EARTHQUAKES – DBSCAN and t-SNE

Michele Guadagnini, Alessandro Lambertini, Alice Pagano, and Michele Puppin
(Dated: May 9, 2020)

Pattern recognition is a key learning task and clustering methods are one of the most powerful implementations for this aim. Moreover, data visualization techniques are needed to analyze data with several features, which can be challenging for learning algorithms. This work presents DBSCAN clustering method for unsupervised learning and t-SNE method for data visualization. After a theoretical overview of the two algorithms, we test their performances using a 5-dimensional labeled dataset with three embedded manifolds.

INTRODUCTION

Machine Learning techniques describe how to learn a model that captures meaningful information from the data. In Unsupervised Learning no labels are given to the learning algorithm, leaving it on its own to find structure in its input. Clustering methods are the simplest and most widely used implementations to achieve this aim. Clustering consists in grouping a set of objects according to some analogies such as the density distribution of the data points. In Density-Based Clustering, clusters are regions of data space with higher density of data points separated from each other by contiguous regions of low point density. The data points in the separating regions of low point density are typically considered noise or outliers [5][4].

One realization of DB Clustering is DBSCAN, or Density Based Spatial Clustering of Applications with Noise, which requires two parameters. The parameter ε-neighborhood defines the radius of neighborhood around each data point, while the parameter $minPts$ is the minimum number of neighbors within ε radius. DBSCAN algorithm analyzes each point on a given dataset and classify them in three main categories. Any point in the dataset, with a neighbor count greater than or equal to $minPts$, is marked as a *core point*. If the number of its neighbors is less than $minPts$, but it belongs to the ε-neighborhood of another core point, it is marked as *border point*. Finally, if a point is neither a core nor a border point, then it is called a noise point or an *outlier*. Moreover, a data point x_i is said to be *density reachable* from another data point x_j if there is a path of core points leading from x_j to x_i. Thus a cluster is defined as a group of density reachable points. The main advantage of DBSCAN, or Density-Based clustering in general, is its capability of discovering clusters with arbitrary shapes. In addition, the computational complexity of efficient implementations of DBSCAN can achieve $O(N \ln N)$, where N is the length of the dataset [5][1].

Often large amount of data with several features are provided and data complexity can be challenging for learning algorithms. However, real data lay in a much lower dimensional space than the original space since in reality they are not uniformly distributed. Furthermore, in order to choose the best predictive model one has to take into account the geometry and distribution of the data [5].

Therefore, data visualization methods are introduced to reduce data complexity and to improve the understanding of data structure. Dimensionality reduction is performed to project the data from the *original space* onto a lower dimensional space, which is called the *latent space*. Specifically, t-stochastic neighbor embedding (t-SNE) is a technique for dimensionality reduction which preserves the dataset local structure and it is particularly well suited for the visualization of high-dimensional datasets. The core point is to associate to each data point a Gaussian probability distribution for its neighborhood. The latter is defined as a symmetrized and normalized distribution $p_{ij} \equiv (p_{i|j} + p_{j|i})/(2N)$ where

$$p_{i|j} = \frac{\exp\left(-\|x_i - x_j\|^2/2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_i - x_k\|^2/2\sigma_i^2\right)} \qquad (1)$$

which provides an high probability for nearby points. The symmetrization ensures that $\sum_j p_{ij} > 1/(2N)$ for all data points, as a result of which each data point makes a significant contribution to the cost function which will be defined below. Moreover, there is not an optimal value of the variance σ_i for all data points: a smaller value of σ_i is suited for points in high density regions. This is ensured by fixing the value of the local entropy $H(p_i) \equiv -\sum_j p_{j|i} \log_2 p_{j|i}$ equal to $\log_2 \Sigma$, where Σ is called *perplexity*. This is equivalent to imposing that all points have the same amount of information. In the latent space an analogous, but long-tail, probability distribution is associated to each embedded data point. It is defined as

$$q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq i} \left(1 + \|y_i - y_k\|^2\right)^{-1}}. \qquad (2)$$

