

**NIST Technical Note
NIST TN 2349**

**Topological Initialization for
Multidimensional Scaling**

Melinda A. Kleczynski
Anthony J. Kearsley

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2349>

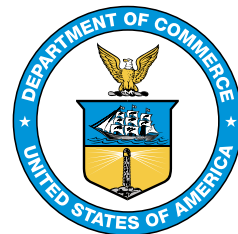
**NIST Technical Note
NIST TN 2349**

**Topological Initialization for
Multidimensional Scaling**

Melinda A. Kleczynski
Anthony J. Kearsley
*Information Technology Laboratory
Applied and Computational Mathematics Division*

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2349>

September 2025



U.S. Department of Commerce
Howard Lutnick, Secretary

National Institute of Standards and Technology
Craig Burkhardt, Acting Under Secretary of Commerce for Standards and Technology and Acting NIST Director

Certain equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

NIST Technical Series Policies

[Copyright, Use, and Licensing Statements](#)

[NIST Technical Series Publication Identifier Syntax](#)

Publication History

Approved by the NIST Editorial Review Board on 2025-04-28

How to cite this NIST Technical Series Publication:

Kleczynski MA, Kearsley AJ (2025) Topological Initialization for Multidimensional Scaling. (National Institute of Standards and Technology, Gaithersburg, MD), NIST TN 2349. <https://doi.org/10.6028/NIST.TN.2349>

Author ORCID iDs

Melinda A. Kleczynski: 0009-0003-6923-8455

Anthony J. Kearsley: 0000-0002-9576-0621

Contact Information

anthony.kearsley@nist.gov

Abstract

Multidimensional scaling is a popular technique for visualizing dissimilarities between objects in complex datasets. For a single dataset, different initial configurations for multidimensional scaling may produce representations with qualitatively different features. This presents both challenges and opportunities for those who use these methods. We introduce the new technique Topological Initialization for Multidimensional Scaling (TIMDS) which employs cycle representatives, a tool from topological data analysis, to generate multidimensional scaling initializations. We show that for some datasets, TIMDS produces representations with competitive stress values and better visualization of key dataset attributes compared to random or classical initialization methods.

Keywords

Cycle representatives; multidimensional scaling; topological data analysis.

Table of Contents

1. Introduction	1
2. Topological Data Analysis Background	2
3. Methods	3
3.1. Data	3
3.2. Software	5
3.3. Multidimensional Scaling	5
3.4. Topological Initialization for Multidimensional Scaling	6
3.5. Angle Prediction	7
4. Results	7
4.1. Cat Food	8
4.2. Toy Duck	9
5. Discussion and Future Work	13
6. Conclusion	14
7. Code Availability	14
References	15

List of Tables

Table 1. The rotating image datasets included in the analysis.	5
Table 2. Iterations required and stress obtained for MDS starting at different initial configurations.	8
Table 3. Comparison of MDS stress after topological initialization and MDS stress after PCA/PCoA initialization.	8

List of Figures

Fig. 1. Simplicial complexes and cycle representative generated by performing topological data analysis of the cheeseburger dataset.	3
Fig. 2. The dimension 1 barcode from the cheeseburger dataset and the barcode from an MDS representation of the dataset.	4
Fig. 3. The dimension 1 barcode of the cat food dataset; the cycle representative corresponding to the longest interval is used for performing TIMDS.	9
Fig. 4. MDS representations of the cat food dataset obtained from random initial configurations.	10
Fig. 5. MDS representations of the cat food dataset obtained using PCoA coordinates or topological initialization.	11

Fig. 6. The dimension 1 barcode of the toy duck dataset; the cycle representative corresponding to the longest interval is used for performing TIMDS.	12
Fig. 7. MDS representations of the toy duck dataset obtained using different initial configurations.	12
Fig. 8. Least squares circle angle versus the actual object rotation for initialization using a random initial configuration, PCA, and topological initialization.	13

Acknowledgments

This research was performed while Melinda A. Kleczynski held a National Institute of Standards and Technology (NIST) National Research Council (NRC) Research Postdoctoral Associateship Award at the NIST Applied and Computational Mathematics Division under the supervision of Anthony J. Kearsley.

Author Contributions

Melinda A. Kleczynski: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Anthony J. Kearsley:** Conceptualization, Formal analysis, Investigation, Methodology, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

1. Introduction

Many modern datasets are difficult to visualize, because the data are high-dimensional and/or non-Euclidean. A common strategy is to obtain a reasonable representation of the data in a low-dimensional Euclidean space. Representations in 2-dimensional space are especially useful for visualization. In this work, we analyze images with corresponding array sizes $128 \times 128 \times 3$ or 128×128 . This results in an unwieldy 49,152-dimensional or 16,384-dimensional space which we can reduce to two dimensions while still retaining important structure.

Multidimensional scaling (MDS) [1, 2] is a widely used technique which attempts to produce points in a (typically low-dimensional) Euclidean space whose pairwise distances approximate the pairwise dissimilarities between the original data points. Performing MDS involves minimizing some quantity, frequently stress [3]. If δ_{ij} is the original dissimilarity between data points i and j and d_{ij} is the Euclidean distance between the corresponding points in the target low-dimensional space, then we seek to minimize the stress,

$$\sum_{i < j} (d_{ij} - \delta_{ij})^2.$$

In practice, it is standard to obtain configurations with low stress through iterative optimization. The goal is to return configurations which are near local minimizers, a process which may be sensitive to the starting point used for optimization [4]. There may be several local minimizers with similar stress values and qualitatively different MDS representations; this makes consideration of multiple initial configurations valuable for visualization [5]. There is interest in considering MDS configurations which have slightly higher stress than the lowest-stress representation identified, but better represent the underlying structure of the dataset ([6, 7]).

In this work, we are particularly interested in datasets whose points are approximately parameterized by a single periodic variable. In some cases, standard MDS pipelines may effectively reveal the circular structure of a dataset. For example, MDS representations of some biological datasets can have a circular structure driven by the cell cycle [8]. For other datasets, such as the ones considered in this work, MDS using default initializations may struggle to capture known properties, such as dependence on a rotation angle.

Structures such as loops and tunnels are readily detectable using topological data analysis (TDA) [9], raising the possibility that we may be able to use techniques from applied topology to improve MDS results. MDS has frequently been used in combination with TDA, but often in the other order, with performing MDS as one of the initial steps in TDA pipelines. This has produced successful results for analyzing data from single-cell RNA sequencing [10], cardiac resynchronization therapy patients [11], and the human gut microbiome [12]. MDS has also been applied to visualize completed topological summaries to study tumor microenvironments [13], chaotic versus periodic dynamics [14], and antibody conformations [15].

