On the effectiveness of user manipulation in multidimensional projections

Samuel G. Fadel and Fernando V. Paulovich
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
São Carlos, Brazil
samuel.fadel@usp.br, paulovic@icmc.usp.br

***Abstract*—With the advent of interactive techniques for multidimensional data visualization using dimensionality reduction, new possibilities of dealing with data complexity were introduced. The idea behind those methods is to allow the user to manually modify the mapping of a subset of the data, steering the mapping of the whole data set. Previous studies suggest that user manipulation is beneficial to the visualization as a means of effectively modifying the final result. Still, those studies focus only on the grouping of instances, such as group separation and compactness. This paper proposes a new view on effectiveness of user manipulation based on well-known evaluation measures. In addition, it provides initial experimental evidence on the effectiveness of user manipulation on state of the art interactive methods. Although the manipulation does affect results, we conclude that it does not lead to definitive improvements.**

***Keywords*-interactive techniques; dimensionality reduction; data visualization; multidimensional data;**

I. Introduction

Nonlinear dimensionality reduction methods are both commonly and traditionally used to visualize multidimensional data [1]. Their purpose is to find low-dimensional representations of high-dimensional data while preserving some aspect of the original structure. In general, this structure is related to pairwise dissimilarities between data items or neighborhood structure. A particular class of such methods, which allow user interactivity, provide new ways to handle complex data by allowing the user to interactively steer the mapping, as shown by recent research [2], [3], [4], [5]. Using a previously mapped subset of the data as input, users can manipulate the subset mapping to indirectly influence (either positively or negatively) the visualization.

Previous studies, however, have focused only on improvements related to cluster separation and compactness. The ability to improve mappings on other aspects, such as preservation of neighborhood structures, were not taken into account. While it is clear that the initial mapping plays an important role on the geometry of the mapping, those studies lack either theoretical or experimental evidence to properly assess user manipulation as an effective tool for incorporating user knowledge into the visualization.

In this paper we review three recent interactive methods (Section II), namely Least-Square Projection (LSP) [6], Part-Linear Multidimensional Projection (PLMP) [2] and Local Affine Multidimensional Projection (LAMP) [3], highlighting their advantages and disadvantages relative to one another. We focus on these methods primarily due to their accuracy and efficiency. Following that, we outline the concept of effective manipulation and establish the quality measures adopted in our study (Section III). We then perform a series of experiments in order to assess whether manipulation is effective on them (detailed in Section IV) and discuss the results attained (Section V).

Contributions: The contributions of this paper are (i) a simple framework for evaluating the effectiveness of user manipulation in interactive methods; (ii) an initial assessment of state of the art interactive methods using this framework.

II. Technical Background

Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ be a set of p-tuples $\mathbf{x}_i \in \mathbb{R}^p$ representing n observations of $p > 3$ variables. We also consider another representation, which corresponds to placing the observations in a matrix $X_{n \times p}$ where each observed variable has a corresponding column and each observation occupies a row. We call $\mathcal{X}$ (or X) a *data set*, each tuple $\mathbf{x}_i$ (or row of X) a *datum*, *instance* or *sample*, each variable (or column of X) an *attribute*, *feature* or *dimension*.

As previously mentioned, the methods discussed are approaches at solving the problem of finding a q-dimensional ($q << p$) representation $\mathcal{Y}$ of $\mathcal{X}$ such that some aspect of the original structure of $\mathcal{X}$ is still observable in $\mathcal{Y}$. In addition to $\mathcal{X}$, another set of tuples $\mathcal{X}' \subset \mathcal{X}$, called *control points*, and its q-dimensional mapping $\mathcal{Y}'$ is also given as input, with $|\mathcal{X}'| = |\mathcal{Y}'| = c$ ($c << n$).

A. Least-Square Projection (LSP)

Let N_i ($|N_i| = k$) be the set of the k-nearest neighbors of $\mathbf{x}_i$. Given each set N_i, $1 \leq i \leq n$, LSP uses a Laplacian operator to preserve this neighborhood structure by mapping each $\mathbf{x}_i$ inside the convex hull of the images of all $\mathbf{x}_j \in N_i$. More precisely,

$$\mathbf{y}_i - \sum_{\mathbf{x}_j \in N_i} \alpha_{ij} \mathbf{y}_j = 0, \tag{1}$$

with $\alpha_{ij} \geq 0$ and $\sum_j \alpha_{ij} = 1$. Solving Equation (1) for all i, however, will generally not produce good results, as a trivial solution ($\mathbf{y}_i = 0, \forall\, i$) is always possible. Because of this, we add restrictions to the solution by imposing that every $\mathbf{x}_i \in \mathcal{X}'$ must be mapped to its image given in $\mathcal{Y}'$.

