Understanding Attribute Variability in Multidimensional Projections

Lucas Pagliosa
ICMC–USP, São Carlos, Brazil

Paulo Pagliosa
FACOM–UFMS, Campo Grande, Brazil

Luis Gustavo Nonato
ICMC–USP, São Carlos, Brazil

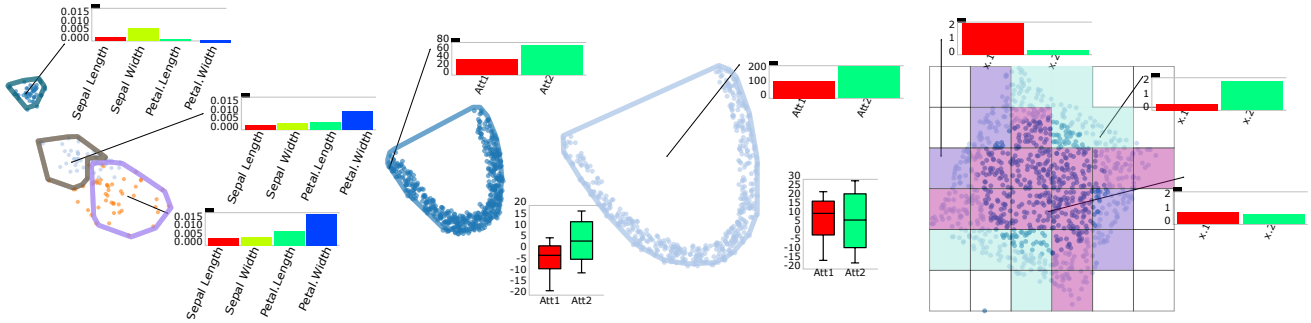

Fig. 1. Our technique enhances the analysis of projected data. Each bar chart (one per cluster in the figure) shows how data attributes behave in each dimension. The higher the bar of an attribute a, the more a contributes to data dispersion. In other words, the plot shows which set of attributes are more or less relevant to data clustering. Left: manual clustering of the Iris dataset. The user has selected three groups of projected points. The visualization can be used to identify attributes that mainly contribute for clusters formation/separation on the visual space. Middle: DBSCAN clustering of the Bananas dataset. The box plot gives more details about the distribution of each attribute and shows important differences between both groups such as min/max values, mean and interquartile range. Right: proposed variance vector clustering over the Circle dataset, which represents a circle inscribed in a square. Our algorithm has found three clusters. Two clusters identify cells containing mainly horizontally and vertically distributed points (sides of the square), and another identifies cells with a similar variation of the points attributes (circle).

***Abstract*—Multidimensional Projection techniques can help users to find patterns in multidimensional data. However, while the visualization literature is rich in techniques designed to improve the projection itself, only a handful of papers shed light into the attributes that contribute to cluster formation or the spread of projected data. In this paper, we present a web-based visualization tool that enriches multidimensional projection layout with statistical measures derived from inputted data. Given a set of regions to analyze, we used statistical measures, such as variance, to highlight relevant attributes that contribute to the points' similarities in each region. Experimental tests show that our technique can help identify important attributes and explain projected data.**

***Keywords*-attribute-based clustering; high-dimensional data visualization; interactive visual analysis**

I. Introduction

Visualizations play an important role in multidimensional data analysis. One strategy for visual data analysis consists of methods that show all attributes (dimensions) simultaneously, e.g., scatterplot matrices [1] and parallel coordinates [2]. Another, Multidimensional Projections (MP), reduces data dimensionality by using most representative attributes [3], [4] or mapping from high to low-dimensional spaces (2D or 3D) using some projection technique [5], [6], [7]. However, the "curse of dimensionality" makes the analysis of such data challenging. As the number of dimensions gets bigger, both methods suffer from clutter, outlier sensitivity and loss of information, thus jeopardizing the data interpretation.

In this paper, we introduce a web-based tool combining both statistical and visual analysis to help users understand the variability of the clustered data. Our pipeline allows users to project, group and explore the data using variability measures. Our main contribution is to enrich standard MP techniques with variability to provide information on the projection, which helps to highlight relevant attributes that contribute to the points' similarities in each cluster (see Fig. 1).

The remainder of the paper is organized as follows. Section II introduces state-of-the-art attribute analysis techniques found in the literature. We present our attribute analysis methodology in Section III. Results and validations are provided in Section IV. Discussion and future work are in Section V. We conclude in Section VI.

II. Related Work

Point clouds resulting from MP methods allow visual analysis of groups and neighborhood structures. However, they do not convey information related to the content of the underlying data by its own. Two main alternatives have been proposed to tackle this issue, namely, summarization and attribute analysis. Summarization methods rely on information derived from MP layouts to group similar instances while visualizing a summary of each group contents. Examples of such visualization resources are word tags [8], [9], [10], textual and image snippets [11], [12], and thumbnail pictures [13], [14]. Attribute analysis techniques (the focus of this paper), on another hand, group points in the visual space according to

their original similarity, focusing on visually revealing relevant attributes in each group.

When dealing with attribute analysis, recent methods have been proposed to define important dimensions. Kandogan [15] ranks the attributes of instances in each group using a set of metrics, highlighting the top ranked attributes as "annotations". One of the main drawbacks of this technique is the demand of well-structured and separated clusters in the visual space. Joia et al. [16] make use of singular value decomposition and word clouds to identify and visualize important attributes. Nonetheless, projection errors can lead to completely wrong conclusions. Broeksema et al. [17] resort to color coded Voronoi partitions to uncover salient attributes from groups of similar instances. Turkay et al [18] create a dual space representation, analyzing both instance and dimension space simultaneously. The main influence of their work was the direct statistical exploration in the visualization, as the brushing and linking reflection in both spaces. Yet et al. [19] modified Turkay's approach by creating a matrix/tree visualization that hierarchically explores subgroups of both dimension and instance spaces. Seo et al. [20] proposed a tool where users can select one and two-dimensional rank features to explore a dimension and a pair of dimensions, respectively; however, no correlation between all attributes and clusters were made. Cao et al. [21] use Voronoi diagrams [22] and Treemaps [23] to model projected groups attributes and mainly facilitate clusters comparison. Henry et al. [24] combine a graph matrix and traditional graph representation to find patterns and relationships between selected nodes.

