Foundations of Data Science
doi:10.3934/fods.2021012

A DENSITY-BASED APPROACH TO FEATURE DETECTION IN PERSISTENCE DIAGRAMS FOR FIRN DATA

Austin Lawson*

Program of Informatics and Analytics
University of North Carolina at Greensboro
Greensboro, NC 27402, USA

Tyler Hoffman

Department of Mathematics
University of Maryland
College Park, MD 20742, USA

Yu-Min Chung

Department of Mathematics and Statistics
University of North Carolina at Greensboro
Greensboro, NC 27402, USA

Kaitlin Keegan

Department of Geological Sciences and Engineering
University of Nevada, Reno
Reno, NV 89557, USA

Sarah Day

Department of Mathematics
William & Mary
Williamsburg, VA 23185, USA

(Communicated by Gunnar Carlsson)

Abstract. Topological data analysis, and in particular persistence diagrams, are gaining popularity as tools for extracting topological information from noisy point cloud and digital data. Persistence diagrams track topological features in the form of k-dimensional holes in the data. Here, we construct a new, automated approach for identifying persistence diagram points that represent robust long-life features. These features may be used to provide a more accurate estimate of Betti numbers for the underlying space. This approach extends

2020 *Mathematics Subject Classification.* Primary: 55N31; Secondary: 62R40, 62H30.

Key words and phrases. Topological data analysis, persistence diagrams, feature extraction, clustering, firn, paleoclimate.

Hoffman and Chung were partially supported by NSF Grant DMS-1950549. Chung, Keegan, and Day were partially supported by the Army Research Office under Grant Number W911NF-20-1-0131. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

* Corresponding author: Austin Lawson.

the established practice of using a lifespan cutoff on the features in order to take advantage of the observation that noisy features typically appear in clusters in the persistence diagram. We show that this approach offers more flexibility in partitioning features in the persistence diagram, resulting in greater accuracy in computed Betti numbers, especially in the case of high noise levels and varying image illumination. This work is motivated by 3-dimensional Micro-CT imaging of ice core samples, and is applicable for separating noise from robust signals in persistence diagrams from noisy data.

1. **Introduction.** Topological data analysis (TDA) is a rising field at the intersection of Mathematics, Statistics, and Machine Learning [14, 4, 25, 6]. Techniques from this field have proven successful in analyzing a variety of scientific problems and datasets (see e.g. [23] which includes a long list of application areas). The main driving force in TDA is the development of *persistent homology* (see e.g. introductory texts [10, 26]), which studies the intrinsic shape of data. Persistent homology is a mathematical construction from the field of algebraic topology, and in practice, *persistence diagrams* (summaries of the persistent homology) yield topological information. A persistence diagram is a multi-set that contains birth and death coordinates for each topological feature in an appropriate filtration of the data. In this work, the filtration is a series of nested subsets of pixels in a grayscale image that correspond to increasing the grayscale threshold whereby pixels with smaller values are considered to be included in the dark/black portion of the image. Because of our main application to ice core samples, we will often invert the images so that we instead study the light/white portion of the image corresponding to ice. The birth and death coordinates record the position in the filtration of the data where a topological feature such as a connected component or a higher dimensional hole first and last appears respectively.

Birth/death coordinates reveal information about when the feature appears in the filtration and about the robustness of a feature. Typically, a feature a with long lifespan (the difference between the death and birth coordinates) is considered to be a robust/true feature while a feature with a short lifespan is treated as less important and most likely due to noise. Generators arising due to noise in the data will also often appear in clusters near the diagonal (where the death coordinate is approximately equal to the birth coordinate) in the persistence diagram. We demonstrate this in a series of benchmark studies in Section 3. One of the fundamental goals in this area is to develop methods for automatically partitioning persistence diagrams in order to separate robust features from features due to noise. In this work, we propose a method called *Density-based Persistent Feature Detection* (DPFD) that trains an extension of the well-known DBSCAN method [11] for clustering, in order to partition persistence diagrams and calculate Betti numbers, the counts of topological features of different dimensions. The application of DPFD in this work focuses on counting and labelling contiguous regions in images, which amounts to considering Betti 0 of sublevel set filtrations. However, it is important to note that the DPFD algorithm may be applied on diagrams of any dimension.

Related work in this area has established algorithmic approaches to this problem. Authors of the work [12], one of the first results in this direction, developed a statistical approach to separate robust features from noise. They describe four bootstrapping and sampling methods for constructing a confidence interval around the diagonal of a persistence diagram. Beyond this interval, generators are considered to be true features; within the interval, generators are considered to be noise. This works well for some large point cloud data sets and filtrations on simplicial

complexes. While the general approach of applying a lifespan cutoff to separate long from short lifespan features works well for some persistence diagrams, especially in the case of small amplitude noise, it does not work well for the ones we consider here, as we demonstrate in Section 3. Two other approaches also focus on partitioning persistence points. In [3, 2], the authors describe a point-wise criteria that separates the topological noise from robust features. This point-wise criteria is based on *persistence entropy* (see [3, 2]), which gives a measure of complexity on persistence diagrams. This work also focuses on point cloud data and simplicial complexes. Finally, image data and cubical complexes motivated the PD Thresholding method [7], designed as a topological optimization procedure for segmenting a grayscale image into a binary one. Persistence points that are present in the thresholded binary image, perhaps with the addition of a lifespan cutoff, are then considered to be 'robust' features.

As we show in Section 3, our proposed method outperforms these methods on the noisy image data sets we consider. In particular, the geometry imposed by both lifespan cutoff and PD Thresholding on the partition boundaries applied in the persistence diagram lack the necessary flexibility to separate robust from noisy features in the case of varying image illumination and moderate noise levels. Our proposed method, on the other hand, uses a clustering algorithm to allow for a more highly adaptive approach and improved accuracy. We demonstrate this first on a conceptual rice image where tagging features in the image allows us to verify the accuracy of the classification of features in the persistence diagram. We then use a machine learning approach on our firn application example to tune the necessary parameters and achieve improved accuracy in Betti number calculations over a range of images.

The motivation for this approach, and application we study in Section 4, is the quantification of the microstructure of *firn*, a porous medium that occurs in polar regions where snow rarely melts. In polar regions, snow continuously accumulates and creates a column of firn layers in the topmost region of ice sheets. These firn layers consist of ice (lighter/grey pixels, Figure 1) and interconnected pore-space (darker/black pixels, Figure 1), and have increasing density with depth due to the increasing overburden pressure from accumulating snowfall at the surface. The pore-space is connected to the overlying atmosphere, and allows gases to diffuse to the bottom of the firn column where the layers reach the density of glacial ice and the pores close off into individual bubbles. These bubbles trap a direct sample of the atmospheric air into the glacial ice, and provide a powerful record of Earth's past atmospheric composition. Understanding the changes in firn structure with depth is critically important for interpreting ice-core paleoclimate records [24].

