Updatable Clustering Via Patches

Anshuman Chhabra , Ashwin Sekhari, and Prasant Mohapatra
Department of Computer Science
University of California
Davis, CA 95616
{chhabra,asekhari,pmohapatra}@ucdavis.edu

1 Introduction

Clustering algorithms are widely employed in machine learning applications, either as a precursor to supervised learning to generate pseudo-labels for training [1, 2] or to group similar samples together in the absence of labeled data [3–5]. A large body of work on clustering approaches aims to add constraints to change the clustering output to accommodate these constraints while minimizing the clustering loss [6–8]. A large majority of these "constrained clustering" algorithms constitute in-processing approaches [9–11], where the algorithm itself has to be changed according to the desired constraints to *update* the clustering output. In particular, constraints can vary widely, and there are many different clustering algorithms. Thus, there are a combinatorial number of "possible" constrained clustering algorithms, and algorithm designers would have to come up with analysis and theoretical guarantees for each of these individually, making it a challenging problem.

Instead, we propose the use of *patches*, which are essentially a small number of "new" data samples that are added to the original dataset to ensure that the clustering output achieves the desired constraints. Our work is in part inspired by the use of *antidote data* [12, 13], where additional data points are used to improve fairness metrics for the machine learning models. However, in our work, we aim to generalize this to any metric defined over the clustering output, through the use of data *patches*.

In this paper, we consider *performance* metrics defined over the clustering output, and utilize zeroth order approaches to optimize the data patch problem, to improve upon these metrics. We consider commonly used clustering metrics in the literature such as Normalized Mutual Information (NMI) [14], Adjusted Rand Index (ARI) [15], and Silhouette Score [16]. The NMI and ARI metrics seek to measure how densely and *accurately* the cluster outputs are with respect to some known ground truth. The Silhouette Score on the other hand, is truly unsupervised, and measures the quality of the clustering output under the assumption that ground truth labels are not available.

The rest of the paper is structured as follows: Section 2 discusses related work, Section 3 introduces the problem statement and the proposed approach for solving the *data patch* problem, Section 4 showcases preliminary experiments and results on real data, and Section 5 concludes the paper.

2 Related Work

No prior work inherently studies "updatable" clustering but the closest work to ours uses *antidote* data to improve the fairness of clustering models [12] or recommendation systems [13] using newly introduced data points. However, while the motivation is similar, the algorithms in these papers are specifically designed to improve upon fairness/polarization/bias metrics, and do not work for any general metric that operates on the clustering output. Through this work, our main goal is to show that simple algorithmic solutions exist to the *data patches* problem, and show that these can be a viable solution to updatable clustering.

UpML Workshop, ICML 2022.

3 Problem Statement

Let the original dataset be denoted as D, and the data patches are denoted as P. Now, let a clustering algorithm be defined as A, which operates on some data samples D, and outputs a clustering output O consisting of labels for each sample in D. That is, $A(D) = O$. Now, let θ be a clustering metric defined on the clustering output obtained as $\theta(O)$ which outputs a scalar value, where a higher value indicates a more viable solution. We would like to maximize θ for our original dataset D, by combining the *data patches* P (i.e., $D \cup P$) with our original dataset, and obtain an extended clustering label set O' (i.e., $A(D \cup P) = O'$). Basically, we seek to improve the performance with respect to the original dataset D by maximizing $\theta(\hat{O}')$, where $\hat{O}'$ are the first $|O|$ labels of O'.