Recently, Stahnke at al. [25] proposed a Web-based tool for analyzing dimensionality reduction results. Their tool enables the analysis of attribute distribution in the visual space using a monochrome heatmap, which consists of a grid where each cell is colored according to the mean value of a particular attribute of the points projected withing that cell. In contrast to our approach, visualizing several attributes simultaneously is not viable with Stahnke's methodology.

III. Attribute-based Data Exploration

When exploring multidimensional data, some data attributes may bring data points closer whereas others contribute to their dispersion. Our goal is to map the variability of those attributes onto the projected data. This map can be visualized by enriching the projected data with attribute-based information derived from the original data. We describe the six stages of our technique, namely: i) multidimensional projection; ii) projection filtering; iii) region definition; iv) variability computation; v) clustering; and vi) data visualization and exploration. Our input data is a CSV file composed by a set of instances $\mathcal{X} \subset \mathbb{R}^n$, where each instance $x \in \mathcal{X}$ is a row of the file. Each element of x represents a data attribute or dimension.

The data projection is a mapping $\hat{p} : \mathbb{R}^n \mapsto \mathbb{R}^2$ of points in n-dimensional space onto the 2D plane. Poorly projected points, i.e., points whose neighborhood relations in $\mathbb{R}^n$ are not preserved in the projection, can be removed by applying a filter $\hat{b} : \mathbb{R}^2 \mapsto \mathbb{R}^2$. Next, a set of regions onto the 2D plane is defined as a map $\hat{r} : \mathbb{R}^2 \mapsto \mathcal{R}$. This can be done manually or automatically. Later, a variability measure (such as variance) is computed for every region as $\hat{v} : \mathcal{R} \mapsto \mathbb{R}^m$, where m is the dimension of the variability measure. Given a set of regions $\mathcal{R}$ enriched with variability measures, we group regions with similar variability measurements into clusters $\mathcal{C}$, a map $\hat{g} : \mathbb{R}^m \mapsto \mathcal{C}$. The last step is to map data to graphs. Putting all together, our mapping can be described as:

$$\mathbb{R}^n \xmapsto{\hat{b}\circ\hat{p}} \mathbb{R}^2 \xmapsto{\hat{r}} \mathcal{R} \xmapsto{\hat{v}\circ\hat{p}^{-1}} \mathbb{R}^m \xmapsto{\hat{g}} \mathcal{C} \mapsto \text{RGB}. \quad (1)$$

This mapping can be improved by exchanging some of its steps without requiring a new data flow. The whole pipeline is shown in Fig. 2. Next, we describe each stage in more details.

A. Multidimensional Projection

The multidimensional projection (MP) step maps high-dimensional data onto a two-dimensional visual space. In our implementation, we use the Local Affine Multidimensional Projection (LAMP) [6], although other projection techniques can be used. Besides preserving neighborhood structures, LAMP allows changes to the projected data by direct manipulation of the projected points.

LAMP uses a set of control points to perform the mapping of a set of high-dimensional data $\mathcal{X}$ to the two-dimensional space. The set of control points is typically a small subset $\mathcal{X}_S \subset \mathcal{X}$ whose counterpart $\mathcal{Y}_S$ in the visual space is known a priori ($\mathcal{X}_S$ can be mapped to the visual space using distance preserving optimization scheme as proposed in [26]). The mapping of each instance $x \in \mathcal{X}$ to a point y in the visual space is carried out by finding the best affine transformation $y = f_x(p) = pM + t$ that minimizes:

$$\sum_i \alpha_i \| f_x(x_i) - y_i \|^2 \text{ subject to } M^{\mathrm{T}}M = I, \quad (2)$$

where the matrix M and vector t are unknowns, I is the identity matrix, $x_i \in \mathcal{X}_S$ is the i-th control point, $y_i \in \mathcal{X}_S$ is the mapping of x_i in the visual space, and $\alpha_i = 1/\|x_i - x\|^2$ is a scalar weight. The orthogonality constraint $M^{\mathrm{T}}M = I$ enforces that the resulting affine transformation behaves like a rigid transformation, thus preserving distances as much as possible and ensuring that errors introduced during the positioning of control points are not drastically propagated during the projection step. We refer the interested reader to [6] for details.

B. Projection Filtering

The projection technique should preserve as much as possible the neighborhood relationships among data instances while reducing them from high space to points onto the visual space. In some cases, however, some points may be poorly projected, creating misplaced structures that jeopardize the original relations. To overcome this situation, we used the Smooth Neighborhood Preservation (SNP) [27] quality metric to guide the user. For each instance $x \in \mathbb{R}^n$ and its corresponding projection $y \in \mathbb{R}^2$, SNP takes into account

