FUZZY C-MEANS CLUSTERING FOR PERSISTENCE DIAGRAMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Persistence diagrams concisely represent the topology of a point cloud whilst having strong theoretical guarantees. Most current approaches to integrating topological information into machine learning implicitly map persistence diagrams to a Hilbert space, resulting in deformation of the underlying metric structure whilst also generally requiring prior knowledge about the true topology of the space. In this paper we give an algorithm for Fuzzy c-Means (FCM) clustering directly on the space of persistence diagrams, enabling unsupervised learning that automatically captures the topological structure of data, with no prior knowledge or additional processing of persistence diagrams. We prove the same convergence guarantees as traditional FCM clustering: every convergent subsequence of iterates tends to a local minimum or saddle point. We end by presenting experiments where our fuzzy topological clustering algorithm allows for unsupervised top-k candidate selection in settings where (i) the properties of persistence diagrams make them the natural choice over geometric equivalents, and (ii) the probabilistic membership values let us rank candidates in settings where verifying candidate suitability is expensive: lattice structure classification in materials science and pre-trained model selection in machine learning.

1 INTRODUCTION

Persistence diagrams, a concise representation of the topology of a point cloud with strong theoretical guarantees, have emerged as a new tool in the field of data analysis (Edelsbrunner & Harer, 2010). Persistence diagrams have been successfully used to analyse problems ranging from financial crashes (Gidea & Katz, 2018) to protein binding (Kovacev-Nikolic et al., 2014), but the non-Hilbertian nature of the space of persistence diagrams means it is difficult to directly use persistence diagrams for machine learning. In order to better integrate diagrams into machine learning workflows, efforts have been made to map them into a more manageable form; primarily through embeddings into finite feature vectors, functional summaries, or by defining a positive-definite kernel on diagram space. In all cases, this explicitly or implicitly embeds diagrams into a Hilbert space which deforms the metric structure, potentially losing important information. Techniques to integrate these persistence-based summaries as topological regularisers and loss functions currently require prior knowledge about the correct topology of the dataset, which is clearly not feasible in most scenarios.

Against this background, we give an algorithm to perform Fuzzy c-Means (FCM) clustering (Bezdek, 1980) *directly* on collections of persistence diagrams, giving an important unsupervised learning algorithm and enabling learning from persistence diagrams without deforming the metric structure. We perform the convergence analysis for our algorithm, giving the same guarantees as traditional FCM clustering: that every convergent subsequence of iterates tends to a local minimum or saddle point. We demonstrate the value of our fuzzy clustering algorithm by using it to cluster datasets that benefit from both the topological and fuzzy nature of our algorithm. We apply our technique in two settings: lattice structures in materials science and the decision boundaries of CNNs. A key property for machine learning in materials science has been identified as "invariance to the basis symmetries of physics... rotation, reflection, translation" (Schmidt et al., 2019). Geometric clustering algorithms do not have this invariance, but persistence diagrams do, making them ideally suited for this application; we can cluster transformed lattice structure datasets where geometric equivalents fail. In addition to this, our probabilistic membership values allow us to rank the top-k most likely lattices assigned to a cluster. This is particularly important in materials science, as further investigation

requires expensive laboratory time and expertise. Our second application is inspired by Ramamurthy et al. (2019), who show that models perform better on tasks if they have topologically similar decision boundaries. We use our algorithm to cluster models and tasks by the persistence diagrams of their decision boundaries. Not only is our algorithm able to successfully cluster models to the correct task, based just on the topology of its decision boundary, but we show that higher membership values imply better performance on unseen tasks.

1.1 Related work

Means of persistence diagrams. Our work relies on the existence of statistics in the space of persistence diagrams. Mileyko et al. (2011) first showed that means and expectations are well-defined in the space of persistence diagrams. Specifically, they showed that the Fréchet mean, an extension of means onto metric spaces, is well-defined under weak assumptions on the space of persistence diagrams. Turner et al. (2012) then developed an algorithm to compute the Fréchet mean. We adapt Turner et al.'s algorithm to the weighted case, but the combinatoric nature of their algorithm makes it computationally intense. Lacombe et al. (2018) framed the computation of means and barycentres in the space of persistence diagram as an optimal transport problem, allowing them to use the Sinkhorn algorithm (Cuturi & Doucet, 2014) for fast computation of approximate solutions. The vectorisation of the diagram required by Lacombe et al.'s algorithm makes it unsuitable for integration into our work, as we remain in the space of persistence diagrams. Techniques to speed up the matching problem fundamental to our computation have also been proposed by Vidal et al. (2020) and Kerber et al. (2017).

Learning with persistence-based summaries. Integrating diagrams into machine learning workflows remained challenging even with well-defined means, as the space is non-Hilbertian (Turner & Spreemann, 2019). As such, efforts have been made to map diagrams into a Hilbert space; primarily either by embedding into finite feature vectors (Kališnik, 2018; Fabio & Ferri, 2015; Chepushtanova et al., 2015) or functional summaries (Bubenik, 2015; Rieck et al., 2019), or by defining a positive-definite kernel on diagram space (Reininghaus et al., 2015; Carrière et al., 2017; Le & Yamada, 2018). These vectorisations have been integrated into deep learning either by learning parameters for the embedding (Hofer et al., 2017; Carrière et al., 2020; Kim et al., 2020; Zhao & Wang, 2019), or as part of a topological loss or regulariser (Chen et al., 2018; Gabrielsson et al., 2020; Clough et al., 2020; Moor et al., 2019). However, the embeddings used in these techniques deform the metric structure of persistence diagram space (Bubenik & Wagner, 2019; Wagner, 2019; Carrière & Bauer, 2019), potentially leading to the loss of important information. Furthermore, these techniques generally require prior knowledge of a 'correct' target topology which can't plausibly be known in most scenarios. In comparison, our algorithm acts in the space of persistence diagrams so does not deform the structure of diagram space via embeddings, and is entirely unsupervised, requiring no prior knowledge about the topology.