In this work, we introduce a new method for MDS initialization based on cycle representatives, a tool from topological data analysis. The general approach of using structural information about the dataset to obtain a strategic initial configuration is known as weak confirmatory MDS (see Sec. 6.1 of Ref. [2]). Our method, Topological Initialization for Multidimensional Scaling (TIMDS), has the potential to increase the utility of MDS for some datasets.

2. Topological Data Analysis Background

We begin with a brief overview of the relevant techniques from TDA. A thorough discussion of TDA is beyond the scope of this work, but can be found in other texts [16–18]. We restrict our attention to analysis of finite sets of data points equipped with pairwise distances between points. We consider only Euclidean or cosine distances, but more general distances or dissimilarities can also be used for topological analysis. Given such a dataset and a fixed distance value ε , we form a simplicial complex by connecting sets of data points with pairwise distances at most ε . This is a particular type of simplicial complex called a Vietoris–Rips complex [19]; as this is the only type of simplicial complex used in the current work, we can use the more general term without ambiguity. Examples of simplicial complexes are shown in Fig. 1. These simplicial complexes capture structural features of the dataset, in this case images of a rotating cheeseburger. In this case, each simplicial complex encircles a single empty region, reflecting the fact that the data points are arranged in a loop. A cycle representative is an element of a class of cycles from a simplicial complex. The cycle representative shown in Fig. 1 is one of a class of cycles surrounding the empty region. Note that some authors may refer to cycle representatives as homological generators [20].

An important consideration when using cycle representatives is that they are not unique. This is frequently addressed through the use of optimal cycle representatives [20] rather than arbitrary cycle representatives. In the current work we use the default cycle representatives returned by Open Applied Topology (OAT)¹ [21]. Determining the cycle optimization criteria which best improve the performance of TIMDS would be an interesting topic for future research.

A key strategy in topological data analysis is to consider the persistence of features with respect to ε . This is accomplished through the computation of persistent homology ([22, 23]). Informally, there is a wide interval of values of ε for which the simplicial complex formed from the data in Fig. 1 will enclose some empty center region. The exact intervals corresponding to loop structures at different distance scales are recorded in a topological

¹Certain equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

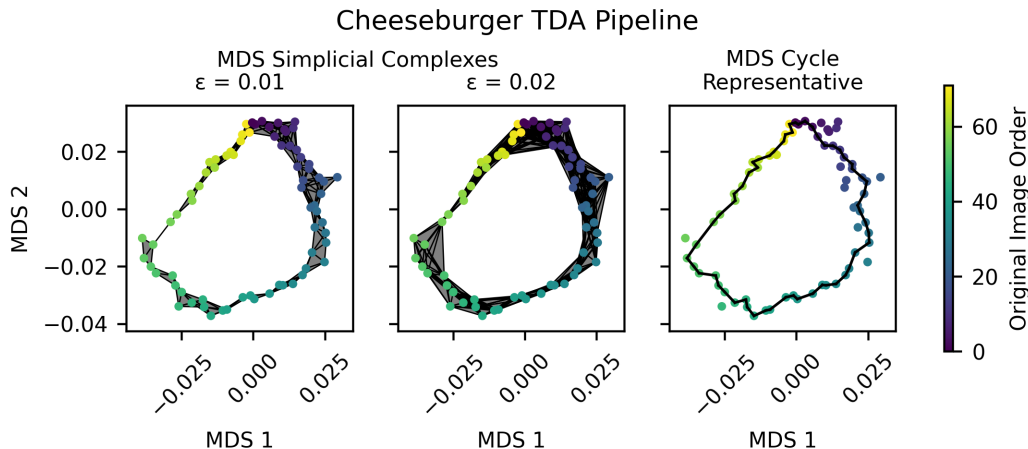


Fig. 1. We form simplicial complexes by connecting sets of points whose pairwise distances are at most ϵ . For data points which are sampled from a closed loop, many choices of ϵ will generate a simplicial complex which encircles an empty center region. A class of cycles surrounds the empty region; a cycle representative is an element of such a class.

summary called a dimension 1 barcode [19]. Simplicial complexes also generate barcodes for other dimensions, but we don't consider them here.

In Fig. 2, we show the dimension 1 barcodes of the cheeseburger dataset and a corresponding MDS representation. Both barcodes have a long interval, or bar. This is an indication of the loop structure of the dataset. For this dataset, MDS with random initialization produces an excellent representation of the circular structure of the dataset. This is supported by the presence of a comparable long interval in both barcodes. We also encounter datasets for which the original data produce a prominent persistent feature in the dimension 1 barcode, but there is no clear circular structure in a typical MDS representation. Our method, TIMDS, may improve the MDS representation in these situations.

3. Methods

In this section, we discuss the data and software used in the current work. We provide the specifications we use for performing MDS. We describe the steps involved in performing TIMDS. Finally, we outline the additional analysis involved in recovering angles from the MDS representation. Similar approaches have been used to relate angles generated by MDS representations of biological data to phases of the cell cycle [8].

3.1. Data

We analyze four datasets from the Columbia Object Image Library (COIL) [24, 25]. The COIL datasets have been used for demonstrating many TDA techniques, such as Mapper [26],

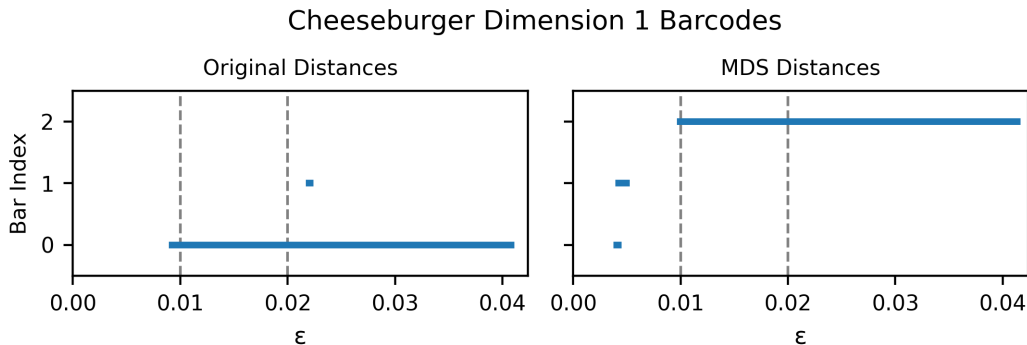


Fig. 2. Barcodes track topological features across values of ϵ . For the cheeseburger dataset, we show a dimension 1 barcode from the original distances between images as well as a barcode from a corresponding MDS representation. Both show a persistent topological feature, corresponding to a long bar in the barcode.

circular coordinate recovery [27, 28], and optimization using topological losses [29–31]. Each dataset consists of 72 images of an everyday object, with the object rotated at a different angle in each image. This can generate circular structure due to the proximity of images with similar rotation angles. The COIL-100 library [24] consists of color images with associated array size $128 \times 128 \times 3$. The COIL-20 (processed) library [25] consists of grayscale images with associated array size 128×128 . For each image, we divide by the maximum value in the corresponding array. Most values in this work are dimensionless quantities obtained from these normalized image arrays, with the exception of angles which are given in radians (rad). We add salt+pepper noise to the grayscale images by creating arrays of uniform random numbers between 0 and 1 with the same size as the image arrays. Image array elements for which the corresponding random number array is less than 0.05 are set to 0, while image array elements for which the corresponding random number array is greater than 0.95 are set to 1.

