Globally local and fast explanations of *t*-SNE-like nonlinear embeddings

Pierre Lambert[1,*], Rebecca Marion[2], Julien Albert[2], Emmanuel Jean[3], Sacha Corbugy[2] and Cyril de Bodt[1,4]

[1]UCLouvain - ICTEAM & TRAIL, Louvain-la-Neuve, Belgium
[2]UNamur - NaDI/PReCISE & TRAIL, Namur, Belgium
[3]Multitel & TRAIL, Mons, Belgium
[4]MIT Media Lab, Cambridge [MA], USA

Abstract

Nonlinear dimensionality reduction (NLDR) algorithms such as *t*-SNE are often employed to visually analyze high-dimensional (HD) data sets in the form of low-dimensional (LD) embeddings. Unfortunately, the nonlinearity of the NLDR process prohibits the interpretation of the resulting embeddings in terms of the HD features. State-of-the-art studies propose post-hoc explanation approaches to locally explain the embeddings. However, such tools are typically slow and do not automatically cover the entire LD embedding, instead providing local explanations around one selected data point at a time. This prevents users from quickly gaining insights about the general explainability landscape of the embedding. This paper presents a **globally local** and **fast** explanation framework for NLDR embeddings. This framework is **fast** because it only requires the computation of sparse linear regression models on subsets of the data, without ever reapplying the NLDR algorithm itself. In addition, the framework is **globally local** in the sense that the entire LD embedding is automatically covered by multiple local explanations. The different interpretable structures in the embedding are directly characterized, making it possible to quantify the importance of the HD features in various regions of the LD embedding. An example use-case is examined, emphasizing the value of the presented framework. Public codes and a software are available at https://github.com/PierreLambert3/glocally_explained.

Keywords

dimensionality reduction, data visualization, interactivity, interpretability, explainability, *t*-SNE, data exploration

1. Introduction

Dimensionality reduction (DR) computes low-dimensional (LD) representations of high-dimensional (HD) data, e.g., to visually explore them or to curb the curse of dimensionality [1]. The relevance of a DR method for a given visualization task typically depends on its preservation of the HD neighborhoods in the resulting LD embedding [2]. Two major frameworks have been proposed for projecting from HD to LD coordinates [1]: one is based on preserving distances [3], while the other is based on reproducing neighborhoods [4, 5]. For instance, distance-preserving methods like principal component analysis (PCA) [6] and classical metric multidimensional scaling (MDS) [3] project HD samples linearly; nonlinear variants of these methods (e.g., [7, 8]) aim to preserve weighted Euclidean or approximately geodesic distances. Numerous other schemes have also been developed that determine the LD embedding design based on HD affinity matrices [9, 10]. Regrettably, the local neighborhood preservation of all of these techniques is limited in visualization contexts by the norm concentration phenomenon [11, 12], most probably due to their distance-preserving nature [1, 13]. In contrast, the native shift invariance of neighbor embedding (NE) algorithms [14] such as Stochastic Neighbor Embedding (SNE) [5] mitigates this phenomenon, leading to astonishing DR quality. These achievements have naturally encouraged the development of numerous SNE-based methods, such as the popular *t*-SNE [15], UMAP [16], multi-scale perplexity-free approaches [17, 18, 19], etc.

While these nonlinear DR (NLDR) algorithms deliver impressively faithful LD embeddings with respect to the HD data, their intrinsic nonlinearity greatly affects the interpretability of the LD representations. Indeed, the obtained LD dimensions are hardly or most often not interpretable in terms of the HD features [20]. Since NLDR methods are not interpretable by design, previous studies have developed techniques to analyze and interpret the LD embeddings, which is known as post-hoc explanation or interpretability [21]. One can for instance cite [22], which proposes to explain visual LD clusters thanks to

Advances in Interpretable Machine Learning and Artificial Intelligence, 2022 October 21, Atlanta, Georgia, USA
*Corresponding author.
pierre.h.lambert@uclouvain.be (P. Lambert);
cyril.debodt@uclouvain.be (C. de Bodt)
https://github.com/PierreLambert3 (P. Lambert);
https://github.com/cdebodt (C. de Bodt)
0000-0003-2347-1756 (C. de Bodt)

