimensions. The resulting SPLOM visualization will be less interpretable, but the use of principle components (and possibly other non-linear dimensions) can result in predicates that better fit the user-selected data.

Adapting Predicate Induction: DimBridge's predicate induction engine uses the RPI and Predicate Regression algorithms, representing two extremes of the accuracy/scalability tradeoff. Future work exploring predicate induction algorithms that take a more balanced approach to this tradeoff could be beneficial. Additionally, the predicate induction algorithm can be further tuned to the task of understanding DR results by explicitly accounting for distortions introduced by the DR algorithm. "Distortion aware" predicate induction could more effectively identify patterns in the original dimensions by adjusting for distortions in regions brushed by a user.

9.3 Limitations

The induction engine can generate predicates from any tabular dataset with continuous dimensions and is agnostic to the DR algorithm used for projection. However, we consider the following limitations:

Continuous Dimensions: Both implemented predicate induction algorithms require continuous dimensions. Off-the-shelf feature-engineering techniques can be used to incorporate ordinal and nominal dimensions by numerically encoding them before using the predicate induction engine.

Meaningful Dimensions: Using the original data dimensions in the SPLOM for explanation assumes that these dimensions are semantically meaningful to the user. For data dimensions that are intrinsically not meaningful (e.g., if each pixel of an image represents a dimension), additional processing to extract semantics, such as the use of CLIP [67] in Figure 1 can be effective for explanation.

Quality of DR Results: The quality of the induction engine's explanations depends on the quality of the DR projection. While similar points in the projection are expected to be similar in the original space, DR algorithms can introduce artifacts that disrupt this correspondence. If an artifact is mistaken for a pattern and selected, the induction engine will still return the best matching predicate, even if no good match exists. As discussed in section 6.4, DimBridge addresses this by highlighting selected data points as false positives or false negatives that the predicate fails to match. These situations should only occur if the DR projection produces misleading visual patterns.

Multiple Predicates: Our current implementation of the predicate induction engine returns only the top predicate for visualization by DimBridge. While a given pattern may have multiple plausible explanations in the original dimensions, addressing this involves considerations beyond this work, such as visualizing multiple predicates and maintaining interactivity. Exploring the potential to explain patterns using multiple predicates is an exciting avenue for future work.

10 Conclusion

In this paper we present DimBridge, a system that *bridge*s a projection space with the original data space using first-order predicate logic. DimBridge connects patterns observed in the projection space to the original data space, helping users understand the pattern within the familiar data space. This decreases the likelihood of false discoveries resulting from spurious structure within the projection. DimBridge is agnostic to the projection algorithm, the visualization technique used within the data space, and the predicate induction algorithm itself. We illustrate three showcases of DimBridge within scientific data, motion-capture data, and imagery data. Finally, we evaluated the utility of DimBridge with a domain expert who found the design to be helpful in her workflow.

Acknowledgments

We sincerely thank Vanessa Meschke and Eric Toberer from the NSF Institute for Data Driven Dynamical Design (ID4) for their valuable support and contribution to this work, and Matthew Berger for his contribution to an earlier draft of this paper. This work was supported by National Science Foundation grants (OAC-2118201, IIS-1452977), the Department of Defense (HQ014719C7056) and Merck's Exploratory Science Center.

References

[1] O. AlOmeir, E. Y. Lai, M. Milani, and R. Pottinger. Summarizing provenance of aggregate query results in relational databases. In *Proc. ICDE*, pp. 1955–1960. IEEE, New York, 2021. doi: 10.1109/ICDE51399.2021.00183

[2] E. Amorim, E. Vital Brazil, J. Mena-Chalco, L. Velho, L. G. Nonato, F. Samavati, and M. Costa Sousa. Facing the high-dimensions: Inverse projection with radial basis functions. *Comput. Graphics*, 48:35–47, 2015. doi: 10.1016/j.cag.2015.02.009

[3] D. F. Andrews. Plots of high-dimensional data. *Biometrics*, 28(1):125–136, 1972. doi: 10.2307/2528964

