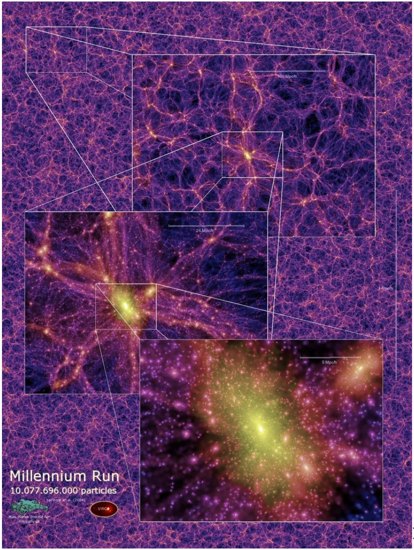
Topological Data Analysis for Feature Extraction and Model Evaluation

Jisu KIM

Topological Data Analysis and Industrial Mathematics
2025-08-08

Topological structures in the data provide information.



[1]

[1] http://www.mpa-garching.mpg.de/galform/virgo/millennium/poster_half.jpg

Persistent Homology: observe topological structure with multi resolutions.

Persistent Homology: observe topological structure with multi resolutions.

Persistent Homology: observe topological structure with multi resolutions.

Persistent Homology: observe topological structure with multi resolutions.

- Georges Seurat, A Sunday afternoon on the island of La Grande Jatte (Un dimanche après-midi à l'Île de la Grande Jatte)

A (very) rough introduction to Machine Learning

- For given problem and data, machine learning / deep learning learns a parametrized model.
  - Given data $\mathcal{X}$,
  - Parametrized model f_θ,
  - Loss funciton $\mathcal{L}$ adapted to a problem,
  - Machine Learning computes a solution that minimizes the loss function: $\arg\min_\theta \mathcal{L}(f_\theta, \mathcal{X})$.
- For many cases, computing an explicit formula for the minimizer is impossible or too expensive (e.g. inverting a large matrix). So, we often use gradient descent using $\nabla_\theta \mathcal{L}(f_\theta, \mathcal{X})$:

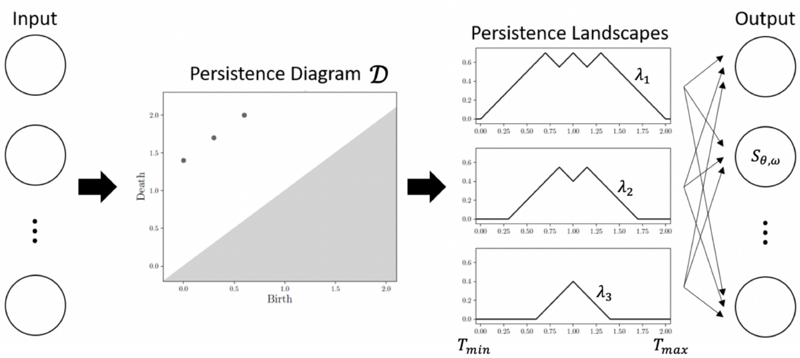
$$\theta_{n+1} = \theta_n - \lambda \nabla_\theta \mathcal{L}(f_\theta, \mathcal{X}).$$

Topological Data Analysis is applied to Machine Learning.

- A Survey of Topological Machine Learning Methods (Hensel, Moor, Rieck, 2021)
- Roughly, there are two directions applying Topological Data Analysis (TDA) to Machine Learning:
  - Make features from TDA to add topological features to data $\mathcal{X}$: more common
    - PLLay: Efficient Topological Layer based on Persistence Landscapes (Kim, Kim, Zaheer, Kim, Chazal, Wasserman, 2020)
    - Generalized penalty for circular coordinate representation (Luo, Patania, Kim, Vejdemo-Johansson, 2021)
    - ECLayr: Fast and Robust Topological Layer based on Differentiable Euler Characteristic Curve (Lee, Kim, Kim, 2025?)
  - Evaluate quality of data $\mathcal{X}$ or model f_θ using TDA: recently of interest
    - TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)

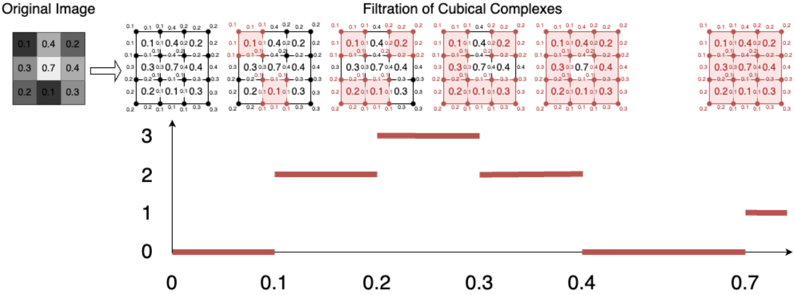
Topological structure is featurized as persistence landscape to be further applied in machine learning framework.

- Featurization using Persistence Landscape
  - PLLay: Efficient Topological Layer based on Persistent Landscapes (Kim, Kim, Zaheer, Kim, Chazal, Wasserman, 2020)



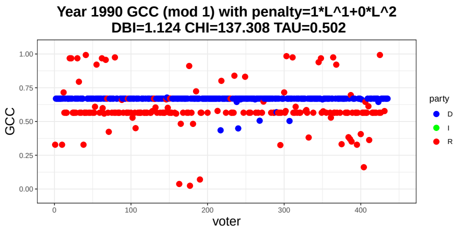
Topological structure is featurized as euler characteristic curve to be further applied in machine learning framework.

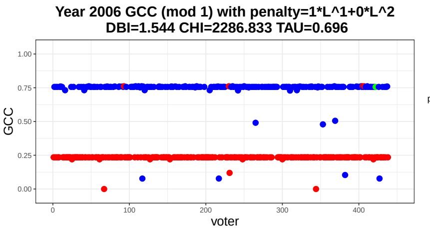
- Featurization using Euler Characteristic Curve
  - ECLayr: Fast and Robust Topological Layer based on Differentiable Euler Characteristic Curve (Lee, Kim, Kim, 2025?)



Topological structure is featurized as circular coordinates to be further applied in machine learning framework.

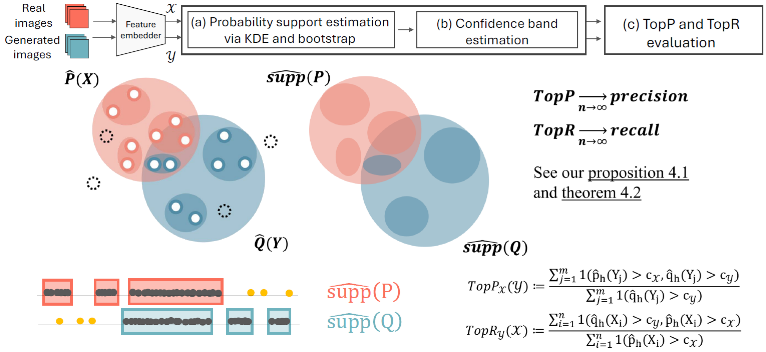
- Featurization using Circular Coordinates
  - Generalized penalty for circular coordinate representation (Luo, Patania, Kim, Vejdemo-Johansson, 2021)



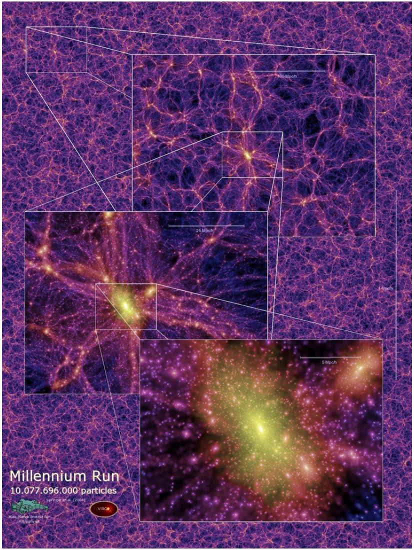


Data or Model is evaluated using Topological Data Analysis.

- Evaluation using Confidence of Topological Data Analysis
  - TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)



Topological holes in the data provide information.



The number of holes is used to summarize geometrical features.

- Geometrical objects :
  - ㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅅ, ㅇ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ
  - A, あ, い, う, え, お
  - 福, 岡, 九, 州, 大, 学, 西, 新, プ, ラ, ザ
- The number of holes of different dimensions is considered.