Hard clustering. Maroulas et al. (2017) gave an algorithm for hard clustering persistence diagrams based on Turner et al.'s algorithm. Lacombe et al. (2018) gave an alternate implementation based on their algorithm for barycentre computation. The primary advantages of our work over previous work on hard clustering are as follows.

(i) The probabilistic membership values allow us to rank datasets in the cluster, enabling top-k candidate selection in settings where verifying correctness is expensive. The value provided by this fuzzy information is demonstrated in the experiments.

(ii) The fuzzy membership values provide information about proximity to all clusters, whereas hard labelling loses most of that information. In our experiments we demonstrate that this additional information can be utilised in practice.

(iii) The weighted cost function makes the convergence analysis (which we provide) entirely non-trivial in comparison to the non-fuzzy case. We consider this convergence analysis a primary contribution of our paper.

(iv) Fuzzy membership values have been shown to be more robust to noise than discrete labels (Klawonn, 2004).

(v) Unlike hard clustering, fuzzy clustering is analytically differentiable, allowing integration of the fuzzy clustering step into deep learning methods (Wilder et al., 2019).

Geometric equivalents. The most similar unsupervised learning technique to our algorithm is Wasserstein Barycentre Clustering (WBC). It clusters datasets of point clouds by the Wasserstein distance between the point clouds, rather than the Wasserstein distance between their persistence diagrams. We compare our algorithm experimentally to WBC using ADMM (Ye & Li, 2014), Bregman ADMM (Ye et al., 2017), Subgradient Descent (Cuturi & Doucet, 2014), Iterative Bregman Projection (Benamou et al., 2015), and full linear programming (Li & Wang, 2008). Each of these algorithms computes or approximates the Wasserstein barycentre in different ways. Theoretically, fuzzy discrete distribution clustering (d. A. T. de Carvalho et al., 2015) is similar to our algorithm, but the addition of the diagonal in the persistence diagram makes our work distinct.

1.2 Our contributions

1. Our main contribution is an algorithm for Fuzzy c-Means clustering of persistence diagrams, along with the convergence analysis. Given a collection of persistence diagrams $\mathbb{D}_1, \ldots, \mathbb{D}_n$, we alternatively calculate cluster centres $\mathbb{M}_1, \ldots, \mathbb{M}_c$ and membership values $r_{jk} \in [0, 1]$ which denote the degree to which diagram $\mathbb{D}_j$ is associated with cluster $\mathbb{M}_k$. We prove Theorem 1, showing that every convergent subsequence of these alternative update steps tends to a local minimum or saddle point of the cost function. This is the same convergence guarantee provided by traditional FCM clustering (Bezdek et al., 1987).

2. Updating the cluster centres requires computing the weighted Fréchet mean. We extend the algorithm given by Turner et al. (2012) to the weighted case, justifying our addition of weights by extending their proof to show that the updated algorithm converges.

3. We implement our algorithm in Python, available in the supplementary materials. It works with persistence diagrams from commonly used open-source libraries for Topological Data Analysis (TDA),[1] so is available for easy integration into current workflows, offering a powerful unsupervised learning algorithm to data science practitioners using TDA.

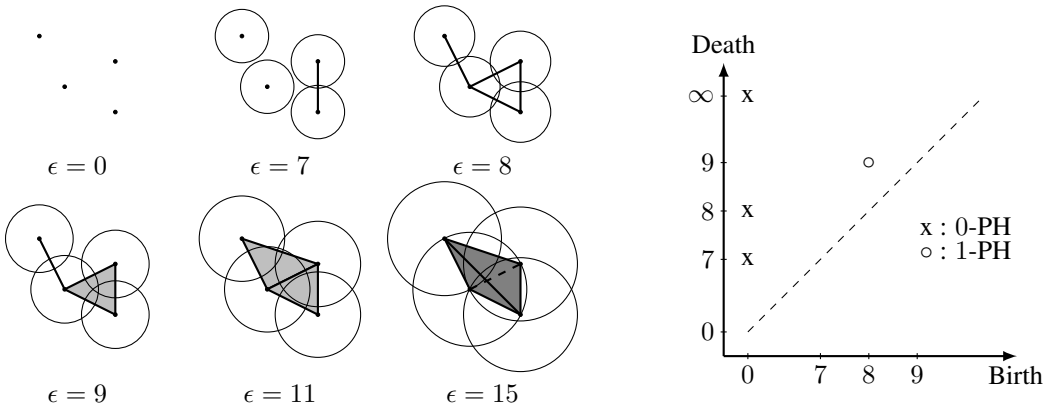
4. We demonstrate the application of our algorithm to settings where (i) the properties of persistence diagrams makes clustering them the natural choice over geometric equivalents and (ii) the probabilistic membership values can be used to rank candidates for top-k selection. Our algorithm classifies transformed lattice structures from materials science where geometric equivalents fail, whilst giving probabilistic rankings to help prioritise expensive further investigation. We also cluster the persistence diagrams of decision boundaries and labelled datasets, showing that our fuzzy clustering captures information about model performance on unseen tasks.

