



HAL
open science

Supervised dimensionality reduction technique accounting for soft classes

Sorina Mustatea, Michaël Aupetit, Jaakko Peltonen, Sylvain Lespinats, Denys
Dutykh

► **To cite this version:**

Sorina Mustatea, Michaël Aupetit, Jaakko Peltonen, Sylvain Lespinats, Denys Dutykh. Supervised dimensionality reduction technique accounting for soft classes. ESANN 2022: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Oct 2022, Bruges, Belgium. pp.13, 10.14428/esann/2022.ES2022-26 . cea-04176573

HAL Id: cea-04176573

<https://cea.hal.science/cea-04176573v1>

Submitted on 3 Aug 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Supervised dimensionality reduction technique accounting for soft classes

Sorina Mustatea¹, Michaël Aupetit², Jaakko Peltonen³,
Sylvain Lespinats¹ and Denys Dutykh⁴ *

1- Univ. Grenoble Alpes, INES, F-73375, Le Bourget du Lac, France

2- Qatar Computing Research Institute, HBKU, Doha, Qatar

3- Tampere University,

Faculty of Information Technology and Communication Sciences, Finland

4- Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA,
73000 Chambéry, France

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 votes of each voters 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 learning deals with soft labeling, while in DR, this practice has to be adopted. 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 the soft labels as ground truth, our method works towards avoiding the separation of classes where a continuum

*The work of SM and SL has been realised 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).

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.

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 organization. Our proposal extends *ClassNeRV* to soft classes preserving all information provided in the dataset during the MDP.

3 Supervised MDP without changing 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 $\beta_i := \{\beta_{ij}\}_{j \neq i}$ and in the embedded space as $b_i := \{b_{ij}\}_{j \neq i}$, defined as follows:

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

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 points i and j are denoted as Δ_{ij} and D_{ij} . Further, σ_i and

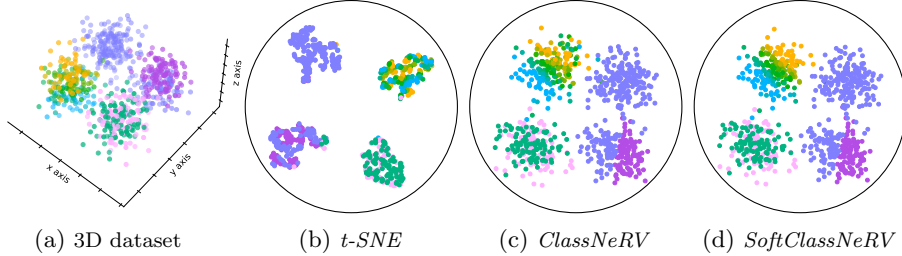


Fig. 1: 2D mappings of four Gaussian clusters dataset with several methods. Insert (a) presents the original dataset. The purple/blue cluster has a hard linear separation, while the light blue/yellow cluster has a soft linear separation; one cluster is pure blue and in the last cluster, pink and dark green items overlap. *t-SNE* (insert (b)) (unsupervised) has flattened these clusters somehow that the intra-class organization is lost. *ClassNeRV* (insert (c)) accounts for the hard blue/purple separation but the embedding of the light blue/yellow cluster is partially mixed. *SoftClassNeRV* (insert (d)) reveals that soft linear separation is possible.

s_i are scale parameters for data space respectively embedded space. *ClassNeRV* stress function is defined as a weighted sum over four Bregman (B) divergences:

$$\begin{aligned} \Phi_{ClassNeRV} := & \sum_i \tau^\epsilon D_B(\beta_i^\epsilon, b_i^\epsilon) + (1 - \tau^\epsilon) D_B(b_i^\epsilon, \beta_i^\epsilon) + \\ & + \tau^\not\epsilon D_B(\beta_i^\not\epsilon, b_i^\not\epsilon) + (1 - \tau^\not\epsilon) D_B(b_i^\not\epsilon, \beta_i^\not\epsilon). \end{aligned} \quad (2)$$

ClassNeRV takes advantage of classes to preserve neighborhood structures, thanks to the trade-off parameters τ^ϵ and $\tau^\not\epsilon$, set according to within-class and between-class relations correspondingly, both belong to the segment $[0, 1]$. Moreover, with B-divergence $D_B = (\beta_{ij} \log(\beta_{ij}/b_{ij}) + b_{ij} - \beta_{ij})$ 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 ($\tau^\epsilon = \tau^\not\epsilon$), *ClassNeRV* reduces to *NeRV*.

Soft labels provide a superior level of information describing the relations among points. These points may interact with elements from other classes sharing the same cluster or not. This information cannot be directly included as part of the input variable as it assumes the order of the labels while they have no specific ordering. Therefore, we cannot adopt an approach similar to “regression” by just using weight for each label as additional inputs.

3.2 *SoftClassNeRV*

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

We designate α and γ , two user-defined inputs, taking values within the segment $[0, 1]$ to tune the degree to which the method is accounting 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.

