Visualizing the Stability of 2D Point Sets from Dimensionality Reduction Techniques

Christian Reinbold, Alexander Kumpf and Rüdiger Westermann

Computer Graphics & Visualization Group, Technische Universität München, Garching, Germany
{christian.reinbold, alexander.kumpf, westermann}@tum.de

Abstract
We use $k$-order Voronoi diagrams to assess the stability of $k$-neighbourhoods in ensembles of 2D point sets, and apply it to analyse the robustness of a dimensionality reduction technique to variations in its input configurations. To measure the stability of $k$-neighbourhoods over the ensemble, we use cells in the $k$-order Voronoi diagrams, and consider the smallest coverings of corresponding points in all point sets to identify coherent point subsets with similar neighbourhood relations. We further introduce a pairwise similarity measure for point sets, which is used to select a subset of representative ensemble members via the PageRank algorithm as an indicator of an individual member’s value. The stability information is embedded into the $k$-order Voronoi diagrams of the representative ensemble members to emphasize coherent point subsets and simultaneously indicate how stable they lie together in all point sets. We use the proposed technique for visualizing the robustness of t-distributed stochastic neighbour embedding and multi-dimensional scaling applied to high-dimensional data in neural network layers and multi-parameter cloud simulations.

Keywords: visualization

ACM CCS: • Human-centred computing → Visualization techniques; • Computing methodologies → Dimensionality reduction and manifold learning

1. Introduction
Dimensionality reduction is used to compute a projection of a set of data points in some high-dimensional space into a low-dimensional space, typically two dimensions, in which visual data exploration can then be performed effectively. If the projection preserves relative distances between the initial data points, or at least allows distinguishing near from far neighbours, coherent subsets can be discerned and even computed in an unsupervised manner using clustering techniques in the low-dimensional space.

The results of many dimensionality reduction techniques, however, are subject to randomly selected initial parameters. For instance, multi-dimensional scaling (MDS) [CC00, KW78] sets the initial objects to random positions and then re-adjusts them. Re-adjustment is performed in a physics-inspired way, by pushing objects apart (or together) if two objects are too close to (or too far from) each other. The result of this relaxation depends on tuning parameters to adjust the strength of non-local repulsive forces. Due to this parameter dependency, multiple MDS projections are usually computed using varying input parameters and the best rest configuration is selected. Another frequently used dimensionality reduction technique is t-distributed stochastic neighbour embedding (t-SNE) [vH08]. In t-SNE, perplexity is an adjustable parameter that controls the size of the neighbourhood around each object that the projection attempts to preserve. Besides, t-SNE is often used with random values to initialize its seed configuration, i.e. the initial locations of projected objects which are considered by gradient descent optimization. Therefore, also t-SNE is usually re-run with different input configurations and the most meaningful projection is selected.

Due to these parametrization dependencies, MDS and t-SNE can be run multiple times and yield a different result each time. The variations can be both with respect to the location of projected data points and their neighbourhoods. It is in particular important to find those subsets of points with similar $k$-neighbourhood, i.e. the $k$ closest points, in all projections. Such stable subsets can be deemed coherent, and they indicate robustness of the dimensionality reduction technique. Analysing the stability of these $k$-neighbourhoods,
in context of the projections that are computed by the used dimensionality reduction technique, is the major focus of this work.

1.1. Contribution

We present a visualization technique to analyse the stability of neighbourhood relations in an ensemble of two-dimensional (2D) data points, and we demonstrate its use for assessing the robustness of dimensionality reduction techniques to variations in the input parametrizations. A novel coherence indicator based on \( k \)-order Voronoi diagrams measures how well \( k \)-neighbourhoods are preserved in multiple ensemble members.

The visual encoding we propose utilizes the partitioning of the projection plane based on \( k \)-order Voronoi diagrams. In this way, the spatial distribution of data points is maintained, and stability information of similar neighbourhoods is placed close to each other. This allows accessing in a very intuitive way region of neighbourhoods with similar stability, which cannot be accessed from non-spatial encodings like, e.g. bar charts. Since for each projection, a \( k \)-order Voronoi diagram is computed, we select the most representative one via the PageRank score. Voronoi cells are colour-coded according to the stability of enclosed subsets of data points, and the stability information is used to simplify the visual encoding by removing cell boundaries. Remaining cell boundaries are coloured according to how strongly separated the subsets in either cell are.

Our specific contributions are:

- the use of \( k \)-order Voronoi diagrams to analyse the stability of neighbourhood relations in an ensemble of 2D point sets;
- a stability measure for \( k \)-neighbourhoods in ensembles of 2D point sets, using smallest coverings of points belonging to a \( k \)-neighbourhood computed for all ensemble members;
- a visual encoding of the stability of neighbourhood relations, by visualizing major separating ridges in a \( k \)-order Voronoi diagram via colours indicating their separation strength.

Figure 1 demonstrates the application of the proposed technique to analyse the robustness of MDS and t-SNE for two different datasets, indicating the specific visual encoding we propose. It provides a general means to analyse the stability of neighbourhood relations in ensembles of 2D point sets. We use this means to shed light on how to determine for a given dimensionality reduction technique whether it can robustly reveal certain structures in the data, yet we do not aim to compare different dimensionality reduction techniques. In particular, we show that the proposed visualizations can be used to find subgroups of objects for which a current projection provides a robust representation of their locality. In a number of examples, including a synthetic dataset, we demonstrate the use of the proposed technique and the information concerning the robustness of dimensionality reduction techniques they provide.

2. Related Work

Our approach takes as input an ensemble of 2D point sets and visualizes which subsets of points are connected to each other, i.e. are neighbours, in all or many ensemble members using a suitable neighbourhood relation. The neighbourhood relation we employ is based on Voronoi diagrams. In visualization, classical one-order Voronoi diagrams have been used to encode properties of points [BD05, NSB04, PFMA06, LA11]. Aupetit [Aup07] discusses the limitations of two-order Voronoi diagrams for visual encoding properties of pairs of points, and suggests an improvement using Segment–Voronoi cells. We use higher order Voronoi diagrams to determine larger contiguous neighbourhoods, and clustering to obtain a meaningful visual encoding, thus resolving some of the limitations pointed out by Aupetit.

Regarding the input to our approach, it falls into the broader category of ensemble visualization techniques (see Wang et al. [WHLS18] for a recent overview). Notably, while ensemble visualization has emerged as an important field, to the best of our knowledge, visualization techniques for ensembles of 2D...
points are not existing. The majority of works in ensemble visualization have addressed ensembles of physical fields, or features derived from such fields, and they have focused on the extraction and visual encoding of their variability. Parametric statistical distributions and distribution shape descriptors for scalar-valued ensembles were presented by Love et al. [LPK05]. Isocontours in an ensemble of 2D scalar fields have been conveyed via spaghetti plots [PWB*09, SZD*10, Wil11]. Different variants of confidence regions were introduced to represent the major geometric trends in ensembles of isocontours and streamlines [WMK13, MWK14, FBW16, FKRW16]. Demir et al. [DJW16] proposed a closest-point representation to convey the central tendency of an ensemble of multi-dimensional shapes. Hummel et al. [HOGJ13] analyse the spread of particle trajectories in an ensemble of vector fields to reveal the transport variability, and Jarema et al. [JDKW15] model directional data ensembles via mixtures of probability density functions to compactly classify complex distributions. Poethkow and Hege [PH13] and Athawale et al. [ASE16] use location-wise estimators of non-parametric distributions from ensemble members and use these distributions to estimate the spread of surface and vector field features.