1. $\beta_0 =\#$ of connected components 
2. $\beta_1 =\#$ of loops (holes inside 1-dim sphere) 
3. $\beta_2 =\#$ of voids (holes inside 2-dim sphere) : if *dim* ≥ 3 

Example : Objects are classified by homologies.

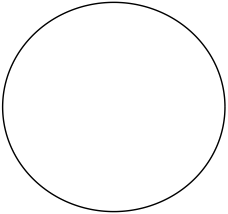
1. β_0 =# of connected components 
2. β_1 =# of loops 

β_0 \ β_1	0	1	2	5
1	ㄱ, ㄴ, ㄷ, ㄹ, ㅅ, ㅈ, ㅋ, ㅌ, 九, 大	ㅁ, ㅇ, ㅂ, ㅍ, A	あ,西	
2	ㅊ, い, う, え, ラ	お, 新, プ		
3	岡, ザ	ㅎ		
4	学			
6	州			
7				福

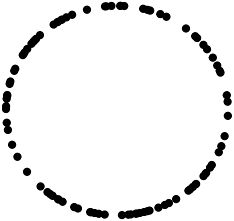
Homology of finite sample is different from homology of underlying ma6nifold, hence it cannot be directly used for the inference.

- When analyzing data, we prefer robust features where features of the underlying manifold can be inferred from features of finite samples.
- Homology is not robust:

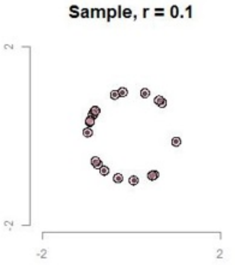
Underlying circle: $\beta_0 = 1$, $\beta_1 = 1$



100 samples: $\beta_0 = 100$, $\beta_1 = 0$

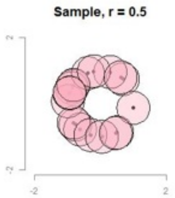


Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.

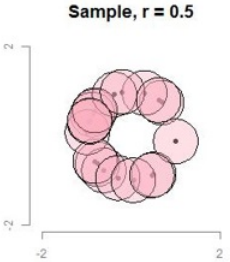


Augmented Set

Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.

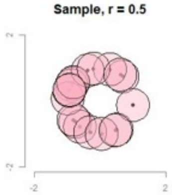


r=0.5 : 1-dim hole is formed

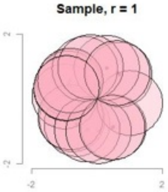


Augmented Set

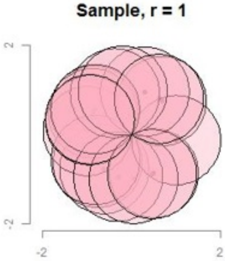
Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.



r=**0.5** : 1-dim hole is formed

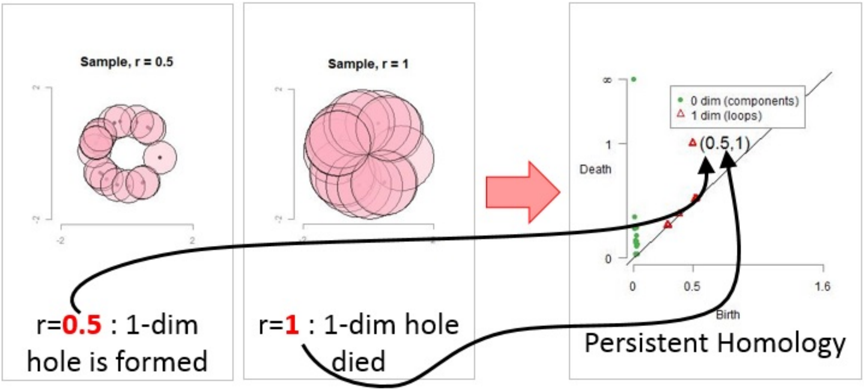


r=**1** : 1-dim hole died



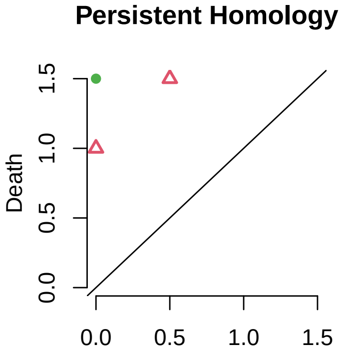
Augmented Set

Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.

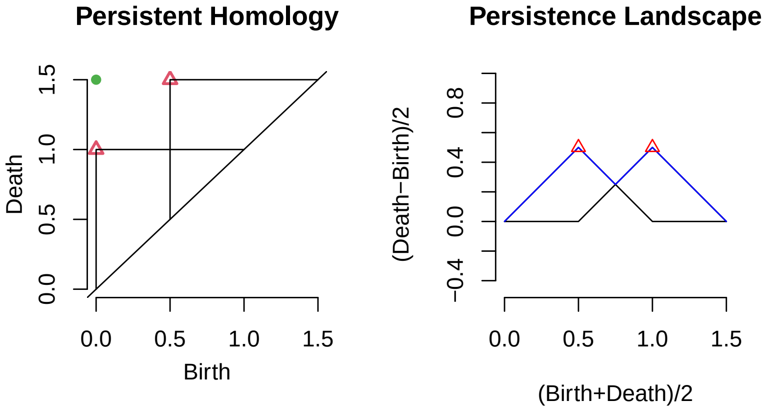


Persistent homology is further summarized and embedded into a Euclidean space or a functional space.

- The space of the persistent homology is complex, so directly applying in machine learning is difficult.
- If the persistent homology is further summarized and embedded into a Euclidean space or a functional space, then applying in machine learning becomes much more convenient.
  - e.g., Persistence Landscape, Persistence Silhouette, Persistence Image



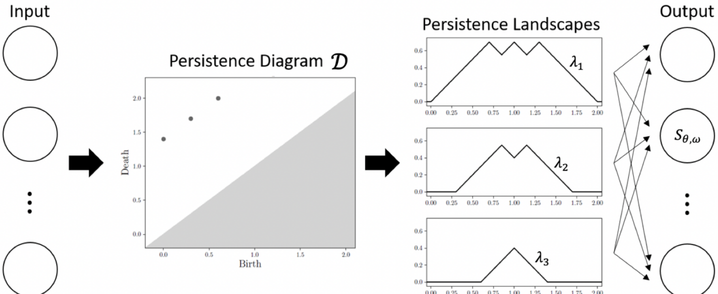
Persistence Landscape is a functional summary of the persistent homology.



PLLay: Build topological layer using Persistence Landscape

- PLLay: Efficient Topological Layer based on Persistence Landscapes (Kim, Kim, Zaheer, Kim, Chazal, Wasserman, 2020)

1. From data X, choose an appropriate simplicial complex K and a function f to compute the Persistece diagram $\mathcal{D}$.
2. From the persistence diagram $\mathcal{D}$, compute the persistence landscape $\lambda : \mathbb{N} \times \mathbb{R} \to \mathbb{R}$.
3. Compute the weighted average function $\bar{\lambda}_\omega(t) := \sum_{k=1}^{K_{\max}} \omega_k \lambda_k(t)$, and vectorize to get $\bar{\Lambda}_\omega \in \mathbb{R}^m$.
4. For a parametrized differentiable map $g_\theta : \mathbb{R}^m \to \mathbb{R}$, compute $S_{\theta,\omega}(\mathcal{D}) := g_\theta(\bar{\Lambda}_\omega)$.



PLLay is differentiable.

- A deep learning model learns its parameters by back propagation, which is to apply gradient descent layer-wise.
- For a deep learning layer to be learnable, it should be differentiable:

Theorem (Theorem 3.1 in Kim et al. [2020])

The PLLay function $S_{\theta,\omega}$ is differentiable with respect to the input data X.

PLLay is stable.

- PLLay is stable with respect to changes in persistence diagrams:

Theorem (Theorem 4.1 in Kim et al. [2020])

For two persistence diagrams $\mathcal{D}, \mathcal{D}'$,

$$|S_{\theta,\omega}(\mathcal{D}) - S_{\theta,\omega}(\mathcal{D}')| = O(d_B(\mathcal{D}, \mathcal{D}')),$$

where d_B is the bottleneck distance.

PLLay is stable.