LSP is a very precise method for preserving neighborhood structures, but since the linear system grows quadratically on n, its computational cost is high, compared to both PLMP and LAMP. As a result, its applicability as an interactive technique is more limited than the others.

B. Part-Linear Multidimensional Projection (PLMP)

Much faster and simpler than LSP, PLMP aims to find a linear map $\tilde{\Phi}_{p\times q}$ that approximates the function Φ which transforms X' into Y', that is, $\Phi(X') = Y'$. This is done by solving the normal equation

$$X'^T X' \phi_j = X'^T \mathbf{b}_j \tag{2}$$

for all $1 \leq j \leq q$, where $\mathbf{b}_j$ is the j-th column of Y'. Each ϕ_j is then used as a column of $\tilde{\Phi}$. Then, $\tilde{\Phi}$ is used to project each $\mathbf{x}_i \notin \mathcal{X}$ in order to approximate the mapping provided. Since, for visualization purposes, q is usually 2 or 3, the result is a very fast technique that can map very large data sets. This, however, comes at the cost of being less precise than both LSP and LAMP in terms of dissimilarity and neighborhood structure preservation.

C. Local Affine Multidimensional Projection (LAMP)

In contrast to other techniques, LAMP determines an individual orthogonal mapping for each $\mathbf{x}_i$. The problem is defined in terms of the already known coordinates $\mathcal{Y}'$. The mapping of each $\mathbf{x}_i$ is the affine transformation $f_{\mathbf{x}_i}(\mathbf{x}) = \mathbf{x}M + \mathbf{t}$ that optimizes

$$\begin{aligned} \min_{f_{\mathbf{x}_i}} \quad & \sum_{j=1}^{c} \alpha_j ||f_{\mathbf{x}_i}(\mathbf{x}'_j) - \mathbf{y}'_j||^2 \\ \text{subject to} \quad & M^T M = I, \end{aligned} \tag{3}$$

where $\mathbf{x}'_j \in \mathcal{X}'$, $\mathbf{y}'_j \in \mathcal{Y}'$ and $\alpha_j = \frac{1}{||\mathbf{x}'_j - \mathbf{x}_i||^2}$ are scalar weights added in order to let more similar control points influence more the mapping of $\mathbf{x}_i$, while, conversely, less similar ones have less influence.

The solution to the minimization problem (3) is to rewrite it in matrix form, giving rise to an Orthogonal Procrustes Problem [7], which has known solution. To do this, we take the partial derivatives with respect to $\mathbf{t}$ equal to zero and rewrite $\mathbf{t}$ in terms of M as $\mathbf{t} = \bar{\mathbf{y}}' - \bar{\mathbf{x}}'M$, where $\bar{\mathbf{x}}' = \frac{\sum_{j=1}^{c} \alpha_j \mathbf{x}'_j}{\sum_{j=1}^{c} \alpha_j}$ and $\bar{\mathbf{y}}' = \frac{\sum_{j=1}^{c} \alpha_j \mathbf{y}'_j}{\sum_{j=1}^{c} \alpha_j}$.

Back to problem (3), we have

$$\begin{aligned} \min_{f_{\mathbf{x}_i}} \quad & \sum_{j=1}^{c} \alpha_j ||f_{\mathbf{x}_i}(\hat{\mathbf{x}}_j) - \hat{\mathbf{y}}_j||^2 \\ \text{subject to} \quad & M^T M = I, \end{aligned} \tag{4}$$

where $\hat{\mathbf{x}}_j = \mathbf{x}'_j - \bar{\mathbf{x}}'$ and $\hat{\mathbf{y}}_j = \mathbf{y}'_j - \bar{\mathbf{y}}'$. Lastly, by writing problem (4) in matrix form, we have

$$\begin{aligned} \min \quad & ||AM - B||_F \\ \text{subject to} \quad & M^T M = I, \end{aligned} \tag{5}$$