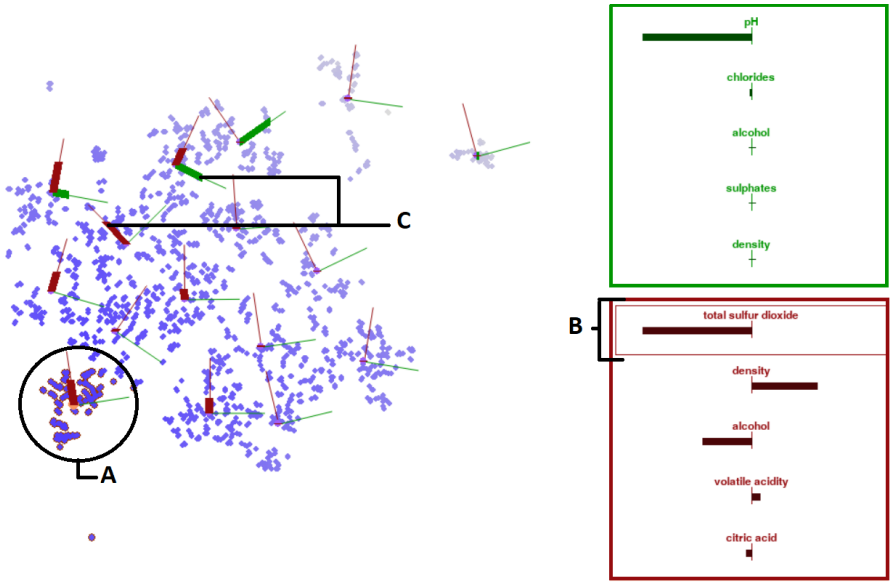

Figure 1: Interface for the proposed globally local and fast explanation framework.

decision trees. On the other hand, [21] locally explain *t*-SNE embeddings by adapting LIME; the authors argue that explaining the entire embedding at once would be difficult, as *t*-SNE usually does not preserve large HD distances well [20]. However, the local nature of *t*-SNE motivates the computation of local explanations in the LD embedding; LIME can then be revisited and performed locally around a user-selected data point. Nevertheless, such an approach has two main limitations: (1) it is slow, and (2) it does not cover the entire LD embedding automatically, as local explanations are only provided around data points that have been selected, one at a time. This approach is slow because, in order to explain a given data point's position in the embedding, *t*-SNE must be reapplied to many artificially simulated points around that data point; the non-parametric nature of *t*-SNE, combined with its significant computational cost, greatly increases computation time, which decreases the potential for interactivity. The second limitation of the method is that the user only receives a local explanation around the selected point in the embedding; she must thus explore the various regions of the embedding manually. This is not realistic in practice, especially when working with large databases, and even more so since the approach is not fast.

This paper aims to address these limitations by developing a **fast** and **globally local** explanation framework for NLDR embeddings. Based on the BIOT explanation approach [23], this framework learns sparse linear regression models for subsets of the data set and does not require a reapplication of the NLDR algorithm, making it **fast**. The **globally local** nature of our approach refers to the fact that multiple local explanations are automatically computed over the entire LD embedding (i.e., globally). Such an automatic processing enables the user to directly glimpse the overall explainability landscape of the embedding, as well as a structured overview of the impact of the HD features in the various parts of the LD embedding. The regions for which local explanations are learned in the LD embedding can be determined in different ways [24]: using a clustering algorithm such as K-means, as in this work, thanks to a manual selection performed by the user, or by recursively splitting the embedding into subcells along the LD dimensions based on a model error criterion.

