EuroVis Workshop on Visual Analytics (2015)
E. Bertini and J. C. Roberts (Editors)

Attribute-based Visual Explanation of Multidimensional Projections

Renato R. O. da Silva[1,2] and Paulo E. Rauber[1] and Rafael M. Martins[2] and Rosane Minghim[2] and Alexandru C. Telea[1]

[1] Institute Johann Bernoulli, University of Groningen, the Netherlands
[2] Institute of Mathematics and Computer Sciences, University of São Paulo, Brazil

Abstract

Multidimensional projections (MPs) are key tools for the analysis of multidimensional data. MPs reduce data dimensionality while keeping the original distance structure in the low-dimensional output space, typically shown by a 2D scatterplot. While MP techniques grow more precise and scalable, they still do not show how the original dimensions (attributes) influence the projection's layout. In other words, MPs show which points are similar, but not why. We propose a visual approach to describe which dimensions contribute mostly to similarity relationships over the projection, thus explain the projection's layout. For this, we rank dimensions by increasing variance over each point-neighborhood, and propose a visual encoding to show the least-varying dimensions over each neighborhood. We demonstrate our technique with both synthetic and real-world datasets.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces—

1. Introduction

The analysis of multidimensional data is important in many areas like text mining and business intelligence. Such datasets have hundreds (or more) data points or observations, each having n (tens up to hundreds of) measured *dimensions* or attributes. Multidimensional projections (MPs) are often used for tasks such as finding groups of similar observations. MPs project the nD observations to a low-dimensional space, typically 2D, keeping the distance-structure of the original nD points as much as possible. Visualizing the 2D MP output by *e.g.* scatterplots lets one easily find groups of similar points, correlations, and outliers [SMT13]. While such visualizations tell us *which* points are similar, they do not tell us *why*.

We address this by enriching 2D MP scatterplots with *explanatory* visuals that highlight the key dimensions that make closely-projected points similar. We compute such explanations over all point neighborhoods of a 2D projection and render them by image-based techniques. The result is a smooth color-and-luminance map that partitions the 2D projection into same-explanation regions. A level-of-detail parameter allows controlling the scale at which explanations are provided and also to filter out noise. We demonstrate our method on several real-world high-dimensional datasets.

2. Related Work

For a dataset $D = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\} \in \mathbb{R}^n$ of N n-dimensional elements $\mathbf{p}_i = (\mathbf{p}_i^1, \ldots \mathbf{p}_i^n)$, a multidimensional projection (MP) performs the transformation $P : \mathbb{R}^n \to \mathbb{R}^m$, where m is a low-dimensional space, typically 2. The projected elements $D^P = \{\mathbf{q}_i = P(\mathbf{p}_i \in D)\}$ are next typically shown as an mD scatterplot. Explaining multidimensional projections is an important problem in visual analytics. Methods designed to this end can be classified in three groups, as follows.

Quality maps explain projections by showing how much they capture the similarity structure of the nD data. Schreck et al. compute a score for each projected point based on the *stress* measure of its neighborhood, and create a continuous 2D error map showing how the projection error varies [SvLB10]. This idea is extended to maps showing false neighbors (points projected too closely to their neighbors) and missing neighbors (points too far away from their neighbors) [MCMT14]. Pagliosa et al. color-code the value of quality measurements on a family of projections and their interpolation, to show differences between projections *vs* a particular error distribution and also to provide alternative mappings between distinct projections [PPM*15]. Showing the type and distribution of projection errors gives detailed insight in a projection's *quality* with little or no user intervention. However, assume a good-quality projection in which we see several dense point-groups: Such techniques do not tell us *what* these groups mean.

Interactive approaches explain MPs by showing additional information on-demand on user-selected point groups to help one define their meaning. The simplest such tool shows the dimension values of the point under the mouse in a tooltip. By brushing a point-group, one can see which dimensions are most similar and thus likely capture the group's meaning. ForceSPICE uses a force-directed spring model to lay out a scatterplot of textual elements [EFN12]. The content similarity of each document can be further inspected and the user can incrementally add annotations over the layout or highlight specific text words. These actions update the spring model to change the layout, to better reflect the user's mental model. Cuadros et al. [CPMT07] use a phylogenetic tree algorithm to project documents by placing similar ones in close nodes of the tree. Next, users can execute a topic extraction algorithm which automatically labels selected tree branches to guide exploration. Such approaches explain an MP on several levels of detail, but require user interaction effort to specify *where* to explain the projection.