Alternatively, clustering has been used to group ensemble members regarding similar data characteristics [BM10, TN14, OKL*14, FBW16, FFST19]. While these techniques compare ensemble members to each other, our approach aims at finding groups of elements in each member which remain ‘close’ to each other in all members. Strehl and Ghosh [SG02] apply different clustering techniques to one single ensemble, and combine the results into a single clustering. Ferstl et al. [FKRW16] cluster different time steps of the same ensemble in a hierarchical way to convey the change of clusters over time. For the clustering of genomic data, Lex et al. [LSP*10] introduce extended parallel coordinate plots to compare different clusterings and analyse the quality of cluster assignments. Kumpf et al. [KTB*18] consider a set of ensembles and cluster each ensemble using the same algorithm and number of clusters. Clusters are matched to clusters in a reference clustering by using their intersection, and cluster variability plots are proposed to analyse the robustness of the clustering results. Our method, in contrast, does not make any assumption about whether the data can be clustered and into how many clusters it can be separated.

In our scenario, each ensemble member is a set of 2D points, which is generated by applying a dimensionality reduction technique to a set of points in a high-dimensional space. Some recent surveys [KH13, LMW*16] give thorough overviews of visualization techniques for high-dimensional data. In combination with dimensionality reduction, clustering is often used to identify groups of points lying close together in the low-dimensional space or forming coherent structures in this space. Wenskovitch et al. [WCR*17] discuss the combination of dimensionality reduction and clustering techniques, and provide recommendations for their concurrent use. General overviews of the numerous techniques for dimensionality reduction and clustering can be found in the surveys by Sorzano et al. [SVM14] and Everitt et al. [ELL11], respectively.

Related to our approach are also techniques which aim to find projections that best represent the structures in high-dimensional data, by using quality measures for projections [FT74, HA85]. Even though the goal of these techniques is different to ours, as we do not attempt to find the best projection for a given dataset, proposed measures indicate the (dis-)similarity between projections and might be used for robustness analysis as well. Examples include vector distance measures for high-dimensional feature descriptors [BvL11] and feature vectors derived from point-wise distance matrices [JHB*17], as well as measures using matrix norms to quantify the dissimilarity of multivariate projections invariant to affine transformations [LT16]. None of these approaches, however, analyses the stability of multiple projections. Instead, they quantify the similarity of projections to order them or find new projections with as less as possible redundancy to previous ones.

3. Method Overview

Let $X$ be a finite set of $n$ entities in some multi-dimensional coordinate or parameter space. Our method analyses an ensemble $\mathcal{E}$ of embeddings of $X$ into the 2D plane. Each embedding is an ordered 2D point set $P = \{(p_i), 1 \leq i \leq n\}$ of cardinality $n$. Points in different ensemble members that have the same index refer to the same entity.

To analyse whether certain subgroups of points are stably connected—using a suitable neighbourhood relation—across all ensemble members, we introduce the pipeline in Figure 2. In general,
for a certain subgroup of size $k$, all $\binom{n}{k}$ available subgroups need to be tested. As this is intractable, we utilize $k$-order Voronoi diagrams—computed in step (a)—to reduce this number to $O(kn)$ many $k$-neighbourhoods. The $k$-order Voronoi diagrams provide topological information that is used to assemble neighbourhoods of fixed size $k$ into subgroups of arbitrary size. The geometric structure of these diagrams is further used to visualize simultaneously the stability information that is obtained from it and a 2D embedding of $X$.

In step (b), the stability of each $k$-neighbourhood is assessed using minimum covering circles of the corresponding point set in all other ensemble members. In this way, multiple stability values are obtained for each neighbourhood. These values are finally aggregated into a single value that quantifies the overall stability of a neighbourhood in the ensemble (step (c)).

Higher order Voronoi diagrams can be used in principle to perform a fine-granular analysis of the neighbourhood relations in the ensemble. However, since they comprise thousands of convex polygons which cannot easily put into relation to the data points, some expertise is required to observe the major trends. To ease the interpretation, an abstract visualization is provided in which the information is condensed. Therefore, the major information regarding stable subgroups and their relations to each other is extracted by clustering the $k$-order Voronoi diagrams (step (d)). The clustering is only used to obtain an alternative visual representation, and it incorporates derived stability information of subgroups. A region’s stability is mapped to its background colour, and ridges between these regions are coloured to indicate how strongly adjacent subgroups are separated. We will subsequently call these plots robustness plots. A robustness plot allows to identify points which are connected via $k$-neighbourhoods of locally high stability. Furthermore, if regions of high stability have a particular shape, such as band- or ball-like, one can conclude that these shapes also appear in most of the ensemble members.

For each $P \in \mathcal{P}$, a distinct $k$-order Voronoi and robustness plot is generated. As demonstrated in Section 5.7, many of these plots carry similar information, in particular regarding the stable regions. Building upon a pairwise similarity matrix that is computed in step (e), a small set of representative plots is finally selected in step (f). These plots carry most of the stability information, and they are used to embed the stability information into the 2D domain. A robustness plot of any ensemble member reveals the agreement of the member to the derived stability information.

4. Stability Analysis

Our notion of stability is based on how well neighbourhoods are preserved in different point sets: Given a $k$-subset $V$ of points in one point set, how large is the spread of these points in all other point sets. This is determined by computing the smallest possible disks that cover the points in $V$ in the other sets. These disks may also cover points not being part of $V$, indicating that additional points interfere with the neighbourhood relationship. The stability value $\text{stab}(V, P)$ of $V$ with respect to $P \in \mathcal{P}$ is then computed as the ratio between $k$ and the number of points covered by the disk. Consider $P$ containing the grey and orange points in Figure 3(a), and $V$ containing the three orange points. Then, $\text{stab}(V, P)$ evaluates to $3/5$, since the cardinality of $V$ is 3 and the smallest disk covering $V$ contains five points. This measure is invariant to uniform scaling and isometric transformations applied to $P$, and it has domain $[0, 1]$, as the disk always contains at least the $k$ points of $V$. We extend this measure to an ensemble of point sets $\mathcal{P} = \{P_1, P_2, \ldots, P_l\}$, by using the mean over all $\text{stab}(V, P_i), i \in [1, 2, \ldots, l]$. Alternatively, aggregations that better represent the distribution of stability values can be used.