Overviews of the datasets included in our analysis are given in Table 1. For two datasets we compute cosine distances between images, while for the other dataset we compute Euclidean distances between images. More sophisticated distances may produce better results for image handling, but the simple distances we consider are sufficient for the demonstration of our method. The circular structure of the cheeseburger dataset is already well-represented by MDS with random initialization; we use this dataset for our TDA overview in Fig. 1 and Fig. 2. We perform TIMDS for the remaining datasets.

In the original datasets, the order of the images of a given object corresponds to the rotation of that object. For the cheeseburger and cat food examples, we shuffle the order of the images prior to computing the distance matrices in order to obscure the ground truth image rotation. For the toy duck example, we choose a random sample of the images, with replacement. This hides the original order as well as allowing for a non-uniform sample of

Table 1. The rotating image datasets included in the analysis.

Dataset	Library	Object	Noise	Distance
cheeseburger	COIL-100	73	none	cosine
cat food	COIL-100	29	none	cosine
toy duck	COIL-20	1	salt+pepper	Euclidean

the object rotations. This could destroy the loop structure of the dataset if the angles in the sample don't cover the circle sufficiently. We guard against this possibility by viewing the dimension 1 barcode to check for the existence of a long interval prior to performing TIMDS. We add noise after sampling the images, so that if the same image is selected more than once, the noisy images are still unique. For the toy duck dataset, we explicitly define the object rotation so that a rotation of $-\frac{\pi}{2}$ means the duck is facing the camera, a rotation of 0 means the duck is facing the right side of the image, a rotation of $\frac{\pi}{2}$ means the duck is facing away from the camera, and a rotation of π means the duck is facing the left.

For the cheeseburger and cat food examples, there are 72 images in the datasets, all of which are included in the analysis. In the toy duck example, there are 72 images in the dataset. We sample 100 images, with replacement. 55 unique images are ultimately included in our sample.

3.2. Software

All analysis is performed in Python 3.13.2. For topological data analysis, we use Open Applied Topology (OAT) [21] with Rust backend `oat_rust` [32] accessed through `oat_python` [33] 0.1.1. We use the package `scikit-learn` [34] 1.7.1 for Principal Component Analysis (PCA), MDS, clustering, and linear regression. We use `SciPy` [35] 1.16.1 for performing least squares minimization, computing pairwise distances, and handling sparse matrices. We also use the packages `Matplotlib` [36] 3.10.5 for plotting, `NumPy` [37] 2.3.2 for handling arrays and numerical computations, and `scikit-bio` [38] 0.7.0 for Principal Coordinate Analysis (PCoA).

3.3. Multidimensional Scaling

We perform MDS using `scikit-learn` ([34]) which utilizes the algorithm SMACOF ([1, 39]) for minimizing stress. We set $\epsilon_{stress} = 10^{-6}$, where ϵ_{stress} determines the threshold for the change in stress values when deciding whether to continue iterative optimization. We set the maximum number of iterations sufficiently high so that reaching the maximum number of iterations is never the reason for termination of the algorithm. For all examples, we input precomputed pairwise distances and output 2-dimensional MDS representations.

For random initialization, we use the technique outlined by [4], which we review here. Our input is an $n \times n$ matrix $\Delta = (\delta_{ij})$ of dissimilarities (in our analysis, δ_{ij} is the dissimilarity

between images i and j). We aim to find an MDS representation in Euclidean space of dimension p (in our analysis, $p = 2$). We set

$$\begin{aligned} m &= n(n-1)/2, \\ S &= \sum_{i < j} \delta_{ij}^2, \\ \sigma &= \sqrt{S/(2pm)}. \end{aligned}$$

For $i = 1, \dots, n$, we obtain pseudorandom values \tilde{x}_i and \tilde{y}_i (and similarly if a higher value of p is desired) from the standard normal distribution. Then the coordinates of the points in the initial configuration are $x_i = \sigma\tilde{x}_i$ and $y_i = \sigma\tilde{y}_i$, $i = 1, \dots, n$. The advantage of generating initial configuration points using this method is that the expected squared distance between points is equal to the average squared dissimilarity of the input data.

In addition to random coordinates, it is common to use the output of another dimensionality reduction technique as an initial configuration. In the cat food example, for which we compute cosine distances between images, we generate an additional MDS representation by using 2-dimensional PCoA coordinates as an initial configuration. In the toy duck example, for which we compute Euclidean distances between images, we generate an additional MDS representation by using 2-dimensional PCA coordinates as an initial configuration.

3.4. Topological Initialization for Multidimensional Scaling

The first recommended step of TIMDS is to check the dimension 1 barcode and verify the presence of a persistent feature. Technically, TIMDS can be performed whenever the barcode is nonempty, but TIMDS may not produce interpretable results without a robust topological feature to use to generate an initial configuration. Shorter intervals in the barcode generally indicate local structure or even noise, so we do not expect the corresponding cycle representatives to generate a reliable global initial configuration.

The cycle representative corresponding to the longest interval in the barcode is the basis for TIMDS. We determine the data points which appear in the cycle representative, and the order in which they appear (using an arbitrary point as the starting point). The cycle representative diameter is defined to be the maximum pairwise distance between data points in the cycle representative.

We must now choose an initial configuration in 2-dimensional space. The points will lie on a circle centered at the origin whose diameter is equal to the diameter of the cycle representative. For each data point which appears in the cycle representative, there is a corresponding point placed on the circle, equally spaced and in the order they appear in the cycle representative. For data points which are not contained in the cycle representative, we find the nearest data point that is in the cycle representative and use the same corresponding point on the circle for a starting position. One could use alternate strategies for points not contained in the cycle representative so that there are no identical points in

the initial configuration, but this is not necessary for the examples we discuss here. This produces an initial configuration which can be provided to the MDS algorithm together with the pairwise distances between data points. An example of one such “topological initialization” is shown in Fig. 5.