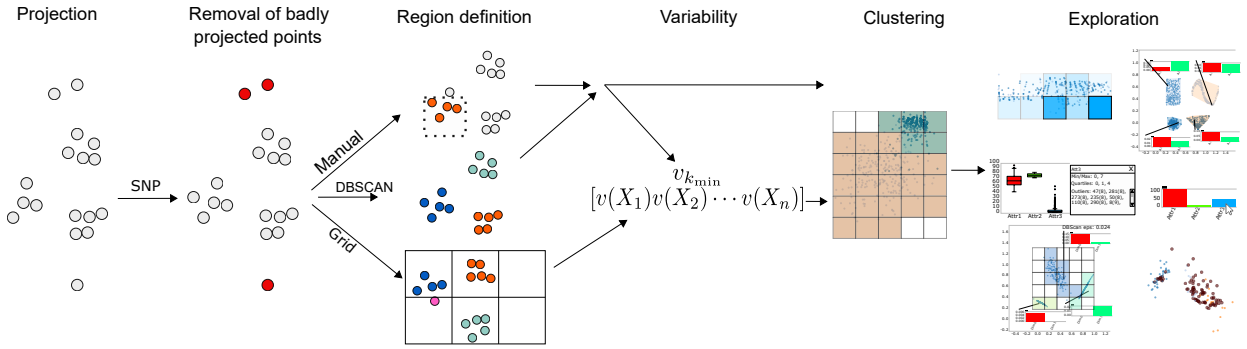

Fig. 2. Pipeline stages. First, the original data is projected onto the visual space using an MP technique. Second, the user may remove poorly projected points. Later, clusters group regions of interest based on some variability measure. The final step consists of interaction, exploration, and analysis.

false negatives and false positives to compute a quality score in the range $[0, 1]$. False negatives are the neighbors of x that were not mapped onto the neighborhood of y, while false positives are the instances that do not belong to the neighborhood of x and were mapped as neighbors of y. As a poorly projected point y receives a lower score, we use it to remove the corresponding instance x from any variability computation while keeping y in the visualization. The user decides the confidence level he/she is comfortable with (see Fig. 3).

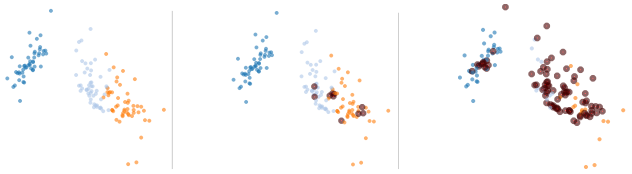

Fig. 3. Poorly projected points (red) are disregarded from the analysis. From left to right: $1, 0.5$ and 0.25 as quality confidence.

C. Defining Regions

We define "regions of interest" (henceforth "regions") as the portions of the visual space where the analysis will be held. Our system provides three ways for defining regions (see Fig. 4): manual selection, automatic selection via DBSCAN [28], and automatic selection via 2D uniform grid. We discard from analysis regions containing only 3 points or less, and outliers.

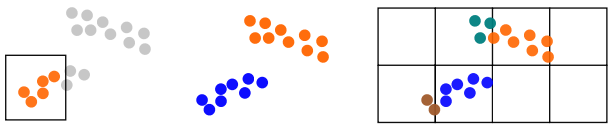

Fig. 4. Left to right: manual selection, DBSCAN, and uniform grid.

DBSCAN can be used to automatically build regions of data in arbitrary dimensions. Given a set of points in $\mathbb{R}^n$, the algorithm defines points that are ϵ-close as regions if they have at least δ points, therefore assuming sparse points as outliers. One advantage of this technique is that DBSCAN can run in two different ways: either grouping points based on the pair (ϵ, δ) or until it reaches k regions.

The 2D uniform grid can be useful in cases where the input data does not make well-defined data groups. In this case, a region is defined by the set of points that fall within a single grid cell. With this approach, the segmentation of visual space into uniform grid cells and further analysis provide an overview of how attributes behave over the projected points on a per-cell basis, such that relevant attributes responsible for local point dispersion can be seen. With such approach, we notice that the analysis is dependent on the grid size and how points are projected. To mitigate those issues, we allow the user to control the grid cell size, such that increasing this parameter (and decreasing the number of grid cells) will reveal important attributes over a wider region. Also, our system automatically runs PCA [4] over the projected points to re-orient the projection, making the two principal components point to the same direction of principal axes of the visual space (see Fig. 5).

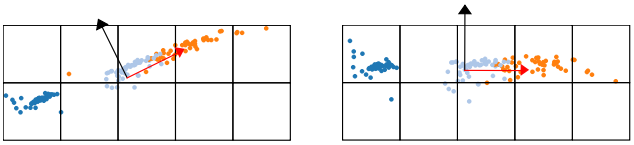

Fig. 5. The uniform grid leads to different regions depending on how the data are projected. The left image shows how regions would be created if the projected points were rotated 45°. On the right, we mitigate this problem by re-orienting the projected points such that the principal component (red arrow) has the direction of the horizontal axis of the visual space.

Regions defined by manual selection are useful either the inputted data do not form well-structured groups or users have previous knowledge on the dataset and want to analyze a specific set of regions from the projected points. As proposed, the user may create regions by directly selecting points or using the uniform grid to specify region cells.

D. Variability Computation

Given a set of regions $r_i \in \mathcal{R}$ from the previously step, we compute two variability measures, namely, k-minimum variance and variance vector. Other statistical measures can be also applied. We define relevant attributes in terms of variance

since it is directly related to the Euclidean distance, which is the metric used in most projections techniques and assumed in this paper.

Assume the values of the a-th attribute of the points contained in a region r_i is an independent random variable X_i^a, $1 \leq a \leq n$. The variance of each attribute is defined as $v(X_i^a) = E((X_i^a - E(X_i^a))^2)$, being E the expected value. The k-minimum variance variability of region r_i is defined as $v_{k_{\min}} = \min_k\{v(X_i^a)\}$. In attempt to correlate all region attributes simultaneously, the variance vector is defined as $\mathbf{x} = [v(X_1)\ v(X_2)\ \cdots\ v(X_n)]$. We normalized the attributes on both variabilities to fairly compare them.

