Journal of Machine Learning Research 25 (2024) 1-74 Submitted 10/21; Revised 7/24; Published 8/24

Stable and Consistent Density-Based Clustering via Multiparameter Persistence

Alexander Rolle ALEXANDER.ROLLE@TUM.DE
Department of Mathematics
Technical University of Munich
Boltzmannstraße 3, 85748 Garching, Germany

Luis Scoccola LUIS.SCOCCOLA@MATHS.OX.AC.UK
Mathematical Institute
University of Oxford
Woodstock Road, Oxford OX2 6GG, United Kingdom

Editor: Sivan Sabato

Abstract

We consider the degree-Rips construction from topological data analysis, which provides a density-sensitive, multiparameter hierarchical clustering algorithm. We analyze its stability to perturbations of the input data using the correspondence-interleaving distance, a metric for hierarchical clusterings that we introduce. Taking certain one-parameter slices of degree-Rips recovers well-known methods for density-based clustering, but we show that these methods are unstable. However, we prove that degree-Rips, as a multiparameter object, is stable, and we propose an alternative approach for taking slices of degree-Rips, which yields a one-parameter hierarchical clustering algorithm with better stability properties. We prove that this algorithm is consistent, using the correspondence-interleaving distance. We provide an algorithm for extracting a single clustering from one-parameter hierarchical clusterings, which is stable with respect to the correspondence-interleaving distance. And, we integrate these methods into a pipeline for density-based clustering, which we call Persistable. Adapting tools from multiparameter persistent homology, we propose visualization tools that guide the selection of all parameters of the pipeline. We demonstrate Persistable on benchmark data sets, showing that it identifies multi-scale cluster structure in data.

Keywords: density-based clustering, topological data analysis, hierarchical clustering, multiparameter persistent homology, interleaving distance, vineyard

Contents

1. Introduction

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a probability density function, and let $\mathcal{S}(f)$ be its support. There is a one-parameter hierarchical clustering $H(f)$ of $\mathcal{S}(f)$ where, for $r > 0$, $H(f)(r)$ is the set of connected components of $\{x \in \mathcal{S}(f) : f(x) \geq r\}$. This is hierarchical in the sense that, if $r < r'$, then $H(f)(r)$ is a refinement of $H(f)(r')$. Following Hartigan (1975), we call $H(f)$ the *density-contour hierarchical clustering*. The central theoretical problem of density-based clustering is to approximate $H(f)$, given finite samples drawn from f.

A large amount of work has been done on the related problem of estimating the density f itself, given a finite sample. If one constructs an estimate $\hat{f}$ from a sample X, the "plug-in" approach would be to estimate $H(f)(r)$ by $H(\hat{f})(r)$, however this is not computationally-tractable (see Chaudhuri and Dasgupta (2010)). Instead, Cuevas et al. (2000) propose to construct a graph on X that encodes distance relations, and then estimate $H(f)(r)$ by taking the connected components of the induced subgraph on the vertices $\{x \in X : \hat{f}(x) \geq r\}$. The graph is the *Rips graph* for a fixed distance scale: for $x, y \in X$, there is an edge between x and y if $||x - y|| \leq s$, for some fixed $s > 0$. We call this approach the *plug-in* algorithm. See Related Work, below, for further references for this idea.

Another popular approach to density-based clustering is the *robust single-linkage* algorithm of Chaudhuri and Dasgupta (2010). This is a density-sensitive modification of the

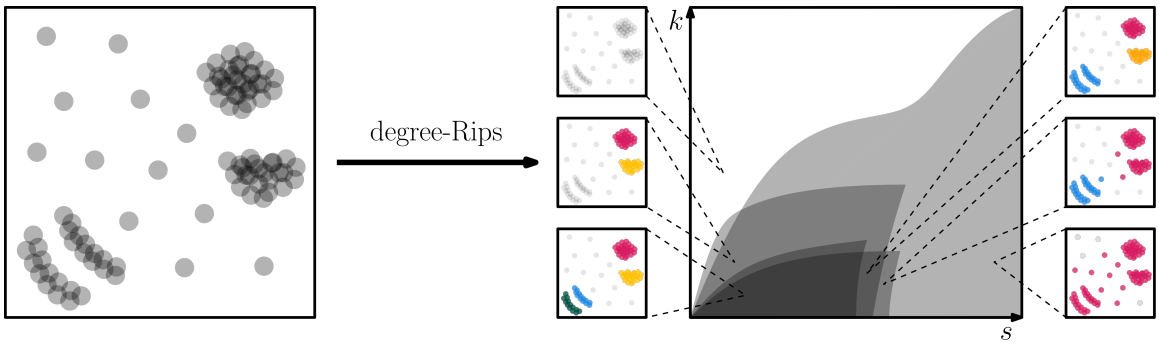

Figure 1: Degree-Rips is a two-parameter hierarchical clustering. The spatial parameter s controls the distance at which points are joined together: at larger values of s, more points are joined together. The density parameter k controls when data points enter the hierarchical clustering: at larger values of k, points must be in denser regions to enter. The robust single-linkage algorithm fixes k and lets s vary (taking a horizontal slice). The plug-in algorithm fixes s and lets k vary (taking a vertical slice).

single-linkage algorithm. Chaudhuri–Dasgupta prove that this method is *Hartigan consistent*: as the size of the sample tends to infinity, the robust single-linkage of a sample of f converges in probability to $H(f)$, using a criterion of Hartigan to compare the density-contour hierarchical clustering with a hierarchical clustering produced from a sample.

McInnes and Healy (2018) observed that the robust single-linkage algorithm is closely connected to the degree-Rips bifiltration (Lesnick and Wright, 2015; Blumberg and Lesnick, 2022) from topological data analysis (TDA). Degree-Rips should be of great interest to researchers in the field of clustering, as it simultaneously generalizes several important methods for density-based clustering. In its original formulation, degree-Rips is a two-parameter filtration of simplicial complexes, but in the setting of clustering, only the underlying graphs are relevant. In detail, let M be a finite metric space, let $s > 0$, and let $k \in (0, 1)$. Define a graph $G_{s,k}$ with vertex set $\{x \in M : |B(x, s)| \geq k \cdot |M|\}$, and with an edge between x and y if $d_M(x, y) \leq s$. Here, $B(x, s)$ is the open ball in M of radius s centered at x. These graphs form a two-parameter filtration, in the sense that there is an inclusion $G_{s,k} \subseteq G_{s',k'}$ for any $s' \geq s$ and any $k' \leq k$. We say that the *degree-Rips hierarchical clustering* of M is the two-parameter hierarchical clustering $\mathrm{DR}(M)$ with $\mathrm{DR}(M)_{s,k}$ given by the connected components of the graph $G_{s,k}$. See Fig. 1.

Both the robust single-linkage algorithm and the plug-in algorithm can be seen as one-parameter *slices* of the degree-Rips hierarchical clustering: if we fix k and let s vary, we recover the robust single-linkage of M; if we fix s and let k vary, we recover the plug-in algorithm, where the density estimate $\hat{f}$ is a kernel density estimate computed with the uniform kernel and bandwidth s, and the Rips graph is constructed also with parameter s.

Furthermore, the degree-Rips hierarchical clustering recovers the popular DBSCAN clustering algorithm (Ester et al., 1996). The clustering $\mathrm{DR}(M)(s, k)$ is exactly the DBSCAN*

clustering of M with respect to the spatial parameter s and the number-of-neighbors parameter $\lceil k \cdot |M| \rceil$. DBSCAN* is a minor modification of the original DBSCAN algorithm, defined by Campello et al. (2013).

For this paper, an important observation is that both robust single-linkage and the plug-in algorithm are *unstable*: small perturbations of the input can lead to large changes in the output. We make this statement precise later in the introduction. We therefore consider an alternative, which is very natural from the perspective of TDA. Rather than use slices of degree-Rips in which one parameter is fixed, we use slices in which both parameters vary.

We now summarize the main contributions of the paper. We elaborate on each point in the remainder of the introduction.

- We introduce the *correspondence-interleaving distance*, a metric for hierarchical clusterings.
- We introduce *kernel linkage*, a density-sensitive, multiparameter hierarchical clustering method that generalizes the degree-Rips hierarchical clustering described above.
- We prove that kernel linkage is stable with respect to the correspondence-interleaving distance and the Gromov–Hausdorff–Prokhorov distance on compact metric probability spaces. This implies that degree-Rips is stable, and that appropriate slices of kernel linkage and degree-Rips are also stable.
- We define a notion of consistency for density-based clustering using the correspondence-interleaving distance, which implies Hartigan consistency. We prove that taking appropriate slices of kernel linkage is consistent in this sense.
- We define the *persistence-based flattening algorithm*, which extracts a single clustering of the underlying data from a one-parameter hierarchical clustering, and prove that it is stable with respect to the correspondence-interleaving distance.
- Persistable is a pipeline for density-based clustering that integrates the algorithms defined in this paper. The Gromov–Hausdorff–Prokhorov stability theorem for kernel linkage implies theoretical guarantees for the entire pipeline, and it justifies a simple approximation scheme that makes it possible to apply the pipeline to large data sets. We describe how the design choices of Persistable are motivated by the results of this paper, we demonstrate Persistable on benchmark data sets, and we show that it identifies meaningful cluster structure in data. In another publication (Scoccola and Rolle, 2023), we described the implementation of Persistable.

1.1 The Correspondence-Interleaving Distance

In order to consider stability questions for hierarchical clustering methods, a natural approach is to use a notion of distance between hierarchical clusterings. For example, this is the approach taken by Carlsson and Mémoli (2010a), who prove a stability result for the single-linkage algorithm using the Gromov–Hausdorff distance from metric geometry. This is possible because the single-linkage of a metric space X defines an ultrametric θ_X on X, and so one can compare the outputs of single-linkage on X and Y by comparing (X, θ_X) and (Y, θ_Y) using Gromov–Hausdorff.

However, a hierarchical clustering of X does not define an ultrametric on X unless it is quite special (in which case we call it an *ultrametric hierarchical clustering*, Definition 12). In this paper, we formalize the notion of multiparameter hierarchical clustering in a way that is analogous to the multiparameter persistence modules from TDA (Carlsson and Zomorodian, 2009). We adapt the notion of interleaving from TDA (Chazal et al., 2009) to this setting, and use it to define the correspondence-interleaving distance (d_{CI}) between multiparameter hierarchical clusterings (Definition 17), which generalizes the Gromov–Hausdorff distance on ultrametric hierarchical clusterings (Proposition 21).

1.2 Stability

There are some choices baked in to the definition of degree-Rips that may not be optimal for some applications. So, we define a generalization: kernel linkage. Degree-Rips estimates the density of the data at a point x by counting the number of data points in a ball centered at x. From the perspective of density estimation, this can be seen as integrating the uniform kernel against the uniform measure defined by the input. One could just as well use other kernels for estimating density, and kernel linkage allows for this. It is also convenient to let kernel linkage take any compact metric probability space as input; if the input is a finite metric space as before, one gives it the uniform probability measure.

Our stability theorem for kernel linkage (Theorem 38) says that kernel linkage is uniformly continuous with respect to the Gromov–Hausdorff–Prokhorov distance on compact metric probability spaces, and the correspondence-interleaving distance on hierarchical clusterings. We note that one can replace the Prokhorov distance with the Wasserstein distance and get the same stability theorem for kernel linkage (Corollary 40). In the special case of degree-Rips, our stability theorem is as follows:

Result A (Corollary 39) *If M and N are finite metric spaces, then*

$$d_{\mathrm{CI}}(\mathrm{DR}(M), \mathrm{DR}(N)) \leq 2 \cdot d_{\mathrm{GHP}}(M, N)\,.$$

Requiring two finite metric spaces to be close in the Gromov–Hausdorff–Prokhorov distance amounts to requiring that they be close in the Gromov–Hausdorff distance (so that their metric geometry is similar), and that they be close in the Gromov–Prokhorov distance (so that their uniform measures are similar). We use this distance for our stability theorem because degree-Rips fails to be continuous with respect to the Gromov–Hausdorff distance or the Gromov–Prokhorov distance (see Remark 41). In order to get a continuity result, one must combine these two kinds of restrictions on the input.

We regard the use of Gromov–Hausdorff–Prokhorov as a strong assumption. But, it leads to correspondingly strong conclusions (uniform continuity in the case of kernel linkage, and Lipschitz-continuity in the special case of degree-Rips). It is useful to know the conditions that lead to these conclusions. For example, a key consequence of our stability theorem is a simple subsampling approximation algorithm for degree-Rips (see Section 3.4).

The Gromov–Hausdorff–Prokhorov stability of degree-Rips is in contrast to the robust single-linkage algorithm and the plug-in algorithm from above, which are discontinuous with respect to the Gromov–Hausdorff–Prokhorov distance, as we show in Section 3.3. In TDA, a standard method for extracting information from a two-parameter persistence

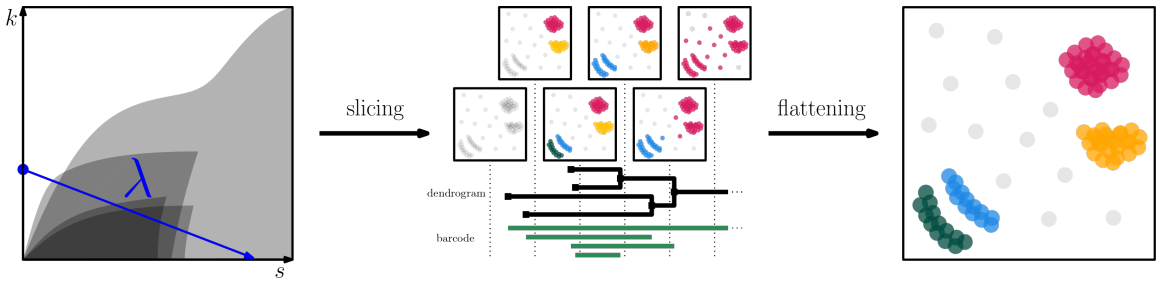

Figure 2: Restricting degree-Rips (Fig. 1) to a line λ with negative slope in the degree-Rips parameter space gives a one-parameter hierarchical clustering we call λ-linkage. The line λ lets both of the degree-Rips parameters s and k vary, in contrast to horizontal or vertical lines (see Fig. 1). This allows λ-linkage to capture multi-scale cluster structure in data, and it leads to better stability properties. The barcode is a visualizable summary of a one-parameter hierarchical clustering. The persistence-based flattening algorithm extracts a single clustering of the underlying data, guided by the barcode.

module is to take one-parameter slices (see Related Work, below, for references). However, one usually takes slices by lines through the parameter space that do not fix either of the parameters. Slices in which both parameters vary have two key advantages. First, they are multi-scale: they capture information across a range of values of both parameters. Second, these slices have better stability properties, since interleavings between multiparameter persistence modules restrict to interleavings between these slices.

The situation is completely analogous in the setting of hierarchical clustering. So, rather than use robust single-linkage or the plug-in algorithm for density-based clustering (which correspond to using horizontal or vertical slices of degree-Rips), we propose using slices of degree-Rips in which both parameters vary. In more detail, given a line λ in the plane with negative slope, restricting $\mathrm{DR}(M)$ to λ gives a one-parameter hierarchical clustering, which we call λ-linkage, denoted $\lambda\text{-}\mathtt{link}(M)$ (see Fig. 2). In contrast to robust single-linkage and the plug-in approach, λ-linkage is multi-scale, and it is stable with respect to the Gromov–Hausdorff–Prokhorov distance: as an immediate corollary of Result A, we obtain the following stability result.

Result B (Corollary 42) *Let λ be a line in the plane with slope $\sigma < 0$. If M and N are finite metric spaces, then*

$$d_{\mathrm{CI}}(\lambda\text{-}\mathtt{link}(M), \lambda\text{-}\mathtt{link}(N)) \leq \max(2|\sigma|, 1) \cdot d_{\mathrm{GHP}}(M, N)\,.$$

1.3 Consistency

Roughly speaking, a "consistency result" for density-based clustering usually says that, given a density function f and an algorithm for computing hierarchical clusterings of finite samples drawn from f, the output of the algorithm converges in probability to the

density-contour hierarchical clustering $H(f)$, as the sample size goes to infinity. To make this precise, one needs to specify what it means to converge in this context. There is a natural notion of consistency associated to the correspondence-interleaving distance, which we call CI-consistency; the idea is that the output of the algorithm should converge to $H(f)$ in the correspondence-interleaving distance, though in fact we require slightly more than this. CI-consistency is stronger than Hartigan consistency, so proving that an algorithm is CI-consistent implies that it is also Hartigan consistent. While this notion of consistency is novel, we remark that CI-consistency is similar in spirit to the notion of consistency of Eldridge et al. (2015). We prove the following consistency result for λ-`link`. In the statement, the notation $\overline{\lambda}$ indicates that the slice λ-`link` has been re-parameterized, using an explicit re-parameterization that only depends on λ; this can be dropped when considering Hartigan consistency, since Hartigan consistency is agnostic to the choice of parameterization.

Result C (Theorem 58) *The hierarchical clustering algorithm $\overline{\lambda}$-`link` is CI-consistent with respect to any continuous, compactly supported probability density function. In particular, λ-`link` is Hartigan consistent with respect to any such density function.*

1.4 Flattening a Hierarchical Clustering

For many applications, one needs a clustering of the input data, not a hierarchical clustering. We say that a *flattening* algorithm takes a hierarchical clustering, and returns a single clustering. An example of such a flattening algorithm is the ToMATo clustering algorithm (Chazal et al., 2013), which computes a flattening of the hierarchical clustering induced by a filtered graph. A major advantage of ToMATo is that its output can be understood in terms of the *barcode* of the input hierarchical clustering. Barcodes are key tools in TDA (Edelsbrunner et al., 2002; Carlsson et al., 2004; Ghrist, 2008); in this case, the barcode is a visualizable summary of the structure of a one-parameter hierarchical clustering (see Fig. 2). On a technical level however, a disadvantage of ToMATo is that its output depends on a choice of ordering of the vertices in the input graph, and in some use cases there may not be a clear way to make this choice.

We define the *persistence-based flattening algorithm* (Definition 82), an adaptation of the ToMATo algorithm that avoids the dependence on an ordering of the input. And, we prove that it is stable with respect to the correspondence-interleaving distance (Theorem 84).

1.5 Persistable

Combining the hierarchical clustering algorithm λ-`link` and the persistence-based flattening algorithm, we obtain a pipeline for density-based clustering with good stability properties. We call this pipeline *Persistable*.

Our stability theorems for degree-Rips and the persistence-based flattening algorithm imply theoretical guarantees for the entire pipeline (Corollary 85 and Corollary 86). The stability of degree-Rips also justifies a simple approximation scheme that makes it possible to apply Persistable to large data sets (e.g., the rideshare data in Section 7.2). This approximation scheme is not valid for related methods that are not Gromov–Hausdorff–Prokhorov stable, such as HDBSCAN (Campello et al., 2013) and DBSCAN (Ester et al., 1996).

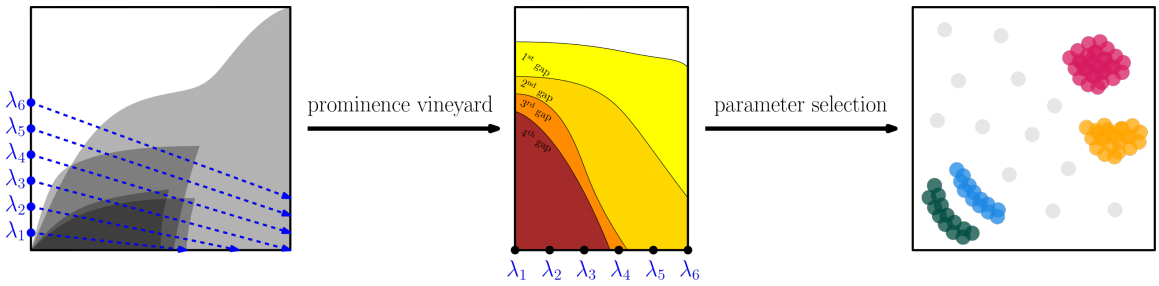

Figure 3: Parameter selection in the Persistable pipeline. The practitioner can choose a slice λ using the prominence vineyard. For each λ in a chosen family, the prominence vineyard plots the length of each bar in the barcode of λ-linkage. As λ varies continuously, the barcode varies continuously as well. The lengths of the bars trace out continuous curves: the top curve shows the length of the longest bar in the barcode of each slice, the second curve shows the length of the second longest, etc. Choosing a gap in the prominence vineyard leads to a clustering of the data. Larger gaps lead to more stable results.

Persistable includes interactive visualization tools that practitioners can use to choose all parameters in the pipeline. The key task for the practitioner is to choose the slice λ. Using a vineyard (Cohen-Steiner et al., 2006), one can see how the barcode of λ-`link` changes with the choice of λ (see Fig. 3). Moreover, one can see from the vineyard which choices of λ lead to particularly stable clusterings of the input data. We demonstrate Persistable, and this approach to parameter selection, on benchmark data sets, and we show that it provides results that capture meaningful cluster structure. These examples also demonstrate that Persistable can identify multi-scale cluster structure that is challenging for related algorithms, such as HDBSCAN.

1.6 Related Work

Distances between (one- and two-parameter) hierarchical clusterings have been studied by Carlsson and Mémoli (2010a,b). The correspondence-interleaving distance is a generalization of this work; see Section 2.2 for a discussion. The formigram distance, introduced by Kim and Mémoli (2018), can also be seen as a particular instance of the correspondence-interleaving distance. Eldridge et al. (2015) introduce the merge distortion metric for one-parameter hierarchical clusterings, which is closely related to the correspondence-interleaving distance.

Work of Rinaldo et al. (2012) and Chazal et al. (2013) addresses the stability of consistent hierarchical clustering methods. In their frameworks, stability is guaranteed when their assumptions on the underlying distribution are satisfied. In contrast, our stability results hold without distributional assumptions.