Clustering can be used to separate the nD data into closely-related point groups. Projecting clusters instead of individual points creates various multi-level visualizations where each projected cluster can be potentially explained by one or a few 'representative elements' drawn atop of it using glyphs. ImageHIVE [TSLX12] applies this idea by defining clusters from a collection of images. Using representatives of each cluster, a graph is created based on

the nD distances between images, which is next drawn in 2D using a graph layout technique. A Voronoi diagram is used to show the representatives' contents. Multi-level maps are also used to visualize documents [NB12]. The document corpus is projected and clustered by a hierarchical clustering method. Cluster representatives are used to create a Voronoi diagram filled with representative words. Showing representatives, however, does not explain, in terms of attributes or dimensions, why documents are placed together. Kandogan [Kan12] visually annotate clusters occurring in scatterplots based on the attribute trends detected in them. Clusters are computed by an image-based scatterplot density estimation. Important attributes are identified based on their statistical relevance. This approach works well when the data and projection can be easily and robustly separated into several clusters, and less well when there is no such clear separation.

3. Proposed visual explanation

Since MPs place similar points closely in 2D, a natural idea is to try to explain what such closely-placed points have in common. We proceed as follows. For each 2D projected point $\mathbf{q}_i$, we define its 2D neighborhood $\nu_i^P = \{\mathbf{q} \in D^P | \|\mathbf{q} - \mathbf{q}_i\| \leq \rho\}$ as all projected points closer to $\mathbf{q}_i$ than a given radius ρ. This defines an nD neighborhood $\nu_i = \{\mathbf{p} \in D | P(\mathbf{p}) \in \nu_i^P\}$ of point $\mathbf{p}_i$. We use ν_i to compute a ranking $\mu_i = (\mu_i^1, \ldots \mu_i^n) \in \mathbb{R}^n$ for all n dimensions of $\mathbf{p}_i$. The lower a rank μ_i^j is, the better can dimension j explain the similarity of points over ν_i. Computing dimension ranks is detailed next.

3.1. Dimension Ranking

To compute the ranks μ_i, we propose two metrics: Euclidean distance contribution and dimension variance, as follows.

Euclidean ranking: We first define the contribution $lc_{\mathbf{p},\mathbf{r}}^j$ of dimension j to the squared distance between two nD points $\mathbf{p}$ and $\mathbf{r}$ as

$$lc_{\mathbf{p},\mathbf{r}}^j = \frac{(\mathbf{p}^j - \mathbf{r}^j)^2}{\|\mathbf{p} - \mathbf{r}\|^2}. \tag{1}$$

Next, for each nD point $\mathbf{p}_i$ of our dataset D, we define the local contribution of a dimension j as the average of the distance-contributions between $\mathbf{p}_i$ and all its neighbors $\mathbf{r} \in \nu_i$

$$\overline{lc_i^j} = \frac{\sum_{\mathbf{r} \in \nu_i} lc_{\mathbf{p}_i,\mathbf{r}}^j}{|\nu_i|}. \tag{2}$$

To explain a neighborhood ν, it is intuitive to highlight dimensions that contribute to similarities in ν *and* that are not similar outside ν – or in other words, dimensions that can discriminate between points inside and outside ν. For this, we first compute the global dimension contributions (gc) for the distance. This is done by defining as the focused point the nD centroid and setting all projected points as its neighborhood. We then compute the Euclidean ranking contribution as the ratio between global and local contributions. Finally, we normalize rankings to indicate the relative importance of different dimensions. Thus, the rank of dimension j for point i is given by

$$\mu_i^j = \frac{\overline{lc_i^j}/gc^j}{\sum_{j=1}^n \left(\overline{lc_i^j}/gc^j\right)}. \tag{3}$$