When using the regular grid, the user has the option to consider as neighbors of a point $x \in r_i$ not only the other points into the cell r_i, but also those e.g. inside the influence radius of some kernel function [29], [30] centered at x. One can see this as a fair approach to correlate projected points. Thus, when computing the variance of the cell r_i, we use the k nearest neighbors (KNN) of each point $x \in r_i$ to perform this smoother analysis. The idea is to mitigate problems where close projected points are apart from each other by an edge of neighboring cells.

E. Clustering

This stage may be skipped if regions are created manually or by using DBSCAN. In such cases, the regions themselves may be considered as clusters in which subsequent analysis will be performed. Otherwise, the user selects a variability measure to be applied to $\mathcal{R}$ to form clusters $\mathcal{C}$ with similar variance. As proposed, clusters may be disjoint groups of regions.

When the k-minimum variance variability is selected, regions are grouped accordingly to $\operatorname{argmin} v_{k_{\min}}$, i.e., regions in which the a-th attribute defines the k-th minimum variance are identified as beginning a group. The user can choose different values for k to see different levels of details. As proposed, variability is mapped to colors in a categorical scale. As an example, we apply the k-minimum variance variability on the Haberman's Survival three-dimensional dataset [31]. Using this variability, each attribute color, red, green and blue in Fig. 6 represents, respectively, the following dimensions: Age (patient's age at the time of operation), Year (patient's year of operation), and Positive (number of positive axillary). This analysis can be useful to identify which attributes are more relevant in each region.

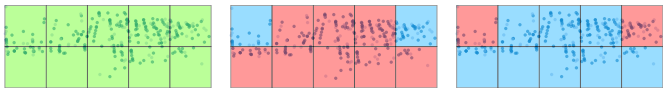

Fig. 6. Each color represents an attribute of the data. Left: 1-minimum variance identifies that the Age attribute has the lowest variance over all regions. Middle: the 2-minimum variance reveals that the Year attribute is rather relevant. Right: the Positive attribute has the greatest variability per region (3-minimum variance).

In addition, users can see the weight of individual attributes on the dispersion of all projected points. For instance, suppose one is interested in the variability of the a-th attribute per region. In this case, only one cluster is formed, and the color transparency of the selected attribute is used to show how much such attribute contributes to the points' dispersion in each region. Regions with low dispersion are more opaque (see Fig. 7).

Fig. 7. Individual attribute relevance over all regions. Stronger colors indicate regions where the selected attribute vary the least.

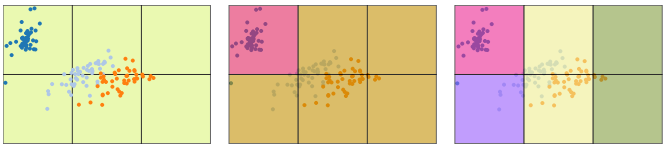

Fig. 8. Groups formed by the variance vector clustering algorithm. New groups are created as the user decreases the DBSCAN ϵ value, identifying similar variance regions.

On the other hand, if the variance vector variability is selected, we propose a novel algorithm to cluster regions based on their variance attributes, called variance-based clustering. We use DBSCAN with the cosine similarity (more suitable to compare vectors) to group regions based on the pair $(\epsilon, 0)$, $\epsilon \in [\epsilon_-, \epsilon_+]$, where ϵ_-, ϵ_+ means the smaller and bigger edge lengths to create 1 and $|\mathcal{R}|$ clusters, respectively. Fig. 8 exemplifies the clustering of regions with attributes simultaneously similar to each other. As the user decrease ϵ, new clusters are formed with even more similarity. Using this variability, each cluster receives a randomly selected color (with no relation to any attribute color).

F. Visualization

Our layout follows traditional projection-based visualization techniques. A 2D canvas is used to display the results of the multidimensional projection. The user can zoom in and out of the projected data, select and move data points, and visualize attributes of each data point x.

The user can also use GMaps [32] to represent each cluster boundary as illustrated in Fig. 9. While not being essential, this mode of visualization provides a way to see clusters more continuously than if the uniform grid was used.

After the projection, variability definition, and clustering, the user can study the cluster properties by using bar charts and boxplots. The bar chart and boxplots show the variance and mean of each individual dimension per region. In addition, the boxplot shows the interquartile range and outliers per dimension. Both are linked so that hovering on one dimension on the bar chart details its statistical information on the boxplot (see Fig. 10). They can be applied to both clusters and individual regions.

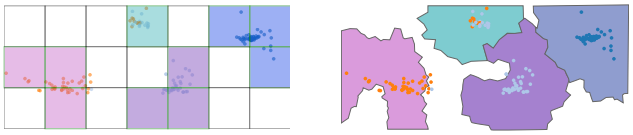

Fig. 9. Left: visualization of regions with similar variance using the 2D uniform grid. Each cluster is mapped to a different color. Right: instead of displaying uniform grid cells, one can represent regions with smoother transactions with blob-like shapes.

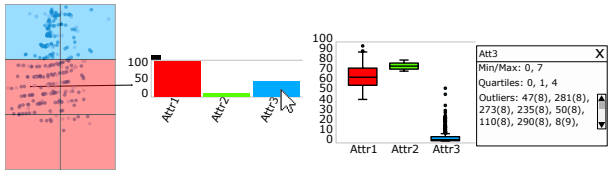

Fig. 10. An example of statistical details of the red cluster.

IV. Results and Validation

A. Validation Experiment I – Synthetic Dataset

We apply the pipeline described in Section III to a synthetic dataset. Our visualization describes the content of the projected data whenever the projection is successful. Otherwise, it describes the variability introduced by the projection itself. To show that, we perform a quantitative evaluation using a synthetic dataset as follows.