The optimization problem can then be written as follows:

$$\begin{aligned} \max_{P} \quad & \theta(\hat{O}') \\ \text{s.t.} \quad & \hat{O}' = O'_{:|O|} \\ & O' = A(D \cup P) \end{aligned} \tag{P.OPT}$$

For experiments in the paper, we employ three different *clustering performance* metrics θ that operate on the clustering output: NMI, ARI, and Silhouette Score (s). These definitions indicate how "good" the outputted clustering is and hence, we aim to improve the original clustering on the dataset D (according to these metrics) by injecting the patches P in the dataset. The analytical definitions for these performance metrics are given below:

$$\theta_{\text{NMI}} = \frac{\sum_{i,j} n_{ij} log \frac{n.n_{ij}}{n_{i+}.n_{+j}}}{\sqrt{(\sum_i n_{i+} log \frac{n_{i+}}{n})(\sum_j n_{+j} log \frac{n_{+j}}{n})}} \tag{NMI}$$

Here, n_{ij} , n_{i+} and nq_{+j} represent the co-occurrence number and cluster size of $i-$th and $j-$th clusters in the obtained partition and ground truth, respectively, and n is the total data instance number.

$$\theta_{\text{ARI}} = \frac{\phi - \mathbb{E}[\phi]}{\max\{\phi - \mathbb{E}[\phi]\}} \tag{ARI}$$

Here, ϕ is the Rand Index defined as $\phi = \frac{a+b}{C_2^n}$ where C_2^n denotes the total possible pairs in the dataset, a denotes the number of pairs of samples in both the predicted clusters and the ground truth clusters, and b denotes the pairs of samples that are in different sets of predicted clusters and in different sets of the ground truth clusters.

$$\theta_s = \frac{b - a}{\max(a, b)} \tag{Silhouette Score}$$

Here, a is the mean distance between a sample and all other points in the same class and b is The mean distance between a sample and all other points in the next nearest cluster.

4 Results

We solve the aforementioned optimization problem P.OPT by using zeroth order optimizers (namely, [17]), as in the general case, we cannot obtain first-order (gradient) information from the clustering output. In our experiments we consider the UCI DIGITS dataset [18] and MNIST [19] dataset, which we subsample to allow for faster clustering performance.

We first show some of our data *patches* visually in comparison to the original input datasets in Figure 1. While clearly out of the input distribution, these samples are able to augment the performance of the dataset on the original dataset as we will show below. We utilize the k-means [20] clustering algorithm in all our experiments.

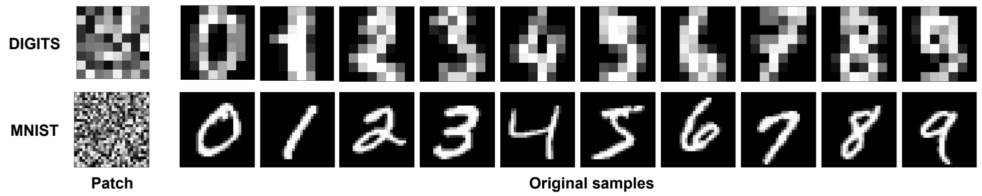
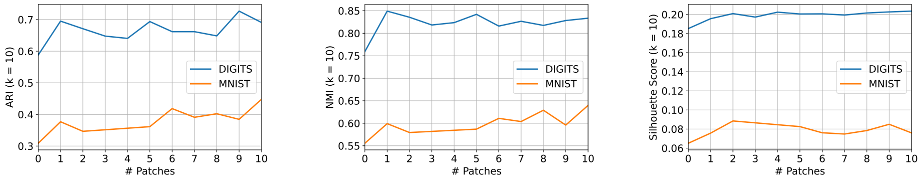


Figure 1: Patches generated by our approach.



(a) ARI

(b) NMI

(c) Silhouette Score

Figure 2: Plots showcasing the effect of increasing the number of patches on the clustering performance.

We observe that even by adding a single patch the performance (ARI/NMI/Silhoutte Score) of the original clustering algorithm can be significantly improved. In general, we observe performance improvements up-to 45% (ARI, Table 1) just by adding $< 10\%$ the number of samples in the dataset as patches.