4.1. Voronoi diagrams

The subsets $V$ that are considered in the stability analysis are computed from the higher order Voronoi diagrams. To reduce the number of neighbourhoods to be computed, the analysis could, in principle, be restricted to the $k$-neighbourhood around each initial point, i.e. the $k$-neighbourhood including the point and its $(k-1)$-nearest neighbours. However, as shown in Figure 3(b), this approach does not always allow for a meaningful assessment of the stability of subgroups in the point set. In the example, when grouping each point with its two nearest neighbours, no 3-subset covering at least one light grey and one dark grey point is obtained. In particular, stability values computed for these subsets are not affected by the distance between the light and dark grey group in other point sets of the ensemble $\mathcal{P}$. Thus, information about the stability of the relative position of the dark grey group to the light grey group cannot be inferred. Both may be adjacent only in the seen point set, or in all point sets of $\mathcal{P}$.

Ideally, the situation in Figure 3(b) can be resolved by sampling the 3-nearest neighbours of the location marked by a blue cross, so that points with different colours are related to each other. In the general case, this approach leads to sampling all $k$-nearest neighbourhoods induced by every possible location in 2D space. When grouping all 2D locations according to their $k$-nearest neighbours, a partition of the 2D space into a finite number of (possibly unbounded) convex regions is obtained. These regions are called Voronoi cells. A many-to-many correspondence between cells and points is established by relating a cell to each of its $k$-nearest points (see Figure 3c). The partition is called the $k$-order Voronoi

![Figure 3](image-url)
The given point set and extends the well-known (one-order) Voronoi diagram.

Considering region-wise neighbourhoods provides several benefits over the approach of only sampling neighbourhoods centred around 2D points. Firstly, it does convey strictly more information by construction since all 2D points may also be considered 2D locations. Secondly, it defines a meaningful spatial relationship by relating two \( k \)-neighbourhoods iff their corresponding Voronoi cells share a common facet. Adjacent neighbourhoods of 100% stability are guaranteed to be located next to each other in every point set. Finally, the computation of \( k \)-order Voronoi diagrams is tractable for modest values of \( k \), with run-time complexity \( O(k^2n \log n) \), and yields \( O(kn) \) neighbourhoods [Lee82].

In our approach, \( k \)-order Voronoi diagrams are computed in step (a) via the algorithm by Lee [Lee82]. It gradually generates Voronoi diagrams of increasing order by cutting cells of the previous order \( k - 1 \). As the \( k - 1 \) nearest neighbours for a cell are already known, the remaining \( k \)-th neighbour is computed by intersecting the cell with the one-order Voronoi diagram obtained when discarding the \( k - 1 \) known neighbours. Then, the resulting cut-outs with similar \( k \)-neighbourhoods are stitched together. It remains to compute one-order Voronoi diagrams for which we use the Qhull implementation [BDH96].

When computing smallest disks for \( k \)-neighbourhoods in a Voronoi diagram, the iterative approach of Lee greatly increases the performance. Each cut-out originates from a cell of order \( k - 1 \), for which the smallest encompassing disk \( D \) of its neighbourhood regarding a point set \( P \) has already been computed. If the added \( k \)-th point, embedded in \( P \), is also contained in \( D \), the disk \( D_{new} \) for all \( k \) points is again \( D \). If not, the point has to be located on the boundary of \( D_{new} \). Since each circle is entirely defined by three points, \( D_{new} \) can be computed in \( O(1) \) whenever three or more cut-outs are stitched together. When a \( k \)-order cell is generated by stitching two cut-outs—less is not possible—a possibly existing third boundary point has to be guessed in \( O(k) \). In practice, approximately every second cell is stitched by three or more pieces.

### 4.2. Clustering stable subgroups of points

As can be seen in Figure 3(c), the interpretation of higher order Voronoi diagrams is challenging. First of all, the size of a Voronoi cell, in general, does not provide relevant information. In addition, since related cells and points may be distant from each other, especially for high orders or areas with low point density, it becomes difficult to infer these relations. Finally, the fine granular cell structures introduce visual clutter and even interfere with the basic stability information to be conveyed (see Figure 1(a2)).

To ease the interpretation of \( k \)-order Voronoi diagrams, density-aware clustering is used (step (d) in Figure 2). The approach builds upon the observation that in \( k \)-order Voronoi diagrams, when their cells are coloured according to the stability of neighbourhoods, regions packed with stable, connected components are often separated by unstable bands. This property is seen in Figures 1(a), 2 and 13.

By means of clustering, these components are extracted and translated to subgroups of stable points. Hence, considerably larger stable areas than \( k \)-neighbourhoods can be identified and put into relation to each other. Two points are located in the same cluster if they are part of the same stable submanifold, whereas a submanifold is considered stable if it occurs in most of the point sets of the ensemble. The topological structure of the submanifold is outlined by ridges.

#### 4.2.1. Clustering of Voronoi cells

From a \( k \)-order Voronoi diagram, enriched by stability information, connected areas of locally high stability can be emphasized by arranging the Voronoi cells in a hierarchical manner. For this, we utilized concepts from persistent homology [EH08, EH10]. More specifically, the function mapping Voronoi cells to their respective stability values can be interpreted as a monotonic function defined on a simplicial complex. Its merge tree yields the desired hierarchical representation, and a persistence value attached to each node in the tree is used to select the final clustering.

Let \( \lambda \in [0, 1] \) be an arbitrary threshold. For a selected \( \lambda \), the superlevel set \( L^+ (\lambda) \) is the set of all Voronoi cells with a stability value greater or equal to \( \lambda \). The superlevel set is composed of a set of connected components \( \text{Comp}(\lambda) \). When \( \lambda \) is decreased towards 0, these components merge until one connected component remains. By relating a component to all components that have been merged to form it, a tree with nodes \( \bigcup_{\lambda \in (0,1)} \text{Comp}(\lambda) \) is generated that serves the desired hierarchical representation. For instance, the Voronoi diagram in Figure 5 yields the hierarchical representation that is shown by the Dendrogram (1) in Figure 4.

In principle, and similar to some common clustering algorithms like Agglomerative Hierarchical clustering and Density-based Spatial Clustering of Applications with Noise (DBSCAN), a clustering can be obtained from a certain level of the merge tree. If clusters are extracted from the same level, however, prominent clusters at other levels will be missed. This situation commonly occurs when one cluster is located in a vastly stable region of the Voronoi diagram (cluster \( \bullet \) in Figure 1c), and a rather unstable cluster is surrounded by significantly less stable cells (cluster \( \circ \) in Figure 1c).

---

**Figure 4:** The cluster hierarchy for the Voronoi diagram in Figure 5. Each box indicates a connected component with the width encoding its significance. Horizontal lines signal merge events. By distorting the stability threshold scale so that the timing scale becomes linear, the height of a box encodes the lifetime of a component and its area corresponds to persistence. Dendrogram (2) displays the simplified hierarchy. The framed clusters form the largest disjoint component set regarding overall persistence. Box colours match the components in Figure 5(a).
To avoid this limitation, we utilize the same concept as used in the clustering technique Hierarchical Density-based Spatial Clustering of Applications with Noise (HDBSCAN) [CMS13]. Instead of taking clusters from the same level, HDBSCAN slices the merge tree at different depths such that components which persist for an extended duration before merging into another one are kept intact. To detect persistent components, sufficiently small components and their merge events are dropped beforehand.

