

TDA: Lecture 4

Statistics on barcodes

Wesley Hamilton

University of North Carolina - Chapel Hill

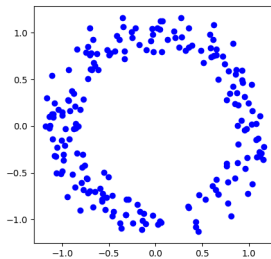
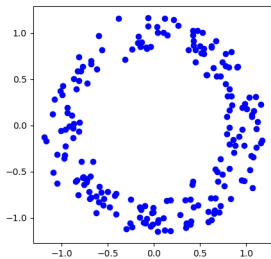
wham@live.unc.edu

02/19/20

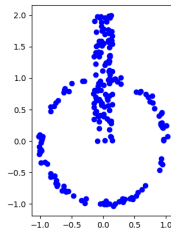
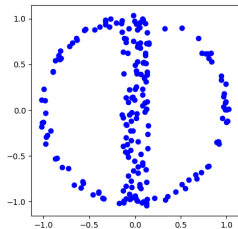
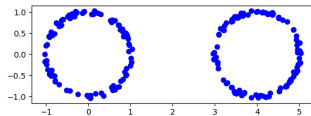
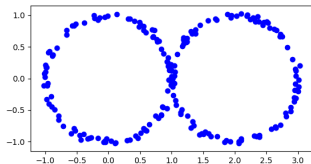
- 1 Some examples
- 2 The Bottleneck and Wasserstein metrics
- 3 Stability
- 4 Statistics
- 5 Other possibilities
- 6 Numerics

Some examples

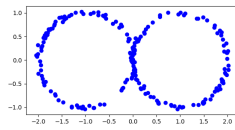
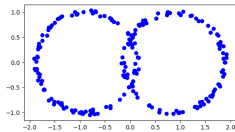
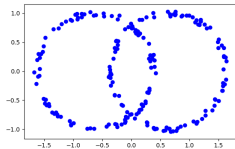
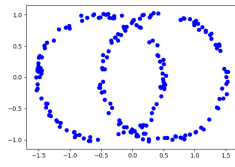
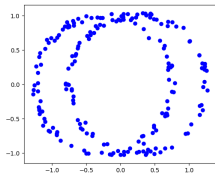
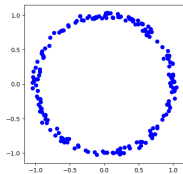
Example 1, resampling



Example 2



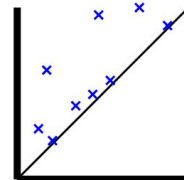
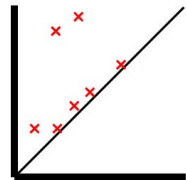
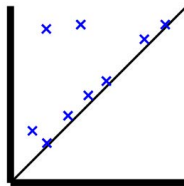
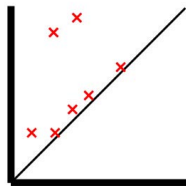
Example 3



Metrics on barcodes

Building up to metrics on barcodes

- Which of the following pairs of barcodes are “closer”?

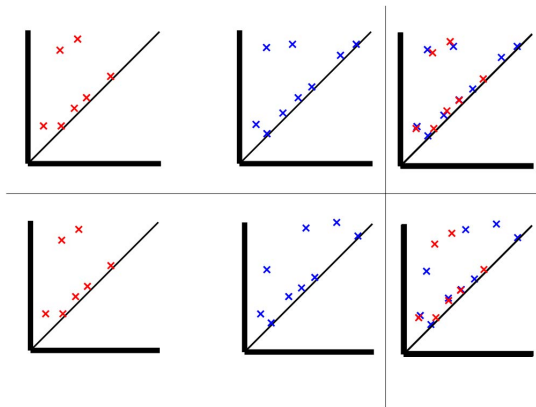


Building up to metrics on barcodes

- Which of the following pairs of barcodes are “closer”?
- Let’s overlay them...

Building up to metrics on barcodes

- Which of the following pairs of barcodes are “closer”?
- Let’s overlay them...



Building up to metrics on barcodes

- Idea: match up nearby features, and use distances between matched features to quantify the distance between two diagrams.

