Feature Compass: Explaining 2D Projections with Feature Gradients

Category: Research

Paper Type: technique

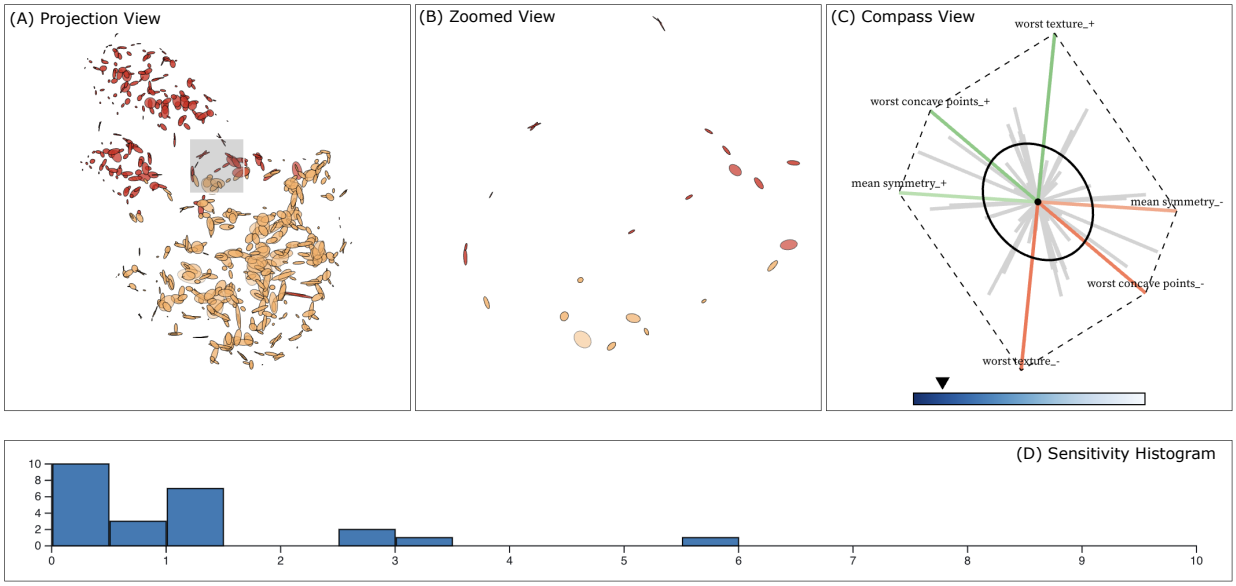

Fig. 1: (A) presents the t-SNE projection of the breast cancer dataset, where each patient instance is represented as a compass ellipse. Red ellipses represent the Malignant label while yellow ellipses represent the Benign label. Ellipses with higher sensitivity have larger size and lower opacity; (B) presents the Zoomed View of the brushed area in the Projection View, revealing the size and anisotropy of points on the boundary between two distinct breast cancer labels; (C) provides a detailed Compass View indicating that *mean symmetry*, *worst concave points*, and *worst texture* significantly influence the positioning of the selected ellipses on the boundary between two different breast cancer labels. Additionally, a sensitivity meter displays the sensitivity level of the selected points, from lowest sensitivity on the left to highest on the right; (D) presents a sensitivity histogram for the selected points.

Abstract—Projections generated by dimension reduction (DR) algorithms effectively expose the underlying structure of high-dimensional data, but interpreting the influence of individual features within 2D projections remains challenging. More specifically, existing methods for visualizing feature behaviors in 2D projections either rely on approximations within neighborhood points that prevent point-wise analysis or lack scalability by requiring one plot per feature. In this paper, we propose Feature Compass, a novel visualization that provides precise point-wise feature sensitivity analysis without approximation and delivers a scalable overview of features within a single plot. Feature Compass enables analysis of point-wise feature influences through its compass view, which provides a novel visual summary of feature sensitivity. By integrating multiple visual navigation components, Feature Compass also delivers a cohesive interface that includes an overview of point placement sensitivities directly within the DR projection. We evaluate the effectiveness of Feature Compass through a case study on a synthetic dataset and two additional case studies on real-world datasets—the Iris dataset and the Breast Cancer dataset. Our results demonstrate that Feature Compass effectively uncovers feature sensitivities, thereby enhancing the interpretability of 2D projection plots.

Index Terms—Dimension reduction, visual annotation, feature vectors, 2D projection plots, sensitivity analysis

1 Introduction

Dimension reduction (DR) techniques are widely used to analyze high-dimensional data and understand the relationships between data points. They are used across a wide range of domains and on a variety of data types, including tabular, image, and text data [18, 26, 32]. DRs, particularly nonlinear DRs such as t-SNE [42], MDS [23], and UMAP [28], provide well-organized low-dimensional projections of complex high-dimensional data. However, because of their opaque reasoning and subsequent barrier to interpretability, DRs give no context as to why projections are organized as they were.

To explain the spatial positioning of data points in DR projections, many approaches have been proposed to help users interpret low-dimensional embeddings, relate them back to the original data features, and interactively explore and explain the results. Among these, feature-based explanation is an effective way to explain DR projections. These methods focused on making DR projections transparent by bridging the gap between abstract low-dimensional projections and the underlying high-dimensional data features. Key examples include DimReader's perturbation-based axes [12], interactive tools like InterAxis and AxiSketcher that put users in control of feature-weighted projections [21, 24], and explanation frameworks like ClusterShapley and SLISEMAP that leverage feature-attribution techniques (Shapley values, local linear models) to clarify DR outcomes [4, 27]. Together, these works represent a toolbox for explaining spatial positioning in DR projections either by annotating and analyzing a given projection with feature information, or by building interpretability into the projection process itself so that the low-dimensional axes and groupings directly correspond to explainable feature combinations. Such methods help ensure that when high-dimensional data are reduced to a 2D plot, we can

trace why points are where they are in terms of the original features that define the data. Despite the emergence of feature-based explanations, there remains a significant gap in achieving scalable interpretability of feature effects in DR projections because existing works either fail to offer **point-wise feature explanation** or require **one plot for each feature** to inspect.

In this work, we present Feature Compass (Fig. 1), a visualization prototype that supports scalable interpretability of feature effects in DR projections through feature sensitivity analysis. Sensitivity analysis explains a model's prediction by evaluating its locally derived gradient [35]. Building on this idea, we represent feature gradients as vectors on each point of a 2D DR projection. These vectors, referred to as **compass needles**, eliminate the need to toggle between disjointed plots when examining different features. To visually encode compass needle attributes, we introduce the **compass baseplate**, which replaces traditional circular markers in the DR plot with ellipses that capture both the orientation and magnitude of these vectors. Together with a dynamic sensitivity meter, the compass needles and compass baseplate form the **compass view** (Fig. 1-C), serving as the central component of Feature Compass. To enable scalable, simultaneous analysis of feature sensitivity, we unify the compass view with a **projection view**, **zoomed view**, and **sensitivity histogram**(Fig. 1-A,B,D). In the projection view, points are depicted as baseplate ellipses rather than circles, while the zoomed view provides a closer look at a selected region within the projection. The sensitivity histogram offers a quantitative summary of feature sensitivity across points in the zoomed area. We first validate Feature Compass using a synthetic dataset of two interlocked rings, demonstrating its capacity to reveal feature sensitivity within DR projections. We then apply Feature Compass to two real-world datasets to further illustrate how it can enhance the interpretability of DR results. In summary, our contributions include:

- **Feature Compass Visualization**:
  - Compass Needles: Provide novel visual summary of point-wise feature sensitivity via gradient vectors;
  - Compass Baseplate: Introduce innovative visual encoding of vector attributes, which encodes both the orientation and magnitude of gradient vectors;
- **Feature Compass Interface**: Integrates a cohesive workflow, enabling simultaneous analysis of feature sensitivity within a single 2D interface;
- **Feature Compass Evaluation**: Demonstrate the effectiveness of Feature Compass through three case studies, including synthetic and real-world datasets.

2 Related Work

2.1 Dimension Reduction

The overarching goal of DR algorithms is to project high-dimensional data into a low-dimensional space while preserving the structures such as outliers and clusters that exist in the original high-dimensional data [25]. Detailed surveys of DR algorithms can be found in the literature [11, 15, 16, 45]. These algorithms are often characterized and presented as two classes: linear and nonlinear algorithms. Mathematically, the difference between these classes is the underlying structure of the topological spaces or manifolds that each algorithm family is capable of learning, where linear spaces are a subset of nonlinear spaces.