Variance ranking: We first compute the global variance $GV = (var(\mathbf{p}^1), \ldots, var(\mathbf{p}^n))$ of all dimensions over all points in D. Next, for each point i we compute the local variance LV_i over ν_i. As for the Euclidean metric, we want to emphasize how dimensions contribute to similarity within local neighborhood. For this, we compute the ratio between the local and the global variance, and normalize this ratio to indicate relative importance of dimensions. If we denote the j^{th} component of GV and LV_i by GV^j and LV_i^j respectively, the rank of dimension j for point i is thus given by

$$\mu_i^j = \frac{LV_i^j/GV^j}{\sum_{j=1}^n (LV_i^j/GV^j)}. \tag{4}$$

Note that, for both the Euclidean and variance ranking, low values indicate dimensions which are better for explaining a local neighborhood. Indeed, a low rank indicates more similar values for that dimension, *i.e.* a stronger cohesion of points from the perspective of the property sampled by that particular attribute.

3.2. Visual encoding

For each point i, we store a ranking vector $\{(j, \mu_i^j)\}_{1 \leq j \leq n}$ with the IDs and ranks of all its n dimensions, sorted increasingly on rank values. Next, we select the C dimensions having top ranks for most of the N points, and map their IDs to colors via categorical colormap having $C = 9$ entries. This way, dimensions which are top-rank for many points get mapped to distinct colors. Dimensions which are top-tank for few points do not get colors (due to the colormap's limited size C) and are mapped to the reserved color dark blue. Using a color coding approach on the visualization allows to quickly identify which regions are mainly explained by the same dimensions. A similar approach is used by Gleicher [Gle13] which employs a color field visualization to quickly judge the importance of a dimension in a projected space.

We also want to show the *confidence* level of a displayed top-rank dimension. We compute this confidence by analyzing the top-ranks of points in a 2D neighborhood ν_c^P centered at $\mathbf{q}_i$, and defined similarly to ν^P but using a smaller radius $\rho_c < \rho$. In detail, we sum the top-ranks of all points in ν_c^P and the value of the top-rank dimension of point i to create a new ranking vector that stores the total weights of the top-rank dimensions of ν_c^P. We define the confidence of the top-rank dimension of i as the ratio of the top-rank value in the summed ranking vector and the sum of all its rank values. Intuitively, this process acts as a smoothing filter with kernel radius ρ_c that assigns high confidence to homogeneous (same top-rank) regions and low confidence to mixed regions (having points with different top ranks). This is also why we set ρ_c to be lower than ρ: Larger ρ values allow a more robust ranking process, that is less sensitive to outliers; lower ρ_c values emphasize the variation of ranking confidence over finer scales (see also Fig. 1 discussed below).

We display the top ranks and their confidences over the projection using the dense map technique based on nearest-neighbor (Voronoi) interpolation in [MCMT14], with top-ranks encoded by color and confidences by brightness respectively. To illustrate this, we use a simple synthetic dataset of 3000 points randomly sampled from three faces of a 3D cube, and additionally perturbed by uniform spatial random noise of amplitude equal to 5% of the dataset's extent. We projected this dataset to 2D using PCA [Jol02] (since PCA is a very well known technique) and ranked all points by the variance metric. The radius parameter ρ is set to 10% of the projection diameter. The resulting explanation (Fig. 1) clearly shows that the 2D projection consists of three 'zones', corresponding to the cube's faces, each being very well explained by a single dimension (as expected). Points close to face intersections are darker, so their explanation by a *single* dimension is less confident (as expected). A global ranking histogram (Fig. 1 top-right) shows which color is assigned to which dimension, and how many points are explained best by that dimension. This shows that the point count is divided in three roughly equal parts, which is correct, given the roughly equal number of samples on the three cube faces. We provide a brush tool to interactively inspect ranks for a given point. Fig. 1 shows the brushed point and its neighborhood ν^P. A second histogram (Fig. 1 bottom-right) shows the rankings μ_i^j for the brushed point i. We see

here that the top-rank dimension (purple), corresponding to dimension 0, has variance 0, which is indeed correct, as the selected point is in the middle of a face having same values for dimension 0.