[4] G. Appleby, M. Espadoto, R. Chen, S. Goree, A. C. Telea, E. W. Anderson, and R. Chang. Hypernp: Interactive visual exploration of multidimensional projection hyperparameters. *CGF*, 41(3):169–181, 2022. doi: 10.1111/cgf.14531

[5] M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, 70(7-9):1304–1330, 2007. doi: 10.1016/j.neucom.2006.11.018

[6] M. Aupetit, N. Heulot, and J.-D. Fekete. A multidimensional brush for scatterplot data analytics. In *Proc. VAST*, pp. 221–222. IEEE, New York, 2014. doi: 10.1109/VAST.2014.7042500

[7] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987. doi: 10.1080/00401706.1987.10488204

[8] R. A. Becker, W. S. Cleveland, and A. R. Wilks. Dynamic Graphics for Data Analysis. *Stat. Sci.*, 2(4):355 – 383, 1987. doi: 10.1214/ss/1177013104

[9] J. Bertin. *Semiology of graphics*. University of Wisconsin Press, Madison, Wis., 1983.

[10] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE TVCG*, 17(12):2203–2212, 2011. doi: 10.1109/TVCG.2011.229

[11] J. Bok, B. Kim, and J. Seo. Augmenting parallel coordinates plots with color-coded stacked histograms. *IEEE TVCG*, 28(7):2563–2576, 2020. doi: 10.1109/TVCG.2020.3038446

[12] M. Brehmer, M. Sedlmair, S. Ingram, and T. Munzner. Visualizing dimensionally-reduced data: Interviews with analysts and a characterization of task sequences. In *Proc. BELIV*, pp. 1–8. ACM, New York, 2014. doi: 10.1145/2669557.2669559

[13] Bureau of Labor Statistics. Healthcare Occupations : Occupational Outlook Handbook: : U.S. Bureau of Labor Statistics — bls.gov. `https://www.bls.gov/ooh/healthcare/home.htm`. [Accessed 29-Mar-2023].

[14] F. Cao and E. T. Brown. Dril: Descriptive rules by interactive learning. In *Proc. VIS*, pp. 256–260. IEEE, New York, 2020. doi: 10.1109/VIS47514.2020.00058

[15] M. Cavallo and Ç. Demiralp. Clustrophile 2: Guided visual clustering analysis. *IEEE TVCG*, 25(1):267–276, 2018. doi: 10.1109/TVCG.2018.2864477

[16] M. Cavallo and c. Demiralp. A visual interaction framework for dimensionality reduction based data exploration. In *Proc. CHI Conference*, p. 1–13. ACM, New York, 2018. doi: 10.1145/3173574.3174209

[17] A. Chatzimparmpas, R. M. Martins, and A. Kerren. t-viSNE: Interactive assessment and interpretation of t-sne projections. *IEEE TVCG*, 26(8):2696–2714, 2020. doi: 10.1109/TVCG.2020.2986996

[18] A. Coenen and A. Pearce. Understanding umap. `https://pair-code.github.io/understanding-umap`. Accessed: 2024-06-23.

[19] R. Cutura, S. Holzer, M. Aupetit, and M. Sedlmair. VisCoDeR: A tool for visually comparing dimensionality reduction algorithms. In *Proc. ESANN*, pp. 105–110, 2018.

[20] E. P. dos Santos Amorim, E. V. Brazil, J. Daniels, P. Joia, L. G. Nonato, and M. C. Sousa. ilamp: Exploring high-dimensional spacing through backward multidimensional projection. In *Proc. VAST*, pp. 53–62. IEEE, New York, 2012. doi: 10.1109/VAST.2012.6400489

[21] K. Eckelt, A. Hinterreiter, P. Adelberger, C. Walchshofer, V. Dhanoa, C. Humer, M. Heckmann, C. Steinparz, and M. Streit. Visual exploration of relationships and structure in low-dimensional embeddings. *IEEE TVCG*, 29(7):3312–3326, 2023. doi: 10.1109/TVCG.2022.3156760

[22] G. Ellis. *Cognitive biases in visualizations*. Springer, 2018.