Combining density estimates and graphs that encode distance relations to estimate the density-contour hierarchical clustering has a long history, and several methods based on this idea have been proposed. Along with the work of Cuevas et al. (2000) already mentioned,

HC	Abbreviation for "hierarchical clustering"
Poset of clusterings $\mathsf{C}(X), \preceq$	Definition 3
The opposite P^{op} of a poset P	Definition 2
P-HC of X, $H : P \to \mathsf{C}(X)$ (with P a poset)	Definition 4
n-parameter HC	Definition 5
Density-contour HC $H(f)$ of a density f	Example 6
Single-linkage $\mathrm{SL}(M)$ of a metric space M	Example 7
Extension $\bar{H}$ of a HC H	Definition 8
Two HCs are $\vec{\varepsilon}$-interleaved	Definition 10
The interleaving distance d_{I}	Definition 11
θ_H, ultrametric HC	Definition 12
Correspondence $R \subseteq X \times Y$, $\pi_X : R \to X$, $\pi_Y : R \to Y$	Definition 15
Two HCs are $\vec{\varepsilon}$-interleaved w.r.t. R	Definition 16
The correspondence-interleaving distance d_{CI}	Definition 17
The Hausdorff distance d_{H}	Definition 19
The Gromov–Hausdorff distance d_{GH}	Definition 20
metric probability space	Definition 22
The uniform measure μ_M of M	Right below Definition 22
The uniform filtration Uni	Definition 23
The degree-Rips HC DR	Definition 24
A kernel K	Definition 25
The uniform kernel	Example 26
The local density estimate $(\mu_{\mathcal{M}} * K_s)$	Definition 27
The kernel filtration $\mathcal{M}_{[s,k]}$ of $\mathcal{M}$	Definition 29
The kernel linkage L^K or L	Definition 30
A curve γ in a poset, a slice H^γ	Definition 31
Robust single-linkage RSL	Example 32
The plug-in algorithm PI	Example 33
λ, $\lambda^{x,y}$, λ_{con}, λ_{cov}, λ-`link`, λ-linkage	Example 34
The Prokhorov distance d_{P}	Definition 35
The Gromov–Hausdorff–Prokhorov distance d_{GHP}	Definition 36
The Gromov–Hausdorff–Wasserstein distance	Right above Corollary 40
The closest point correspondence R_c	Definition 37
CI-consistency	Definition 50
The associated cluster tree $\mathcal{F}H$	Example 53
Hartigan consistency	Definition 55
$\overline{\lambda}$	Definition 57
Persistent cluster $\mathbf{C}$, underlying set $U(\mathbf{C})$, life, birth, length	Definition 60
Poset of persistent clusters PC	Definition 61
The set of leaves	Definition 62
Persistence-based pruning $H_{\geq\tau}$	Definition 63
finite, pointwise finite, essentially finite HC	Definition 64
The barcode $\mathcal{B}(H)$	Definition 68
The bottleneck distance d_{B}	Right above Proposition 69
Prominence diagram	Definition 75
Prominence diagram $\mathsf{Pr}(H)$ of a HC H	Definition 77 and Definition 79
Gap gap_n, gap size $\mathsf{gapsize}_n$	Definition 76
Gap $\mathsf{gap}_n(H)$, gap size $\mathsf{gapsize}_n(H)$ of a HC H	Notation 80
Persistence-based flattening PF	Definition 82
$R_X : H(\vec{r}) \to E(\vec{r} + \vec{v}\vec{\varepsilon})$	Notation 122
$B(x, r)$, the open ball of radius r centered at x	

Table 1: Definitions and frequently used notation.

see, e.g., Biau et al. (2007); Rinaldo and Wasserman (2010); Stuetzle and Nugent (2010); Chazal et al. (2013); Bobrowski et al. (2017). For another perspective, see Aragam et al. (2020); the authors also combine density estimation with the single-linkage algorithm, but approach the clustering problem using the idea of Bayes optimal partitions from parametric model-based clustering.

The consistency of robust single-linkage was first established by Chaudhuri and Dasgupta (2010), and then generalized to density functions supported on manifolds by Balakrishnan et al. (2013). Eldridge et al. (2015) introduced a notion of consistency that is closely related to CI-consistency, and, building on results of Chaudhuri–Dasgupta, they show that robust single-linkage is consistent in this sense.

Multiparameter hierarchical clustering is a topic of increasing interest, as multiparameter hierarchical clusterings have the potential to capture very rich cluster structure in data. See, e.g., Carlsson and Mémoli (2010b); Buchin et al. (2015); Kim and Mémoli (2018); Jardine (2020b); Bauer et al. (2020); Cai et al. (2020). We expect that the correspondence-interleaving distance will be useful for analyzing the properties of multiparameter hierarchical clustering methods in settings beyond this paper.

As mentioned earlier, our approach to taking slices of the degree-Rips hierarchical clustering is motivated by the standard practice in TDA of studying multiparameter persistence modules via one-parameter slices. See, for example, Cerri et al. (2009); Cagliari et al. (2010); Lesnick and Wright (2015); Landi (2018); Corbet et al. (2019); Vipond (2020); Carrière and Blumberg (2020).

When the input data is a finite subset of Euclidean space, and γ is a line with constant s-component, the slice of kernel linkage by γ recovers the connected components of the weighted Čech filtration introduced by Anai et al. (2019), when their parameter p is set to ∞. In particular, their stability result applies to this slice of kernel linkage.

As we have already mentioned, the persistence-based flattening we introduce is a modification of the ToMATo clustering algorithm (Chazal et al., 2013). The persistence-based flattening is defined using a pruning procedure we call the persistence-based pruning, which resembles the pruning of Kim et al. (2016).

Blumberg and Lesnick (2022) prove a stability result for the simplicial degree-Rips bifiltration, which we discuss in Remark 41. Jardine (2020a) has also proved results about the stability of degree-Rips, using a hypothesis involving configuration spaces, rather than a distance on metric probability spaces. Scoccola (2020, Section 6.5) shows that results in this paper can be lifted to the stability of the kernel filtration (Definition 29), which in particular implies that other topological invariants of this multi-filtration are Gromov–Hausdorff–Prokhorov stable.

2. Hierarchical Clustering

The notion of a hierarchical clustering (HC) has been formalized in a variety of ways in the clustering literature; see Carlsson and Mémoli (2010a) and references therein. In this section we introduce a new formalization of this notion, which, in particular, allows for HCs with multiple parameters. We introduce the *correspondence-interleaving* distance between HCs, which generalizes the distance on dendrograms introduced by Carlsson and Mémoli (2010a), and we develop its basic properties. In later sections of the paper, we will use

the correspondence-interleaving distance to formulate stability and consistency results for hierarchical clustering algorithms.

We also define the degree-Rips and kernel linkage hierarchical clusterings, as well as one-parameter slices of these constructions. These are the basis for all the clustering methods we consider in the rest of the paper.

2.1 The Definition of a Hierarchical Clustering

In order to define the notion of hierarchical clustering, we first define the notion of clustering. See Fig. 8 for an example.

Definition 1 *Let X be a set. A **clustering** of X is a set of non-empty, disjoint subsets of X. The elements of a clustering are called **clusters**.*

We will formalize hierarchical clusterings using the notion of a partially ordered set. There are many good references for this notion, for example (Chiossi, 2021, Ch. 2.2.2).

Definition 2 *A **partially ordered set (poset)** is a set P together with a binary relation $\preceq$ such that (1) for all $p \in P$, $p \preceq p$; (2) for all $p, q \in P$, if $p \preceq q$ and $q \preceq p$ then $p = q$; (3) for all $p, q, r \in P$, if $p \preceq q$ and $q \preceq r$ then $p \preceq r$. If P, Q are posets, and $f : P \to Q$ is a function, then f is **order-preserving** if for all $p, p' \in P$ with $p \preceq p'$, $f(p) \preceq f(p')$ in Q. If P is a poset, the **opposite poset** P^{op} is the poset with the same underlying set, and with $p \preceq p'$ in P^{op} if and only if $p \succeq p'$ in P.*

Definition 3 *Let X be a set. The **poset of clusterings** of X, denoted $\mathsf{C}(X)$, is the poset whose elements are the clusterings of X, and where $S \preceq T \in \mathsf{C}(X)$ if, for each cluster $A \in S$, there is a (necessarily unique) cluster $B \in T$ such that $A \subseteq B$.*

Definition 4 *Let P be a poset, and let X be a set. A **P-hierarchical clustering** of X is an order-preserving function $H : P \to \mathsf{C}(X)$.*

The notion of a P-hierarchical clustering generalizes the dendrograms of Carlsson and Mémoli (2010a, Section 3.1), where the indexing poset was taken to be $[0, \infty)$.

Definition 5 *Let X be a set, and let $n \geq 1$. An **n-parameter hierarchical clustering** of X is a P-hierarchical clustering $H : P \to C(X)$, where $P = I_1 \times \cdots \times I_n$ with I_j an interval of $\mathbb{R}$ or $\mathbb{R}^{\mathrm{op}}$ for all $1 \leq j \leq n$.*

Note that one-parameter HCs come in two flavors, depending on whether clusters merge as the real parameter increases or decreases; borrowing terminology from category theory, if $I \subseteq \mathbb{R}$ is an interval, we call an I-hierarchical clustering *covariant*, and if $I \subseteq \mathbb{R}^{\mathrm{op}}$, we call an I-hierarchical clustering *contravariant*. One-parameter HCs can be visualized by *dendrograms*: see Fig. 4. We now give two key examples of one-parameter HCs.

Example 6 *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a probability density function, and let $\mathcal{S}(f)$ be its support. Following Hartigan (1975), the **density-contour** hierarchical clustering $H(f)$ is the contravariant, $(0, \infty)^{\mathrm{op}}$-hierarchical clustering of $\mathcal{S}(f)$, where, for $r > 0$, $H(f)(r)$ is the set of connected components of $\{x \in \mathcal{S}(f) \,:\, f(x) \geq r\}$.*

Example 7 *Let M be a metric space. The **single-linkage** hierarchical clustering $\mathrm{SL}(M)$ (Sibson, 1973) is the covariant, $(0,\infty)$-hierarchical clustering of M, where, for $r > 0$, $\mathrm{SL}(M)(r)$ is the partition of M defined by the smallest equivalence relation $\sim_r$ on M with $x \sim_r y$ if $d_M(x,y) \leq r$. Single-linkage can also be defined in terms of the **Rips graph** $\mathrm{R}(M)$. For $r > 0$, let $\mathrm{R}(M)_r$ be the graph with vertex set M and with an edge between x and y if $d_M(x,y) \leq r$. Then $\mathrm{SL}(M)(r)$ is the partition of M by the vertex sets of the connected components of $\mathrm{R}(M)_r$.*

We now describe a way to *extend* any n-parameter hierarchical clustering $H : P \to \mathsf{C}(X)$ to an $\mathbb{R}^n$-hierarchical clustering $\bar{H} : \mathbb{R}^n \to \mathsf{C}(X)$. This will be useful when we consider distances between HCs, since we can compare any two n-parameter HCs, with possibly different indexing posets, by first extending them to $\mathbb{R}^n$-HCs, and then comparing the extensions. The idea is to first make H covariant in each parameter, by replacing any interval of $\mathbb{R}^{\mathrm{op}}$ in P with its negative, and then to extend H to all of $\mathbb{R}^n$ using the empty clustering $\emptyset$ (the minimum in $\mathsf{C}(X)$) and the clustering $\{X\}$ (the maximum in $\mathsf{C}(X)$).

Say $(I, \preceq) \subseteq \mathbb{R}^{\mathrm{op}}$ is an interval: as a set I is a real interval, and $a \preceq b$ in I if and only if $a \geq b$ as real numbers. Let $-I = \{-a : a \in I\}$. There is an isomorphism of posets $\rho_I : (-I, \preceq)^{\mathrm{op}} \to (I, \preceq)$ with $\rho_I(a) = -a$, and $(-I, \preceq)^{\mathrm{op}} = (-I, \leq)$ is an interval of $\mathbb{R}$.

Definition 8 *Say $P = I_1 \times \cdots \times I_n$ with each I_j an interval of $\mathbb{R}$ or $\mathbb{R}^{\mathrm{op}}$. Let P' be the poset obtained from P by replacing each interval $I_j \subseteq \mathbb{R}^{\mathrm{op}}$ with the interval $-I_j \subseteq \mathbb{R}$. Then we have an isomorphism of posets $\rho_P : P' \to P$. If X is a set, and $H : P \to \mathsf{C}(X)$ is an n-parameter hierarchical clustering of X, let $H' : P' \to \mathsf{C}(X)$ be $H \circ \rho_P$. The **extension** of H is the $\mathbb{R}^n$-hierarchical clustering $\bar{H} : \mathbb{R}^n \to \mathsf{C}(X)$ with*

$$\bar{H}(r) = \begin{cases} H'(r) & \text{if } r \in P' \\ \{X\} & \text{if } r \in \mathbb{R}^n \setminus P' \text{ and there is } p \in P' \text{ with } p < r \\ \emptyset & \text{else.} \end{cases}$$

2.2 The Correspondence-Interleaving Distance

The distances for hierarchical clusterings we consider are based on the notion of *interleaving*, which we have adapted from persistent homology (Chazal et al., 2009). In the HC setting, interleavings have a simple definition, which we now give.

Notation 9 *We write $\vec{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n) \geq \vec{0}$ if $\varepsilon_i \geq 0$ for $1 \leq i \leq n$.*

Definition 10 *Let H and E be n-parameter hierarchical clusterings of a set X, and let $\vec{\varepsilon} \geq \vec{0}$. We say that H and E are $\vec{\varepsilon}$-**interleaved** if, for all $\vec{r} \in \mathbb{R}^n$, we have $\bar{H}(\vec{r}) \preceq \bar{E}(\vec{r} + \vec{\varepsilon})$ and $\bar{E}(\vec{r}) \preceq \bar{H}(\vec{r} + \vec{\varepsilon})$ in $\mathsf{C}(X)$.*

Definition 11 *Let H and E be n-parameter hierarchical clusterings of a set X. Define the **interleaving distance***

$$d_{\mathrm{I}}(H, E) = \inf\{\varepsilon \geq 0 : H, E \text{ are } (\varepsilon, \ldots, \varepsilon)\text{-interleaved}\}.$$

In the special case of one-parameter HCs, the interleaving distance has a very concrete, alternative formulation. We give this now, in order to provide intuition for interleavings.

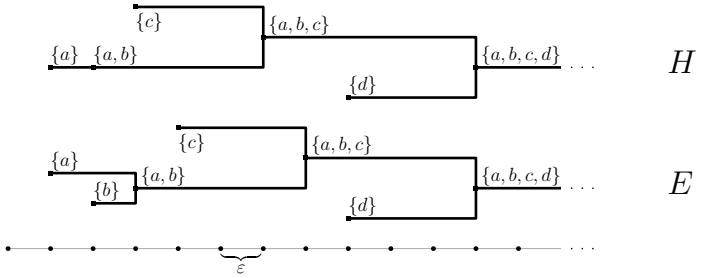

Figure 4: Two dendrograms representing one-parameter, covariant HCs H and E of the set $\{a, b, c, d\}$. The parameter values where points enter the HC, and where clusters merge, are perturbed by at most ε, so the HCs are ε-interleaved.

Definition 12 *Let $H : I \to \mathsf{C}(X)$ be a one-parameter hierarchical clustering of a set X. Define $\theta_H \colon X \times X \to [-\infty, \infty]$ by $\theta_H(x, y) = \inf\{r \in \mathbb{R} : \exists C \in \bar{H}(r),\ x, y \in C\}$. We say H is an **ultrametric hierarchical clustering** if $I = [0, \infty)$; for all $x \in X$, there is $r_x > 0$ such that for any r in the interval $[0, r_x)$, the clustering $H(r)$ contains the singleton cluster $\{x\}$; and there is $r \in [0, \infty)$ such that $H(r) = \{X\}$.*

For example, the single-linkage of a finite metric space is an ultrametric hierarchical clustering. If H is an ultrametric hierarchical clustering of X, then the function θ_H defines an ultrametric on X. See Carlsson and Mémoli (2010a) for a detailed discussion of this perspective. For H, E one-parameter HCs of X, we write $d_\infty(\theta_H, \theta_E) = \sup\{|\theta_H(x, y) - \theta_E(x, y)| : x, y \in X\}$.

Proposition 13 *If H and E are one-parameter hierarchical clusterings of a set X, then $d_{\mathrm{I}}(H, E) = d_\infty(\theta_H, \theta_E)$.*

The proof is elementary; see Appendix A.1. This formulation of the interleaving distance shows that, if H and E are ε-interleaved, then the parameter values where clusters are born and merge are perturbed by at most ε. See Fig. 4 for an example. We now give a simple example of a stability result that can be formulated using interleavings.

Proposition 14 *Let $f, g : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ be probability density functions with the same support. Then $d_{\mathrm{I}}(H(f), H(g)) = ||f - g||_\infty$.*

We give an elementary proof in Appendix A.1. This kind of stability result for real-valued functions is standard in topological data analysis. See, for example, Chazal et al. (2016, Example 4.3). We now extend the interleaving distance to HCs of different sets, using correspondences.

Definition 15 *A **correspondence** R between sets X and Y is given by a set $R \subseteq X \times Y$ such that the projections $\pi_X : R \to X$ and $\pi_Y : R \to Y$ are surjective.*

If $\psi : Y \to X$ is a function between sets, and $S = \{C_i\}$ is a clustering of X, then $\psi^*(S) = \{\psi^{-1}(C_i)\}$ is a clustering of Y. This defines an order-preserving map $\psi^* : \mathsf{C}(X) \to$

$\mathsf{C}(Y)$. If P is a poset and H is a P-hierarchical clustering of X, then $\psi^*(H) = \psi^* \circ H$ is a P-hierarchical clustering of Y.

Definition 16 *Let H and E be n-parameter hierarchical clusterings of sets X and Y respectively, let $R \subseteq X \times Y$ be a correspondence, and let $\vec{\varepsilon} \geq 0$. We say that H and E are **$\vec{\varepsilon}$-interleaved with respect to** R if $\pi_X^*(H)$ and $\pi_Y^*(E)$ are $\vec{\varepsilon}$-interleaved as n-parameter hierarchical clusterings of R.*

Definition 17 *Let H and E be n-parameter hierarchical clusterings of sets X and Y respectively. Define the **correspondence-interleaving distance***

$$d_{\mathrm{CI}}(H, E) = \inf_R \inf\{\varepsilon \geq 0 : H, E \text{ are } (\varepsilon, \ldots, \varepsilon)\text{-interleaved w.r.t. } R\}.$$

where the infimum is over all correspondences R between X and Y.

Aside from set-theoretic concerns, d_{CI} defines an extended-pseudo-metric on n-parameter hierarchical clusterings (see Appendix A.1 for the elementary proof):

Proposition 18 *The distance d_{CI} satisfies the following properties, for all n-parameter hierarchical clusterings:* (1) *for any H, $d_{\mathrm{CI}}(H, H) = 0$;* (2) *for any H, E, $d_{\mathrm{CI}}(H, E) = d_{\mathrm{CI}}(E, H)$;* (3) *for any H, E, F, $d_{\mathrm{CI}}(H, F) \leq d_{\mathrm{CI}}(H, E) + d_{\mathrm{CI}}(E, F)$.*

Using correspondences to extend the interleaving distance to HCs of different sets is inspired by the Gromov–Hausdorff distance from metric geometry (Burago et al., 2001, Chapter 7.3). In fact, there is a close connection between the correspondence-interleaving distance and the Gromov–Hausdorff distance. In their work on hierarchical clustering, Carlsson and Mémoli (2010a) use the Gromov–Hausdorff distance between the ultrametrics induced by HCs such as the single-linkage HC of a finite metric space. We now recall the definition of the Gromov–Hausdorff distance, and show that the correspondence-interleaving distance recovers this distance, in the special case of ultrametric hierarchical clusterings.

Definition 19 *Let A, B be compact subsets of a metric space M. The **Hausdorff distance** between A and B is $d_{\mathrm{H}}^M(A, B) = \inf\{\varepsilon > 0 : A \subseteq B^\varepsilon \text{ and } B \subseteq A^\varepsilon\}$, where, for any $W \subseteq M$, $W^\varepsilon = \{x \in M : \exists w \in W, d_M(x, w) < \varepsilon\}$.*

Definition 20 *Let M, N be compact metric spaces. The **Gromov–Hausdorff** distance is*

$$d_{\mathrm{GH}}(M, N) = \inf_{i,j} \; d_{\mathrm{H}}^Z(i(M), j(N)) \;,$$

where the infimum is taken over all isometric embeddings $i : M \to Z$ and $j : N \to Z$ into a common metric space Z.

Proposition 21 *Let H and E be ultrametric hierarchical clusterings of sets X and Y respectively. Then $d_{\mathrm{CI}}(H, E) = 2 \cdot d_{\mathrm{GH}}((X, \theta_H), (Y, \theta_E))$.*

Proof Let R be a correspondence between X and Y. One says that the *distortion* of R is $\mathrm{dis}(R) = \sup\{|\theta_H(x,x') - \theta_E(y,y')| : (x,y),(x',y') \in R\}$ (Burago et al., 2001, Definition 7.3.21). Then, one has $d_{\mathrm{GH}}((X,\theta_H),(Y,\theta_E)) = \frac{1}{2}\inf_R \mathrm{dis}(R)$, where the infimum is taken over all correspondences between X and Y (Burago et al., 2001, Theorem 7.3.25). Now, the proposition follows from the fact that, for any correspondence R, $\mathrm{dis}(R) = \inf\{\varepsilon \geq 0 : H, E \text{ are } \varepsilon\text{-interleaved w.r.t. } R\}$, which is Lemma 87. ■

2.3 Degree-Rips and Kernel Linkage

We now introduce *degree-Rips* and *kernel linkage*, the multiparameter hierarchical clustering methods that are the basis for all the clustering algorithms we consider in this paper. In the introduction, we described degree-Rips in the case that the input is a finite metric space. However, it is convenient to consider a natural generalization of this construction. Metric measure spaces (Gromov, 2007; Villani, 2009) are metric spaces together with a Borel measure (Dudley, 2002). Since the measures we consider will always be probability measures, we use the notion of metric probability space:

Definition 22 *A **metric probability space** consists of a metric space $\mathcal{M}$ together with a Borel probability measure $\mu_{\mathcal{M}}$ on $\mathcal{M}$.*

The degree-Rips hierarchical clustering we define in this section takes a metric probability space as input. If M is a finite metric space, and one equips M with the *uniform measure* μ_M, such that $\mu_M(A) = |A| \,/\, |M|$ for any $A \subseteq M$, then the degree-Rips hierarchical clustering of (M, μ_M) recovers the version of degree-Rips we described in the introduction. Unless otherwise stated, we equip finite metric spaces with the uniform measure.

Working in the generality of metric probability spaces has two main advantages. First, if f is a density function on Euclidean space, we can consider the degree-Rips hierarchical clustering of the metric probability space $(\mathcal{S}(f), \mu_f)$, where $\mathcal{S}(f)$ is the support of f, and μ_f is the probability measure defined by f. This construction plays a key role in the proof of our consistency theorem. Second, finite metric spaces with non-uniform measures are useful for computational purposes. In Section 3.4, we describe an approximation scheme for degree-Rips, in which a large input M (a finite metric space) is approximated by a small subset $N \subset M$, where N has a non-uniform measure that approximates the uniform measure of M.

Definition 23 *Let $\mathcal{M}$ be a metric probability space, and let $s, k > 0$. Let $\mathrm{Uni}(\mathcal{M})_{[s,k]} = \{x \in \mathcal{M} \,:\, \mu_{\mathcal{M}}(B(x,s)) \geq k\}$. Here and throughout the paper, $B(x,s)$ is the open ball in $\mathcal{M}$ of radius s centered at x. We have $\mathrm{Uni}(\mathcal{M})_{[s,k]} \subseteq \mathrm{Uni}(\mathcal{M})_{[s',k']}$ whenever $s' \geq s$ and $k' \leq k$. This forms a 2-parameter filtration of $\mathcal{M}$, which we call the **uniform filtration** of $\mathcal{M}$.*

Blumberg and Lesnick (2022) call this the "measure bifiltration". We combine the uniform filtration with single-linkage (Example 7) to define degree-Rips.