More important for this work is the human interpretability of these algorithmic classes. Linear algorithms produce a linear mapping or transformation to convert the high-dimensional data into its lower-dimensional embedding [8]. Because of this linear relationship, it is relatively straightforward to explain the resulting projection in terms of the original high-dimensional data. For example, PCA [31] operates by determining the axes of maximum variance in the collection of observations. Given a high-dimensional dataset, these PCA axes can be represented by applying straightforward affine transformations to the high-dimensional data. Other linear algorithms such as Factor Analysis [20], Projection Pursuit [17], and Classical MDS [40] have similar mechanisms for embedding data using linear combinations of high-dimensional data.

In contrast, nonlinear DR algorithms rely upon more complex mathematical processes to embed data into low-dimensional projections. These nonlinear embeddings often better represent the high-dimensional spaces (for example, recently seen in fields such as kinematics [33] and gerontology [22]). Common linear algorithms such as PCA and MDS have even been extended to nonlinear variants [36, 43]. However, these nonlinear algorithms are accompanied by significant interpretability challenges. Perhaps the most popular nonlinear DR algorithm, t-SNE [42], relies heavily on the appropriate selection of algorithmic parameters, and poor parameter selection can lead to visual interpretations such as clusters in the embedding that do not exist in the source data [44]. Our goal with this work is to further support the interpretability of these nonlinear embeddings.

2.2 Feature-Based Explanation of DR Projections

Challenges with the interpretation of nonlinear DR embeddings are a well-recognized problem [2, 6, 41], leading to a variety of design principles and guidance to drive the appropriate usage of these algorithms [5, 37, 38, 44]. Sedlmair specifically notes the "nonlinear unmapping gap" as an unsolved challenge, with the goal of understanding how the dimensions in nonlinear embeddings map to the original high-dimensional data [37].

In recent years, feature-based explanations of DR projections have been of increasing focus because they create a tangible link between the low-dimensional visualization and the original high-dimensional data features [10, 12, 29, 30, 39]. Stahnke *et al.* [39] first introduces "probing" interactions to interrogate why points are positioned as they are in a DR scatterplot. The system reveals which features most distinguish a pair of points or a cluster from others, effectively letting users *"visualize the influence of data dimensions on the projection."* Following similar "probing" idea, DimReader [12] recovers "readable axes"in an existing non-linear projection by analyzing infinitesimal perturbations of the original features. Analogous to standard x/y axes, these generalized axis lines indicate the direction along which that features varies. Hypertrix [34] applies a local linear transformation to a neighborhood of data points to visualize the distortions introduced by DR algorithms. Similarly, Feature Clock [30] leverages neighborhood information to solve a linear regression, which they define as feature contribution. By representing feature contributions as vectors, Feature Clock provides a visual interpretation of feature impacts. Additionally, Montambault *et al.* [29] propose DimBridge, an interactive tool that lets users select visual patterns in a projection and then finds human-readable rules in the original feature space that correspond to those patterns. For example, a user might highlight an interesting cluster or draw a shape through points; DimBridge then uses first-order predicate logic to identify a combination of feature conditions that covers those points. These post hoc visualization designs demonstrate how linking scatter plots with feature statistics or logical conditions helps analysts interpret clusters and patterns in terms of original variables.

In addition to the post hoc visualization systems, techniques that integrate interpretability into DR are also effective in providing feature-based reasoning in DR projections. For example, Kim *et al.* [21] proposes InterAxis, a visual analytics method that lets users define custom projection axes in terms of original features. Similarily, AxiSketcher [24] extends the idea of user-defined axes to nonlinear mappings. In AxiSketcher, users sketch a free-form line through the scatterplot (by clicking through a sequence of points of interest), indicating a desired curved trend among the data. The system then fits a nonlinear model that maps the high-dimensional data to a single axis following that sketched curve. The result is a nonlinear axis (imagine a curved dimension through the cloud of points) that aligns with a complex pattern the user perceives.

While our work builds upon these advancements, we hope to better enhance projections to effectively communicate attribute information and provide deeper insights into high-dimensional data structures.

3 Design Goals

Limitation of Existing Methods. After a thorough review of existing methods, we identified two major limitations that hinder their

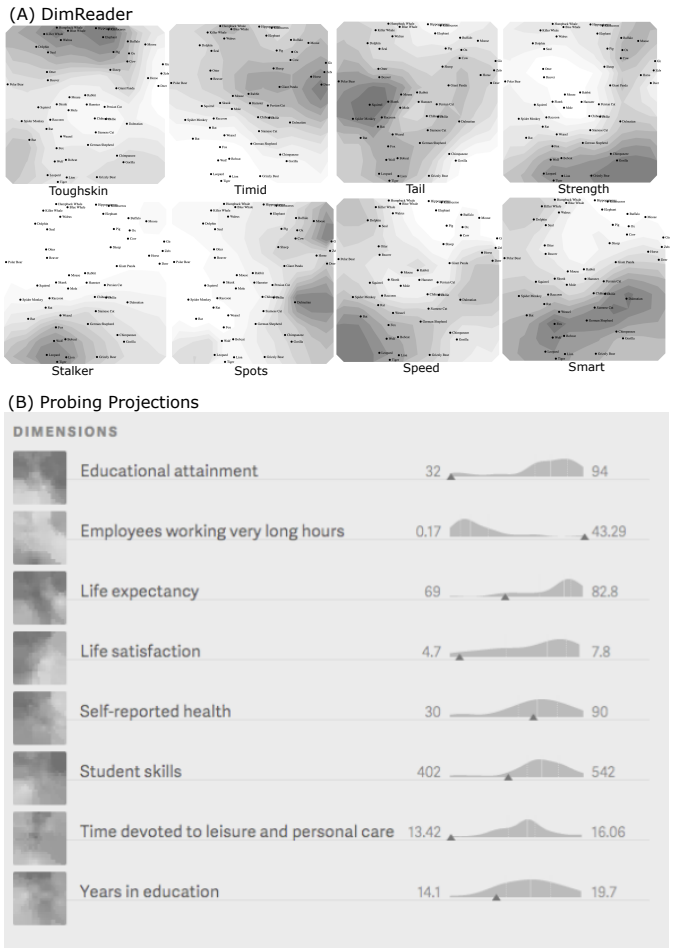

Fig. 2: Examples of existing approaches that require a separate visualization for each feature.

effectiveness in interpreting DR results. First, many existing methods rely heavily on neighborhood information to approximate feature impacts on the position of points in 2D projections ([30, 34]). While this approach provides a straightforward means of analyzing the projected 2D view, it introduces inaccuracies due to its reliance on local proximity, which may fail to capture the point-wise influence of individual features or the global structure of the dataset. Second, most visual designs require generating multiple plots to explore feature behaviors, resulting in visual redundancy and scalability challenges when dealing with high-dimensional datasets. This thus reintroduces the problem of dimensionality that DR was intended to eliminate. Fig. 2-(A) shows axes lines generated from DimReader [12], where eight separate plots are required to inspect the tangent map of eight different features. And Fig. 2-(B) shows Probing Projections [39], which also requires one plot per dimension.

Design Goals. Based on these limitations, we realized the following design goals to enable more precise and scalable analyses in feature-based explanation systems for DR projections.

DG1: Enable Point-wise measurements with multilevel visualization of Feature Impacts. A critical requirement for effective DR analysis is the ability to calculate precise point-wise measurements of feature impacts, while also supporting global analyses to provide comprehensive insights into DR results. This entails computing interpretable metrics for each feature at every point in the projection and aggregating them at multiple levels of granularity—ranging from individual points and clusters to the entire dataset. While point-wise measurements furnish a solid foundation for interpreting feature behaviors, the system also ensures flexibility in exploration and fosters an understanding of feature interactions across the overall data structure.

DG2: Scalable to Any Number of Features. Scalability to any number of features is essential in a feature-based explanation system because it ensures that the system remains effective even when dealing with datasets with large number of features. This can be achieved through advanced visual encoding techniques, such as summarizing feature distributions with concise yet informative visual elements and employing opacity or color schemes to highlight key patterns. By simplifying visual representations, systems can ensure clarity and help interpret key feature patterns.

These design goals form the foundation for the development of systems that address the limitations of current methods, offering deeper insights into the relationships between high-dimensional features and their 2D representations. The subsequent sections will elaborate on the methodologies and visualization design employed to achieve these objectives.