- PLLay is stable with respect to perturbations in input X:

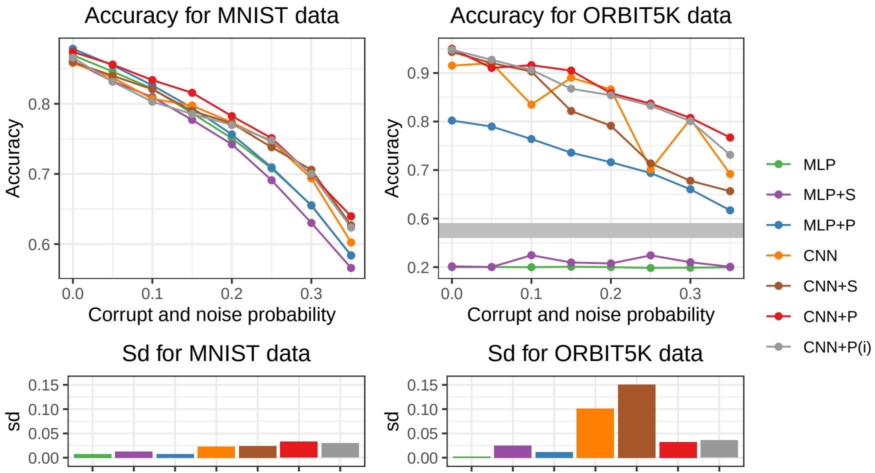
Theorem (Theorem 4.2 in Kim et al. [2020])

Let $X \sim P$ and P_n be the empirical distribution. Further, let $\mathcal{D}_P, \mathcal{D}_X$ be the persistence diagrams of P, X, respectively. Then

$$|S_{\theta,\omega}(\mathcal{D}_X) - S_{\theta,\omega}(\mathcal{D}_P)| = O(W_2(P_n, P)),$$

where W_2 is 2-Wasserstein distance.

Experiments

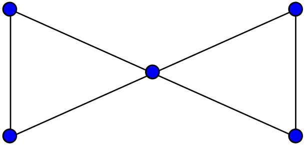
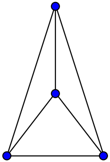


Euler Characteristic is computationally efficient.

- Euler Characteristic of a simplex or cubical complex is an alternating sum of betti numbers: for a simplex / cubical complex K,

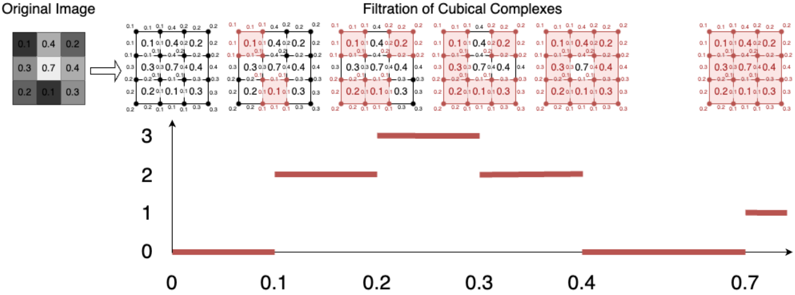
$$\chi(K) = \sum_{k=0}^{\infty} (-1)^k |K^k| = \sum_{k=0}^{\infty} (-1)^k \beta_k,$$

where K^k is the set of k-dimensional simplices in K, and β_k is the k-th Betti number of K.

-  $\chi(K) = 5 - 6 = 1 - 2 = -1$
-  $\chi(K) = 4 - 6 = 1 - 3 = -2$

Euler Characteristic Curve is computationally efficient compared to Persistent Homology.

- Euler Characteristic Curve (ECC) $\mathcal{C} : \mathbb{R} \to \mathbb{R}$ computes the Euler characteristic along a filtration.
- ECC does not involve computing persistent homology, hence more computationally efficient compard to persistent homology.



EClayr: Build topological layer using Euler Characteristic Curves

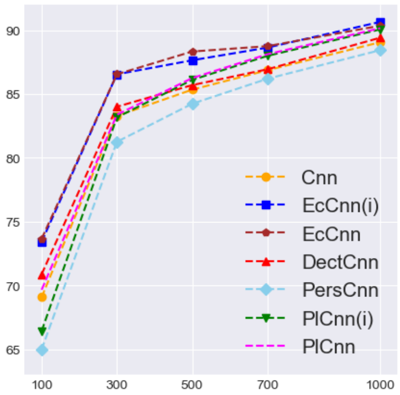
- ECLayr: Fast and Robust Topological Layer based on Differentiable Euler Characteristic Curve (Lee, Kim, Kim, 2025?)

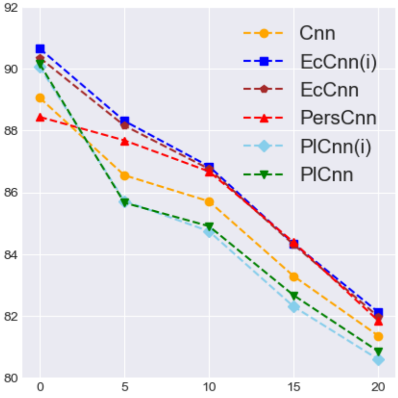
1. From data X, choose an appropriate simplicial complex K and a function f to build a filtration.
2. From the filtration, compute the Euler Characteristic Curve $\mathcal{C} : \mathbb{R} \to \mathbb{R}$, and vectorize to get $\mathcal{E} \in \mathbb{R}^v$.
3. For a parametrized differentiable map $g_\theta : \mathbb{R}^m \to \mathbb{R}$, compute $\mathcal{O}_\theta := g_\theta(\mathcal{E})$.

Computation Time

Model	Data		
	MNIST	Br35H	Synthetic
ECC	3.129 sec	0.458 sec	2.17 sec
PH	33.700 sec	11.033 sec	59.288 sec

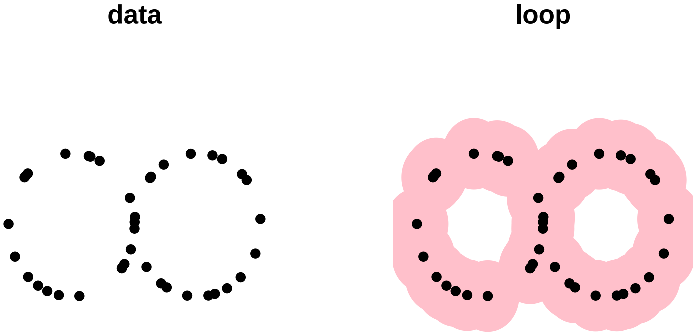
Experiments





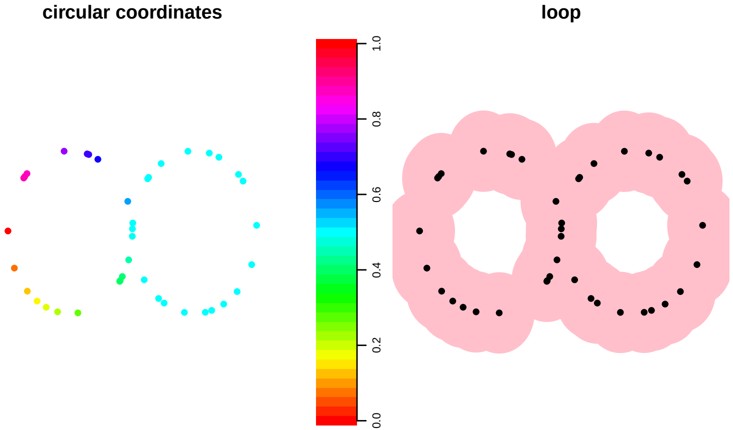
Circular coordinates provide topological representations of reduced dimension.