Definition 24 *Let $\mathcal{M}$ be a metric probability space. Define the **degree-Rips** hierarchical clustering of $\mathcal{M}$ as the 2-parameter hierarchical clustering:*

$$\begin{aligned}\mathrm{DR}(\mathcal{M}) : \mathbb{R}_{>0} \times \mathbb{R}_{>0}^{\mathrm{op}} &\to \mathsf{C}(\mathcal{M}) \\ (s,k) &\mapsto \mathrm{SL}\left(\mathrm{Uni}(\mathcal{M})_{[s,k]}\right)(s)\ .\end{aligned}$$

See Fig. 1 for an illustration of degree-Rips. As described in the introduction, we are motivated to consider the degree-Rips hierarchical clustering because of its close connection to well-established methods for data analysis, such as the DBSCAN clustering algorithm and the degree-Rips bifiltration. However, there are some choices baked in to the definition that may not be optimal for some applications. So, we will define a generalization of degree-Rips, which we call *kernel linkage*.

As motivation, notice that degree-Rips estimates the density near a point x by taking the measure of the ball $B(x,s)$. Equivalently, with respect to the measure $\mu_{\mathcal{M}}$, one integrates the *uniform kernel*, which is equal to one on this ball and vanishes elsewhere. One could just as well use another kernel when estimating density, as with kernel density estimators (Silverman, 1986). Second, notice that the definition of degree-Rips uses the s parameter twice: as the radius of the ball $B(x,s)$, and as the spatial parameter for single-linkage. It is not necessary for these two values to be equal, and in fact, the robust single-linkage algorithm (Example 32) allows these two values to differ by a constant factor. These two observations motivate the definition of kernel linkage.

In the setting of non-parametric density estimation, a *kernel* (Silverman, 1986, Ch. 4.2) quantifies local-ness; given a point x, a kernel quantifies the extent to which any other point x' is close to x. Since we are working with metric spaces, we will apply kernel functions to the distance between x and x'.

Definition 25 *A **kernel** is a non-increasing function $K : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ that is continuous from the right and such that $0 < \int_0^\infty K(r)\,\mathsf{d}r < \infty$.*

Note that, in particular, $K(0) > 0$ and $\lim_{r\to\infty} K(r) = 0$.

Example 26 *Many kernels used for density estimation are kernels in the above sense (see Remark 28). We will be particularly interested in $K = \mathbf{1}_{\{r<1\}} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, with $K(x) = 1$ if $x < 1$ and $K(x) = 0$ otherwise. We refer to this as the **uniform kernel**.*

Definition 27 *Let K be a kernel, and let $\mathcal{M}$ be a metric probability space. Define the **local density estimate** of a point $x \in \mathcal{M}$ at scale $s > 0$ as*

$$(\mu_{\mathcal{M}} * K_s)(x) := \int_{x' \in \mathcal{M}} K\left(\frac{d_{\mathcal{M}}(x,x')}{s}\right) \mathsf{d}\mu_{\mathcal{M}}.$$

Remark 28 *Let $\mathcal{M}$ be the metric probability space given by Euclidean space $\mathbb{R}^d$ equipped with the empirical measure defined by a finite set of points $Z \subset \mathbb{R}^d$. The formula for the local density estimate is*

$$(\mu_{\mathcal{M}} * K_s)(x) = \frac{1}{|Z|} \sum_{z \in Z} K\left(\frac{||x-z||}{s}\right)\ .$$

Based on the usual formula for kernel density estimates (Silverman, 1986, Section 4.2.1), one might expect a factor of $1/s^d$ here. However, we need our local density estimate to be monotonic in s, in order to define the kernel filtration, below. In effect, one can re-introduce the factor $1/s^d$ after taking one-parameter slices, and this is what we do to prove our consistency result (see Definition 57).

Definition 29 *Let K be a kernel, let $\mathcal{M}$ be a metric probability space, and let $s, k > 0$. Let $\mathcal{M}_{[s,k]} = \{x \in \mathcal{M} \,:\, (\mu_{\mathcal{M}} * K_s)(x) \geq k\}$. Note that, since K is non-increasing, we have $\mathcal{M}_{[s,k]} \subseteq \mathcal{M}_{[s',k']}$ whenever $s' \geq s$ and $k' \leq k$. This forms a 2-parameter filtration of $\mathcal{M}$, which we call the **kernel filtration** of $\mathcal{M}$.*

In analogy to the definition of degree-Rips, we combine the kernel filtration with single-linkage to define kernel linkage:

Definition 30 *Let K be a kernel, and let $\mathcal{M}$ be a metric probability space. Define the **kernel linkage** of $\mathcal{M}$ as the 3-parameter hierarchical clustering of $\mathcal{M}$:*

$$\begin{aligned} \mathrm{L}^K(\mathcal{M}) : \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}^{\mathrm{op}} &\to \mathsf{C}(\mathcal{M}) \\ (s,t,k) &\mapsto \mathrm{SL}\left(\mathcal{M}_{[s,k]}\right)(t)\ . \end{aligned}$$

If there is no risk of confusion, we suppress K from the notation, and write $\mathrm{L}(\mathcal{M})$.

To build intuition about kernel linkage, it is helpful to first think about degree-Rips, which is easier to visualize. We provide examples and visualizations in Section 7, where we describe Persistable. The interested reader may wish to look at these visualizations before reading the theoretical material in Section 3.

2.4 Slices of Kernel Linkage and λ-linkage

We now formally define the notion of a one-parameter slice of a hierarchical clustering. This is analogous to taking a one-parameter slice of a multiparameter persistence module; see the Related Work section of the introduction for references. Taking one-parameter slices of kernel linkage, one recovers well-known methods for density-based clustering.

Definition 31 *Let P be a poset. A **curve** in P is given by an interval I_γ of $\mathbb{R}$ or $\mathbb{R}^{\mathrm{op}}$, and an order-preserving function $\gamma : I_\gamma \to P$. If $H : P \to \mathsf{C}(X)$ is a P-hierarchical clustering of a set X, and $\gamma : I_\gamma \to P$ is a curve in P, then the **slice** of H by γ is the one-parameter hierarchical clustering $H^\gamma : I_\gamma \to \mathsf{C}(X)$ given by $H \circ \gamma$.*

As discussed in the introduction, some well-known methods for density-based clustering can be recovered by taking slices of kernel linkage.

Example 32 *The robust single-linkage algorithm of Chaudhuri and Dasgupta (2010) can be recovered by taking slices of kernel linkage. Let M be a finite metric space with $n = |M|$. Let $\kappa \in \mathbb{N}$ be the density threshold parameter of robust single-linkage, and let $\alpha > 0$ be its scale parameter. The robust single-linkage of M is $\mathrm{RSL}_{\kappa,\alpha}(M) = \mathrm{L}^K(M)^\gamma$, where we take K to be the uniform kernel, and γ is the covariant curve $\gamma\colon (0,\infty) \to \mathbb{R}^{\times 3}_{>0}$ with $\gamma(r) = (r, \alpha r, \kappa/n)$. This is a line through the kernel linkage parameter space, which fixes the density threshold parameter k, and allows the spatial parameters s and t to vary.*

Example 33 *If we fix the spatial parameters* s *and* t*, and allow the density threshold parameter* k *to vary, we recover the plug-in algorithm for density-based clustering, described in the introduction. See, for example, Cuevas et al. (2000); Chazal et al. (2013). In detail, let M be a finite metric space. For any $s, t > 0$, and for any kernel K, the plug-in hierarchical clustering of M is $\mathrm{PI}^K_{s,t}(M) = \mathrm{L}^K(M)^\gamma$ for the contravariant curve $\gamma\colon (0,\infty) \to \mathbb{R}^{\times 3}_{>0}$ with $\gamma(r) = (s,t,r)$.*

Slices in which one parameter is fixed, like in the previous two examples, lead to stability problems, as we show in Section 3.3. Moreover, such slices can struggle to capture multi-scale cluster structure in data (see the rideshare data in Section 7.2). So, for Persistable, we use lines in the kernel linkage parameter space in which all parameters vary.

Example 34 *For Persistable, we take slices of kernel linkage by a family of curves λ that we specify now. See Fig. 2. Each λ parameterizes a line in the (s,k)-space $\mathbb{R}_{>0} \times \mathbb{R}^{\mathrm{op}}_{>0}$, and we extend this to a curve in the (s,t,k)-space $\mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}^{\mathrm{op}}_{>0}$ by setting $s = t$. We specify a line λ by choosing an s-intercept $x > 0$ and a k-intercept $y > 0$. We write $\lambda^{x,y}$ if we need to specify the intercepts. Let $\sigma = -y/x$ be the slope of λ. If we parameterize λ with the k coordinate, we get the curve $\lambda_{\mathrm{con}} : (0,y)^{\mathrm{op}} \to \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}^{\mathrm{op}}_{>0}$ defined by $\lambda_{\mathrm{con}}(r) = ((r/\sigma) + x,\ (r/\sigma) + x,\ r)$. If we parameterize with the s coordinate, we get the curve $\lambda_{\mathrm{cov}} : (0,x) \to \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}^{\mathrm{op}}_{>0}$ defined by $\lambda_{\mathrm{cov}}(r) = (r,\ r,\ \sigma r + y)$.*

We say that the λ-linkage of a metric probability space $\mathcal{M}$ is the hierarchical clustering

$$\lambda\text{-}\mathtt{link}(\mathcal{M}) := \mathrm{L}^K(\mathcal{M})^\lambda$$

where K is the uniform kernel. Since we use the uniform kernel, the slices λ-`link` *are slices of the degree-Rips hierarchical clustering.*

When the input is a finite metric space, computing λ-linkage is similar to computing robust single-linkage. So, one can adapt the algorithms of McInnes and Healy (2018) to compute λ-linkage. This is what we do for our implementation of Persistable (Scoccola and Rolle, 2023).

3. Stability

In the introduction to this paper, we stated Result A, which says that the degree-Rips hierarchical clustering method is 2-Lipschitz, with respect to the Gromov–Hausdorff–Prokhorov distance on finite metric spaces, and the correspondence-interleaving distance on hierarchical clusterings. In Section 2.3 we defined the degree-Rips hierarchical clustering not just of a finite metric space, but in the generality of metric probability spaces. In this section, we prove that degree-Rips is 2-Lipschitz for compact metric probability spaces (this includes Result A as a special case). Furthermore, we consider the kernel linkage construction, also defined in Section 2.3, and show that it is uniformly continuous with respect to the Gromov–Hausdorff–Prokhorov and correspondence-interleaving distances.

3.1 Stability of Kernel Linkage

We begin by recalling the definition of the Gromov–Hausdorff–Prokhorov distance. See Villani (2009, p. 762) or Miermont (2009) (though note that Villani 2009 takes a sum

instead of the maximum of d_{H} and d_{P} in the definition). We discussed the Hausdorff distance in Section 2.2. The second ingredient we need is the Prokhorov distance (Dudley, 2002, Chapter 11.3).

Definition 35 *Let μ, ν be Borel probability measures on a metric space M. The* ***Prokhorov distance*** *between μ and ν is*

$$d_{\mathrm{P}}(\mu,\nu) = \inf\{\varepsilon > 0 \,:\, \mu(A) \leq \nu(A^{\varepsilon}) + \varepsilon \text{ and } \nu(A) \leq \mu(A^{\varepsilon}) + \varepsilon \text{ for all Borel sets } A \subseteq M\}\,.$$

Now, the Gromov–Hausdorff–Prokhorov distance is a metric on the set of isometry-equivalence classes of compact metric probability spaces (see, e.g., Miermont, 2009).

Definition 36 *Let $(\mathcal{M}, \mu_{\mathcal{M}}), (\mathcal{N}, \mu_{\mathcal{N}})$ be compact metric probability spaces. The* ***Gromov–Hausdorff–Prokhorov*** *distance between $(\mathcal{M}, \mu_{\mathcal{M}})$ and $(\mathcal{N}, \mu_{\mathcal{N}})$ is*

$$d_{\mathrm{GHP}}(\mathcal{M},\mathcal{N}) = \inf_{i,j} \left\{\max(d_{\mathrm{H}}^{Z}(i(\mathcal{M}), j(\mathcal{N})), d_{\mathrm{P}}(i_*\mu_{\mathcal{M}}, j_*\mu_{\mathcal{N}}))\right\}\,,$$

where the infimum is taken over all isometric embeddings $i : \mathcal{M} \to Z$ and $j : \mathcal{N} \to Z$ into a common metric space Z.

Before proving the stability of kernel linkage, we define a canonical correspondence between two compact metric spaces embedded in a common metric space.

Definition 37 *Let M and N be compact metric spaces, let Z be any metric space, and let $i : M \to Z$ and $j : N \to Z$ be isometric embeddings. Define the* ***closest point correspondence*** *$R_c \subseteq M \times N$, where $(x, y) \in R_c$ if and only if $d_Z(i(x), j(y)) = \min_{y' \in N} d_Z(i(x), j(y'))$ or $d_Z(i(x), j(y)) = \min_{x' \in M} d_Z(i(x'), j(y))$.*

Theorem 38 *Kernel linkage is uniformly continuous with respect to the Gromov–Hausdorff–Prokhorov distance on compact metric probability spaces, and the correspondence-interleaving distance. If kernel linkage is defined using the uniform kernel, then it is 2-Lipschitz.*

Proof Let K be a kernel. We prove the following: for every $\varepsilon > 0$, there exists $\delta > 0$ such that if $\mathcal{M}$ and $\mathcal{N}$ are compact metric probability spaces and $i : \mathcal{M} \to Z$ and $j : \mathcal{N} \to Z$ are isometric embeddings into a metric space Z with $d_{\mathrm{H}}(i(\mathcal{M}), j(\mathcal{N})), d_{\mathrm{P}}(i_*\mu_{\mathcal{M}}, j_*\mu_{\mathcal{N}}) < \delta$, then $\mathrm{L}^K(\mathcal{M})$ and $\mathrm{L}^K(\mathcal{N})$ are $(\varepsilon, \varepsilon, \varepsilon)$-interleaved with respect to the closest point correspondence $R_c \subseteq \mathcal{M} \times \mathcal{N}$.

Let $r' \in (0, K(0))$ and $\delta > 0$, and define $\delta_s = \frac{2\delta}{K^{-1}(r')}$ and $\delta_k = K(0)^2/r' - K(0) + K(0)\delta$. We now prove that if $\mathcal{M}$ and $\mathcal{N}$ are compact metric probability spaces and $i : \mathcal{M} \to Z$ and $j : \mathcal{N} \to Z$ are isometric embeddings with $d_{\mathrm{H}}(i(\mathcal{M}), j(\mathcal{N})), d_{\mathrm{P}}(i_*\mu_{\mathcal{M}}, j_*\mu_{\mathcal{N}}) < \delta$, then $\mathrm{L}^K(\mathcal{M})$ and $\mathrm{L}^K(\mathcal{N})$ are $(\delta_s, 2\delta, \delta_k)$-interleaved with respect to R_c. This implies the statement of the previous paragraph, by taking $r' \in (0, K(0))$ such that $K(0)^2/r' - K(0) < \varepsilon/2$, and $\delta > 0$ such that $2\delta/K^{-1}(r') < \varepsilon$, $2\delta < \varepsilon$, and $K(0)\delta < \varepsilon/2$.

It suffices to show that, for any $s, t > 0$ and $k > \delta_k$, we have relations in $\mathsf{C}(R_c)$:

$$\pi^*_{\mathcal{M}}\left(\mathrm{L}^K(\mathcal{M})\right)(s,t,k) \preceq \pi^*_{\mathcal{N}}\left(\mathrm{L}^K(\mathcal{N})\right)(s+\delta_s, t+2\delta, k-\delta_k)$$
$$\pi^*_{\mathcal{N}}\left(\mathrm{L}^K(\mathcal{N})\right)(s,t,k) \preceq \pi^*_{\mathcal{M}}\left(\mathrm{L}^K(\mathcal{M})\right)(s+\delta_s, t+2\delta, k-\delta_k)\,.$$

We show that the first relation holds, and the second relation follows from a symmetric argument. Let $(x, y) \in R_c$. If (x, y) belongs to a cluster of $\pi^*_{\mathcal{M}}(\mathrm{L}^K(\mathcal{M}))(s, t, k)$, then it belongs to a cluster of $\pi^*_{\mathcal{N}}(\mathrm{L}^K(\mathcal{N}))(s + \delta_s, t + 2\delta, k - \delta_k)$, by Lemma 89. Now, assume that (x, y) and $(x', y') \in R_c$ belong to the same cluster in $\pi^*_{\mathcal{M}}(\mathrm{L}^K(\mathcal{M}))(s, t, k)$. This means that $x \sim_t x'$ in $\mathcal{M}_{[s,k]}$. Since $|d_{\mathcal{M}}(x_1, x_2) - d_{\mathcal{N}}(y_1, y_2)| < 2\delta$ for every $(x_1, y_1), (x_2, y_2) \in R_c$, we have that $y \sim_{t+2\delta} y'$ in $\mathcal{N}_{[s+\delta_s, k-\delta_k]}$ as required.

It remains to consider the case where K is the uniform kernel. Then $K(0) = 1$, and, for every $r' \in (0, 1)$ we have $K^{-1}(r') = 1$, since $K^{-1} = K$. Letting $r' \to 1$, the interleaving we constructed above approaches a $(2\delta, 2\delta, \delta)$-interleaving, as needed. ■

Corollary 39 *If $\mathcal{M}$ and $\mathcal{N}$ are compact metric probability spaces, then*

$$d_{\mathrm{CI}}(\mathrm{DR}(\mathcal{M}), \mathrm{DR}(\mathcal{N})) \leq 2 \cdot d_{\mathrm{GHP}}(\mathcal{M}, \mathcal{N}) .$$

Proof Since degree-Rips is defined using the uniform kernel, it is 2-Lipschitz by Theorem 38. ■

Theorem 38 implies a similar result for the Gromov–Hausdorff–Wasserstein distance, which is defined just as in Definition 36, except one replaces the Prokhorov distance with the Wasserstein distance (Gibbs and Su, 2002, p. 424).

Corollary 40 *Kernel linkage is uniformly continuous with respect to the Gromov–Hausdorff–Wasserstein distance on compact metric probability spaces, and the correspondence-interleaving distance.*

Proof By Gibbs and Su (2002, Theorem 2), if μ and ν are probability measures on a compact metric space, then $d_{\mathrm{P}}(\mu, \nu)^2 \leq d_{\mathrm{W}}(\mu, \nu)$, where d_{W} denotes the Wasserstein distance. Now the corollary follows immediately from Theorem 38. ■

Remark 41 *We now discuss why we use the Gromov–Hausdorff–Prokhorov distance for analyzing the stability of the degree-Rips and kernel linkage hierarchical clusterings. Because these constructions are density-sensitive, they are not continuous with respect to the Gromov–Hausdorff distance, unlike single-linkage (Carlsson and Mémoli, 2010a). They are also not continuous with respect to the Gromov–Prokhorov distance. This was observed for the simplicial degree-Rips bifiltration by Blumberg and Lesnick (2022, Remark 3.8), using the* homotopy interleaving distance *on simplicial bifiltrations. The same example shows that the degree-Rips hierarchical clustering is not continuous with respect to the Gromov–Prokhorov distance on finite metric spaces (equipped with the uniform measure) and the correspondence-interleaving distance. However, as we have shown, if one uses the Gromov–Hausdorff–Prokhorov distance, degree-Rips is continuous, and even Lipschitz.*

We note that Blumberg and Lesnick (2022, Theorem 1.7) prove a Gromov–Prokhorov stability result for the simplicial degree-Rips bifiltration using homotopy interleavings. Necessarily, the conclusion is weaker than continuity. This stability result is complementary to our results. By working with the Gromov–Prokhorov distance, they make weaker assumptions on the input, and get correspondingly weaker conclusions.

3.2 Stability of Slices of Kernel Linkage

Interleavings between multiparameter hierarchical clusterings restrict to interleavings between slices, provided the slice does not fix any parameters. This is analogous to the behavior of interleavings and slices of multiparameter persistence modules; see the Related Work section of the Introduction for references.

Because the curves λ that we use for Persistable (Example 34) allow all parameters of kernel linkage to vary, we get Gromov–Hausdorff–Prokhorov stability for λ-`link` as an immediate corollary of Theorem 38.

Corollary 42 *Let $\lambda = \lambda^{x,y}$ for $x, y > 0$, and let σ be the slope of λ. Then, with respect to the Gromov–Hausdorff–Prokhorov distance on compact metric probability spaces and the correspondence-interleaving distance:*

1. λ_{con}-`link` *is* $\max(2|\sigma|, 1)$*-Lipschitz,*
2. λ_{cov}-`link` *is* $\max(|1/\sigma|, 2)$*-Lipschitz.*

Proof If $\mathcal{M}$ and $\mathcal{N}$ are compact metric probability spaces and $\delta > d_{\text{GHP}}(\mathcal{M}, \mathcal{N})$, then the proof of Theorem 38 shows that $\text{L}(\mathcal{M})$ and $\text{L}(\mathcal{N})$ are $(2\delta, 2\delta, \delta)$-interleaved with respect to the closest-point correspondence. Restricting this interleaving to the line λ, as in e.g. Landi (2018, Lemma 1), we get the required interleavings. ■

Based on this result, we say that λ_{con}-`link` and λ_{cov}-`link` are *stable with respect to the Gromov–Hausdorff–Prokhorov distance*. The slices λ_{con}-`link` and λ_{cov}-`link` are also stable in the choice of λ:

Proposition 43 *Let $\mathcal{M}$ be a metric probability space. Let $\lambda = \lambda^{x,y}$ with slope $\sigma = -y/x$ be defined by intercepts $x, y > 0$, and let $\lambda' = \lambda^{x',y'}$ with slope $\sigma' = -y'/x'$ be defined by intercepts $x', y' > 0$.*

1. $d_{\text{CI}}\left(\lambda_{\text{con}}\text{-}\texttt{link}(\mathcal{M}), \lambda'_{\text{con}}\text{-}\texttt{link}(\mathcal{M})\right) \leq \max\left(|y - y'|,\ |x - x'| \cdot \min(|\sigma|, |\sigma'|)\right)$.
2. $d_{\text{CI}}\left(\lambda_{\text{cov}}\text{-}\texttt{link}(\mathcal{M}), \lambda'_{\text{cov}}\text{-}\texttt{link}(\mathcal{M})\right) \leq \max\left(|x - x'|,\ |y - y'| \cdot \min(|1/\sigma|, |1/\sigma'|)\right)$.

Proof One can construct the required interleavings as in e.g. Landi (2018, Lemma 2). ■

3.3 Instability of Related Methods

In the introduction, we discussed two well-known methods for density-based clustering, which can be recovered by taking slices of kernel linkage; these are robust single-linkage (Example 32) and the plug-in algorithm (Example 33). In contrast to the hierarchical clusterings λ-`link` we use for Persistable, we now show that these methods are discontinuous with respect to the Gromov–Hausdorff–Prokhorov distance.