where $||\cdot||_F$ denotes the Frobenius norm and matrices A and B are given by

$$A = \begin{bmatrix} \sqrt{\alpha_1}\hat{\mathbf{x}}_1^T \\ \sqrt{\alpha_2}\hat{\mathbf{x}}_2^T \\ \vdots \\ \sqrt{\alpha_c}\hat{\mathbf{x}}_c^T \end{bmatrix}, \quad B = \begin{bmatrix} \sqrt{\alpha_1}\hat{\mathbf{y}}_1^T \\ \sqrt{\alpha_2}\hat{\mathbf{y}}_2^T \\ \vdots \\ \sqrt{\alpha_c}\hat{\mathbf{y}}_c^T \end{bmatrix}. \tag{6}$$

As mentioned before, problem (5) has well-known solution. By writing $A^T B = UDV^T$ where UDV^T is the singular value decomposition (SVD) of $A^T B$, we have that $M = UV^T$. Therefore, $f_{\mathbf{x}_i}$, which gives us $\mathbf{y}_i$, is given by

$$f_{\mathbf{x}_i}(\mathbf{x}) = (\mathbf{x} - \bar{\mathbf{x}}')M + \bar{\mathbf{y}}'. \tag{7}$$

Since we can limit the number of control points that influence each instance via the a_{ij} coefficients, LAMP is capable of being a truly local technique. In addition, it is a very efficient technique and, while still slower than PLMP, has very competitive accuracy.

III. Evaluating user manipulation

We propose an evaluation framework based on well-known evaluation measures for dimensionality reduction. Since we are interested in evaluating the effectiveness of user manipulation on LSP, PLMP and LAMP, we need to measure how transformations on $\mathcal{Y}'$ propagate to $\mathcal{Y}$. Specifically, suppose a transformation is applied to $\mathcal{Y}'$, producing $\tilde{\mathcal{Y}}'$. Then, one of the methods discussed in Section II is applied to $\mathcal{Y}'$ and $\tilde{\mathcal{Y}}'$, producing $\mathcal{Y}$ and $\tilde{\mathcal{Y}}$, respectively. Given an evaluation measure m, we say that the user manipulation is effective if

$$m(\mathcal{Y}') \leq m(\tilde{\mathcal{Y}}') \Rightarrow m(\mathcal{Y}) \leq m(\tilde{\mathcal{Y}})$$

or

$$m(\mathcal{Y}') \geq m(\tilde{\mathcal{Y}}') \Rightarrow m(\mathcal{Y}) \geq m(\tilde{\mathcal{Y}}).$$

In other words, if the user manipulation increases (or decreases) m on the mapped subset, we expect that the final map will also increase (or decrease) m.

A. Evaluation measures

Three different aspects of the mappings are evaluated by the measures. Stress [8] is a measure of how distorted is the representation of the dissimilarities δ_{ij} (the dissimilarity between $\mathbf{x}_i$ and $\mathbf{x}_j$) in $\mathcal{Y}$. This distortion is given by

$$\sigma = \frac{1}{\sum_{i<j} \delta_{ij}} \sum_{i<j} \frac{(\delta_{ij} - d_{ij})^2}{\delta_{ij}}, \tag{8}$$

where d_{ij} represents the distance between $\mathbf{y}_i$ and $\mathbf{y}_j$.

We also use the silhouette coefficient [9], which measures how well grouped are the instances in $\mathcal{Y}$, based on their class. In this metric, being well grouped means that an instance is far from instances of other classes and that the average distance between instances of the same class is low. Let l_i be the class of $\mathbf{y}_i$, $a(i)$ the average distance between $\mathbf{y}_i$ and every other instance with class equal to l_i and $b(i)$ the smallest distance

TABLE I
DATA SETS USED IN THE EXPERIMENTS.

Data set	n	p	Source
WDBC	569	30	[11]
WINE	178	13	[11]

between $\mathbf{y}_i$ and every other instance with class different from l_i. The silhouette coefficient $s(i)$ of $\mathbf{y}_i$ is

$$s(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}}. \tag{9}$$

Lastly, we evaluate the representation of neighborhood structures using the neighborhood preservation index [10]. Let $k\mathrm{NN}(\mathbf{x}_i)$ and $k\mathrm{NN}(\mathbf{y}_i)$ be the sets containing indices of the k nearest neighbors of instance i in $\mathcal{X}$ and in $\mathcal{Y}$, respectively. The neighborhood preservation index is given by

$$\mathrm{NP}_k(i) = \frac{|k\mathrm{NN}(\mathbf{y}_i) \cap k\mathrm{NN}(\mathbf{x}_i)|}{|k\mathrm{NN}(\mathbf{x}_i)|}. \tag{10}$$

Since both the silhouette coefficient and neighborhood preservation index are calculated for each individual instance and not the whole map, we average them over all instances in order to globally evaluate a map.