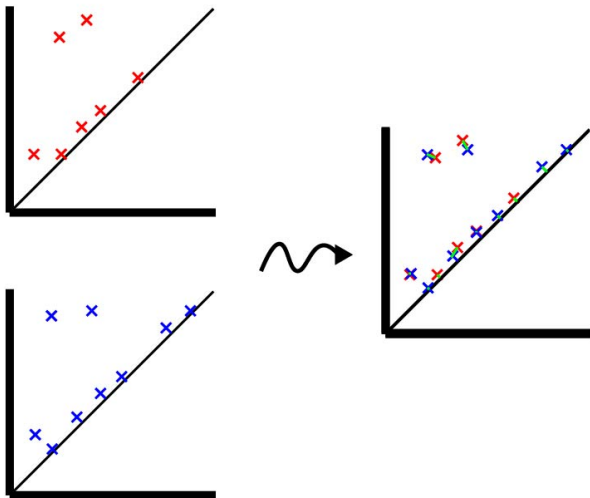
Building up to metrics on barcodes

- Idea: match up nearby features, and use distances between matched features to quantify the distance between two diagrams.
- Issue: diagrams may have different numbers of features.

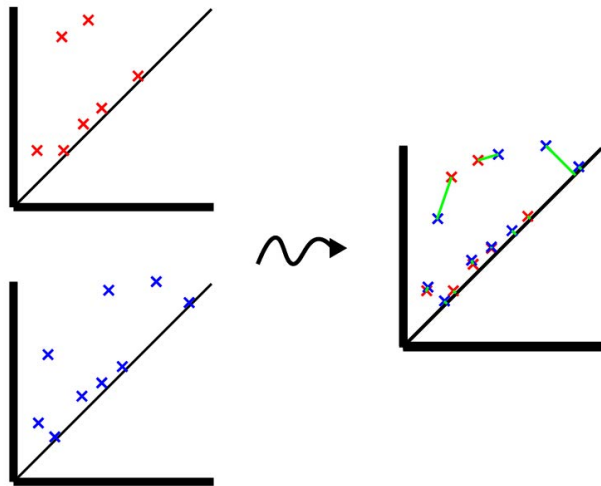
Building up to metrics on barcodes

- Idea: match up nearby features, and use distances between matched features to quantify the distance between two diagrams.
- Issue: diagrams may have different numbers of features.
- Workaround: allow features to be matched with the diagonal. Points on the diagonal are “ghost features”.

Building up to metrics on barcodes



Building up to metrics on barcodes



Bottleneck and Wasserstein distances

Definition

The **Bottleneck distance** between two persistence diagrams D_1, D_2 is

$$W_\infty(D_1, D_2) := \inf_{\gamma \in \Gamma} \sup_{p \in D_1 \cup \Delta} \|p - \gamma(p)\|_\infty,$$

where Γ is the set of bijections (one-to-one correspondences) between $D_1 \cup \Delta$ and $D_2 \cup \Delta$, with $\Delta = \{(x, x) : x \in \mathbb{R}\}$.

Bottleneck and Wasserstein distances

Definition

The **Bottleneck distance** between two persistence diagrams D_1, D_2 is

$$W_\infty(D_1, D_2) := \inf_{\gamma \in \Gamma} \sup_{p \in D_1 \cup \Delta} \|p - \gamma(p)\|_\infty,$$

where Γ is the set of bijections (one-to-one correspondences) between $D_1 \cup \Delta$ and $D_2 \cup \Delta$, with $\Delta = \{(x, x) : x \in \mathbb{R}\}$.

Definition

The q -**Wasserstein distance** between two diagrams D_1, D_2 is

$$W_q(D_1, D_2) := \left[\inf_{\gamma \in \Gamma} \sum_{p \in D_1 \cup \Delta} \|p - \gamma(p)\|_\infty^q \right]^{\frac{1}{q}},$$

where Γ is the set of bijections (one-to-one correspondences) between $D_1 \cup \Delta$ and $D_2 \cup \Delta$, with $\Delta = \{(x, x) : x \in \mathbb{R}\}$.

Some comments

- We're assuming the diagrams D_1, D_2 don't have any features with death time ∞ .
- $W_\infty(D_1, D_2) \leq W_q(D_1, D_2)$ for all q .
- As $q \rightarrow \infty$, $W_q(D_1, D_2) \rightarrow W_\infty(D_1, D_2)$.
- The fact that barcodes may have different numbers of features is accounted for by considering $D_i \cup \Delta$, instead of just Δ .

Stability

Stability of barcodes

- Particularly for example 1 (two samples of the same (noisy) circle), we would like to be confident that resampling/subsampling/whatever other “perturbations” there might be won’t affect the barcodes.