We begin with robust single-linkage. If one fixes the robust single-linkage parameters $\kappa \in \mathbb{N}$ and $\alpha > 0$, then one can think of robust single-linkage $\text{RSL}_{\kappa,\alpha}$ as a function that takes a

finite metric space as input and produces a one-parameter hierarchical clustering as output, and this function is discontinuous:

Proposition 44 *Let $\kappa \geq 2$ and $\alpha > 0$. With respect to the Gromov–Hausdorff–Prokhorov distance and the correspondence-interleaving distance,* $\mathrm{RSL}_{\kappa,\alpha}$ *is discontinuous.*

We prove this by giving a simple example in Appendix A.2. One could also formalize robust single-linkage differently, taking the density threshold parameter to be a ratio $k \in (0,1)$, and then letting $\mathrm{RSL}_{k,\alpha}(M) = \mathrm{L}(M)^{\gamma}$ for the covariant curve $\gamma\colon (0,\infty) \to \mathbb{R}^{\times 3}_{>0}$ with $\gamma(r) = (r, \alpha r, k)$. We show in Appendix A.2 that this variant is also discontinuous with respect to the Gromov–Hausdorff–Prokhorov distance.

In contrast to the stability of λ-`link` in λ (Proposition 43), changing the density threshold parameter κ of robust single-linkage can lead to arbitrarily large changes in the output:

Proposition 45 *Let $\kappa, \kappa' \in \mathbb{N}$ with $\kappa \neq \kappa'$, and let $\alpha > 0$. For any $D > 0$, there is a finite metric space M such that $d_{\mathrm{CI}}(\mathrm{RSL}_{\kappa,\alpha}(M), \mathrm{RSL}_{\kappa',\alpha}(M)) > D$.*

There is an analogous result for the variant of robust single-linkage that takes a density threshold $k \in (0,1)$ instead of κ. See Appendix A.2.

We now consider the plug-in algorithm. As before, if one fixes the parameters $s, t > 0$, then $\mathrm{PI}_{s,t}$ is a function that takes a finite metric space as input and produces a one-parameter hierarchical clustering as output, and we have the following:

Proposition 46 *Let $s, t > 0$, and let* PI *be defined using any kernel. With respect to the Gromov–Hausdorff–Prokhorov distance and the correspondence-interleaving distance,* $\mathrm{PI}_{s,t}$ *is discontinuous.*

We prove this in Appendix A.2 by giving a simple example. Finally, we consider the instability of the plug-in algorithm in its parameters. For a fixed metric probability space $\mathcal{M}$, Proposition 43 implies that (in both the covariant and contravariant versions) λ-`link`$(\mathcal{M})$ is continuous as a function from its parameter space $\{\lambda^{x,y}\}_{x,y>0}$ to the space of one-parameter hierarchical clusterings endowed with the correspondence-interleaving distance. Similarly, if we fix a finite metric space M, then the plug-in algorithm can be seen as a function $\mathrm{PI}_{-,-}(M)$ that takes input $s, t > 0$ and produces a one-parameter hierarchical clustering as output. However, this is not continuous (see Appendix A.2 for the proof):

Proposition 47 *Let* PI *be defined using any kernel, and let M be any finite metric space with $|M| \geq 2$. Then $\mathrm{PI}_{-,-}(M)$ is discontinuous, with respect to the Euclidean distance on $\mathbb{R}^2_{>0}$ and the correspondence-interleaving distance.*

3.4 Approximation of λ-linkage by Subsampling

Because degree-Rips and λ-linkage are Gromov–Hausdorff–Prokhorov stable, they admit a very simple approximation algorithm. For example, say $\lambda = \lambda_{\mathrm{con}}$, and we want to compute λ-`link`(M), where M is a finite metric space, equipped with the uniform measure. Say $N \subset M$ is a subsample, with $\varepsilon = d_{\mathrm{H}}(M, N)$. Then, by Proposition 48, one can compute a

probability measure on N such that $d_{\mathrm{GHP}}(M,N) \leq \varepsilon$, and therefore, by Corollary 42, we have $d_{\mathrm{CI}}(\lambda\text{-}\mathtt{link}(M), \lambda\text{-}\mathtt{link}(N)) \leq \max(2|\sigma|, 1) \cdot \varepsilon$.

Therefore, if we can find a small subsample of M that is close in the Hausdorff distance, we need only compute λ-$\mathtt{link}$ of the subsample in order to approximate λ-$\mathtt{link}(M)$. Persistable implements several subsampling methods, which can be used to get fast results on large data sets. We present an example in Section 7.

Proposition 48 *Let $(\mathcal{M}, \mu_{\mathcal{M}})$ be a finite metric probability space. Let $\mathcal{N} \subseteq \mathcal{M}$ be a subset and let $i : \mathcal{N} \to \mathcal{M}$ denote the inclusion. Choose any function $f : \mathcal{M} \to \mathcal{N}$ with the property that, for every $x \in \mathcal{M}$, the point $f(x) \in \mathcal{N}$ is a closest point of $\mathcal{N}$ to x. Define a probability measure on $\mathcal{N}$ by $\mu_{\mathcal{N}} = f_*(\mu_{\mathcal{M}})$. Then $d_{\mathrm{P}}(\mu_{\mathcal{M}}, i_*\mu_{\mathcal{N}}) \leq d_{\mathrm{H}}(\mathcal{M},\mathcal{N})$ and, in particular, $d_{\mathrm{GHP}}(\mathcal{M},\mathcal{N}) \leq d_{\mathrm{H}}(\mathcal{M},\mathcal{N})$.*

Proof Let $\varepsilon > 0$ be such that $\mathcal{M} \subseteq \mathcal{N}^\varepsilon$; it is enough to show that $d_{\mathrm{P}}(\mu_{\mathcal{M}}, i_*\mu_{\mathcal{N}}) \leq \varepsilon$. We prove that, for every $A \subseteq \mathcal{M}$ we have $\mu_{\mathcal{M}}(A) \leq \mu_{\mathcal{N}}(A^\varepsilon)$ and $\mu_{\mathcal{N}}(A) \leq \mu_{\mathcal{M}}(A^\varepsilon)$.

Note that $d_{\mathcal{M}}(x, f(x)) \leq \varepsilon$ for all $x \in \mathcal{M}$. It follows that $f^{-1}(A \cap \mathcal{N}) \subseteq A^\varepsilon$ and $f(A) \subseteq A^\varepsilon \cap \mathcal{N}$ for all $A \subseteq \mathcal{M}$. Note also that $i_*\mu_{\mathcal{N}}(B) = \mu_{\mathcal{M}}(f^{-1}(B \cap \mathcal{N}))$ for every $B \subseteq \mathcal{M}$, by definition of i_* and f_*. Let $A \subseteq \mathcal{M}$. Using the above, we get on one hand $i_*\mu_{\mathcal{N}}(A) = \mu_{\mathcal{M}}(f^{-1}(A \cap \mathcal{N})) \leq \mu_{\mathcal{M}}(A^\varepsilon)$. On the other hand, $A \subseteq f^{-1}(f(A)) \subseteq f^{-1}(A^\varepsilon \cap \mathcal{N})$, and thus $\mu_{\mathcal{M}}(A) \leq \mu_{\mathcal{M}}(f^{-1}(A^\varepsilon \cap \mathcal{N})) = i_*\mu_{\mathcal{N}}(A^\varepsilon)$. ■

4. Consistency

There is a natural notion of consistency for hierarchical clustering algorithms associated to the correspondence-interleaving distance. In this section, we define this, show that it implies Hartigan consistency, and show that λ-linkage is consistent with respect to the correspondence-interleaving distance.

In this section, unless otherwise stated, a hierarchical clustering will be a one-parameter hierarchical clustering (Definition 5).

4.1 Notions of Consistency of Hierarchical Clustering Algorithms

Definition 49 *A **hierarchical clustering algorithm** $\mathbb{A}$ with parameter space Θ is a mapping that assigns to each finite metric space M and each parameter $\theta \in \Theta$ a hierarchical clustering $\mathbb{A}^\theta(M)$ of M.*

We now define the notion of consistency associated to the correspondence-interleaving distance, using the density-contour hierarchical clustering $H(f)$ (Example 6) and the closest point correspondence R_c (Definition 37).

Definition 50 *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a probability density function with support $\mathcal{S}(f)$. A hierarchical clustering algorithm $\mathbb{A}$ with parameter space Θ is **CI-consistent** with respect to f if for every $n \in \mathbb{N}$ there exists a parameter $\theta_n \in \Theta$ such that, for every $\varepsilon > 0$ and X_n an i.i.d. n-sample of $\mathcal{S}(f)$ with distribution f, the probability that $\mathbb{A}^{\theta_n}(X_n)$ and $H(f)$ are ε-interleaved with respect to R_c goes to 1 as n goes to ∞.*

Remark 51 *In practice, one may want an explicit rule for choosing the parameters θ_n of Definition 50 as a function of n. Moreover, one may also want rates of convergence for the algorithm. Although we do not specifically address this in this paper, we mention that such results can be extracted from the proof of the consistency result Theorem 108 together with rates of convergence of samples in the Hausdorff distance (Cuevas and Rodríguez-Casal, 2004) and in the Prokhorov distance (Dudley, 1969).*

We now define Hartigan consistency, following Hartigan (1981).

Definition 52 *Let X be a set. A **cluster tree** of X is given by a family $\mathcal{T}$ of subsets of X with the property that whenever A and B are distinct elements of $\mathcal{T}$, then one of the following is true: $A \cap B = \emptyset$, $A \subseteq B$, or $B \subseteq A$. The elements of $\mathcal{T}$ are called **clusters**.*

Example 53 *Let $H : I \to \mathsf{C}(X)$ be a hierarchical clustering of a set X. We can define an associated cluster tree $\mathcal{F}H = \{C \in H(r) : r \in I\}$.*

Definition 54 *A **cluster tree algorithm** $\mathbb{A}$ with parameter space Θ is a mapping that assigns to each finite metric space M and each parameter $\theta \in \Theta$ a cluster tree $\mathbb{A}^\theta(M)$ of M.*

Definition 55 (cf. Hartigan, 1981) *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a probability density function with support $\mathcal{S}(f)$. A cluster tree algorithm $\mathbb{A}$ with parameter space Θ is **Hartigan consistent** with respect to f if for every $n \in \mathbb{N}$ there exists a parameter $\theta_n \in \Theta$ such that, given A and A' distinct elements of $H(f)(r)$ for some $r > 0$, and X_n an i.i.d. n-sample of $\mathcal{S}(f)$ with distribution f we have*

$$P(A_n \cap A'_n = \emptyset) \xrightarrow{n\to\infty} 1,$$

where A_n is the smallest cluster in $\mathbb{A}^{\theta_n}(X_n)$ that contains $A \cap X_n$, and A'_n is the smallest cluster in $\mathbb{A}^{\theta_n}(X_n)$ that contains $A' \cap X_n$.

The proof of the following result is in Appendix A.3.

Proposition 56 *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous and compactly supported probability density function. If a hierarchical clustering algorithm $\mathbb{A}$ is CI-consistent with respect to f, then the associated cluster tree algorithm $\mathcal{F}\mathbb{A}$ is Hartigan consistent with respect to f.*

4.2 Consistency of λ-linkage

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous and compactly supported probability density function with support $\mathcal{S}(f)$, and let μ_f be the probability measure defined by f. We now prove that the hierarchical clustering algorithm λ-`link` is CI-consistent with respect to f. The strategy is to construct an interleaving between $H(f)$ and the λ-`link` of the metric probability space $(\mathcal{S}(f), \mu_f)$. Then, the stability of λ-`link` implies that, for a sufficiently good sample X_n of f, the λ-`link` of X_n is a good approximation of the λ-`link` of $(\mathcal{S}(f), \mu_f)$.

However, in order to interleave λ-`link` and $H(f)$, we must first reparameterize λ-`link`, as discussed in Remark 28.

Definition 57 *Let $\lambda = \lambda^{x,y}_{\mathrm{con}}$ for $x, y > 0$ (see Example 34). For $s > 0$, we write v_s for the volume of a ball in $\mathbb{R}^d$ of radius s. Define an order-preserving function $\varphi : (0, y)^{\mathrm{op}} \to \mathbb{R}^{\mathrm{op}}_{>0}$ by $\varphi(r) = \frac{r}{v_{\lambda_s(r)}}$. Note that φ is a bijection; we write $\overline{\lambda} = \lambda \circ \varphi^{-1}$. For any metric probability space $\mathcal{M}$, we write $\overline{\lambda}$-$\mathtt{link}(\mathcal{M}) = \mathrm{L}^K(\mathcal{M})^{\overline{\lambda}}$, with K the uniform kernel.*

Theorem 58 *The hierarchical clustering algorithm $\overline{\lambda}$-$\mathtt{link}$ with parameter space $\{\lambda^{x,y}_{\mathrm{con}}\}_{x,y>0}$ is CI-consistent with respect to any continuous, compactly supported probability density function $f : \mathbb{R}^d \to \mathbb{R}$.*

This is a special case of Theorem 108, which is proved in Appendix A.3.

Remark 59 *For any $\lambda \in \{\lambda^{x,y}_{\mathrm{con}}\}_{x,y>0}$, λ-$\mathtt{link}$ and $\overline{\lambda}$-$\mathtt{link}$ produce the same underlying cluster tree. So, it follows from the preceding theorem that the algorithm λ-$\mathtt{link}$ with parameter space $\{\lambda^{x,y}_{\mathrm{con}}\}_{x,y>0}$ is Hartigan consistent with respect to any continuous, compactly supported probability density function $f : \mathbb{R}^d \to \mathbb{R}$.*

5. Structure of One-Parameter Hierarchical Clusterings

Barcodes are used in topological data analysis to summarize structural information about data (Edelsbrunner et al., 2002; Carlsson et al., 2004; Ghrist, 2008). Since they were first introduced, a rich theory has been developed for barcodes (see e.g., Chazal et al. 2016). Barcodes can be defined in many different contexts, and are used to summarize various geometric and topological properties of different kinds of data. In particular, one-parameter hierarchical clusterings have barcodes, and these are a key ingredient in the Persistable pipeline. See Fig. 5 for an example of a barcode.

Barcodes of hierarchical clusterings and related structures are a standard topic in topological data analysis (see e.g. Curry 2018; Cai et al. 2020). An important point in practice is that the so-called "elder rule" can be used to efficiently compute the barcode (Edelsbrunner and Harer, 2010, Ch. VII.1). In the setting of one-parameter hierarchical clusterings, it is possible to define the barcode and describe an algorithm for computing it without using any topological or algebraic machinery. So, for the benefit of readers who are not already familiar with topological data analysis, in this section we provide a definition of the barcode and describe some of its basic properties. Some readers may wish to skim this section on a first reading of the paper, and refer to it as needed when encountering barcodes.

5.1 The Poset of Persistent Clusters

We now describe a fundamental object associated to a hierarchical clustering, which we call the poset of persistent clusters. Picturing a hierarchical clustering as a dendrogram, the basic idea is to identify the edges in the dendrogram and define a partial order on them (see Fig. 6 for an illustration). To the best of our knowledge, the poset of persistent clusters was first defined by Kim et al. (2016, Appendix A), in the setting of the density-contour hierarchical clustering (though they did not use the terminology "persistent cluster"). This construction was also considered by McInnes and Healy (2018, Section 2.3) in the setting of robust single-linkage (Example 32), although phrased in the language of sheaf theory. Jardine (2020b, Section 1) defines an extension of this construction to 2-parameter hierarchical clusterings. We first define the notion of *persistent cluster*.

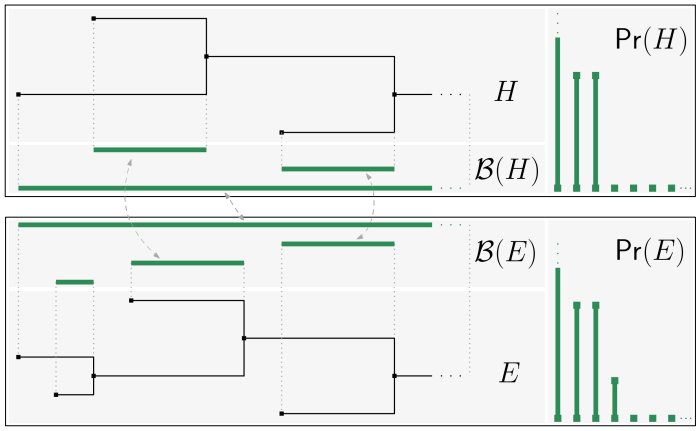

Figure 5: The **barcode** $\mathcal{B}(H)$ of a hierarchical clustering H is a collection of real intervals, called *bars* (displayed in green). Informally, the barcode is constructed using the following two rules: (1) If a new cluster enters H at parameter r, start a new bar with left endpoint r. (2) If two clusters merge at r, take the cluster that entered the hierarchy later (i.e. at a larger parameter value), and end its bar at r. The second rule is called the *elder rule*, since the elder bar survives. In the case of HCs induced by filtered graphs, we give pseudocode for this procedure (Algorithm 1). A **matching** is shown between the barcodes of H and E. The **prominence diagram** $\mathsf{Pr}(H)$ is simply the data of the lengths of the bars in $\mathcal{B}(H)$ (Section 5.4).

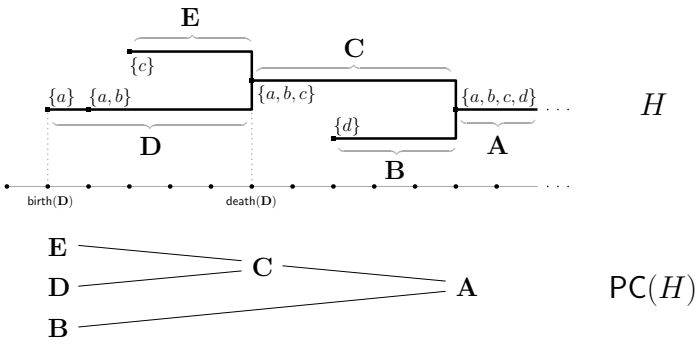

Figure 6: For the hierarchical clustering H, the poset of persistent clusters $\mathsf{PC}(H)$ is the poset with elements $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E}$, where $\mathbf{E}, \mathbf{D} < \mathbf{C}$, etc. The leaves of H are $\mathbf{E}, \mathbf{D}$, and $\mathbf{B}$. The parameter values $\mathsf{birth}(\mathbf{D})$ and $\mathsf{death}(\mathbf{D})$ are marked; the underlying set of $\mathbf{D}$ is $U(\mathbf{D}) = \{a, b\}$.

Definition 60 *Let X be a set. A **persistent cluster** $\mathbf{C}$ of X consists of an interval $\mathsf{life}(\mathbf{C}) \subseteq \mathbb{R}$ together with an order-preserving function $\mathbf{C} : \mathsf{life}(\mathbf{C}) \to \mathcal{P}(X)$, where $\mathcal{P}(X)$ is the power set of X ordered by inclusion. The **underlying set** of the persistent cluster $\mathbf{C}$ is $U(\mathbf{C}) = \cup_{r \in \mathsf{life}(\mathbf{C})} \mathbf{C}(r)$, and two persistent clusters are **disjoint** if their underlying sets are disjoint. Let $\mathsf{birth}(\mathbf{C}) = \inf \mathsf{life}(\mathbf{C})$, $\mathsf{death}(\mathbf{C}) = \sup \mathsf{life}(\mathbf{C})$, and $\mathsf{length}(\mathbf{C}) = \mathsf{death}(\mathbf{C}) - \mathsf{birth}(\mathbf{C})$.*

We remark that persistent clusters are, in particular, one-parameter hierarchical clusterings, so one can consider interleavings between persistent clusters.

Definition 61 *Let $I \subseteq \mathbb{R}$ be an interval and let H be an I-hierarchical clustering. The **poset of persistent clusters** of H, denoted $\mathsf{PC}(H)$, is the poset whose underlying set is the quotient set $\left(\coprod_{r \in I} H(r)\right) / \sim$ where:*

- *The set $\coprod_{r \in I} H(r)$ denotes the disjoint union of all clusterings as r varies in I.*
- *The relation $\sim$ is the symmetric closure of the following relation. For $r_1 \leq r_2$, $C_1 \in H(r_1)$, and $C_2 \in H(r_2)$, we have that C_1 and C_2 are related if and only if $C_1 \subseteq C_2$ and, for every $r_3 \in [r_1, r_2]$, there is exactly one cluster $C_3 \in H(r_3)$ such that $C_3 \subseteq C_2$.*

Let $\mathbf{C} \in \mathsf{PC}(H)$. The equivalence class $\mathbf{C}$ is naturally a persistent cluster in the sense of Definition 60, with $\mathsf{life}(\mathbf{C}) = \{r \in I : \exists C \in H(r) \text{ with } \mathbf{C} = [C]\}$ and such that, for $r \in \mathsf{life}(\mathbf{C})$, we let $\mathbf{C}(r) = C$, with $C \in H(r)$ the only cluster in $H(r)$ such that $[C] = \mathbf{C}$. With this in mind, we define the partial order on $\mathsf{PC}(H)$ by letting $\mathbf{C} \leq \mathbf{D}$ if $U(\mathbf{C}) \subseteq U(\mathbf{D})$.

The second poset axiom (Definition 2) for $\mathsf{PC}(H)$ is established in Lemma 110. The other poset axioms follow immediately from the definition.

Definition 62 *Let H be a one-parameter hierarchical clustering. The set of **leaves** of H, denoted $\mathsf{leaves}(H)$, is the set of minimal elements of $\mathsf{PC}(H)$.*

See Fig. 6 for an illustration of the poset of persistent clusters and of the leaves of a hierarchical clustering.

5.2 Tameness Conditions

We now define several tameness conditions that one can impose on hierarchical clusterings in order to get a notion of a barcode. The barcode is most naturally defined for *pointwise finite* HCs. However, some HCs of interest may not be pointwise finite (see Example 66). So, we introduce a notion of *essentially finite* HCs. While essentially finite HCs may not have barcodes, they at least have *prominence diagrams*, a closely related notion (see Section 5.4).