One way to examine the firn microstructure is to three-dimensionally scan samples using x-ray computed tomography (Micro-CT) [17]. This process produces a stack of two-dimensional grayscale images that represent the original firn sample. The 2D reconstructed images are also referred to as "slices" because they represent the cross-section of firn microstructure at specific sample depths, as if you were slicing through the sample at that depth to reveal the internal structure. Stacking the slices generates a three-dimensional reconstruction of the entire firn sample, where Slice 1 and Slice 900 represent the top and bottom of the sample, respectively. As counts of topological features, Betti numbers, β_i, can be used to quantify topological shifts in the firn microstructure with depth. We focus here on β_0, which gives a count of the number of connected components of a given set. Betti numbers are

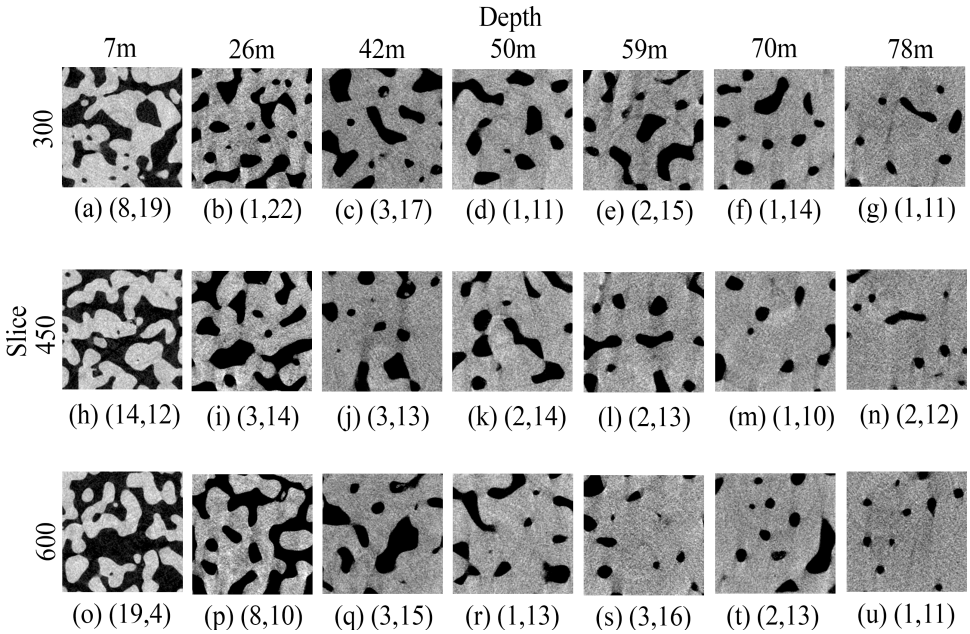

FIGURE 1. Samples of firn Micro-CT images at different depths. Lighter grey regions represent ice-space and darker regions represent pore-space in each image. In this work, there are 14 firn samples (not columns) in total, the depths of which range from 7 to 78 m in the firn column. Only select samples from that range of depths are shown here. Each sample consists of over 900 cross-sectional slices of 2D grayscale images of size 400×400 pixels. In other words, each sample is a 3D image of dimension $400 \times 400 \times 900$ voxels. We refer readers to our github page (`https://github.com/azlawson/Firn`) for a visualization of these 3D images. Only the 300th, 450th, and 600th slices from each sample's stack of slices are shown here. Counts of contiguous ice and pore-space regions, respectively, as determined by an expert in firn, are shown in parentheses.

shown in Figure 1 for the ice (lighter) and pore-space (darker) portions of the images and we will refer to these numbers as the counts of ice regions and pore regions respectively to avoid confusion. As shown in Section 2 and as is typical for noisy images, points most likely due to noise dominate persistence diagrams. Methods for separating features corresponding to the firn pore or ice spaces from features arising from measurement/image noise are essential in order to get useful estimates of Betti numbers for firn. As we show in Section 4, the method we propose here produces Betti numbers with reasonably low error. As a sample result, Figure 12 shows the resulting Betti number estimates as a function of firn depth, recording the topological shift that occurs in the firn column.

In what follows, we outline some mathematical background on persistent homology in Section 2 and motivate the need for a new approach to partitioning persistence diagrams. We present the proposed method, DPFD, and give empirical

results in Section 3. In Section 4, we apply DPFD to the study of firn, calculating Betti numbers as a means of tracking topological changes in the microstructure as a function of depth. Finally, we conclude the paper with a discussion of the firn results and directions for future work in Section 5.

2. Background and motivation.

Persistent homology is a tool from TDA that is used to extract topological and geometric information from data. It is a versatile tool that can be applied to many types of complex datasets such as point clouds, time series, images, and more. We provide a light introduction to homology and persistent homology via cubical homology, but we note that other homology theories will suffice.

2.1. Cubical sets.

This section loosely follows the cubical homology development found in [16]. An **elementary interval** is a closed interval $I \subset \mathbb{R}$ where $I = [\ell, \ell+1]$ or $I = [\ell, \ell]$ (degenerate) and $\ell \in \mathbb{Z}$. An **elementary cube** Q is a direct product of elementary intervals and the **dimension** of an elementary cube Q is the number of non-degenerate elementary intervals in the product. For example, a 0-dimensional cube is a single vertex, a 1-dimensional cube is a unit length segment, and a 2-dimensional cube is a square with unit length sides. As our main application in this paper is to 2-dimensional images, we need only focus on cubes of dimension 2 or less. A cube P is said to be a face of Q if $P \subset Q$. A **cubical set** is a subset of real numbers $X \subset \mathbb{R}$ that can be written as a finite union of elementary cubes. We define $\mathcal{K}_k(X)$ to be the collection of k-dimensional cubes in X.

2.2. Cubical homology.

Homology is a tool from algebraic topology that allows us to assign algebraic objects (namely a group or in our case a vector space) to a topological space. Cubical homology allows us to systematically assign these objects to cubical sets. Typically, homology is defined over a general ring of coefficients, however for our treatment, we will fix the ring to be $\mathbb{Z}_2$. This allows us to forgo discussion of orientation on cubical sets. Let X be a cubical set. For each k, we generate vector spaces of formal linear combinations $C_k(X)$ with basis $\mathcal{K}_k(X)$ and coefficients coming from $\mathbb{Z}_2$. We next define linear maps $\partial_k : C_k(X) \to C_{k-1}(X)$ called *boundary maps* that satisfy $\partial_{k+1} \circ \partial_k \equiv 0$. We will not explicitly define this map here, but we will note that this latter property tells us that the kernel of this map, $\ker \partial_k$, also called the *k-cycles*, is a subspace of $\operatorname{im} \partial_k$, of the *$k+1$-boundaries*. Thus, this allows us to immediately define the k-th homology group $H_k(X)$ as the quotient group $H_k(X) = \ker \partial_k / \operatorname{im} \partial_{k+1}$. The nontrivial elements in the homology group correspond exactly to those cycles or combinations of elementary cubes that are not part of a boundary. Intuitively, generators of $H_k(X)$ correspond to k-dimensional topological features or "holes" in X.