When HDBSCAN is used in the current scenario, the size of a component can be determined by the number of encompassed Voronoi cells. However, since we ultimately aim for clustering points but not Voronoi cells, we introduce a different notion of size, called significance. The significance of a Voronoi cell roughly represents by that cell (see Section 4.2.2). Then, the significance of a connected component is the sum of significances of its encompassed Voronoi cells. The box width in Dendrogram (1) of Figure 4 encodes the significance of each component.

The modified HDBSCAN algorithm now works as follows: Firstly, the hierarchical representation is simplified and made robust against noise by dropping connected components. In our case, it is reasonable to drop all components with a significance lower than the order k of neighbourhoods. This process results in Dendrogram (2) in Figure 4. In a second step, a persistence term $\varphi_C$ is assigned to each connected component based on its significance integrated over its lifetime $\Delta t_C$. In Figure 4, the area of each component represents its persistence. The time $t(\lambda)$ of a merge event is derived from the value of $\lambda$ when the event occurred. The negative of the derivative $-\partial t(\lambda)/\partial \lambda$ serves as an unnormalized weight distribution along the domain of $\lambda$, and the lifetime between $t(\lambda_2)$ and $t(\lambda_1)$ reduces to integrating weights along $[\lambda_1, \lambda_2]$ by

$$t(\lambda_2) - t(\lambda_1) = \int_{\lambda_1}^{\lambda_2} -\frac{\partial t(\lambda)}{\partial \lambda} \, d\lambda.$$ 

We suggest to use $t(\lambda) \mapsto 1 - \lambda^2$ with the weight distribution $-\partial t(\lambda)/\partial \lambda \propto \lambda$ to favour stable regions. In all our experiments, this lead to the most plausible clustering results. Depending on the application, the use of other weight distributions such as a constant distribution—yielding $t(\lambda) \mapsto -\lambda$—may be worth considering as well.

Finally, a set of disjoint connected components is selected such that the sum of their persistence values becomes maximal. The selected set represents the final clustering of the Voronoi cells, of which—as in DBSCAN—some may not be covered and thus considered noise. In Figure 4, the selected components are framed by dashes. As seen in Figure 5(a), the components do not cover the unstable cells, thus letting the user instantly distinguish between stable and unstable groups of points when they are overlaid.

### 4.2.2. Point clusters

To relate the Voronoi cell clusters to the initial point set, these clusters are interpreted as fuzzy point clusters, and hard point clusters are derived by a fix assignment of points to their most likely cluster. Since an initial point can be part of many equally stable neighbourhoods, it contributes fractions to each cell $C$ the point is related to.

![Figure 5: Extracting point clusters from a three-order Voronoi diagram. (a) Connected components of Voronoi cells in the simplified cluster hierarchy shown in Figure 4. (b) The contributions of the orange point are indicated by overplotting the related cells with their respective contribution values. (c) Contributions after concentration. (d, e) Pie charts illustrate proportions of a point’s contribution to (d) the highlighted cell (orange) and all other cells (white), and (e) fuzzy point clusters stemming from the framed Voronoi cell clusters of Figure 4. White slices indicate contributions to cells considered noise.

Fractions are chosen such that they are proportional to $\text{stab}(C, \epsilon)$ and add up to one. The contributions of an exemplary point are visualized in Figure 5(b).

Compared to assigning the whole contribution of a point to a single cell, however, this approach smooths out densities over adjacent cells. As a consequence, it prevents HDBSCAN from identifying meaningful clusters. To avoid this, we decided to let the contributions of a point $p$ ‘flow’ upwards to adjacent, related cells with higher stability until the whole contribution is concentrated in cells with spatial-local maximal stability. Applying this process to the situation in Figure 5(b) gives the concentrated contributions shown in Figure 5(c).

A Voronoi cell can now be related to a fuzzy point cluster by gathering the contributions from points contained in its $k$-nearest neighbourhood. For a single cell, the sum of all contributions defines its significance (see Figure 5d). Note that the contribution of the highlighted point in Figure 5(c) to the marked cell in Figure 5(d) is visualized in both figures. A Voronoi cell cluster may now be interpreted as a fuzzy point cluster, by adding up all contributions of the cells contained in the cluster. When applying the process to the selected Voronoi cell clusters of Figure 4, the fuzzy point clustering depicted in Figure 5(e) is obtained. Defuzzying yields the expected clustering that separates points contributing to the upper and lower Voronoi cells, respectively.

### 4.3. Visualizing stability in point sets

We propose an abstract visualization of $k$-order Voronoi diagrams which refrains from the limitations of $k$-order Voronoi diagrams mentioned in Section 4.2. In addition to showing the point set in a
scatterplot, points in different clusters as well as points deemed noise are separated via ridges. The ridges are extracted by querying the (regular) one-order Voronoi diagram. Here, each ridge is a boundary between two adjacent cells, each containing a single point. If the points fall into different clusters, the ridge is drawn. The ridge colour is determined by the least stable cell that is touched when following the line connecting both points in the original $k$-Voronoi diagram. It encodes how strongly both clusters are separated. The background colour of a cluster is defined by a weighted average of the stability values of the cells assigned to the cluster, computed by

$$\frac{\sum_{c} \omega_c \text{stab}(c, \mathcal{E})}{\sum_{c} \omega_c} \quad \text{with} \quad \omega_c := \sigma_c \cdot \Delta t_c,$$

where $\sigma_c$ is the significance of cell $c$ and $\Delta t_c$ its lifetime in the cluster. Thus, cells contributing most and for the longest time to the cluster are emphasized.

The colourmap of choice originates from a subrange of the perceptually uniform cmocean ice colourmap [TGH*16]. It emits a pleasing light colour for highly robust regions while effectively revealing small changes in robustness. Both factors are especially important if most of the screen is filled with areas depicting high robustness. The background of noise clusters is coloured in grey. It can be clearly distinguishable from the chosen colourmap without introducing any undesirable signaling effect. Points are coloured according to cluster membership. Assigned colours have been carefully selected to be perceptually discriminative without conflicting with the background.

### 4.4. Selection of representative point sets

Since a single plot for each ensemble member would overburden the visualization, we guide the user’s selection process of viewing certain point sets in step (f) by computing a pairwise similarity matrix in step (e) of the pipeline. The pairwise similarity between point sets $P, Q \in \mathcal{E}$ encodes how well the $k$-neighbourhoods arising in $Q$ are preserved in the point set $P$. It is computed by using the stability measure $\text{stab}(V, P)$ proposed in Section 4. The stability values of all neighbourhoods $V$ occurring in the $k$-order Voronoi diagram of $Q$ are averaged.