Our fast and globally local explanation framework can be viewed as taking the best of both *linear* and *nonlinear* projection worlds: the LD embedding can indeed be generated by a *nonlinear* DR algorithm, achieving much better DR quality in terms of data visualization thanks to increased flexibility and adaptability [12, 15, 2]. On the other hand, the computed local explanations are *linear* and sparse, which promotes interpretability. Moreover, the globally local explanations make it possible to readily depict the importance of the HD features in the different regions of the LD embedding. As an experiment, an example use-case on a public data set is presented, highlighting the usefulness of the proposed approach. Free code and software are publicly available online (https://github.com/PierreLambert3/glocally_explained), enabling the easy use of the proposed framework.

This paper is organized as follows: Section 2 first re-

views some related works. Section 3 then presents our proposed approach, while Section 4 discusses an example use-case. Section 5 draws final conclusions.

2. Related works

Interpreting NLDR techniques is a challenging task. To tackle this challenge, various approaches have been proposed. Some papers (e.g., [25, 26, 27]) have proposed methods for explaining the LD embedding dimensions with respect to the HD features. Since local NLDR algorithms such as *t*-SNE do not effectively preserve large distances, explaining the resulting embedding dimensions with these methods may be misleading. Other methods attempt to interpret NLDR results by explaining visual clusters [22, 28, 29]. For example, in [22], the authors propose an interactive pipeline for explaining clusters in the LD embedding using decision trees; this pipeline enables the user to manually select LD clusters, which are then explained in terms of the HD features with a decision tree, an interpretable model. The resulting model can be used to explain why certain data points are clustered together and to identify the HD features that distinguish the different clusters. In contrast, our proposed approach aims to understand intra-cluster positions, i.e., the HD features that make two points from the same cluster lie at different corners of this cluster. Moreover, our framework makes it possible to not only explain LD clusters, but more generally interpret the overall positions of the points in the embedding.

Other existing methods aim to locally and linearly explain the position of a specific instance in the LD space. In particular, [21] adapts LIME [30] to locally explain *t*-SNE embeddings. The original version of LIME involves three steps. First, it samples instances around a point of interest. Then, it queries the model for these instances. Finally, it fits an interpretable model with the result of the queries. In [21], the authors use a SMOTE oversampling technique [31] to create new artificial neighbors for the point of interest. To query *t*-SNE, the entire DR process is re-applied for each sampled instance, since the *t*-SNE mapping function is unknown. Finally, BIR [32] —which is the predecessor of BIOT [23], a method employed in our work —is used to produce local explanations; BIR finds the rotation of the queried sampled data that results in the best explanation model (in terms of model sparsity and error). While the approach presented in [21] provides nice intuitions about the LD embedding structure, it has several limitations. First, it can only compute one local explanation at a time, for one selected point. Second, the obtained explanation is highly dependent on the artificial sampling. Finally, running the entire NLDR process for all sampled instances is (very) time consuming, and thus prohibits interactivity. The approach presented in this paper addresses the limitations of [21] by (1) directly providing local explanations everywhere in the LD embedding (i.e., globally local explanations), (2) avoiding the need to sample new artificial data points, and (3) relying only on the calculation of linear regression models, which ensures fast processing and hence facilitates interactivity.

3. Proposed approach

This section introduces our proposed approach for globally local and fast explanations of NLDR embeddings. Section 3.1 first summarizes our notations. Section 3.2 then details our methodology, and Section 3.3 finally presents an optional fine-tuning strategy.

3.1. Notations

Matrices are denoted with bold-faced capital letters (e.g., $\vec{X}$), vectors with bold-faced lower-case letters (e.g., $\vec{x}$) and scalars with lower-case letters (e.g., x). A single element from a matrix is denoted with a lower-case letter with two subscripts (e.g., x_{ij}), the first indicating the row and the second indicating the column. Instances are indexed by the letter $i \in \{1, ..., n\}$, features by the letter $j \in \{1, ..., d\}$, embedding dimensions by the letter $k \in \{1, ..., m\}$ and regions or subcells of the embedding by the letter $\ell \in \{1, ..., L\}$.