IV. EXPERIMENTS

Using the concept of effectiveness and the evaluation measures described in Section III, we perform a series of experiments in order to assess how effective is the manipulation on each technique. For each data set, we apply automated transformations on $\mathcal{Y}'$ to simulate an improvement of each measure. The data sets we performed our experiments on are detailed in Table I.

The first step is to choose $\mathcal{X}'$, which will be used to find the initial mapping. We use $c = 3\sqrt{n}$ and, since we are not interesting in evaluating how representative are the control points, we randomly pick instances to compose $\mathcal{X}'$. Because of this, all experiments are repeated 30 times. To determine $\mathcal{Y}'$, we use the classical scaling method [12] on $\mathcal{X}'$, as this is a linear method. Consequently, we can more easily improve its mapping according to each measure using nonlinear strategies. Next, we describe the strategies adopted to artificially produce $\tilde{\mathcal{Y}}'$.

To observe improvements in stress, we use Sammon's nonlinear mapping [8], which is a direct optimization of Equation (8). The transformation to improve the silhouette coefficient is the one described in [13], which adds more attributes to instances based on their class. This transformation is known to improve class separation, directly improving the silhouette coefficient. Finally, to observe improvements in neighborhood preservation, we use the t-SNE technique [14], which is known to produce maps with high neighborhood preservation [15].

The techniques were tuned as follows. LAMP was set to use 25% nearest control points with the default values of $alpha_j$, while the other 75% control points use $alpha_j = 0$. The neighborhood map used by LSP was built using 15 nearest neighbors.

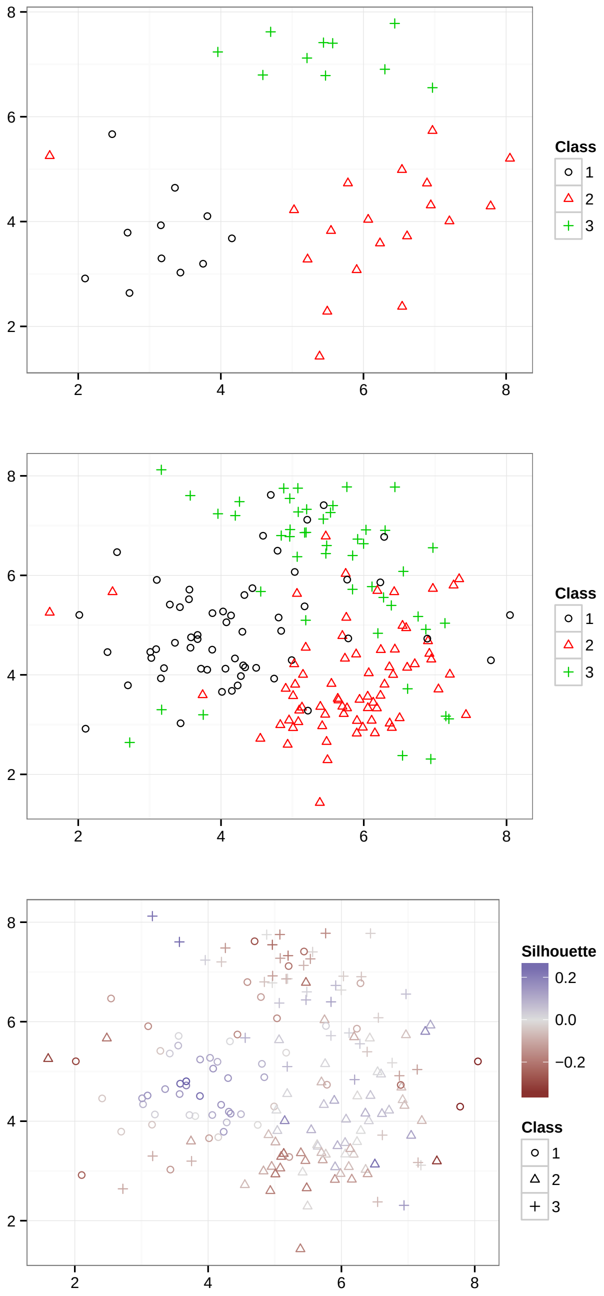

Fig. 1. The manipulation process for the WINE data set in LAMP. From top to bottom, the first plot illustrates $\tilde{\mathcal{Y}}'$ and the second plot illustrates $\tilde{\mathcal{Y}}$. In the last plot, instances are colored according to the observed difference in the silhouette coefficient before and after the manipulation, with zero as gray, above zero (increase) as blue and below zero (decrease) as red.

