FPGheArt – Data visualization with t-SNE and clustering with DBSCAN

Lorenzo Buriola, Filippo Conforto, and Lorenzo Domenichetti

(Dated: August 5, 2021)

Clustering has always been one of the most challenging computing tasks and it gets harder as the data complexity or dimensionality increases. Density- and distance-based approaches have been developed to succeed over a wide range of sets - DBSCAN, belonging to the former group, is among the most successful ones. As labels are not always present, a visualization tool for results check is paramount. Among the most powerful is t-SNE, a dimensionality reduction technique not preserving neither original distance nor density [1]. Nevertheless, it often manages to identify relevant local structures in the original samples - if present. The application of these two techniques (alone and coupled) is presented in this paper. For a meaningful comparison, two different datasets were chosen. Results may be surprising at first glance, but surely recall the complexity of the clustering task.

INTRODUCTION

The application of clustering algorithms has enlarged since the digital revolution. The task of grouping different objects into subsets has been among the most challenging one - placed directly into the unsupervised learning frame.

To thoroughly test the performance of different algorithms, two different datasets are studied - the first one contains 5-dimensional points in which the samples are somehow *nested*; the second one instead contains 36-dim bits (or spins), which aims to challenging density-based algorithms - as DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) [2]. Even if clustering is typically an unsupervised task (so labels are unknown), in this application for both the first and the second set the *true labels* are available. This feature helps classifying different procedures and parameters - one of the main goals of this work. From the previous information, it is possible to learn that the first set is formed by three distinct clusters, the second one by five.

One of the most challenging problems in data science is tackling effectively high-dimensional samples - the *curse of dimensionality* always kicks in [1]. Various techniques have been developed to reduce the number of features; the most famous one, based on linear transformation, is the PCA (*Principal Component Analysis*) technique [3]. More sophisticated approaches try to preserve local structures also applying non-linear transformation, as for example t-SNE (*t-Stochastic Neighbor Embedding*)[4]. In this article, an analysis of how different dimensionality reduction tools and clustering algorithms behave on the two datasets and how their performance couples is proposed.

METHODS

Two different techniques are exploited for dimensionality reduction - PCA and t-SNE. While the former's results are related to the highest variability features, the performances of the latter depend on the *perplexity*, a parameter linked to the intrinsic density of the data that "can be interpreted as a smooth measure of the effective number of neighbors" [4]. Typical parameters lay in the $[5, 50]$ interval, as suggested in the original paper. The main idea behind t-SNE is to associate a probability for two points in the original space to belong to the same local structure. Then, t-SNE looks for a transformation to a lower dimensional space in which most probabilities are preserved. The result is optimized using the Kullback-Leibler divergence, so that high penalties are returned when two originally close points are placed far away and viceversa.

Unsupervised learning on data is performed using a density-based algorithm, DBSCAN. This is in general among the most powerful clustering methods. Its performances strongly depend on two parameters: *min_Pts* and ϵ. The former is related to the minimum number of close points required to form a cluster, while the latter represents the minimum distance for two data to be considered as belonging to the same group. As the name suggests, DBSCAN manages to group together points looking at the local density. It starts building clusters from data which have at least *min_Pts* points in their ϵ-neighbourhood, labeling as outliers those in low-density areas. The two parameters must be tuned to get a reasonable output.

The code was developed in *python* using a *jupyter* framework, and it's available here. All methods used are part of the scikit-learn library (v. 0.24.2).

RESULTS

5-Dimensional Dataset

An initial visualization of the dataset is produced simply selecting the first three features, displayed on a 3D plot (fig. 1).

While observing carefully the reductions in fig. 2, it is noticeable that using only linear transformations, the nested clusters are not actually separated - they only seem to be rotated. Regarding t-SNE, changing the perplexity parameter drastically modifies the reconstruction. A too low parameter implies a failure in the reconstruc-

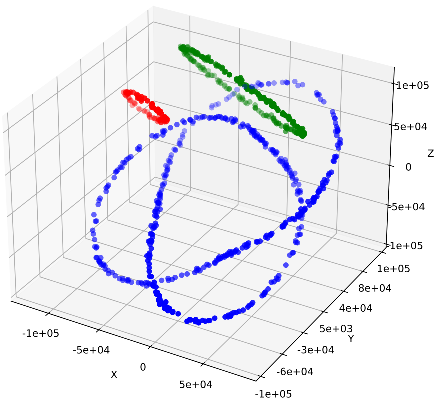

FIG. 1. 3D projection of the first dataset.