We begin by introducing the *persistence-based pruning* of an HC; see Fig. 7. This pruning procedure is similar in spirit to the pruning of Kim et al. (2016, Section 4.2): the persistence-based pruning shortens all branches by a chosen amount, making some of them disappear, while the pruning of Kim et al. (2016) removes all branches shorter than the chosen amount, and leaves the rest of the branches intact. In particular, the persistence-based pruning is stable with respect to interleavings (Proposition 121), while the pruning of

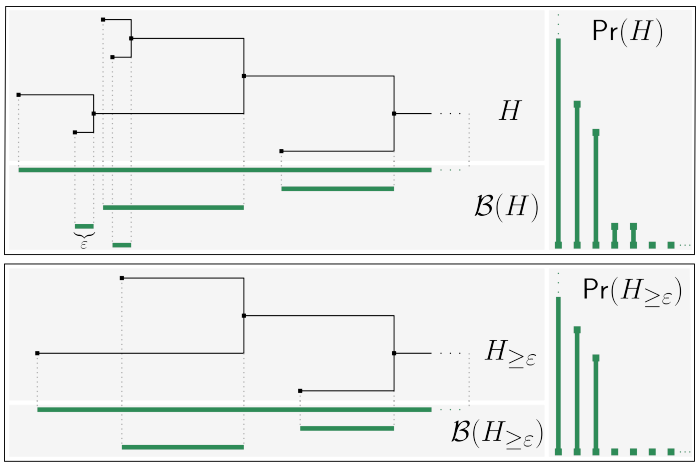

Figure 7: A hierarchical clustering H and the persistence-based pruning $H_{\geq\varepsilon}$. The barcode of H contains two short bars, reflecting short leaves of H. As the third-longest bar of H is much longer than the fourth-longest, $\mathsf{gapsize}_3(H)$ is large. Pruning H by an appropriate ε removes the short leaves.

Kim et al. (2016) is not. Let $I \subseteq \mathbb{R}$ be an interval and let $H : I \to \mathsf{C}(X)$ be a one-parameter hierarchical clustering of a set X. For $r \leq r' \in I$ we write $H(r \leq r') : H(r) \to H(r')$ for the function that takes $C \in H(r)$ to the unique $D \in H(r')$ such that $C \subseteq D$.

Definition 63 *Let H be an $\mathbb{R}$-hierarchical clustering of a set X. Let $\tau \geq 0$. The **persistence-based pruning** of H with respect to the threshold τ is the $\mathbb{R}$-hierarchical clustering $H_{\geq\tau}$ of X such that, for all $r \in I$, we let*

$$H_{\geq\tau}(r) \;:=\; \operatorname{Im} H(r-\tau \leq r) \;=\; \{C \in H(r) : \exists D \in H(r-\tau) \text{ with } D \subseteq C\}.$$

Definition 64 *An $\mathbb{R}$-hierarchical clustering H is **finite** if $\mathsf{PC}(H)$ is finite; **pointwise finite** if, for all $r \in \mathbb{R}$, the cardinality of $H(r)$ is finite; and **essentially finite** if $H_{\geq\tau}$ is finite for every $\tau > 0 \in \mathbb{R}$. A one-parameter hierarchical clustering is finite (respectively pointwise finite, essentially finite) if its extension (Definition 8) is finite (respectively pointwise finite, essentially finite).*

The above notion of finite hierarchical clustering was introduced by Kim et al. (2016). For readers familiar with the theory of persistence modules, we now briefly explain the connection between the other two tameness conditions and well-known tameness conditions for persistence modules. Given an $\mathbb{R}$-hierarchical clustering H and a choice of field $\mathbb{F}$, there is a persistence module $\mathbb{F}H$ generated by H (see Appendix A.4 for details). Now, an HC H is pointwise finite if and only if $\mathbb{F}H$ is pointwise finite-dimensional (Chazal et al., 2016, Section 3.8). Say that H is *bounded* if there is $s \leq t \in \mathbb{R}$ such that H is constant on $(-\infty, s)$ and (t, ∞). As we show in the proof of Lemma 119, assuming H is bounded, H is essentially finite if and only if $\mathbb{F}H$ is q-tame (Chazal et al., 2016, Section 3.8).

Example 65 *Any one-parameter hierarchical clustering H of a finite set X is finite, since $|PC(H)|$ is bounded above by the number of subsets of X, by Lemma 110.*

We prove the claims in the following example as Lemma 119.

Example 66 *For any $\lambda \in \{\lambda_{\text{con}}^{x,y}\}_{x,y>0}$ and any compact metric probability space $\mathcal{M}$, the hierarchical clustering λ-$\texttt{link}(\mathcal{M})$ is essentially finite. If $f : \mathbb{R}^d \to \mathbb{R}$ is continuous and compactly supported, then $H(f)$ is essentially finite. Note that, even if $f : \mathbb{R}^d \to \mathbb{R}$ is continuous and compactly supported, the hierarchical clustering $H(f)$ need not be pointwise finite, as the following simple example with $d = 1$ shows.*

Let $h(x) = x \cdot \sin(1/x) + 1$ if $x \neq 0$ and $h(0) = 1$. Let $g : \mathbb{R} \to \mathbb{R}_{\geq 0}$ be continuous, such that $g|_{(-1/2,1/2)} = c > 0$, $g|_{(-\infty,-1)\cup(1,\infty)} = 0$, and such that $f(x) := h(x)g(x)$ integrates to 1 on $\mathbb{R}$; such a function g can be taken to be piecewise linear, or even smooth. Then f is a pdf and $H(f)(c)$ has at least as many clusters as there are connected components in $\{x \in \mathbb{R} : x \cdot \sin(1/x) \geq 0, x \in (0, 1/2)\} = \{x \in \mathbb{R} : \sin(1/x) \geq 0, x \in (0, 1/2)\} = \{1/y : \sin(y) \geq 0, y > 2\}$, which clearly has countably many connected components.

5.3 The Barcode

We now define the barcode of a pointwise finite one-parameter HC. In order to give a definition that does not require any homology theory, we follow Carlsson and Zomorodian (2009), and define the barcode in terms of the rank invariant, also known as persistent Betti numbers (Edelsbrunner et al., 2002). For zero-dimensional homology, which is the relevant context for clustering, the rank invariant had already been introduced under the name of "size function" by Frosini (1990, 1992a,b); the multiparameter version of size functions is due to Biasotti et al. (2008).

Let $I \subseteq \mathbb{R}$ be an interval and let $H : I \to \mathsf{C}(X)$ be a one-parameter hierarchical clustering of a set X. Given $r \leq r' \in I$, define the **rank invariant** of H at $r \leq r'$ as

$$\text{rk}(H)(r \leq r') \;:=\; |\text{Im}\, H(r \leq r')| \;=\; |\{C \in H(r') : \exists D \in H(r) \text{ with } D \subseteq C\}|.$$

We interpret the quantity $\text{rk}(H)(r \leq r')$ as the number of clusters in $H(r')$ that have lived for at least $r' - r$ time. Equivalently, it is the maximum cardinality of a set of clusters in $H(r)$ that survive, as distinct clusters, until $H(r')$.

The rank invariant of a pointwise finite I-HC is, a priori, a function mapping comparable elements of I to natural numbers, and, as such, can be hard to visualize. Nevertheless, the function can be encoded as a multiset of subintervals of I in a unique way, as the following theorem asserts. In the generality of pointwise finite HCs, this theorem follows directly from a theorem of Crawley-Boevey; see Appendix A.4 for a proof of Theorem 67, as well as for the precise definition of multiset.

Theorem 67 *Let $I \subseteq \mathbb{R}$ be an interval and let H be a pointwise finite I-hierarchical clustering. There exists a unique multiset of non-empty intervals $\mathcal{B}(H) = \{B_j \subseteq I\}_{j \in J}$ with the property that, for all $r \leq r' \in \mathbb{R}$, we have $\text{rk}(H)(r \leq r') = |\{j \in J : r, r' \in B_j\}|$.*

Definition 68 *Let H be a pointwise finite one-parameter HC. The **barcode** of H is the multiset of intervals $\mathcal{B}(H)$ of Theorem 67.*

5.3.1 Bottleneck Stability of Barcodes

A standard way to compare barcodes is the *bottleneck distance*. For readers unfamiliar with this notion, we give an intuitive explanation, and refer to Bauer and Lesnick (2015, Section 3.2) for details. As in Definition 68, a *barcode* is a multiset of intervals of the real line. For barcodes $\mathcal{B}$ and $\mathcal{C}$, a *matching* between $\mathcal{B}$ and $\mathcal{C}$ is a bijection between a sub-multiset $\mathcal{B}'$ of $\mathcal{B}$ and a sub-multiset $\mathcal{C}'$ of $\mathcal{C}$. For $\delta \geq 0$, a δ-matching is a matching such that every interval of $\mathcal{B}$ and $\mathcal{C}$ with length greater than 2δ is matched (i.e., is in $\mathcal{B}'$ or $\mathcal{C}'$); and such that, when an interval $B \in \mathcal{B}'$ is matched with an interval $C \in \mathcal{C}'$, the endpoints of B and C differ by at most δ. Then, the bottleneck distance between $\mathcal{B}$ and $\mathcal{C}$ is

$$d_{\mathrm{B}}(\mathcal{B}, \mathcal{C}) = \inf\{\delta \geq 0 \mid \exists \text{ a } \delta\text{-matching between } \mathcal{B} \text{ and } \mathcal{C}\}.$$

We now give a proposition that relates the correspondence-interleaving distance between hierarchical clusterings with the bottleneck distance between their barcodes. This is just a translation of the well-known bottleneck stability theorem for persistence barcodes into the setting of hierarchical clusterings. The original versions of this stability result are due to d'Amico et al. (2003) in the case of zero-dimensional homology, and to Cohen-Steiner et al. (2007) and Chazal et al. (2009) in the case of homology in arbitrary dimension. We use the explicit formulation of Bauer and Lesnick (2015, Theorem 6.4) to easily derive the following proposition (see Appendix A.4 for the proof).

Proposition 69 *Let H and E be pointwise finite $\mathbb{R}$-hierarchical clusterings of sets X and Y. Let $\varepsilon \geq 0$. If H and E are ε-interleaved with respect to a correspondence between X and Y, then there exists an ε-matching between $\mathcal{B}(H)$ and $\mathcal{B}(E)$. In particular, $d_{\mathrm{B}}(\mathcal{B}(H), \mathcal{B}(E)) \leq d_{\mathrm{CI}}(H, E)$.*

The stability results for λ-linkage in Section 3.2 give upper bounds on the correspondence-interleaving distance between certain HCs. Combining these with Proposition 69, one gets upper bounds on the bottleneck distance between the barcodes of these HCs.

5.3.2 The Barcode of a Finite Hierarchical Clustering

We now describe the barcode of a finite hierarchical clustering in terms of its leaves. Fix an interval $I \subseteq \mathbb{R}$. Let H be a finite I-hierarchical clustering of a set X and let $\mathbf{C} \in \mathsf{leaves}(H)$. Define $H \setminus \mathbf{C}$ to be the I-hierarchical clustering of X with

$$(H \setminus \mathbf{C})(r) = \{D \in H(r) : [D] \neq \mathbf{C} \in \mathsf{PC}(H)\}.$$

Definition 70 *Let H be a finite I-hierarchical clustering. Let $\mathbf{C}, \mathbf{D} \in \mathsf{leaves}(H)$. We say that $\mathbf{D}$ **is born earlier** than $\mathbf{C}$ if, for every $r \in \mathsf{life}(\mathbf{C})$, there exists $r' \in \mathsf{life}(\mathbf{D})$ such that $r' \leq r$. A **minimal leaf** of H is a leaf $\mathbf{C}$ such that $\mathsf{length}(\mathbf{C})$ is minimal among leaves of H, and such that if $\mathbf{D}$ is another leaf of minimal length, then $\mathbf{D}$ is born earlier than $\mathbf{C}$.*

If H is a finite I-hierarchical clustering that is not constantly empty, then it admits some minimal leaf.

Proposition 71 *Let H be a finite I-hierarchical clustering. We define a sequence of I-hierarchical clusterings $H_0, \ldots, H_k$ ending at $k = |\mathsf{leaves}(H)|$. Define $H_0 := H$. Given H_i, let $\mathbf{C}_{i+1}$ be any minimal leaf of H_i and define $H_{i+1} := H_i \setminus \mathbf{C}_{i+1}$. Then, this sequence of hierarchical clusterings is well-defined and $\mathcal{B}(H) = \{\mathsf{life}(\mathbf{C}_i)\}_{1 \leq i \leq k}$.*

5.3.3 Computation of Barcodes Using the Elder Rule

We now describe an algorithm that uses the elder rule to compute the barcode of a hierarchical clustering induced by a finite filtered graph. The problem of computing the barcode of a filtered graph has been discussed extensively in the persistent homology literature. The textbook of Edelsbrunner and Harer (2010, Ch. VII.2) explains how the general persistence algorithm can be optimized in this case. Curry (2018) describes the elder rule for so-called Morse sets, which are an abstraction of the path components of a Morse function. Cai et al. (2020) describe the elder rule for so-called treegrams, which are hierarchical clusterings with a constructibility condition. The textbook of Dey and Wang (2022, Section 3.5.3) also describes how the general persistence algorithm can be adapted to the case of a filtered graph.

Despite the wealth of references for this topic, we give a description of the elder rule in our setting, for the convenience of readers who are not familiar with notions such as simplicial homology.

Definition 72 *A **finite filtered graph** is a pair (G, f), where G is a graph on a finite set X, instantiated as a set of vertices and edges, i.e., G is a set of subsets of X, with $\{x\} \in G$ for all $x \in X$, and there is an edge between x and y if and only if $\{x, y\} \in G$; and $f : G \to \mathbb{R}$ is a function such that if $\sigma_1 \subseteq \sigma_2$ in G, then $f(\sigma_1) \leq f(\sigma_2)$. A finite filtered graph (G, f) induces a covariant hierarchical clustering $H(G, f) : \mathbb{R} \to \mathsf{C}(X)$, with $H(G, f)(r)$ the set of connected components of the subgraph $f^{-1}((-\infty, r]) \subseteq G$.*

We are motivated to consider this case because the hierarchical clustering λ-$\texttt{link}(M)$ is induced by a finite filtered graph, for any finite metric space M. In more detail, let $\lambda = \lambda^{x,y}_{\text{cov}}$, and let $\sigma = -y/x$, as in Example 34. For $a \in M$, let $f(a) = \inf\{r > 0 : |B(a, r)| \geq (\sigma r + y) \cdot |M|\}$. For $a, b \in M$, let $f(\{a, b\}) = \min(x, \max(f(a), f(b), d_M(a, b)))$. Let G be a minimum spanning tree of the complete graph on M, weighted by f. Then λ-$\texttt{link}(M)$ is induced by (G, f).

Algorithm 1 Compute the barcode of the HC induced by a finite filtered graph

procedure Barcode(G, f)
  Order elements of G as $[\sigma_1, \ldots, \sigma_p]$ with $f(\sigma_i) \leq f(\sigma_{i+1})$ and $\sigma_i \subseteq \sigma_j \Rightarrow i \leq j$
  Let `conn_comp` $\leftarrow \{\}$ and `barcode` $\leftarrow \{\}$
  for $1 \leq i \leq p$ **do**
    if $\sigma_i = \{x\}$ **then** ▷ A vertex appears and a connected component is born
      `conn_comp` $\leftarrow$ `conn_comp` $\cup \{(\{x\}, f(\sigma_i))\}$
      `barcode` $\leftarrow$ `barcode` $\cup \{[f(\sigma_i), \infty)\}$
    else if $\sigma_i = \{x, y\}$ **then** ▷ An edge appears
      Let $(c, u), (d, v) \in$ `conn_comp` be such that $x \in c$ and $y \in d$
      if $c \neq d$ **then** ▷ Two distinct connected components are being merged
        `conn_comp` $\leftarrow$ (`conn_comp` $\setminus \{(c, u), (d, v)\}$) $\cup \{(c \cup d, \min(u, v))\}$ ▷ Merge components
        `barcode` $\leftarrow$ (`barcode` $\setminus \{[u, \infty), [v, \infty)\}$) $\cup \{[\min(u, v), \infty), [\max(u, v), f(\sigma_i))\}$ ▷ Elder rule
      end if
    end if
  end for
  Remove from `barcode` all intervals of the form $[t, t)$
  return `barcode`
end procedure

As is well-known (Edelsbrunner and Harer, 2010, Ch. VII.2), Algorithm 1 can be implemented to have time complexity in $O(p \log p)$, where p is the size of the input graph, that is, the number of vertices plus the number of edges. To see this, note that the operations between Line 4 and Line 15 of the algorithm, and specifically the check of Line 10, can be implemented with a union-find data structure, also known as a disjoint-set data structure (Tarjan, 1983), which keeps track of the connected components as the graph is filtered by f. Thus, the time complexity is dominated by that of Line 2, which sorts vertices and edges according to their f-value, and which has time complexity in $O(p \log p)$.

While Algorithm 1 requires an ordering of the elements of G, the output of the algorithm does not depend on this ordering:

Lemma 73 *The output of Algorithm 1 is independent of the ordering of the elements of G chosen in Line 2.*

Proposition 74 *Let (G, f) be a finite filtered graph. When given (G, f) as input, Algorithm 1 returns the barcode of $H(G, f)$.*

The proofs of Lemma 73 and Proposition 74 are in Appendix A.4.

5.4 The Prominence Diagram

In the theory of barcodes, the lengths of the bars play an important role. The length of a bar is called the "persistence" (Cohen-Steiner et al. 2010, Edelsbrunner and Harer 2010, Ch. VII.1) or the "prominence" (Chazal et al., 2013) of the bar. Following Chazal et al. (2013) we adopt the term prominence, which avoids confusion with the notion of persistence diagram (Edelsbrunner and Harer, 2010, Ch. VII.1). We will sort all the prominences of a barcode in descending order, and call the result the *prominence diagram*. This construction was considered by Bauer et al. (2017) in the setting of mode hunting, where the sorted list of prominences (divided by two) was called the "persistence signature". Our main motivation for considering the prominence diagram is the persistence-based flattening algorithm we introduce in Section 6. As we explain there, parameter selection for this algorithm involves choosing a cut-off between long and short bars in a barcode, and Persistable provides visualizations of prominence diagrams to guide this choice.

The proofs of all results in this sub-section are in Appendix A.4.

Definition 75 *A **prominence diagram** consists of a non-increasing function $P : \mathbb{N} \to [0, \infty]$ with $P(j) \to 0$ as $j \to \infty$. Define a distance d_∞ between prominence diagrams by letting*

$$d_\infty(P, Q) = \sup_{i \in \mathbb{N}} |P(i) - Q(i)|,$$

for all $P, Q : \mathbb{N} \to [0, \infty]$, with the convention that $|\infty - x| = |x - \infty|$ is equal to ∞ if $x \in [0, \infty)$ and to 0 if $x = \infty$.

Definition 76 *Let $n \in \mathbb{N}_{\geq 1}$. The **$n^{\text{th}}$ gap** of a prominence diagram $P : \mathbb{N} \to [0, \infty]$ is the (possibly empty) interval $\mathsf{gap}_n(P) = (P(n), P(n-1)) \subseteq [0, \infty]$. The **$n^{\text{th}}$ gap size** is the length of the gap, $\mathsf{gapsize}_n(P) = P(n-1) - P(n)$.*

Let H be a finite $\mathbb{R}$-hierarchical clustering. It follows from Proposition 71 that the barcode $\mathcal{B}(H) = \{B_j\}_{j\in J}$ of H contains finitely many intervals. Thus, $\{\mathsf{length}(B_j) \in [0,\infty]\}_{j\in J}$ is a finite multiset of elements of $[0,\infty]$.

Definition 77 *Let H be a finite $\mathbb{R}$-hierarchical clustering and let $\{\ell_0,\ldots,\ell_k\} \subseteq [0,\infty]$ denote the lengths of the intervals in $\mathcal{B}(H)$, with repetitions and ordered from largest to smallest. The **prominence diagram** of a finite $\mathbb{R}$-hierarchical clustering H is the decreasing sequence $\mathsf{Pr}(H) : \mathbb{N} \to [0,\infty]$ such that $\mathsf{Pr}(H)(i) = \ell_i$ if $0 \leq i \leq k$ and $\mathsf{Pr}(H)(i) = 0$ otherwise.*

It is a consequence of the stability of barcodes (Proposition 69) that the prominence diagram is stable with respect to the correspondence-interleaving distance:

Lemma 78 *Let H and E be finite $\mathbb{R}$-hierarchical clusterings. Then*

$$d_\infty(\mathsf{Pr}(H),\mathsf{Pr}(E)) \leq 2\, d_{\mathrm{CI}}(H,E).$$

Definition 79 *Let H be an essentially finite one-parameter hierarchical clustering of a set X and let $\bar{H} : \mathbb{R} \to \mathsf{C}(X)$ be its extension as in Definition 8. By Lemma 78 and Proposition 121, the prominence diagrams $\mathsf{Pr}(\bar{H}_{\geq\tau})$ converge uniformly as $\tau \to 0$ to a prominence diagram which we denote by $\mathsf{Pr}(H)$ and refer to as the **prominence diagram** of H.*

Notation 80 *Let H be an essentially finite one-parameter hierarchical clustering. The n^{th} prominence gap of H is $\mathsf{gap}_n(H) = \mathsf{gap}_n(\mathsf{Pr}(H))$ and the n^{th} gap size of H is $\mathsf{gapsize}_n(H) = \mathsf{gapsize}_n(\mathsf{Pr}(H))$.*

We note that Lemma 78 is true also for essentially finite hierarchical clusterings:

Lemma 81 *Let H and E be essentially finite $\mathbb{R}$-hierarchical clusterings. Then*

$$d_\infty(\mathsf{Pr}(H),\mathsf{Pr}(E)) \leq 2\, d_{\mathrm{CI}}(H,E).$$

6. Persistence-Based Flattening of One-Parameter Hierarchical Clusterings

For many applications, one needs a clustering of the input data (in the sense of Definition 1), not a hierarchical clustering. We say that a *flattening* algorithm takes a hierarchical clustering of a set X, and returns a clustering of X. Persistable clusters data by first constructing a hierarchical clustering of the data (using the λ-`link` algorithm from Section 2.4), and then applying the *persistence-based flattening algorithm*, which we introduce in this section.

The most obvious flattening algorithm takes a hierarchical clustering H, and returns $H(r)$ for some index r. However, it can happen that H encodes multi-scale clustering structure in the data that is not reflected in $H(r)$ for any single choice of r. We want a flattening algorithm that can extract clusters at multiple scales.

An example of such an algorithm is the ToMATo clustering algorithm (Chazal et al., 2013), which computes a flattening of the hierarchical clustering induced by a filtered graph.

A major advantage of ToMATo is its innovative parameter selection process: the user determines how fine the output clustering will be by choosing a merging parameter τ, and this choice is guided by the barcode of the hierarchical clustering (Fig. 5). On a technical level however, one disadvantage of this algorithm is that its output depends on a choice of ordering of the vertices in the input graph, and in some use cases there may not be a clear way to make this choice. The persistence-based flattening algorithm (PF) is an adaptation of the ToMATo algorithm that avoids the dependence on an ordering of the input.

As input, PF takes a one-parameter hierarchical clustering H. We prove a stability theorem for this algorithm that is stated in terms of interleavings; so, this result is compatible with our stability and consistency results for λ-$\texttt{link}$. Parameter selection is very similar to that of the ToMATo algorithm, however, for PF, the user determines how fine the output clustering will be by choosing the number of clusters, guided by the barcode of the input.