4 Gradient Computation

First, our method requires point-wise measurement of feature sensitivity. Sensitivity analysis is defined as a method to explain a model's prediction by evaluating its locally derived gradient [35]. This approach assumes that the most relevant input features are those to which the model output is most sensitive. In this paper, we use regular letters to denote scalars, boldface lowercase letters to denote vectors, and boldface uppercase letters to denote matrices. Given a model function f, the importance of each input variable i is quantified as:

$$R_i = \| \frac{\partial}{\partial x_i} f(\mathbf{x}) \|. \tag{1}$$

In this paper, we adopt the same terminology to assess the behavior of features in the context of DR. Specifically, for a given DR mapping $f : \mathbb{R}^n \to \mathbb{R}^2$ that projects a high-dimensional point $\mathbf{p}_i$ to a 2D point $\mathbf{v}_i$, where $v_{i,x} = f_x(p_i)$ and $v_{i,y} = f_y(p_i)$, and a feature indexed by k, we define the sensitivity of $\mathbf{v}_i$ to feature k as:

$$\mathbf{g}_{i,k} = \Big[\frac{\partial f_x(p_i)}{\partial p_{i,k}}, \frac{\partial f_y(p_i)}{\partial p_{i,k}} \Big]. \tag{2}$$

Subsequently, if the dataset has d features, we generate a $2n \times d$ Jacobian matrix,

$$\mathbf{J} = \begin{bmatrix} \mathbf{g}_{1,1}^T & \cdots & \mathbf{g}_{1,d}^T \\ \vdots & \ddots & \vdots \\ \mathbf{g}_{n,1}^T & \cdots & \mathbf{g}_{n,d}^T \end{bmatrix}. \tag{3}$$

The Jacobian matrix captures the response of the 2D coordinates to variations in the input features.

Implementation To calculate $\mathbf{J}$, we use the built-in automatic differentiation capabilities of deep learning libraries, such as Torch [7] and TensorFlow [1]. Specifically, we modify existing implementations of DR methods to allow Torch tensors as input. Then, as the DR operates on the tensors, Torch constructs a computation graph to record all operations modifying the tensor. After projection, for each projected point $\mathbf{v}_i$, we use Torch to propagate backward through the computation graph to $\mathbf{p}_i$, calculating $[\mathbf{g}_{i,1}, \mathbf{g}_{i,1}, \ldots, \mathbf{g}_{i,d}]$. This requires two passes: one to calculate the gradients with respect to $\mathbf{v}_{i,x}$ and one with respect to $\mathbf{v}_{i,y}$. To calculate the entirety of $\mathbf{J}$, it requires $2n$ backwards passes through the computation graph, yielding a runtime of $O(2n \cdot b)$ where b is the time it takes to propagate one output point back to its high-dimensional input through the computation graph.

5 Feature Compass Visualization

In this section, we present the formulation of the compass, which serves as the fundamental visual representation of the system. The compass view imitates the traditional navigation compass, which comprises two primary components: the compass needle and the compass baseplate. In our system, the compass is composed of multiple feature needles, an elliptical compass baseplate, and a dynamic sensitivity meter (Fig. 1-C).

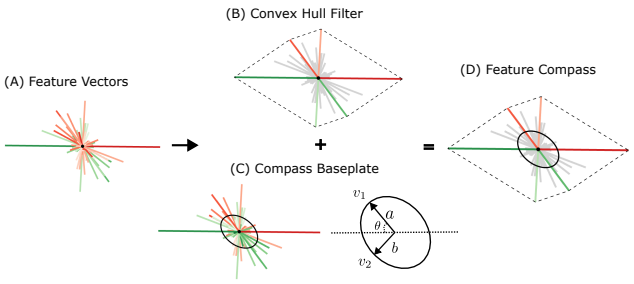

Fig. 3: This figure illustrates the formulation of the compass in Feature Compass. (A) depicts all compass needles; (B) demonstrates the application of a convex hull filter to identify significant features; (C) shows the generation of an ellipse encoding the gradient vector's directional and magnitude information; (D) presents the final compass, integrating the convex hull filter and ellipse.

5.1 Compass Needle

Recall, for each projected point $\mathbf{v}_i$ and each feature k, we have $\mathbf{g}_{i,k}$ which contains the gradients of $\mathbf{v}_i$ with respect to feature k (see Eq. (2)). Together, these gradients tells us in what direction and how strongly the given feature will pull the projected point in the 2D projection space. As such, the natural visual encoding for this gradient is a vector or arrow. We use vectors, referred to as **compass needles**, as the base visual encoding for the gradients, see Fig. 3-A.

Needle Color The needle serves as a visual representation of the gradients, which, by default, indicate the sensitivity to the positive changes in the corresponding feature. However, in real-world scenarios, changes in feature values are often bi-directional. To address this, we use a color-coding scheme where **Green** represents positive changes and **Red** represents negative changes. This color representation allows users to quickly discern the direction of feature influence.

Needle Opacity To strengthen the connection between the projection view and the underlying raw data values, we apply a sequential color scale to the raw data. For a given high-dimensional point $\mathbf{p}_i$ and a high-dimensional feature k, the opacity of its corresponding compass needle is determined by the magnitude of its raw value $p_{i,k}$. Specifically, needles are rendered with high opacity when $p_{i,k}$ is large, and with low opacity when $p_{i,k}$ is small. Paired with needles, this design choice helps to visually indicate relationships between feature values and feature sensitivities.

Needle Filter With the capability to calculate gradients, Feature Compass can generate compass needles for a given point, as shown in Fig. 3-A. While this provides an interpretable depiction of feature influences, it can result in visual clutter when the number of vectors is too large. To address this, Feature Compass incorporates a convex hull filter to identify and highlight the most significant compass needles. The convex hull filter operates by applying a convex hull to the endpoints of all compass needles. Specifically, the convex hull of a set of points is defined as the smallest convex set that contains all points ([9]-Chapter 1). In the context of Feature Compass, for each point $\mathbf{v}_i$, we compute the convex hull of the endpoints of its compass needles and select those needles whose endpoints lie on the convex hull's boundary as significant. These needles represent features with a substantial impact on the position of the point, both in terms of magnitude and direction. This process is visualized in Fig. 3-B. By emphasizing these boundary needles, Feature Compass remains both clear and informative, regardless of how many features the dataset contains(satisfying **DG2**).

5.2 Compass Baseplate

While the convex hull filter provides a clear view for individual points, the selected compass needles can still make interpreting all points in a single projection view challenging. To address this, Feature Compass generates an ellipse as a representation for a point instead of using traditional circles in standard scatter plots. The ellipse captures both the orientation and the spread of compass needles at each point, offering a compact visual summary of compass needles.

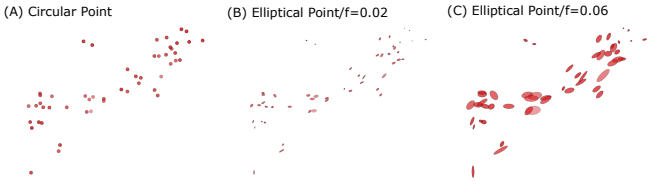

Fig. 4: This figure illustrates three projection views. (A) all points are depicted as circles. (B) points are represented as baseplate ellipses with a desired fraction of 0.02. (C) points are elliptical with the desired fraction of 0.06.

To analyze the distribution and orientation of a set of two-dimensional compass needles, we generated an ellipse based on their covariance matrix. This provides a concise representation of the data's variability and principal directions. The method proceeds as follows: Let the dataset $\mathbf{X} \in \mathbb{R}^{n\times 2}$ represent the n compass needles, where each row corresponds to the elements of a compass needle. First, the covariance matrix Σ of the data is computed as:

$$\Sigma = \frac{1}{n-1}(\mathbf{X}-\bar{\mathbf{X}})^\top(\mathbf{X}-\bar{\mathbf{X}}), \tag{4}$$

where $\bar{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{X}_i$. Then the eigenvalue decomposition of Σ yields eigenvalues λ_1 and λ_2 where $\lambda_1 \geq \lambda_2$, and corresponding eigenvectors $\mathbf{e}_1$ and $\mathbf{e}_2$, which correspond to the principal directions of the data's variability. The eigenvalues quantify the variance along these directions. Using these components, we compute the parameters of the ellipse, also referred to as Compass Baseplate, as shown in Fig. 3-C:

- The **semi-major axis** a is given by the square root of the largest eigenvalue λ_1, and the **semi-minor axis** b is given by the square root of the second eigenvalue λ_2:

$$a = \sqrt{\lambda_1}, b = \sqrt{\lambda_2}. \tag{5}$$

- The **rotation angle** θ of the ellipse is the angle between the horizontal axis and the eigenvector $\mathbf{e}_1$:

$$\theta = \arctan 2(e_{1,y}, e_{1,x}), \tag{6}$$

where $e_{1,x}$ and $e_{1,y}$ are the elements of $\mathbf{e}_1$.