3.2. General methodology

In [23], the Best Interpretable Orthogonal Transformation (BIOT) method was proposed to explain the dimensions of multidimensional scaling (MDS) embeddings. In the case of *t*-SNE, such an explanation strategy is not directly applicable because *t*-SNE only preserves local structure from the high-dimensional data. However, as proposed in [21], *t*-SNE embeddings may be explained locally. Instead of learning a BIOT explanation model for the entire embedding (i.e., a single global explanation), we propose learning different BIOT models for different regions (or subcells) of the embedding (i.e. local explanation). For a given region, the BIOT model identifies the features that best explain the positioning of points within that region of the embedding, independently of all other regions. This approach can be applied to any non-linear 2-D embedding, including embeddings generated by *t*-SNE and its extensions (e.g., [33, 19]) or by other NLDR algorithms (e.g., [16, 17, 9, 34]).

Let $\vec{X}(n \times d)$ be the matrix of d features used to generate the embedding $\vec{Y}(n \times 2)$. Furthermore, let $\vec{W}(d \times 2)$ and $\vec{w}_0$ (2×1) contain the weights and intercepts for the linear models relating the features in $\vec{X}$ to each dimension of the embedding $\vec{Y}$, where there is one model per dimension.

Finally, $\vec{R}$ (2x2) is an orthogonal transformation matrix that is applied to $\vec{Y}$ to promote model sparsity and prediction quality, and $\lambda > 0$ is a hyperparameter to control model sparsity. For 2-D embeddings, the BIOT objective function for global explanation is

$$J_0(\vec{W}, \vec{w}_0, \vec{R}) = \frac{1}{2n} \sum_{i=1}^{n} \sum_{k=1}^{2} (\vec{y}_i^\top \vec{r}_k - w_{0k} - \vec{x}_i^\top \vec{w}_k)^2 + \lambda \sum_{k=1}^{2} \|\vec{w}_k\|_1, \tag{1}$$

which is minimized w.r.t $\vec{W}$, $\vec{w}_0$ and $\vec{R}$ under the constraint that $\vec{R}$ is an orthogonal matrix ($\vec{R}\vec{R}^\top = \vec{R}^\top \vec{R} = \vec{I}_2$).

Clearly, this objective function can be extended to the case where different model parameters $\vec{W}^{(\ell)}$, $\vec{w}_0^{(\ell)}$ and $\vec{R}^{(\ell)}$ are optimized for different regions ℓ of the embedding, where the set of instances in region ℓ is denoted $\mathcal{S}_\ell$. In practice, the best segmentation of the embedding into regions is unknown. In this paper, we propose segmenting the embedding automatically by performing K-means on the embedding data. The choice of the hyperparameter K depends on the topology apparent in the embedding and of the granularity of details desired by the user. Other strategies are possible, for instance by recursively dividing the LD dimensions along their medians.

3.3. Fine-tuning

In Section 3.2, the proposed strategy for automatic segmentation (K-means) depends on the coordinates of the instances in the embedding. However, the shape and size of the zone that can be explained may not directly depend on the spatial coordinates of the embedding. This means that the regions identified using K-means may not be the most optimal with respect to the quality of the resulting explanations. In some cases, it is hence useful to fine-tune the final regions by directly considering explanation quality. To do so, we propose a method called Clustered BIOT, which reassigns instances i to explanation regions $\mathcal{S}_\ell$ based on a modification of BIOT. Further details on Clustered BIOT can be found in Appendix A.

4. Experiments and discussion

This section presents an example use-case for the proposed method using an interactive user interface. This user interface is available on the public repository indicated in the abstract. All of the featured embeddings are representations of the *winequality-red* dataset, available in the UCI machine learning repository [35]. This data set contains 11 physico-chemical variables describing various red wines. The embeddings are produced by a recent NE algorithm that mixes t-SNE gradients with those of a fast stochastic approximation of MDS, which preserves HD data structures across multiple scales [34].