[23] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE TVCG*, 14(6):1141–1148, 2008. doi: 10.1109/TVCG.2008.153

[24] M. Espadoto, G. Appleby, A. Suh, D. Cashman, M. Li, C. Scheidegger, E. W. Anderson, R. Chang, and A. C. Telea. Unprojection: Leveraging inverse-projections for visual analytics of high-dimensional data. *IEEE TVCG*, 29(2):1559–1572, 2023. doi: 10.1109/TVCG.2021.3125576

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

[26] S. Eyuboglu, M. Varma, K. Saab, J.-B. Delbrouck, C. Lee-Messer, J. Dunnmon, J. Zou, and C. Ré. Domino: Discovering systematic errors with cross-modal embeddings. 2022. doi: 10.48550/arXiv.2203.14960

[27] R. Faust, D. Glickenstein, and C. Scheidegger. Dimreader: Axis lines that explain non-linear projections. *IEEE TVCG*, 25(1):481–490, 2018. doi: 10.1109/TVCG.2018.2865194

[28] M. C. Ferreira de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: a survey. *IEEE TVCG*, 9(3):378–394, 2003. doi: 10.1109/TVCG.2003.1207445

[29] M. A. Fisherkeller, J. H. Friedman, and J. W. Tukey. Prim-9: An interactive multi-dimensional data display and analysis system. In *ACM Pacific*. ACM, New York, 1975.

[30] T. Fujiwara, Y.-H. Kuo, A. Ynnerman, and K.-L. Ma. Feature learning for nonlinear dimensionality reduction toward maximal extraction of hidden patterns. In *Proc. PacificVis*, pp. 122–131. IEEE, New York, 2023. doi: 10.1109/PacificVis56936.2023.00021

[31] T. Fujiwara, O. Kwon, and K. Ma. Supporting analysis of dimensionality reduction results with contrastive learning. *IEEE TVCG*, 26(1):45–55, 2020. doi: 10.1109/TVCG.2019.2934251

[32] T. Fujiwara, X. Wei, J. Zhao, and K.-L. Ma. Interactive dimensionality reduction for comparative analysis. *IEEE TVCG*, 28(1):758–768, 2022. doi: 10.1109/TVCG.2021.3114807

[33] S. Garg, J. E. Nam, I. Ramakrishnan, and K. Mueller. Model-driven visual analytics. In *Proc. VAST*, pp. 19–26. IEEE, New York, 2008. doi: 10.1109/VAST.2008.4677352

[34] M. F. Hanafi, A. Abouzied, L. Chiticariu, and Y. Li. Seer: Auto-generating information extraction rules from user-specified examples. In *Proc. CHI*, pp. 6672–6682. ACM, New York, 2017. doi: 10.1145/3025453.3025540

[35] J. Hartigan. Printer graphics for clustering. *J. Stat. Comput. Simul.*, 4(3):187–213, 1975. doi: 10.1080/00949657508810123

[36] H. Hauser, F. Ledermann, and H. Doleisch. Angular brushing of extended parallel coordinates. In *Proc. InfoVis*, pp. 127–130. IEEE, New York, 2002. doi: 10.1109/INFVIS.2002.1173157

[37] C. Healey and J. Enns. Attention and visual memory in visualization and computer graphics. *IEEE TVCG*, 18(7):1170–1188, 2011. doi: 10.1109/TVCG.2011.127

[38] J. Heinrich and D. Weiskopf. State of the Art of Parallel Coordinates. In M. Sbert and L. Szirmay-Kalos, eds., *Eurographics 2013 - State of the Art Reports*, pp. 95–116. The Eurographics Association, 2013. doi: /10.2312/conf/EG2013/stars/095-116

[39] N. Helwig and E. Hsiao-Wecksler. Multivariate Gait Data. UCI Machine Learning Repository, 2022.