2.3. Persistent homology.

Persistent homology allows us to track changes in homological features over a nested sequence of topological spaces. A sequence of topological spaces X_i satisfying $X_{i_1} \subseteq X_{i_2} \subseteq \ldots \subseteq X_{i_N}$, is called a *filtration*. The resulting inclusion maps $\iota^{p,q} : X_{i_p} \to X_{i_q}$, where $p < q$, then induce linear homomorphisms $\iota_k^{p,q} : H_k(X_{i_p}) \to H_k(X_{i_q})$. These maps on homology allow us to track changes in the set of topological features over this filtration. We say that a feature $\alpha \in H_k(X_b)$ is *born* at index b if $\alpha \notin \operatorname{im}(\iota^{b-1,b})$ (or $b = 1$). We say that this same feature α *dies* at d if $\alpha \in \operatorname{im}(\iota^{b,d-1})$ and $\alpha \notin \operatorname{im}(\iota^{b,d})$. If $\alpha \in \operatorname{im}(\iota^{b,N})$, then we say that α "never dies" and is assigned a death coordinate of ∞. In practice, we replace ∞ with i_N. The *lifespan* of α, $l = d - b$, is the difference in these birth

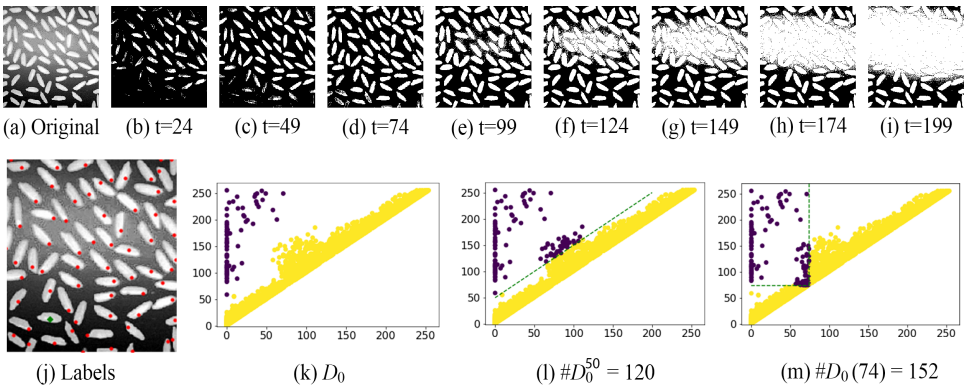

FIGURE 2. Motivational example of sublevel set filtration and the corresponding persistence diagram. (a) The original grayscale image f. (b)-(i) Sublevel set of f at different values of threshold t. We color the sublevel set as white. (j) Labels for the rice grains. We used `GUDHI`[9] to calculate persistence and track generators. Red points are the generators corresponding to the purple points in (k), and the green one is the infinite generator in (k). (k) 0-dimensional persistence diagram, D_0. The purple points correspond to features with red labels in (j). (l) Illustration of the lifespan cutoff with selected points in purple. (m) Illustration of the PD Thresholding method with selected points in purple.

and death times. The collection of birth and death time pairs for each feature in the sequence is called a *persistence diagram*.

2.4. **Persistence diagrams and feature extraction.** In this work, our main application is measuring topological structure in image data. We view an 8-bit grayscale image as a function, i.e. $f : P \to \{0, 1, 2, \cdots, 255\}$ where P is a collection of pixels or voxels. There are some ways to create a filtration associated with f [13, 21]. We adopt the widely-used sublevel set filtration. Formally, given a threshold t, the corresponding sublevel set of f is defined as $f_t^- := \{p \in P \mid f(p) \leq t\}$. By construction, f_t^- is a cubical set (see [16]) and these sublevel sets form a filtration

$$f_0^- \subseteq f_1^- \subseteq f_2^- \subseteq \cdots \subseteq f_{255}^-.$$

For visualization purposes, we use white to represent the sublevel set. (Note that to make this consistent with the usual interpretation of grayscale values where 0 is black and 255 is white, one would first need to invert the scale.)

For illustration, we consider the rice grain image shown in Figure 2(a). This is a grayscale image with noise in the grayscale values and illumination levels that vary across the image, two of the challenges that we also see in firn micro-CT data. Figure 2(b)-(i) illustrate the sublevel sets at different threshold values. Figure 2(k) shows the 0-dimensional persistence diagram, denoted by D_0, for the sublevel set filtration of the rice image. Visually, one may count directly that there are 79 grains of rice in Figure 2(a), shown tagged with a red or green dot in Figure 2(j). D_0 contains the information about connected components in sublevel sets f_t, which include the grains of rice as well as additional components due to noise in the

image. It is our goal to construct an algorithm to extract those 79 features from the persistence diagram; more precisely the purple (darker) points in Figure 2(k).

Remark 1. It is important to note the possibility that an object in an image may have several corresponding homology generators depicted as multiple points in the persistence diagram. It is our goal to, for each object, select the longest-life generator of that object that exists in the diagram. We refer to these particular generators as robust generators (represented by purple or dark points in our figures), while points not selected by these methods are called "noise" and are represented by yellow or light colored points in our figures.

The most common way to extract presumably robust features is to apply a lifespan cutoff L to D_0. More precisely, consider $D_0^L := \{(b,d) \in D_0 \mid d-b > L\}$. The intuition is that long lifespan points in D_0 are likely the robust/true features. As shown in Figure 2(l), the green dotted line represents the lifespan cutoff $d-b=50$ with points above it labeled as features, while points below it are labeled as noise. One may count that $\#D_0^{50} = 120 \neq 79$. In fact, as can be seen in Figure 2(l), no matter which lifespan cutoff L we use, D_0^L never yields the desired set (purple points in Figure 2(k)). The main difficulty is that there is no way to avoid those noisy generators located in the region $[90,150]\times[90,150]$ while capturing all of the desired points. Thus, a linear cutoff of this type would not work.