The interface displayed in Fig. 1 shows an embedding with multiple local linear explanations: each explanation is composed of a green and a burgundy axis. Explanation Ⓐ has been selected by the user; the color transparency of the points increases linearly with the absolute difference between their position in the embedding and the position predicted by the selected linear model (i.e., the greater the error, the more transparent). This enables the user to visualize the portion of the embedding for which the selected linear model is faithful. The right panel depicts the relative importance of the HD features for each axis of the selected explanation (i.e., Ⓐ in this case), as quantified by the local linear model weights; the horizontal bar under each feature name represents the feature's signed linear projection weight (LPW) on the considered axis, highlighting the importance of the feature in the local explanation. For visual clarity, only the 5 features with the greatest LPW magnitudes are depicted for each local explanation axis. The feature *total sulfur dioxide* has been selected by the user (mark Ⓑ). When selecting a feature in the right panel, thick indicators appear on both axes of all local explanations, with lengths proportional to the LPW magnitudes of the corresponding feature on all axes; mark Ⓒ shows two such indicators. This makes it possible to grasp the influence of an HD feature in the various regions of the entire embedding.

Each view in Fig. 2 shows the importance of a particular feature in the embedding, with the respective feature indicated at the bottom of the panel. The left view highlights that free sulfur dioxide is particularly important when explaining the top portion of the embedding along a vertical direction, whereas the horizontal direction can be partly explained by the concentration of citric acid. We observe that the structures apparent in the bottom-left part of the embedding are not very dependent on the three analyzed features.

5. Conclusion

This work proposes a globally local and fast explanation framework that provides multiple local linear explanations for 2-D data embeddings, enabling the user to assess, at a glance, the importance of different HD features, both locally and across the whole LD embedding. An example use-case demonstrates that the method can effectively reveal zones in the embedding where points are organized according to specific HD features. Finally, some accompanying software is provided (https://github.com/PierreLambert3/glocally_explained), targeting both DR researchers and experts seeking to analyse their data with nonlinear dimensionality reduction visualization tools.

Further works will include testing our framework with actual end-users in the context of a real use case; their

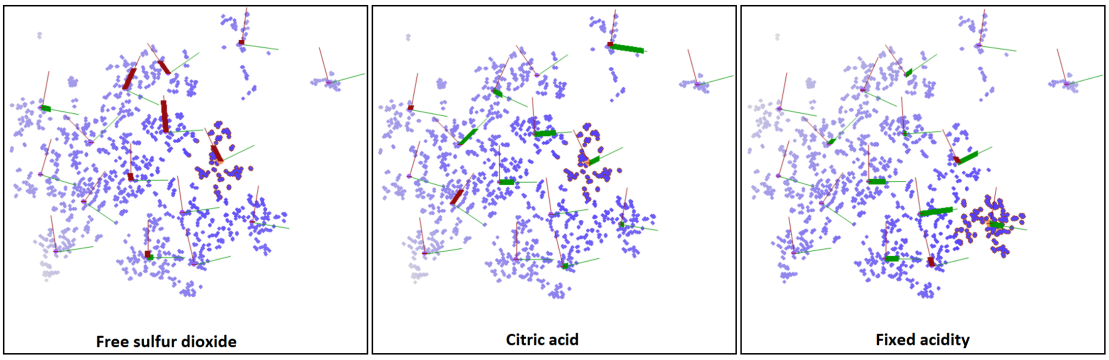

Figure 2: Importance of 3 features in the local explanations of an embedding.

feedback will enable the improvement of the various design choices of our interface. In addition, a qualitative comparison with other explainability methods such as LIME will enable a more comprehensive evaluation of the proposed method.

Acknowledgments

This work was supported by Service Public de Wallonie Recherche under grant n° 2010235-ARIAC by DIGITALWALLONIA4.AI. SC is supported by a FRIA grant (F.R.S.-FNRS).

References

[1] J. A. Lee, M. Verleysen, Nonlinear dimensionality reduction, Springer Science & Business Media, 2007.

[2] J. Venna, J. Peltonen, K. Nybo, H. Aidos, S. Kaski, Information retrieval perspective to nonlinear dimensionality reduction for data visualization., Journal of Machine Learning Research 11 (2010).

[3] I. Borg, P. J. F. Groenen, Modern Multidimensional Scaling: Theory and applications, Springer Science & Business Media, 2005.

[4] T. Kohonen, The self-organizing map, Proceedings of the IEEE 78 (1990) 1464–1480. DOI: 10.1109/5.58325.