In many TDA applications, barcodes are used to distinguish significant features in data from noise. A cut-off is chosen between "long" and "short" bars; the long bars correspond to significant features, and the short bars to noise (Ghrist, 2008; Fasy et al., 2014). In order to choose the number of clusters for $\mathsf{PF}(H)$, the practitioner chooses how many bars in the barcode of H to regard as significant features. If n bars are chosen, the output of PF will consist of n clusters. We call the difference between the length of the n^{th} longest and $(n+1)^{\text{th}}$ longest bars the n^{th} *gap size* of H. This quantity plays the key role in our stability theorem for PF. The larger the gap size, the more stable the output will be. So, choosing the number of clusters boils down to looking at the barcode of H, and finding choices of n such that the n^{th} gap size is large.

In this section, we restrict attention to $\mathbb{R}$-hierarchical clusterings. One can apply the constructions and results of this section to any one-parameter hierarchical clustering H by first taking the $\bar{H}$ construction from Definition 8.

We now define PF. The basic idea is that one can extract a clustering from a one-parameter hierarchical clustering by taking the leaves (Definition 62, Fig. 6). However, noise in the underlying data can lead to spurious, short leaves. So, we first prune the hierarchical clustering H by taking the persistence-based pruning $H_{\geq\tau}$ (Definition 63, Fig. 7).

The construction uses the n^{th} prominence gap of H (Notation 80), the notion of persistent cluster (Definition 60), and the prominence diagram $\mathsf{Pr}(H)$ (Definition 79).

Definition 82 *Let H be an essentially finite $\mathbb{R}$-hierarchical clustering of a set X. Assume that the n^{th} prominence gap of H is non-empty. The* ***persistence-based flattening*** *of H with respect to the n^{th} prominence gap of H is the set of n pairwise-disjoint persistent clusters of X given by $\mathsf{PF}(H,n) = \mathsf{leaves}(H_{\geq\tau})$, where $\tau = (\mathsf{Pr}(H)(n-1) + \mathsf{Pr}(H)(n))/2$.*

The output of PF is a set of pairwise-disjoint persistent clusters. This is important for our stability theorem. However, if we want a clustering of X in the sense of Definition 1, we take the underlying set (Definition 60) of each persistent cluster in $\mathsf{PF}(H,n)$.

When the input of PF is a hierarchical clustering induced by a finite filtered graph (Definition 72), PF can be computed by adapting the ToMATo algorithm (Chazal et al., 2013). This is what we do for our implementation of Persistable (Scoccola and Rolle, 2023).

In Definition 82, we take τ to be the average of $\mathsf{Pr}(H)(n-1)$ and $\mathsf{Pr}(H)(n)$ for convenience. If one takes a different τ in the n^{th} prominence gap, one gets the same clustering of the underlying data by the following proposition, which is proved in Appendix A.5.

Proposition 83 *Let H be an essentially finite $\mathbb{R}$-hierarchical clustering, and say $n \geq 1$ and $\tau, \tau' \in \mathsf{gap}_n(H)$. There is a bijection $m : \mathsf{leaves}(H_{\geq\tau}) \to \mathsf{leaves}(H_{\geq\tau'})$ such that for all $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})$, the underlying sets of $\mathbf{C}$ and $m(\mathbf{C})$ are equal.*

There are many ways to measure the similarity between two clusterings of a data set (see, e.g., Meilă (2007) and references therein), so there are many ways one could try to formulate a stability result for a flattening procedure. Our approach is based on the fact that PF produces a set of persistent clusters. The following stability theorem guarantees that if H and E are hierarchical clusterings that are sufficiently close in the correspondence-interleaving distance, then the persistent clusters in $\mathsf{PF}(H, n)$ are interleaved with the persistent clusters in $\mathsf{PF}(E, n)$. Here, the n^{th} gap size of H determines what "sufficiently close" means. The proof of the theorem is in Appendix A.5.

Theorem 84 *Let H and E be essentially finite $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively. Let $n \geq 1$, and assume there is $\varepsilon < \mathsf{gapsize}_n(H)/16$ such that H and E are ε-interleaved with respect to a correspondence $R \subseteq X \times Y$. Then there is a bijection $m : \mathsf{PF}(H, n) \to \mathsf{PF}(E, n)$ such that for all $\mathbf{C} \in \mathsf{PF}(H, n)$, $\mathbf{C}$ and $m(\mathbf{C})$ are 3ε-interleaved with respect to R.*

The interleavings guaranteed by this theorem imply that, if $\mathbf{C} \in \mathsf{PF}(H, n)$, and $x \in U(\mathbf{C})$ appears early enough in the lifetime of $\mathbf{C}$, then every point in Y that corresponds to x under R must belong to $U(m(\mathbf{C}))$.

Because this stability theorem for persistence-based flattening is stated in terms of interleavings, it can be combined with the stability and consistency results proved earlier in this paper. As an example, we state the following stability results for λ-$\mathtt{link}$ (Example 34). The combination of λ-$\mathtt{link}$ and persistence-based flattening is the core algorithm of Persistable.

The first result concerns stability in the input data:

Corollary 85 *Let $\mathcal{M}$ be a compact metric probability space, let $\lambda = \lambda^{x,y}_{\text{cov}}$ with slope σ, and assume $\mathsf{gap}_n(\lambda\text{-}\mathtt{link}(\mathcal{M}))$ is non-empty. Let $\mathcal{N}$ be a compact metric probability space with*

$$d_{\text{GHP}}(\mathcal{M}, \mathcal{N}) < \frac{\mathsf{gapsize}_n(\lambda\text{-}\mathtt{link}(\mathcal{M}))}{16 \cdot \max(|1/\sigma|, 2)}\,.$$

There is a bijection $m : \mathsf{PF}(\lambda\text{-}\mathtt{link}(\mathcal{M}), n) \to \mathsf{PF}(\lambda\text{-}\mathtt{link}(\mathcal{N}), n)$ such that, for all $\mathbf{C} \in \mathsf{PF}(\lambda\text{-}\mathtt{link}(\mathcal{M}), n)$, $\mathbf{C}$ and $m(\mathbf{C})$ are 3ε-interleaved with respect to a correspondence between $\mathcal{M}$ and $\mathcal{N}$, for some $\varepsilon < \mathsf{gapsize}_n(\lambda\text{-}\mathtt{link}(\mathcal{M}))/16$.

Proof By Corollary 42(2), $d_{\text{CI}}(\lambda\text{-}\mathtt{link}(\mathcal{M}), \lambda\text{-}\mathtt{link}(\mathcal{N})) \leq d_{\text{GHP}}(\mathcal{M}, \mathcal{N}) \cdot \max(|1/\sigma|, 2)$. So, we can take ε with $d_{\text{CI}}(\lambda\text{-}\mathtt{link}(\mathcal{M}), \lambda\text{-}\mathtt{link}(\mathcal{N})) < \varepsilon < \mathsf{gapsize}_n(\lambda\text{-}\mathtt{link}(\mathcal{M}))/16$. Then the result follows from Theorem 84. ■

The second result concerns stability in the choice of λ:

Corollary 86 *Let $\mathcal{M}$ be a compact metric probability space, let $\lambda = \lambda^{x,y}_{\text{cov}}$ with slope σ, and let $\lambda' = \lambda^{x',y'}_{\text{cov}}$ with slope σ'. Say*

$$\varepsilon := \max\left(|x - x'|,\ |y - y'| \cdot \min(|1/\sigma|, |1/\sigma'|)\right) < \mathsf{gapsize}_n(\lambda\text{-}\mathtt{link}(\mathcal{M}))/16\,.$$

Then there is a bijection $m : \mathsf{PF}(\lambda\texttt{-link}(\mathcal{M}), n) \to \mathsf{PF}(\lambda'\texttt{-link}(\mathcal{M}), n)$ *such that, for all* $\mathbf{C} \in \mathsf{PF}(\lambda\texttt{-link}(\mathcal{M}), n)$, $\mathbf{C}$ *and* $m(\mathbf{C})$ *are* 3ε*-interleaved.*

Proof By Proposition 43, $\lambda\texttt{-link}(\mathcal{M})$ and $\lambda'\texttt{-link}(\mathcal{M})$ are ε-interleaved. So, the result follows from Theorem 84. ■

The ToMATo algorithm takes as input a finite graph and a real-valued function f on its vertices. This induces the upper-star filtration on the graph, where an edge $\{x, y\}$ appears at $\min\{f(x), f(y)\}$. We now describe the *exhaustive persistence-based flattening algorithm* (Algorithm 2), which is essentially a generalization of ToMATo to the more general filtered graphs of Definition 72. This is of interest because the $\lambda\texttt{-link}$ of a finite metric space is induced by a filtered graph, but not by an upper-star filtration. We describe the precise relationship between ExhaustivePF and ToMATo in Remark 127 in Appendix A.5.

We call the algorithm "exhaustive" because, unlike PF, ExhaustivePF clusters every point in its input. For PF, points that enter $H_{\geq\tau}$ outside of a leaf do not get clustered. ExhaustivePF uses the data of the input graph and an ordering of its simplices to assign such points to some leaf. For Persistable, we prefer PF to ExhaustivePF, because of the good stability properties of PF, and the fact that it does not depend on an ordering of the input. However, ExhaustivePF also produces interesting results (see the Olive oil data in Section 7.2 for an example).

Algorithm 2 Exhaustive persistence-based flattening of the HC induced by a finite filtered graph

procedure ExhaustivePF($G = [\sigma_1, \ldots, \sigma_p], f, \tau$)
  ▷ Assume the simplices of G are ordered such that $f(\sigma_i) \leq f(\sigma_{i+1})$ and $\sigma_i \subseteq \sigma_j \Rightarrow i \leq j$
  Let `clusters` $\leftarrow \{\}$
  for $1 \leq i \leq p$ **do**
    if $\sigma_i = \{x\}$ **then** ▷ A vertex appears and a cluster is born
      `clusters` $\leftarrow$ `clusters` $\cup \{(\{x\}, f(\sigma_i))\}$
    else if $\sigma_i = \{x, y\}$ **then** ▷ An edge appears
      Let $(c, u), (d, v) \in$ `clusters` be such that $x \in c$ and $y \in d$
      if $c \neq d$ **then**
        if $f(\sigma_i) - u \leq \tau$ or $f(\sigma_i) - v \leq \tau$ **then** ▷ At least one cluster did not persist enough
          `clusters` $\leftarrow$ (`clusters` $\setminus \{(c, u), (d, v)\}$) $\cup \{(c \cup d, \min(u, v))\}$ ▷ Merge clusters
        end if
      end if
    end if
  end for
  return `clusters`
end procedure

7. Persistable

Persistable is a pipeline for density-based clustering that integrates the algorithms defined in this paper. In another publication (Scoccola and Rolle, 2023), we described the implementation of Persistable. In this section, we describe the design choices of Persistable in detail, and explain how these choices are motivated by the theoretical results in this paper.

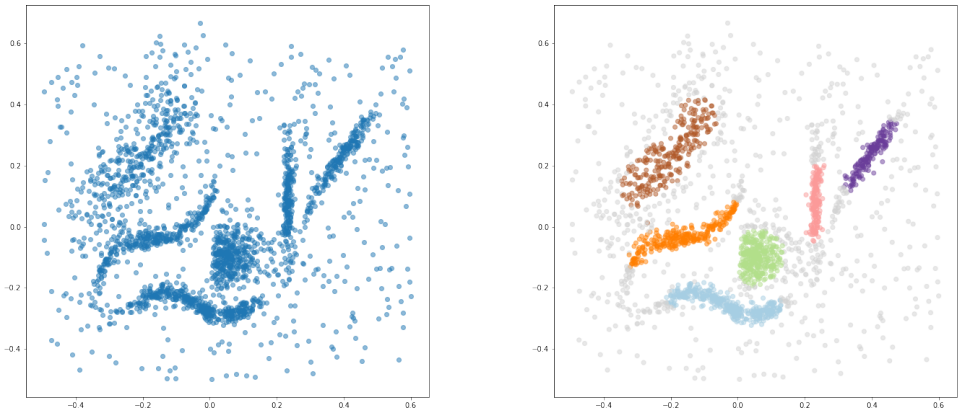

Figure 8: The running example. We cluster the data with Persistable: clusters are indicated by colors, and gray points are classified as noise.

Compared to existing clustering methods, a novel feature of Persistable is that the parameter selection process is guided by interactive visualization tools. This parameter selection process is based on the stability results proved in this paper; the visualization tools are inspired by tools from multiparameter persistent homology, in particular, the software library RIVET (2020).

We begin by demonstrating the Persistable pipeline on a simple running example, and then we evaluate its performance on real-world benchmark data sets. See the Persistable software repository (link available in Scoccola and Rolle 2023) for code that replicates all the examples in this section, as well as for further evaluations of Persistable on benchmark data sets.

7.1 The Persistable Pipeline

As input, Persistable takes a finite metric space M, and produces a clustering of M, in the sense of Definition 1. As a running example, we use a synthetic data set from the hdbscan clustering library (McInnes et al., 2017). This data set is designed to be challenging for clustering algorithms, while being easy to visualize: see Fig. 8.

Conceptually, Persistable begins with the degree-Rips hierarchical clustering $\mathrm{DR}(M)$ that was described in the introduction (see Fig. 1, and see Definition 24 for the formal definition). We can get insight into $\mathrm{DR}(M)$ by plotting the **component counting function**, which is the function defined on the first quadrant of the plane where at (s, k) we simply count the number of clusters in $\mathrm{DR}(M)(s, k)$. The first visualization in the Persistable pipeline is a heat map of the component counting function. See Fig. 9 for this visualization on the running example.

Now, Persistable constructs a clustering of the input M in two steps.

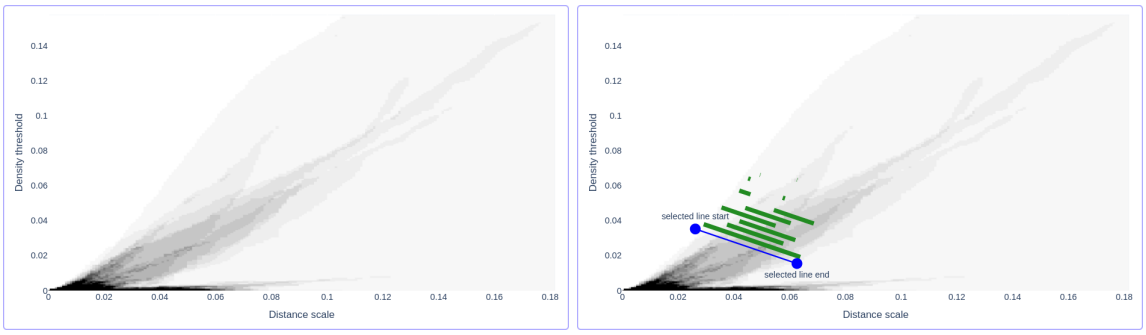

Figure 9: The first Persistable visualization is **the component counting function**; we show this for the running example. One can see typical behavior of degree-Rips (DR): when the distance scale s is small and the density threshold k is large, no points are clustered; when s is large and k is small, all points are clustered together. In between these two regimes is a band of interesting cluster structure. The blue line segment defines a slice of DR, and its barcode is plotted in green. The sixth gap size of this slice is quite large (the sixth-longest bar is much longer than the seventh-longest bar). So, we choose six clusters for the persistence-based flattening. The output is displayed in Fig. 8.

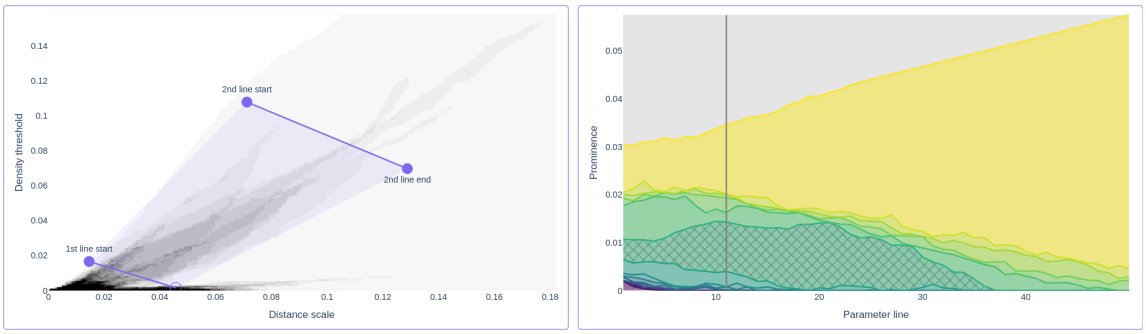

Figure 10: The second Persistable visualization is the **prominence vineyard**. The user chooses two slices of degree-Rips by choosing the start and end points. This determines a one-parameter family of slices that interpolate between the two chosen slices. For each slice in the family, Persistable computes the barcode, and the prominence vineyard (on the right) displays the prominence (i.e., length) of each bar in the barcode. So, the first (top) curve in the prominence vineyard shows the length of the longest bar in the barcode of each slice, the second curve shows the length of the second-longest bar, etc. We call these curves *vines*. The first gap (between the first and second vines, displayed in yellow) is very large: this just reflects the fact that the longest bar of an HC is typically much longer than any other bar. The sixth gap (marked in the figure) is also large. This means there are many slices in the vineyard such that the sixth gap size of the slice is large. We used this prominence vineyard to choose the slice in Fig. 9.

7.1.1 Step One: Reduce a Two-Parameter HC to a One-Parameter HC

The first step is to reduce from the two-parameter hierarchical clustering $\mathrm{DR}(M)$ to a one-parameter hierarchical clustering, by taking a slice (see Fig. 2). Using the notation established in Example 34, this means taking $\lambda\text{-}\texttt{link}(M)$, where λ is a choice of line in the (s,k)-parameter space. For example, see the blue line segment in Fig. 9. The practitioner can choose such a slice by clicking on the component counting visualization to choose the start and end points of a line segment. This determines an interval $(s_{\text{start}}, s_{\text{end}})$ on which the one-parameter hierarchical clustering $\lambda\text{-}\texttt{link}(M)$ is defined, and for $s \in (s_{\text{start}}, s_{\text{end}})$, $\lambda\text{-}\texttt{link}(M)(s) = \mathrm{DR}(M)(s, \sigma \cdot s + y)$, where y is the y-intercept of the selected line, and σ is its slope. The second visualization in the Persistable pipeline is an interactive tool for choosing slices. We introduce this tool after we discuss the second step of Persistable.

7.1.2 Step Two: Reduce a One-Parameter HC to a Clustering

The second step is to reduce from the one-parameter hierarchical clustering $\lambda\text{-}\texttt{link}(M)$ to a clustering of M, by applying the persistence-based flattening procedure, defined in Section 6. To apply the persistence-based flattening, one chooses the number of clusters in the output, guided by the barcode of the HC. The barcode is a visual summary of an HC (see Fig. 5). If one chooses n clusters, these will correspond to the n longest bars in the barcode. We call the difference between the length of the n^{th} longest and $(n+1)^{\text{th}}$ longest bars the n^{th} *gap size* of the HC. As explained in Section 6, the larger the n^{th} gap size, the more stable the output with n clusters will be. So, choosing the number of clusters boils down to looking at the barcode, and finding choices of n such that the n^{th} gap size is large.

In the case of the running example, there is a drop-off between the sixth and seventh longest bars, so choosing six bars (i.e., six clusters) is a reasonable choice (see Fig. 9). The resulting clustering of the data is displayed in Fig. 8.

7.1.3 Choosing the Slice

To complete the description of the Persistable pipeline, it remains to discuss how the practitioner chooses a slice. The answer is motivated by Step 2. When one applies the persistence-based flattening, one looks for large gap sizes in the barcode of $\lambda\text{-}\texttt{link}(M)$. So, the second visualization tool in the Persistable pipeline helps the user identify slices λ that lead to barcodes with large gap sizes.

The practitioner begins with the component counting visualization (Fig. 9). From this, one can find the region of the DR parameter space where interesting cluster structure is captured. The practitioner is asked to choose two slices in the DR parameter space, which determine a one-parameter family of slices that interpolate between the two chosen slices. As λ varies in the family, the slice $\lambda\text{-}\texttt{link}(M)$ changes in a continuous way by Proposition 43. Thus, the barcode of $\lambda\text{-}\texttt{link}(M)$ also changes in a continuous way. In the **prominence vineyard** visualization, Persistable plots the *prominence* (i.e., length) of each bar in the barcode of $\lambda\text{-}\texttt{link}(M)$. As λ varies, these prominences trace out curves, which we call *vines* (this is standard terminology in topological data analysis, beginning with Cohen-Steiner et al., 2006). See Fig. 3. In Fig. 10, we display a prominence vineyard for the running example. There is a large gap between the first and second vines; this is typical behavior, as the longest bar in the barcode of an HC is just the interval on which the HC is non-empty,

which is often much longer than any other bar. More interesting structure is captured by the other gaps in the prominence vineyard. For example, there is a large gap between the sixth and seventh vines, which is marked in Fig. 10. If we choose a slice that includes this gap, then the sixth gap size of the barcode of this slice is large. Indeed, this is how we picked the slice that we used in Step 2, above.

7.2 Examples of Persistable on Benchmark Data Sets

We now demonstrate how Persistable can identify meaningful cluster structure in data.

7.2.1 Rideshare Data

We consider a data set consisting of approximately 560 000 rideshare pickup locations in the New York City area from April, 2014. The data set is the result of a Freedom of Information request by the website FiveThirtyEight (2015). This data set has very complex cluster structure, at many different scales. There are informative clusterings at a very coarse level, and also at much finer levels.

To get a feel for the data, we begin by considering the subset of points in a square centered at LaGuardia Airport (see Fig. 11), which consists of approximately 10 000 points.

While this data set is easily visualizable, it is challenging for many density-based clustering algorithms, because its apparent cluster structure takes place at very different levels of density. For example, there are approximately 4000 data points clustered near Terminal B of the airport, and meanwhile, there is a cluster of approximately 60 data points near the LaGuardia Airport Marriott hotel, and an even smaller cluster of 12 data points near the P.S. 127 Aerospace Science Magnet School.

In particular, this data set is challenging for HDBSCAN (Campello et al., 2013), a popular density-based clustering algorithm that is based on the robust single-linkage algorithm of Example 32 (for a description in these terms, see McInnes and Healy, 2018). Unlike the related DBSCAN algorithm (Ester et al., 1996), HDBSCAN can detect multi-scale cluster structure that takes place at a range of distance scales. However, like robust single-linkage, it relies on a fixed density threshold, and for this reason, cannot find the multi-scale cluster structure in this example (i.e., it cannot simultaneously find the large cluster near Terminal B and the small cluster near the hotel). See the jupyter notebook for this data set at the Persistable repository (2023) to try clustering the data with HDBSCAN.

On the other hand, Persistable is sensitive to this kind of multi-scale clustering structure, because it uses slices of DR in which both the density threshold and the distance scale vary. In Fig. 12, we show Persistable visualizations for the data. Guided by these visualizations, one can use Persistable to find clusterings of the rideshare data that simultaneously capture the large clusters near the airport terminals and the small clusters in the surrounding neighborhood.

