Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study

Mebarka Allaoui[1](✉), Mohammed Lamine Kherfi[2,3], and Abdelhakim Cheriet[3]

[1] LAGE Laboratory, Kasdi Merbah University Ouargla, Ouargla, Algeria
moubarakaallaoui1994@gmail.com
[2] LAMIA Laboratory, Université du Québec à Trois-Rivières, Trois-Rivières, Canada
[3] Kasdi Merbah University Ouargla, Ouargla, Algeria

Abstract. Dimensionality reduction is widely used in machine learning and big data analytics since it helps to analyze and to visualize large, high-dimensional datasets. In particular, it can considerably help to perform tasks like data clustering and classification. Recently, embedding methods have emerged as a promising direction for improving clustering accuracy. They can preserve the local structure and simultaneously reveal the global structure of data, thereby reasonably improving clustering performance. In this paper, we investigate how to improve the performance of several clustering algorithms using one of the most successful embedding techniques: Uniform Manifold Approximation and Projection or UMAP. This technique has recently been proposed as a manifold learning technique for dimensionality reduction. It is based on Riemannian geometry and algebraic topology. Our main hypothesis is that UMAP would permit to find the best clusterable embedding manifold, and therefore, we applied it as a preprocessing step before performing clustering. We compare the results of many well-known clustering algorithms such ask-means, HDBSCAN, GMM and Agglomerative Hierarchical Clustering when they operate on the low-dimension feature space yielded by UMAP. A series of experiments on several image datasets demonstrate that the proposed method allows each of the clustering algorithms studied to improve its performance on each dataset considered. Based on Accuracy measure, the improvement can reach a remarkable rate of 60%.

Keywords: Dimensionality reduction · UMAP · Clustering · Embedding manifold · Big data analytics · Machine learning · Comparative study

1 Introduction

Clustering is a fundamental pillar of unsupervised machine learning and it is widely used in a range of tasks across disciplines. In past decades, a variety

A. El Moataz et al. (Eds.): ICISP 2020, LNCS 12119, pp. 317–325, 2020.
https://doi.org/10.1007/978-3-030-51935-3_34

of clustering algorithms have been developed [5] such as k-means [6], Gaussian Mixture Models (GMMs) [14], HDBSCAN [1], and hierarchical algorithms [15]. However, these clustering algorithms typically require features to be hand crafted or learned for each dataset and task. Then, those features should be analyzed using feature selection, in order to eliminate redundant or poor quality features. Those requirements are more challenging in the unsupervised setting. Additionally, this process is time-consuming and brittle [17], since the choice of features has a large influence on the performance of the clustering algorithm. In this paper, we formulate the following hypothesis: if we apply an adequate embedding on our raw data, i.e., an embedding which allows to find a good distance preserving manifold, than this could help clustering algorithms in doing their job. One key question was: which embedding technique to apply it to find the best embedding manifold. Many methods exists, including those performing a linear transformation of data like the well-known Principal Component Analysis (PCA) [10]. However, PCA is a linear method and does not perform well in cases where relationships are non-linear. Thankfully, alternative non-linear manifold learning methods exist, and can be categorized by their focus on finding local or global structure. Isomap [7] is well known globally focused method. While T-SNE [8] is considered as locally focused method. More recently manifold learning technique is UMAP [11], UMAP showed better performance to preserve both the local and global structure. In this paper, we will investigate the use of this latter technique: because it outperforms its concurrents [11] and it has proven to be able to exactly meet our needs [16,18]. In this paper, Our main focus was on measuring the improvement achieved by each clustering algorithm thanks to the application of UMAP embedding manifold, and in order to validate our method we conduct a number of experiments on five datasets. We empirically observe that this method allows to the clustering algorithms to be competitive with state-of-the-art techniques. The rest of this paper is organized as follows. We present more details about UMAP technique in Sect. 2. In Sect. 3 we introduce our idea. Section 4 discusses the experimental results in five image datasets. Section 5 concludes our work.