[5] G. Hinton, S. Roweis, Stochastic neighbor embedding, in: NIPS, volume 15, 2002, pp. 833–840.

[6] I. T. Jolliffe, Principal component analysis and factor analysis, in: Principal component analysis, Springer, 1986, pp. 115–128.

[7] J. W. Sammon, A nonlinear mapping for data structure analysis 100 (1969) 401–409.

[8] J. B. Tenenbaum, V. De Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323. DOI: 10.1126/science.290.5500.2319.

[9] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, science 290 (2000) 2323–2326.

[10] J. Suykens, Data visualization and dimensionality reduction using kernel maps with a reference point, IEEE Trans. Neural Netw. 19 (2008) 1501–1517.

[11] D. Francois, V. Wertz, M. Verleysen, The concentration of fractional distances 19 (2007) 873–886.

[12] J. A. Lee, M. Verleysen, Quality assessment of dimensionality reduction: Rank-based criteria, Neurocomputing 72 (2009) 1431–1443.

[13] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural computation 10 (1998) 1299–1319.

[14] J. A. Lee, M. Verleysen, Shift-invariant similarities circumvent distance concentration in stochastic neighbor embedding and variants, Procedia Computer Science 4 (2011) 538–547.

[15] L. Van der Maaten, G. Hinton, Visualizing data using t-sne., Journal of machine learning research 9 (2008).

[16] L. McInnes, J. Healy, J. Melville, Umap: Uniform manifold approximation and projection for dimension reduction, arXiv preprint arXiv:1802.03426 (2018).

[17] J. A. Lee, D. H. Peluffo-Ordóñez, M. Verleysen, Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure, Neurocomputing 169 (2015) 246–261.

[18] C. de Bodt, D. Mulders, M. Verleysen, J. A. Lee, Perplexity-free t-SNE and twice Student tt-SNE, in: ESANN, 2018, pp. 123–128.

[19] C. de Bodt, D. Mulders, M. Verleysen, J. A. Lee, Fast multiscale neighbor embedding, IEEE Transactions on Neural Networks and Learning Systems (2020).

[20] M. Wattenberg, F. Viégas, I. Johnson, How to use t-sne effectively, Distill 1 (2016) e2.
[21] A. Bibal, V. M. Vu, G. Nanfack, B. Frénay, Explaining t-sne embeddings locally by adapting lime., in: ESANN, 2020, pp. 393–398.
[22] A. Bibal, A. Clarinval, B. Dumas, B. Frénay, Ixvc: An interactive pipeline for explaining visual clusters in dimensionality reduction visualizations with decision trees, Array 11 (2021) 100080.
[23] A. Bibal, R. Marion, R. von Sachs, B. Frénay, Biot: Explaining multidimensional nonlinear mds embeddings using the best interpretable orthogonal transformation, Neurocomputing 453 (2021) 109–118.
[24] L. Pagliosa, P. Pagliosa, L. G. Nonato, Understanding attribute variability in multidimensional projections, in: 2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), IEEE, 2016, pp. 297–304.
[25] D. B. Coimbra, R. M. Martins, T. T. Neves, A. C. Telea, F. V. Paulovich, Explaining three-dimensional dimensionality reduction plots, Information Visualization 15 (2016) 154–172.
[26] M. Cavallo, Ç. Demiralp, A visual interaction framework for dimensionality reduction based data exploration, in: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018, pp. 1–13.
[27] X. Yuan, D. Ren, Z. Wang, C. Guo, Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data, IEEE Transactions on Visualization and Computer Graphics 19 (2013) 2625–2633.
[28] T. Fujiwara, O.-H. Kwon, K.-L. Ma, Supporting analysis of dimensionality reduction results with contrastive learning, IEEE transactions on visualization and computer graphics 26 (2019) 45–55.
[29] W. E. Marcilio-Jr, D. M. Eler, Explaining dimensionality reduction results using shapley values, Expert Systems with Applications 178 (2021) 115020.
[30] M. T. Ribeiro, S. Singh, C. Guestrin, " why should i trust you?" explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.
[31] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority oversampling technique, Journal of artificial intelligence research 16 (2002) 321–357.
[32] R. Marion, A. Bibal, B. Frénay, Bir: A method for selecting the best interpretable multidimensional scaling rotation using external variables, Neurocomputing 342 (2019) 83–96.
[33] B. Kang, D. García García, J. Lijffijt, R. Santos-Rodríguez, T. De Bie, Conditional t-sne: more informative t-sne embeddings, Machine Learning 110 (2021) 2905–2940.
[34] P. Lambert, C. de Bodt, M. Verleysen, J. A. Lee, Squadmds: A lean stochastic quartet mds improving global structure preservation in neighbor embedding like t-sne and umap, Neurocomputing 503 (2022) 17–27.
[35] D. Dua, C. Graff, UCI machine learning repository, 2017. URL: http://archive.ics.uci.edu/ml.