2 Topological preliminaries

Topological Data Analysis emerged from the study of algebraic topology, providing a toolkit to fully describe the topology of a dataset. We offer a quick summary below; for more comprehensive details see Edelsbrunner & Harer (2010). A set of points in $\mathbb{R}^d$ are indicative of the shape of the distribution they are sampled from. By placing a ball of radius $\epsilon > 0$ around each point and connecting points where balls intersect, we can create an approximation of the distribution, called the Čech complex. Specifically, we add the convex hull of k points to the ϵ-Čech complex when the corresponding k balls intersect in a single point. As we increase ϵ, more balls intersect, changing the granularity of our approximation of the distribution. An example of this is shown in Figure 1(a). In our implementation we use a very similar, but more computationally efficient, construction called the Rips complex. This is constructed by adding the convex hull of any collection of points that are pairwise at most ϵ apart to the ϵ-Rips complex. The collection of complexes for all ϵ is called a filtration.

For each ϵ, we compute the p-homology group. This tells us the topology of the ϵ-Rips complex: the 0-homology counts the number of connected components, the 1-homology counts the number of holes, the 2-homology counts the number of voids, etc. (Edelsbrunner et al., 2000). The p-persistent homology (p-PH) group is created by summing the p-homology groups over all ϵ. This results in a p-PH group that summarises information about the topology of the dataset at all granularities. If a topological feature, such as a connected component or hole, persists throughout a large range of granularities, then it's more likely to be a feature of the distribution. If it only persists for a short

[1] Dionysus and Ripser



(a) Čech filtration

(b) Persistence diagram

Figure 1: An example Čech filtration with its persistence diagram

amount of time, then it's more likely to be noise (Cohen-Steiner et al., 2007). We can stably map a p-PH group into a multiset in the extended plane called a persistence diagram (Chazal et al., 2012). Each topological feature has a birth and death: a feature is born when it enters the complex, and dies when the complex grows enough to destroy it. For example, in Figure 1(a), a feature is born at $\epsilon = 8$ when three lines form a hole. This feature dies at $\epsilon = 9$ when the hole is filled in. This is shown in the persistence diagram in Figure 1(b) as a point at $(8, 9)$ in 1-PH. By computing the birth/death points for each topological feature in the filtration, we get a complete picture of the topology of the point cloud at all granularities (Zomorodian & Carlsson, 2005). The persistence diagram is the collection of birth/death points, along with the diagonal $\Delta = \{(a, a) : a \in \mathbb{R}\}$ with infinite mutiplicity, added in order to make the space of persistence diagrams complete (Mileyko et al., 2011).

3 Algorithmic Design

3.1 Clustering persistence diagrams

In order to cluster we need a distance on the space of persistence diagrams. We use the 2-Wasserstein L_2 metric as it is stable on persistence diagrams of finite point clouds (Chazal et al., 2012). The Wasserstein distance is an optimal transport metric that has found applications across machine learning. In the Euclidean case, it quantifies the smallest distance between optimally matched points. Given diagrams $\mathbb{D}_1, \mathbb{D}_2$, the distance is

$$W_2(\mathbb{D}_1, \mathbb{D}_2) = \left(\inf_{\gamma : \mathbb{D}_1 \to \mathbb{D}_2} \sum_{x \in \mathbb{D}_1} ||x - \gamma(x)||_2^2 \right)^{1/2},$$

where the infimum is taken over all bijections $\gamma : \mathbb{D}_1 \to \mathbb{D}_2$. Note that as we added the diagonal with infinite multiplicity to each diagram, these bijections exist. If an off-diagonal point is matched to the diagonal the transportation cost is simply the shortest distance to the diagonal. In fact, the closer a point is to the diagonal, the more likely it is to be noise (Cohen-Steiner et al., 2007), so this ensures our distance is not overly affected by noise.

We work in the space $\mathcal{D}_{\mathrm{L}^2} = \{\mathbb{D} : W_2(\mathbb{D}, \Delta) < \infty\}$,[2] as this leads to a geodesic space with known structure (Turner et al., 2012). Given a collection of persistence diagrams $\{\mathbb{D}_j\}_{j=1}^n \subset \mathcal{D}_{\mathrm{L}^2}$ and a fixed number of clusters c, we wish to find cluster centres $\{\mathbb{M}_k\}_{k=1}^c \subset \mathcal{D}_{\mathrm{L}^2}$, along with membership

[2]To ensure that our persistence diagrams are all in this space, we map points at infinity to a hyperparameter T that is much larger than other death values in the diagram. As we're using the Rips complex, there will only ever be one point at infinity (which will be in 0-PH). This hyperparameter ensures that the one point at infinity will always be matched to the corresponding point at infinity when computing the Wasserstein distance between diagrams.

values $r_{jk} \in [0,1]$ that denote the extent to which $\mathbb{D}_j$ is associated with cluster $\mathbb{M}_k$. We follow probabilistic fuzzy clustering, so that $\sum_k r_{jk} = 1$ for each j.