- Persistent cohomology and circular coordinates (de Silva, Morozov, Vejdemo-Johansson, 2011)
- Topological Learning for Motion Data via Mixed Coordinates (Vejdemo-Johansson, Pokorny, Skraba, Kragic, 2015)



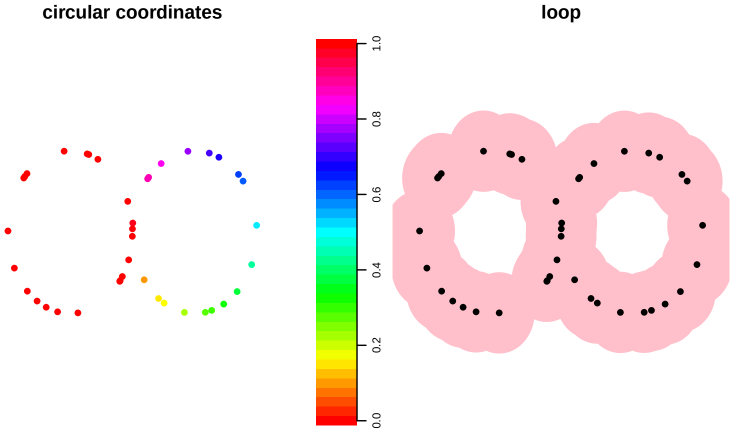
Circular coordinates provide topological representations of reduced dimension.

- circuiar coordinate is a function that maps from data points X to circle S^1.



Circular coordinates provide topological representations of reduced dimension.

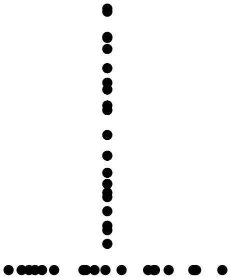
- circuiar coordinate is a function that maps from data points X to circle S^1.



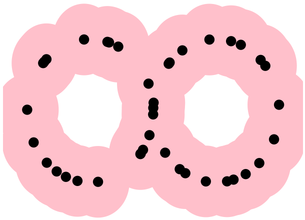
Circular coordinates provide topological representations of reduced dimension.

- circuiar coordinate is a function that maps from data points X to torus $\mathbb{T}^k = (S^1)^k$.

circular coordinates

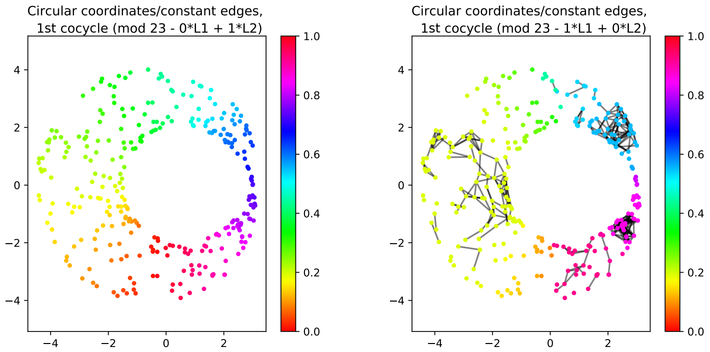


loop



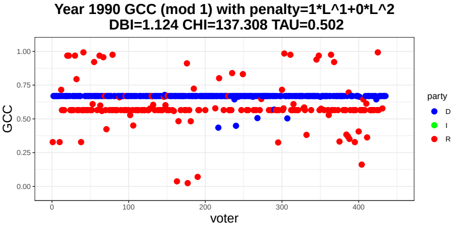
Circular coordinates with generalized penalty better visualizes topological information from data.

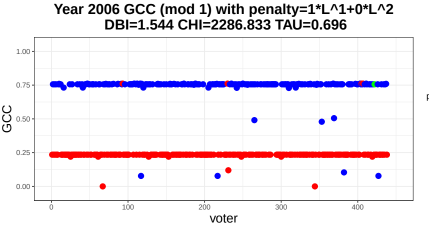
- Generalized penalty for circular coordinate representation (Luo, Patania, Kim, Vejdemo-Johansson, 2021)
- When computing circular coordinates, we solve an optimization problem.
- We switch L_2 loss by L_1 loss for circuiar coordinate values to change more abructly: better visualizes topological information from data.



Circular coordinates with generalized penalty better visualizes topological information from data.

- Generalized penalty for circular coordinate representation (Luo, Patania, Kim, Vejdemo-Johansson, 2021)
- Voting data in 2006 is more bipolarized than voting data in 1990.



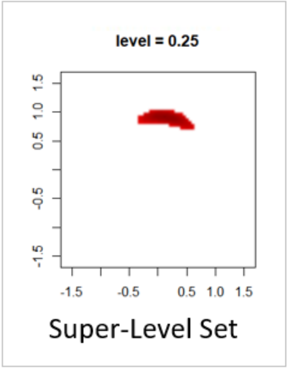


We rely on the kernel density estimator to extract topological information of the underlying distribution.

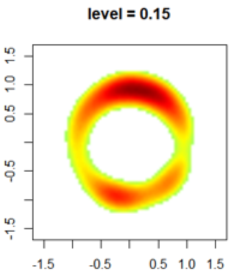
- The kernel density estimator is

$$\hat{p}_h(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right).$$

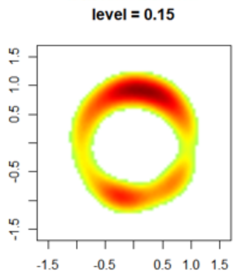
Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.



Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.

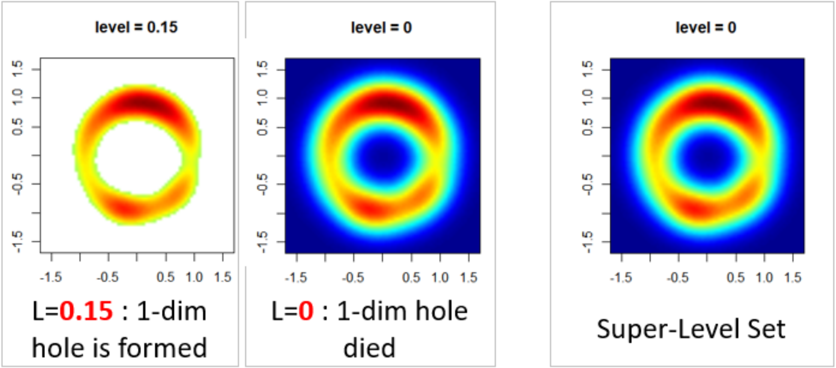


L=0.15 : 1-dim hole is formed

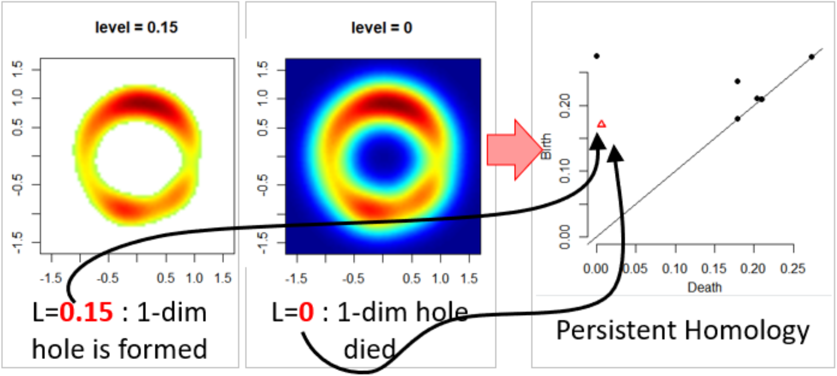


Super-Level Set

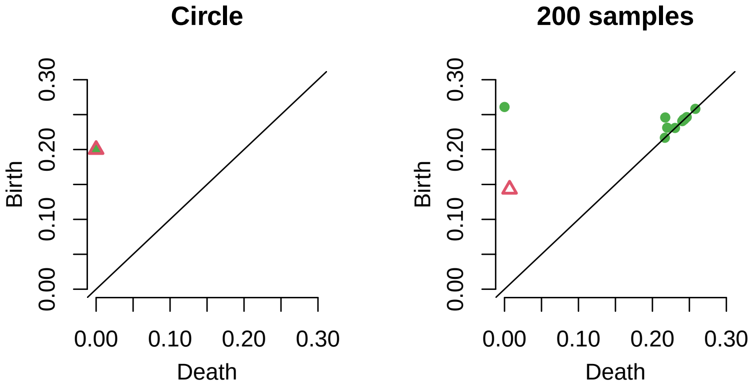
Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.



Persistent homology computes homologies on collection of sets, and tracks when topological features are born and when they die.



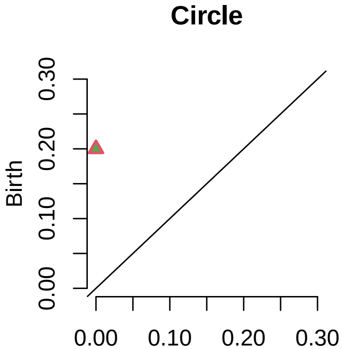
Persistent homology of the underlying manifold can be inferred from persistent homology of finite samples.