The long-tail shape ensures that short distance information are preserved while far-away points are mapped to relatively large distances. The aim of t-SNE is to find a low-dimensional data representation that minimizes the mismatch between p_{ij} and q_{ji}. This is achieved by minimizing the cost function given by the Kullback-Leibler
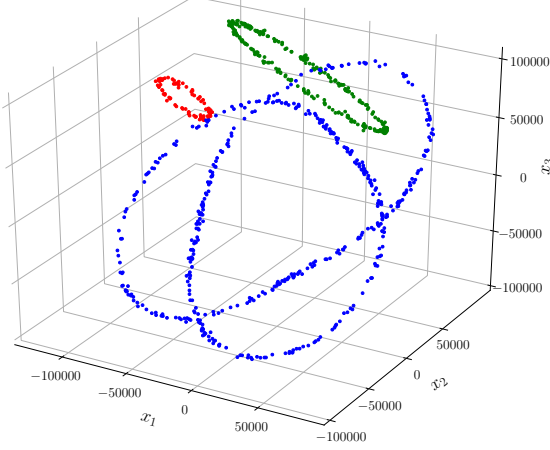
FIG. 1: Representation of the manifolds using only the first three dimensions of the original space. Different colors are used to discern the three manifolds.

FIG. 2: Histogram of distances between each point and its k nearest neighbours.

divergence between the distributions in the original and latent space:

$$\mathcal{C}(Y) = D_{KL}(p||q) \equiv \sum_{ij} p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right). \qquad (3)$$

Gradient descent is used for minimizing the cost function and the gradient results

$$\partial_{y_i}\mathcal{C} = \sum_{j\neq i} 4p_{ij}q_{ij}Z_i(y_i - y_j) - \sum_{j\neq i} 4q_{ij}^2 Z_i(y_i - y_j) \quad (4)$$

where $Z_i = 1/(\sum_{k\neq i}(1 + \|y_k - y_i\|^2)^{-1})$. Note that the minimization via gradient descent produces a stochastic algorithm. Moreover, t-SNE can produce equivalent result by rotating data and can deform scales in the latent space. Eventually, it is a computationally intensive algorithm with a complexity of $O(N^2)$ [5][3].

METHODS

In this work we analyze a dataset containing three embedded manifolds with a linear closed structure in a 5-dimensional space as represented in Fig. 1. We use pre-labeled data for the aim of testing the performances of DBSCAN and t-SNE.

In order to discern the three clusters, we use DBSCAN algorithm of the *scikit-learn* library [6] on the rescaled 5-dimensional dataset. A similar, but more noisy, dataset is tested as well. To make the algorithm work properly, meaningful values of the parameters ε and $minPts$ are needed. First of all, we choose $\ln(N)$ as optimal value of $minPts$ where N is the number of data [2]. Then, we compute the distances between each point and its k
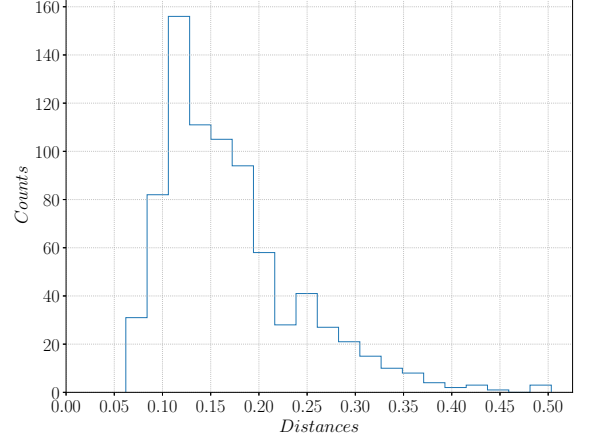
nearest neighbours, where k is equal to $minPts$. In Fig. 2 we plot an histogram of the distances: the greater value at which the tail of the histogram ends defines our ε parameter. Indeed, this value in the histogram represents a change in density distribution among points. Eventually we tune the obtained parameters by setting up a grid of values. Since the data labels are provided, the performance of DBSCAN is evaluated using the Normalized Mutual Information (NMI) that measures the fraction of correctly predicted labels [5].

We use t-SNE algorithm to represent the rescaled original data in a 2-dimensional *latent space* using the algorithm implementation of the *scikit-learn* library [7]. We test several values of the perplexity parameter Σ in order to compare results. Moreover, both the default random initialization and PCA initialization of the algorithm are tested.

Afterwards, we apply DBSCAN method to the dataset projected to the 2-dimensional *latent space* with t-SNE algorithm.