Baseplates in DR plots As shown in Fig. 4, circular points in DR plots are isotropic, lacking orientation information. Ellipses, however, explicitly encode directional bias via their axes. By replacing circular markers with ellipses, Feature Compass enhances interpretability by encoding directional variance and multivariate relationships inherent to the data points. To better facilitate scalability of ellipses, we formalize a scale mechanism to adjust the scale factor using a desired fraction. Let $frac$ denote the desired fraction, P be the maximal axis-aligned span of the data, which specifies the largest scaled gradient should occupy:

$$P = max(max(v_{i,x}) - min(v_{i,x}), max(v_{i,y}) - min(v_{i,y})), \tag{7}$$

and G_{max} represents the maximum gradient magnitude, computed as the supremum of the Euclidean norms of all compass needles:

$$G_{max} = sup||\nabla||_2. \tag{8}$$

we can compute the scale factor $s \in \mathbb{R}^+$ as:

$$s = \frac{frac \cdot P}{G_{max}}, \tag{9}$$

This ensures the largest compass needle, when scaled by s, spans $frac \cdot P$, directly coupling its visual length to the dataset's spatial

extent. A larger $frac$ amplifies the vectors relative to P (Fig. 4C), while a smaller $frac$ attenuates them, preserving proportional visibility across varying data scales (Fig. 4B).

The elliptical visual encoding of Compass Baseplate can also offer insights into the information about anisotropy and sensitivity. Ellipses quantify variability through their semi-axis lengths, which correspond to the spread of compass needles along orthogonal directions. The semi-major axis represents the maximum variance (dominant feature influence), while the semi-minor axis captures secondary variance. This dual-axis encoding distinguishes regions of high anisotropy (elongated ellipses, indicating strong directional sensitivity, as shown in Fig. 5A&D) from isotropic regions (near-circular ellipses, suggesting uniform influence, as shown in Fig. 5B&C). Circles, which collapse variance into a single radius, fail to convey such nuanced distinctions.

5.3 Sensitivity Meter

To enhance interpretability of feature influence dynamics, we introduce a sensitivity meter integrated into the compass view. This component dynamically quantifies and visualizes the sensitivity level of selected points, providing immediate feedback on how strongly high-dimensional features govern their positions in the projection. In this work we use the area of the compass baseplate to indicate the sensitivity level of each compass. For a given domain of compass areas $[area_{min}, area_{max}]$, we define a sensitivity meter in which the left end (highest opacity) represents the lowest sensitivity level and the right end (lowest opacity) represents the highest. The meter provides a continuous scale between low-sensitivity regions (e.g., stable points with minimal positional variability, as in Fig. 5C&D) and high-sensitivity regions (e.g., points with strong directional feature influence, as in Fig. 5A&B). By correlating sensitivity levels with the compass's elliptical geometry and needle orientations, users can choose to identify stable areas (low sensitivity) or focus on high-sensitivity points. The meter enables larger visualization of needles and baseplates for points that have low sensitivity and would otherwise be too small to discern. This quantitative feedback bridges the gap between the compass's visual encoding (baseplate, needles) and actionable insights.

5.4 Aggregated Compass

To visualize the compass for a set of projected 2D points $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\} \subset \mathbb{R}^2$, we first introduce an aggregated point $\hat{\mathbf{v}}$ to represent the entire set. For a feature indexed by k, we define the sensitivity of $\hat{\mathbf{v}}$ to feature k as:

$$\hat{\mathbf{g}}_k = \frac{1}{n} \sum_{i=1}^{n} \mathbf{g}_{i,k}. \tag{10}$$

Then we can compute a $2 \times d$ Jacobian matrix of $\hat{\mathbf{v}}$:

$$\mathbf{J} = \begin{bmatrix} \hat{\mathbf{g}}_1^T & \cdots & \hat{\mathbf{g}}_d^T \end{bmatrix}, \tag{11}$$

which consolidates all compass needles of $\hat{\mathbf{v}}$. This aggregated point serves as a representation of the entire set, thereby enabling both pointwise and multilevel visualization of feature sensitivity (satisfying **DG1**).

6 Feature Compass Interface

The interface of Feature Compass comprises multiple navigation components, including the projection view, zoomed view, compass view, and sensitivity histogram, that together create a cohesive workflow for interactive exploration and interpretation of DR projections. These components are designed to support multiscale analysis and help transition seamlessly between global patterns and localized feature behaviors.

Projection View The projection view (Fig. 1-A) serves as the primary interface for visualizing the dataset in a 2D embedding generated by DR algorithms. Each point in the scatter plot is represented as a compass baseplate, the elliptical glyph that encodes both the directional orientation and sensitivity level of the corresponding high-dimensional data point. The orientation of the ellipse reflects the principal directions of feature influence, while its size and anisotropy indicate the magnitude and anisotropy of sensitivity. This representation allows users to identify global patterns, such as clusters or outliers, while preserving local feature dynamics.

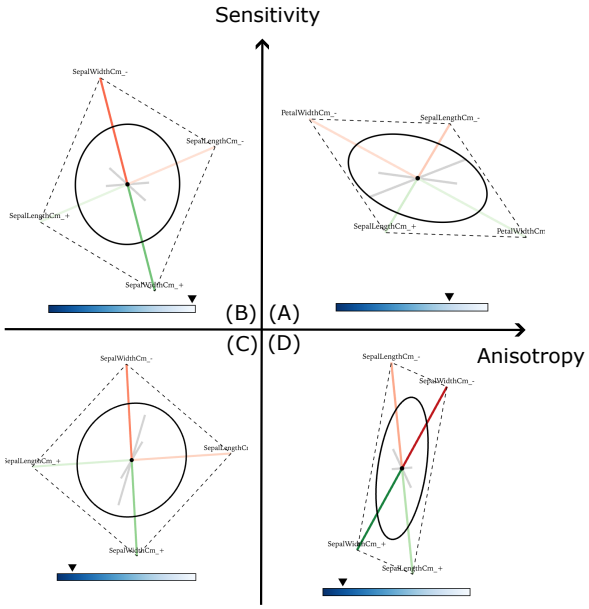

Fig. 5: Four compass patterns introduced by Feature Compass. **Anisotropy** describes how feature influence varies by direction, manifesting as elongated ellipses in the visualization. Points with high anisotropy appear to be more elliptical while others appear to be more circular. **Sensitivity** measures the strength of those directional influences on a point's position, offering feedback on how susceptible that point is to changes in high-dimensional feature space. As for the sensitivity meter, the blue end (left) indicates low sensitivity while the white end (right) indicates high sensitivity.

Zoomed View Interactive brushing in the projection view dynamically highlights regions of interest, which are then displayed in the zoomed view(Fig. 1-B) for detailed inspection. The zoomed view maintains a square frame to ensure consistent scaling and proportionality, preserving spatial relationships between points. This design choice enhances interpretability, particularly when analyzing regions with high point density or complex feature interactions. By linking the projection and zoomed views, Feature Compass enables users to explore data at multiple levels of granularity, from broad trends to fine-grained feature behaviors.

Compass View As shown in Fig. 5, Compass View is composed of two components, a compass and a sensitivity meter, which together reveal the sensitivity level of selected points and how that sensitivity is distributed across all directions. By default, Compass View displays the **aggregated compass** of all points contained in the Zoomed View (Fig. 1-B,C). When the user hovers over a particular point in the Zoomed view, the Compass View updates to show the individual **pointwise compass** corresponding to that specific point only (Fig. 9-D).

Sensitivity Histogram Complementing the visual exploration tools, the sensitivity histogram (Fig. 1-D) provides a quantitative summary of the sensitivity distribution across selected points. The histogram bins points based on their sensitivity levels, allowing users to quickly identify subsets of points with high or low sensitivity.

7 Case Studies

In this section, we applied Feature Compass to three distinct datasets to demonstrate its capability in facilitating explanation of DR plots of high-dimensional data. By leveraging its visualization and analytical features, we showcase how Feature Compass enables users to uncover meaningful patterns and insights across diverse datasets.