2 UMAP Embedding Technique for Dimensionality Reduction

Uniform Manifold Approximation and Projection (UMAP) is a recently proposed manifold learning method, which seeks to accurately represent local structure and better incorporate global structure [9]. Compared to t-SNE it has a number of advantages. UMAP has been shown to scale well with large datasets, while t-SNE typically struggles with them. UMAP relies on three hypothesis, namely that 1) the data is uniformly distributed on a Riemannian manifold, 2) the Riemannian metric is locally constant 3) the manifold is locally connected. From these assumptions it is possible to represent the manifold with a fuzzy topological structure of high dimensional data points. The embedding manifold is found by searching for a fuzzy topological structure of low dimensional projection of the data. To construct the fuzzy topological structure UMAP represents the data points by a high-dimensional graph. The constructed high-dimensional graph is weighted graph, with edge weights representing the likelihood that two points are connected. UMAP uses exponential probability distribution to compute the similarity between high dimensional data points:

$$p_{i|j} = \exp(-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}) \tag{1}$$

Where d(x_i,x_j) is the distance between the i-th and j-th data points and ρ is the distance between i-th data points and its first nearest neighbor. In cases that the weight of the graph between i and j nodes is not equal to the weight between j and i nodes. UMAP uses a symmetrization of the high-dimensional probability:

$$p_{ij} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i} \tag{2}$$

As we said above the constructed graph is a likelihood graph, and UMAP needs to specify k the number of nearest neighbor:

$$k = 2^{\sum_i p_{ij}} \tag{3}$$

Once the high-dimensional graph is constructed, UMAP constructs and optimizes the layout of a low-dimensional analogue to be as similar as possible. For modelling distance in low dimensions, UMAP uses probability measure similar to Student t-distribution:

$$q_{ij} = (1 + a(y_i - y_j)^{2b})^{-1} \tag{4}$$

where $a \approx 1.93$ and $b \approx 0.79$ for default UMAP.

UMAP uses binary cross-entropy (CE) as a cost function due to its capability of capturing the global data structure:

$$CE(P, Q) = \sum_i \sum_j [p_{ij} \log(\frac{p_{ij}}{q_{ij}}) + (1 - p_{ij}) \log(\frac{1 - p_{ij}}{1 - q_{ij}})] \tag{5}$$

Where P is the probabilistic similarity of the high dimensional data points, and Q is for the low dimensional data points.

The derivative of the cross-entropy used to update the coordination of the low-dimensional data points to optimize the projection space until the convergence. UMAP applied Stochastic Gradient Descent (SGD) due to its faster convergence and it reduces the memory consumption since we compute the gradients for a subset of the data set.

UMAP has a number of important hyper-parameters that influence its performance. These hyper-parameters are:

- The dimensionality of the target embedding
- The number of neighbor k, choosing small value means the interpretation will be very local and capture fine detail structure. While choosing a large value means the estimation will be based on larger regions, and thus, will missing some of the fine detail structure.

- The minimum allowed distance between points in the embedding space. Lower values of this minimum distance will more accurately capture the true manifold structure, but may lead to dense clouds that make visualization difficult.

3 Our Method

Our method relies primarily on the application of clustering algorithms on embedding manifold extracted by manifold learning methods UMAP [9] due to its success in preserving both the local and the global structure. We chose four of well-known algorithms as clustering algorithms which are represented in k-means [6], HDBSCAN [1], GMM [14] and Agglomerative Clustering [15]. We will show that by augmenting the clustering task with a manifold learning technique which explicitly takes local structure into account, we can increase the quality of clustering performance of the different algorithms. Figure 1 represents the architecture of our method.