We build the synthetic dataset with n attributes (dimensions) and $|\mathcal{C}| = 2^n - 1$ distinct clusters, being $\mathcal{C}$ the set of clusters in the input data. Each cluster $c \in \mathcal{C}$ is generated by sampling a multivariate Normal distribution $N(\mu_c, \Sigma_c)$ centered at μ_c and covariance Σ_c. Even though each point $x \in c$ contains n attributes ($x \in \mathbb{R}^n$), we defined only a combination of dimensions of x to be nonzero. For example, for $n = 3$, let X^a be the set formed by the a-th attribute, we obtain seven sets of attributes that are nonzero, namely $\{X^1\}, \{X^2\}, \{X^3\}, \{X^1, X^2\}, \{X^1, X^3\}, \{X^2, X^3\}$ and $\{X^1, X^2, X^3\}$. By doing so we guarantee to have a cluster

Fig. 11. Projection of two distinct, normally distributed points in $\mathbb{R}^3$. The left (resp. right) dataset composed by several sparse (resp. dense) clusters (see Section IV-A for details).

in each possible subspace of the $\mathbb{R}^n$. The covariance matrix is built by setting $\Sigma_c = UDU^{-1}$, where U is a random unitary matrix and D is a random diagonal matrix containing the eigenvalues of Σ_c. We generate two datasets: one with dense cluster (i.e., low eigenvalues) and another with sparse clusters, as shown in Fig. 11. We expect the denser cluster to be projected with fewer errors, allowing our visualization to reveal the attributes that contribute to cluster dispersion in each dimension.

Let us take the second dataset as an example. Recall that the dataset consists of seven well-defined clusters. To find those groupings, we create regions with the uniform grid and applied the vector-based clustering, as illustrated in Fig. 12. All seven clusters were correctly found. The bar charts show the combination of attributes that defines each cluster. For highly-correlated points (leftmost plots) the projection can easily distinguish clusters. As the groups begin to share some attributes with same distribution (rightmost clusters), the projection loses accuracy.

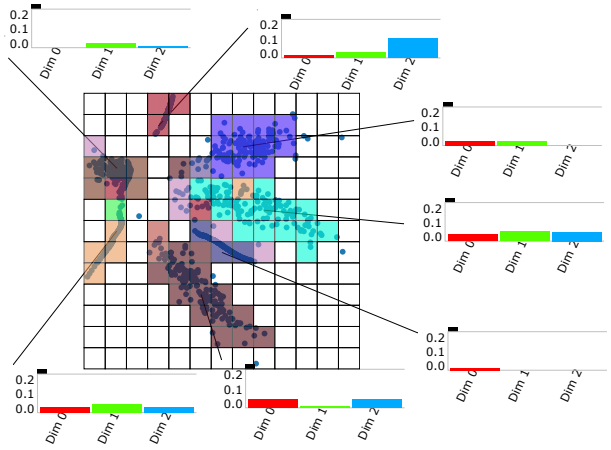

Fig. 12. Clustering detection and analysis. The plot shows how the attributes vary in each cluster.

B. Validation Experiment II – Wisconsin Breast Cancer

We have used the well-known Wisconsin Breast Cancer (Original) dataset in our second experiment. The dataset contains 699 instances of both malignant (cancerous) and benign (non-cancerous) cells. For this example 16 instances were removed, for lack of information. Table I formalizes all cell attributes based on the features described in [33] (σ denotes standard deviation).

TABLE I
Benign and Malign Cells Attribute

Attribute	Malign	Benign
Clump Thickness	high σ	low σ
Uniformity of Cell Size	high σ	low σ
Uniformity of Cell Shape	high σ	low σ
Single Epithelial Cell Size	high value	low value
Marginal Adhesion	high value	low value
Bare Nuclei	high value	low value
Bland Chromatin	high σ/value	low σ/value
Normal Nucleoli	high value	low value
Mitoses	high value	low value

As observed, both cell types present distinguish variance patterns, the ideal scenario for the vector-based clustering,

which has correctly found the regions belonging to each cluster, as illustrated in Fig. 13. The user can understand the set of relevant attributes responsible for the points' similarities in each cluster, as well as the outliers in each dimension. The visualization also tells that the Clump Thickness attribute presents less variance and lower values in normal cells, while the Uniformity of Cell Size and the Uniformity of Cell Shape attributes present higher variance on cancerous cells. In addition, the excessive number of outliers noted in the Mitosis attribute, for both cells types, suggests that the procedure applied to acquire/measure such attribute may be rather imprecise and should be reevaluated.

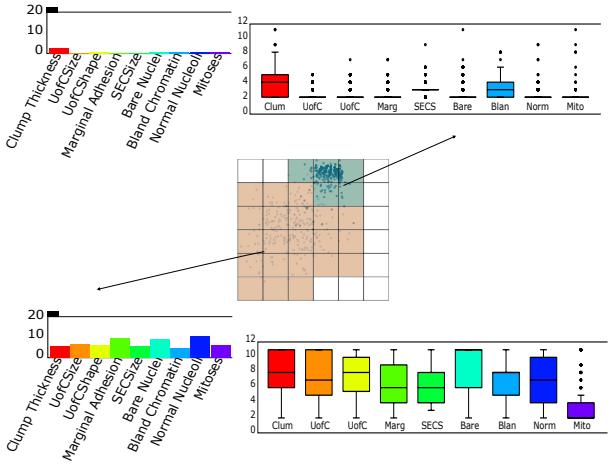

Fig. 13. Our algorithm has found two clusters, identified in the plot by green and beige colors. Each box plots and bar charts gives more details of how distributed attributes are in each grouped data. For viewing them, the user can easily understand, for instance, that normal and cancerous cells differ in almost all dimensions, being the determinant dissimilarity caused by the Normal Nucleoli and Bare Nuclei attributes.