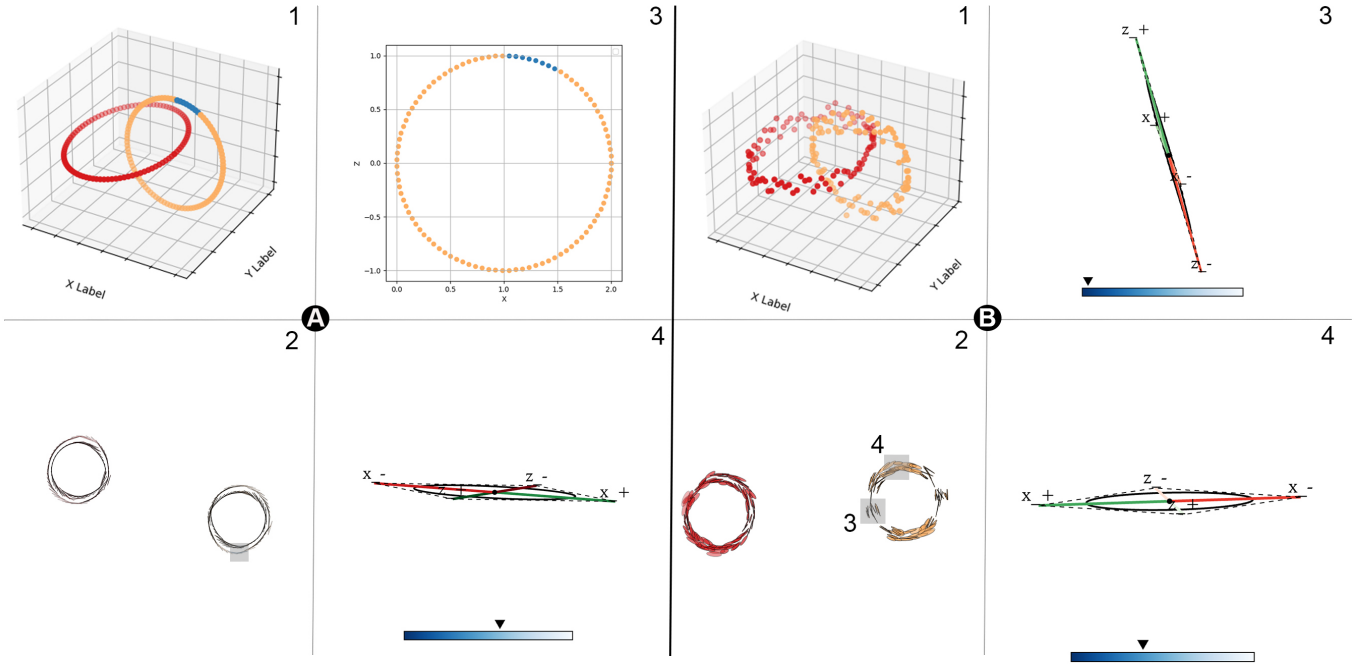

Fig. 6: Illustration of interlocked three-dimensional rings. The red ring is parallel to the XY plane, and the orange ring is parallel to the XZ plane. (A) shows the rings generated without noise. (A)-2 presents the projection view from Feature Compass, highlighting two regions on the orange ring(right) where points exhibit significantly different feature behaviors. (B) depicts the rings generated with noise introduced in the z-dimension. (B)-2 displays the projection view from Feature Compass, illustrating how the orange and red rings respond differently to the applied noise. Additionally, (B)-2 highlights two areas on the orange ring that also react distinctly to the noise. A detailed discussion of this figure is in Section 7.1

.

7.1 Synthetic Dataset

To demonstrate the capability of Feature Compass to capture feature sensitivity, we applied Feature Compass to a synthetic dataset consisting of two interlocked 3-dimensional rings, as shown in Fig. 6. The dataset comprises a red ring parallel to the XY plane, centered at $(0,0,0)$ with a radius of 1, and an orange ring parallel to the XZ plane, centered at $(1,0,0)$ with a radius of 1:

$$\begin{cases} (x-1)^2+z^2=1, \\ y=0. \end{cases} \tag{12}$$

Fig. 6A illustrates the dataset without noise. Among the orange ring, points with x between $[1,1.5]$ and z between $[0.5,1]$ are highlighted in blue as shown in Fig. 6A-1. In the projection view (Fig. 6A-2), Feature Compass employs ellipses to represent points, effectively revealing feature behaviors. The selected area in Fig. 6A-2 includes the blue points, whose feature compass is shown in Fig. 6A-4. Additionally, Fig. 6A-3 presents a scatter plot of the orange ring directly projected onto the plane. Let $\mathbf{p}_i=(p_{i,x},p_{i,y},p_{i,z})\in\mathbb{R}^3$ be a point. According to Eq. (12), we can compute gradients with respect to x and z:

$$\frac{\partial p_{i,z}}{\partial p_{i,x}}=-\frac{p_{i,x}-1}{p_{i,z}}. \tag{13}$$

To quantify relative influence between two gradients, we compute the ratio:

$$Ratio=\frac{\frac{\partial p_{i,z}}{\partial p_{i,x}}}{\frac{\partial p_{i,x}}{\partial p_{i,z}}}=\frac{-\frac{p_{i,x}-1}{p_{i,z}}}{-\frac{p_{i,z}}{p_{i,x}-1}}=\frac{(p_{i,x}-1)^2}{p_{i,z}^2}=\frac{(p_{i,x}-1)^2}{1-(p_{i,x}-1)^2}. \tag{14}$$

When $p_{i,x}$ is between $[1,1.5]$, the computed ratio $Ratio$ lies between $[0,0.3]$, consistently remaining less than 1. This indicates that blue points are more sensitive to changes in x compared to z, which is consistent with the feature compass shown in Fig. 6A-4. The compass demonstrates that the position of selected points are significantly influenced by both x and z, with x being the most dominant feature due to the highest magnitude of its corresponding compass needle. Furthermore, the opacity of needles in the compass reveals that the selected points exhibit extremely high z value and relatively lower x value. The direction of the needles further indicates that negative changes in z align with positive changes in x, while positive changes in z align with negative changes in x. This behavior aligns with the geometric configuration of the blue points within the orange ring, as depicted in Figure 6A-3. This corroborates the capability of our system to accurately capture feature sensitivity.

Capture Feature Sensitivity in Dataset with Noise To simulate experimental sensitivity in the synthetic dataset, we introduce additive noise the z-coordinate of each point. We add a small random offset ε_i only to the z-coordinate, resulting in a perturbed point $\mathbf{p}_i'=(p_{i,x},p_{i,y},p_{i,z}+\varepsilon_i)$, where $\varepsilon_i\sim\mathscr{U}(-\alpha,\alpha)$. Here, $\mathscr{U}(-\alpha,\alpha)$ denotes a uniform distribution on the interval $[-\alpha,\alpha]$. In our experiments, $\alpha=0.1$ bounds the perturbation magnitude such that $|p_{i,z}'-p_{i,z}|\leq 0.1$. This level of noise was chosen to introduce a small amount of variability without substantially altering the overall geometry of the ring structures.

The projection view (Fig. 6B-2) highlights distinct responses to noise between the red and orange rings. For the red ring, which lies parallel to the XY-plane, perturbation in z uniformly shift all points along the projection's z axis. Conversely, the orange ring, parallel to the XZ-plane exhibits two distinct behavioral patterns:

- **High-Sensitivity Regions** (Fig. 6B-3): These regions are characterized by large partial derivatives, indicating strong sensitivity of the projected coordinates to z-variations. Critically, such regions

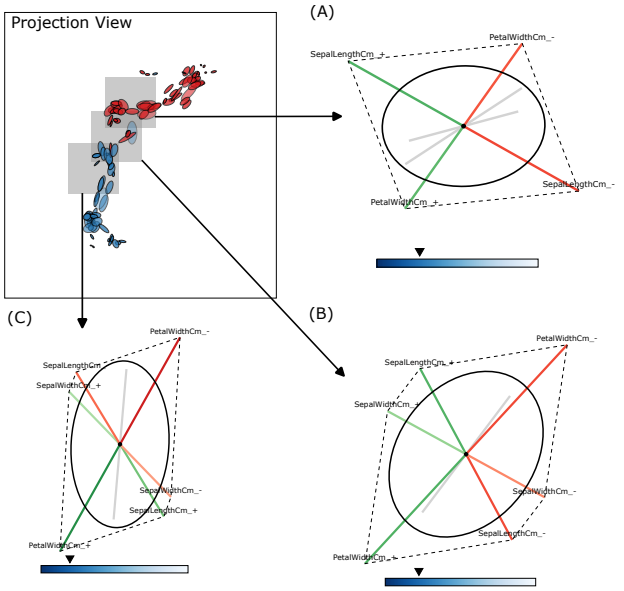

Fig. 7: Feature Compass for three regions of the Iris dataset that lie near or on the boundary between the versicolor and virginica classes.