Since the $k$-order Voronoi diagrams of $P$ and $Q$ contain different neighbourhoods, interchanging the roles of $P$ and $Q$ changes the result. This makes the measure non-symmetric. It is also not reflexive in situations as in Figure 3(a), where the centre of the smallest disk encompassing a $k$-neighbourhood $V$ is not located in the $k$-order Voronoi cell associated to $N$. When comparing the point set with itself, its three-order Voronoi diagram contains the orange coloured cell. It relates to the neighbourhood formed by the orange points with its stability given by 3/5 (see Section 4). Thus, the average of stability values over all cells is strictly lower than one. However, in all of our experiments, strong symmetry and a sharp diagonal in the similarity matrices are observed (see, for instance, Figure 6).

To extract clusters of similar point sets in the ensemble $\mathcal{E}$, matrix seriation [BBHR*16] is used to sort columns and rows such that hidden patterns become apparent. We utilize the two rank ellipse seriation of Chen [Che02] for identifying blocks of point sets, as shown in Figure 6(b). When the blocks are identified, for each block, a representative is picked that is determined by the highest score when applying the PageRank algorithm [BP98].

### 5. Results and Evaluation

We now analyse the stability of $k$-neighbourhoods in ensembles of 2D point sets that are generated via dimensionality reduction techniques from high-dimensional data points. To generate the robustness plots, DBSCAN and k-means are used for clustering. Performance measurements are performed on a standard desktop PC, equipped with an Xeon CPU E5-1650 v2 with 6 cores @ 3.50GHz, 32 GB RAM, and an NVIDIA GeForce GTX 1070 Ti graphics card with 8 GB VRAM.

The following datasets are used:

- **MNIST**: A subset of the MNIST dataset [LBBH98]. It is composed of 1k 196-dimensional points, each representing the 196 activations in the second layer of a convolutional network that was trained to classify input images to digits 0–9. Since all digits occur equally often in the 1k inputs, 10 equally large clusters can be expected.
- **Cloud**: A 3D multi-parameter cloud simulation [WBJ*18]. The dataset contains roughly 11k points, each representing a 12-dimensional vector showing physical parameters such as ice, graupel and water contents at different locations in the 3D simulation domain. It is unknown whether the dataset contains clusters.
- **Gaussian**: A synthetic dataset, generated by sampling a mixture of eight isolated multi-variate Gaussian distributions in 3D space with random covariance matrices. The dataset contains 1k points equally spread over all distributions.

For each dataset, an ensemble of 2D points is created via MDS or Barnes-Hut t-SNE [VDM14] with randomly generated initial point positions. One hundred different random initializations are used to represent the ensemble variability. For a certain dataset, this number can be adapted by generating new random initializations until the average of the pairwise distances introduced in Section 4.4 falls below a threshold. The size $k$ of the neighbourhoods we aim
to analyse is set to 10, independently of the dataset. Due to space restrictions, we analyse only the overall best ranked projection of each dataset and do not consider other representations of, e.g. the MNIST dataset projected with MDS.

The times required to generate the Voronoi plots using a single thread on our target architecture are given in Figure 7. The approach scales approximately linearly, only degraded by the quadratic runtime behaviour of naively computing the smallest disk enclosing for a certain neighbourhood. By employing a hierarchical acceleration structure, the complexity of neighbour identification can be decreased to $O(n \log n)$. For computing the pairwise similarity matrix, all stages listed in Figure 7, except for the clustering stage, are required. They can be executed for all projections in parallel.

5.1. Choice of $k$

The parameter $k$ controls the locality of our proposed method. By construction, our approach cannot detect stable subsets of maximal size < $k$ since each $k$-neighbourhood containing a stable subset of smaller size also contains some unstable points enforcing low stability values. On the other hand, as we assemble stable regions by collecting circular shaped neighbourhoods, all detected regions have to covered by disks not containing unstable points. If the parameter $k$, and thus the disks, become too large, stable regions of intricate shape may not be detected anymore. For this reason, we favour a small value for $k$ in our experiments.

To detect a stable group of points, our approach requires this group to be locally stable as well, i.e. its $k$-neighbourhoods must be stable over the entire ensemble. For instance, by increasing $k$ from 10 to 60 in the example shown in Figure 8(a), the points are bisected into two stable regions (light background). For $k = 10$, this information is lost. Instead, the background reveals the stability of smaller subgroups. If no large stable subgroup exists, such as in Figure 8(b), increasing $k$ smooths out the patterns arising for smaller $k$. Animations with $k$ ranging from 2 to 100 are provided in the Supporting Information.

We suggest to inspect several point sets of the ensemble $\mathcal{E}$ to determine a reasonable value of $k$. By covering prominent structures with sufficiently small disks, one can compute the average amount of points covered per disk and choose this value as an initial guess for $k$. Due to the iterative nature of our approach, $k$ may be lowered without costly re-computations. In practice, the magnitude of $k$ is limited by the quadratic complexity in $k$ of computing Voronoi diagrams (see Section 4.1).

5.2. Cluster variability plots

In recent work by Kumpf et al. [KTB*18], a method for analysing the variation in composition of clusters over multiple clusterings and the variability in cluster membership for individual ensemble members was introduced. The method takes a set of ensembles, each composed of a number of high-dimensional data vectors, and then clusters these data points separately for each ensemble. The method assumes that the number of clusters is known, and a reference clustering exists to which all clusters can be matched. Via pie glyphs, the method encodes for each data point the frequency of cluster membership over the set of clusterings (see Figures 10a,b and 11b). Stable clusters are indicated by groups of glyphs coloured predominantly with the same colour, with no or only a small number of pieces with different colours. Highly fragmented pie glyphs indicate frequent changes in cluster assignments over the ensemble, hinting to instable groups. By picking a glyph, all data points with similar cluster membership over all clusterings can be highlighted.

Since the method strongly relies on the clustering results, including the selected number of clusters and clustering algorithm, it can obscure the relationships between data points. This is demonstrated in Figure 9, where each member of an ensemble of three 1D point sets including eight elements is clustered via k-means. Even though the ensemble members are only slightly jittered without changing the neighbourhood relationships, the clustering algorithm—depending on the point positions as well as a random initial positions of centroids—groups different sub-sets together. Hence, the visualization using pie charts (bottom row in Figure 9) exhibits a transitioning artefact. It suggests that both clusters are connected in most ensemble members, and thus, form a single, connected cluster in the ensemble. In the following experiments, we demonstrate the

---

**Figure 7:** Performance measurements of our pipeline for computing a robustness plot in a single thread.

**Figure 8:** Voronoi plots of two small-scale datasets (100 points) with (x1) $k = 10$ and (x2) $k = 60$.

**Figure 9:** Top rows: Three 1D point sets, each split by k-means into two clusters. Bottom row: the resulting pie glyphs of the method of Kumpf et al. [KTB*18].
different strengths and weaknesses of the method by Kumpf et al. in comparison to Voronoi plots. We will also show that both visualizations can be combined effectively, the Voronoi plot makes use of the visual background channel not occupied by Kumpf et al.