3.5. Angle Prediction

MDS is frequently used as a visualization tool, but MDS representations can also facilitate quantitative analysis tasks. One such application is recovering circular coordinates [40]. We can find a circle which best fits the points in the MDS representation by minimizing the sum of squares of the residuals $r_i - R$, where r_i is the distance from point i to the center of the circle and R is the radius of the circle. Then each point in the MDS representation has an angle with respect to this circle (assuming no point lies exactly in the center of the circle, which does not occur here).

We can compare the least squares circle angle to the actual object rotation. We need to add 2π rad to some least squares circle angles to facilitate performing linear regression. To do so, we form a collection of points whose x-coordinates are the rotation angles of the toy duck and whose y-coordinates are the least squares circle angles obtained from a given MDS representation. We then perform single linkage clustering with a distance threshold of 1 rad. We add 2π rad to the least squares circle angles of the points in the cluster with the lowest average least squares circle angle, continuing until there is only one cluster (or until a maximum number of iterations is reached).

If one is only performing MDS for the purpose of extracting angles, then circular MDS may be of interest (see Sec. 6.4 of Ref. [2]). However, we prefer not constraining the final MDS configuration to lie on a circle. A standard representation in 2-dimensional space has benefits such as allowing visualization of the effect of the noise added to the images.

4. Results

We obtained one MDS representation for the cheeseburger dataset, five MDS representations for the cat food dataset, and three MDS representations for the toy duck dataset. Each MDS representation depends on the dataset and the initial configuration; otherwise, the same method was used to obtain all MDS representations. For the cheeseburger dataset, we only performed MDS using a random initial configuration. We used the cheeseburger example to generate Fig. 1 and Fig. 2, and did not perform TIMDS or further analysis. For the cat food dataset, we performed MDS by using three random initial configurations and by using PCoA coordinates for an initial configuration. We also performed TIMDS. For the toy duck dataset, we performed MDS by using one random initial configuration and by using PCA for an initial configuration. We also performed TIMDS. The number of iterations required to perform MDS and the stress of each final configuration are shown in Table 2. For the cat food and toy duck datasets, we compared the stress of the

MDS representation from TIMDS and the stress of the MDS representation from PCA/PCoA initialization; these results are shown in Table 3.

Table 2. Iterations required and stress obtained for MDS starting at different initial configurations.

Dataset	Initialization	Iterations	Stress
cheeseburger	random (seed 0)	50	4.30×10^{-2}
cat food	random (seed 0)	247	1.09×10^0
	random (seed 1)	173	1.22×10^0
	random (seed 2)	101	1.30×10^0
	PCoA	90	9.80×10^{-1}
	topological	68	1.02×10^0
toy duck	random (seed 0)	551	1.25×10^6
	PCA	131	1.25×10^6
	topological	62	1.34×10^6

Table 3. Comparison of MDS stress after topological initialization and MDS stress after PCA/PCoA initialization.

Dataset	Stress		
	Topological Initialization	PCA/PCoA Initialization	Ratio
cat food	1.02×10^0	9.80×10^{-1}	1.04
toy duck	1.34×10^6	1.25×10^6	1.07

4.1. Cat Food

We computed dimension 1 persistent homology from the pairwise distances between the images in the cat food dataset. The resulting dimension 1 barcode is shown in Fig. 3. The intervals, or bars, in the barcode are ordered based on the left endpoint of each interval for the purposes of plotting, but barcodes are not vectors and the bars have no inherent order. The cycle representative corresponding to the longest interval in the barcode was used for performing TIMDS. There are 61 points included in the persistent homology cycle representative, out of 72 points in the dataset. Consequently, some points have identical locations in the topological initial configuration.

The three random initial configurations are shown in the left column of Fig. 4. The resulting MDS representations are shown in the right column. Each point corresponds to a single image of the can of cat food. The lightness of the points corresponds to the original order of the images (ground truth information not used in performing MDS). The PCoA coordinates

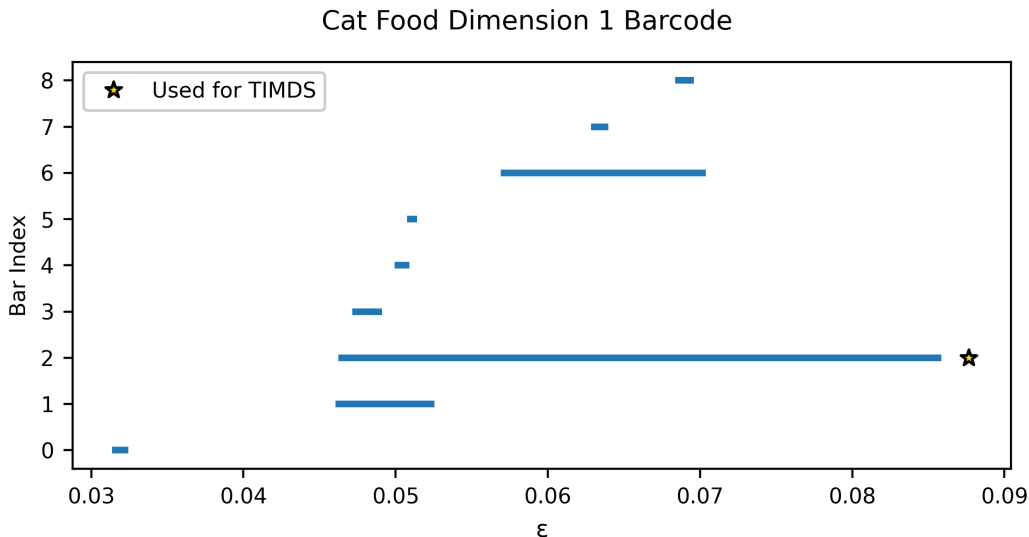


Fig. 3. The dimension 1 barcode of the cat food dataset; the cycle representative corresponding to the longest interval is used for performing TIMDS.

and the topological initialization are shown in the left column of Fig. 5. The corresponding MDS representations are shown in the right column. The iterations needed to obtain the five MDS representations of the cat food dataset, as well as the corresponding final stress values, are shown in Table 2.

4.2. Toy Duck

The dimension 1 barcode of the toy duck dataset is shown in Fig. 6. The cycle representative corresponding to the longest interval is used for performing TIMDS. There are 63 points in the cycle representative, out of 100 points. Each point corresponds to a noisy image of the toy duck.