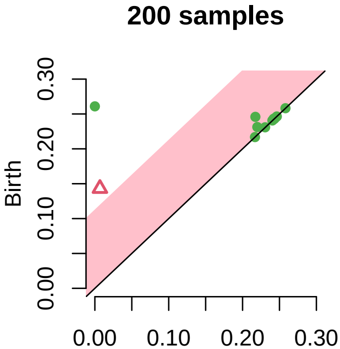


Confidence band for persistent homology separates homological signal from homological noise.

Let M be a compact manifold, and $X = \{X_1, \cdots, X_n\}$ be n samples. Let f_M and f_X be corresponding functions whose persistent homology is of interest. Given the significance level $\alpha \in (0,1)$, $(1-\alpha)$ confidence band $c_n = c_n(X)$ is a random variable satisfying

$$\mathbb{P}\left(d_B(Dgm(f_M),\, Dgm(f_X)) \leq c_n\right) \geq 1 - \alpha.$$



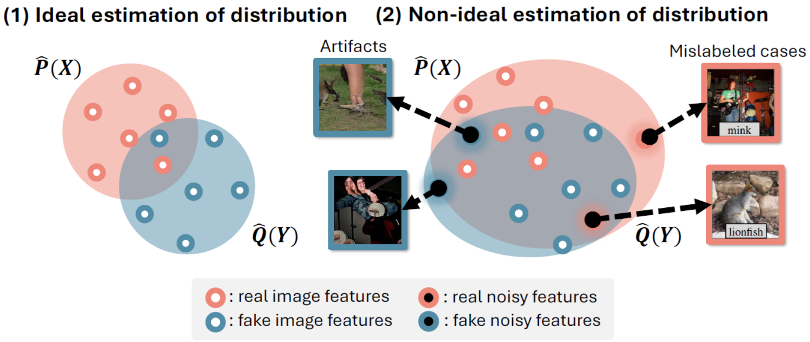


Confidence band for the persistent homology can be computed using the bootstrap algorithm.

1. Given a sample $X = \{x_1, \ldots, x_n\}$, compute the kernel density estimator $\hat{p}_h$.
2. Draw $X^* = \{x_1^*, \ldots, x_n^*\}$ from $X = \{x_1, \ldots, x_n\}$ (with replacement), and compute $\theta^* = \sqrt{nh^d}||\hat{p}_h^*(x) - \hat{p}_h(x)||_\infty$, where $\hat{p}_h^*$ is the density estimator computed using X^*.
3. Repeat the previous step B times to obtain $\theta_1^*, \ldots, \theta_B^*$
4. Compute $\hat{z}_\alpha = \inf\left\{q : \frac{1}{B}\sum_{j=1}^{B} I(\theta_j^* \geq q) \leq \alpha\right\}$
5. The $(1-\alpha)$ confidence band for $\mathbb{E}[p_h]$ is $\left[\hat{p}_h - \frac{\hat{z}_\alpha}{\sqrt{nh^d}}, \hat{p}_h + \frac{\hat{z}_\alpha}{\sqrt{nh^d}}\right]$.

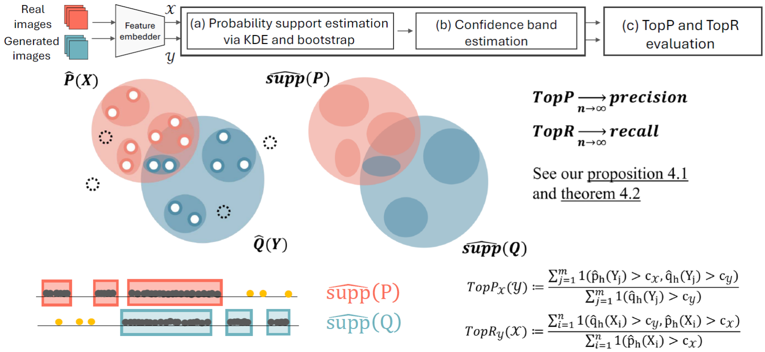
Existing evaluation metrics for generative models are vulnerable to noise.

- TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)
- To evaluate generative models, metrics compare the support of real image distributions and fake image distributions.
- Existing evaluation metrics tend to overestimate the support of the data distribution: vulnerable to noise



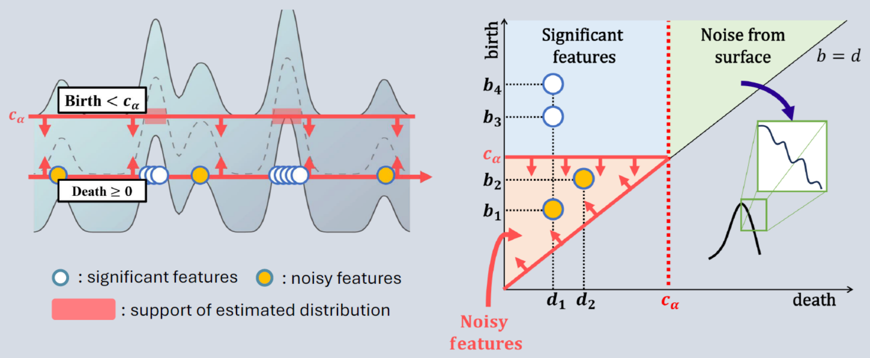
TopP&R robustly evaluates generative models by retaining only topologically and statistically significant features with confidence.

- TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)



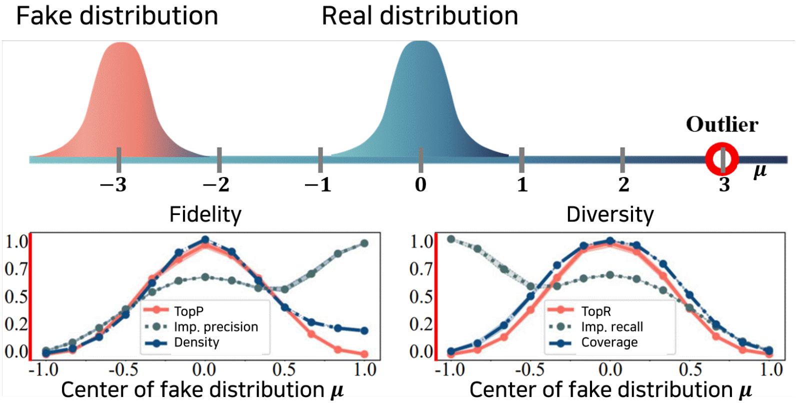
We find threshold c_α that selects statistically and topologically significant features.

- TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)



Experiments

- TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models (Kim, Jang, Kim, Yoo, 2024)



There are many programs for Topological Data Analysis.

- There are many programs for Topological Data Analysis: e.g., Dionysus, DIPHA, GUDHI, javaPlex, Perseus, PHAT, Ripser, TDA, TDAstats

R Package TDA provides an R interface for C++ libraries for Topological Data Analysis.

- website: `https://cran.r-project.org/web/packages/TDA/index.html`
- Author: Brittany Terese Fasy, Jisu Kim, Fabrizio Lecci, Clément Maria, David Milman, and Vincent Rouvreau.
- R is a programming language for statistical computing and graphics.
- R has short development time, while C/C++ has short execution time.
- R package TDA provides an R interface for C++ library GUDHI/Dionysus/PHAT, which are for Topological Data Analysis.

References I

Peter Bubenik. Statistical topological data analysis using persistence landscapes. *arXiv preprint arXiv:1207.6437*, 2012.

Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. The structure and stability of persistence modules. *arXiv preprint arXiv:1207.3674*, 2012.

Frédéric Chazal, Brittany Terese Fasy, Fabrizio Lecci, Alessandro Rinaldo, and Larry Wasserman. Stochastic convergence of persistence landscapes and silhouettes. In *Annual Symposium on Computational Geometry*, pages 474–483. ACM, 2014.

Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, 45(4):737–759, 2011.

H. Edelsbrunner and J. Harer. *Computational Topology: An Introduction*. Applied mathematics. American Mathematical Society, 2010. ISBN 9780821849255. URL `http://books.google.com/books?id=MDXa6gFRZuIC`.