We extend the FCM algorithm originally proposed by Bezdek (1980). Our r_{jk} is the same as traditional FCM clustering, adapted with the Wasserstein distance. That is,

$$r_{jk} = \left(\sum_{l=1}^{c} \frac{W_2(\mathbb{M}_k, \mathbb{D}_j)}{W_2(\mathbb{M}_l, \mathbb{D}_j)} \right)^{-1}. \tag{1}$$

To update $\mathbb{M}_k$, we compute the weighted Fréchet mean of the persistence diagrams $\{\mathbb{D}_j\}_{j=1}^n$ with the weights $\{r_{jk}^2\}_{j=1}^n$. Specifically,

$$\mathbb{M}_k \longleftarrow \arg\min_{\hat{\mathbb{D}}} \sum_{j=1}^{n} r_{jk}^2 W_2(\hat{\mathbb{D}}, \mathbb{D}_j)^2, \text{ for } k = 1, \ldots, c. \tag{2}$$

As the weighted Fréchet mean extends weighted centroids to general metric spaces, this gives our fuzzy cluster centres. The computation of the weighted Fréchet mean is covered in Section 3.2. By alternatively updating (1) and (2) we get a sequence of iterates. Theorem 1, proven in Appendix A, provides the same convergence guarantees as traditional FCM clustering.

Theorem 1. *Every convergent subsequence of the sequence of iterates obtained by alternatively updating membership values and cluster centres with (1) and (2) tends to a local minimum or saddle point of the cost function* $J(R, \mathbb{M}) = \sum_{j=1}^{n} \sum_{k=1}^{c} r_{jk}^2 W_2(\mathbb{M}_k, \mathbb{D}_j)^2$.

Observe that we only guarantee the convergence of subsequences of iterates. This is the same as traditional FCM clustering, so we follow the same approach to a stopping condition and run our algorithm for a fixed number of iterations. The entire algorithm is displayed in Algorithm 1.

Algorithm 1 FPDCluster

Input Diagrams $\mathbb{D} = \{\mathbb{D}_j\}_{j=1}^n$, number of clusters c, maximum iterations MAXITER
Output Cluster centres $\mathbb{M} = \{\mathbb{M}_k\}_{k=1}^c$, membership values $R = \{r_{jk}\}$

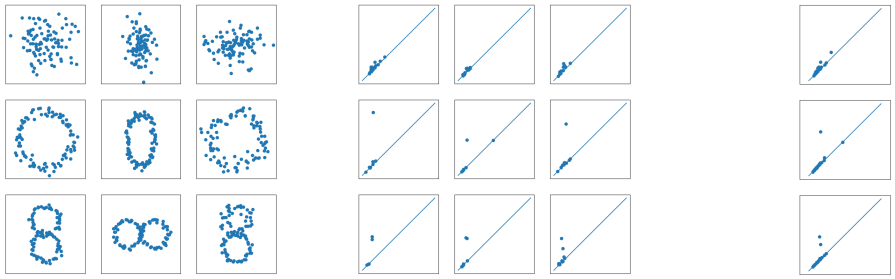
```
D = AddDiagonals(D)
M = InitCentres(D)
for count in 1..MAXITER do
    for j in 1..n do
        for k in 1..c do
            r_jk ← ( Σ_{l=1}^{c} W_2(M_k, D_j) / W_2(M_l, D_j) )^{-1}
        end for
    end for
    for k in 1..c do
        M_k ← WFrechetMean(D, R_k)
    end for
end for
return M, R
```

3.2 Computing the weighted Fréchet mean

Turner et al. (2012) give an algorithm for the computation of Fréchet means. In this section we extend their algorithm and proof of convergence to the weighted case. In Algorithm 1 we add copies of the diagonal to ensure that each diagram has the same cardinality; denote this cardinality as m. To compute the weighted Fréchet mean, we need to find $\mathbb{M}_k = \{y^{(i)}\}_{i=1}^m$ that minimises the Fréchet function in (2). Implicit to the Wasserstein distance is a bijection $\gamma_j : y^{(i)} \mapsto x_j^{(i)}$ for each j. Supposing we know these bijections, we can rearrange the Fréchet function into the form $F(\mathbb{M}_k) = \sum_{j=1}^{n} r_{jk}^2 W_2(\mathbb{M}_k, \mathbb{D}_j)^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} r_{jk}^2 ||y^{(i)} - x_j^{(i)}||^2$.

In this form, the summand is minimised for $y^{(i)}$ by the weighted Euclidean centroid of the points $\{x_j^{(i)}\}_{j=1}^n$. Therefore to compute the weighted Fréchet mean, we need to find the correct bijections. We start by using the Hungarian algorithm to find an optimal matching between $\mathbb{M}_k$ and each $\mathbb{D}_j$. Given a $\mathbb{D}_j$, for each point $y^{(i)} \in \mathbb{M}_k$, the Hungarian algorithm will assign an optimally matched point $x_j^{(i)} \in \mathbb{D}_j$. Specifically, we find matched points

$$\left[x_j^{(i)} \right]_{i=1}^{m} \longleftarrow \text{Hungarian}(\mathbb{M}_k, \mathbb{D}_j), \text{ for each } j = 1, \ldots, n. \tag{3}$$



(a) Synthetic datasets (b) 1-PH persistence diagrams (c) Cluster centres

Figure 2: Our algorithm successfully clustered the persistence diagrams, finding cluster centres that represent the topology of the datasets. The cluster centres have zero, one, or two significant off-diagonal points, representing zero, one, or two holes in the datasets.