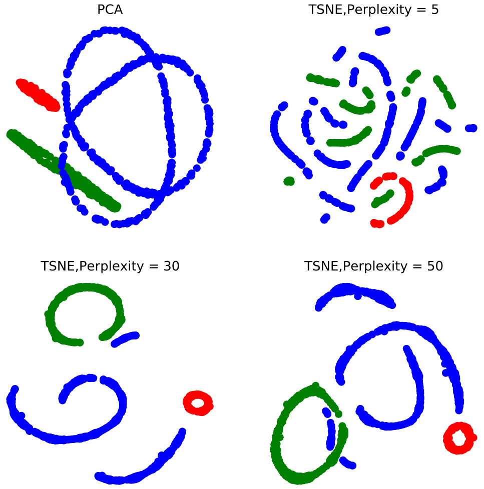

FIG. 2. Visual comparison of 2D t-SNE and PCA.

tion of the initial links, while higher ones do not manage to separate completely the nested clusters. The intermediate perplexity case is the most accurate one - clusters are not properly grouped, but the knot is loose. This last reduction will be the one exploited for presenting fig. 3. The choice of the DBSCAN parameters has to be performed carefully. While not many references on how to pick the right *min_Pts* parameter are available, some proposals have been displayed for the ϵ selection [2] [5]. The main idea is to plot the distance of a point from its k-th neighbour, and then to select the ϵ parameter as the point in which the maximum curvature is reached.

Fig. 3 shows the results for the 3- and 5- neighbour case - these two values were found to be the best choice for the *min_Pts* parameter (see next - fig. 4). The 3-neighbour case reaches its maximum curvature point at about 7500, while the other one at 10000. These two parameters should be, accordingly to the literature, our best choice for the ϵ parameter. Nonetheless, when running DBSCAN with such parameters, the results are not as good as one may expect. Such a procedure has proven successful where there is the need to discriminate between inliers and outliers. Points which have a distance larger than the selected ϵ are the ones not included in any cluster. In this specific case, where no outliers are present, this procedure results in finding multiple clusters without capturing the bigger picture.

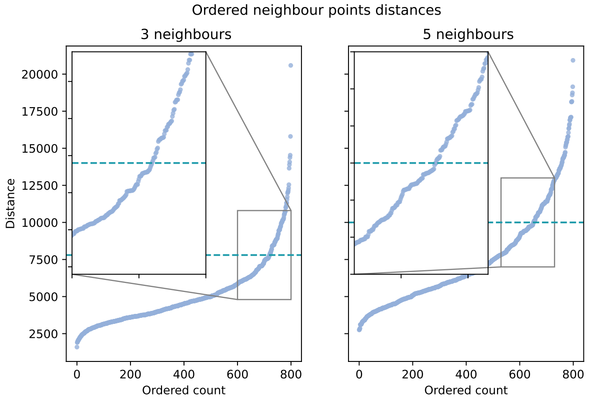

FIG. 3. Distance of each point from k-th neighbour.

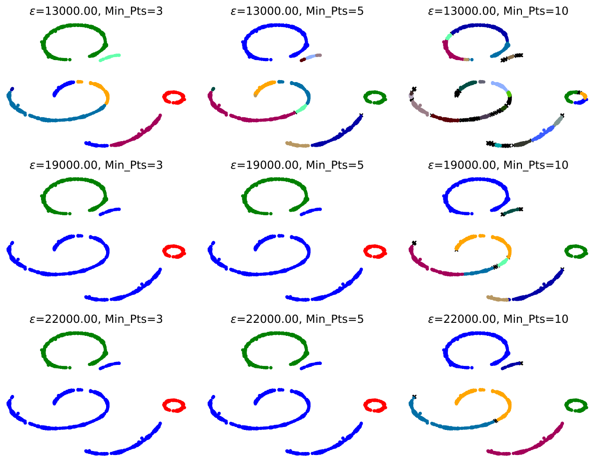

FIG. 4. Grid search for best DBSCAN parameters. Black crosses - if any - represent outliers.

In the best case, DBSCAN manages to reconstruct the original cluster exactly. However, even lower ϵ values return results comparable to the first row ones: the ϵ procedure fails in this case. In general, it is not safe to rely on such technique, as also in other cases it does not reach the best result [6]. The main information on the k-th distance plot is related to the order of magnitude to look for in the ϵ search. More advanced clustering algorithms nowadays have been developed to overcome the ϵ choice, one of the main weaknesses of DBSCAN. One example based on DBSCAN is OPTICS [7], that does not need an ϵ parameter to be tuned and returns directly the best choice with a single line of code. On the other hand, OPTICS performances are reported to be 1.6 times slower compared to a DBSCAN run. In this specific application, OPTICS works as the best DBSCAN case - the only parameter chosen was *min_Pts*, and again both 3 and 5 performed smoothly.