V. RESULTS AND DISCUSSION

In Fig. 1 we illustrate the results of the manipulation for the WINE data set, along with a plot illustrating the differences in the silhouette coefficient for each instance. The complete results for all techniques and data sets for each measure are shown in Fig. 3 (silhouette), Fig. 4 (neighborhood preservation) and Fig. 2 (stress).

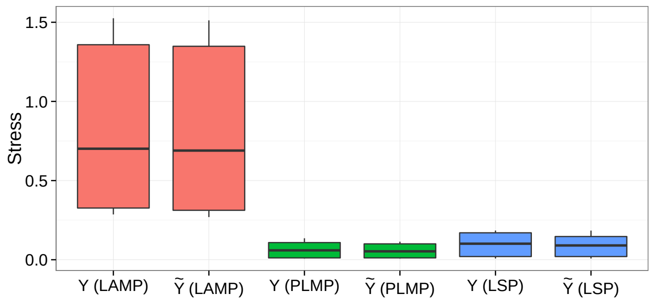

Fig. 2. Results for stress (averaged over all data sets).

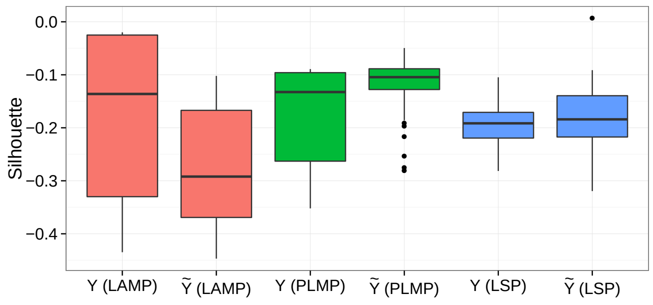

Fig. 3. Results for the silhouette coefficient (averaged over all data sets).

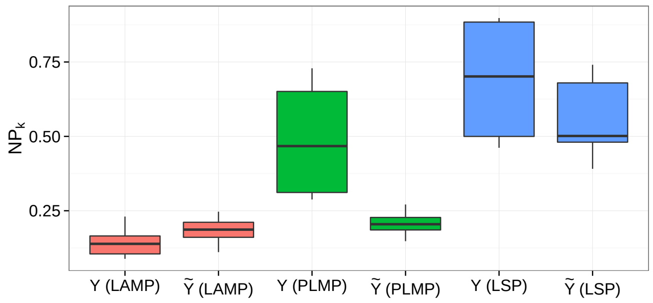

Fig. 4. Results for neighborhood preservation ($k = 15$; averaged over all data sets).

The results for stress (Fig. 2) provide the first significant evidence for our study. At first glance, we see that all three techniques are not affected by the manipulation. Further inspection reveals that it is in fact lower, altough the reduction is very small. While stress is directly optimized in the manipulation, the improvement itself is not very large. Since the initial solution is found by classical scaling, this suggests that finding an optimal map in terms of stress for a subset of the data in this case is not necessary, since it will not be significantly improved.

In addition, LAMP shows very high values of stress when compared to the other techniques. This is expected since, as previously described, we employ only 25% of the control points when mapping each instance. As a consequence, the technique becomes highly local, introducing discontinuities in the map [3]. This results in larger distortions of dissimilarities between instances that have few or no common control points when we select the 25% nearest.

Conversely to stress, the silhouette coefficient (Fig. 3) showed larger variation before and after manipulation. Only PLMP showed improvements in the final map but, although much larger than stress, still not significant enough. LAMP, however, suffered a descrease in silhouette, which was not an expected outcome. This could happen for two reasons. First, both data sets had very different average silhouette values, as highlighted by the many outliers in PLMP. Second, since very few data sets were evaluated, a limitation of this paper, if the method used to improve the silhouette was not enough to significantly improve the silhouette in one of them, this will become evident in the boxplots.