Stability of barcodes

- Particularly for example 1 (two samples of the same (noisy) circle), we would like to be confident that resampling/subsampling/whatever other “perturbations” there might be won’t affect the barcodes.
- Good news: they (more-or-less) shouldn’t.

Stability of barcodes

Theorem

Suppose K is a fixed simplicial complex, with two (monotonic) functions $f, g: K \rightarrow \mathbb{R}$. Suppose D_f is the persistence diagram (in any dimension) for the sub/super level set filtration on K via f ; respectively, D_g is the diagram corresponding to g . Then

$$W_\infty(D_f, D_g) \leq \|f - g\|_\infty = \sup_{\sigma \in K} |f(\sigma) - g(\sigma)|.$$

- There's a similar, but more technically involved, result for $W_q(D_f, D_g)$ for $q < \infty$; see “Computational Topology” by Edelsbrunner and Harer.

Statistics on barcodes

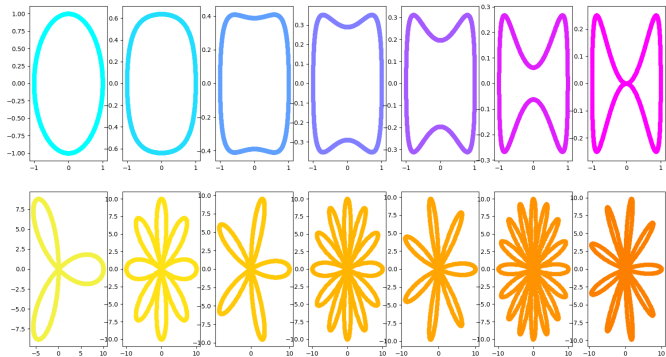
Statistics of barcodes

- Instead of computing the barcode of a single sample, and inferring something about the topological features, we can now use barcodes as data objects.
- For a dataset $\{x_1, \dots, x_n\}$, compute the barcodes $\{D_1, \dots, D_n\}$, and then compute pairwise distances $w_{ij} = W_q(D_i, D_j)$; can then perform statistical analysis on the distance matrix $W = (w_{ij})_{ij}$.
- One possible next step: dimensionality reduction using multidimensional scaling (MDS).

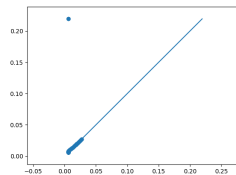
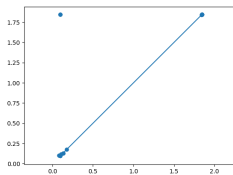
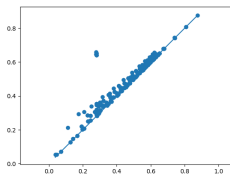
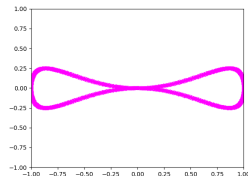
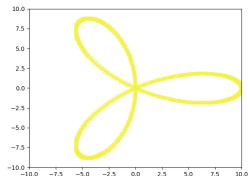
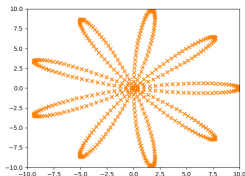
Statistics example: fancier toy example

- Our “fancier” toy dataset contains: 100 samples from a family of curves, starting as with circle, ending with a pinched circle (figure eight); 7 flower petals added in as outliers.
- 1-homology barcodes are computed for each sample.
- 2-Wasserstein distances are computed between each barcode.
- Sample points are visualized by an MDS embedding into 5 dimensions.