As shown by the bar charts and box plots, all attributes information, such as variance, mean and values, are corresponding to those expected, and the visualization can explain why and where the clusters differ the most, being the Normal Nucleoli and Bare Nuclei the major attributes to differing one cell type from each other.

C. Validation Experiment III – Inverse Anscombe's Quartet

We also propose the analysis of a dataset collected from a user experiment. This synthetic dataset contains three classes of objects, namely, A, B and C, where A is a group of points created from a uniform distribution on the open interval $(0, 1)$, and B and C are groups of points formed by scaling the points in A by factors r and $2r$, respectively, $r \neq 0$. The idea is to create three different groups of the same variance, but indistinguishable for the multidimensional projection, since the distances among points in each group are the same. Fig. 14 shows the groups clustered by manual selection, using $r = 10$.

The motivation behind this study is to show the inverse scenario of the Anscombe's quartet [34], where identical visualizations represent different data. The aim is to highlight that exploration of multidimensional projections can be improved by using statistical analysis, which is provided by our tool. In this context, the bar charts reveal why a group is equal to each other (same variance), but also shows why they are projected separately (different mean).

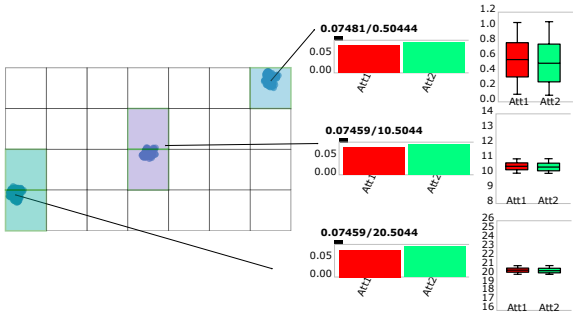

Fig. 14. All groups present the same individual distance among their points. Despite the correctness of the projection, the variance by itself cannot clarify what distinguishes one group from another. As an example, above the red bars are shown the values of the variance/mean of the corresponding attributes.

D. Case Study

We have applied our clustering method and exploratory techniques in the Wine dataset collected from the UCI Machine Learning Repository [35]. This dataset is formed by 178 instances of three types of wine, each one having thirteen components. The Wine dataset results from a chemical analysis of wines from the same region in Italy but derived from three different wineries.

The vector-based variability has correctly identified the three classes of wine (Fig. 15). The analysis of the boxplots (Fig. 16) of each dimension shows that the Magnesium and Proanthocyanins attributes contain a large number of outliers, suggesting those attributes are not relevant enough to adequately represent the different types of wine. The user could reevaluate the process of obtaining such data attributes or perform further analysis without those dimensions.

The use of the uniform grid together with the k-minimum variance can also be applied to reveal other types of information. Fig. 17 shows the relevant dimensions of three individual cells (top), confirming the idea that attributes variances are directly related to data similarities, and as a consequence, to distances among projected points (see points dispersion in each cell). The bar charts (bottom) show that the Malic Acid attribute varies accordingly with the k-minimum variability opacity in all regions.

V. Discussions

It is worth mentioning that the region definition stage plays an important role in the analysis since statistical measures will be applied on the subsequently formed regions. Consider regions defined by a 2D grid for instance: different cell sizes may lead to different analysis as more or fewer information details will be captured in each region. However, our tool allows user interaction in order to tackle this issue, since he/she

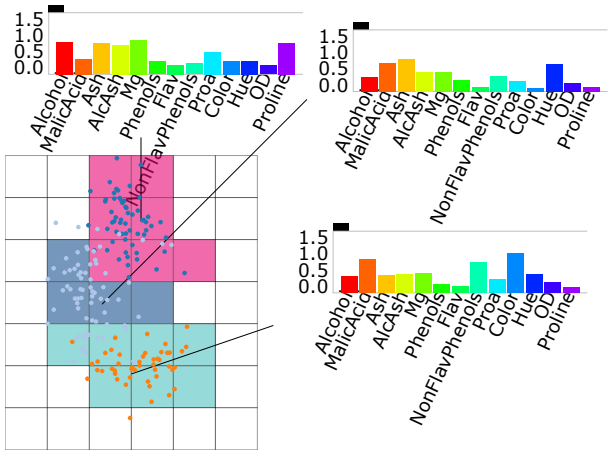

Fig. 15. Vector-based clustering. The bar charts show the main differences among groups of wine. For instance, if a sampled group of wines presents high variance (more than 0.5) for the attributes Alcohol and Proline, this sample can be labeled as the first (superior) type of wine. In the same way, the remaining groups (middle and inferior) can be identified by the lower variance of those attributes, being differentiated by low and high variance in the Color attribute, respectively.

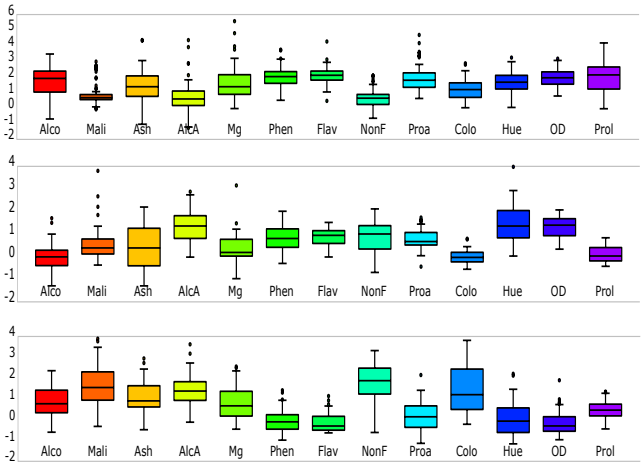

Fig. 16. Boxplots of the Wine dataset attributes.

can appropriately adjust the cell size, in this case, as well as any other parameters involved in the analysis, such as the values of KNN and DBSCAN ϵ. Fig. 18 illustrates the idea. The higher the KNN, the most likely each region variance tends to be equal to each other. The higher the DBSCAN ϵ, the lower the number of clusters.