Now, for each $y^{(i)} \in \mathbb{M}_k$ we need to find the weighted average of the matched points $\left[x_j^{(i)}\right]_{j=1}^n$. However, some of these points could be copies of the diagonal, so we need to consider three distinct cases: that each matched point is off-diagonal, that each one is a copy of the diagonal, or that the points are a mixture of both. We start by partitioning $1, \ldots, n$ into the indices of the off-diagonal points $\mathcal{J}_{\text{OD}}^{(i)} = \left\{j : x_j^{(i)} \neq \Delta\right\}$ and the indices of the diagonal points $\mathcal{J}_{\text{D}}^{(i)} = \left\{j : x_j^{(i)} = \Delta\right\}$ for each $i = 1, \ldots, m$. Now, if $\mathcal{I}_{\text{OD}} = \emptyset$ then $y^{(i)}$ is a copy of the diagonal. If not, let $w = \left(\sum_{j \in \mathcal{J}_{\text{OD}}^{(i)}} r_{jk}^2\right)^{-1} \sum_{j \in \mathcal{J}_{\text{OD}}^{(i)}} r_{jk}^2 x_j^{(i)}$ be the weighted mean of the off-diagonal points. If $\mathcal{J}_{\text{D}}^{(i)} = \emptyset$, then $y^{(i)} = w$. Otherwise, let w_Δ be the point on the diagonal closest to w. Then our update is

$$y^{(i)} \longleftarrow \frac{\sum_{j \in \mathcal{J}_{\text{OD}}^{(i)}} r_{jk}^2 x_j^{(i)} + \sum_{j \in \mathcal{J}_{\text{D}}^{(i)}} r_{jk}^2 w_\Delta}{\sum_{j=1}^n r_{jk}^2}, \text{ for } i = 1, \ldots, m. \tag{4}$$

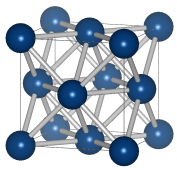
We alternate between (3) and (4) until the matching remains the same. Theorem 2, proving that this algorithm converges to a local minimum of the Fréchet function, is proven in Appendix B.

Theorem 2. *Given diagrams $\mathbb{D}_j$, membership values r_{jk}, and the Fréchet function $F(\hat{\mathbb{D}}) = \sum_{j=1}^n r_{jk}^2 W_2(\hat{\mathbb{D}}, \mathbb{D}_j)^2$, then $\mathbb{M}_k = \{y^{(i)}\}_{i=1}^m$ is a local minimum of F if and only if there is a unique optimal pairing from $\mathbb{M}_k$ to each of the $\mathbb{D}_j$ and each $y^{(i)}$ is updated via (4).*

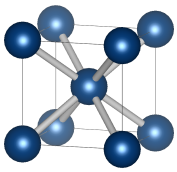
4 Experiments

4.1 Synthetic data

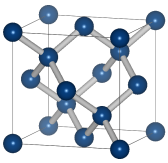
We start by demonstrating our algorithm on a simple synthetic dataset designed to highlight its ability to cluster based on the topology of the underlying datasets. We produce three datasets of noise, three datasets of a ring, and three datasets of figure-of-eights, all shown in Figure 2(a). In Figure 2(b) we show the corresponding 1-PH persistence diagrams. Note that the persistence diagrams have either zero, one, or two significant off-diagonal points, corresponding to zero, one, or two holes in the datasets. We then use our algorithm to cluster the nine persistence diagrams into three clusters. Having only been given the list of diagrams, the number of clusters, and the maximum number of iterations, our algorithm successfully clusters the diagrams based on their topology. Figure 2(c) shows that the cluster centres have zero, one, or two off-diagonal points: our algorithm has found cluster centres that reflect the topological features of the datasets. Because we are reducing the cardinality and dimensionality of datasets by mapping into persistence diagrams, we also demonstrate a speed-up



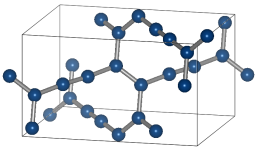
(a) Face-centred cubic structure



(b) Body-centred cubic structure



(c) Diamond



(d) Cis-hinger polydiacetylene

Figure 3: Cubic structures (top) and carbon allotropes (bottom). Our algorithm can cluster transformed lattice structures where comparable geometric algorithms fail.

of at least an order of magnitude over Wasserstein barycentre clustering methods. Details of these timing experiments are in Appendix C.1.[3]

4.2 Lattice Structures

A key property for machine learning in materials science has been identified as "invariance to the basis symmetries of physics... rotation, reflection, translation" (Schmidt et al., 2019). Removing the need for a standardised coordinate system allows machine learning methods to be applied to a broader range of existing coordinate datasets generated by experimental methods (e.g., x-ray diffraction and scanning electron microscope imaging) and computational methods (e.g., density functional theory). Persistence diagrams, which capture affine transformation-invariant properties of datasets, are ideally suited for application in this domain. Our algorithm enables successful unsupervised learning on these datasets for the first time. Additionally, the fuzzy membership values allow top-k ranking of candidates suggested by our algorithm. This is particularly important in materials science, where further investigation of materials can be extremely costly.

The large majority of solids are comprised of lattices: regularly repeating unit cells of atoms. This lattice structure directly determines the properties of a material (Hull & Bacon, 2011) and it has been predicted that machine learning will reveal presently unknown links between structure and property by identifying new trends across materials (Meredig, 2019; Wei et al., 2019). We apply our algorithm to two examples of lattice structures from materials science: cubic structures and carbon allotropes. Cubic structures are important because they are ubiquitous. The most common lattice structures, particularly amongst pure metals, are face-centred cubic (FCC) structures and body-centred cubic (BCC) structures (Putnis, 1992), shown in Figures 3(a) and 3(b). Carbon allotropes, such as graphene and diamond, are widely anticipated to revolutionise electronics and optoelectronics (Wang et al., 2016). We focus on the carbon allotropes diamond and cis-hinged polydiacetylene, shown in Figures 3(c) and 3(d).

