ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Supervised dimensionality reduction technique accounting for soft classes

Sorina Mustatea[1,2], Michaël Aupetit[3], Jaakko Peltonen[4], Sylvain Lespinats[1] and Denys Dutykh[2] *

1 - Univ. Grenoble Alpes, INES, F-73375, Le Bourget du Lac, France
2 - Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000 Chambéry, France
3 - Qatar Computing Research Institute, HBKU, Doha, Qatar
4 - Tampere University, Faculty of Information Technology and Communication Sciences, Finland

Abstract. Exploratory visual analysis of multidimensional labeled data is challenging. Multidimensional Projections for labeled data attempt to separate classes while preserving neighborhoods. In this work, we consider the case where instances are assigned multiple labels with probabilities or weights: for example, the output of a probabilistic classifier, fuzzy membership functions in fuzzy logic, or the share of votes for each candidate in an election. We propose a new technique to better preserve neighborhoods of such data. Our experiments show improved qualitative results compared to unsupervised, and existing dimensionality reduction techniques.

1 Introduction

Multidimensional Projection (MDP) techniques [1] are used for visualization and analysis of multidimensional data. They use Dimensionality Reduction (DR) techniques to visualize the data in a two-dimensional projection space graphically encoded as a scatterplot.

Supervised MDP takes labels into account in the embedding process. Whereas in a traditional scenario data would each be annotated with one of several categories, in soft labeling each sample has partial membership or assignment into multiple categories. For example, a person may work part-time in multiple jobs, having multiple job categories, a piece of music may be a mix of various music style categories, or a handwritten digit may be in between symbolic writing style categories. The soft labeling may represent true partial membership, the uncertainty of the annotator, or a combination of both. Regardless, it represents the best available ground truth for the samples and should be taken into account in DR.

We propose *SoftClassNeRV*, a supervised DR technique capable of considering soft classes in the embedding process. With soft labels as ground truth, our method works towards avoiding the separation of classes where a continuum exists. The soft labels become the main counter for the penalization degree in the embedding process. We show the essential aspects of our approach in Fig. 1.

*The work of SM and SL has been realized with the participation of INES.2S. The work of DD has been supported by the French National Research Agency, through the Investments for Future Program (ref. ANR−18−EURE−0016 — Solar Academy).

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

2 Related work

Unsupervised techniques such as Principal Component Analysis (*PCA*) [2], Isometric feature mapping (*Isomap*) [3], Uniform Manifold Approximation and Projection (*UMAP*) [4], Stochastic Neighbor Embedding (*SNE*) [5], $t-$distributed *SNE* (*t-SNE*) [6], Neighborhood Retrieval Visualizer (*NeRV*) [7, 8], focus on multidimensional neighborhoods. Labels are simply ignored, even if this information is available.

Supervised techniques such as Neighborhood Component Analysis (*NCA*) [9], supervised *Isomap* (*S-Isomap*) [10, 11], supervised *NeRV* (*S-NeRV*) [12], supervised *UMAP* (*S-UMAP*) [4] pursue the projection while seeking at the same time to preserve the class structure. However, those objectives might be contradictory, leading to class separation in the layout, even if they substantially overlap in the data space. Whatever the method, it is achieved by modifying the original distances according to classes.

Recently, *ClassNeRV* [13] was proposed to limit over-separation of the classes by balancing the type of distortion in the stress function. *ClassNeRV* focuses on data structure preservation but increases class preservation when it is compatible with the data neighborhoods. Our proposal extends *ClassNeRV* to soft classes preserving all information provided in the dataset during the MDP.

3 Supervised MDP without changing the original distances

We present here *SoftClassNeRV* which extends the *ClassNeRV* technique detailed below.