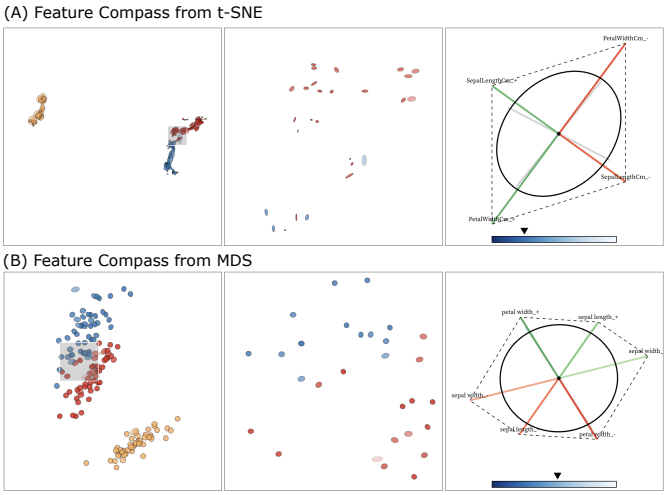

Fig. 8: Feature Compass for Iris dataset from t-SNE (A) and MDS (B)

inherently possess significant natural variability in z (i.e., the "signal" dominates, with $Var(z) \gg \alpha^2$). The additive noise ε_i is negligible relative to this intrinsic spread, resulting in minimal distortion of the projected geometry.

- **Low-Sensitivity Regions** (Fig. 6B-4): Here, the projection exhibits weak dependence on z, and the natural z-variability is inherently small. The noise magnitude α constitutes a substantial perturbation relative to the signal ($Var(z) \sim \alpha^2$), leading to disproportionate distortion in the projection.

To justify the observation, let σ^2 denote the natural variance of z in a region. The signal-to-noise ratio(SNR) is defined as:

$$SNR = \frac{\sigma_z^2}{\alpha/3}. \tag{15}$$

For the uniform distribution, noise variance is $\alpha/3$. In high-sensitivity regions ($\sigma_z^2 \gg \alpha/3$), $SNR \gg 1$, and noise effects are dwarfed by natural variability. In low-sensitivity regions($\sigma_z^2 \ll \alpha/3$), $SNR \ll 1$, and noise dominates, causing obvious geometric distortion. This dichotomy underscores how Feature Compass disentangles feature sensitivity from inherent data variability, enabling precise diagnosis of noise robustness in nonlinear projections.

7.2 Iris Dataset

We also applied Feature Compass to the Iris dataset [14], a benchmark dataset widely used in clustering and classification tasks. The Iris dataset consists of three classes of iris plants: setosa, versicolor, and virginica. Each class is characterized by four features: sepal length (s_l), sepal width (s_w), petal length (p_l), and petal width (p_w). Its simplicity and well-structured nature make it ideal for evaluating clustering algorithms, while the inherent overlap between some classes challenges feature-based separation methods. The chosen clusters, iris-virginica and iris-versicolor, are particularly significant because of their close similarity. These two classes often exhibit overlapping feature distributions, making their boundary challenging to analyze. This overlap provides an excellent test case for Feature Compass's ability to uncover subtle feature-driven separations and patterns.

Fig. 7 illustrates Feature Compass applied to three distinct regions of the projected Iris dataset, with blue points representing virginica and red points representing versicolor. Fig. 7-B focuses on the boundary region between these two clusters, which is particularly informative as it contains points most influenced by the features that differentiate the classes. By analyzing compass needles at the boundary, Feature Compass identifies s_w, p_w, and s_l as the most influential features in this region, as highlighted in Fig. 7-B. Among all these features, we identified that petal width (p_w) vectors have the strongest alignment with the spread of points near the boundary of two clusters(depicted in Fig. 7-Projection View). This alignment suggests that small changes in p_w lead to substantial positional shifts in the projected space, underscoring its role as a critical factor in defining the boundary structure and separating the two clusters. Based on the color of the p_w vectors, virginica exhibits higher p_w values compared to virginica.

To substantiate these observations, Feature Compass was applied to two areas near the boundary of two clusters (Fig. 7-A&C). The opacity of p_w vectors reveal that points in C possess higher petal width than points in A and B. This finding aligns with the insights from the color of p_w vectors in Fig. 7-B, further emphasizing the importance of p_w in distinguishing between versicolor and virginica. Identifying p_w as a discriminative feature has significant implications for clustering and classification tasks. In the context of the Iris dataset, these insights clarify how individual features contribute to the dimensional reduction process. By focusing on the boundary an deploying precise quantitative analysis, Feature Compass provides actionable insights into the dynamics of feature influence, demonstrating its utility in uncovering subtle but critical patterns in real-world datasets.

Comparison between t-SNE and MDS To investigate how different DR methods reveal feature patterns at cluster boundaries, we applied Feature Compass to the Iris dataset using MDS and compared it to t-SNE, as shown in Fig. 8. The MDS method employed here is based on the SMACOF (Scaling by Majorizing a Complicated Function) algorithm, which iteratively adjusts the low-dimensional coordinates to minimize a nonlinear stress function. Although the compass view for both methods indicates that p_w is the critical feature distinguishing versicolor and virginica, significant differences in the compass patterns emerge in selected regions. In particular, points in the t-SNE projection exhibit a more elongated pattern with substantial variation in point sensitivity levels, whereas points in the MDS projection appear more circular with noticeably lower sensitivity variability.

The different compass patterns of t-SNE and MDS reveal the nature of their underlying objective function. MDS minimizes a quadratic function of the distance differences:

$$\sigma = \frac{\sum_{i,j}(D^{HD}ij - D^{LD}ij)^2}{\sum_{i,j}(D^{HD}ij)^2} \tag{16}$$

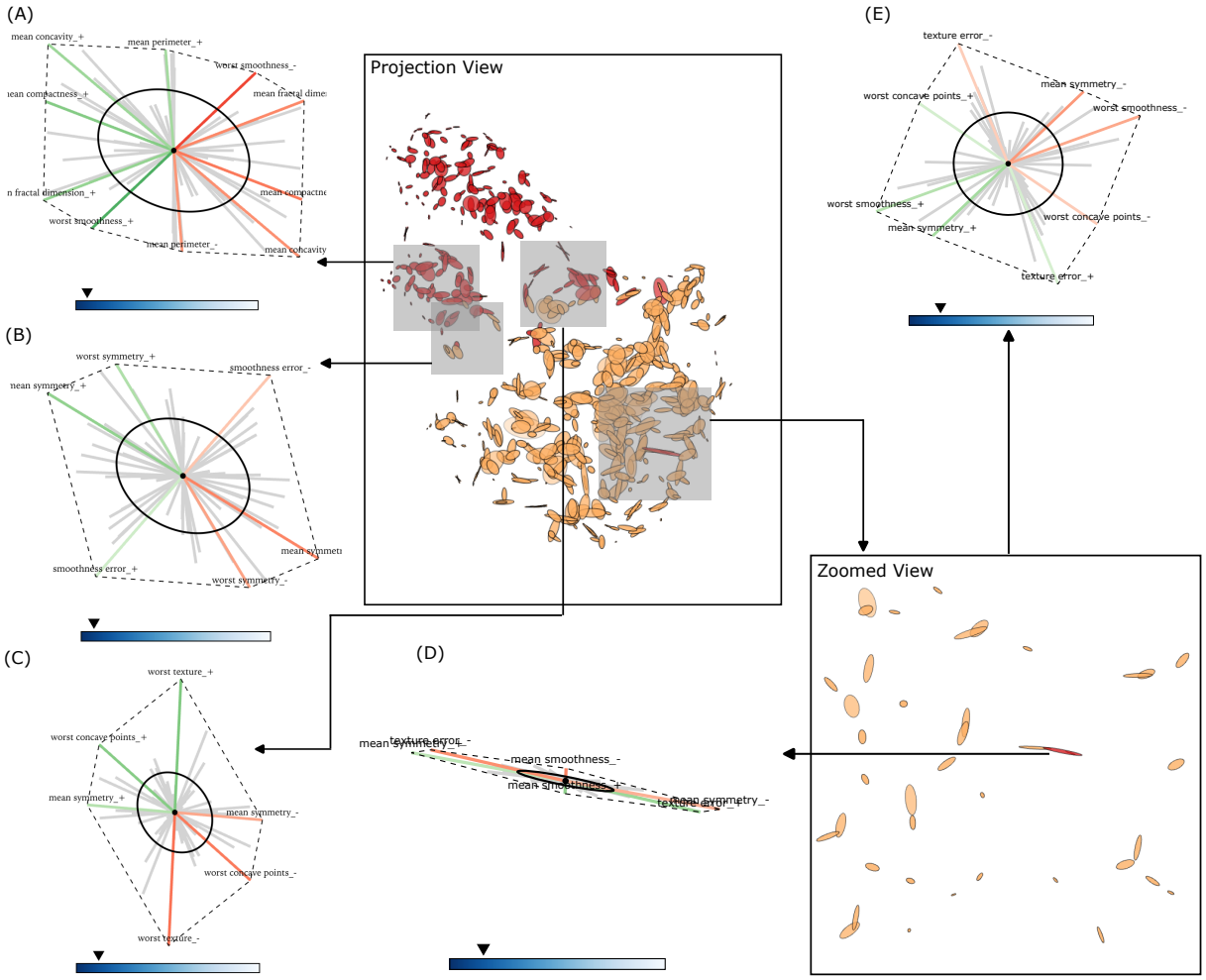

Fig. 9: Feature Compass applied to the Breast Cancer Dataset. Panel (A) displays the compass view of a sub-cluster within the Malignant group; panels (B) and (C) illustrate the compass views of two distinct boundary regions; panel (D) presents the compass view for a Malignant outlier inside the Benign cluster; and panel (E) shows the compass view of points in the Zoomed View.