5.3. Cloud dataset

The robustness of t-SNE to variations in its input configurations is analysed in the following. For the Cloud dataset, the pairwise similarity matrix renders all projections equally similar (see Figure 6a). Thus, a single representative projection is selected via the PageRank algorithm (see Section 4.4). Corresponding Voronoi and robustness plots are displayed in Figures 1(a) and (b). The visualizations show that t-SNE is largely robust, and only a few scattered outliers are produced. As seen in the region dominated by points in Figure 1, some clusters can be deemed adjacent in many projections since no ridges form. Conversely, a cluster enclosed by dark ridges (in Figure 1) is not considered adjacent to its currently depicted neighbouring clusters in many projections.

To compare Voronoi plots to the pie glyphs by Kumpf et al., we use their method in combination with the DBSCAN clustering algorithm. The use of DBSCAN is motivated by the band structures in the projections, from which the existence of manifold structures in the original data can be concluded. Although the clusterings of individual projections look meaningful, it is difficult to relate them to each other over multiple projections. Due to different linkage thresholds in each projection, vastly different split and merge events occur, and the number of computed clusters varies. This makes it impossible to compute meaningful matchings between different clusterings. As a consequence, many pie glyphs are highly fragmented (in Figure 10a). Over that, transitioning artefacts, as seen in region (1) and discussed in Section 5.2, are introduced and need to be resolved.

Voronoi plots, on the other hand, do not suffer from these deficiencies, as they do not rely on separate clusterings for each projection. In Figure 10(c), areas of high stability are reliable determined. Two dominant cluster colours are observed in region (2) of Figure 10(a), yet the robustness plot (Figure 10c) does not show a ridge between them. When analysing projections manually, one sees that both clusters indeed are located next to each other in all projections, confirming the lack of a ridge in the robustness plot. In comparison, pie glyphs show multiple dominant colours, but not a smooth transition between them. Hence, spatial proximity cannot be concluded solely from the pie glyphs. DBSCAN fails to reliably extract this information, since both groups of points are often separated by a small area of zero density. The same mismatch holds true for region (3).

The result when using k-means with 16 clusters instead of DBSCAN is shown in Figure 10(b). Region (4) now splits into four clusters, which can only be joined manually by following smooth transitions. Transitions, on the other hand, can be misleading. For instance, one could assume a transition between colours and in region (5). When analysing the projections in combination with the robustness plot, it turns out that points being mainly dark blue move independently of the rose ones. Furthermore, region (6) is perceived as highly instable. Yet, besides being broken apart into two or three components from time to time, the components themselves are isolated and stable when browsing the ensemble of projections. Thus,
the robustness plot correctly classifies this region as a slightly less stable and strongly isolated cluster. The robustness plot also captures the proximity of the crescents in region (7), and it reveals that they are always placed next to points of region (8) with only a small strip of empty space between them. Pie glyphs fail to communicate this relation.

5.4. MNIST dataset

The assumption of 10 clusters in the MNIST dataset—one for each digit 0 to 9—suggests using the method by Kumpf et al. [KTB*18] to analyse the stability of these clusters. In our implementation, we use k-means for clustering the initial 2D point sets.

In particular, it can be seen that the method of Kumpf et al. fails to reveal certain anomalies, i.e. small inter-cluster details. The pie glyphs in Figure 11(a1) suggest a strongly stable cluster. The robustness plot in the background, on the other hand, shows a separating ridge of medium stability. When plotting the input digits corresponding to each data point (see Figure 11a2), we observe that the robustness plot separates the hand-written digit 2 with and without loops. By following the cluster over the ensemble, we discover that the cluster of digit 2, in fact, consists of two adjacent sub-clusters warping independently of each other.

From the structure and colouring of the pie glyphs in Figure 11(b), stable and instable regions, outliers as well as transitions between clusters can be concluded. For the MNIST dataset, the Voronoi plot in the background conveys mostly similar information (Figure 1c shows the Voronoi plot and derived groups of stable points). However, Voronoi plots, in this case, are not effective in relating distant points to each other. Although we point out instable neighbourhoods in regions (1) and (2), only due to the pie glyphs, one can detect that both regions regularly exchange some points.

5.5. Gaussian dataset

For the Gaussian dataset, the plot in Figure 13 reveals all isolated Gaussians. In particular, the following well-known properties of t-SNE can be observed. Firstly, t-SNE tries to preserve neighbourhoods so that points of the same isolated Gaussian are always clustered together. Highly stable regions around each Gaussian confirm this assumption. Secondly, the arrangement of the eight clusters is random since no neighbourhood considered by t-SNE touches two Gaussians simultaneously. The plot reveals this instability by dark ridges separating the Gaussians.

5.6. Sensitivity of k-order Voronoi diagrams

To investigate the sensitivity of Voronoi plots to small perturbations in the reference point set, the following experiments are performed: Each point is jittered from its 2D location by an offset uniformly sampled from $[-x, x]^2$. $x$ is increased continually from one experiment to the next. We choose $x \in \{0.01, 0.02, 0.05, 0.1, 0.12, 0.15, 0.2, 0.5, 1.0, 2.0\}$, with 0.12 being the average nearest neighbour distance of all selected projections. Only the transitions with the most significant changes are shown in Figure 12. The unperturbed plots are shown in Figures 1(a) and (b).

Up to a box radius of $x = 0.2$, one can hardly perceive any difference in the coloured k-order Voronoi diagrams, although already for $x = 0.1$, over 50% of all Voronoi cells in the unperturbed diagram are lost. For values $x > 0.2$, the stability values start to degrade notably. Note that even for $x = 2.0$, the Voronoi plot still shows the same ridge structures. When inspecting the robustness plots, significant changes cannot be observed up to $x = 0.15$. Starting at $x = 0.2$, the clustering approach starts to fragment the previously generated clusters. We attribute this behaviour to the use of a weight...
distribution favouring high stability in Section 4.2.1. While numerous small, highly persistent components are maintained, the overall persistence of larger components degrades. As soon as the persistence of a larger component drops below the accumulated persistence of its subcomponents, it fractures. This effect can be counteracted in principle, by either exchanging the weight distribution or by raising the significance threshold when simplifying the component hierarchy.

Since we do not think that inspecting globally instable projections is particularly meaningful, we adhere to the suggested weight distribution and keep the significance threshold as low as reasonable. Overall, we conclude that the proposed method is robust to noise of a magnitude similar to the average nearest neighbour distance. Thus, adjacent points may flip positions without significant changes.

5.7. Verifying representative projections

We further evaluate how well the selected representative projection emphasizes the major trends in the data. Therefore, different projections are compared by mapping point colours in a random projection to the clustering of similar \( k \)-neighbourhoods obtained from the best ranked projection. The results are shown in Figure 14.

Notably, the representative projection of the Cloud dataset seen in Figure 14(a) as well as the randomly selected projection in Figure 14(b) both show similar structures. For instance, the two dominating clusters \( \bigcirc \; \bigcirc \), various smaller bands, as well as a joint connecting an identical subset of bands exist in both projections. Furthermore, clusters are not intermingled. This is in accordance with the pairwise similarity matrix in Figure 6(a), which suggests that each projection represents the ensemble fairly well.