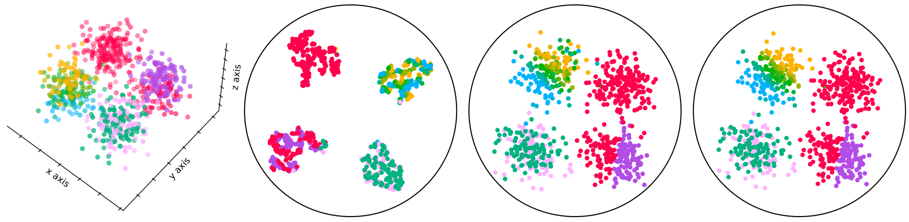
3.1 *ClassNeRV* stress function

Neighborhood Embedding (NE) techniques aim at the preservation of the local manifold structures following the rules indicated through a probabilistic framework. A probabilistic distribution for a point j to the neighborhood of point i is estimated in the original and projection space. In *ClassNeRV* [13, 14], this distribution is denoted in the data space as $\beta_i \coloneqq \{\beta_{ij}\}_{j \neq i}$ and in the embedded space as $b_i \coloneqq \{b_{ij}\}_{j \neq i}$, defined as follows:

$$\beta_{ij} = \frac{\exp\left(-\Delta_{ij}^2 / \left(2\sigma_i^2\right)\right)}{\sum_{k\neq i} \exp\left(-\Delta_{ik}^2 / \left(2\sigma_i^2\right)\right)} \quad \text{and} \quad b_{ij} = \frac{\exp\left(-D_{ij}^2 / \left(2s_i^2\right)\right)}{\sum_{k\neq i} \exp\left(-D_{ik}^2 / \left(2s_i^2\right)\right)}, \qquad (1)$$

where the Greek characters are used exclusively to describe the data space (input) while the Latin letters are used for the embedded space (output). The distances between the points i and j are indicated by Δ_{ij} and D_{ij}. Further, σ_i and s_i are scale parameters for data space and embedded space respectively. The *ClassNeRV* stress function is defined as a weighted sum over four Bregman

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.



(a) 3D dataset (b) *t-SNE* (c) *ClassNeRV* (d) *SoftClassNeRV*

Fig. 1: 2D mappings of four Gaussian clusters dataset (Section 4.1) with several methods. (a) The original dataset. (b) *t-SNE*(unsupervised) flattens the clusters, loosing their intra-class organization. (c) *ClassNeRV* accounts for the hard red/purple separation but the embedding of the soft light blue/yellow cluster is messed up. In contrast, (d) *SoftClassNeRV* shows a good mapping of both hard (bottom right) and soft classes (top left).

(B) divergences:

$$\begin{aligned}\Phi_{ClassNeRV} \coloneqq \sum_i \tau^{\in} D_B\left(\beta_i^{\in}, b_i^{\in}\right) + \left(1-\tau^{\in}\right) D_B\left(b_i^{\in}, \beta_i^{\in}\right) + \\ + \tau^{\notin} D_B\left(\beta_i^{\notin}, b_i^{\notin}\right) + \left(1-\tau^{\notin}\right) D_B\left(b_i^{\notin}, \beta_i^{\notin}\right).\end{aligned} \tag{2}$$

ClassNeRV takes advantage of classes to preserve neighborhood structures, thanks to the trade-off parameters $\tau^{\in}$ and $\tau^{\notin}$ (both in $[0,1]$) set according to within-class and between-class relations, respectively. Moreover, with B-divergence $D_B(\beta_i, b_i) = \sum_j \left(\beta_{ij} \log\left(\beta_{ij}/b_{ij}\right) + b_{ij} - \beta_{ij}\right)$, the positivity of the stress function is ensured over the four intervals. When the values for the trade-off parameters are set to be equal ($\tau^{\in} = \tau^{\notin}$), *ClassNeRV* reduces to *NeRV* [7,8] (see [13] for details).

Soft labels provide a supplementary level of information describing the similarity among points in terms of classes, so in the data output space, independent of the input space where clusters stand.

3.2 *SoftClassNeRV*

As in *ClassNeRV*, we aim at the preservation of data structures. In that pursuit, we take advantage of soft classes to drive the possible distortions where they impact the classification less. We advocate that the projection should take into account the soft labels assigned to each instance.