Now we consider the complete Rideshare data set. The complexity of the data is reflected in the Persistable visualizations; see Fig. 13. Using Persistable, one can obtain informative clusterings of this data at coarser or finer scales. For example, see Fig. 14 for coarse but informative results. Using smaller gaps in the prominence vineyard, one can obtain finer results, with, for example, clusters centered at Penn Station and the Meatpacking district

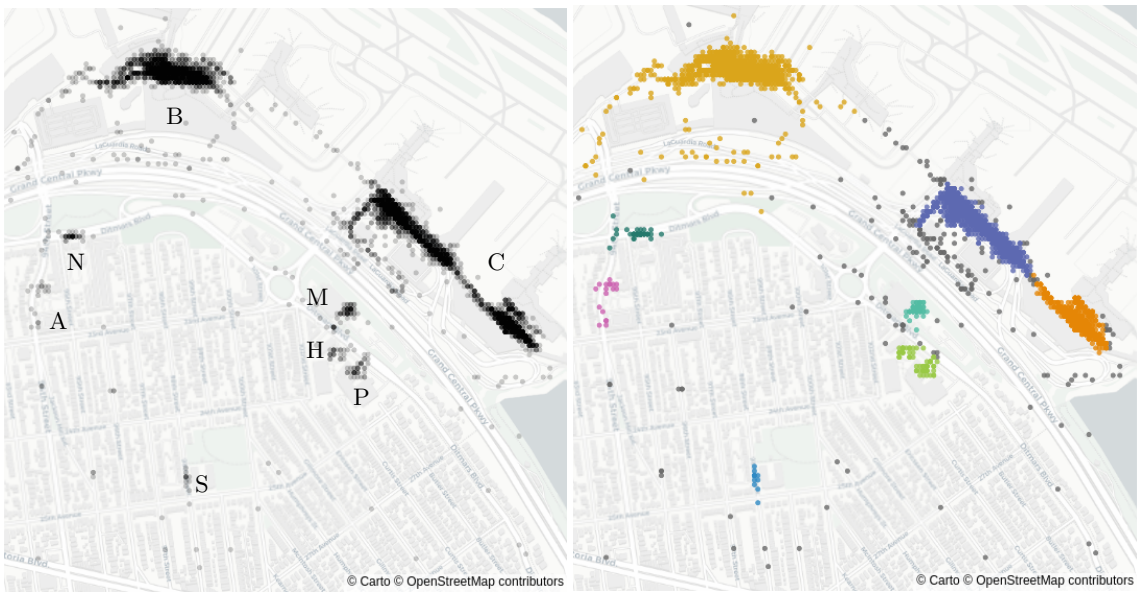

Figure 11: The subset of the Rideshare data in a square centered at LaGuardia Airport. On the left, some relevant landmarks are marked: ACE Rent A Car (A), National Car Rental (N), LaGuardia Airport Terminal B (B), Terminal C (C), Marriott Hotel (M), Hampton Inn Hotel (H), LaGuardia Plaza Hotel (P), P.S. 127 Aerospace Science Magnet School (S). On the right, a clustering produced by Persistable, using the slice displayed in Fig. 12. Gray points are unclustered.

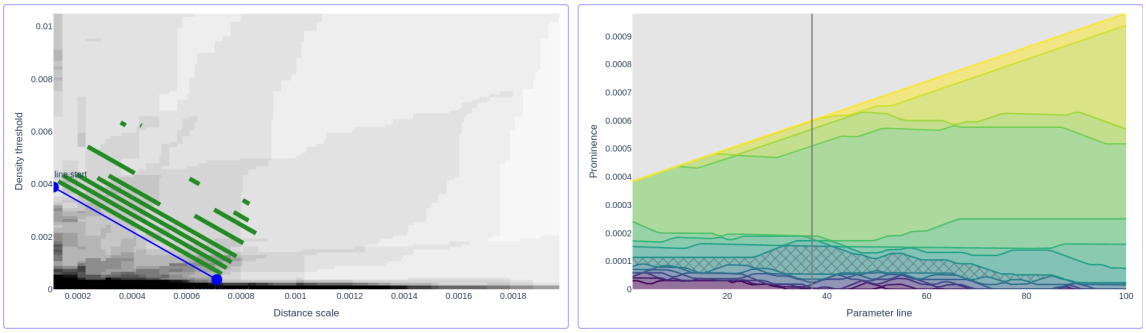

Figure 12: Persistable visualizations for the subset of the Rideshare data near LaGuardia Airport. On the left, the component counting visualization, and on the right, a prominence vineyard visualization. We use the prominence vineyard to choose a slice, which appears as the blue line segment on the component counting visualization; the corresponding slice in the vineyard is marked by a vertical line. The barcode of this slice is displayed in green. Several gaps in this vineyard lead to interesting clusterings. If we choose the gap below the eighth vine (marked in this figure), we get the clustering of the data displayed in Fig. 11. See Fig. 16 in Appendix A.6 for the result of choosing the gap below the fourth vine.

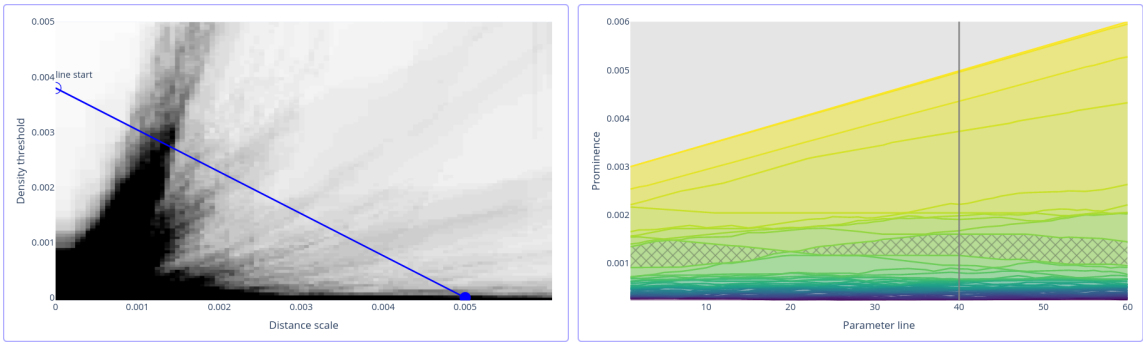

Figure 13: Persistable visualizations for the Rideshare data. On the left, the component counting visualization, and on the right, a prominence vineyard visualization. While the component counting visualization is very complicated, one can easily identify interesting gaps in the prominence vineyard. We use the prominence vineyard to choose a slice, marked on the component counting function by a blue line and on the vineyard by a vertical line. If we choose the marked gap, we get the clustering of the data displayed in Fig. 14.

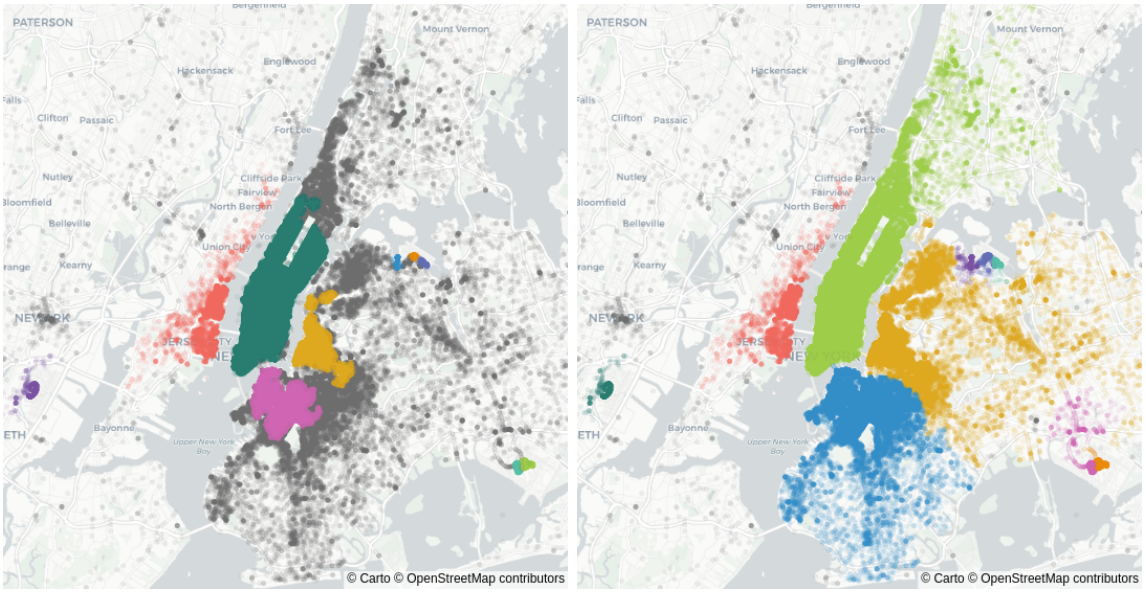

Figure 14: The Rideshare data, clustered using Persistable. Gray points are unclustered. In both cases, the clustering is obtained by choosing the slice indicated in Fig. 13. On the left, the clustering is obtained using the usual Persistable pipeline, choosing the gap marked in Fig. 13. On the right, the clustering is obtained using ExhaustivePF, rather than PF. We find large clusters in Manhattan, Brooklyn, Queens, and New Jersey, as well as smaller clusters at Newark, LaGuardia, and John F. Kennedy airports.

in Manhattan, and Williamsburg and Downtown Brooklyn. See the jupyter notebook for this data set at the Persistable repository (2023) for details.

We are able to easily compute clusterings of this data set with Persistable using the subsampling approximation described in Section 3.4 (which is justified by the Gromov–Hausdorff–Prokhorov stability of λ-linkage). Using a subsample of 30 000 data points, we are able to compute clusterings of the complete Rideshare data set in a matter of seconds, using approximately 200 MB of RAM, using a laptop with an Intel(R) Core(TM) i5 CPU (4 cores, 1.6GHz) and 8 GB RAM, running GNU/Linux.

Without the subsampling approximation, clustering this data set would be a significant computational challenge. For context, we clustered the data using the high-performance implementation of HDBSCAN of McInnes et al. (2017). This is a natural comparison, because HDBSCAN and Persistable are very similar on an algorithmic level, and because the implementation of McInnes et al. (2017) is very similar to our implementation of Persistable (indeed, important components of our implementation come directly from this implementation of HDBSCAN). The key performance advantage that Persistable has in this example is the subsampling approximation. An analogous approximation scheme is not valid for HDBSCAN, as the hierarchical clustering algorithm underlying HDBSCAN (robust single-linkage) is not Gromov–Hausdorff–Prokhorov stable (see Section 3.3).

See Table 2 in Appendix A.6 for the results. The memory usage of HDBSCAN scales with $n \cdot k$, where n is the number of data points and k is the density threshold parameter min_samples. This means that, on the laptop described above, we are only able to run HDBSCAN with very small values of the min_samples parameter, producing only very fine clusterings.

7.2.2 Olive Oil Data

We consider a data set concerning the fatty acid composition of 572 samples of olive oil. For each sample, the percentages of 8 fatty acids were measured. The samples were taken from nine regions of Italy, and these regions are grouped into three larger areas (South Italy, Sardinia, and North Italy). Each sample is labeled with its region of origin.

This is a useful test data set for classification and clustering methods, and it is used as a benchmark data set for density-based clustering by Stuetzle and Nugent (2010). The data set is due to Forina et al. (1983), and we obtained it from the supplementary materials of Stuetzle and Nugent (2010). Using Persistable, one can recover much of the hierarchical clustering structure defined by the regions of origin.

We consider the data as points in $\mathbb{R}^8$ with the Euclidean metric; each feature is independently centered to have mean zero and scaled to unit variance. One can see many large gaps in the prominence vineyard visualization (see Fig. 15). This is a consequence of the multi-scale clustering structure of the data. For example, say we choose the slice indicated in Fig. 15. If we choose the large gap between the third and fourth vines (i.e., we choose three clusters in the persistence-based flattening), we get a clustering of the data that fits the large area labels perfectly, with 89% of the data clustered (see Table 3 in Appendix A.6 for the confusion matrix). Meanwhile, if we choose the gap between the eighth and ninth vines marked in Fig. 15, we get a clustering that fits the region labels very accurately: the

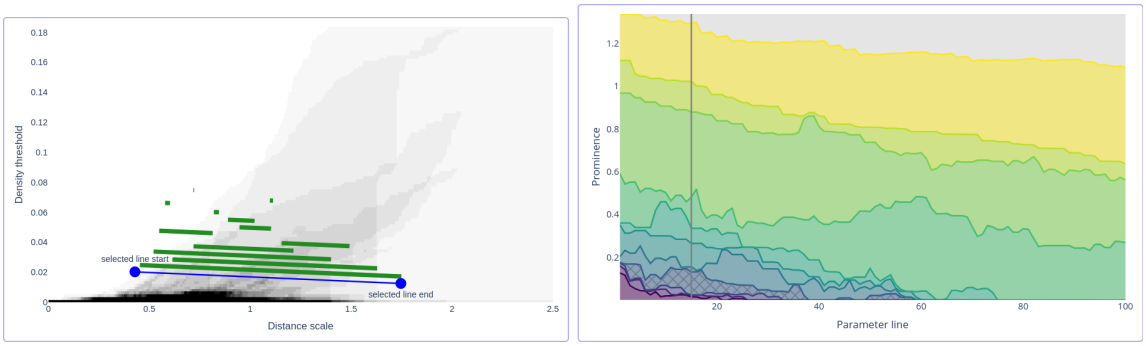

Figure 15: Persistable visualizations for the olive oil data. On the left, the component counting visualization, and on the right, a prominence vineyard visualization. We use the prominence vineyard to choose a slice, which appears as the blue line segment on the component counting visualization; the corresponding slice in the vineyard is marked by a vertical line. The barcode of this slice is displayed in green. Several gaps in this vineyard lead to interesting clusterings. If we choose the gap below the eighth vine (marked in this figure), we get a clustering that fits the region labels very accurately. If we choose the gap below the third vine, we get a clustering that fits the large area labels.

adjusted Rand index is 0.98, and 49% of the data is clustered (see Table 4 in Appendix A.6 for the confusion matrix).

In order to cluster more data points, we also run the exhaustive persistence-based flattening (Algorithm 2). We choose the same slice and gap as before, but replace the usual persistence-based flattening with the exhaustive persistence-based flattening. The result clusters 95% of the data, with an adjusted Rand index of 0.90 with respect to the region labels (see Table 5 in Appendix A.6 for the confusion matrix).

For comparison, Stuetzle and Nugent (2010) apply a density-based clustering algorithm to this data, and report an adjusted Rand index of 0.61 with respect to the region labels, with all data points clustered.

8. Conclusions

We conclude by mentioning some possible directions for future work. As we explained in Remark 41, Blumberg and Lesnick (2022) prove a stability result for the simplicial degree-Rips bifiltration using the Gromov–Prokhorov distance. The authors also provide experimental evaluation of their stability result, and suggest the goal of developing a stability theory for degree-Rips that better explains such experimental results (Blumberg and Lesnick, 2022, Remark A.3). It may be fruitful to pursue this goal in the setting of the degree-Rips hierarchical clustering.

In this paper we used slices of kernel linkage given by lines λ in the parameter space. Our stability theorem for kernel linkage implies that appropriately chosen non-linear slices are also stable, and our consistency theorem applies also to appropriate non-linear slices.

An interesting question is whether there are use cases in which non-linear slices lead to more informative clusterings.

For Persistable, we use a two-step process: we first reduce from degree-Rips to a one-parameter hierarchical clustering by taking a slice, then we reduce to a clustering using the persistence-based flattening. We begin by taking a slice of degree-Rips because one-parameter hierarchical clusterings are much simpler than multiparameter hierarchical clusterings. This distinction between one-parameter hierarchical clusterings and multiparameter hierarchical clusterings is analogous to the distinction between one-parameter persistence modules and multiparameter persistence modules; see Bauer et al. (2020) for a discussion of the structural complexity of multiparameter persistence modules with a focus on persistence modules arising from hierarchical clusterings. For our purposes, it is particularly important that the barcode and the persistence-based flattening algorithm are only defined for one-parameter hierarchical clusterings. Not much is known about flattening multiparameter hierarchical clusterings directly, but see Jardine (2020b) and Shiebler (2021). An interesting question for future research is how one can stably extract a single clustering from degree-Rips or kernel linkage, without taking a one-parameter slice.

Acknowledgments

We thank Michael Lesnick for telling us about the good properties of one-parameter slices in which all parameters vary, and for telling us about his work with Blumberg on the stability of degree-Rips, which inspired our work here. We thank Leland McInnes for helpful discussion about HDBSCAN. And, we thank Dan Christensen, Rick Jardine, Michael Kerber, and Matt Piekenbrock for helpful discussions about this project and related topics.

A.R. was partially supported by Austrian Science Fund (FWF) grant number P 29984-N35, and by the German Research Foundation (DFG) Project-ID 195170736 "TRR109 Discretization in Geometry and Dynamics". L.S. was partially supported by the National Science Foundation through grant CCF-2006661 and CAREER award DMS-1943758, as well as by EPSRC grant "New Approaches to Data Science: Application Driven Topological Data Analysis", EP/R018472/1. For the purpose of Open Access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

Appendix A. Missing Details

A.1 Details from Section 2

Proof (of Proposition 13) First, say H and E are ε-interleaved for $\varepsilon \geq 0$. Let $x, y \in X$; we show $|\theta_H(x,y) - \theta_E(x,y)| \leq \varepsilon$. Without loss of generality, $\theta_H(x,y) \leq \theta_E(x,y)$. Let $r > \theta_H(x,y)$. Then there is $C \in \bar{H}(r)$ with $x, y \in C$. By the ε-interleaving property, there is $D \in \bar{E}(r+\varepsilon)$ with $C \subseteq D$, and thus $\theta_E(x,y) \leq r+\varepsilon$. This shows that $|\theta_H(x,y) - \theta_E(x,y)| \leq \varepsilon$, and it follows that $d_{\mathrm{I}}(H,E) \geq d_\infty(\theta_H, \theta_E)$.

Now, say $\varepsilon > d_\infty(\theta_H, \theta_E)$. We show H and E are ε-interleaved. Let $r \in \mathbb{R}$, and let $C \in \bar{H}(r)$. Choose $x \in C$. As $\theta_H(x,x) \leq r$, we have $\theta_E(x,x) < r + \varepsilon$, so that there is $D \in \bar{E}(r+\varepsilon)$ with $x \in D$. Similarly, for any $y \in C$, $\theta_H(x,y) \leq r$, so that $\theta_E(x,y) < r + \varepsilon$,

and thus $y \in D$, and therefore $C \subseteq D$. This shows that $\bar{H}(r) \preceq \bar{E}(r+\varepsilon)$. A symmetric argument shows that $\bar{E}(r) \preceq \bar{H}(r+\varepsilon)$, and thus H and E are ε-interleaved. It follows that $d_{\mathrm{I}}(H,E) \leq d_\infty(\theta_H, \theta_E)$, and the proposition follows. ■

Proof (of Proposition 14) Let $\varepsilon \geq 0$. We show that $||f-g||_\infty \leq \varepsilon$ if and only if $H(f)$ and $H(g)$ are ε-interleaved, and the proposition follows. If $||f-g||_\infty \leq \varepsilon$, then, for every $r \geq \varepsilon$, we have $\{f \geq r\} \subseteq \{g \geq r-\varepsilon\}$, and $\{g \geq r\} \subseteq \{f \geq r-\varepsilon\}$. This implies that, after taking connected components, every connected component of $\{f \geq r\}$ is included in a connected component of $\{g \geq r-\varepsilon\}$, and that every connected component of $\{g \geq r\}$ is included in a connected component of $\{f \geq r-\varepsilon\}$. If $H(f)$ and $H(g)$ are ε-interleaved, then, for every x in the support of the functions, if $f(x) \geq \varepsilon$, then there exists a cluster in $\{g \geq f(x)-\varepsilon\}$ containing x, since $x \in \{f \geq f(x)\}$. This implies that for any x in the support of the functions we have $g(x) \geq f(x) - \varepsilon$. A symmetric argument shows that $f(x) \geq g(x) - \varepsilon$ for every x in the support, concluding the proof. ■

Proof (of Proposition 18) The only non-trivial case is the triangle inequality, which is proved by composing correspondences. If X, Y, Z are sets, and $R \subseteq X \times Y$ and $S \subseteq Y \times Z$ are correspondences, $S \circ R \subseteq X \times Z$ is the correspondence $S \circ R = \{(x,z) : \exists y \in Y \text{ with } (x,y) \in R \text{ and } (y,z) \in S\}$. If H and E are $\vec{\varepsilon}$-interleaved with respect to R, and E and F are $\vec{\delta}$-interleaved with respect to S, then H and F are $(\vec{\varepsilon}+\vec{\delta})$-interleaved with respect to $S \circ R$. From this it follows that $d_{\mathrm{CI}}(H,F) \leq d_{\mathrm{CI}}(H,E) + d_{\mathrm{CI}}(E,F)$. ■

Lemma 87 *Let H and E be ultrametric hierarchical clusterings of sets X and Y respectively. If R is a correspondence between X and Y, then the distortion of R is* $\mathrm{dis}(R) = \inf\{\varepsilon \geq 0 : H, E \text{ are } \varepsilon\text{-interleaved w.r.t. } R\}$.

Proof First we show that if H and E are ε-interleaved with respect to R, then $\mathrm{dis}(R) \leq \varepsilon$. Let $(x,y),(x',y') \in R$. If $r > \theta_H(x,x')$, then there is $C \in \bar{H}(r)$ containing x, x', and thus $\pi_X^{-1}(C)$ contains $(x,y),(x',y')$. By the interleaving property, there is a cluster in $\pi_Y^*(\bar{E})(r+\varepsilon)$ containing $(x,y),(x',y')$, and thus there is a cluster in $\bar{E}(r+\varepsilon)$ containing y, y'. So, $\theta_E(y,y') \leq r+\varepsilon$, and thus $\theta_E(y,y') \leq \theta_H(x,x') + \varepsilon$. Together with a symmetric argument, we have $|\theta_H(x,x') - \theta_E(y,y')| \leq \varepsilon$. Thus $\mathrm{dis}(R) \leq \varepsilon$. Now, we show that H and E are ε-interleaved with respect to R, for any $\varepsilon > \mathrm{dis}(R)$, which finishes the proof. Let $r \in \mathbb{R}$ and let $C \in \bar{H}(r)$. We need to show there is $D \in \bar{E}(r+\varepsilon)$ such that $\pi_X^{-1}(C) \subseteq \pi_Y^{-1}(D)$. Let $x, x' \in C$ and let $(x,y),(x',y') \in R$. We have $r \geq \theta_H(x,x')$, therefore $\theta_E(y,y') \leq r + \mathrm{dis}(R)$, and thus there is $D \in \bar{E}(r+\varepsilon)$ with $y, y' \in D$. It follows that $(x,y),(x',y') \in \pi_Y^{-1}(D)$, and as x, x' were arbitrary, we have $\pi_X^{-1}(C) \subseteq \pi_Y^{-1}(D)$. ■

Lemma 88 *Let K be a kernel, and let $\mathcal{M}$ be a metric probability space. Let $K^{-1} : \mathbb{R}_{>0} \to \mathbb{R}_{\geq 0}$ be defined as $K^{-1}(t) = \min\{u : K(u) \leq t\}$. Then K^{-1} is a non-increasing function with compact support, and we have, for every $x \in \mathcal{M}$,*

$$(\mu_{\mathcal{M}} * K_s)(x) = \int_0^\infty \mu_{\mathcal{M}}\left(B(x, sK^{-1}(r))\right)\, \mathsf{d}r.$$

Proof Since $K(r) \to 0$ as $r \to \infty$, for every $t > 0$ the set $\{u : K(u) \leq t\}$ is non-empty. Moreover, K is continuous from the right, so the set has a minimum, and thus K^{-1} is well-defined. The fact that K^{-1} is non-increasing is clear, and the fact that it has compact support follows from the fact that K is bounded.