The three MDS representations of the toy duck dataset (with images selected with replacement, and noise added) are shown in Fig. 7. The lightness of the points corresponds to the object rotation. The rotation is periodic, so that very light and very dark points correspond to similar images of the toy duck. The iterations needed to obtain the three MDS representations of the toy duck dataset, as well as the corresponding final stress values, are shown in Table 2.

Figure 8 shows the actual rotation of the toy duck and the least squares circle angle obtained from the three MDS representations. We added 2π rad to the least squares circle angle of the points plotted as plus signs. Initialization using random or PCA coordinates failed to produce a linear relationship of the least squares circle angle versus the actual object rotation, so we did not perform linear regression in these cases. Topological initial-

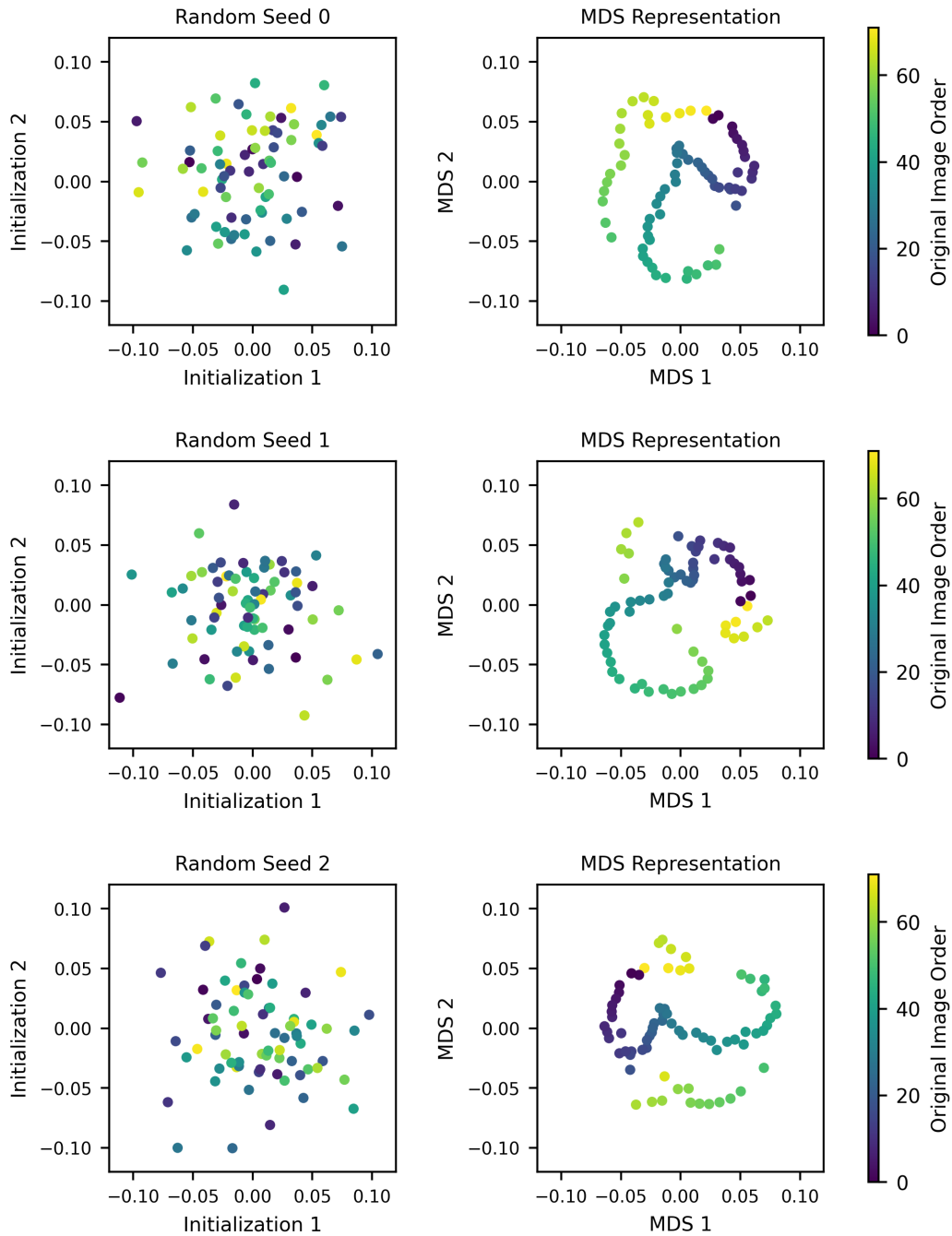


Fig. 4. MDS representations of the cat food dataset obtained from random initial configurations.

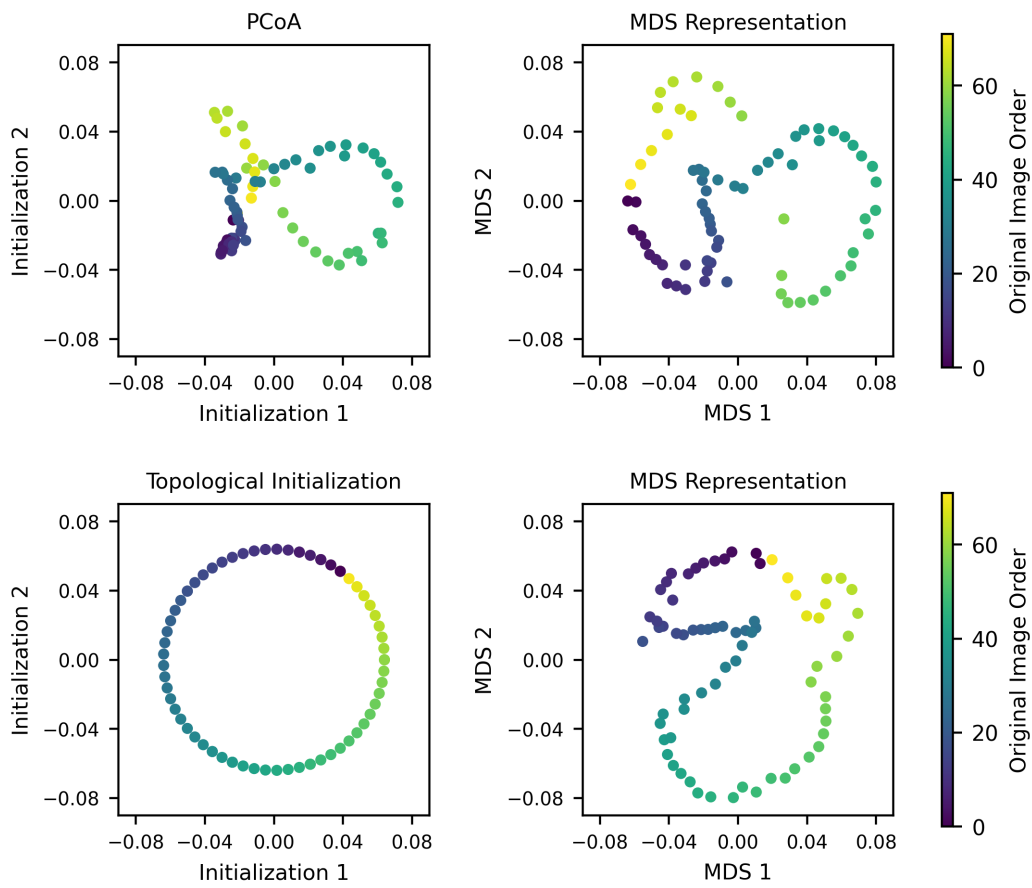


Fig. 5. MDS representations of the cat food dataset obtained using PCoA coordinates or topological initialization.