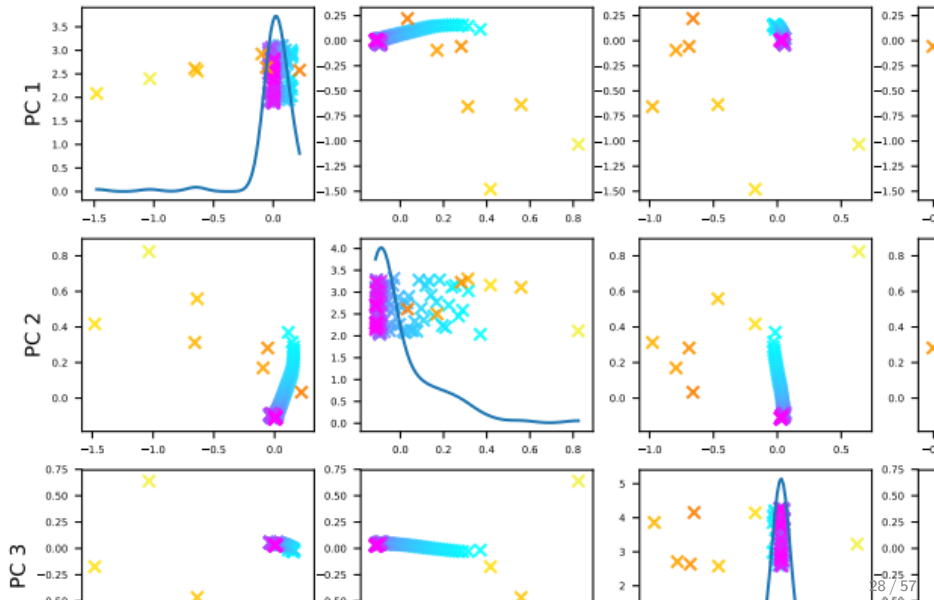
Statistics example: fancier toy example



Statistics example: fancier toy example



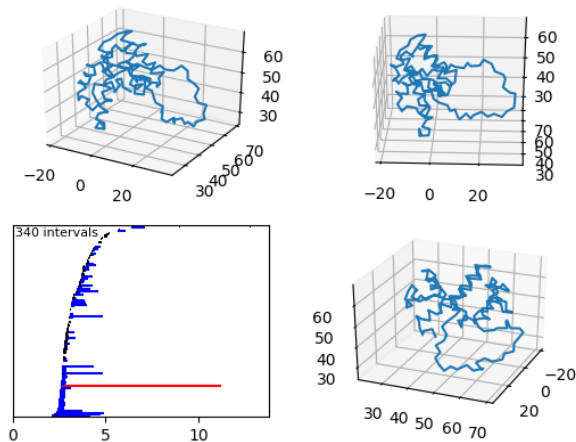
Statistics example: fancier toy example



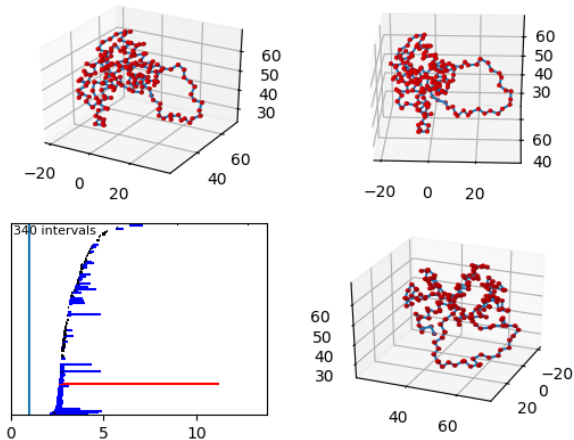
Statistics example: protein structure

- 8,155 proteins in the data set.
- 1-homology barcodes are computed for each protein using an alpha shape filtration in $\mathbb{R}^3$.
- “Distances” between proteins are computed as the 2-Wasserstein distance between the corresponding barcodes.