We use atomic positions for the unit-cells of iron mp-150 and iron mp-13 from the Materials Project (Jain et al., 2013), representing BCC and FCC structures respectively, for our first experiment. For our second experiment we use diamond and cis-hinged polydiacetylene unit-cell atomic positions from the Samara Carbon Allotrope Database (Hoffmann et al., 2016). We simulate distinct collections of lattices by transforming the atomic coordinates, with no information about bonds given to the algorithms. The properties of persistence diagrams mean that we can successfully cluster the atomic coordinates derived from the same base unit-cell regardless of the transformations applied to the coordinate system, fulfilling the key property identified above (we consider a clustering successful when all datasets from the same lattice structure have their highest membership values for the same cluster). In comparison, we run Wasserstein barycentre clustering on the same datasets using several

[3]Further details of all experiments are available in Appendix C.

state-of-the-art algorithms for barycentre computation and approximation. Each can only successfully cluster the cubic structures after reflection, and none of them successfully cluster the carbon allotropes after any transformations. We give specific values for these results in Appendix C.2.

4.3 Decision boundaries

Learnt models have been shown to perform better on datasets which have a similar persistence diagram to the model's decision boundary (Ramamurthy et al., 2019). In fact, topological complexity has been shown to correlate with generalisation ability (Guss & Salakhutdinov, 2018; Gabrielsson & Carlsson, 2019). We utilise our algorithm to cluster the topology of models and tasks, showing that high membership values imply better performance on tasks. Specifically, given a dataset with n classes, we fix one class to define $n-1$ *tasks*: binary classification of the fixed class vs each of the remaining classes. On each of these tasks, we train a *model*. We compute the decision boundary of the model f, defined as $(x_1, \ldots, x_m, f(x))$ where $f(x)$ is the model's prediction for $x = (x_i)_i$, and the decision boundary of the tasks, defined via the labelled dataset as $(x_1, \ldots, x_m, y)$ where y is the true label. We compute the 1-persistence diagrams of the tasks' and models' decision boundaries and cluster them to obtain membership values and cluster centres. To view task and model proximity through our clustering, we find the cluster centre with the highest membership value for each task, and consider the models closest to that cluster centre. Note that in general you cannot do this with hard clustering: most of the time a path will not exist from task to cluster centre to model, because each task/model is only associated with a single cluster. This contrasts with fuzzy clustering, where you have information about how close each model/task is to each cluster centre. We further discuss why this does not work for hard clustering in Appendix C.3.

To assess the ability of our model/task clustering, we performed the above experiment on three different datasets: MNIST (LeCun et al., 2010), FashionMNIST (Xiao et al., 2017), and Kuzushiji-MNIST (Clanuwat et al., 2018). We repeat each experiment three times. Our goal is to evaluate whether or not the clustering is capturing information about model performance on tasks, so as a baseline we use the average performance of all models on a fixed task, averaged over all tasks. Recall that we are just using the topology of the decision boundary: no information about model performance on the task was used until we assessed the ability of the clustering to capture information about the tasks. We start by verifying what happens if we use the model closest to the cluster centre associated with the task (i.e., top-1). We see a significant increase in performance, indicating that the topological fuzzy clustering has selected the model trained on the task: this means the clustering is working. We also compute the top-3 and top-2 performance change over average. We still see a statistically significant increase in performance over average performance, indicating that our membership values are capturing information about model performance on unseen tasks. These results are shown in Table 1. By successfully matching models to tasks using fuzzy clustering, we offer a new technique for future model marketplaces, demonstrating the value of topological clustering when analysing decision boundaries.

Table 1: Performance increase/decrease over average task performance when using learnt fuzzy membership values for model selection. The increase in performance demonstrates that our fuzzy clustering automatically clusters models near tasks they perform well on.

	Performance change vs random model selection (%)		
	Top-3	Top-2	Top-1
MNIST	$+6.17^{\pm 2.18}$	$+10.81^{\pm 1.88}$	$+20.88^{\pm 4.08}$
FashionMNIST	$+16.46^{\pm 4.00}$	$+21.94^{\pm 4.73}$	$+23.30^{\pm 8.72}$
Kuzushiji	$+6.61^{\pm 1.78}$	$+11.18^{\pm 2.45}$	$+21.89^{\pm 5.54}$

5 Conclusion

We have developed FCM clustering on the space of persistence diagrams, adding an important class of unsupervised learning to Topological Data Analysis' toolkit. Our topological fuzzy clustering paves the way for applications in materials science and pre-trained model marketplaces, where we envisage our algorithm being used as part of automated materials discovery and model selection.

REFERENCES

Jean-David Benamou, Guillaume Carlier, Marco Cuturi, Luca Nenna, and Gabriel Peyré. Iterative bregman projections for regularized transportation problems. *SIAM Journal on Scientific Computing*, 2015. doi: 10.1137/141000439.

J. C. Bezdek. A convergence theorem for the fuzzy isodata clustering algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-2(1):1–8, Jan 1980. ISSN 1939-3539. doi: 10.1109/TPAMI.1980.4766964.

J. C. Bezdek, R. J. Hathaway, M. J. Sabin, and W. T. Tucker. Convergence theory for fuzzy c-means: Counterexamples and repairs. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(5): 873–877, 1987.

Peter Bubenik. Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research*, 16(3):77–102, 2015. URL http://jmlr.org/papers/v16/bubenik15a.html.