Another method for extracting features is the PD Thresholding method [7], which was designed to automatically threshold a grayscale image to produce a binary image. The essential idea in PD Thresholding is to scan over the fundamental boxes [10], defined as $D_0(t) = \{(b,d) \in D_0 \mid b \leq t, \text{and, } d > t\}$, for all $t \in \{0,1,2,\cdots,255\}$, trying to choose a value of t that maximizes the containment of long lifespan features while minimizing the number of short lifespan points. For instance, $D_0(74)$ is depicted as the region enclosed by the green dotted line in Figure 2(m). Applying PD thresholding to the rice image, returns a threshold of 74. Thus, one could use points in $D_0(74)$ as features. We observe $\#D_0(74) = 152 > 120 > 79$ yields an even larger overestimate of the features than the lifespan cutoff approach. This over counting is a common problem for PD Thresholding when short lifespan generators appear to be spread out along the diagonal and are therefore unavoidable when optimizing over fundamental boxes [7]. Note also that the spread of true features (purple points in Figure 2(k)) in the persistence diagram also prevents any one fundamental box from capturing all desired features. Essentially, varying illumination in the image prevents any one global threshold from capturing every grain of rice.

At this point, we have seen that both lifespan cutoff and PD Thresholding run into trouble for the rice grain example, and, by extension, similar noisy grayscale images with varying illumination. This is often the case for our motivating example of firn, which we study in Section 4. We, therefore, develop the more flexible approach in Section 3 that is the focus of this work.

3. DPFD and empirical results.

As seen in Section 2, persistence diagrams arising from real data contain points corresponding to features that we are trying to measure, as well as points that are due to noise. The rice grain example demonstrates the somewhat standard heuristic that true features tend to have longer lifespans and points generated by noise tend to form clusters that sit close to the diagonal. Figure 2 also shows that partitioning these points by lifespan (using a

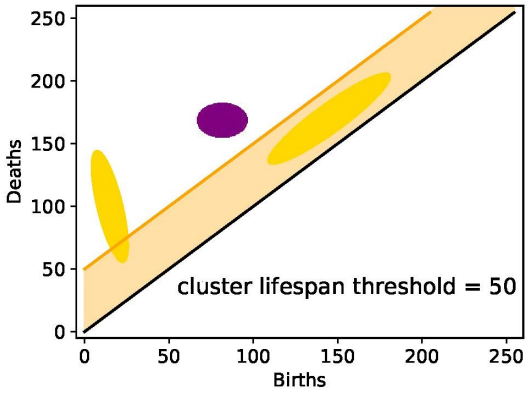

FIGURE 3. Illustration of the DPFD method. The yellow ellipses represent clusters that are labelled to be noise, while the purple cluster is labelled as features since its minimum lifespan is above the designated cluster cutoff.

lifespan cutoff) or by threshold (as in PD Thresholding) may not produce accurate results. Instead, we need an approach with the ability to identify the shape of true feature (purple) points versus noisy feature (yellow) points that we see in Figure 2(k) and shown conceptually in Figure 3.

Towards this goal, we present the Density-based Persistent Feature Detection (DPFD) method, so named for its application of the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [11] algorithm for density-based clustering, to persistence diagrams. Conceptually, we seek to use density-based techniques to identify clusters in a persistence diagram and then, using a lifespan cut off parameter, label clusters as either "noise" or "features." This reflects the observation that noise tends to be clustered in the persistence diagram and in clusters that include points with short lifespans. While short lifespan features may be removed via a lifespan cutoff, it is possible, as indicated in Figure 5, that diagram points with longer lifespans should also be considered noise. DPFD classifies these points as noise by taking into account the proximity of these points to a noisy cluster.

The heart of DPFD is the DBSCAN clustering algorithm. To account for complexity and noise in the persistence diagram, we first run DBSCAN on points in the persistence diagram and retrieve a labeling indicating clusters and isolated points. DBSCAN requires two parameters: ε, a cluster radius within which points are considered neighbors, and M, the minimum number of points required inside such a radius in order to form a cluster. A natural equivalence relation is formed given these two parameters: two points are in the same cluster if they are connected by a chain of points that each have M points within ε distance. The outputs of DBSCAN are isolated points and clusters. Since short lifespan generators tend to be far more prevalent in diagrams, important long lifespan features are typically returned as outliers by DBSCAN.

Finally, to classify features and noise, the DPFD method lifts the idea of a lifespan cutoff from a pointwise application to a "clusterwise" application. After extracting the isolated points and clusters from DBSCAN, DPFD computes the minimum

Algorithm 1 The Density-based Persistent Feature Detection (DPFD) method.

Input
- D persistence diagram
- L_0 cluster relabel lifespan cutoff
- ε cluster radius
- M minimum points to cluster

Output
- $\mathcal{F}$ set of features of the diagram

(isolated points $\mathcal{I}$, clusters $\mathcal{C}$) $\leftarrow$ DBSCAN(D, ε, M)
features $\mathcal{F} \leftarrow \mathcal{I}$
for $C \in \mathcal{C}$ **do** $m \leftarrow \min_{(b,d)\in C} d - b$
  if $m > L_0$ **then**
    $\mathcal{F} \leftarrow \mathcal{F} \cup C$
  end if
end for
for $(b, d) \in \mathcal{F}$ **do**
  if $d - b < L_0$ **then**
    $\mathcal{F} \leftarrow \mathcal{F} \setminus \{(b, d)\}$
  end if
end for

lifespan, $m(C) = \min_{(b,d)\in C}(d - b)$ of each cluster C. Then the cutoff is applied: if $m(C)$ is less than some inputted cutoff L_0, the cluster is labeled as noise; otherwise, the entire cluster lies beyond the short-lived region and is labeled as features. We repeat the cutoff application for each isolated point. This outputs a list of diagram points we call features that have long lifespans and are far away from noisy clusters. Figure 3 provides a conceptual illustration of the DPFD method while Algorithm 1 provides the corresponding pseudocode.

DPFD has three parameters: ε and M, the parameters for DBSCAN, and L_0, the cluster labeling cutoff. The DBSCAN parameters depend solely on the scale of the persistence diagram. In our application of DPFD, we introduce an auxiliary parameter for determining ε based on the scale of the diagram. Specifically, the parameter is a quantile q of pairwise distances of points in the diagram. In our implementation we fix the lifespan cutoff $L_0 = 50$, and perform a grid search for the optimal pair of (q, M). To test this approach, we first applied it to an image of rice where we set the potential values for q as multiples of 0.03 up to 0.3 and the potential values for M as multiples of 10 up to 200. Figure 4 shows the result of DPFD with parameters $M = 20$, $q = 0.1$, and $L_0 = 50$ for the original rice grain example, showing that this approach has the flexibility to accurately partition the points. By comparison, using a lifespan cut-off and optimizing for accuracy in the count of 79 features results in the labeling shown in Figure 5, showing that the lifespan cutoff approach is not able to simultaneously detect all rice grain features.