[40] N. E. Helwig, S. Hong, E. T. Hsiao-Wecksler, and J. D. Polk. Methods to temporally align gait cycle data. *J. Biomech.*, 44(3):561–566, 2011. doi: 10.1016/j.jbiomech.2010.09.015

[41] N. E. Helwig, K. A. Shorter, P. Ma, and E. T. Hsiao-Wecksler. Smoothing spline analysis of variance models: A new tool for the analysis of cyclic biomechanical data. *J. Biomech.*, 49(14):3216–3222, 2016. doi: 10.1016/j.jbiomech.2016.07.035

[42] P. Hoffman and G. Grinstein. A survey of visualizations for high-dimensional data mining. *Information Visualization in Data Mining and Knowledge Discovery*, 104:47–82, 2002.

[43] A. Inselberg. The plane with parallel coordinates. *Visual Comput.*, 1(2):69–91, 1985. doi: 10.1007/BF01898350

[44] H. Jeon, M. Aupetit, S. Lee, H.-K. Ko, Y. Kim, and J. Seo. Distortion-aware brushing for interactive cluster analysis in multidimensional projections. *arXiv preprint arXiv:2201.06379*, 2022. doi: 10.48550/arXiv.2201.06379

[45] P. Joia, F. Petronetto, and L. Nonato. Uncovering representative groups in multidimensional projections. *CGF*, 34(3):281–290, 2015. doi: 10.1111/cgf.12640

[46] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila. Alias-free generative adversarial networks. In *Proc. NeurIPS*, pp. 852–863. Curran Associates Inc., Red Hook, NY, 2021. doi: 10.5555/3540261.3540327

[47] D. Kobak and P. Berens. The art of using t-sne for single-cell transcriptomics. *Nat. Commun.*, 10(5416), 2019. doi: 10.1038/s41467-019-13056-x

[48] J. B. Kruskal and M. Wish. *Multidimensional scaling*. Sage, Thousand Oaks, CA, 1978. doi: 10.4135/9781412985130

[49] B. C. Kwon, B. Eysenbach, J. Verma, K. Ng, C. De Filippi, W. F. Stewart, and A. Perer. Clustervision: Visual supervision of unsupervised clustering.

IEEE TVCG, 24(1):142–151, 2018. doi: 10.1109/TVCG.2017.2745085

[50] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring N-dimensional databases. In *Proc. VIS*, pp. 230—237. IEEE, New York, 1990. doi: 10.1109/VISUAL.1990.146386

[51] S. Lespinats and M. Aupetit. Checkviz: Sanity check and topological clues for linear and non-linear mappings. *CGF*, 30(1):113–125, 2011. doi: 10.1111/j.1467-8659.2010.01835.x

[52] J. Li and C.-q. Zhou. Incorporation of human knowledge into data embeddings to improve pattern significance and interpretability. *IEEE TVCG*, 29(1):723–733, 2023. doi: 10.1109/TVCG.2022.3209382

[53] H. Liao, Y. Wu, L. Chen, and W. Chen. Cluster-based visual abstraction for multivariate scatterplots. *IEEE TVCG*, 24(9):2531–2545, 2018. doi: 10.1109/TVCG.2017.2754480

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

[55] W. E. Marcílio-Jr and D. M. Eler. Explaining dimensionality reduction results using shapley values. *ESWA*, 178:115020, 2021. doi: 10.1016/j.eswa.2021.115020

[56] W. E. Marcílio-Jr, D. M. Eler, and R. E. Garcia. Contrastive analysis for scatterplot-based representations of dimensionality reduction. *Comput. Graphics*, 101:46–58, 2021. doi: 10.1016/j.cag.2021.08.014

[57] A. R. Martin and M. O. Ward. High dimensional brushing for interactive exploration of multivariate data. In *Proc. VIS*, p. 271. IEEE, New York, 1995. doi: 10.5555/832271.833844

[58] R. M. Martins, D. B. Coimbra, R. Minghim, and A. Telea. Visual analysis of dimensionality reduction quality for parameterized projections. *Comput. Graphics*, 41:26–42, 2014. doi: 10.1016/j.cag.2014.01.006

[59] L. McInnes, J. Healy, N. Saul, and L. Grossberger. Umap: Uniform manifold approximation and projection. *JOSS*, 3(29):861, 2018. doi: 10.21105/joss.00861

[60] Y. Ming, H. Qu, and E. Bertini. Rulematrix: Visualizing and understanding classifiers with rules. *IEEE TVCG*, 25(1):342–352, 2019. doi: 10.1109/TVCG.2018.2864812

[61] B. Montambault, C. D. Brumar, M. Behrisch, and R. Chang. Pixal: Anomaly reasoning with visual analytics, 2022. doi: 10.48550/arXiv.2205.11004

[62] M. Mustafa. Diabetes prediction dataset. `https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset`, 2023. Accessed: March 19 2024.