Peter Bubenik and Alexander Wagner. Embeddings of persistence diagrams into hilbert spaces. *CoRR*, abs/1905.05604, 2019. URL http://arxiv.org/abs/1905.05604.

Mathieu Carrière and Ulrich Bauer. On the metric distortion of embedding persistence diagrams into separable hilbert spaces. In *Symposium on Computational Geometry*, 2019.

Mathieu Carrière, Marco Cuturi, and S. Oudot. Sliced wasserstein kernel for persistence diagrams. In *ICML*, 2017.

Mathieu Carrière, Frédéric Chazal, Yuichi Ike, T. Lacombe, Martin Royer, and Y. Umeda. Perslay: A neural network layer for persistence diagrams and new graph topological signatures. In *AISTATS*, 2020.

Frédéric Chazal, Vin Silva, Marc Glisse, and Steve Oudot. *The Structure and Stability of Persistence Modules*. 07 2012. doi: 10.1007/978-3-319-42545-0.

Chao Chen, Xiuyan Ni, Qinxun Bai, and Yusu Wang. Toporeg: A topological regularizer for classifiers. *CoRR*, abs/1806.10714, 2018. URL http://arxiv.org/abs/1806.10714.

Sofya Chepushtanova, Tegan Emerson, Eric Hanson, Michael Kirby, Francis Motta, Rachel Neville, Chris Peterson, Patrick Shipman, and Lori Ziegelmeier. Persistence images: An alternative persistent homology representation. 07 2015.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature, 2018.

J. Clough, N. Byrne, I. Oksuz, V. A. Zimmer, J. A. Schnabel, and A. King. A topological loss function for deep-learning based image segmentation using persistent homology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, Jan 2007. ISSN 1432-0444. doi: 10.1007/s00454-006-1276-5. URL https://doi.org/10.1007/s00454-006-1276-5.

Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. volume 32 of *Proceedings of Machine Learning Research*, pp. 685–693, Bejing, China, 22–24 Jun 2014. PMLR. URL http://proceedings.mlr.press/v32/cuturi14.html.

F. d. A. T. de Carvalho, A. Irpino, and R. Verde. Fuzzy clustering of distribution-valued data using an adaptive l2 wasserstein distance. In *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–8, 2015.

Herbert Edelsbrunner and John Harer. *Computational Topology - an Introduction.* American Mathematical Society, 2010. ISBN 978-0-8218-4925-5.

Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. volume 28, pp. 454 – 463, 02 2000. ISBN 0-7695-0850-2. doi: 10.1109/SFCS.2000.892133.

Barbara Di Fabio and Massimo Ferri. Comparing persistence diagrams through complex vectors. In *Image Analysis and Processing — ICIAP 2015*, pp. 294–305. Springer International Publishing, 2015. doi: 10.1007/978-3-319-23231-7_27. URL https://doi.org/10.1007/978-3-319-23231-7_27.

Rickard Brüel Gabrielsson and G. Carlsson. Exposition and interpretation of the topology of neural networks. *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pp. 1069–1076, 2019.

Rickard Brüel Gabrielsson, Bradley J. Nelson, Anjan Dwaraknath, and Primoz Skraba. A topology layer for machine learning. volume 108 of *Proceedings of Machine Learning Research*, pp. 1553–1563, Online, 26–28 Aug 2020. PMLR. URL http://proceedings.mlr.press/v108/gabrielsson20a.html.

Marian Gidea and Yuri Katz. Topological data analysis of financial time series: Landscapes of crashes. *Physica A: Statistical Mechanics and its Applications*, 491:820–834, February 2018. doi: 10.1016/j.physa.2017.09.028. URL https://doi.org/10.1016/j.physa.2017.09.028.

William H. Guss and R. Salakhutdinov. On characterizing the capacity of neural networks using algebraic topology. *ArXiv*, abs/1802.04443, 2018.

Christoph Hofer, Roland Kwitt, Marc Niethammer, and Andreas Uhl. Deep learning with topological signatures. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 1634–1644. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6761-deep-learning-with-topological-signatures.pdf.

Roald Hoffmann, Artem Kabanov, Andrey Golov, and Davide Proserpio. Homo citans and carbon allotropes: For an ethics of citation. *Angewandte Chemie International Edition*, 55, 07 2016. doi: 10.1002/anie.201600655.

D. Hull and D.J. Bacon. *Introduction to Dislocations (Fifth Edition)*. Butterworth-Heinemann, Oxford, fifth edition edition, 2011. ISBN 978-0-08-096672-4. doi: https://doi.org/10.1016/B978-0-08-096672-4.00019-0. URL http://www.sciencedirect.com/science/article/pii/B9780080966724000190.

Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin a. Persson. The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013. ISSN 2166532X. doi: 10.1063/1.4812323. URL http://link.aip.org/link/AMPADS/v1/i1/p011002/s1&Agg=doi.

Sara Kališnik. Tropical coordinates on the space of persistence barcodes. *Foundations of Computational Mathematics*, 19(1):101–129, January 2018. doi: 10.1007/s10208-018-9379-y. URL https://doi.org/10.1007/s10208-018-9379-y.

Michael Kerber, Dmitriy Morozov, and Arnur Nigmetov. Geometry helps to compare persistence diagrams. *Journal of Experimental Algorithmics (JEA)*, 22:1–20, 2017.

Kwangho Kim, Jisu Kim, J. Kim, Frédéric Chazal, and L. Wasserman. Efficient topological layer based on persistent landscapes. *ArXiv*, abs/2002.02778, 2020.