References II

Felix Hensel, Michael Moor, and Bastian Rieck. A survey of topological machine learning methods. *Frontiers Artif. Intell.*, 4:681108, 2021. doi: 10.3389/frai.2021.681108. URL https://doi.org/10.3389/frai.2021.681108.

Kwangho Kim, Jisu Kim, Manzil Zaheer, Joon Sik Kim, Frédéric Chazal, and Larry Wasserman. PLLay: Efficient Topological Layer based on Persistent Landscapes. *arXiv e-prints*, art. arXiv:2002.02778, February 2020.

Pum Jun Kim, Yoojin Jang, Jisu Kim, and Jaejun Yoo. TopP&R: Robust Support Estimation Approach for Evaluating Fidelity and Diversity in Generative Models. *arXiv e-prints*, art. arXiv:2306.08013, June 2024. doi: 10.48550/arXiv.2306.08013.

Hengrui Luo, Alice Patania, Jisu Kim, and Mikael Vejdemo-Johansson. Generalized penalty for circular coordinate representation. *Foundations of Data Science*, 3(4):729–767, 2021.

References III

Mikael Vejdemo-Johansson, Florian T Pokorny, Primoz Skraba, and Danica Kragic. Cohomological learning of periodic motion. *Applicable algebra in engineering, communication and computing*, 26(1):5–26, 2015.

Thank you!

Statistical Inference for Persistent Homology

Featurization using Persistent Homology

R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

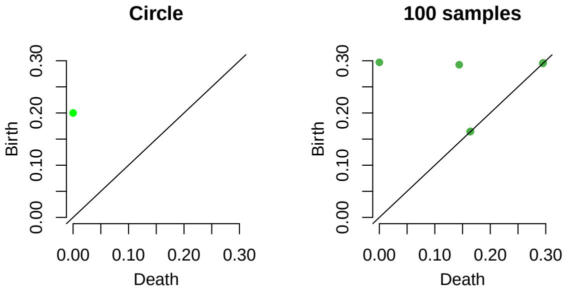
Bottleneck distance gives a metric on the space of persistent homology.

Definition

Let D_1, D_2 be multiset of points. Bottleneck distance is defined as

$$d_B(D_1, D_2) = \inf_{\gamma} \sup_{x \in D_1} \|x - \gamma(x)\|_\infty,$$

where γ ranges over all bijections from D_1 to D_2.



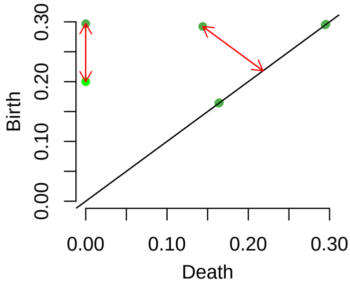
Bottleneck distance gives a metric on the space of persistent homology.

Definition

Let D_1, D_2 be multiset of points. Bottleneck distance is defined as

$$d_B(D_1, D_2) = \inf_{\gamma} \sup_{x \in D_1} \|x - \gamma(x)\|_\infty,$$

where γ ranges over all bijections from D_1 to D_2.



$$\sup_{x \in D_1} \|x - \gamma_1(x)\|_\infty = 0.1$$

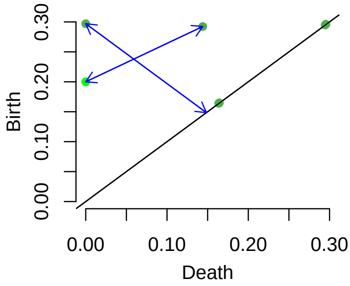
Bottleneck distance gives a metric on the space of persistent homology.

Definition

Let D_1, D_2 be multiset of points. Bottleneck distance is defined as

$$d_B(D_1, D_2) = \inf_{\gamma} \sup_{x \in D_1} \|x - \gamma(x)\|_\infty,$$

where γ ranges over all bijections from D_1 to D_2.



$$\sup_{x \in D_1} \|x - \gamma_2(x)\|_\infty = 0.15$$

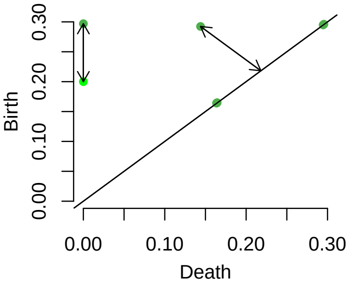
Bottleneck distance gives a metric on the space of persistent homology.

Definition

Let D_1, D_2 be multiset of points. Bottleneck distance is defined as

$$d_B(D_1, D_2) = \inf_{\gamma} \sup_{x \in D_1} \|x - \gamma(x)\|_\infty,$$

where γ ranges over all bijections from D_1 to D_2.



$$\inf_{\gamma} \sup_{x \in D_1} \|x - \gamma(x)\|_\infty = 0.1$$

Bottleneck distance can be controlled by the corresponding distance on functions: Stability Theorem.

Theorem

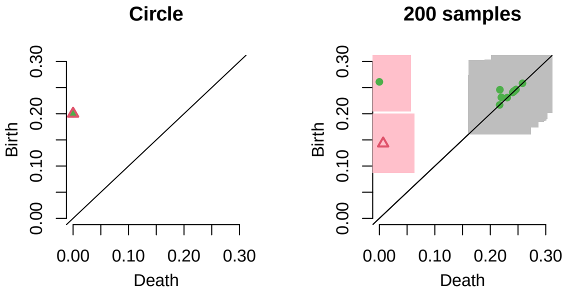
[Edelsbrunner and Harer, 2010][Chazal, de Silva, Glisse, and Oudot, 2012] Let $\mathbb{X}$ *be finitely triangulable space and* $f,\ g : \mathbb{X} \to \mathbb{R}$ *be two continuous functions. Let* $Dgm(f)$ *and* $Dgm(g)$ *be corresponding persistence diagrams. Then*

$$d_B(Dgm(f), Dgm(g)) \leq \|f - g\|_\infty.$$

Confidence band for the persistent homology is a random quantity containing the persistent homology with high probability.

Let M be a compact manifold, and $X = \{X_1, \cdots, X_n\}$ be n samples. Let f_M and f_X be corresponding functions whose persistent homology is of interest. Given the significance level $\alpha \in (0, 1)$, $(1 - \alpha)$ confidence band $c_n = c_n(X)$ is a random variable satisfying

$$\mathbb{P}\left(Dgm(f_M) \in \{\mathcal{D} : d_B(\mathcal{D}, Dgm(f_X)) \leq c_n\}\right) \geq 1 - \alpha.$$



Confidence band for the persistent homology can be obtained by the corresponding confidence band for functions.

From Stability Theorem, $\mathbb{P}\left(||f_M - f_X|| \le c_n\right) \ge 1 - \alpha$ implies

$$\mathbb{P}\left(d_B(Dgm(f_M), Dgm(f_X)) \le c_n\right) \ge \mathbb{P}\left(||f_M - f_X||_\infty \le c_n\right) \ge 1 - \alpha,$$

so the confidence band of corresponding functions f_M can be used for confidene band of persistent homologies $Dgm(f_M)$.

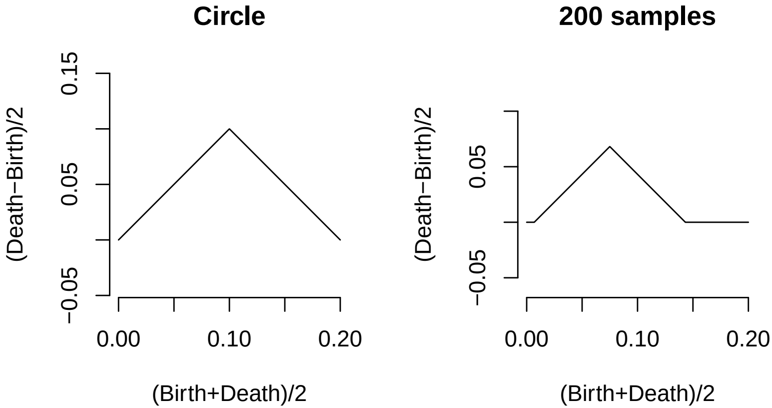
Statistical Inference for Persistent Homology

Featurization using Persistent Homology

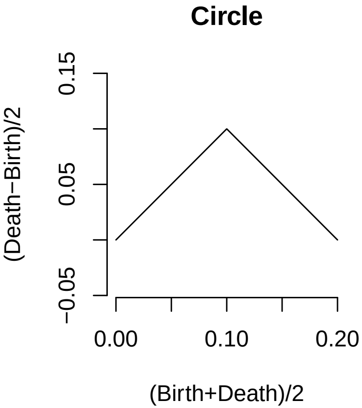
R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

Persistence Landscape of the underlying manifold can be inferred from Persistence Landscape of finite samples.