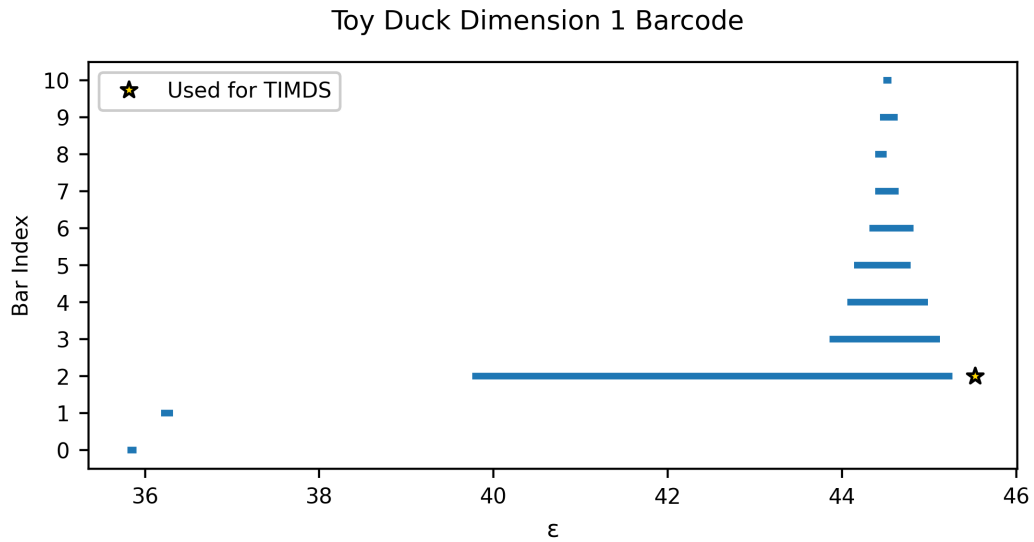


Fig. 6. The dimension 1 barcode of the toy duck dataset; the cycle representative corresponding to the longest interval is used for performing TIMDS.

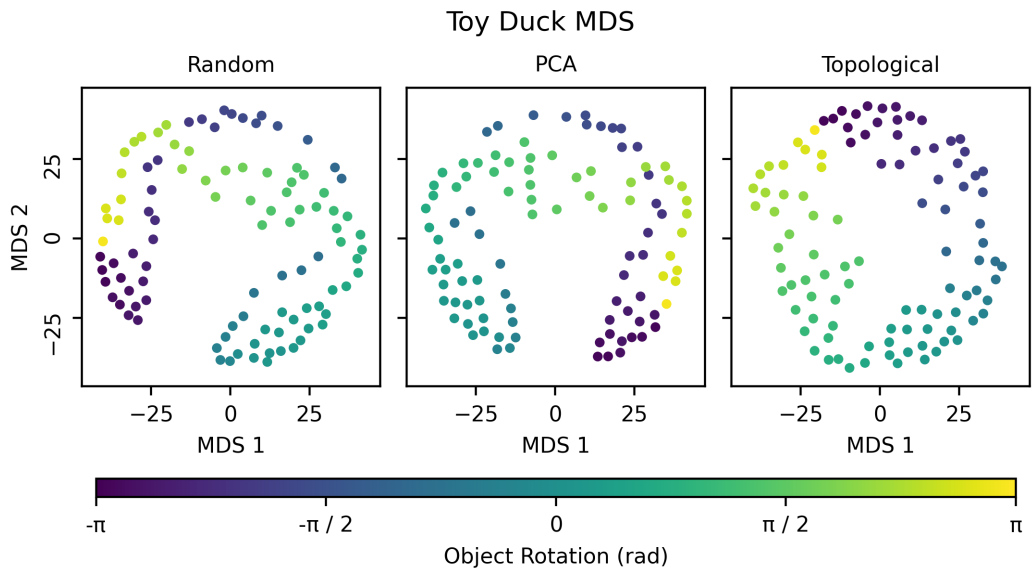


Fig. 7. MDS representations of the toy duck dataset obtained using different initial configurations.

ization did produce a clear linear relationship, with $R^2 = 0.994$ after adding 2π rad to some least squares circle angles.

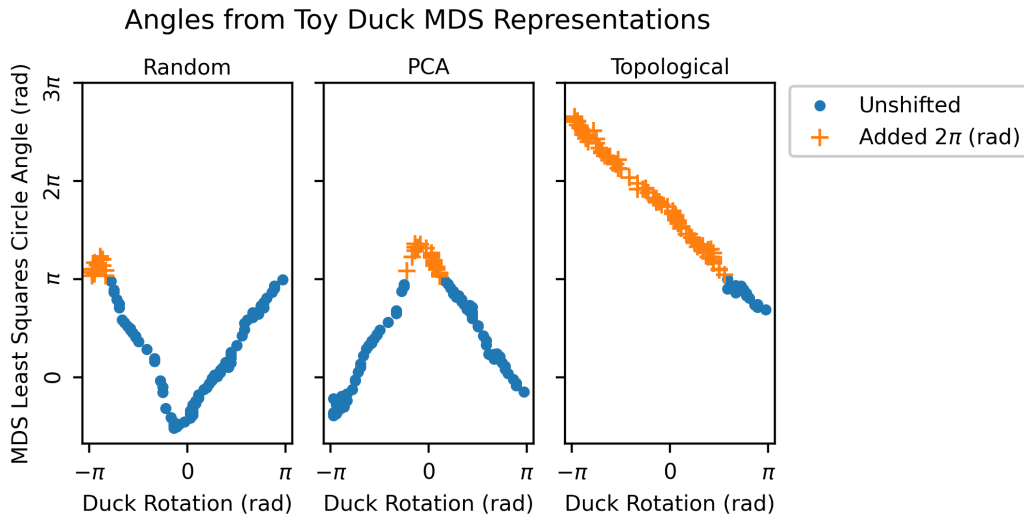


Fig. 8. Least squares circle angle versus the actual object rotation for initialization using a random initial configuration, PCA, and topological initialization.