From here, we set up an experiment to gauge the robustness of the (q, M) parameter pair. The true feature set (purple points) output by DPFD with $M = 20$, $q = 0.1$, and $L_0 = 50$ is the set containing the longest lifespan generator for each grain of rice (in this example, there is only one such generator for each grain of rice), as shown in Figure 4. We, therefore, designate the corresponding labeling of

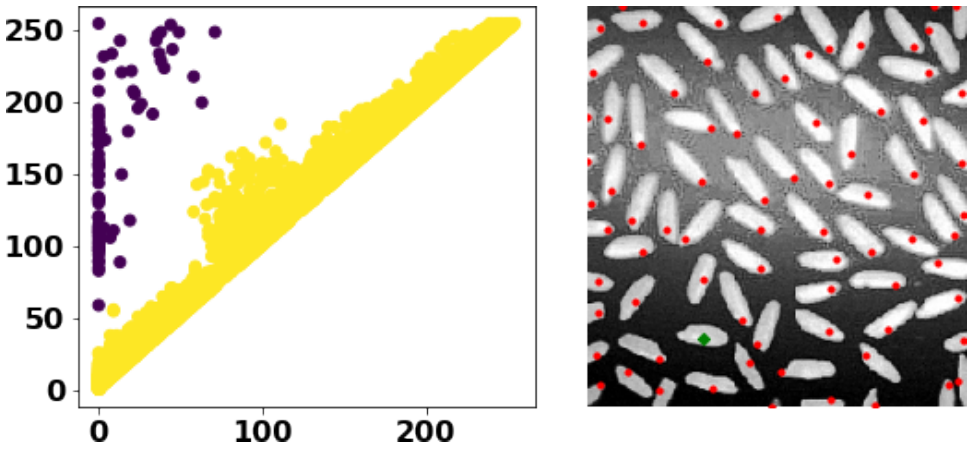

FIGURE 4. Application of DPFD to the rice example shown in Figure 2. The parameters are $M = 20$, $q = 0.1$, and $L_0 = 50$. Purple (darker) points are detected by the DPFD, and the exact 79 features are detected.

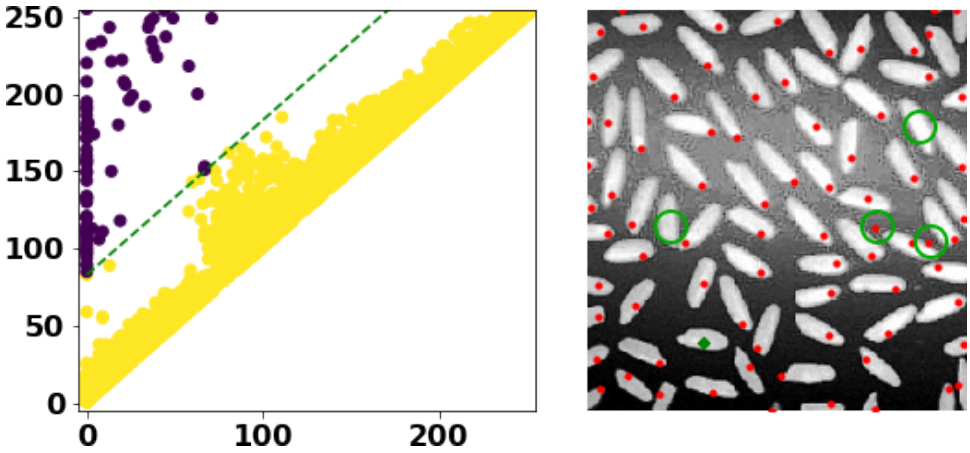

FIGURE 5. Application of lifespan cutoff to the rice example shown in Figure 2. Purple (darker) points are D_0^{83}. $\#D_0^{83} = 78$. Missing and mislabeled features are highlighted by green circles.

the persistence diagram points by feature/not feature as the 'optimal labeling' and compare the results given by DPFD for other (q, M) pairs. Due to the imbalanced data (79 features and 4540 noise), we measure the performance of (q, M) pairs by using precision and recall. Here, precision is the proportion of predicted features that were actually true features and recall is the proportion of the true features that are detected as features. For both, a higher value is a better score. Figure 6 shows two plots of (q, M) pairings colored separately by the precision and recall scores of their labeling. One immediate point of interest is the apparent robustness of the M parameter. A change in this value over the computed range does not seem to affect model performance much. On the other hand, DPFD seems a bit more sensitive to

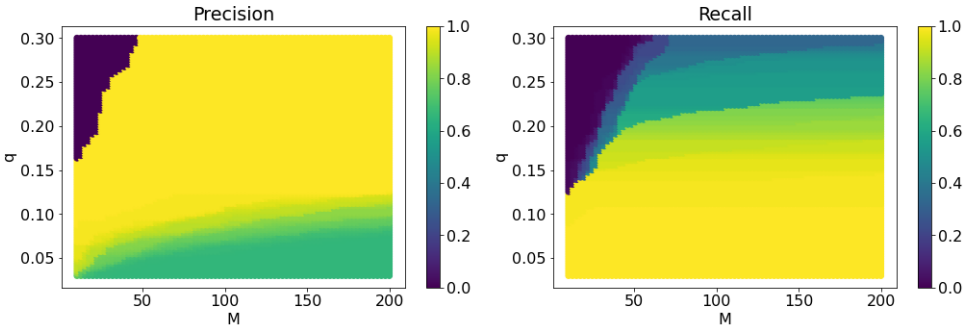

FIGURE 6. Precision and Recall plots for various (q, M) pairs.

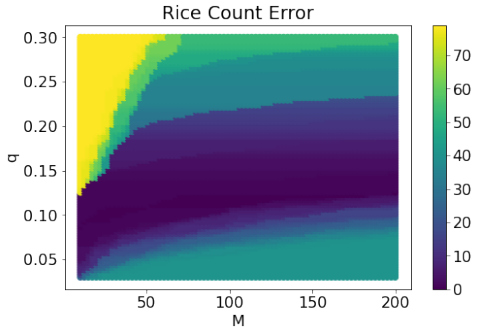

FIGURE 7. Rice grain count errors for various (q, M) pairs.

the q parameter over the computed range. This may largely be caused by the fact that when M is too low and q is high, fewer clusters are formed making it more likely that true features will be clustered with noise. We also implement a second scoring metric by way of the predicted number of rice grains. Figure 7 shows the relevant plot. Here again we can see the robustness of the approach with respect to the M and q parameters.

4. **Application to firn and its impact.** Given the experimental success of DPFD for identifying rice grains, we move now to our motivating application of quantifying ice and pore-space regions in firn images.