Frank Klawonn. Fuzzy clustering: Insights and a new approach. *Mathware & soft computing, ISSN 1134-5632, Vol. 11, Nº. 3, 2004, pags. 125-142*, 11, 01 2004.

Violeta Kovacev-Nikolic, Peter Bubenik, Dragan Nikolić, and Giseon Heo. Using persistent homology and dynamical distances to analyze protein binding. Statistical Applications in Genetics and Molecular Biology. January 2016, Volume 15, Issue 1, Pages 19-38, 2014.

Theo Lacombe, Marco Cuturi, and Steve Oudot. Large scale computation of means and clusters for persistence diagrams using optimal transport. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 9770–9780. Curran Associates, Inc., 2018.

Tam Le and Makoto Yamada. Persistence fisher kernel: A riemannian manifold kernel for persistence diagrams. In *NeurIPS*, 2018.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

J. Li and J. Z. Wang. Real-time computerized annotation of pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):985–1002, 2008.

H. Ling and K. Okada. An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):840–853, 2007.

Vasileios Maroulas, Joshua Mike, and Andrew Marchese. K-means clustering on the space of persistence diagrams. In *SPIE*, pp. 29, 08 2017. doi: 10.1117/12.2273067.

Bryce Meredig. Five high-impact research areas in machine learning for materials science. *Chemistry of Materials*, 31(23):9579–9581, 2019. doi: 10.1021/acs.chemmater.9b04078. URL https://doi.org/10.1021/acs.chemmater.9b04078.

Yuriy Mileyko, Sayan Mukherjee, and John Harer. Probability measures on the space of persistence diagrams. *Inverse Problems - INVERSE PROBL*, 27, 12 2011. doi: 10.1088/0266-5611/27/12/124007.

M. Moor, Max Horn, Bastian Alexander Rieck, and K. Borgwardt. Topological autoencoders. *ArXiv*, abs/1906.00722, 2019.

Andrew Putnis. *An Introduction to Mineral Sciences*. Cambridge University Press, 1992. doi: 10.1017/CBO9781139170383.

Karthikeyan Natesan Ramamurthy, Kush Varshney, and Krishnan Mody. Topological data analysis of decision boundaries with application to model selection. volume 97 of *Proceedings of Machine Learning Research*, pp. 5351–5360, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/ramamurthy19a.html.

J. Reininghaus, S. Huber, U. Bauer, and R. Kwitt. A stable multi-scale kernel for topological machine learning. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4741–4748, 2015.

Bastian Alexander Rieck, F. Sadlo, and H. Leitte. Topological machine learning with persistence indicator functions. *ArXiv*, abs/1907.13496, 2019.

Jonathan Schmidt, Mário R. G. Marques, Silvana Botti, and Miguel A. L. Marques. Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials*, 5(1):83, 2019. ISSN 2057-3960. doi: 10.1038/s41524-019-0221-0. URL https://doi.org/10.1038/s41524-019-0221-0.

Katharine Turner and Gard Spreemann. Same but different: Distance correlations between topological summaries. *arXiv: Algebraic Topology*, 2019.

Katharine Turner, Yuriy Mileyko, Sayan Mukherjee, and John Harer. Fréchet means for distributions of persistence diagrams. *Discrete & Computational Geometry*, 52:44–70, 2012.

J. Vidal, J. Budin, and J. Tierny. Progressive wasserstein barycenters of persistence diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):151–161, 2020.

Alexander Wagner. Nonembeddability of persistence diagrams with p>2 wasserstein metric. *ArXiv*, abs/1910.13935, 2019.

Zhanyu Wang, F. Dong, Bo Shen, R. Zhang, Y. Zheng, L. Chen, Songyou Wang, Chongmin Wang, K. Ho, Yuan-Jia Fan, Bih-Yaw Jin, and Wan-Sheng Su. Electronic and optical properties of novel carbon allotropes. *Carbon*, 101:77–85, 01 2016. doi: 10.1016/j.carbon.2016.01.078.

Jing Wei, Xuan Chu, Xiang-Yu Sun, Kun Xu, Hui-Xiong Deng, Jigen Chen, Zhongming Wei, and Ming Lei. Machine learning in materials science. *InfoMat*, 1:338–358, 09 2019. doi: 10.1002/inf2.12028.

B. Wilder, Eric Ewing, B. Dilkina, and Milind Tambe. End to end learning and optimization on graphs. In *NeurIPS*, 2019.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

J. Ye and J. Li. Scaling up discrete distribution clustering using admm. In *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 5267–5271, 2014.

Jianbo Ye, Panruo Wu, James Wang, and Jia Li. Fast discrete distribution clustering using wasserstein barycenter with sparse support. *IEEE Transactions on Signal Processing*, PP:1–1, 01 2017. doi: 10.1109/TSP.2017.2659647.

W.I. Zangwill. *Nonlinear programming: a unified approach*. Prentice-Hall international series in management. Prentice-Hall, 1969. URL https://books.google.co.uk/books?id=TWhxLcApH9sC.

Qi Zhao and Yusu Wang. Learning metrics for persistence-based summaries and applications for graph classification. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 9859–9870. Curran Associates, Inc., 2019.

Afra Zomorodian and Gunnar Carlsson. Computing Persistent Homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005. ISSN 1432-0444. doi: 10.1007/s00454-004-1146-y. URL https://doi.org/10.1007/s00454-004-1146-y.