5. Discussion and Future Work

For both the cat food and toy duck datasets, the closed loop structure and original image order were better represented by TIMDS than multidimensional scaling representations from random initial configurations or PCA/PCoA coordinates. As shown in Table 2, topological initialization required fewer iterations to obtain the final MDS representation compared to the other initial configurations used. For the toy duck example, TIMDS resulted in better recovery of circular coordinates.

For the cat food dataset, the stress of the MDS representation obtained using topological initialization is about 4% higher than the stress obtained using PCoA initialization. For the toy duck, the stress obtained using topological initialization is about 7% higher than the stress obtained using PCA initialization. These increases in stress for topological initialization versus PCA/PCoA initialization are well within what has been considered acceptable when characterizing outputs of confirmatory MDS ([1, 2]).

Although TIMDS makes use of tools from applied topology, it is still a multidimensional scaling technique. We use a standard loss function (stress), and the final configuration is still a set of points whose pairwise distances approximate the original data's pairwise dissimilarities. Since MDS is a well-established and popular visualization method, the fact that TIMDS is a subclass of MDS may make it more approachable for potential users who

don't typically perform topological data analysis. This is especially beneficial, since dataset visualization is frequently one of many steps in the data analysis pipeline.

There are many interesting technical questions raised by this preliminary work, such as the dependence of the results on the particular choice of cycle representative. It would also be interesting to investigate the robustness of the TIMDS method to noise. One option for exploring this further would be to start with a dataset for which TIMDS works well, such as one of the examples included in this work, and progressively add noise to the data. At the extreme, when the added noise completely overtakes the image data, we would expect TIMDS to fail. Characterizing how TIMDS performs for intermediate levels of noise could yield insights regarding its expected effectiveness for other datasets.

Techniques such as topological autoencoders [41] combine topology and dimensionality reduction by using persistent homology to inform the loss function. For the examples in this work it was sufficient to use TDA to obtain the initial configuration. However, if needed one could also consider topological adjustments to the MDS loss function.

Different random initial configurations produce qualitatively different results. In addition to TIMDS, another approach is to generate a large number of MDS representations from random initial configurations and use TDA to search for representations with good interpretability and reasonable stress. We are exploring this avenue in ongoing work [42].

6. Conclusion

Multidimensional scaling representations are typically obtained through iterative optimization to minimize quantities such as stress. This process is sensitive to the choice of initial configuration. There may exist qualitatively different final MDS representations with similar stress values. If so, how does one find the representations which provide interpretable visualizations or which improve the performance of quantitative analysis tasks? We show that TIMDS accomplishes this for some datasets.

7. Code Availability

Code to reproduce the examples in this manuscript is available at <https://github.com/usnistgov/TI-MDS>.

References

- [1] Borg I, Groenen PJF (2005) *Modern Multidimensional Scaling: Theory and Applications* (Springer, New York, NY), 2nd Ed. <https://doi.org/10.1007/0-387-28981-X>
- [2] Borg I, Groenen PJF, Mair P (2018) *Applied Multidimensional Scaling and Unfolding* (Springer, Cham, Switzerland), 2nd Ed. <https://doi.org/10.1007/978-3-319-73471-2>
- [3] Kruskal JB (1964) Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis. *Psychometrika* 29:1–27. <https://doi.org/10.1007/BF02289565>
- [4] Kearsley AJ, Tapia RA, Trosset MW (1998) The solution of the metric STRESS and SSTRESS problems in multidimensional scaling using Newton’s method. *Computational Statistics* 13:369–396.
- [5] Borg I, Mair P (2017) The choice of initial configurations in multidimensional scaling: Local minima, fit, and interpretability. *Austrian Journal of Statistics* 46(2):19–32. <https://doi.org/10.17713/ajs.v46i2.561>
- [6] Borg I (2020) Data fit (Stress) vs. model fit (recovery) in multidimensional scaling. *Austrian Journal of Statistics* 49(2):43–52. <https://doi.org/10.17713/ajs.v49i2.918>
- [7] Mair P, Borg I, Rusch T (2016) Goodness-of-fit assessment in multidimensional scaling and unfolding. *Multivariate Behavioral Research* 51(6):772–789. Available at <https://www.tandfonline.com/doi/abs/10.1080/00273171.2016.1235966>.
- [8] Liu J, Lin D, Yardımcı GG, Noble WS (2018) Unsupervised embedding of single-cell Hi-C data. *Bioinformatics* 34(13):i96–i104. <https://doi.org/10.1093/bioinformatics/bty285>
- [9] Carlsson G (2009) Topology and data. *Bulletin of the American Mathematical Society* 46(2):255–308. <https://doi.org/10.1090/S0273-0979-09-01249-X>
- [10] Rizvi AH, Camara PG, Kandror EK, Roberts TJ, Schieren I, Maniatis T, Rabadan R (2017) Single-cell topological RNA-seq analysis reveals insights into cellular differentiation and development. *Nature Biotechnology* 35:551–560. <https://doi.org/10.1038/nbt.3854>
- [11] Veres B, Schwertner WR, Tokodi M, Szijártó Á, Kovács A, Merkel ED, Behon A, Kuthi L, Masszi R, Gellér L, Zima E, Molnár L, Osztheimer I, Becker D, Kosztin A, Merkely B (2024) Topological data analysis to identify cardiac resynchronization therapy patients exhibiting benefit from an implantable cardioverter-defibrillator. *Clinical Research in Cardiology* 113:1430–1442. <https://doi.org/10.1007/s00392-023-02281-6>
- [12] Lymberopoulos E, Gentili GI, Alomari M, Sharma N (2021) Topological data analysis highlights novel geographical signatures of the human gut microbiome. *Frontiers in Artificial Intelligence* 4. <https://doi.org/10.3389/frai.2021.680564>
- [13] Stolz BJ, Dhese J, Bull JA, Harrington HA, Byrne HM, Yoon IHR (2024) Relational persistent homology for multispecies data with application to the tumor microenvironment. *Bulletin of Mathematical Biology* 86:128. <https://doi.org/10.1007/s11538-024-01353-6>