Confidence band for persistent homology quantifies the randomness of the persistence landscape.

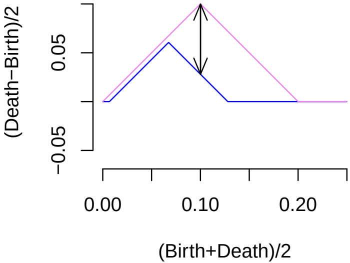


∞-landscape distance gives a metric on the space of persistence landscapes.

Definition

[Bubenik, 2012] Let D_1, D_2 be multiset of points, and λ_1 , λ_2 be corresponding persistence landscapes. ∞-landscape distance is defined as

$$\Lambda_\infty(D_1, D_2) = \|\lambda_1 - \lambda_2\|_\infty.$$



∞-landscape distance can be controlled by the corresponding distance on functions: Stability Theorem.

Theorem

Let $f, g : \mathbb{X} \to \mathbb{R}$ be two functions, and let $Dgm(f)$ and $Dgm(g)$ be corresponding persistent homologies. Then

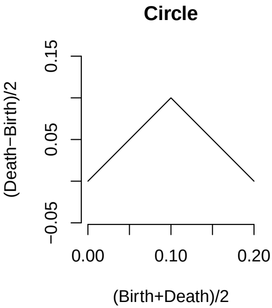
$$\Lambda_\infty(Dgm(f),\, Dgm(g)) \leq \|f - g\|_\infty.$$

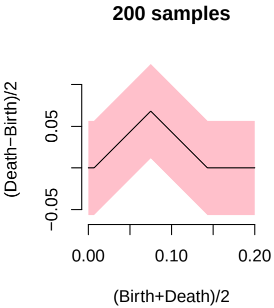
Confidence band for the persistence landscape can be computed using the bootstrap algorithm.

- Let λ_M and λ_X be persistence landscapes of the manifold M and samples X. From Stability Theorem, $\mathbb{P}(||f_M - f_X|| \leq c_n) \geq 1 - \alpha$ implies

$$\mathbb{P}(\lambda_X(t) - c_n \leq \lambda_M(t) \leq \lambda_X(t) + c_n \,\forall t) \geq \mathbb{P}(||f_M - f_X|| \leq c_n) \geq 1 - \alpha,$$

so the confidence band of corresponding functions f_M can be used for confidene band of the persistence landscape λ_M.





Confidence band for the persistence landscape can be computed using the bootstrap algorithm.

- Confidence band for the persistence landscape can be also computed using multiplier bootstrap; see [Chazal, Fasy, Lecci, Rinaldo, and Wasserman, 2014].

Statistical Inference for Persistent Homology

Featurization using Persistent Homology

R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

Statistical Inference for Persistent Homology

Featurization using Persistent Homology

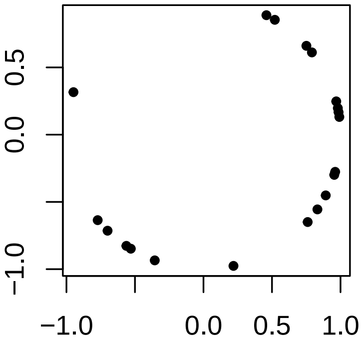
R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

R Package TDA provides a function to sample on a circle.

The function circleUnif() generates n sample from the uniform distribution on the circle in $\mathbb{R}^2$ with radius r.

```
circleSample <- circleUnif(n = 20, r = 1)
plot(circleSample, xlab = "", ylab = "", pch = 20)
```



R Package TDA provides distance functions and density functions over a grid.

Suppose $n = 400$ points are generated from the unit circle, and grid of points are generated.

```
X <- circleUnif(n = 400, r = 1)

lim <- c(-1.7, 1.7)
by <- 0.05
margin <- seq(from = lim[1], to = lim[2], by = by)
Grid <- expand.grid(margin, margin)
```

R Package TDA provides KDE function over a grid.

The Gaussian Kernel Density Estimator (KDE) $\hat{p}_h : \mathbb{R}^d \to [0, \infty)$ is defined as

$$\hat{p}_h(y) = \frac{1}{n(\sqrt{2\pi}h)^d} \sum_{i=1}^{n} \exp\left(\frac{-\|y - x_i\|_2^2}{2h^2} \right),$$

where h is a smoothing parameter.
The function kde() computes the KDE function $\hat{p}_h$ on a grid of points.

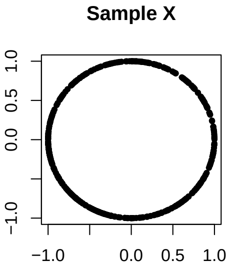
```r
h <- 0.3
KDE <- kde(X = X, Grid = Grid, h = h)

par(mfrow = c(1,2))
plot(X, xlab = "", ylab = "", main = "Sample X", pch = 20)
persp(x = margin, y = margin,
  z = matrix(KDE, nrow = length(margin), ncol = length(margin)),
  xlab = "", ylab = "", zlab = "", theta = -20, phi = 35, scale = FALSE,
  expand = 3, col = "red", border = NA, ltheta = 50, shade = 0.5,
  main = "KDE")
```

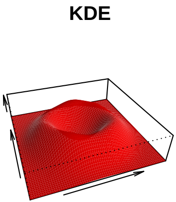
R Package TDA provides KDE function over a grid.

The Gaussian Kernel Density Estimator (KDE) $\hat{p}_h : \mathbb{R}^d \to [0, \infty)$ is defined as

$$\hat{p}_h(y) = \frac{1}{n(\sqrt{2\pi}h)^d} \sum_{i=1}^{n} \exp\left(\frac{-\|y - x_i\|_2^2}{2h^2} \right),$$

where h is a smoothing parameter.
The function kde() computes the KDE function $\hat{p}_h$ on a grid of points.





Statistical Inference for Persistent Homology

Featurization using Persistent Homology

R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

R Package TDA computes Persistent Homology over a grid.

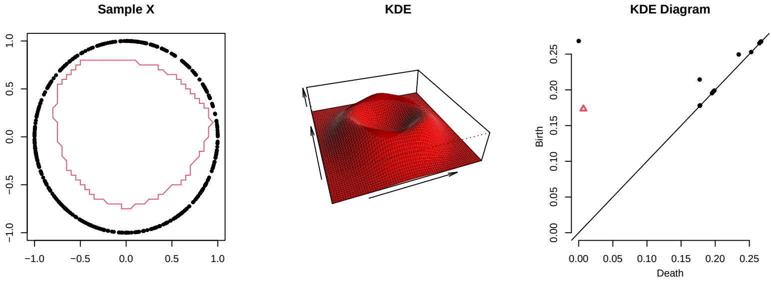
- The function gridDiag() computes the persistence diagram of sublevel (and superlevel) sets of the input function.
  - gridDiag() evaluates the real valued input function over a grid.
  - gridDiag() constructs a filtration of simplices using the values of the input function.
  - gridDiag() computes the persistent homology of the filtration.
- The user can choose to compute persistent homology using either C++ library GUDHI, Dionysus, or PHAT.

R Package TDA computes Persistent Homology over a grid.

```
DiagGrid <- gridDiag(X = X, FUN = kde, lim = c(lim, lim), by = by,
    sublevel = FALSE, library = "Dionysus", location = TRUE,
    printProgress = FALSE, h = h)

par(mfrow = c(1,3))
plot(X, xlab = "", ylab = "", main = "Sample X", pch = 20)
one <- which(DiagGrid[["diagram"]][, 1] == 1)
for (i in seq(along = one)) {
  for (j in seq_len(dim(DiagGrid[["cycleLocation"]][[one[i]]])[1])) {
    lines(DiagGrid[["cycleLocation"]][[one[i]]][j, , ], pch = 19, cex = 1,
        col = i + 1)
  }
}
persp(x = margin, y = margin,
  z = matrix(KDE, nrow = length(margin), ncol = length(margin)),
  xlab = "", ylab = "", zlab = "", theta = -20, phi = 35, scale = FALSE,
  expand = 3, col = "red", border = NA, ltheta = 50, shade = 0.9,
  main = "KDE")
plot(x = DiagGrid[["diagram"]], main = "KDE Diagram")
```

R Package TDA computes Persistent Homology over a grid.