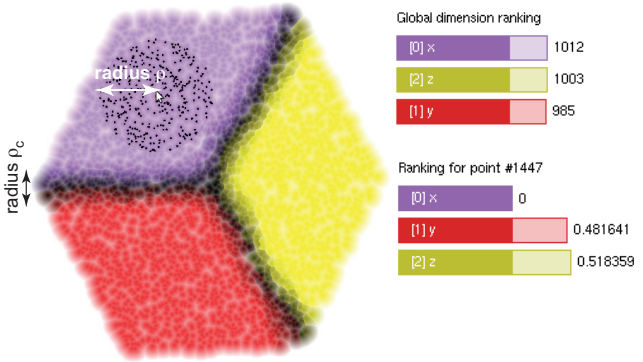

Figure 1: *Visual explanation of synthetic cube dataset.*

If the top k ranks of a neighborhood are very similar, the 'winning' top dimension may be subject to noise. Hence, using a *single* dimension to explain the similarity of a neighborhood might lead to wrong conclusions. As such, we offer the possibility of using a *dimension set* for explanations. Given a point i, its dimension-set (used for explaining the projection around i), contains all the top-ranked dimensions μ_i^j whose rank values sum up to a value equal of just larger than a user-defined small threshold value τ. Intuitively, these are all dimensions whose cumulative effect on the distance (or variance) is lower than τ. If many dimensions have low rank values, then the dimension-set will be large – meaning that we need many dimensions to explain the similarity of the neighborhood around point i. If few dimensions have low rank values, then the dimension-set will be small – in the limit case, it contains a single element, and the explanation becomes identical to the single-dimension explanation presented earlier. To visualize dimension-sets, we assign categorical colors to the C most-frequent dimension sets in the projection, and map the remaining sets by the reserved color dark blue.

4. Applications

We next use our method to explain projections from three real datasets. As projection P, we used LAMP [JPC*11] due to its accuracy and computational speed. We tried both Euclidean and variance ranking, and observed that they give very similar results in terms of which dimension (or dimension-set) is chosen for a point. The following examples use the variance metric as it is slightly more noise-resistant and faster to compute than the Euclidean metric. As parameter values we used $\rho = 10\%$ for the projection diameter (largest distance between any two points) and $\tau = 0.05$. Dimension labels were added manually on the projection to help easier indentification.

4.1. Wine quality

This dataset has 6497 samples of Portuguese *vinho verde* wine (4898 red wine; 1599 white wine) [CCA*09]. Each sample has $n = 12$ physicochemical measures like acidity, residual sugar, and alcohol rate. The projection creates a single clump (see shape in Fig. 2 a). The single-dimension explanation splits this clump into three regions defined by the top-rank dimensions *alcohol rate*, *sodium chloride/dm*^{3} and *residual sugar*, and a smaller group defined by *volatile acidity*. Zones close to region borders are dark, intuitively showing that they cannot be explained by a single dimension. Using the brush tool, we discover that the first two dimensions account for 5% of the total rankings on several areas. The dimension-set explanation (Fig. 2 b) splits the above regions into finer detail. First, *residual sugar* is split into two subregions A_1 and A_2. A_1 also include the dimensions *free sulfur dioxide* and *total sulfur dioxide* in its explanation, and A_2 also includes *total sulfur dioxide*. Hence, sulfur dioxide is closely related to residual sugar to explain these regions. Region A_3 appears in the border of two regions of the previous map, and is defined by the union of these dimensions. In subregion A_4, the dimension *wine quality* was added to the explanation, showing samples with similar quality and alcohol values. Subregion A_5 covers the union of the former *alcohol* and *volatile acidity* regions. Other subregions remain best explained by the same top-ranked dimensions since, over them, the sum of ranks between the first and second top-rank dimensions is above the threshold τ. Finally, about 12% of the points are explained by less-frequent dimension sets, mapped by the color dark blue.