We denote α and γ user-defined parameters in $[0,1]$ to tune the degree to which the method accounts for the soft labels. Parameter α describes the level of supervision: $\alpha = 1$ leads to full supervision, while $\alpha = 0$ corresponds to no supervision at all. Parameter γ controls the balance between penalization of false and missed neighbors: $\gamma = 0$ denotes the least amount of false neighbors while $\gamma = 1$ denotes the least amount of missed neighbors.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

We denote the soft label assignment for each instance i as $\omega_i = [\omega_{i1}, \ldots, \omega_{iC}]$ where ω_{ic} is the membership in class c, and C is the number of classes. Further, we determine the class community λ_{ij}, the similarity between two points (i, j) according to the soft classes distribution: $\lambda_{ij} \coloneqq 1 - \sum_c (| \omega_{ic} - \omega_{jc} | / 2)$. The class community controls the penalization degree with which possible distortions must be weighted. Then, we redefine the stress function with the introduction of a parameter $\Lambda_{ij}^{\in}$ dependent on the class community and the user-defined parameters α and γ:

$$\Lambda_{ij}^{\in} \coloneqq (1-\alpha)\,\gamma + \alpha\,\lambda_{ij}. \tag{3}$$

Therefore, *SoftClassNeRV* stress function can be recast to a summation over two weighted B-divergences:

$$\Phi_{SoftClassNeRV} = \sum_i D_B\left(\beta_i, b_i; \Lambda_i^{\in}\right) + D_B\left(b_i, \beta_i; \left(1 - \Lambda_i^{\in}\right)\right) \tag{4}$$

where $D_B\left(\beta_i, b_i; \Lambda_i^{\in}\right) = \sum_j \Lambda_{ij}^{\in}\left(\beta_{ij} \log\left(\beta_{ij}/\, b_{ij}\right) + b_{ij} - \beta_{ij}\right)$ is a weighted B-divergence with weights $\Lambda_i^{\in} = \{\Lambda_{ij}^{\in}\}_j$.

In *ClassNeRV*, $\tau^{\in}$ prescribes the penalties degree of false and missed neighbors within classes, while $\tau^{\notin}$ prescribes the penalties between classes. In *SoftClassNeRV*, these parameters balance the contribution of the class community in the embedding process. The connection between *ClassNeRV* and *SoftClassNeRV* is dictated by the synergy between the user-defined parameters α and γ and the trade-off parameters $\tau^{\in}$ and $\tau^{\notin}$ through an internal rescaling process: $\tau^{\in} = (1-\alpha)\,\gamma + \alpha$ and $\tau^{\notin} = (1-\alpha)\,\gamma$. Therefore, *SoftClassNeRV* reduces to *ClassNeRV* when the ground truth is available as hard labels and it leads to *NeRV* in the absence of any labels.

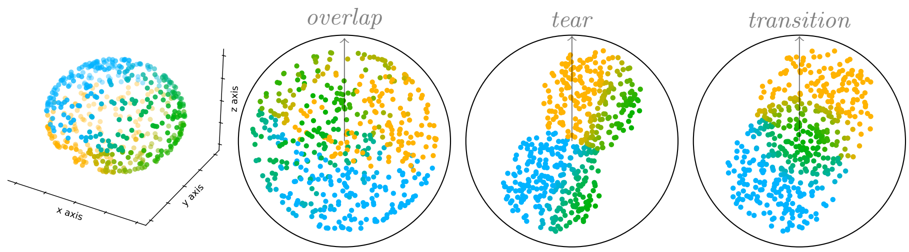
4 Experiments

We illustrate the characteristics of *SoftClassNeRV* and evaluate its performances qualitatively by comparing it with *NeRV* (unsupervised) and *ClassNeRV* (hard class supervised) on two synthetic datasets. The four Gaussian clusters dataset highlights the behavior of the mapping methods on data with hard and soft labels related or unrelated to the spatial neighborhoods. The "Pacman" dataset highlights the behavior of projecting manifold data with soft labels when distortions are unavoidable.

4.1 Datasets

The "four clusters dataset" (Fig. 1(a)) is a set of 800 points distributed equally between four Gaussian clusters. The points are spread in the $x-y$ plane with the same Gaussian noise distribution along the $z-$axis. The four clusters are a mix of six unbalanced classes: a cluster composed purely of one class (purple); a cluster of two hard classes separated in half by a surface parallel with the $x-y$ plane (purple, red); a cluster of two randomly distributed hard classes (dark green, pink) and a cluster exhibiting two soft classes (light blue and yellow) melting

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.