Our data consists of Micro-CT images of firn samples taken from Summit, Greenland. For each depth, we selected 3 images from each reconstructed stack – the 300, 450, and 600th slice, giving a total of 21 images. These images were subsequently analyzed by an expert who counted the number of contiguous white regions (ice) and the number of contiguous black regions (pore-space). We used this data to train and test two models. Specifically, our training set consists of the 300th and 600th slices from each depth, the testing set contains only the 450th slice from each depth, and we know the counts of the number of features of each type but not the correct labeling of points in the persistence diagrams. We will, therefore, use mean squared error on the counts of features rather than the precision and recall errors computed in the rice example. Our goal is to provide a count of ice regions and pore-space regions. Finally, we use mean squared error to score these models. Before building

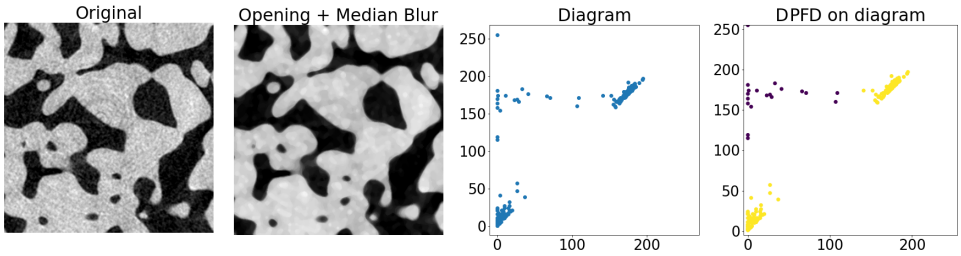

FIGURE 8. The process for the DPFD model. From left to right we have the original image, the image resulting from the application of median blur and the morphological opening, the persistence diagram of the preprocessed image, and finally the application of DPFD.

our models, we remove small scale features from the images, including sample dust, by applying the morphological operation of opening using a 7×7 structuring element followed by a median blur filter with a 9×9 kernel. A sample original image and that same image following this preprocessing step are shown in Figure 8 and are consistent with the domain expert's view of the images.

The DPFD model is based on the DPFD algorithm. Recall that for this algorithm, we have three parameters: the cluster radius (ϵ), the minimum points to cluster (M) and the lifespan cutoff (L_0). In our model, we decided to select ϵ based on the input diagram by taking a specific quantile q of the pairwise distances of the points in the diagram (excluding the distances from the points to themselves). Moreover, for the DPFD model, we fixed $L_0 = 50$. Finally, for each depth, we selected the optimal M and q to minimize the training error, that is, minimizing the mean squared error for the count of features on that depth. Thus, the DPFD model is a collection of models that have varying M and q parameters based on depth. Figure 8 gives an example of the process for each image. On the left is the original image. Figure 9 shows the plots of the q and M parameters that were considered optimal by the DPFD model. Perhaps most notable is the variance of the parameters over depth. There are several possible reasons for this, but perhaps most likely is the small size of the training set at just 2 images per depth. Figure 10 shows an example of DPFD applied to one of the firn test images along with the tagged ice regions based on the diagram points detected by the DPFD. Notably here, the DPFD model has accurately tagged all of the relevant ice regions in this image. Finally, Figure 11 shows our worst performance on a test image. We see in this figure that the ice is still properly tagged, however, it has been tagged three times.

The Lifespan model was built similarly as a collection of models. This model was based on lifespans. In particular, we had one parameter (lifespan cutoff) that counted features with a lifespan above the given cutoff. Tuning this parameter for each depth was again accomplished through a simple grid search. In firn, as in the rice example, there are situations where no lifespan threshold will work to accurately count and tag features in an image. The diagram in Figure 10 is an example of such a situation.

We applied both models to the separate tasks of ice regions and pore-space regions. After tuning the parameters for each depth and for each model, we saw

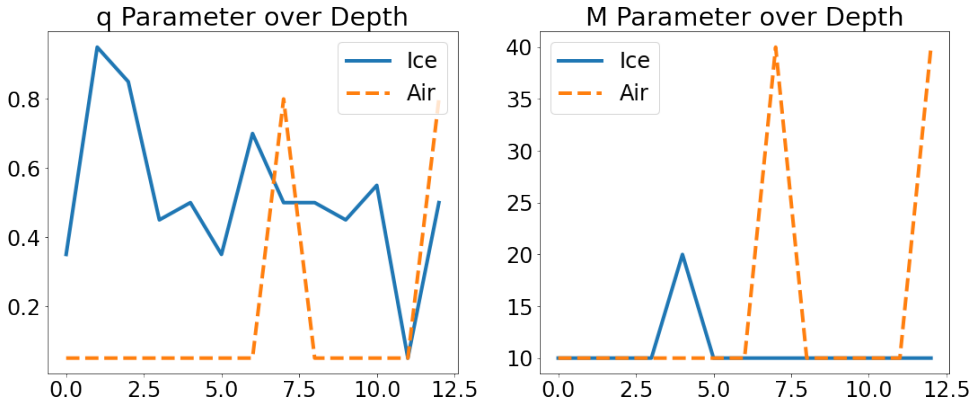

FIGURE 9. A plot of the computed optimal q and M over depth

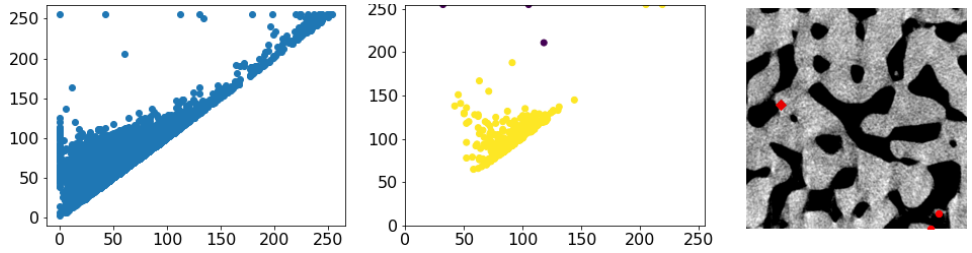

FIGURE 10. An example of DPFD on a firn image. The image is tagged based on the selected diagram points. From left to right are: the persistence diagram computed on the original image, the persistence diagram computed on the preprocessed image, and the original image tagged by points corresponding to the features identified by DPFD.