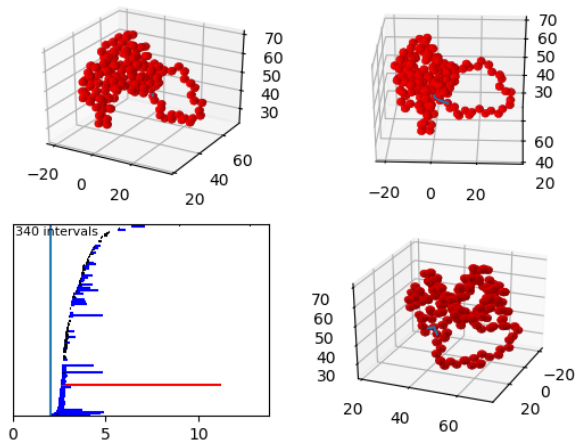
Statistics example: protein structure



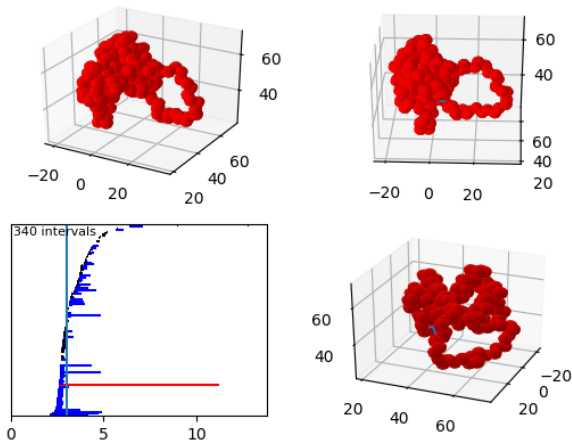
Statistics example: protein structure



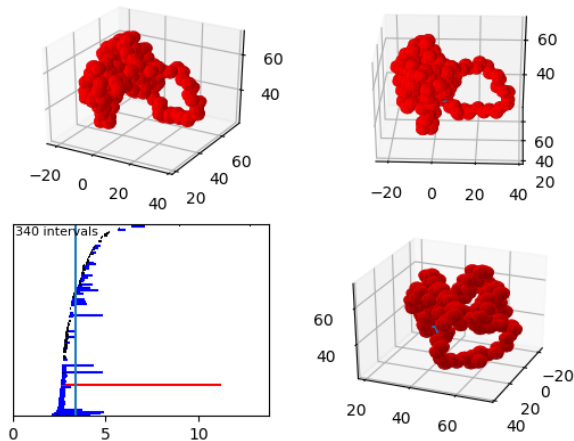
Statistics example: protein structure



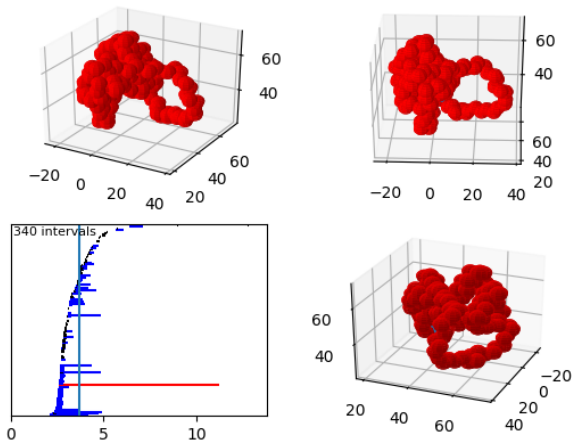
Statistics example: protein structure



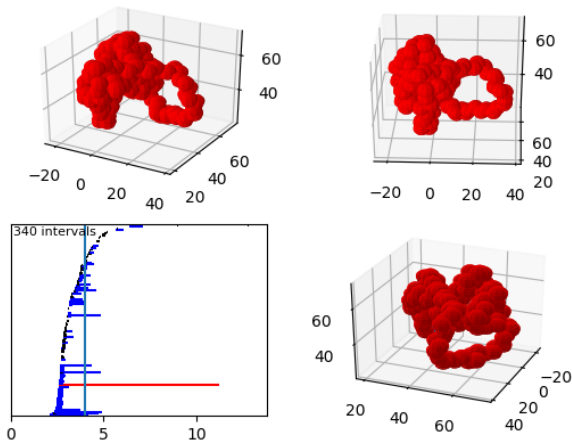
Statistics example: protein structure



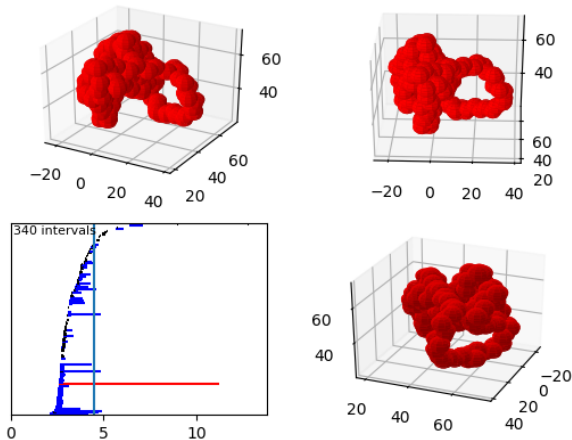
Statistics example: protein structure



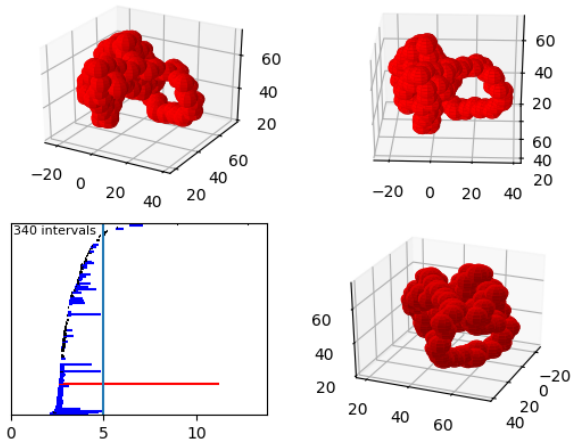
Statistics example: protein structure



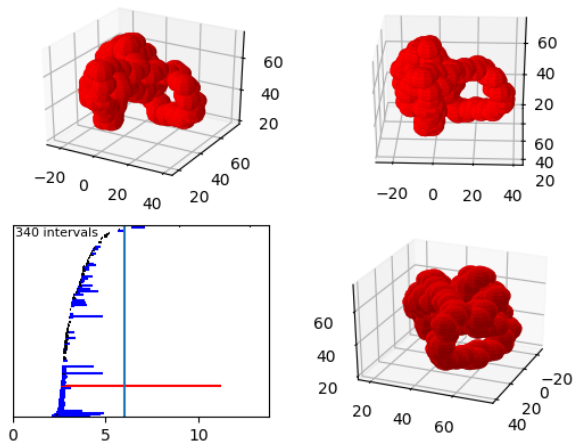
Statistics example: protein structure



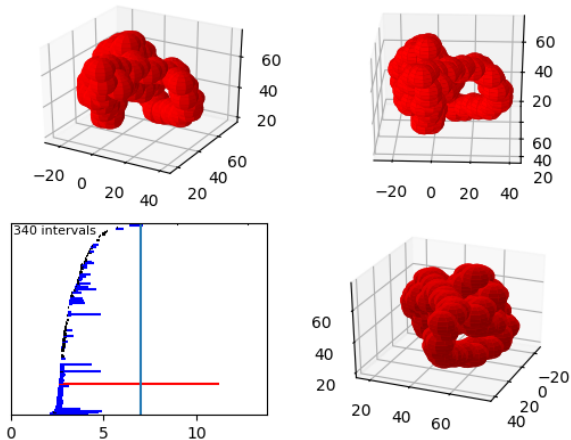
Statistics example: protein structure



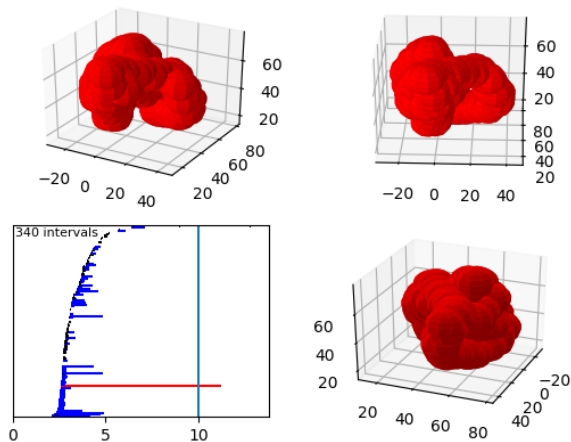
Statistics example: protein structure



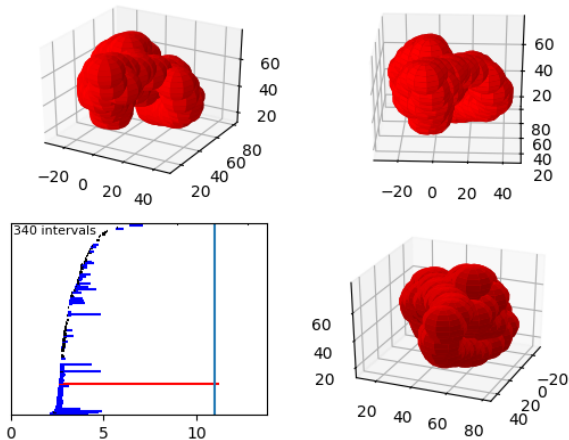
Statistics example: protein structure