(a) 3D dataset (b) *NeRV* (c) *ClassNeRV* (d) *SoftClassNeRV*

Fig. 2: The sensitivity of different mapping techniques to hard and soft labels is illustrated with the "Pacman" dataset (a) described in Section 4.1. (b) *NeRV* ignores class labels and shows overlapping of all classes whether soft or hard. (c) *ClassNeRV* considers only hard classes and unnecessarily tears the soft classes (green area split among the two hemispheres). *SoftClassNeRV* accounts for soft classes by tearing the Pacman sphere only between yellow and blue hard classes (d), preserving better the soft transition between them (greenish points).

into each other following a linear interpolation along the $z-$axis, forming a soft classification encoded as a continuum of green tints from light blue to yellow. Each point of this latter cluster can be seen as a soft distribution of green tints between yellow and light blue, according to their position in the data (original) space relative to the separation plane.

The "Pacman" dataset in Fig. 2 is a collection of 500 uniformly distributed points on the surface of a sphere. The hemispheres along the $z-$axis delimit two hard classes (light blue and yellow points). These classes are pure in the left hemisphere along the $x-$axis, while they are soft in the right hemisphere, with a progressive mix based on the angular position around the $y-$axis, graphically encoded as a continuum of green tints from pure light blue to pure yellow).

4.2 Results and discussion

We refer to Fig. 1 and Fig. 2 for the description of the results. The difference between various maps lies in the choice of the penalization of the different types of (unavoidable) distortions by the DR techniques. Our experiments show that when soft labels match well with the data neighborhood structure, *SoftClassNeRV* can concentrate the distortions in locations less detrimental to this matching: between two clusters or between hard classes in a single cluster. When soft labels are ignored (*ClassNeRV*, *NeRV*), distortions may appear anywhere, including where they unnecessarily blur the organization of classes impairing the visual analysis of soft class structures.

Of course, if a more complex mix of soft classes exists in the data, it may not be possible to embed them without distortions in 2D maps even with *SoftClassNeRV*. Still, our experiments show that even simple data-class structures can be missed by advanced nonlinear mapping techniques like *ClassNeRV* and *NeRV* when they are not designed to preserve them.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

5 Conclusion

Multidimensional projection techniques cannot avoid mapping distortions [1]. This work presents *SoftClassNeRV*, an extension of *ClassNeRV* [13] to deal with data with soft label assignments. We illustrate its main characteristics and benefits in reducing distortions, compared to *ClassNeRV* and *NeRV* when mapping soft-label data. We aim to use *SoftClassNeRV* to explore the output of probabilistic classifiers or voting results. Still, further extensive quantitative experiments with real data are needed and planned as future work.

References

[1] Luis Gustavo Nonato and Michael Aupetit. Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2650–2673, 2018.

[2] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

[3] Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[4] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[5] Geoffrey E Hinton and Sam Roweis. Stochastic neighbor embedding. *Advances in neural information processing systems*, 15, 2002.

[6] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[7] Jarkko Venna and Samuel Kaski. Nonlinear dimensionality reduction as information retrieval. In Marina Meila and Xiaotong Shen, editors, *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, volume 2 of *Proceedings of Machine Learning Research*, pages 572–579, San Juan, Puerto Rico, 21–24 Mar 2007. PMLR.

[8] Jarkko Venna, Jaakko Peltonen, Kristian Nybo, Helena Aidos, and Samuel Kaski. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *Journal of Machine Learning Research*, 11(2), 2010.

[9] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. *Advances in neural information processing systems*, 17, 2004.

[10] Chun-Guang Li and Jun Guo. Supervised isomap with explicit mapping. In *First International Conference on Innovative Computing, Information and Control-Volume I (ICICIC'06)*, volume 3, pages 345–348. IEEE, 2006.

[11] Xin Geng, De-Chuan Zhan, and Zhi-Hua Zhou. Supervised nonlinear dimensionality reduction for visualization and classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 35(6):1098–1107, 2005.

[12] Jaakko Peltonen, Helena Aidos, and Samuel Kaski. Supervised nonlinear dimensionality reduction by neighbor retrieval. In *2009 IEEE international conference on acoustics, speech and signal processing*, pages 1809–1812. IEEE, 2009.

[13] Benoît Colange, Jaakko Peltonen, Michaël Aupetit, Denys Dutykh, and Sylvain Lespinats. Steering distortions to preserve classes and neighbors in supervised dimensionality reduction. *Advances in neural information processing systems*, 33:13214–13225, 2020.

[14] Sylvain Lespinats, Benoit Colange, and Denys Dutykh. *Nonlinear Dimensionality Reduction Techniques: A Data Structure Preservation Approach.* Springer, 2022.