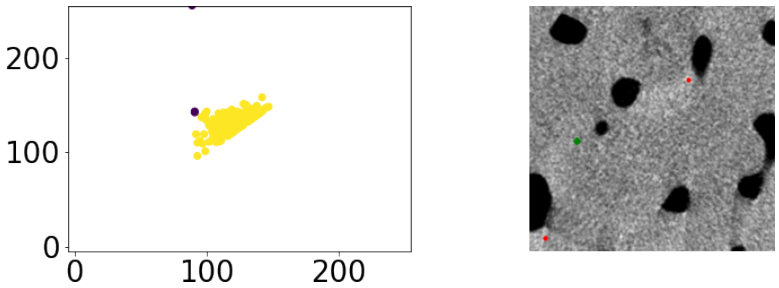

FIGURE 11. An example of DPFD over-labelling ice on a firn image

that the DPFD model obtained an average test error of 0.5 for counting ice regions and 0.6 for counting pore-space regions. The Lifespan model obtained errors of 0.6 and 0.9 for ice and air respectively. We did not consider PD Thresholding as a viable model due to its tendency to overcount by identifying points near the diagonal as features. Overall, the DPFD model had a lower error on average that the Lifespan model in counting both ice and pore-space regions. The right plot in Figures 12

and 13 illustrate the test error at each depth, where the true values were counted and reported by an expert.

Finally, with the DPFD model trained, we applied it to all depths and images from the set of Summit, Greenland firn samples to count both the ice and pore-space regions. Figures 12 and 13 show the average estimates of the number of contiguous ice regions and the number of contiguous pore-space regions, respectively. As expected, we find that the number of contiguous ice regions decreases with depth as the firn becomes more dense. As firn densification proceeds, disconnected ice particles (e.g., Figures 1a, h, o) merge into fewer distinct ice particles with depth until a single ice-matrix is formed around the pores (e.g., Figures 1g, n, u). We also find that the estimated number of pore-space regions increases, with a sharp rise below 23 m depth, and continues to remain large with depth, as expected. During firn densification the large, interconnected pore-space closes off into individual pores as the ice matrix grows. While the size of the individual pores continues to decrease with depth due to densification, the number of the pores remains fairly constant, which we see in the results.

The trends in the estimated values of both the number of ice regions and the number of pore-space regions match that of the true values including the rise in values between 23 and 26 m depth (Figure 12, Figure 13). This sharp rise in values is potentially a result of the shift between pore-space dominated (e.g., Figures 1a, h, o) and ice dominated images (e.g., Figure 1b, i, p) around 23 m depth at this site. Considering the estimated values over the full range of depths, both models overestimate the number of ice regions at shallow depths, as well as the number of pore-space regions at deeper depths. This is most likely due to small scale features that remain following the minimal preprocessing performed on the images. Overall, the DPFD model better estimates both the ice and pore-space regions at all depths compared to the Lifespan model.

These results are promising, showing that we can accurately identify the correct number of ice regions and internal pore spaces in the Micro-CT firn images. Improvements in firn-densification models, which are critical for improved ice-core paleoclimate records, require incorporation of firn microstructural information [20]. This is no easy task, as our best tool for retrieving microstructure data is the Micro-CT, which produces terabytes of data for each firn core that is three-dimensionally reconstructed. Therefore, finding automatic ways to accurately detect basic microstructure values, such as the Betti numbers, is a crucial step in developing the next generation of firn densification models.

5. **Discussion and conclusion.** Motivated by the observation that persistence diagram points corresponding to noise tend to be clustered near the diagonal, we develop DPFD, a density-based method for extracting features from persistence diagrams. On the empirical rice grain example, DPFD outperforms both lifespan cutoff and PD Thresholding methods. We verified the accuracy of DPFD in the rice example, where tagging features classified as true features allows us to check that they actually correspond to grains of rice. Moreover, we tested the robustness of the parameters for DFPD and found a wide range of acceptable parameters. For the firn example, we tested both DPFD and lifespan cutoff, and DPFD produced smaller model errors. We also demonstrate a training method to select the parameters required for DPFD given benchmark counts of features, in our example given by

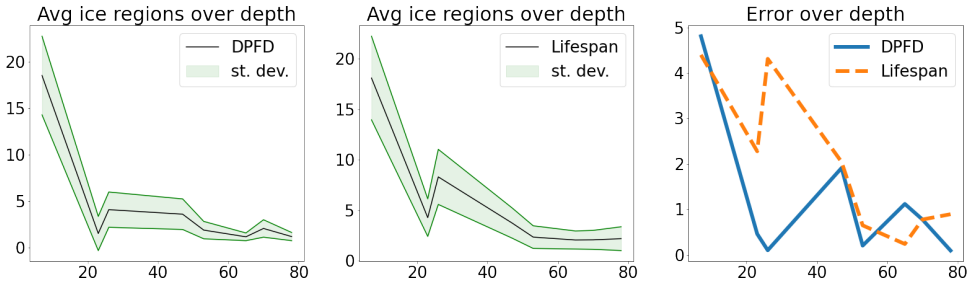

FIGURE 12. *Left*: The average count of ice regions estimated by the DPFD model with a band of one standard deviation in green as a function of depth. *Middle*: The average count of ice regions estimated by the Lifespan model with a band of one standard deviation in green as a function of depth. *Right*: The difference between the true number of ice regions in the test images, as defined by an expert, and the estimates given by the DPFD and Lifespan models.

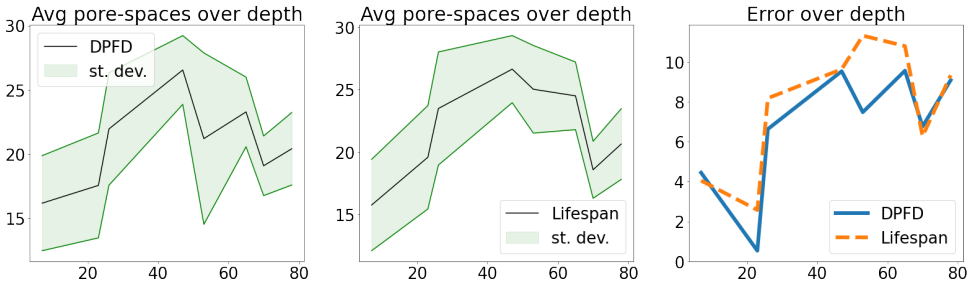

FIGURE 13. *Left*: The average count of pore-space regions estimated by the DPFD model with a band of one standard deviation in green as a function of depth. *Middle*: The average count of pore-space regions estimated by the Lifespan model with a band of one standard deviation in green as a function of depth. These values represent the number of contiguous pore-space regions with depth. *Right*: The difference between the true number of pore-space regions in the test images, as defined by an expert, and the estimates given by the DPFD and Lifespan models.

the domain expert and visual calculation. We conclude this paper by outlining several potential future directions.