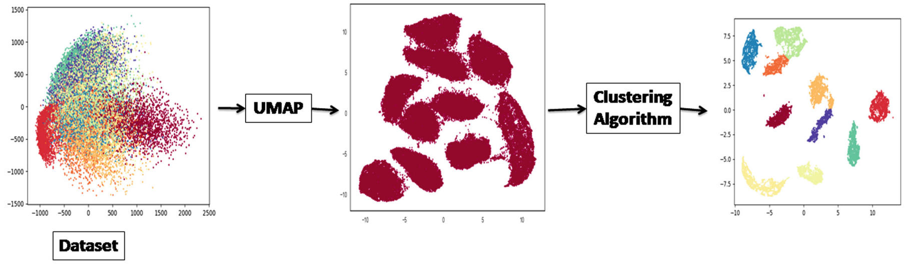

Fig. 1. The structure of our method.

4 Experiments

To assess the improvement of using UMAP with the clustering algorithms studied, we conduct experiments on a range of diverse datasets, including standard datasets widely used to evaluate clustering algorithms.

4.1 Datasets

We conducted our experiments on five diverse image datasets, including standard datasets used to evaluate deep clustering algorithms. Those datasets are MNIST [2], Fashion MNIST [3], USPS [13], Pen Digits [4] and UMIST Face Cropped [12]. Table 1 summarizes the main characteristics of each dataset.

Table 1. Datasets statistics.

Dataset	Number of images	Number classes	Feature vector length
MNIST	20000	10	784
F-MNIST	20000	10	784
USPS	9298	10	256
Pen digits	1797	10	64
UMIST face	575	20	10304

4.2 Evaluation Metrics

In order to validate the performance of unsupervised clustering algorithms, we use the two standard evaluation metrics, accuracy (ACC) and Normalized Mutual Information (NMI).

$$ACC = max_m \frac{\sum_{i=1}^{n} 1\{y_i = m(c_i)\}}{n} \tag{6}$$

$$NMI = \frac{2I(y,c)}{[H(y) + H(c)]} \tag{7}$$

4.3 Results

Figure 2 shows the resulting clusters when using k-means for visualization purposes. We could see that the visualization is better when we apply the algorithm on the UMAP embedded manifold of the five datasets. However, in order to better understand the effectiveness of our method at clustering we will study each clustering algorithm via measuring its own results on the different datasets using the accuracy and NMI, as well as when we apply it on the extracted features by UMAP.

Table 2 and Table 3 show the accuracy and NMI results for the clustering algorithms on five different datasets comparable to the same algorithms applied on embedding manifold of the datasets extracted by UMAP. In both tables, improvement score rows represent the difference between the results of the algorithms and the results after the application of these algorithms on the features extracted by UMAP. By doing so, we can see clearly how UMAP can help the four clustering algorithms and to what extent the results improved. Actually, great results were achieved by the algorithms on embedded data points, where the results are improved by an increase of up to 60% in term of accuracy, and in range of 5% to 48% in term of NMI. What is striking is how UMAP helped HDBSCAN to improve its result by 60 % points on USPS dataset. Also, it had an improvement better than the other algorithms in 2 of the 5 datasets with at least 50% in term of accuracy and over than 38% in term of NMI measure. GMM is improved better than the other, on 3 of the 5 datasets, with percentage over than 34% in term of accuracy, and over than 25% in term NMI measure.

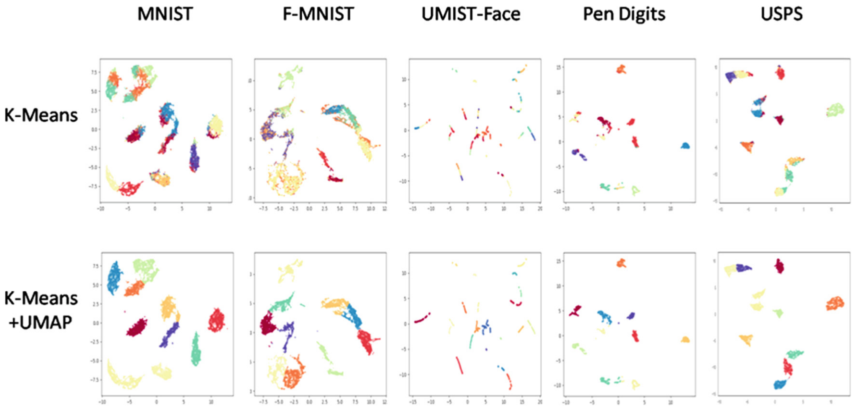

Fig. 2. Visualization of K-Means applied to all five datasets. The first row represents the K-Means visualization of the five datasets themselves, and the second row represents the visualization of K-Means on the UMAP embedded manifold of these datasets.