A. Clustered BIOT

As mentioned in Section 3.3, the main method proposed in this paper can be fine-tuned with a method we call Clustered BIOT. Let $z_{i\ell} = 1$ if instance i is in region ℓ and 0 otherwise. The matrix $\vec{Z}$ containing all elements $z_{i\ell}$ respects the general conventions of hard clustering (each instance belongs to exactly one cluster and each cluster contains at least one instance). Then, the objective function for Clustered BIOT is

$$J_1(\vec{Z}, \{\vec{W}^{(\ell)}, \vec{w}_0^{(\ell)}, \vec{R}^{(\ell)}\}|_{\ell=1}^{L}) = \frac{1}{2n} \sum_{i=1}^{n} \sum_{\ell=1}^{L} z_{i\ell} \sum_{k=1}^{2} (\vec{y}_i^{\top} \vec{r}_k^{(\ell)} - w_{0k}^{(\ell)} - \vec{x}_i^{\top} \vec{w}_k^{(\ell)})^2 + \lambda \sum_{k=1}^{2} \|\vec{w}_k^{(\ell)}\|_1 \tag{2}$$

which is minimized w.r.t $\vec{Z}$ and $\{\vec{W}^{(\ell)}, \vec{w}_0^{(\ell)}, \vec{R}^{(\ell)}\}|_{\ell=1}^{L}$ under the constraints that (i) $\vec{R}^{(\ell)}$ is an orthogonal matrix $\forall \ell$ and (ii) $\vec{Z}$ respects the clustering conventions above.

For fixed $\vec{Z}$, the solution for $\{\vec{W}^{(\ell)}, \vec{w}_0^{(\ell)}, \vec{R}^{(\ell)}\}|_{\ell=1}^{L}$ can be found by training BIOT on each subset of instances $\mathcal{S}_\ell$, where $\mathcal{S}_\ell := \{i \mid z_{i\ell} = 1\}$. For fixed $\vec{W}^{(\ell)}$, $\vec{w}_0^{(\ell)}$ and $\vec{R}^{(\ell)}$ and a given instance i, the solution for $\vec{z}_i$ is the vector that minimizes

$$\sum_{\ell=1}^{L} z_{i\ell} \sum_{k=1}^{2} (\vec{y}_i^{\top} \vec{r}_k^{(\ell)} - w_{0k}^{(\ell)} - \vec{x}_i^{\top} \vec{w}_k^{(\ell)})^2. \tag{3}$$

Since only one element of $\vec{z}_i$ can be equal to one (instance i can belong to only one cluster), the optimal cluster for instance i is whichever model ℓ minimizes the prediction error:

$$\arg\min_{\ell} (\vec{y}_i^{\top} \vec{r}_k^{(\ell)} - w_{0k}^{(\ell)} - \vec{x}_i^{\top} \vec{w}_k^{(\ell)})^2. \tag{4}$$

Thus, Clustered BIOT can be optimized by alternating between clustering instances according to prediction error and fitting BIOT models to the clusters. An instance i is assigned to cluster ℓ if BIOT model ℓ has the lowest prediction error for that instance compared to the other models.