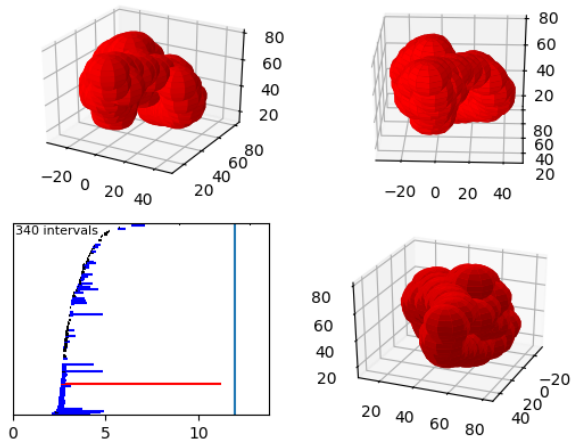
Statistics example: protein structure



Statistics example: protein structure



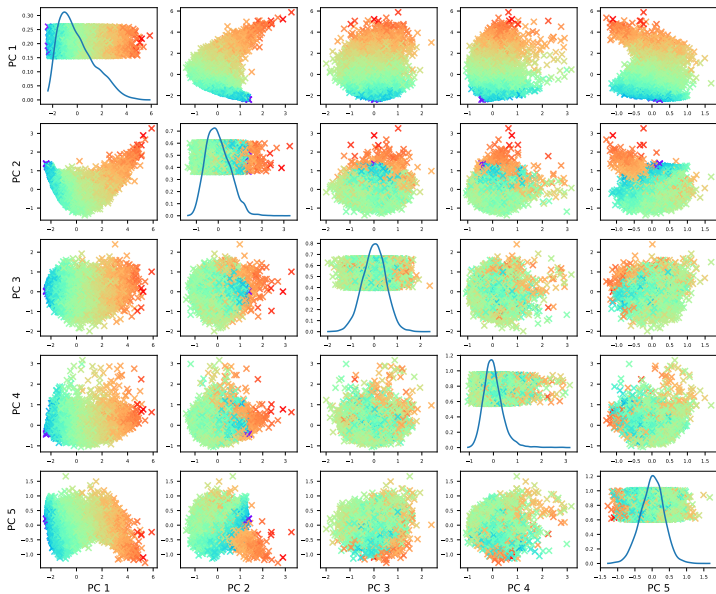
Statistics example: protein structure



Statistics example: protein structure

- Next, we performed MDS using the Wasserstein distance matrix to embed/visualize the structure of these proteins.
- Colours correspond to protein size/barcode size.
- Embedding dimension is $d = 50$, first 5 dimensions (corresponding to directions of maximal variance) plotted.

Statistics example: protein structure



Other possibilities/approximations

Persistent landscapes

- This is work by Bubenik, one of the earliest, quick ways to add a metric structure to the space of barcodes.
- Idea: treat barcodes as “landscape” functions/functions in a nice function space. Can then use L^p norms (or other integral metrics) to compute distances between functions.