RESULTS

For DBSCAN method on the 5-dimensional dataset we set the value of the parameter $minPts= 7$ and $\varepsilon = 0.46$. We illustrate the results for a grid of the parameters $minPts$ and ε in Table I. Several combination of the parameters return a NMI of 1.00 and we choose $minPts= 7$ and $\varepsilon = 0.46$ as best parameters since they are the expected ones.

In Fig. 3a the results are plotted on the 2-dimensional dataset obtained with t-SNE for better visualization. We observe that the three clusters are thoroughly labeled, indeed we compute a value of NMI equal to 1.
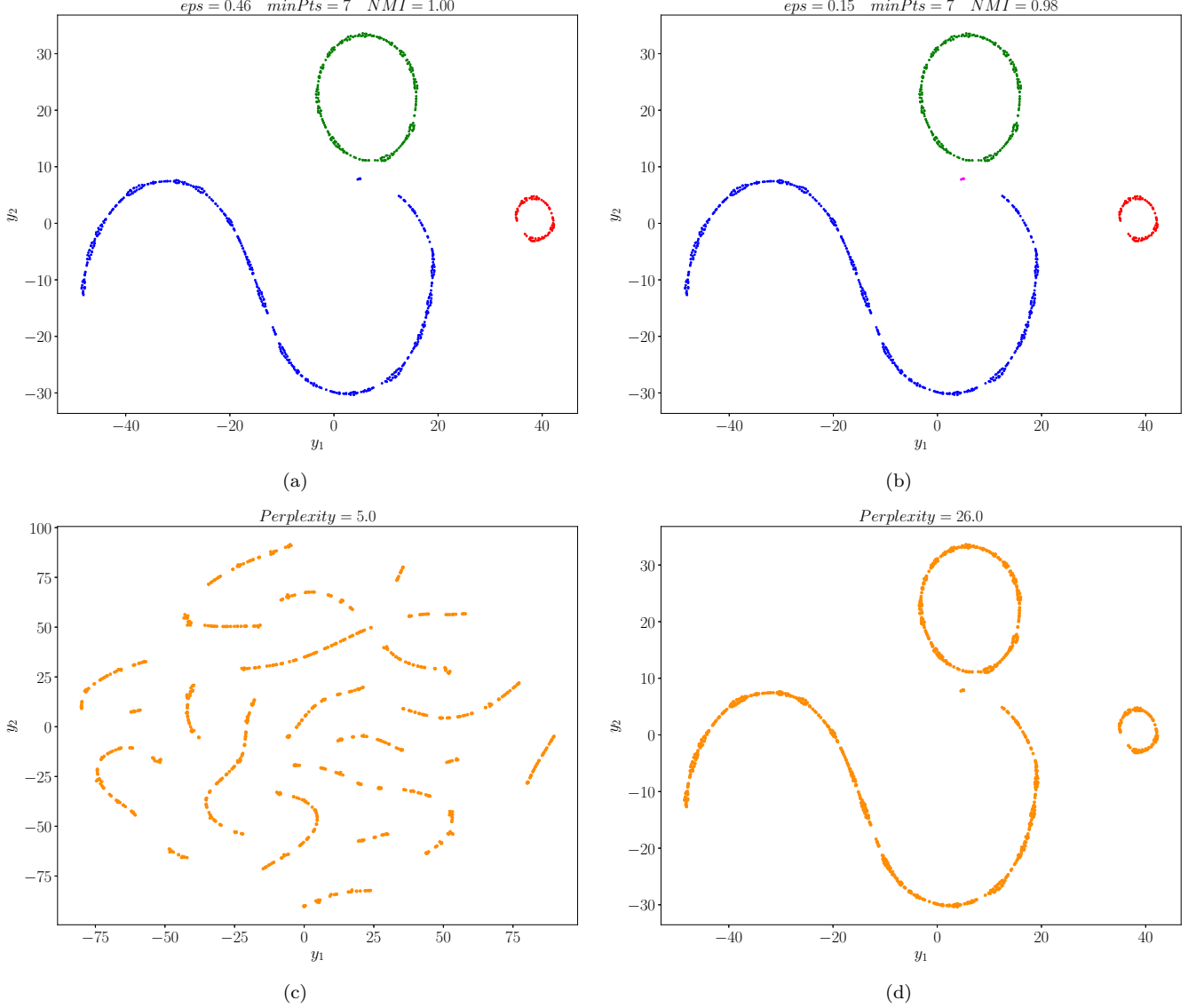
FIG. 3: In (a) and (b) we plot the clustered manifold obtained by DBSCAN method on the 5-dim and 2-dim dataset respectively; for better visualization both are represented on the 2-dimensional *latent space*. The three clusters are colored in blue, red and green, while outliers in magenta. In (c) and (d) we illustrate the dataset projected to the 2-dimensional *latent space* with t-SNE for perplexity $\Sigma = 5$ and $\Sigma = 26$ respectively.