4.2. Quality of software projects

This dataset describes 6773 software projects from *sourceforge.net* written in C [MSM*10]. Each project has 12 dimensions (11 software quality metrics and the project's total download count). The projection shows two large connected regions. Single-dimension explanation (Fig. 2 c) shows that the left region is best explained by dimension *total lines of code*. The right region is best explained by dimensions *total lines of code* and *lack of function cohesion*. Several small groups and a low-confidence border connect the above two regions. Dimension-set explanation shows that most subregions can be explained by two dimensions (Fig. 2 d). The left region becomes now mainly blue, showing that there are too many small-scale explanations, using *more* than one dimension, to be shown by our limited colormap. Exceptions are the subregions A_1, which adds the quality metric *number of public variables*, and A_2, which adds the metric *number of source files*, which is also related to the neighbor green region. The right region is split in several compact sub-regions: A_3 is a union between *lines of code* and *lack of function cohesion*; A_5 adds the same dimension of A_3 and also the *number of function parameters*; A_4 also adds the number of function parameters; finally, A_6 adds the metric *number of public variables* to its explanation.

4.3. US counties

This 12-dimensional dataset describes social, economic, and environmental data from 3138 USA cities [oM14]. Its projection yields a single visual cluster. Single-dimension explanation shows six main regions, mainly given by dimensions related to social statistics (Fig. 2 e). The dimension-set explanation (Fig. 2 f) splits these regions, as follows: The former *below 18* region gets split into four. One subregion (A_2) remains best explained by the *below 18* dimension. A_2 is explained by the *unemployed* and *population density* dimensions which also defined the two neighbor regions in the single-dimension explanation. A_3 is explained by the same dimensions, plus the dimension *percent of college/higher graduates*. Hence, A_3 can be seen as a more specific subset of A_2. Finally, A_1 is defined by the same dimensions as A_3, plus the dimension *median of owner-occupied housing value*, being thus an even more specialized subset of A_2. The subregion A_4 is defined by dimensions *percent of high school graduates age 25+* and *population* $\geq$ *65 years old*. Finally, the region defined by *median of owner-occupied housing value* stayed the same as the single-dimension explanation map, indicating that this dimension is sufficient to clearly define this region.

5. Discussion and conclusions

We have presented a simple and automatic technique that visually explains 2D scatterplots (created by multidimensional projections) by the names of the original dimensions.

Advantages: Our method is intuitive, easy to use, computationally efficient (runs in real-time for datasets up to 10K points on a typical PC for a C++ CPU single-threaded implementation), and generic (can use any projection and/or dataset having quantitative dimensions). The partition of the 2D projection space into same-explanation regions occurs automatically and implicitly, without the need to select or set *any* clustering parameters. Our three parameters are intuitive and simple to control: ρ acts as a scale parameter