	DIGITS	MNIST
NMI	12	15.13
ARI	23.62	44.98

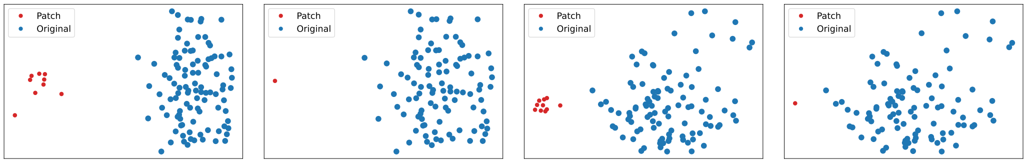
Table 1: Maximum percentage increase in performance as a result of adding patches.

The general trend as observed in Figure 2 is that by adding more patches, we can increase performance much more. However, note that it is generally better to add only a few additional patches to the dataset, as this increases both the computational requirements of the clustering task, as well as the compute required to obtain the patch samples via the zeroth order optimizer.

We also use Principal Component Analysis [21] to visualize the first 2 principal components of both the patches and the original dataset. These are shown in Figure 3. Quite evidently, the patch samples are significantly different from the input data distribution, as also visually observed in the images shown in Figure 1.

5 Conclusion

In this paper, we propose a simple approach to updatable constrained clustering by using data augmentation in the form of *patches*. We utilize zeroth order optimizers, to solve our formulated data patch problem and generate patches to improve the performance of the output clustering. In particular, we observe that adding even a single patch to the dataset can lead to significant performance improvements, paving the way for future work that improves upon the 1) zeroth order optimization algorithms, 2) applies patches to other non-trivial clustering metrics, and 3) provides theoretical guarantees for patch generation.



(a) DIGITS ($|P| = 9$)

(b) DIGITS ($|P| = 1$)

(c) MNIST ($|P| = 10$)

(d) MNIST ($|P| = 1$)

Figure 3: PCA components of the patches and the original dataset.

References

[1] Hao Wu and Saurabh Prasad. Semi-supervised deep learning using pseudo labels for hyperspectral image classification. *IEEE Transactions on Image Processing*, 27(3):1259–1270, 2017.

[2] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11557–11568, 2021.

[3] Nizar Grira, Michel Crucianu, and Nozha Boujemaa. Unsupervised and semi-supervised clustering: a brief survey. *A review of machine learning techniques for processing multimedia content*, 1:9–16, 2004.

[4] Kristina P Sinaga and Miin-Shen Yang. Unsupervised k-means clustering algorithm. *IEEE access*, 8:80716–80727, 2020.

[5] Isak Gath and Amir B. Geva. Unsupervised optimal fuzzy clustering. *IEEE Transactions on pattern analysis and machine intelligence*, 11(7):773–780, 1989.

[6] Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. Contrastive clustering. In *2021 AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[7] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

[8] Sugato Basu, Ian Davidson, and Kiri Wagstaff. *Constrained clustering: Advances in algorithms, theory, and applications*. CRC Press, 2008.

[9] Paul S Bradley, Kristin P Bennett, and Ayhan Demiriz. Constrained k-means clustering. *Microsoft Research, Redmond*, 20(0):0, 2000.

[10] Ian Davidson and SS Ravi. Clustering with constraints: Feasibility issues and the k-means algorithm. In *Proceedings of the 2005 SIAM international conference on data mining*, pages 138–149. SIAM, 2005.

[11] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. Constrained k-means clustering with background knowledge. In *Icml*, volume 1, pages 577–584, 2001.

[12] Anshuman Chhabra, Adish Singla, and Prasant Mohapatra. Fair clustering using antidote data. In *Algorithmic Fairness through the Lens of Causality and Robustness workshop*, pages 19–39. PMLR, 2022.

[13] Bashir Rastegarpanah, Krishna P Gummadi, and Mark Crovella. Fighting fire with fire: Using antidote data to improve polarization and fairness of recommender systems. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 231–239, 2019.

[14] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, 2002.

[15] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

[16] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[17] Yang Yu, Hong Qian, and Yi-Qi Hu. Derivative-free optimization via classification. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[18] Lei Xu, Adam Krzyzak, and Ching Y Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE transactions on systems, man, and cybernetics*, 22(3):418–435, 1992.

[19] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[20] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[21] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.