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

[64] A. Novick, Q. Nguyen, R. Garnett, E. Toberer, and V. Stevanović. Simulating high-entropy alloys at finite temperatures: An uncertainty-based approach. *Phys. Rev. Mater.*, 7(6):063801, 2023. doi: 10.1103/PhysRevMaterials.7.063801

[65] L. Pagliosa, P. Pagliosa, and L. G. Nonato. Understanding attribute variability in multidimensional projections. In *Proc. SIBGRAPI*, pp. 297–304. IEEE, New York, 2016. doi: 10.1109/SIBGRAPI.2016.048

[66] K. Pearson. LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. doi: 10.1080/14786440109462720

[67] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *Proc. ICML*, pp. 8748–8763. PMLR, 2021. doi: 10.48550/arXiv.2103.00020

[68] P. E. Rauber, R. R. O. d. Silva, S. Feringa, M. E. Celebi, A. X. Falcão, and A. C. Telea. Interactive Image Feature Selection Aided by Dimensionality Reduction. In E. Bertini and J. C. Roberts, eds., *Proc. EuroVA*. The Eurographics Association, 2015. doi: 10.2312/eurova.20151098

[69] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-precision model-agnostic explanations. In *Proc. AAAI*, pp. 1527–1535. AAAI Press, Washington, DC, 2018. doi: 10.5555/3504035.3504222

[70] B. Ristevski and M. Chen. Big data analytics in medicine and healthcare. *Journal of Integrative Bioinformatics*, 15(3):20170030, 2018. doi: doi:10.1515/jib-2017-0030

[71] R. C. Roberts, R. S. Laramee, G. A. Smith, P. Brookes, and T. D'Cruze. Smart brushing for parallel coordinates. *IEEE TVCG*, 25(3):1575–1590, 2019. doi: 10.1109/TVCG.2018.2808969

[72] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim. Visual interaction with dimensionality reduction: A structured literature analysis. *IEEE TVCG*, 23(1):241–250, 2016.

[73] A. Sarikaya and M. Gleicher. Scatterplots: Tasks, data, and designs. *IEEE TVCG*, 24(1):402–412, 2017. doi: 10.1109/TVCG.2017.2744184

[74] M. Sedlmair, T. Munzner, and M. Tory. Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE TVCG*, 9:2634–2643, 2013. doi: 10.1109/TVCG.2013.153

[75] J. Seo and B. Shneiderman. A rank-by-feature framework for interactive exploration of multidimensional data. *Inf. Visualization*, 4(2):96–113, 2005. doi: 10.1057/palgrave.ivs.9500091

[76] K. A. Shorter, J. D. Polk, K. S. Rosengren, and E. T. Hsiao-Wecksler. A new approach to detecting asymmetries in gait. *Clin. Biomech.*, 23(4):459–467, 2008. doi: 10.1016/j.clinbiomech.2007.11.009

[77] J.-T. Sohns, M. Schmitt, F. Jirasek, H. Hasse, and H. Leitte. Attribute-based explanation of non-linear embeddings of high-dimensional data. *IEEE TVCG*, 28(1):540–550, 2021. doi: 10.1109/TVCG.2021.3114870

[78] A. Srinivasan, H. Park, A. Endert, and R. C. Basole. Graphiti: Interactive specification of attribute-based edges for network modeling and visualization. *IEEE TVCG*, 24(1):226–235, 2017. doi: 10.1109/TVCG.2017.2744843

[79] J. Stahnke, M. Dörk, B. Müller, and A. Thom. Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions. *IEEE TVCG*, 22(1):629–638, 2015. doi: 10.1109/TVCG.2015.2467717