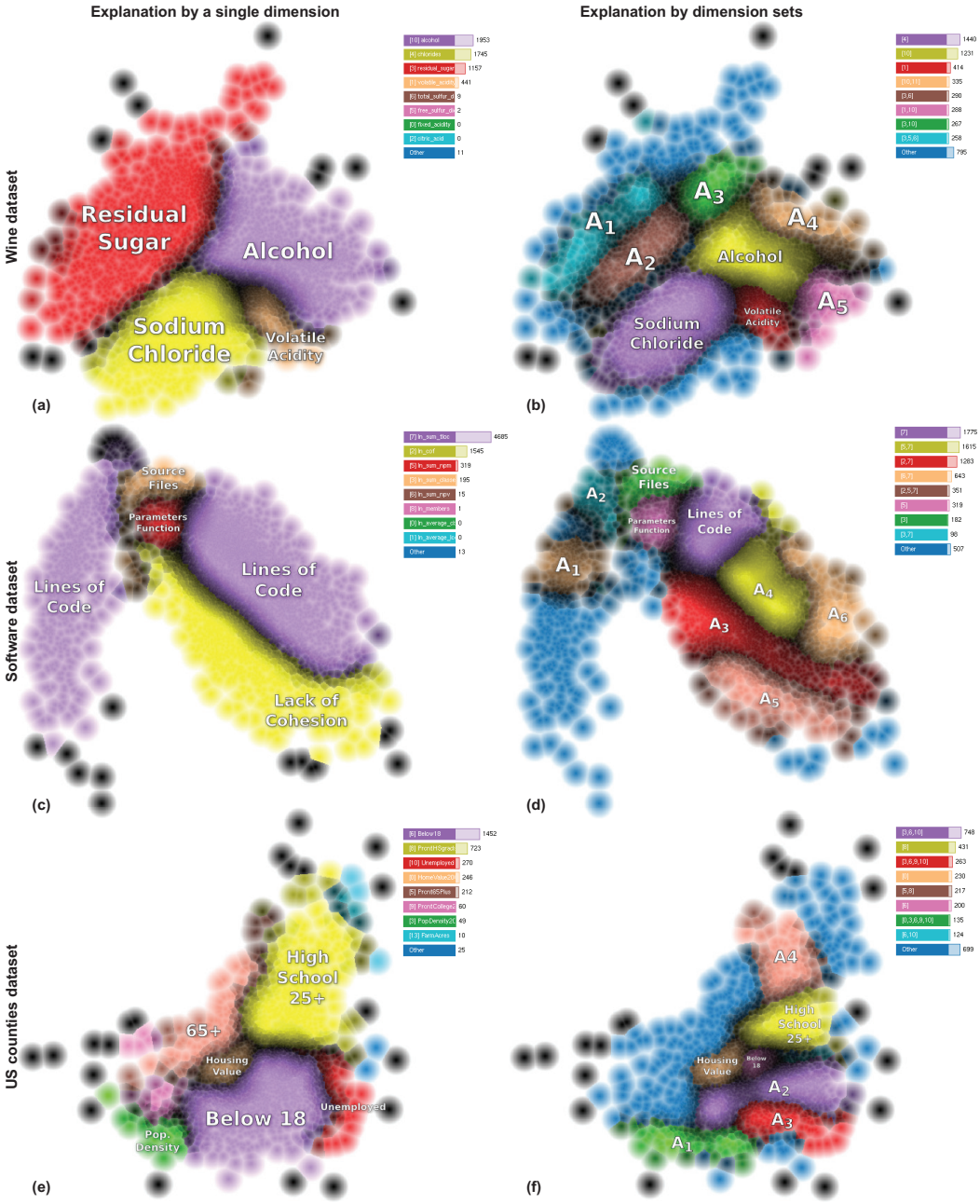

Figure 2: *Visual explanations of three datasets using a single dimension (left column) and dimension-sets (right column). See Sec. 4.*

– larger values create less regions but thicker fuzzy borders, thus a coarse scale explanation; small values create more detailed explanations and thinner region borders, but also emphasize outliers more. ρ_c acts as a smoothing filter: large values create smooth regions but thicker borders; small values create more noisy regions but thinner borders. τ controls the coherence of points in a region: large values create many strongly-coherent regions; small values create fewer less-coherent regions. In any case our approach does not need the precomputation of regions, since they are formed by the same procedure that calculates attribute ranking.

Limitations: Color-coding explanations are inherently limited to the maximum number of colors that a categorical colormap can reasonably use. This can often be less than the number of regions we can detect. Our same-color regions show which *dimensions* contribute to point proximity, but not their *values* or ranges. Finally, better explanation metrics can be envisaged for 2D neighborhoods, *e.g.* based on dimension correlations or outlier detection. Any such metric can be adapted to the application and easily added to the current implementation of our method.

We aim to extend our explanatory tools in several directions: (1) automatically segmenting same-explanation regions (our current compact same-color areas) and use automatic dimension-labeling; from that users could alternate from color to labeling for a reasonable number of dimensions; (2) explaining regions by both dimensions and dimension-values, thereby leading to more refined explanations; (3) using Shepard interpolation instead of nearest-neighbor to achieve a smoother and easier to perceive plot separation in compact regions. This has interesting connections with methods using shaded cushions to display various types of quantitative and categorical data [TE10, BT09]; (4) testing our method for datasets having hundreds of dimensions, and adapting its heuristics and parameters to compactly and intuitively explain 2D projections of such data.

Acknowledgements: This work is supported by FAPESP research financial agency, São Paulo, Brazil (grants 2011/18838-5, 2012/24121-9 and 2012/07722-9).