A deeper indicator that can be computed to evaluate the performance of the clustering algorithm is the Normalized Mutual Information (*NMI*). This concept is borrowed from the information theory field [8]. In this case, it measures the amount of information that the predicted labels convey about the true ones. The value of this estimator can be between 0 and 1. A NMI equal to 1 defines the best possible result as it corresponds to the case in which all clusters are correctly classified - everything about the true labels can be known looking at the predicted ones. Instead, if the NMI is 0, the formed clusters do not convey any information about the original ones.

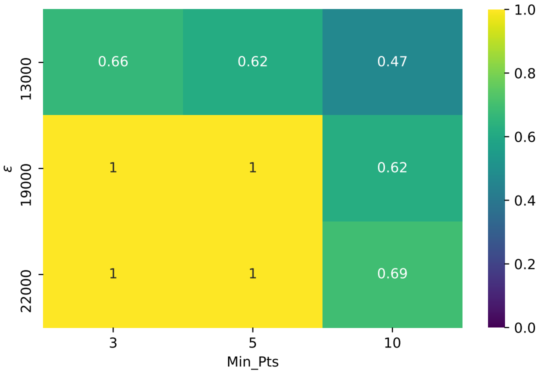

FIG. 5. NMIs associated to the *gridSearch* in fig. 4

The results provided in fig. 5 by the Mutual Information reflect the results contained in fig. 4 - the best choice of DBSCAN's parameters leads to a NMI equal to 1 as the clusters are correctly reconstructed. Moreover, OPTICS provides a full score for both *min_Pts* 3 and 5.

36-Dimensional Bits

To visualize the 36-dim bins, the t-SNE procedure is exploited one other time (fig. 6). There are no major differences in this case when changing the perplexity parameter. The most challenging cluster to be recognised seems to be the central one, as it is often mixed up with other groups.

One of the main challenges of this dataset is the definition of a meaningful distance. A simple choice is represented by the L1 distance (*"Manhattan" distance*). Anyhow, trying to run DBSCAN on the original datasets provides really poor performance. Using this algorithm, none of the available distances, included L1, seem to provide acceptable results. At this point, the failure with this application may be attributed either to 36-dim distances or to a density approach or to DBSCAN itself. Such issues are commonplace when dealing with high-dimensional categorical data. Therefore new approaches have been developed, for example coupling hierarchical and density-based clustering [9][1]. Among those kind of approaches is also HDBSCAN (*Hierarchical DBSCAN*)[2][10] - in this case, the ϵ selection is overcome by the hierarchical first step.

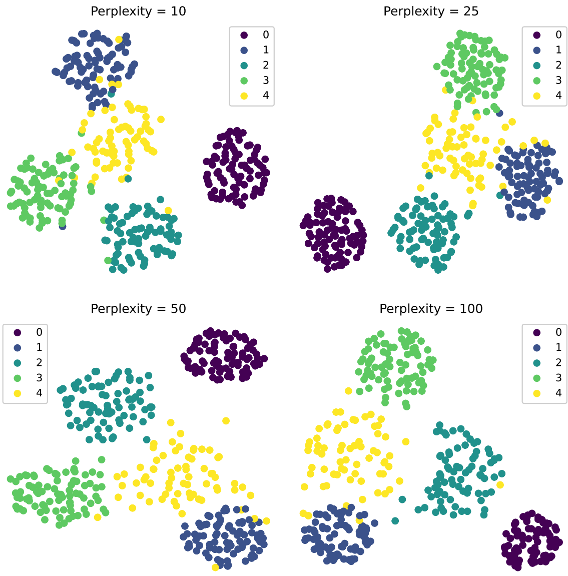

FIG. 6. Visualization of the second set.

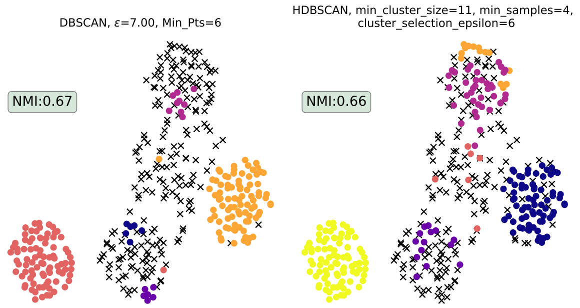

FIG. 7. Comparison of best results in the case of DB- and HDBSCAN. Black crosses represent outliers.

As shown in fig. 7, the hierarchical algorithm manages to find some structures in all five clusters. However, not all points are recognised as inliers and the trickiest central

[1] The algorithm proposed in this paper has not yet a dedicated python library, but its publication is now under development.