- Doesn't exactly give any Wasserstein distance, but these L^p norms are comparable to Wasserstein distances.
- For more info: [arXiv:1810.04963](https://arxiv.org/abs/1810.04963)

Persistent landscapes

- For a birth/death pair (a, b) , define basis functions

$$f_{(a,b)}(t) = \max\{0, \min\{a + t, b - t\}\}.$$

- For a diagram $D = \{(a_i, b_i)\}$, define the k th **persistent landscape** as

$$\lambda_k(t) = \text{kmax}\{f_{(a_i,b_i)}(t)\}_i;$$

kmax denotes the k th largest element.

Persistent landscapes

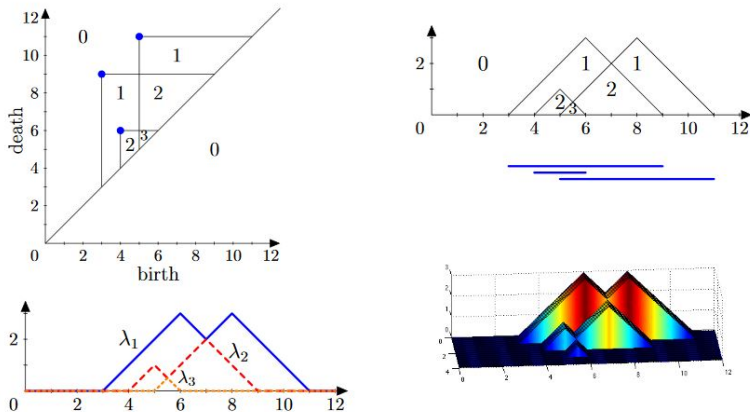


Figure: From arXiv:1207.6437

Rank function PCA

- The k th dimensional **rank function** of a filtration K is

$$\begin{aligned}\beta_k(K): \mathbb{R}_+^2 &\rightarrow \mathbb{Z} \\ (a, b) &\mapsto \dim H_k(a, b);\end{aligned}$$

$\dim H_k(a, b)$ is the number of topological features born before a and die after b .