Neighborhood preservation (Fig. 4) is the only measure to provide significant observable changes. Initially, we can see that LSP is the least affected technique. This is probably due to its robustness in faithfully mapping the neighborhood graph used to represent the original data. Since we evaluate the same number of neighbors as the technique employs, we can see that manipulation is not significant enough to distort the maps attained. This was not the case for LAMP and PLMP, however.

In the case of LAMP, the very low values of neighborhood preservation attained can be attributed to the very small neighborhoods created by the 25% nearest control points. Since we evaluate 15 neighbors for all maps, a value independent of the number of control points used to map each instance in LAMP, the distortions created by the previously mentioned discontinuities are probably responsible for this. One insteresting result, however, is that it its the only technique on which we can observe consistent improvement in neighborhood preservation. This can be a consequnce of the fact that the technique already attained very low values in the first place, leaving more room for improvement. Contrarily, PLMP suffered heavily from the manipulation done by t-SNE. Since it is the only truly global technique, the local distortions introduced by t-SNE which improve neighborhood structure representation probably led to increasingly distorted dissimilarities on the rest of the instances.

A. *Limitations*

The main limitation of this study is the amount of data sets used in the evaluation. As an initial assessment, it provides some but limited evidence. Additionally, studies were conducted globally on the mappings. Consequently, possible local modifications in badly mapped regions which happen to improve those regions are not taken into account.

VI. Conclusions

In this paper, we reviewed three state of the art techniques for interactive multidimensional data visualization using dimensionality reduction. We proposed a concept of interaction effectiveness and assessed the reviewed techniques under this concept using well-known evaluation measures. The proposed assessment framework serves as an initial formulation for more elaborate studies.

In general, the experiments conducted suggest that the manipulation did affect the mappings regarding neighborhood preservation and silhouette, either for better or worse, but stress remained largely unaffected. A direct implication is that while some distortions in dissimilarities are alleviated by the manipulation, others are introduced to compensate. Moreover, the inconsistency in either improving or degrading the mappings suggest that user manipulation might not an effective action to improve them globally.

Further studies should be done on larger and more complex data sets in order to attain more significant results. Additionally, local modifications such as improvement of badly mapped regions and strategies to improve those using specially tailored manipulations would provide an interesting insight into possible improvements of currently available interactive methods.

Acknowledgments

The authors are supported by CNPq and FAPESP (grant numbers #2014/13010-7 and #2011/22749-8).

References

[1] J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski, "Information retrieval perspective to nonlinear dimensionality reduction for data visualization," vol. Volume 11, pp. 451–490.

[2] F. V. Paulovich, C. Silva, and L. Nonato, "Two-phase mapping for projecting massive data sets," vol. 16, no. 6, pp. 1281–1290.

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

[4] F. V. Paulovich, C. Silva, and L. Nonato, "User-centered multidimensional projection techniques," vol. 14, no. 4, pp. 74–81.

[5] G. M. H. Mamani, F. M. Fatore, L. G. Nonato, and F. V. Paulovich, "User-driven feature space transformation," vol. 32, no. 3, pp. 291–299.

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

[7] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*. Oxford Statistical Science Series 30.

[8] J. W. Sammon, "A nonlinear mapping for data structure analysis," vol. 18, no. 5, pp. 401–409.

[9] P. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, no. 1, pp. 53–65, Nov. 1987.

[10] F. V. Paulovich and R. Minghim, "Hipp: A novel hierarchical point placement strategy and its application to the exploration of document collections," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 14, no. 6, pp. 1229–1236, 2008.

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

[12] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*, 2nd ed. CRC Press, Sep. 2000.

[13] M. Schaefer, L. Zhang, T. Schreck, A. Tatu, J. A. Lee, M. Verleysen, and D. A. Keim, "Improving projection-based data analysis by feature space transformations," vol. 8654.

[14] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," vol. 9, pp. 2579–2605.

[15] S. G. Fadel, F. M. Fatore, F. S. L. G. Duarte, and F. V. Paulovich, "LoCH: A neighborhood-based multidimensional projection technique for high-dimensional sparse spaces," *Neurocomputing*, vol. 150, Part B, pp. 546–556, Feb. 2015.