Future work includes other statistical measures in order to improve analysis and cluster comparison, such kurtosis and skew, covariance matrices and PCA. Functionalities to automatically remove outliers or irrelevant dimensions can be also investigated, as well as space-filling-based strategies (e.g. [36]) to smoothly visualize relevant attributes over the projection.

VI. Conclusion

Multidimensional Projection (MP) techniques are useful to find patterns and correlate multidimensional data instances.

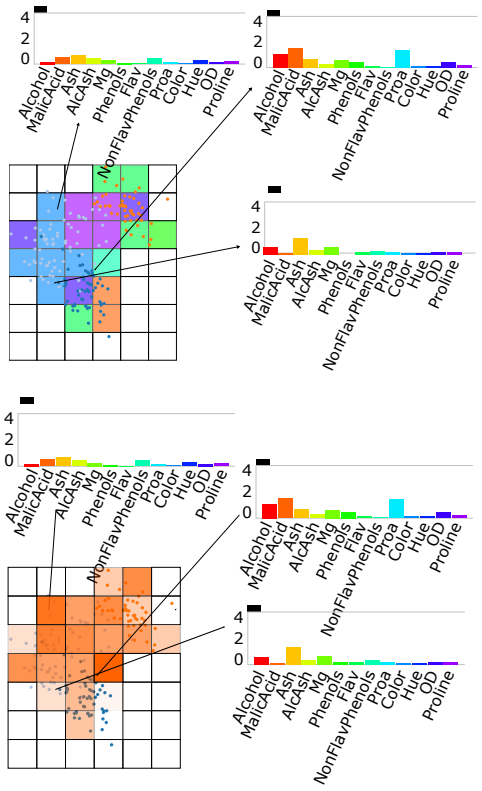

Fig. 17. Most relevant dimensions (top) and specific relevance of Malic Acid attribute (bottom). The k-minimum variance variability can reveal how attributes behave in each cluster, while the bar charts show an overview of all attributes simultaneously.

However, being able to visualize the original data relations may not be enough to user analysis. Indeed, many questions arise after projection, such as "what are the relevant attributes that contribute to cluster formation or point dispersion?". In order to help answer this question, we have proposed a pipeline to discretize projected data into regions and group them in clusters according to attribute variability. We mainly focus on using attribute variance, since it is directly related to the Euclidean distance, a metric largely used in MP techniques.

The proposed pipeline summarizes our visualization tool, a mixture of plots and traditional statistical measures to improve MP-based analysis. We have presented experiments and a case study that demonstrate our technique can correctly identify important sets of attributes over the projected data, which may be used to explain clusters and points dispersion.

Acknowledgment

This paper is based upon projects sponsored by FAPESP (São Paulo Research Foundation), grants #2013/15928-9 and #2011/22749-8, and CNPq (National Counsel of Technological and Scientific Development), Brazil, grant #302643/2013-3. Special thanks go to Tiago Etiene for his valuable contributions to this paper.

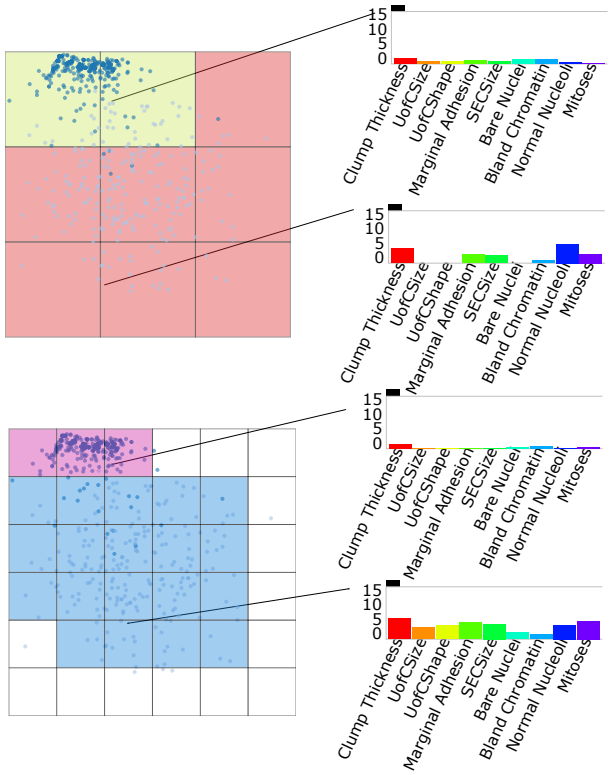

Fig. 18. When using the 2D grid, different cell sizes can lead to different analysis. However, by adjusting the KNN and DBSCAN ϵ parameters the user can achieve similar results.

REFERENCES

[1] R. A. Becker and W. S. Cleveland, "Brushing scatterplots," *Technometrics*, vol. 29, no. 2, pp. 127–142, 1987.

[2] A. Inselberg and B. Dimsdale, "Parallel coordinates: A tool for visualizing multi-dimensional geometry," in *VIS*, 1990, pp. 361–378.

[3] T. F. Cox and M. Cox, *Multidimensional Scaling*, 2nd ed. Chapman and Hall/CRC, 2000.

[4] I. Jolliffe, *Principal Component Analysis*. Springer, 1986.

[5] F. V. Paulovich, L. G. Nonato, R. Minghim, and H. Levkowitz, "Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping," *TVCG*, vol. 14, no. 3, pp. 564–575, 2008.