- Rank function PCA is functional PCA on the rank functions.
- For more info: [arXiv:1507.01454](https://arxiv.org/abs/1507.01454)

Rank function PCA

- Suppose $f_1, \dots, f_n$ are the rank functions for diagrams $D_1, \dots, D_n$, and let $v_1, \dots, v_n$ are discretizations of $f_1, \dots, f_n$. The algorithm is:
 - 1 Calculate the mean $\bar{v}$ of $v_1, \dots, v_n$, center the discretizations by $\tilde{v}_i := v_i - \bar{v}$.
 - 2 Calculate the inner product matrix $D = (\tilde{v}_i \cdot \tilde{v}_j)_{ij}$.
 - 3 Compute the eigenvalues/eigenvectors of D , sorted by largest eigenvalue to smallest.
 - 4 Let $\lambda_j, w_j = (a_1, \dots, a_n)$ be an eigenvalue/eigenvector pair of D . The j th principal component is

$$\zeta_j = \frac{1}{\|\sum_i a_i \tilde{v}_i\|} \sum_{i=1}^N a_i \tilde{v}_i.$$

- The vectors ζ_j are the values of the principal directions on the underlying discretization.
- ζ_1 encodes how barcodes vary the most, ζ_2 encodes the second greatest direction of variation, etc.

Persistence images

- A similar kind of discretized “rank function” is achieved by (more naive) methods.
- Idea: “voxelize” the persistence diagram by using a Gaussian (or other) distribution.
- For more info: [arXiv:1507.06217](https://arxiv.org/abs/1507.06217)

Persistence images

- Suppose $D = \{(a_i, b_i)\}$ is a persistence diagram, $T(D) = \{(a_i, b_i - a_i)\}$, $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ is some weight function that is zero on the x -axis, and $\phi_{(x_0, y_0)}$ is some probability distribution, such as a Gaussian

$$\phi_{(x_0, y_0)}(x, y) = \frac{1}{2\pi\sigma^2} e^{-[(x-x_0)^2 + (y-y_0)^2]/(2\sigma^2)}.$$

Persistence images

- Suppose $D = \{(a_i, b_i)\}$ is a persistence diagram,
 $T(D) = \{(a_i, b_i - a_i)\}$, $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ is some weight function that is zero on the x -axis, and $\phi_{(x_0, y_0)}$ is some probability distribution, such as a Gaussian

$$\phi_{(x_0, y_0)}(x, y) = \frac{1}{2\pi\sigma^2} e^{-[(x-x_0)^2 + (y-y_0)^2]/(2\sigma^2)}.$$

- The **persistence surface** $\rho_D: \mathbb{R}^2 \rightarrow \mathbb{R}$ corresponding to the diagram D is the function

$$\rho_D(x, y) = \sum_{(a, b) \in T(D)} f(a, b) \phi_{(a, b)}(x, y).$$

- The **persistence image** is a discretization of the persistence image.

Persistence images

- In the original paper, Pls were proposed as a vectorization/input layer for deep learning.
- Pls can certainly be used for statistical needs.
- Process is generally similar to a kernel density estimate.

Persistence images

- In the original paper, PIs were proposed as a vectorization/input layer for deep learning.
- PIs can certainly be used for statistical needs.
- Process is generally similar to a kernel density estimate.

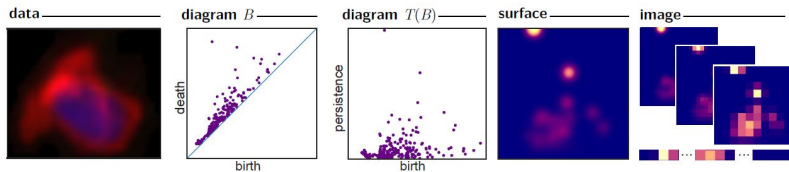


Figure 1: Algorithm pipeline to transform data into a persistence image.

Figure: From arXiv:1507.06217

Numerics

Numerical implementations

- The following software can compute Wasserstein distances:
 - TDA in R (a slower method),
 - Dionysus (a faster method),
 - GUDHI (just bottleneck),
 - JavaPLEX (just bottleneck).
- Otherwise, persistent landscapes give an L^2 based distance, rank functions and persistence images both give discrete analogues of L^2 distances.