References

[BT09] BYELAS H., TELEA A.: Visualizing metrics on areas of interest in software architecture diagrams. In *Proc. IEEE PacificVis* (2009), pp. 33–40. 4

[CCA*09] CORTEZ P., CERDEIRA A., ALMEIDA F., MATOS T., REIS J.: Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems 47*, 4 (2009), 547 – 553. Smart Business Networks: Concepts and Empirical Evidence. URL: http://www.sciencedirect.com/science/article/pii/S0167923609001377, doi:http://dx.doi.org/10.1016/j.dss.2009.05.016. 3

[CPMT07] CUADROS A., PAULOVICH F., MINGHIM R., TELLES G.: Point placement by phylogenetic trees and its application to visual analysis of document collections. In *Proceedings of the 2007 IEEE Symposium on Visual Analytics Science and Technology (VAST)* (Oct 2007), pp. 99–106. doi:10.1109/VAST.2007.4389002. 1

[EFN12] ENDERT A., FIAUX P., NORTH C.: Semantic interaction for visual text analytics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), CHI 12, pp. 473–482. URL: http://doi.acm.org/10.1145/2207676.2207741, doi:10.1145/2207676.2207741. 1

[Gle13] GLEICHER M.: Explainers: Expert explorations with crafted projections. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2042–2051. doi:http://doi.ieeecomputersociety.org/10.1109/TVCG.2013.157. 2

[Jol02] JOLLIFFE I.: *Principal Component Analysis*, 3 ed. Springer, 2002. 2

[JPC*11] JOIA P., PAULOVICH F., COIMBRA D., CUMINATO J., NONATO L.: Local affine multidimensional projection. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (Dec 2011), 2563–2571. doi:10.1109/TVCG.2011.220. 3

[Kan12] KANDOGAN E.: Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST)* (Oct 2012), pp. 73–82. doi:10.1109/VAST.2012.6400487. 2

[MCMT14] MARTINS R., COIMBRA D., MINGHIM R., TELEA A.: Visual analysis of dimensionality reduction quality for parameterized projections. *Computers & Graphics 41* (2014), 26–42. 1, 2

[MSM*10] MEIRELLES P., SANTOS C., MIRANDA J., KON F., TERCEIRO A., CHAVEZ C.: A study of the relationships between source code metrics and attractiveness in free software projects. In *Proceedings of the 2010 Brazilian Symposium on Software Engineering (SBES)* (Sept 2010), pp. 11–20. doi:10.1109/SBES.2010.27. 3

[NB12] NOCAJ A., BRANDES U.: Organizing search results with a reference map. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (Dec 2012), 2546–2555. doi:10.1109/TVCG.2012.250. 2

[oM14] OF MARYLAND U.: US counties dataset, 2014. URL: http://archive.ics.uci.edu/ml. 3

[PPM*15] PAGLIOSA P. A., PAULOVICH F. V., MINGHIM R., LEVKOWITZ H., NONATO L. G.: Projection inspector: Assessment and synthesis of multidimensional projections. *Neurocomputing 150* (2015), 599–610. URL: http://www.sciencedirect.com/science/article/pii/S0925231214012880. 1

[SMT13] SEDLMAIR M., MUNZNER T., TORY M.: Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (Dec 2013), 2634–2643. doi:10.1109/TVCG.2013.153. 1

[SvLB10] SCHRECK T., VON LANDESBERGER T., BREMM S.: Techniques for precision-based visual analysis of projected data. *Information Visualization 9*, 3 (June 2010), 181–193. URL: http://dx.doi.org/10.1057/ivs.2010.2, doi:10.1057/ivs.2010.2. 1

[TE10] TELEA A., ERSOY O.: Image-based edge bundles: Simplified visualization of large graphs. *Computer Graphics Forum 29*, 3 (2010), 543–551. 4

[TSLX12] TAN L., SONG Y., LIU S., XIE L.: Imagehive: Interactive content-aware image summarization. *IEEE Computer Graphics and Applications 32*, 1 (Jan 2012), 46–55. doi:10.1109/MCG.2011.89. 1