[80] G. Sun, S. Zhu, Q. Jiang, W. Xia, and R. Liang. Evosets: tracking the sensitivity of dimensionality reduction results across subspaces. *IEEE TBDATA*, 8(6):1566–1579, 2021. doi: 10.1109/TBDATA.2021.3079200

[81] D. A. Szafir, S. Haroz, M. Gleicher, and S. Franconeri. Four types of ensemble coding in data visualizations. *JOV*, 16(5):11–11, 2016. doi: 10.1167/16.5.11

[82] A. Tatu, F. Maaß, I. Färber, E. Bertini, T. Schreck, T. Seidl, and D. Keim. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In *Proc. VAST*, pp. 63–72, 2012. doi: 10.1109/VAST.2012.6400488

[83] A. Tatu, L. Zhang, E. Bertini, T. Schreck, D. Keim, S. Bremm, and T. von Landesberger. Clustnails: Visual analysis of subspace clusters. *Tsinghua Sci. Technol.*, 17(4):419–428, 2012. doi: 10.1109/TST.2012.6297588

[84] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *J. Mach. Learn. Res.*, 9(86):2579–2605, 2008.

[85] J. Wagemans, J. Feldman, S. Gepshtein, R. Kimchi, J. R. Pomerantz, P. A. Van der Helm, and C. Van Leeuwen. A century of gestalt psychology in visual perception: Ii. conceptual and theoretical foundations. *Psychological Bulletin*, 138(6):1218–1252, 2012. doi: 10.1037/a0029334

[86] T. Wang, Y. Tao, A. Gilad, A. Machanavajjhala, and S. Roy. Explaining differentially private query results with DPXPlain. *Proc. VLDB Endow.*, 16(12):3962–3965, 2023. doi: 10.14778/3611540.3611596

[87] M. Wattenberg, F. Viégas, and I. Johnson. How to use t-SNE effectively. *Distill*, 2016. doi: 10.23915/distill.00002

[88] J. Wenskovitch, I. Crandell, N. Ramakrishnan, L. House, and C. North. Towards a systematic combination of dimension reduction and clustering in visual analytics. *IEEE TVCG*, 24(1):131–141, 2017. doi: 10.1109/TVCG.2017.2745258

[89] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Proc. InfoVis*, pp. 157–164. IEEE Computer Society, IEEE, New York, 2005. doi: 10.1109/INFVIS.2005.1532142

[90] E. Wu and S. Madden. Scorpion: Explaining away outliers in aggregate queries. *Proc. VLDB Endow.*, 6(8):553–564, June 2013. doi: 10.14778/2536354.2536356

[91] J. Xia, L. Huang, W. Lin, X. Zhao, J. Wu, Y. Chen, Y. Zhao, and W. Chen. Interactive visual cluster analysis by contrastive dimensionality reduction. *IEEE TVCG*, 29(1):734–744, 2023. doi: 10.1109/TVCG.2022.3209423

[92] L. Xiao, J. Gerth, and P. Hanrahan. Enhancing visual analysis of network traffic using a knowledge representation. In *Proc. VAST*, pp. 107–114. IEEE, New York, 2006. doi: 10.1109/VAST.2006.261436

[93] J. Yuan, B. Barr, K. Overton, and E. Bertini. Visual exploration of machine learning model behavior with hierarchical surrogate rule sets. *IEEE TVCG*, 30(2):1470–1488, 2024. doi: 10.1109/TVCG.2022.3219232

[94] J. Yuan, J. Vig, and N. Rajani. isea: An interactive pipeline for semantic error analysis of nlp models. In *Proc. IUI*, p. 878–888. ACM, New York, 2022. doi: 10.1145/3490099.3511146

[95] F. C. Zegarra, J. C. Carbajal Ipenza, B. Omidvar-Tehrani, V. P. Moreira, S. Amer-Yahia, and J. L. Comba. Visual exploration of rating datasets and user groups. *Future Gener. Comput. Syst.*, 105:547–561, 2020. doi: 10.1016/j.future.2019.12.011