First, it would be interesting to investigate the stability of the method. Stability in this context means that small perturbation of the data leads to a small change in the estimated Betti numbers which count the number of points labelled as features. Stability of the persistence diagrams has been studied and proven [5, 8] under certain reasonable classes of perturbations. As mentioned in the preprints [15, 19], the DBSCAN method can be viewed as a Vietoris–Rips complex. It would be interesting to combine results in [5, 8] and to investigate the stability of the DPFD method.

Second, as discussed in [1], studying the distribution of short lifespan generators in persistence diagrams is a challenging task. The core of the DPFD method is to

extract the features. One may use DPFD as a medium to isolate the noisy generators (e.g. the yellow points in Figure 2(k)), and further investigate the statistical properties of the points generated by noise.

Third, in terms of the application, the current paper concerns 2D slices of the firn data. Our next project is to study 3D images of firn using DPFD. An important aspect of firn modeling is determining the depths at which pores close off into individual bubbles, because this is where atmospheric samples get trapped into the layers of glacial ice. Currently, this depth of pore close-off is approximated using the current-generation firn densification models, which are known to perform poorly at some sites [18]. Studying our 3D firn data in a similar method will allow us to track cavities as a function of depth and providing more accurate estimates of where bubbles form in the firn column. Ultimately, this will improve climate scientists' estimates of the timing of abrupt climate events from ice core records. One of the practical issues for 3D cubical persistence diagram computation is memory usage [22]. Recent advances in software[1] allow for the efficient computation of larger 3D images. We plan to combine this new software and our DPFD approach to study topological features of 3D firn images as a means of better understanding firn microstructure.

REFERENCES

[1] R. J. Adler, O. Bobrowski, M. S. Borman, E. Subag and S. Weinberger, Persistent homology for random fields and complexes, in *Borrowing Strength: Theory Powering Applications–A Festschrift for Lawrence D. Brown*, Inst. Math. Stat. (IMS) Collect., 6, Inst. Math. Statist., Beachwood, OH, 2010, 124–143.

[2] N. Atienza, R. Gonzalez-Diaz and M. Rucco, Persistent entropy for separating topological features from noise in vietoris-rips complexes, *J. Intelligent Information Systems*, **52** (2019), 637–655.

[3] N. Atienza, R. Gonzalez-Diaz and M. Rucco, Separating topological noise from features using persistent entropy, in *Software Technologies: Applications and Foundations*, Lecture Notes in Computer Science, 9946, Springer, Cham, 2016, 3–12.

[4] G. Carlsson, Topology and data, *Bull. Amer. Math. Soc. (N.S.)*, **46** (2009), 255–308.

[5] F. Chazal, V. de Silva and S. Oudot, Persistence stability for geometric complexes, *Geom. Dedicata*, **173** (2014), 193–214.

[6] F. Chazal and B. Michel, An introduction to Topological Data Analysis: Fundamental and practical aspects for data scientists, preprint, arXiv:1710.04019.

[7] Y.-M. Chung and S. Day, Topological fidelity and image thresholding: A persistent homology approach, *J. Math. Imaging Vision*, **60** (2018), 1167–1179.

[8] D. Cohen-Steiner, H. Edelsbrunner and J. Harer, Stability of persistence diagrams, *Discrete Comput. Geom.*, **37** (2007), 103–120.

[9] P. Dlotko, Cubical complex, in *GUDHI User and Reference Manual*, GUDHI Editorial Board, 3.3.0 edition, 2020. Available from: https://gudhi.inria.fr/.

[10] H. Edelsbrunner and J. L. Harer, *Computational Topology. An Introduction*, American Mathematical Society, Providence, RI, 2010.

[11] M. Ester, H.-P. Kriegel, J. Sander and X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in *KDD-96 Proceedings*, 1996, 226–231.

[12] B. T. Fasy, F. Lecci, A. Rinaldo, L. Wasserman, S. Balakrishnan and A. Singh, Confidence sets for persistence diagrams, *Ann. Statist.*, **42** (2014), 2301–2339.

[13] A. Garin and G. Tauzin, A topological "reading" lesson: Classification of MNIST using TDA, 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA), Boca Raton, FL, 2019, 1551–1556.

[14] R. Ghrist, Barcodes: The persistent topology of data, *Bull. Amer. Math. Soc.*, **45** (2008), 61–75.

[15] J. F. Jardine, Data and homotopy types, preprint, arXiv:1908.06323.

[1] https://bitbucket.org/hubwag/cubicle/src/master/Readme.md

[16] T. Kaczynski, K. Mischaikow and M. Mrozek, *Computational Homology*, Applied Mathematical Sciences, 157, Springer-Verlag, New York, 2004.
[17] K. Keegan, M. R. Albert and I. Baker, The impact of ice layers on gas transport through firn at the North Greenland Eemian Ice Drilling (NEEM) site, Greenland, *The Cryosphere*, **8** (2014), 1801–1806.
[18] A. Landais, J. M. Barnola, K. Kawamura, N. Caillon and M. Delmotte, et al., Firn-air δ15n in modern polar sites and glacial–interglacial ice: A model-data mismatch during glacial periods in Antarctica?, *Quaternary Science Reviews*, **25** (2006), 49–62.
[19] M. Lesnick and M. Wright, Interactive visualization of 2-D persistence modules, preprint, arXiv:1512.00180.
[20] J. M. D. Lundin, C. M. Stevens, R. Arthern, C. Buizert and A. Orsi, et al., Firn model intercomparison experiment (FirnMICE), *J. Glaciology*, **63** (2017), 401–422.
[21] I. Obayashi, Y. Hiraoka and M. Kimura, Persistence diagrams with linear machine learning models, *J. Appl. Comput. Topol.*, **1** (2018), 421–449.
[22] N. Otter, M. A. Porter, U. Tillmann, P. Grindrod and H. A. Harrington, A roadmap for the computation of persistent homology, *EPJ Data Science*, **6** (2017).
[23] A. Patania, F. Vaccarino and G. Petri, Topological analysis of data, *EPJ Data Science*, **6** (2017).
[24] J. Schwander and B. Stauffer, Age difference between polar ice and the air trapped in its bubbles, *Nature*, **311** (1984), 45–47.
[25] L. Wasserman, Topological data analysis, *Annu. Rev. Stat. Appl.*, **5** (2018), 501–535.
[26] A. Zomorodian, Topological data analysis, in *Advances in Applied and Computational Topology*, Proc. Sympos. Appl. Math., 70, Amer. Math. Soc., Providence, RI, 2012, 1–39.

Received January 2021; revised April 2021.

E-mail address: azlawson@uncg.edu
E-mail address: thoffman@umd.edu
E-mail address: y_chung2@uncg.edu
E-mail address: kkeegan@unr.edu
E-mail address: sldayx@wm.edu