[6] P. Joia, F. Paulovich, D. Coimbra, J. Cuminato, and L. Nonato, "Local affine multidimensional projection," *TVCG*, vol. 17, no. 12, pp. 2563–2571, 2011.

[7] F. V. Paulovich, D. M. Eler, J. Poco, C. P. Botha, R. Minghim, and L. G. Nonato, "Piecewise Laplacian-based projection for interactive data exploration and organization," *CGF*, vol. 30, no. 3, pp. 1091–1100, 2011.

[8] J. Choo, C. Lee, C. Reddy, and H. Park, "Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization," *TVCG*, vol. 19, no. 12, pp. 1992–2001, 2013.

[9] F. Paulovich, F. Toledo, G. Telles, R. Minghim, and L. Nonato, "Semantic wordification of document collections," *CGF*, vol. 31, pp. 1145–1153, 2012.

[10] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma, "Semantic-preserving word clouds by seam carving," *CGF*, vol. 30, no. 3, pp. 741–750, 2011.

[11] E. Gomez-Nieto, W. Casaca, D. Motta, I. Hartmann, G. Taubin, and L. G. Nonato, "Dealing with multiple requirements in geometric arrangements," *TVCG*, vol. 22, no. 3, pp. 1223–1235, 2016.

[12] E. Gomez-Nieto, F. San Roman, P. Pagliosa, W. Casaca, E. Helou, M. Oliveira, and L. Nonato, "Similarity preserving snippet-based visualization of web search results," *TVCG*, vol. 20, no. 3, pp. 457–470, 2014.

[13] E. Gansner, Y. Hu, and S. North, "Visualizing streaming text data with dynamic maps," *arXiv preprint*, 2012.

[14] M. Steiger, J. Bernard, S. Mittelstädt, H. Lücke-Tieke, D. Keim, T. May, and J. Kohlhammer, "Visual analysis of time-series similarities for anomaly detection in sensor networks," *CGF*, vol. 33, no. 3, pp. 401–410, 2014.

[15] E. Kandogan, "Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations," in *VAST*, 2012, pp. 73–82.

[16] P. Joia, F. Petronetto, and L. Nonato, "Uncovering representative groups in multidimensional projections," *CGF*, vol. 34, no. 3, pp. 281–290, 2015.

[17] B. Broeksema, A. Telea, and T. Baudel, "Visual analysis of multi-dimensional categorical data sets," *CGF*, vol. 32, no. 8, pp. 158–169, 2013.

[18] C. Turkay, P. Filzmoser, and H. Hauser, "Brushing dimensions - a dual visual analysis model for high-dimensional data," *TVCG*, vol. 17, no. 12, pp. 2591–2599, 2011.

[19] X. Yuan, D. Ren, Z. Wang, and C. Guo, "Dimension Projection Matrix/Tree: Interactive Subspace Visual Exploration and Analysis of High Dimensional Data," *TVCG*, vol. 19, no. 12, pp. 2625–2633, 2013.

[20] J. Seo and B. Shneiderman, "A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections," in *INFOVIS*, M. O. Ward and T. Munzner, Eds., 2004, pp. 65–72.

[21] N. Cao, D. Gotz, J. Sun, and H. Qu, "DICON: Interactive visual analysis of multidimensional clusters," *TVCG*, vol. 17, no. 12, pp. 2581–2590, 2011.

[22] F. Aurenhammer, "Voronoi diagrams - a survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, 1991.

[23] B. Shneiderman, "Tree visualization with tree-maps: A 2-d space-filling approach," *ACM TOG*, vol. 11, pp. 92–99, 1991.

[24] N. Henry, J. D. Fekete, and M. J. McGuffin, "Nodetrix: a hybrid visualization of social networks," *TVCG*, vol. 13, no. 6, pp. 1302–1309, 2007.

[25] J. Stahnke, M. Drk, B. Mller, and A. Thom, "Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions," *TVCG*, vol. 22, no. 1, pp. 629–638, 2016.

[26] E. Tejada, R. Minghim, and L. G. Nonato, "On improved projection techniques to support visual exploration of multidimensional data sets," *Information Visualization*, vol. 2, no. 4, pp. 218–231, 2003.

[27] P. Pagliosa, F. V. Paulovich, R. Minghim, H. Levkowitz, and L. G. Nonato, "Projection inspector: Assessment and synthesis of multidimensional projections," *Neurocomputing*, vol. 150, pp. 599–610, 2015.

[28] M. Ester, H. peter Kriegel, J. S, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." AAAI Press, 1996, pp. 226–231.

[29] M. D. Buhmann and M. D. Buhmann, *Radial Basis Functions*. Cambridge University Press, 2003.

[30] D. A. Fulk and D. W. Quinn, "An analysis of 1-d smoothed particle hydrodynamics kernels," *J. Comp. Physics*, vol. 126, no. 1, pp. 165–180, 1996.

[31] S. J. Haberman, "Generalized residuals for log-linear models," in *Proc. of the 9th International Biometrics Conference*, 1976, pp. 104–122.

[32] E. Gansner, Y. Hu, and S. Kobourov, "GMap: Visualizing graphs and clusters as maps," in *PacificVis*, 2010, pp. 201–208.

[33] G. I. Salama, M. B. Abdelhalim, and M. A. elghany Zeid, "Breast cancer diagnosis on three different datasets using multi-classifiers, int," J. of Comput. and Inform. Technology, Tech. Rep., 2012.

[34] F. Miller, A. Vandome, and J. McBrewster, *Anscombe's Quartet*. VDM Publishing, 2010.

[35] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[36] R. M. Martins, D. B. Coimbra, R. Minghim, and A. Telea, "Visual analysis of dimensionality reduction quality for parameterized projections," *C&G*, vol. 41, pp. 26–42, 2014.