This objective penalizes any deviation from the original distances, promoting global fidelity and leading to more uniformly distributed, often circular clusters. On the other hand, t-SNE minimizes a KL divergence:

$$KL(P\|Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{17}$$

which is not symmetric and tends to overemphasize local similarities. The use of a heavy-tailed distribution q_{ij} allows for larger differences in distances (especially for moderately separated points) without incurring a significant penalty. This typically results in elongated clusters with greater variability in local density.

7.3 Breast Cancer Dataset

The Breast Cancer Wisconsin (Diagnostic) dataset is a widely recognized benchmark in medical diagnostics, comprising 569 samples with 30 continuous features computed from fine needle aspirate (FNA) images of breast masses. They're labeled in two classes: Benign and Malignant. The diagnostic version of breast cancer dataset computes the mean, standard error and "worst" or largest (mean of the three largest values) of 10 features for each image, resulting in 30 features in total. These computed features provide a nuanced representation of tumor characteristics, which not only enhances the reliability of classification between malignant and benign tumors but also makes the dataset particularly suitable for applying Feature Compass, as it facilitates a detailed examination of how each feature influences the DR projections. In this section, we showcase how Feature Compass can bring interesting insights into this dataset as shown in Fig. 9.

Difference of Statistical Representations Figure Fig. 9-C illustrates how different statistical representations of the same metric can exert distinct influences on an individual data point. Feature Compass reveals that both mean texture and worst texture have considerable impacts on the point, albeit with slightly different directional orientations. Similarly, the system shows that mean smoothness and worst smoothness significantly affect the point, with each metric contributing a unique directional bias. These differences highlight the importance of examining multiple statistical descriptors, as each representation can capture distinct facets of the underlying data variability. These insights are critical to diagnostic precision and for improving predictive models, as they help to understand complex feature behaviors.

Boundary Points Investigation By applying Feature Compass along the boundary between two classes, we can analyze feature behaviors at critical decision points. As illustrated in Fig. 9-C. in this region, *worst texture*, *worst concave points*, and *mean symmetry* emerge as key discriminators between the classes. On the other hand, Fig. 9-B reveals that *mean symmetry*, *worst symmetry*, and *smoothness error* are the predominant features separating two clusters. These observations demonstrate that feature sensitivity—and consequently the impact

of various statistical representations of the same metric—can differ markedly across boundary areas. In some regions, pronounced differences in feature values between classes yield steeper gradients, whereas in other areas, greater overlap leads to more subtle variations. These insights can help to understand the patterns and details that separate malignant and benign cancers, ultimately improving both diagnostic accuracy and predictive modeling.

Sub-cluster Reasoning In addition to evaluating relationships between clusters with different labels, Feature Compass also facilitates within-cluster reasoning, as demonstrated in Fig. 9-A. Although labeled as Malignant, a noticeable gap exists between the lower cluster in Fig. 9-A and the major Malignant cluster. Feature Compass highlights that the lower region is characterized by a smaller *mean perimeter*, and higher *worst smoothness* and *mean fractal dimension*. These distinctions suggest that even within a single diagnostic category, there are specific sub-patterns that may correspond to differences in tumor morphology or aggressiveness. By capturing these subtle variations, Feature Compass enables a more refined analysis of tumor heterogeneity, which can inform tailored diagnostic assessments and treatment strategies.

Feature Sensitivity: Local vs. Global Fig. 9 demonstrates the capability of Feature Compass to visualize both local and global feature sensitivity. In panel Fig. 9-E, the feature compass is applied to all points in the Zoomed View, providing an aggregated perspective of feature sensitivity across this data region. This view indicates that points in this region are most sensitive to *worst smoothness, worst concave points, mean symmetry*, and *texture error*. Notably, all points in this region are labeled as Benign, except for one Malignant point. The feature compass for this Malignant point, shown in panel Fig. 9-D, reveals that its position is most sensitive to *texture error, mean symmetry*, and *mean smoothness*. The fact that the anomaly stands out within in aggregated benign region highlights the importance of local analysis. It suggests that global trends might mask critical local variations, an that examining individual points can reveal crucial differences.

In summary, the Breast Cancer Wisconsin (Diagnostic) dataset provides a rich, multi-dimensional foundation for evaluating diagnostic features, and Feature Compass serves as an effective tool for extracting nuanced insights. By leveraging diverse statistical representations, investigating boundary and sub-cluster regions, and offering both local and global perspectives, Feature Compass enhances our understanding of feature behaviors and their impacts on class separation. These capabilities not only refine diagnostic precision but also support the development of more accurate predictive models.

8 Discussion

8.1 Feature Behavior Representation: Why Gradients?

Existing visualization frameworks have employed various metrics to analyze feature behavior in DR projections. For instance, Hypertrix [34] visualizes distortions introduced by DR algorithms, Feature Clock [30] quantifies feature contributions, and DimReader utilizes gradient-based analysis. We adopt and enhance the gradient-based approach because it offers a direct and mathematically precise measurement of how each feature influences a point's position in the DR projection. This fundamental property distinguishes it from derived metrics, minimizing the risk of misinterpretation. Specifically, the gradient of a DR projection with respect to a feature quantifies the sensitivity of the projection to changes in that feature. As a result, it provides explanations of **how** the position of a point or several points are influenced by high-dimensional features, which is an essential capability for XAI applications.

8.2 The Value of Point-wise Feature Sensitivity Analysis

Point-wise feature sensitivity analysis provides granular, localized insights into how individual features influence the placement of individual data points in a DR projection. Unlike global or aggregated methods, which summarize feature contributions across the entire dataset, point-wise sensitivity analysis captures local variations in feature influence. This fine-grained interpretability is crucial for understanding why specific clusters form, why outliers emerge, and how features interact differently across various regions of the projection.

Another advantage of point-wise sensitivity analysis is its ability to facilitate localized feature comparisons. Feature influence is not uniform across a projection, as some features may be dominant in certain clusters but insignificant elsewhere. By examining feature sensitivity at a point-wise level, users can compare how different features drive projections in different regions. This is particularly useful in exploratory data analysis, where understanding the regional importance of features can guide insights into dataset structure and classification patterns.

8.3 Scalability

Feature Compass is designed to be visually scalable in both points and features. Multi-level compass aggregation —ranging from individual point-wise analysis to aggregated views across clusters— scales with the number of data points, allowing users to seamlessly explore local details and global patterns. Compact baseplate representations in DR plots, combined with compact needle representations in the compass view, enable scalability in number of features without requiring additional views per feature.

8.4 Generalized Feature-based XAI

In addition to sensitivity analysis, a variety of generalized feature-based explanation methods are widely applied to attribute a model's predictions to its input features. By tracing predictions back to individual features, these methods pave the way for more transparent, interpretable, and trustworthy AI systems by enabling any per-feature explanation to feed directly into a visualization at the atomic level(e.g. pixel in image data, token in text data, value in tabular data). For instance, Layer-Wise Relevance Propagation (LRP) [3] proposes a general strategy for explaining nonlinear deep neural networks by propagating the prediction score backward from the output layer to the model inputs. LRP outputs per-feature relevance scores that can be visualized as heatmaps, a technique originally demonstrated in image-based tasks. Similarly, Han et al. [19] employ weighted backpropagation to generate saliency maps that highlight the visual features influencing image placement. These generalized approaches build upon feature-based explanation frameworks to offer deeper insights into how specific inputs drive model outputs.