Nonetheless, t-SNE breaks some bands differently in every projection. When viewing only the projection in Figure 14(a), the connectivity of separated bands cannot be recovered since neighbourhoods of distant points are not considered. For instance, it cannot be recovered that clusters \( \bigcirc \; \bigcirc \; \bigcirc \) as well as \( \bigcirc \; \bigcirc \) are connected in most projections. If the projection in Figure 14(b) is chosen as the representative, other connections get lost, such as these of cluster \( \bigcirc \). When using the computed clusterings for each projection—obtained as in Section 4.2—as input for the method of Kumpf et al., some of the missing links can be exposed by querying for cluster-robust subsets as seen in Figure 1(a1). Here, the point marked with a cross is queried for all points being located in the same cluster for at least 70% of all projections. All highlighted points are covered by the two red lenses. As a result, the connection between \( \bigcirc \) and \( \bigcirc \) can be identified.

Next, the same analysis is run for the MNIST dataset projected with MDS. It provides two blocks of projections, as shown in Figure 6(b). By applying the PageRank algorithm, a representative for the larger block is obtained (see Figure 1c), which is compared to a randomly selected projection of each block. Figure 14(c) clearly shows that the projection selected from the larger block is a rotated version of the representative, while many clusters are intermingled when viewing the projection of the smaller block shown in Figure 14(d). We make this observation for all projections we randomly sampled from both blocks, indicating that the selected representative indeed represents the larger block of the dataset.

6. Conclusion and Future Work

We have proposed a new approach for visualizing the stability of \( k \)-neighbourhoods in ensembles of 2D point sets. We have demonstrated the use of this approach for identifying coherent subgroups of points over many ensemble members, and thus, assessing the robustness of dimensionality reduction techniques. For this purpose, we have introduced a novel stability measure for \( k \)-order Voronoi diagrams, and we have used this measure to determine stable neighbourhoods over many point sets. The similarity measure has also been applied to determine a subset of representative ensemble members using matrix seriation techniques as well as the PageRank algorithm. In comparison to the cluster-driven analysis of point sets, the proposed approach does not rely on the choice of an intermediate clustering and succeeds in detecting inter-cluster details and intra-cluster relationships.

In a number of examples, we have shown that the proposed approach gives interesting insights into the behaviour of dimensionality reduction techniques. Besides quantifying the
sensitivity of techniques to variations in their initial parametrizations, certain characteristics of the used techniques could be revealed. Furthermore, the proposed approach has revealed structures in datasets that were previously unknown.

In future work, we will address the following issues: Firstly, we will investigate whether the proposed similarity measure can be used to progressively find the most representative ensemble members, without the need to compute many of them in advance. Secondly, we will examine performance improvements to enable interactive insertion and re-positioning of points. In this way, we hope to gain further insight into the particular strengths and weaknesses of dimensionality reduction techniques, and eventually provide data-specific guidelines for their use. Lastly, we intend to apply the proposed technique in other fields such as crowd analysis and fluid dynamics. We see in particular the application to particle sets which are seeded in a certain region, to determine subgroups that remain stably together during particle integration.

Acknowledgements

The research leading to these results has been done within the subprojects B5 ‘Data-driven ensemble visualization’ of the Transregional Collaborative Research Center SFB/TRR 165 ‘Waves to Weather’ funded by the German Research Foundation (DFG). We thank all the reviewers for their constructive criticism and valuable comments.

References

[ASE16] Athawale T., Sakhare E., Entezari A.: Isosurface visualization of data with nonparametric models for uncertainty. IEEE Transactions on Visualization and Computer Graphics 22, 1 (January 2016), 777–786.

[Aup07] Aupetit M.: Visualizing distortions and recovering topology in continuous projection techniques. Neurocomputing 70, 7 (2007), 1304–1330. Advances in Computational Intelligence and Learning.

[BBHR*16] Bährisch M., Bach B., Henry Riche N., Schreck T., Fekete J.-D.: Matrix reordering methods for table and network visualization. Computer Graphics Forum 35, 3 (2016), 693–716.

[BD05] Balzer M., Deussen O.: Voronoi treemaps. In IEEE Symposium on Information Visualization, 2005 (INFOVIS 2005) (October 2005), pp. 49–56.

[BDH96] Barber C. B., Dobkin D. P., Huhdanpaa H.: The quickhull algorithm for convex hulls. ACM Transactions on Mathematical Software 22, 4 (1996), 469–483.

[BM10] Brückner S., Möller T.: Result-driven exploration of simulation parameter spaces for visual effects design. IEEE Transactions on Visualization and Computer Graphics 16, 6 (2010), 1468–1476.

[BP98] Brin S., Page L.: The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30, 1 (1998), 107–117. Proceedings of the Seventh International World Wide Web Conference.

[BvLBS11] Bremm S., von Landesberger T., Bernard J., Schreck T.: Assisted descriptor selection based on visual comparative data analysis. Computer Graphics Forum 30, 3 (2011), 891–900.

[CC00] Cox T. F., Cox M. A.: Multidimensional Scaling. Chapman and Hall/CRC, London, 2000.

[Che02] Chen C.-H.: Generalized association plots: Information visualization via iteratively generated correlation matrices. Statistica Sinica 12, 1 (2002), 7–29.

[CMS13] Campello R. J. G. B., Moulavi D., Sander J.: Density-based clustering based on hierarchical density estimates. In Advances in Knowledge Discovery and Data Mining (Berlin, Heidelberg, 2013), J. Pei, V. S. Tseng, L. Cao, H. Motoda and G. Xu (Eds.), Springer, Berlin, Heidelberg, pp. 160–172.

[DJW16] Demir I., Jarema M., Westermann R.: Visualizing the central tendency of ensembles of shapes. In SIGGRAPH ASIA 2016 Symposium on Visualization (2016), ACM.

[EH08] Edelsbrunner H., Harer J.: Persistent homology — A survey. Contemporary Mathematics 453 (2008), 257–282.

[EH10] Edelsbrunner H., Harer J.: Computational Topology: An Introduction. American Mathematical Society, Providence, RI, 2010.

[EJS11] Everitt B. S., Landau S., Leese M., Stahl D.: Cluster Analysis. John Wiley & Sons, Ltd, West Sussex, 2011.

[FBW16] Fertl F., Bürger K., Westermann R.: Streamline variability plots for characterizing the uncertainty in vector field ensembles. IEEE Transactions on Visualization and Computer Graphics 22, 1 (January 2016), 767–776.

[FFST19] Favelier G., Faraj N., Summa B., Tierny J.: Persistence atlas for critical point variability in ensembles. IEEE Transactions on Visualization and Computer Graphics 25, 1 (January 2019), 1152–1162.

[FKRW16] Fertl F., Kanzler M., Rautehaus M., Westermann R.: Visual analysis of spatial variability and global correlations in ensembles of Iso-Contours. Computer Graphics Forum 35, 3 (2016).