[2] HDBSCAN is not yet a part of the scikit-learn package. A reliable python library is available online.

cluster is only slightly captured. Even if the performance of HDBSCAN seems better from a visual point of view, the NMIs computed are similar - probably due to some incorrect classifications. In both cases, the results are far from being optimal. The failure of both algorithms may then be related to the density-based approach itself. It is also worth noticing that in this case the application of DBSCAN *after* t-SNE gives a smooth result, as expected from the intuitive visualization. Instead, in the 36-dimension spin space classification fails. There is still an open debate on the chance to use these two techniques combined, but there seems not to be a clear response - as t-SNE does not preserve distance nor density, the results of such a process may not reflect the structure of the initial set.

Trying to attribute furthermore the failure to the density-based approaches, two additional classical algorithms have been tried - K-Means and Agglomerative Clustering [1]. On the downside, these kinds of algorithms need the number of clusters that have to be reconstructed, but the results are surprisingly accurate.

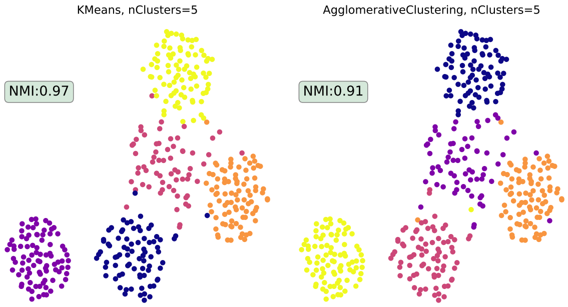

FIG. 8. Comparison of K-means (Euclidean distance) and Agglomerative Clustering (Manhattan distance).

Fig. 8 definitely addresses the issue of clustering 36-dim bins to the density-based approach. Indeed, the two cases show an accurate reconstruction (high NMI) using two distances (*Euclidean* and *Manhattan*) over which DBSCAN fails. Therefore, both L1 and L2 distances manage to find meaningful structures in the dataset, even if at first sight they may seem not appropriate for a categorical space.

CONCLUSIONS

Unsupervised tasks have proven to be among the most challenging and difficult ones. Since most times labels are not present, the study of the results is always approximate. The two main strategies for tackling the clustering task are distance- and density-based. Each one of them has its own weaknesses, and choosing the best one strongly depends on the considered case. Moreover, clustering gets increasingly complicated as the dimensionality of the sample increases. Dimensionality reduction techniques such as t-SNE or PCA can be helpful for visualisation and cross-check. PCA preserves distances and so it is more suited for coupling with clustering algorithms. Anyhow, in some applications the coupling of mighty techniques such as t-SNE proves to be fundamental to get an acceptable result.

As shown in this paper, the application of DBSCAN succeeds on a five-dimensional dataset with some complications introduced by the nesting of two clusters. t-SNE on his side also manages to loose the bind between the two. Anyhow, it is worth noticing that in this case the application of DBSCAN to the 2D reduced data after t-SNE leads to poor performance.

On the other hand, the application of DBSCAN to 36-dimension categorical data clearly fails - and neither its hierarchical version is able to find meaningful clusters. In this case, applying DBSCAN after the t-SNE reduction has proven successful. However, this result must not be generalized, as t-SNE does not preserve distance nor density. The information contained in the original data may be lost in the reduced ones. Distance-based approaches instead manage to be successful in the case of categorical data and their main downside is the need of the definition of the number of clusters to be searched for. In many applications, such a requirement may be a too strong limitation.

[1] Mehta, Pankaj, et al. *"A high-bias, low-variance introduction to machine learning for physicists"*, Physics reports 810 (2019): 1-124.

[2] Ester, M. et al., *"A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise"*, Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, (1996).

[3] K. Pearson, *"On Lines and Planes of Closest Fit to Systems of Points in Space"*. Philosophical Magazine. 2 (11): 559–572. (1901).

[4] L. v.d. Maaten and G. Hinton, *"Visualizing Data using t-SNE"*, Journal of Machine Learning Research, Vol. 9: pp 2579–2605, 2008.

[5] N. Rahmah N. and I. S. Sitanggang, 2016 IOP Conf. Ser.: Earth Environ. Sci. 31 012012

[6] Available - Online

[7] M. Ankerst *et al.*, *"OPTICS: Ordering Points To Identify the Clustering Structure"*,ACM SIGMOD international conference on Management of data. ACM Press. pp. 49–60. 1999.

[8] X. Liu *et al.*, *"Evaluation of Community Detection Methods"*, 2018, arXiv:1807.01130

[9] B. Andreopoulos *et al.*, *"Efficient layered density-based clustering of categorical data"*, Journal of Biomedical Informatics, 42 – 2, 2009. pp. 365–376.

[10] McInnes, L., and Healy, J. (2017), *"Accelerated Hierarchical Density Based Clustering"*, 2017 IEEE International Conference on Data Mining Workshops (ICDMW).