- [14] Myers AD, Chumley MM, Khasawneh FA, Munch E (2023) Persistent homology of coarse-grained state-space networks. *Phys Rev E* 107:034303. <https://doi.org/10.1103/PhysRevE.107.034303>
- [15] Kleczynski M, Bergonzo C, Kearsley AJ (2025) Spatial and sequential topological analysis of molecular dynamics simulations of IgG1 Fc domains. *Journal of Chemical Theory and Computation* <https://doi.org/10.1021/acs.jctc.5c00161>
- [16] Chazal F, Michel B (2021) An introduction to topological data analysis: Fundamental and practical aspects for data scientists. *Frontiers in Artificial Intelligence* 4. <https://doi.org/10.3389/frai.2021.667963>
- [17] Dey TK, Wang Y (2022) *Computational Topology for Data Analysis* (Cambridge University Press). <https://doi.org/10.1017/9781009099950>
- [18] Ghrist R (2014) *Elementary Applied Topology* (Createspace), 1st Ed.
- [19] Ghrist R (2008) Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society* 45(1):61–75. <https://doi.org/10.1090/S0273-0979-07-01191-3>
- [20] Li L, Thompson C, Henselman-Petrusek G, Giusti C, Ziegelmeier L (2021) Minimal cycle representatives in persistent homology using linear programming: An empirical study with user’s guide. *Frontiers in Artificial Intelligence* 4. <https://doi.org/10.3389/frai.2021.681117>
- [21] Henselman-Petrusek G, Hang H, Ziegelmeier L, Giusti C (2024) Open Applied Topology. Available at <https://github.com/openappliedtopology>.
- [22] Otter N, Porter MA, Tillmann U, Grindrod P, Harrington HA (2017) A roadmap for the computation of persistent homology. *EPJ Data Science* 6:17. <https://doi.org/10.1140/epjds/s13688-017-0109-5>
- [23] Zomorodian A, Carlsson G (2005) Computing persistent homology. *Discrete & Computational Geometry* 33:249–274. <https://doi.org/10.1007/s00454-004-1146-y>
- [24] Nene SA, Nayar SK, Murase H (1996) Columbia Object Image Library (COIL-100). *Technical Report, Department of Computer Science, Columbia University CUCS-006-96*. Available at <https://cave.cs.columbia.edu/repository/COIL-100>.
- [25] Nene SA, Nayar SK, Murase H (1996) Columbia Object Image Library (COIL-20). *Technical Report, Department of Computer Science, Columbia University CUCS-005-96*. Available at <https://cave.cs.columbia.edu/repository/COIL-20>.
- [26] Carrière M, Michel B, Oudot S (2018) Statistical analysis and parameter selection for Mapper. *Journal of Machine Learning Research* 19(12):1–39. Available at <http://jmlr.org/papers/v19/17-291.html>.
- [27] Perea JA (2020) Sparse circular coordinates via principal \mathbb{Z} -bundles. *Topological Data Analysis*, eds Baas NA, Carlsson GE, Quick G, Szymik M, Thaulé M (Springer, Cham, Switzerland), pp 435–458. https://doi.org/10.1007/978-3-030-43408-3_17
- [28] Schonsheck NC, Schonsheck SC (2024) Spherical coordinates from persistent cohomology. *Journal of Applied and Computational Topology* 8:149–173. <https://doi.org/10.1007/s41468-023-00141-w>

- [29] Carrière M, Theveneau M, Lacombe T (2024) Diffeomorphic interpolation for efficient persistence-based topological optimization. *Advances in Neural Information Processing Systems*, eds Globerson A, Mackey L, Belgrave D, Fan A, Paquet U, Tomczak J, Zhang C (Curran Associates, Inc.), Vol. 37, pp 27274–27294. Available at https://proceedings.neurips.cc/paper_files/paper/2024/hash/2feff80094b297bcfb42dbb01f34b875-Abstract-Conference.html.
- [30] Clémot M, Digne J, Tierny J (2025) Topological Autoencoders++: Fast and accurate cycle-aware dimensionality reduction. *arXiv preprint*. Available at <https://arxiv.org/abs/2502.20215>.
- [31] Nelson BJ, Luo Y (2022) Topology-preserving dimensionality reduction via interleaving optimization. *arXiv preprint*. Available at <https://arxiv.org/abs/2201.13012>.
- [32] Henselman-Petrusek G, Hang H, Ziegelmeier L, Giusti C (2024) oat_rust: User-friendly tools for applied topology. Available at https://crates.io/crates/oat_rust.
- [33] Henselman-Petrusek G, Hang H, Ziegelmeier L, Giusti C (2024) oat_python: User-friendly tools for applied topology in Python. Available at https://crates.io/crates/oat_python.
- [34] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830. Available at <http://jmlr.org/papers/v12/pedregosa11a.html>.
- [35] Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Millman KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat İ, Feng Y, Moore EW, VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P, SciPy 1.0 Contributors (2020) SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods* 17:261–272. <https://doi.org/10.1038/s41592-019-0686-2>
- [36] Hunter JD (2007) Matplotlib: A 2D graphics environment. *Computing in Science & Engineering* 9(3):90–95. <https://doi.org/10.1109/MCSE.2007.55>
- [37] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Río JF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE (2020) Array programming with NumPy. *Nature* 585(7825):357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- [38] Rideout JR, Bolyen E, McDonald D, Baeza YV, Alastuey JC, Pitman A, Morton J, Zhu Q, Navas J, Gorlick K, Aton M, Debelius J, Xu Z, Ilcooljohn, Shorestein J, Luce L, Treuren WV, Chase J, charudatta-navare, Gonzalez A, Brislawn CJ, Patena W, Schwarzberg K, teravest, Sfiligoi I, Reeder J, Caporaso G, shiffer1, nbresnick, Tapo C (2025) scikit-bio/scikit-bio: scikit-bio 0.7.0. <https://doi.org/10.5281/zenodo.15988672>.
- [39] de Leeuw J, Mair P (2009) Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software* 31(3):1–30. <https://doi.org/10.18637/jss.v031.i03>

- [40] de Silva V, Morozov D, Vejdemo-Johansson M (2011) Persistent cohomology and circular coordinates. *Discrete & Computational Geometry* 45:737–759. <https://doi.org/10.1007/s00454-011-9344-x>
- [41] Moor M, Horn M, Rieck B, Borgwardt K (2020) Topological autoencoders. *Proceedings of the 37th International Conference on Machine Learning* (PMLR), *Proceedings of Machine Learning Research*, Vol. 119, pp 7045–7054. Available at <https://proceedings.mlr.press/v119/moor20a.html>.
- [42] Kleczynski MA, Kearsley AJ (2025) Clustering persistence barcodes of multidimensional scaling representations from random initial configurations [Manuscript in preparation].