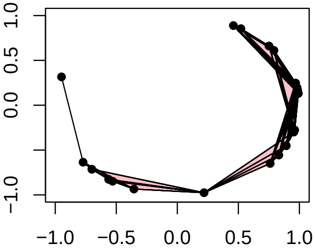
- The function gridDiag() computes the persistent homology of sublevel (and superlevel) sets of the input function.
  - gridDiag() evaluates the real valued input function over a grid.
  - gridDiag() constructs a filtration of simplices using the values of the input function.
  - gridDiag() computes the persistent homology of the filtration.
- The user can choose to compute persistent homology using either GUDHI, Dionysus, or PHAT.



R Package TDA computes Vietoris-Rips Persistent Homology.

- Vietoris-Rips complex consists of simplices whose pairwise distances of vertices are at most $2r$ apart, i.e.

$$\mathrm{Rips}(\mathcal{X}, r) = \{\{x_1, \ldots, x_k\} \subset \mathcal{X} : d(x_i, x_j) < 2r, \text{ for all } 1 \leq i, j \leq k\}.$$



- Rips filtration is formed by Rips complices with gradually increasing r.

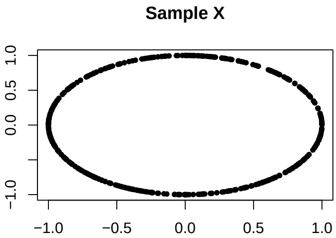
R Package TDA computes Vietoris-Rips Persistent Homology.

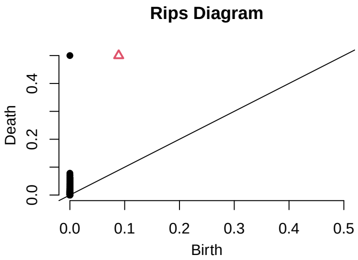
- The function ripsDiag() computes the persistence diagram of the Rips filtration built on top of a point cloud.
  - ripsDiag() constructs the Vietoris-Rips filtration using the data points.
  - ripsDiag() computes the persistent homology of the Vietoris-Rips filtration.
- The user can choose to compute persistent homology using either C++ library GUDHI, Dionysus, or PHAT.

```
DiagRips <- ripsDiag(X = X, maxdimension = 1, maxscale = 0.5,
    library = c("GUDHI", "Dionysus"), location = TRUE)

par(mfrow = c(1,2))
plot(X, xlab = "", ylab = "", main = "Sample X", pch = 20)
plot(x = DiagRips[["diagram"]], main = "Rips Diagram")
```

R Package TDA computes Vietoris-Rips Persistent Homology.

- The function ripsDiag() computes the persistence diagram of the Rips filtration built on top of a point cloud.
  - ripsDiag() constructs the Vietoris-Rips filtration using the data points.
  - ripsDiag() computes the persistent homology of the Vietoris-Rips filtration.
- The user can choose to compute persistent homology using either C++ library GUDHI, Dionysus, or PHAT.





R Package TDA computes Persistence Landscape.

- Let Λ_p be created by tenting each point $p = (x, y) = \left(\frac{b+d}{2}, \frac{d-b}{2}\right)$ representing a birth-death pair (b, d) in the persistence diagram D.
- The persistence landscape of D is the collection of functions

$$\lambda_k(t) = k \max_p \Lambda_p(t), \quad t \in [0, T], k \in \mathbb{N},$$

  where $k \max$ is the kth largest value in the set.
- The function landscape() evaluates the persistence landscape function $\lambda_k(t)$.

```
tseq <- seq(0, 0.2, length = 1000)
Land <- landscape(DiagGrid[["diagram"]], dimension = 1, KK = 1, tseq = tseq)

par(mfrow = c(1,2))
plot(x = DiagGrid[["diagram"]], main = "KDE Diagram")
plot(tseq, Land, type = "l", xlab = "(Birth+Death)/2",
     ylab = "(Death-Birth)/2", asp = 1, axes = FALSE, main = "Landscape")
axis(1); axis(2)
```

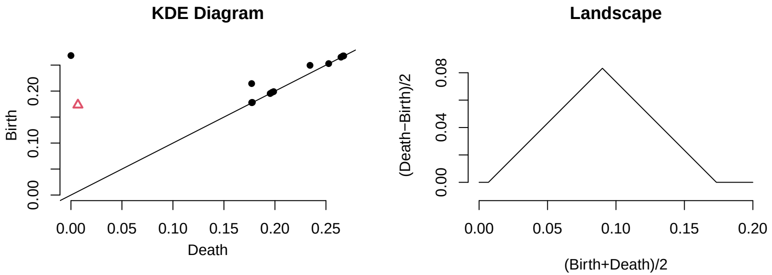
R Package TDA computes Persistence Landscape.

- Let Λ_p be created by tenting each point $p = (x, y) = \left(\frac{b+d}{2}, \frac{d-b}{2}\right)$ representing a birth-death pair (b, d) in the persistence diagram D.
- The persistence landscape of D is the collection of functions

$$\lambda_k(t) = k\max_p \Lambda_p(t), \quad t \in [0, T], k \in \mathbb{N},$$

where $k\max$ is the kth largest value in the set.

- The function landscape() evaluates the persistence landscape function $\lambda_k(t)$.



Statistical Inference for Persistent Homology

Featurization using Persistent Homology

R Package TDA: Statistical Tools for Topological Data Analysis

- Sample on manifolds, Distance Functions, and Density Estimators
- Persistent Homology and Persistence Landscape
- Statistical Inference on Persistence Homology and Persistence Landscape

R Package TDA computes the bootstrap confidence band for a function.

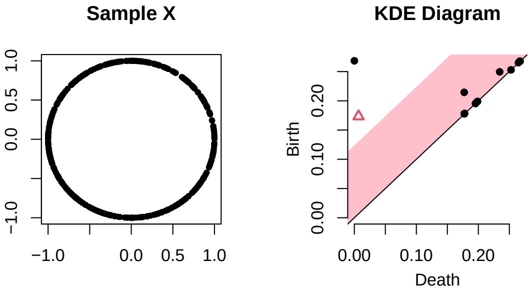
The function bootstrapBand() computes $(1 - \alpha)$ bootstrap confidence band for $\mathbb{E}[\hat{p}_h]$.

```
bandKDE <- bootstrapBand(X = X, FUN = kde, Grid = Grid, B = 20,
    parallel = FALSE, alpha = 0.1, h = h)
print(bandKDE[["width"]])

##        90%
## 0.06189347
```

R Package TDA computes the bootstrap confidence band for the persistent homology.

The $(1-\alpha)$ bootstrap confidence band for $\mathbb{E}[\hat{p}_h]$ is used as the confidence band for the persistent homology.

```
par(mfrow = c(1,2))
plot(X, xlab = "", ylab = "", main = "Sample X", pch = 20)
plot(x = DiagGrid[["diagram"]], band = 2 * bandKDE[["width"]],
     main = "KDE Diagram")
```



R Package TDA computes the bootstrap confidence band for the persistence landscape.

The $(1-\alpha)$ bootstrap confidence band for $\mathbb{E}[\hat{p}_h]$ is used as the confidence band for the persistence landscape.

```
par(mfrow = c(1,2))
plot(X, xlab = "", ylab = "", main = "Sample X", pch = 20)
plot(tseq, Land, type = "l", xlab = "(Birth+Death)/2",
     ylab = "(Death-Birth)/2", asp = 1, axes = FALSE, main = "200 samples")
axis(1); axis(2)
polygon(c(tseq, rev(tseq)), c(Land - bandKDE[["width"]],
        rev(Land + bandKDE[["width"]])), col = "pink", lwd = 1.5,
        border = NA)
lines(tseq, Land)
```

R Package TDA computes the bootstrap confidence band for the persistence landscape.

The $(1-\alpha)$ bootstrap confidence band for $\mathbb{E}[\hat{p}_h]$ is used as the confidence band for the persistence landscape.