$minPts = 7$									
ε	0.38	0.40	0.42	0.44	0.46	0.48	0.50	0.52	0.54
NMI	0.81	0.98	0.98	1.00	1.00	1.00	1.00	1.00	0.00

$\varepsilon = 0.46$									
$minPts$	3	4	5	6	7	8	9	10	11
NMI	1.00	1.00	1.00	1.00	1.00	1.00	0.81	0.72	0.70

TABLE I: NMI results for grid of $minPts$ and ε values for DBSCAN in 5-dim dataset.

Then, in Fig. 3c and Fig. 3d we visualize the dataset projected to the 2-dimensional *latent space* with t-SNE with random initialization for perplexity $\Sigma = 5, 26$ respectively. For $\Sigma = 5$ the algorithm is not able to project

the data in a meaningful way, while for $\Sigma = 26$ we can see that the structure of the three clusters is preserved. Moreover, we verify that PCA initialization does not produce sensibly different results.

In Fig. 3b we apply DBSCAN method on the 2-dimensional dataset. After the grid search we obtain that best performance are achieved for $minPts= 7$ and $\varepsilon = 0.15$. We observe that the three clusters are not thoroughly labeled, indeed we compute a value of NMI equal to 0.98. We note that t-SNE projects badly the overlapping part of two clusters labeling it as an outlier.

Eventually, we reproduce the same analysis on a more noisy dataset. We observe that DBSCAN method has

lower performances and require higher value of *minPts*. Moreover, the noise challenges t-SNE making it harder to discern the overlapping parts of the clusters.

CONCLUSIONS

In this work we show that DBSCAN on the 5-dimensional dataset with three embedded manifolds is able to discern clusters correctly. In particular, the method we choose for selecting *minPts* and ε parameters proves to be very effective. Moreover, we show that t-SNE is able to project meaningfully the dataset in a 2-dimensional *latent space* for proper value of the perplexity parameter. However, the loss of information in the projected *latent space* challenges DBSCAN making it misclassify some overlapping points as outliers. Eventually, we observe that with more noisy dataset performance is generally reduced.

REFERENCES

[1] Martin Ester et al. "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD'96. Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[2] Derya Birant and Alp Kut. "ST-DBSCAN: An algorithm for clustering spatial–temporal data". In: *Data Knowledge Engineering* 60.1 (2007). Intelligent Data Mining, pp. 208–221. ISSN: 0169-023X. DOI: `https : / / doi . org / 10 . 1016 / j . datak . 2006 . 01 . 013`. URL: `http : / / www . sciencedirect . com / science / article / pii / S0169023X06000218`.

[3] Laurens van der Maaten and Geoffrey Hinton. "Visualizing Data using t-SNE". In: *Journal of Machine Learning Research* 9 (2008), pp. 2579–2605. URL: `http : / / www . jmlr.org/papers/v9/vandermaaten08a.html`.

[4] Hans-Peter Kriegel et al. "Density-based clustering". In: *WIREs Data Mining and Knowledge Discovery* 1.3 (2011), pp. 231–240. DOI: `10 . 1002 / widm . 30`. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/ widm.30`. URL: `https : / / onlinelibrary . wiley . com / doi/abs/10.1002/widm.30`.

[5] Pankaj Mehta et al. "A high-bias, low-variance introduction to Machine Learning for physicists". In: *Physics Reports* 810 (May 2019), pp. 1–124. ISSN: 0370-1573. DOI: `10 . 1016 / j . physrep . 2019 . 03 . 001`. URL: `http://dx. doi.org/10.1016/j.physrep.2019.03.001`.

[6] *Scikit-learn DBSCAN reference page*. 2020. URL: `https: / / scikit – learn . org / stable / modules / generated / sklearn.cluster.DBSCAN.html`.

[7] *Scikit-learn t-SNE reference page*. 2020. URL: `https : / / scikit – learn . org / stable / modules / generated / sklearn.manifold.TSNE.html`.