Table 2. Comparison between the different clustering algorithms on the five datasets according to the accuracy measure.

	MNIST	F-MNIST	UMIST face	Pen digits	USPS
K-means	0.5278	0.4750	0.4348	0.7028	0.6678
UMAP + K-means	**0.9054**	**0.5865**	**0.7409**	**0.8843**	**0.8105**
Improvement score	0.3776	0.1115	0.3061	0.1815	0.1427
Agglomerative	0.5751	0.5766	0.4539	0.7451	0.6834
UMAP + Agglomerative	**0.8918**	**0.5925**	**0.7270**	**0.8737**	**0.9584**
Improvement score	0.3167	0.0159	0.2731	0.1286	0.2740
HDBSCAN	0.2765	0.2140	0.4904	0.5453	0.3529
UMAP + HDBSCAN	**0.7765**	**0.3458**	**0.6730**	**0.9004**	**0.9553**
Improvement score	0.5000	0.1318	0.1826	0.3551	0.6024
GMM	0.4507	0.4579	0.3826	0.4836	0.4802
UMAP+GMM	**0.9159**	**0.5885**	**0.7287**	**0.8748**	**0.6727**
Improvement score	0.4652	0.1306	0.3461	0.3912	0.1925

The accuracy and NMI measures showed us that the studied clustering algorithms in general and HDBSCAN as a particular case had bad results and especially in MNIST and Fashion MNIST datasets. The problem here is all the clustering algorithms tend to suffer from the curse of dimensionality: high dimensional data requires more observed samples to produce much density. If we could reduce the dimensionality of the data more we would make the density more evident and make it far easier for those algorithms to cluster the data. What we need is strong manifold learning, and this is where UMAP can come into play. One of the reasons which help the studied algorithms to perform well on

Table 3. Comparison between the different clustering algorithms on the five datasets according to the NMI measure

	MNIST	F-MNIST	UMIST face	Pen digits	USPS
K-means	0.4774	0.5139	0.6647	0.6998	0.6266
UMAP + K-means	**0.8494**	**0.6377**	**0.8663**	**0.8545**	**0.8602**
Improvement score	0.3720	0.1238	0.2016	0.1547	0.2336
Agglomerative	0.6360	0.6080	0.6673	0.7965	0.7250
UMAP + Agglomerative	**0.8463**	**0.6511**	**0.8764**	**0.8456**	**0.9000**
Improvement score	0.2103	0.0431	0.2091	0.0491	0.1750
HDBSCAN	0.3674	0.2535	0.6933	0.5804	0.4442
UMAP + HDBSCAN	**0.8315**	**0.6323**	**0.8427**	**0.8871**	**0.8923**
Improvement score	0.4641	0.3788	0.1494	0.3067	0.4481
GMM	0.3882	0.5471	0.6160	0.5203	0.4232
UMAP+GMM	**0.8654**	**0.6424**	**0.8648**	**0.8447**	**0.8231**
Improvement score	0.4772	0.0953	0.2488	0.3244	0.3999

the learned manifold is to set the min distance (the hyper-parameter of UMAP) to be 0. And thus make the points packed together densely as well as making cleaner separations between clusters.

Table 4. The execution time before and after applying UMAP on the different clustering algorithms on the five datasets.