We designate ω the soft label assigned to each instance. Further, we determine the class community, the degree of similarity between two points according to the soft classes distribution: $\lambda_{ij} := 1 - \sum_{i,j} (|\omega_i - \omega_j| / 2)$. Then, we redefine the stress function with the introduction of a parameter Λ dependent on the class community and the user-defined parameter:

$$\Lambda^\epsilon := (1 - \alpha)\gamma + \alpha\lambda. \quad (3)$$

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

$$\Phi_{SoftClassNeRV} = \sum_i \Lambda^\epsilon D_B(\beta_i, b_i) + (1 - \Lambda^\epsilon) D_B(b_i, \beta_i). \quad (4)$$

In *ClassNeRV*, τ^ϵ prescribes the penalties degree of false and missed neighbors within classes, while τ^\neq prescribes the penalties between classes. In *SoftClassNeRV*, those parameters balance the contribution of the class community has 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 through an internal rescaling process: $\tau^\epsilon = (1 - \alpha)\gamma + \alpha$ and $\tau^\neq = (1 - \alpha)\gamma$. Therefore, the method reduces to *ClassNeRV* when the ground truth is available as hard labels and it leads to *NeRV* in the absence of any labels.

4 Experiment

To illustrate the characteristics of our method and to evaluate its performance we compare it with *NeRV* (unsupervised) and *ClassNeRV* (hard class supervised).

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 organization.

The ‘‘Pacman’’ dataset highlights the behavior of projecting data with soft labels when distortions are unavoidable.

4.1 Datasets

The ‘‘four clusters dataset’’ (fig. 1(a)) embodies Gaussian clusters summing up to 800 points distributed equally between the clusters. The points are dispersed in a $x - y$ plane with noise along z -axis. The clusters are split in 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, blue); a cluster of two randomly distributed hard classes (dark green, pink) and a

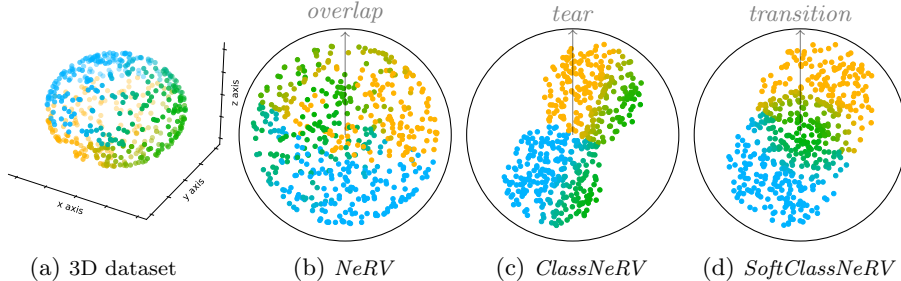


Fig. 2: Method’s sensitivity to the labels definition is displayed on the “Pacman” dataset(a) described in section 4.1, the continuum in the green hue codes for the soft labels between class blue and yellow. The map reached by *NeRV*(b) (which does not consider classes) shows overlapping. *ClassNeRV* (which only consider hard classes) unnecessary tears the soft class (c). *SoftClassNeRV* maps accounted for soft classes which allows tearing the sphere only between yellow and blue classes (d).

cluster exhibiting two soft classes (light blue and yellow) melting into each other according to a linear function, along z -axis, into a soft representation (encoded as a continuum of green tints from light blue to yellow). Each point of this cluster can be seen as a soft distribution of green tints between yellow and blue, according to their position in the data (original) space relative to the separation plane.

The “Pacman” dataset embedded in fig. 2 represents a collection of 500 uniformly distributed points on the surface of a sphere. The hemispheres identified along the z -axis embody three categories of classes. One hemisphere is split by a surface parallel with the $x-y$ plane and in the middle of the z -axis exhibiting two hard classes (light blue and yellow points). The other hemisphere illustrates a soft class collection of points (encoded as a continuum of green tints from light blue to yellow). The level in green tint for each point is determined by the position on the sphere in rapport with the light blue and yellow classes.

4.2 Discussion

Distortions are unavoidable in the embedding process. The difference among various embeddings lies in the choice of the type and the position of distortions.

Our experiments show that when soft labels convey the data structure, the positioning of unavoidable distortions may be directed to a position where they disturb the mapping less. When soft structures are disregarded, tears or overlaps may appear anywhere, including where they unnecessarily blur the organization of classes leading to confusion in the visualization and interpretations process.

5 Conclusion

This work presents an extension of an existing DR technique. Due to the curse of dimensionality, the incomplete sampling, or the synergy between the dataset

and the technique, distortions are unavoidable in MDP. Our work contributes in diminishing the severity of distortions for samples annotated with soft labels. The soft labeling may come from the way the data is annotated or from a pre-processing step. Moreover, the method provides a way to project the hard label data, either by reverting to *ClassNeRV* [13] or with a pre-processing step, creating subsequent soft labels, for example, a classifier’s output. This technique may provide another way to visualize uncertainty in the data, it may be a way to visualize and explore the output of a classifier, or it may be a solution to evaluate different classifiers.

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, Benoît Colange, and Denys Dutykh. *Nonlinear Dimensionality Reduction Techniques: A Data Structure Preservation Approach*. Springer, 2022.