9 Limitation and Future Work

A key limitation of our system is that the projection view in Feature Compass currently encodes feature sensitivity for each point in a discrete manner, rather than a continuous one. Building on a wind-map-inspired approach [13] could make the sensitivity gradients of different points more related in the DR projection, offering a holistic view of how sensitivity trends evolve across the projection space.

10 Conclusion

In this work, we introduced Feature Compass—a gradient-based visualization technique designed to enhance the interpretability of DR results. By leveraging point-wise gradients and consolidating them into visually expressive compass ellipses, Feature Compass provides a unified, scalable means of examining how high-dimensional features influence the 2D placement of data. This approach offers both global overviews and localized, fine-grained insights into data behavior, addressing key limitations of prior techniques. By integrating interactive brushing, zoomed views, and sensitivity histograms, Feature Compass serves as a flexible workflow for deeper data exploration. Through case studies on synthetic rings, the Iris dataset, and breast cancer data, we demonstrated how Feature Compass reveals essential patterns and feature interactions hidden within DR embeddings. Looking ahead, the method can be further extended to a broader range of non-linear DR algorithms, or coupled with predictive models for even richer XAI pipelines. As interpretability remains an open challenge in data visualization and machine learning, we envision Feature Compass as a step toward more transparent, user-guided exploration of complex, high-dimensional data.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 3

[2] N. Ahmad and A. B. Nassif. Dimensionality reduction: Challenges and solutions. In *ITM Web of Conferences*, vol. 43, p. 01017. EDP Sciences, 2022. 2

[3] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015. 9

[4] A. Björklund, J. Mäkelä, and K. Puolamäki. Slisemap: Supervised dimensionality reduction through local explanations. *Machine Learning*, 112(1):1–43, 2023. 1

[5] A. Buja and D. F. Swayne. Visualization methodology for multidimensional scaling. *Journal of Classification*, 19(1):7, 2002. 2

[6] P. Chhikara, N. Jain, R. Tekchandani, and N. Kumar. Data dimensionality reduction techniques for industry 4.0: Research results, challenges, and future research directions. *Software: Practice and Experience*, 52(3):658–688, 2022. 2

[7] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011. 3

[8] J. P. Cunningham and Z. Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *The Journal of Machine Learning Research*, 16(1):2859–2900, 2015. 2

[9] M. De Berg. *Computational geometry: algorithms and applications*. Springer Science & Business Media, 2000. 4

[10] 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 Transactions on Visualization and Computer Graphics*, 29(2):1559–1572, 2021. 2

[11] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea. Toward a quantitative survey of dimension reduction techniques. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):2153–2173, 2021. doi: 10.1109/TVCG.2019.2944182 2

[12] R. Faust, D. Glickenstein, and C. Scheidegger. Dimreader: Axis lines that explain non-linear projections. *IEEE transactions on visualization and computer graphics*, 25(1):481–490, 2018. 1, 2, 3

[13] M. W. Fernanda Bertini Viégas. wind map, 2012. 9

[14] R. A. Fisher. Iris. UCI Machine Learning Repository, 1936. DOI: https://doi.org/10.24432/C56C76. 7

[15] I. K. Fodor. *A Survey of Dimension Reduction Techniques*. May 2002. doi: 10.2172/15002155 2

[16] S. L. France and J. D. Carroll. Two-way multidimensional scaling: A review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(5):644–661, Sept 2011. doi: 10.1109/TSMCC.2010.2078502 2

[17] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, C-23(9):881–890, Sept 1974. doi: 10.1109/T-C.1974.224051 2

[18] A. Gisbrecht and B. Hammer. Data visualization by nonlinear dimensionality reduction. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(2):51–73, 2015. 1

[19] H. Han, R. Faust, B. F. Keith Norambuena, J. Lin, S. Li, and C. North. Explainable interactive projections of images. *Machine Vision and Applications*, 34(6):100, 2023. 9

[20] H. H. Harman. Modern factor analysis. 1960. 2

[21] H. Kim, J. Choo, H. Park, and A. Endert. Interaxis: Steering scatterplot axes via observation-level interaction. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):131–140, 2016. doi: 10.1109/TVCG.2015.2467615 1, 2

[22] O. Krakovska, G. Christie, A. Sixsmith, M. Ester, and S. Moreno. Performance comparison of linear and non-linear feature selection methods for the analysis of large survey datasets. *Plos one*, 14(3):e0213584, 2019. 2

[23] J. B. Kruskal and M. Wish. *Multidimensional scaling*. Number 11. Sage, 1978. 1

[24] B. C. Kwon, H. Kim, E. Wall, J. Choo, H. Park, and A. Endert. Axisketcher: Interactive nonlinear axis mapping of visualizations through user drawings. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):221–230, 2017. doi: 10.1109/TVCG.2016.2598446 1, 2

[25] J. A. Lee and M. Verleysen. *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007. 2

[26] W. Liu, C. North, and R. Faust. Visualizing spatial semantics of dimensionally reduced text embeddings. *arXiv preprint arXiv:2409.03949*, 2024. 1

[27] W. E. Marcilio-Jr and D. M. Eler. Explaining dimensionality reduction results using shapley values. *Expert Systems with Applications*, 178:115020, 2021. 1

[28] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. 1

[29] B. Montambault, G. Appleby, J. Rogers, C. D. Brumar, M. Li, and R. Chang. Dimbridge: Interactive explanation of visual patterns in dimensionality reductions with predicate logic. *IEEE Transactions on Visualization and Computer Graphics*, 2024. 2

[30] O. Ovcharenko, R. Sevastjanova, and V. Boeva. Feature clock: High-dimensional effects in two-dimensional plots. In *2024 IEEE Visualization and Visual Analytics (VIS)*, pp. 151–155, 2024. doi: 10.1109/VIS55277.2024.00038 2, 3, 9

[31] K. Pearson. Principal components analysis. *The London, Edinburgh and Dublin Philosophical Magazine and Journal*, 6(2):566, 1901. 2

[32] A. Plaza, P. Martínez, J. Plaza, and R. Pérez. Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations. *IEEE Transactions on Geoscience and remote sensing*, 43(3):466–479, 2005. 1

[33] A. A. Portnova-Fahreeva, F. Rizzoglio, I. Nisky, M. Casadio, F. A. Mussa-Ivaldi, and E. Rombokas. Linear and non-linear dimensionality-reduction techniques on full hand kinematics. *Frontiers in Bioengineering and Biotechnology*, 8, 2020. doi: 10.3389/fbioe.2020.00429 2

[34] S. Raval, F. Viégas, and M. Wattenberg. Hypertrix: An indicatrix for high-dimensional visualizations. In *2024 IEEE Visualization and Visual Analytics (VIS)*, pp. 1–5, 2024. doi: 10.1109/VIS55277.2024.00073 2, 3, 9

[35] W. Samek. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017. 2, 3

[36] M. Scholz, F. Kaplan, C. L. Guy, J. Kopka, and J. Selbig. Non-linear PCA: a missing data approach. *Bioinformatics*, 21(20):3887–3895, 08 2005. doi: 10.1093/bioinformatics/bti634 2

[37] M. Sedlmair, M. Brehmer, S. Ingram, and T. Munzner. Dimensionality reduction in the wild: Gaps and guidance. *Dept. Comput. Sci., Univ. British Columbia, Vancouver, BC, Canada, Tech. Rep. TR-2012-03*, 2012. 2

[38] M. Sedlmair, T. Munzner, and M. Tory. Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2634–2643, 2013. doi: 10.1109/TVCG.2013.153 2

[39] J. Stahnke, M. Dörk, B. Müller, and A. Thom. Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):629–638, 2016. doi: 10.1109/TVCG.2015.2467717 2, 3

[40] W. S. Torgerson. Theory and methods of scaling. 1958. 2

[41] A. Ullah, U. Qamar, F. H. Khan, and S. Bashir. Dimensionality reduction approaches and evolving challenges in high dimensional data. In *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*, IML '17, article no. 67, 8 pages. Association for Computing Machinery, New York, NY, USA, 2017. doi: 10.1145/3109761.3158407 2

[42] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9:2579–2605, Sept. 2008. 1, 2

[43] J. Venna and S. Kaski. Local multidimensional scaling. *Neural Networks*, 19(6):889–899, 2006. Advances in Self Organising Maps - WSOM'05. doi: 10.1016/j.neunet.2006.05.014 2

[44] M. Wattenberg, F. Viégas, and I. Johnson. How to use t-sne effectively. *Distill*, 1(10):e2, 2016. 2

[45] A. Wismüller, M. Verleysen, M. Aupetit, and J. A. Lee. Recent advances in nonlinear dimensionality reduction, manifold and topological learning. In *ESANN*, 2010. 2