Time in second	MNIST	F-MNIST	UMIST face	Pen digits	USPS
K-means	112.13	74.69	17.24	0.94	12.93
UMAP + K-means	**1.22**	**1.20**	**0.33**	**0.26**	**0.57**
Agglomerative	710.08	674.14	6.57	0.48	47.93
UMAP + Agglomerative	**88.31**	**100.14**	**0.03**	**0.28**	**8.51**
HDBSCAN	1603.26	1660.25	17.77	1.14	117.56
UMAP + HDBSCAN	**5.13**	**4.49**	**0.03**	**0.12**	**0.75**
GMM	24.51	26.27	3.49	0.58	25.26
UMAP+GMM	**0.51**	**0.42**	**0.06**	**0.03**	**0.14**

Table 4 gives us the execution time taken for each clustering algorithm on the different datasets compared to the run-time of these algorithms applied to the embedding manifold of the five datasets. We can observe that the run-time is also improved, where it was reduced to a few seconds and sometimes to a few split-seconds, and this is really a good achievement for our method compared to the size of the datasets. Especially for agglomerative and HDBSCAN algorithms, the run-time of HDBSCAN is reduced from over than 26 min until around 5 s in

MNIST and Fashion MNIST datasets. From these results, we demonstrate that these clustering algorithms can now handle large databases well.

5 Conclusion

In this paper, we investigated the use of UMAP technique for dimensionality reduction before applying a number of well-known clustering algorithms on datasets. We showed that it can drastically improve the performance of the studied algorithms, both in terms of clustering accuracy and time. Experimental results indicate that the proposed approach can improve clustering performance obviously, we show how our proposed method can make the mentioned clustering algorithms competitive with the current state-of-the-art clustering approaches. It is also validated by experiments that our method allows to the clustering algorithms considered to deal better on larger data sets.

References

1. Campello, R.J.G.B., Moulavi, D., Sander, J.: Density-based clustering based on hierarchical density estimates. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013. LNCS (LNAI), vol. 7819, pp. 160–172. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37456-2_14
2. Deng, L.: The MNIST database of handwritten digit images for machine learning research [best of the web]. IEEE Sig. Process. Mag. **29**(6), 141–142 (2012)
3. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
4. Alpaydin, E., Alimoglu, F.: Pen-based recognition of handwritten digits data set. University of California, Irvine. Machine Learning Repository. Irvine: University of California, **4**(2) (1998)
5. Xu, D., Tian, Y.: A comprehensive survey of clustering algorithms. Annals of Data Science **2**(2), 165–193 (2015). https://doi.org/10.1007/s40745-015-0040-1
6. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297. University of California Press, Berkeley (1967)
7. Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)
8. Maaten, L.V.D., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
9. McInnes, L., Healy, J., Melville, J.: Umap: uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)
10. Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space. Lond. Edinb. Dublin Philos. Mag. J. Sci. **2**(11), 559–572 (1901)
11. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
12. Graham, D.B., Allinson, N.M.: Characterising virtual eigen signatures for general purpose face recognition. In: Wechsler, H., Phillips, P.J., Bruce, V., Soulié, F.F., Huang, T.S. (eds.) Face Recognition. NATO ASI Series (Series F: Computer and Systems Sciences), vol. 163, pp. 446–456. Springer, Heidelberg (1998). https://doi.org/10.1007/978-3-642-72201-1-25

13. Hull, J.J.: A database for handwritten text recognition research. IEEE Trans. Pattern Anal. Mach. Intell. **16**(5), 550–554 (1994). https://doi.org/10.1109/34.291440
14. Rasmussen, C.E.: The infinite Gaussian mixture model. In: NIPS 1999 Proceedings of the 12th International Conference on Neural Information Processing Systems, pp. 554–560. MIT Press, Cambridge (2000)
15. Madhulatha, T.S.: An overview on clustering methods. J. Eng. **2**(4), 719–725 (2012)
16. McConville, R., Santos-Rodriguez, R., Piechocki, R.J., Craddock, I.: N2D: (Not Too) deep clustering via clustering the local manifold of an auto encoded embedding. arXiv preprint arXiv:1908.05968 (2019)
17. Miao, J., Niu, L.: A survey on feature selection. Procedia Comput. Sci. **91**, 919–926 (2016)
18. Becht, E., et al.: Dimensionality reduction for visualizing single-cell data using UMAP. Nat. Biotechnol. **37**(1), 38 (2019)