[FT74] Friedman J. H., Tukey J. W.: A projection pursuit algorithm for exploratory data analysis. IEEE Transactions on Computers C-23, 9 (September 1974), 881–890.

[HA85] Hubert L., Arabie P.: Comparing partitions. Journal of classification 2, 1 (1985), 193–218.

[HOGJ13] Hummel M., Obermaier H., Garth C., Joy K.: Comparative visual analysis of Lagrangian transport in CFD ensembles. IEEE Transactions on Visualization and Computer Graphics 19, 12 (December 2013), 2743–2752.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd
JDKW15] Jarema M., Demir I., Kehrer J., Westermann R.: Comparative visual analysis of vector field ensembles. In 2015 IEEE Conference on Visual Analytics Science and Technology (VAST) (October 2015), pp. 81–88.

JHB*17] Jäckle D., Hund M., Behrisch M., Keim D. A., Schreck T.: Pattern trails: Visual analysis of pattern transitions in subspaces. In IEEE Conference on Visual Analytics Science and Technology (VAST) (2017).

KH13] Kehrer J., Hauser H.: Visualization and visual analysis of multifaceted scientific data: A survey. IEEE Transactions on Visualization and Computer Graphics 19, 3 (March 2013), 495–513.

KTB*18] Kumpf A., Tost B., Baumgart M., Riemer M., Westermann R., Rautehaus M.: Visualizing confidence in cluster-based ensemble weather forecast analyses. IEEE Transactions on Visualization and Computer Graphics 24, 1 (2018), 109–119.

KW78] Kruskal J. B., Wish M.: Multidimensional Scaling. Number 07–011 in Sage University Paper series on quantitative applications in the social sciences. Sage Publications, Beverly Hills, 1978.

LA11] Lespinats S., Aupetit M.: Checkviz: Sanity check and topological clues for linear and non-linear mappings. Computer Graphics Forum 30, 1 (2011), 113–125.

LBBH98] Lecun Y., Bottou L., Bengio Y., Haffner P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 11 (November 1998), 2278–2324.

Lee82] Lee D.-T.: On k-nearest neighbor Voronoi diagrams in the plane. IEEE Transactions on Computers C-31, 6 (June 1982), 478–487.

LMW*16] Liu S., Maljovec D., Wang B., BREMER P.-T., Pascucci V.: Visualizing high-dimensional data: Advances in the past decade. IEEE Transactions on Visualization and Computer Graphics 23, 3 (2017), 1249–1268.

LPK05] Love A. L., Pang A., Kao D. L.: Visualizing spatial multivalue data. IEEE Computer Graphics and Applications 25, 3 (May 2005), 69–79.

LSP*10] LEX A., Streit M., Partl K., Kashofer K., Schmalstieg D.: Comparative analysis of multidimensional, quantitative data. IEEE Transactions on Visualization and Computer Graphics 16, 6 (November 2010), 1027–1035.

LT16] Lehmann D. J., Theisel H.: Optimal sets of projections of high-dimensional data. IEEE Transactions on Visualization & Computer Graphics 22, 1 (January 2016), 609–618.

MKW14] Mirzargar M., Whitaker R., Kirby R. M.: Curve boxplot: Generalization of boxplot for ensembles of curves. IEEE Transactions on Visualization and Computer Graphics 20, 12 (2014), 2654–2663.

NSB04] Nocke T., Schumann H., Böhm U.: Methods for the visualization of clustered climate data. Computational Statistics 19, 1 (February 2004), 75–94.

OLK*14] Oeltze S., Lehmann D. J., Kuhn A., Janiga G., Theisel H., Preim B.: Blood flow clustering and applications in virtual stenting of intracranial aneurysms. IEEE Transactions on Visualization and Computer Graphics 20, 5 (2014), 686–701.

PFMA06] Pinho R., Ferreira de Oliveira M. C., Minghim R., Andrade M. G.: Voromap: A voronoi-based tool for visual exploration of multi-dimensional data. In 10th International Conference on Information Visualisation (IV’06) (July 2006), pp. 39–44.

PH13] Pöthkow K., Hege H.-C.: Nonparametric models for uncertainty visualization. Computer Graphics Forum 32, 3 (2013), 131–140.

PWB*09] Potter K., Wilson A., Bremer P.-T., Williams D., Doutraux C., Pascucci V., Johnson C. R.: Ensemble-Vis: A framework for the statistical visualization of ensemble data. In Proceedings of the IEEE International Conference on Data Mining Workshops (2009), pp. 233–240.

SG02] Strehl A., Ghosh J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3 (December 2002), 583–617.

SVM14] Sorzano C. O. S., Vargas J., Montano A. P.: A survey of dimensionality reduction techniques. arXiv preprint arXiv:1403.2877, 2014.

SZD*10] Sanyal J., Zhang S., Dyer J., Mercer A., Amiurn P., Moorhead R. J.: Noodles: A tool for visualization of numerical weather model ensemble uncertainty. IEEE Transactions on Visualization and Computer Graphics 16 (2010), 1421–1430.

TGH*16] Thyng K. M., Greene C. A., Heitland R. D., Zimmerle H. M., DiMarco S. F.: True colors of oceanography: Guidelines for effective and accurate colormap selection. Oceanography 29 (September 2016). https://doi.org/10.5670/oceanog.2016.66.

TN14] Thomas D. M., Natarajan V.: Multiscale symmetry detection in scalar fields by clustering contours. IEEE Transactions on Visualization and Computer Graphics 20, 12 (2014), 2427–2436.

VDMA14] Van Der Maaten L.: Accelerating t-sne using tree-based algorithms. The Journal of Machine Learning Research 15, 1 (2014), 3221–3245.

vH08] Van der Maaten L., Hinton G.: Visualizing high-dimensional data using t-sne. Journal of Machine Learning Research 9 (November 2008), 2579–2605.

WBJ*18] Wellmann C., Barrett A. I., Johnson J. S., Kunz M., Vogel B., Carslaw K. S., Hoose C.: Using emulators to understand the sensitivity of deep convective clouds and hail to...
environmental conditions. *Journal of Advances in Modeling Earth Systems* 10 (2018). https://doi.org/10.1029/2018MS001465.

[WCR*17] Wenskovitch J., Crandall I., Ramakrishnan N., House L., Leman S., North C.: Towards a systematic combination of dimension reduction and clustering in visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 24 (2018), 131–141.

[WHLS18] Wang J., Hazarika S., Li C., Shen H.: Visualization and visual analysis of ensemble data: A survey. *IEEE Transactions on Visualization and Computer Graphics* 25 (2019), 2853–2872.

[Wil11] Wilks D. S.: *Statistical Methods in the Atmospheric Sciences*. Academic Press, Waltham, MA, 2011.

[WMK13] Whitaker R. T., Mirzargar M., Kirby R. M.: Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2713–2722.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Video of varying k - Example a

Video of varying k - Example b