To prove the statement about $(\mu_{\mathcal{M}} * K_s)$, we need the following straightforward fact about K^{-1}: for every $s, t \in \mathbb{R}_{\geq 0}$ we have $K^{-1}(t) > s$ if and only if $t < K(s)$. We finish the proof by computing

$$\begin{aligned}
\int_{x' \in \mathcal{M}} K\left(\frac{d(x,x')}{s}\right) \mathsf{d}\mu_{\mathcal{M}} &= \int_{x' \in \mathcal{M}} \int_0^\infty \mathbf{1}_{\left\{r < K\left(\frac{d(x,x')}{s}\right)\right\}} \mathsf{d}r \; \mathsf{d}\mu_{\mathcal{M}} \\
= \int_{x' \in \mathcal{M}} \int_0^\infty \mathbf{1}_{\{d(x,x') < sK^{-1}(r)\}} \mathsf{d}r \; \mathsf{d}\mu_{\mathcal{M}} &= \int_0^\infty \int_{x' \in \mathcal{M}} \mathbf{1}_{\{d(x,x') < sK^{-1}(r)\}} \mathsf{d}\mu_{\mathcal{M}} \; \mathsf{d}r \\
&= \int_0^\infty \mu_{\mathcal{M}}\left(B(x, sK^{-1}(r)\right) \mathsf{d}r,
\end{aligned}$$

as required. ■

A.2 Details from Section 3

Lemma 89 *Let K be a kernel and let $r' \in (0, K(0))$. Let Z be a compact metric space and let μ and ν be Borel probability measures on Z such that $d_{\mathrm{P}}(\mu, \nu) < \varepsilon$ for $\varepsilon > 0$. Let $x, y \in Z$ such that $d_Z(x, y) < \varepsilon'$. Then, for all $s > 0$, we have $(\mu * K_s)(x) \leq (\nu * K_{s+\varepsilon_s})(y) + \varepsilon_k$, for $\varepsilon_s = \frac{\varepsilon+\varepsilon'}{K^{-1}(r')}$ and $\varepsilon_k = K(0)\left(\frac{K(0)}{r'} - 1\right) + K(0)\varepsilon$.*

Proof Using Lemma 88, we know that $(\mu * K_s)(x) = \int_0^{K(0)} \mu(B(x, sK^{-1}(r)))\, \mathsf{d}r$, since, if $r > K(0)$, then $K^{-1}(r) = 0$. Note that, for any radius $R \geq 0$, we have

$$\mu\left(B(x,R)\right) \leq \nu\left(B(x,R)^\varepsilon\right) + \varepsilon \leq \nu\left(B(x, R+\varepsilon)\right) + \varepsilon \leq \nu\left(B(y, R+\varepsilon+\varepsilon')\right) + \varepsilon,$$

so we can bound the local density estimate of x as follows:

$$(\mu * K_s)(x) \leq \int_0^{K(0)} \nu(B(y, sK^{-1}(r)+\varepsilon+\varepsilon')) + \varepsilon \; \mathsf{d}r = \int_0^{K(0)} \nu(B(y, sK^{-1}(r)+\varepsilon+\varepsilon'))\; \mathsf{d}r + K(0)\varepsilon.$$

Since K^{-1} is non-increasing, and $r' < K(0)$, it follows that $K^{-1}(rr'/K(0)) \geq K^{-1}(r)$ for every $r \geq 0$. Moreover, for any $0 \leq r \leq K(0)$, we have $K^{-1}(rr'/K(0)) \geq K^{-1}(r')$. These two considerations imply that, for $0 \leq r \leq K(0)$, we have

$$sK^{-1}(r) + \varepsilon + \varepsilon' \leq \left(s + (\varepsilon+\varepsilon')/K^{-1}(r')\right) K^{-1}\left(rr'/K(0)\right).$$

Combining this with the above bound for the local density estimate of x we get

$$\begin{aligned}
(\mu * K_s)(x) &\leq \int_0^{K(0)} \nu\left(B\left(y, \left(s + (\varepsilon+\varepsilon')/K^{-1}(r')\right) K^{-1}\left(rr'/K(0)\right)\right)\right)\; \mathsf{d}r + K(0)\varepsilon \\
&= \frac{K(0)}{r'} \int_0^{r'} \nu\left(B\left(y, \left(s + (\varepsilon+\varepsilon')/K^{-1}(r')\right) K^{-1}(r)\right)\right)\; \mathsf{d}r + K(0)\varepsilon \\
&\leq \frac{K(0)}{r'}\; \left(\nu * K_{(s+(\varepsilon+\varepsilon')/K^{-1}(r'))}\right)(y) + K(0)\varepsilon.
\end{aligned}$$

Finally, note that, for $0 \leq a \leq M < \infty$ and $c \geq 1$, we have $ca \leq a + M(c-1)$. As ν is a probability measure, any local density estimate is bounded by $K(0)$. This implies that

$$(\mu * K_s)(x) \leq \left(\nu * K_{(s+(\varepsilon+\varepsilon')/K^{-1}(r'))}\right)(y) + K(0)\left(\frac{K(0)}{r'} - 1\right) + K(0)\varepsilon,$$

as required. ■

Proof (of Proposition 44) We will construct a finite metric space M such that $\mathrm{RSL}_{\kappa,\alpha}$ is not continuous at M. For any $\delta > 0$, we will show that there is a finite metric space N with $d_{\mathrm{GHP}}(M, N) < \delta$ such that $d_{\mathrm{CI}}(\mathrm{RSL}_{\kappa,\alpha}(M), \mathrm{RSL}_{\kappa,\alpha}(N)) > 1/2$.

Let $M \subset \mathbb{R}$ be a subset with $0, 1 \in M$, and with $\kappa - 2$ points in the interval $(-\frac{1}{10}, 0)$ and $\kappa - 2$ points in the interval $(1, \frac{11}{10})$. For $\ell \geq 1$, let $M_\ell = M \cup \{x + \frac{1}{\ell} : x \in M\}$. We have $d_{\mathrm{GHP}}(M, M_\ell) \leq \frac{1}{\ell}$ for all $\ell \geq 1$. It remains to show that $d_{\mathrm{CI}}(\mathrm{RSL}_{\kappa,\alpha}(M), \mathrm{RSL}_{\kappa,\alpha}(M_\ell)) > 1/2$ for all sufficiently large ℓ. Note that $\mathrm{RSL}_{\kappa,\alpha}(M)(r) = \emptyset$ for $r < 1$; however, for $\ell \geq 10$, 0 is in a cluster of $\mathrm{RSL}_{\kappa,\alpha}(M_\ell)(r)$ for any $r > 1/10$. This finishes the proof. ■

As we described in Section 3, one could also formalize robust single-linkage by taking the density threshold parameter to be a ratio $k \in (0, 1)$, and then letting $\mathrm{RSL}_{k,\alpha}(M) = \mathrm{L}(M)^\gamma$ for the covariant curve $\gamma\colon (0, \infty) \to \mathbb{R}^{\times 3}_{>0}$ with $\gamma(r) = (r, \alpha r, k)$. This variant also fails to be continuous with respect to the Gromov–Hausdorff–Prokhorov distance:

Proposition 90 *Let $k \in (0, 1)$ be rational, and let $\alpha > 0$. With respect to the Gromov–Hausdorff–Prokhorov distance and the correspondence-interleaving distance, $\mathrm{RSL}_{k,\alpha}$ is discontinuous.*

Proof Write $k = p/q$. Without loss of generality, we may assume $p \geq 2$. We will construct a finite metric space M such that $\mathrm{RSL}_{k,\alpha}$ is not continuous at M. Let $M \subset \mathbb{R}$ be a subset with $0 \in M$, with $|M \cap (-\frac{1}{10}, 0)| = p - 1$ and with $1, 2, \ldots, q - p \in M$. For $n \geq 1$, let

$$M_n = \left(M \cup M + \frac{1}{n^2} \cup \cdots \cup M + \frac{n-1}{n^2}\right) \setminus \left\{\frac{n-1}{n^2}\right\},$$

where $M + a = \{x + a : x \in M\}$. The idea is that we replace each point of M with n points that are tightly grouped together, except 0, which we replace with only $n - 1$ points (hence we remove the point $\frac{n-1}{n^2}$). We have $d_{\mathrm{GHP}}(M, M_n) \to 0$ as $n \to \infty$. We have 0 in a cluster of $\mathrm{RSL}_{k,\alpha}(M)(r)$ for any $r > 1/10$; however, for n sufficiently large, $\mathrm{RSL}_{k,\alpha}(M_n)(r) = \emptyset$ for any $r \leq 1/2$. This shows that $d_{\mathrm{CI}}(\mathrm{RSL}_{k,\alpha}(M), \mathrm{RSL}_{k,\alpha}(M_n)) > 4/10$ for sufficiently large n, finishing the proof. ■

Proof (of Proposition 45) For simplicity, we assume $\kappa' = \kappa + 1$, but the construction can easily be extended to the general case. Let $M \subset \mathbb{R}$ consist of κ points in the interval $(-1, 0)$, as well as the point $D + 2$. Let $x \in M \cap (-1, 0)$. Then x is in a cluster of $\mathrm{RSL}_{\kappa,\alpha}(M)(r)$ for any $r > 1$, but $\mathrm{RSL}_{\kappa',\alpha}(M)(r) = \emptyset$ for all $r \leq D + 2$. ■

We also have the analogue of Proposition 45 for the variant of robust single-linkage that takes a density threshold $k \in (0, 1)$ instead of κ:

Proposition 91 *Let $k, k' \in (0,1)$ be rational with $k \neq k'$, and let $\alpha > 0$. For any $D > 0$, there is a finite metric space M such that $d_{\mathrm{CI}}(\mathrm{RSL}_{k,\alpha}(M), \mathrm{RSL}_{k',\alpha}(M)) > D$.*

Proof The construction of M is similar to the proof of Proposition 90. Write $k = p/q$. Without loss of generality, we may assume $p \geq 2$, and $k < k'$. Let $M \subset \mathbb{R}$ be a subset with $0 \in M$, with $|M \cap (-\frac{1}{2}, 0)| = p - 1$, and with $D+1, 2(D+1), \ldots, q - p(D+1) \in M$. If $s \geq 1/2$, then 0 is in a cluster of $\mathrm{RSL}_{k,\alpha}(M)$. However, if $s < D+1$, then $\mathrm{RSL}_{k',\alpha}(M) = \emptyset$, and the proposition follows. ■

Proof (of Proposition 46) Let $M = \{0, t\} \subset \mathbb{R}$. Say PI is defined using the kernel K. Because we always use isotropic kernels, we have $(\mu_M * K_s)(0) = (\mu_M * K_s)(t) =: M$. So, $\mathrm{PI}_{s,t}(M)(r) = \emptyset$ if $r > M$ and $\mathrm{PI}_{s,t}(M)(r) = \{M\}$ if $r \leq M$. Now, for any $\varepsilon > 0$, let $M_\varepsilon = \{0, t + \varepsilon\}$. For any correspondence R between M and M_ε, we have $\pi_M^*(\mathrm{PI}_{s,t}(M))(r) = \{R\}$ for $r \leq M$, but $\pi_{M_\varepsilon}^*(\mathrm{PI}_{s,t}(M_\varepsilon))(r) \neq \{R\}$ for any $r > 0$. So, for any $0 \leq \delta < M$, $\mathrm{PI}_{s,t}(M)$ and $\mathrm{PI}_{s,t}(M_\varepsilon)$ are not δ-interleaved with respect to R, and thus $d_{\mathrm{CI}}(\mathrm{PI}_{s,t}(M), \mathrm{PI}_{s,t}(M_\varepsilon)) > \delta$. But, as $\varepsilon \to 0$, we have $d_{\mathrm{GHP}}(M, M_\varepsilon) \to 0$. ■

Proof (of Proposition 47) Say PI is defined using the kernel K. Let $s > 0$ be arbitrary, let $t = \min_{x \neq x'} d_M(x, x')$, and let $x_0, x_1 \in M$ be such that $d_M(x_0, x_1) = t$. If $t' < t$, then $\mathrm{PI}_{s,t'}(M)(r)$ consists of singletons for all $r > 0$, while $\mathrm{PI}_{s,t}(M)(r)$ has a cluster containing x_0 and x_1 for all $r \leq \min((\mu_M * K_s)(x_0), (\mu_M * K_s)(x_1))$. So, $d_{\mathrm{CI}}(\mathrm{PI}_{s,t}(M), \mathrm{PI}_{s,t'}(M))$ does not go to zero as $t' \to t$ from below. ■

A.3 Details from Section 4

In order to prove that CI-consistency implies Hartigan consistency, we need a lemma, which is similar to Chaudhuri and Dasgupta (2010, Lemma 14, Appendix: Consistency), except that we do not require super-level sets to have finitely many connected components.

Lemma 92 *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous, compactly supported probability density function, let $r > 0$ and $A \neq A' \in H(f)(r)$. There exists $\varepsilon > 0$ and $B, B' \in H(f)(r - \varepsilon)$ with $B \neq B'$ such that $A \subseteq B$ and $A' \subseteq B'$.*

Proof Say, towards a contradiction, that for all n with $1/n < r$, there is $B_n \in H(f)(r - 1/n)$ with $A \subset B_n$ and $A' \subset B_n$. It is a standard fact that if $K_1 \supseteq K_2 \supseteq \cdots$ is a nested sequence of non-empty, compact, connected sets in Euclidean space, then the intersection $\cap_{i=1}^{\infty} K_i$ is connected (Császár, 1978, 10.1.23). So, the intersection $B = \cap B_n$ is connected. For all $b \in B$, we have $f(b) \geq r - 1/n$ for all n large enough, so we must have $f(b) \geq r$. As B is contained in $\{f \geq r\}$ and B is connected, B must intersect only one connected component of $\{f \geq r\}$, but we have $A \subset B$ and $A' \subset B$, a contradiction. ■

Proof (of Proposition 56) Given $r > 0$ and distinct elements A and A' of $H(f)(r)$, we show that the probability of $A_n \cap A'_n = \emptyset$ goes to 1 as $n \to \infty$, where A_n is the smallest

cluster in $\mathbb{A}^{\theta_n}(X_n)$ that contains $A \cap X_n$ and likewise for A', and the θ_n are the parameters whose existence is given by CI-consistency of $\mathbb{A}$.

From Lemma 92 it follows that there exists $\varepsilon > 0$ and distinct elements $B, B' \in H(f)(r-\varepsilon)$ such that $A \subseteq B$ and $A' \subseteq B'$. Let $\delta \in (0,1)$. By assumption, there exists N such that, if $n \geq N$, then the probability that $\mathbb{A}^{\theta_n}(X_n)$ and $H(f)$ are $\varepsilon/2$-interleaved with respect to the closest point correspondence $R_c \subseteq X_n \times \mathcal{S}(f)$ is greater than $1-\delta$. As R_c contains the pairs (x,x) for $x \in X_n$, if $\mathbb{A}^{\theta_n}(X_n)$ and $H(f)$ are $\varepsilon/2$-interleaved with respect to R_c, then $\mathbb{A}^{\theta_n}(X_n)$ and $i^*(H(f))$ are $\varepsilon/2$-interleaved as hierarchical clusterings of X_n, where $i : X_n \to \mathcal{S}(f)$ is the inclusion. It is therefore enough to show that if $\mathbb{A}^{\theta_n}(X_n)$ and $i^*(H(f))$ are $\varepsilon/2$-interleaved, then $A_n \cap A'_n = \emptyset$. Now, if $\mathbb{A}^{\theta_n}(X_n)$ and $i^*(H(f)))$ are $\varepsilon/2$-interleaved, then there exist $C, C' \in \mathbb{A}^{\theta_n}(X_n)(r-\varepsilon/2)$ such that $A \cap X_n \subseteq C \subseteq B$ and $A' \cap X_n \subseteq C' \subseteq B'$. As $A_n \subseteq C$ and $A'_n \subseteq C'$, and $B \cap B' = \emptyset$, we have $A_n \cap A'_n = \emptyset$. ■

We now prove Theorem 58, the CI-consistency of $\overline{\lambda}$-`link`. Because the argument works in greater generality, we actually prove this for any "admissible family" of curves (this is Theorem 108). The curves $\{\lambda_{\mathrm{con}}^{x,y}\}_{x,y>0}$ from Theorem 58 will be an example of an admissible family. We will use the following curves in order to define slices of kernel linkage.

Definition 93 *A **slicing curve** consists of an interval $I_\gamma = (0, \max_\gamma)$ with $\max_\gamma \in (0,\infty]$, and an order-preserving function $\gamma = (\gamma_s, \gamma_t, \gamma_k) : I_\gamma^{\mathrm{op}} \to \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}^{\mathrm{op}}$, which we assume is continuous when viewed as a map between subspaces of Euclidean space. For any slicing curve γ and any metric probability space $\mathcal{M}$, we write γ-`link`$(\mathcal{M}) = \mathrm{L}(\mathcal{M})^\gamma$.*

As in Definition 57, we have to re-parameterize slices of kernel linkage in order to interleave them with the density-contour hierarchical clustering. We do this as follows.

Definition 94 *Let K be a kernel, and let γ be a slicing curve. For $s > 0$, we write $v_s = \int_{\mathbb{R}^d} K(||x||/s)\,\mathtt{d}x$. Define an order-preserving function $\varphi : I_\gamma \to \mathbb{R}_{>0}$ by $\varphi(r) = \gamma_k(r)/v_{\gamma_s(r)}$. We say that γ is **covering** if γ_s and γ_k are injective, $\gamma_s(r) \to 0$ as $r \to \max_\gamma$, and $\gamma_k(r) \to 0$ as $r \to 0$. If γ is covering, then φ is a bijection. In that case, we write $\overline{\gamma} = \gamma \circ \varphi^{-1} : \mathbb{R}_{>0}^{\mathrm{op}} \to \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}^{\mathrm{op}}$. If γ is a slicing curve that is covering, then $\overline{\gamma}$ is also a slicing curve.*

From now on, fix a continuous, compactly-supported density function $f : \mathbb{R}^d \to \mathbb{R}$ with support $\mathcal{S}(f)$.

Notation 95 *For $s > 0$, define $f_s : \mathcal{S}(f) \to \mathbb{R}$ by*

$$f_s(x) = \int_{\mathbb{R}^d} K\left(\frac{||x-y||}{s}\right) f(y)\ \mathtt{d}y.$$

Note that $f_s(x) = (f * K_s)(x) = (\mu_f * K_s)(x)$. As $\mathcal{S}(f)$ is compact, the continuous function f is uniformly continuous. An elementary consequence (see e.g. Folland, 2013, Theorem 8.14) is that f_s/v_s approximates f for small enough s:

Lemma 96 *Given $\varepsilon > 0$ there exists $\delta > 0$ such that if $s < \delta$ then $||f_s/v_s - f||_\infty < \varepsilon$.*

Notation 97 *Let $I \subseteq \mathbb{R}^{\mathrm{op}}_{>0}$ be an interval, and let $H : I \to \mathsf{C}(X)$ be a contravariant hierarchical clustering of a set X. We write $\eta_H : X \to [0, \infty]$ for the function defined by $\eta_H(x) = \sup\{r \in I \ : \ \exists C \in H(r), x \in C\}$.*

Notation 98 *We write $\mathrm{L}(f)$ for the kernel linkage of the metric probability space $(\mathcal{S}(f), \mu_f)$. For any slicing curve γ, we write $h^\gamma = \eta_{\mathrm{L}(f)^\gamma}$. If γ is covering, we write $h^{\overline{\gamma}} = \eta_{\mathrm{L}(f)^{\overline{\gamma}}}$.*

Lemma 99 *Let γ be a slicing curve that is covering. Then, for any $x \in \mathcal{S}(f)$, the quantity $h^\gamma(x)$ satisfies $f_{\gamma_s(h^\gamma(x))}(x) = \gamma_k(h^\gamma(x))$. And, there exists $r_1 \in I_\gamma$ such that $h^\gamma(x) < r_1$ for every $x \in \mathcal{S}(f)$.*

Proof Note first that, for any $x \in \mathcal{S}(f)$, $h^\gamma(x) = \sup\{r \in I_\gamma : f_{\gamma_s(r)}(x) \geq \gamma_k(r)\}$. We begin the proof by showing the following: there is $r_1 \in I_\gamma$ such that $\{x \in \mathcal{S}(f) : f_{\gamma_s(r_1)}(x) \geq \gamma_k(r_1)\} = \emptyset$, and there is $r_0 \in I_\gamma$ such that $\{x \in \mathcal{S}(f) : f_{\gamma_s(r_0)}(x) \geq \gamma_k(r_0)\} = \mathcal{S}(f)$. For the existence of r_1, note that $f_{\gamma_s(r)}(x) \leq v_{\gamma_s(r)} \cdot \max(f)$ for all $x \in \mathcal{S}(f)$ and $r \in I_\gamma$. So r_1 exists, since, as $r \to \max_\gamma$, we have $v_{\gamma_s(r)} \cdot \max(f) \to 0$ while $\gamma_k(r)$ is increasing. For the existence of r_0 note that, for every $r \in I_\gamma$, the function $f_{\gamma_s(r)}(x)$ is continuous in x and strictly positive for every $x \in \mathcal{S}(f)$, so we have $\min(f_{\gamma_s(r)}(x)) > 0$ for any $r \in I_\gamma$. So r_0 exists since, as $r \to 0$, we have $\gamma_k(r) \to 0$ while $\min(f_{\gamma_s(r)}(x))$ is increasing.

Now, the function $f_{\gamma_s(r)}(x)$ is decreasing and continuous in r, and $\gamma_k(r)$ is continuous and strictly increasing in r. Since $f_{\gamma_s(r_0)}(x) \geq \gamma_k(r_0)$ and $f_{\gamma_s(r_1)}(x) < \gamma_k(r_1)$, we have that $h^\gamma(x)$ is the unique number $r \in [r_0, r_1]$ such that $f_{\gamma_s(r)}(x) = \gamma_k(r)$, as required. ■

Lemma 100 *Let $\varepsilon > 0$, and let γ be a slicing curve that is covering. There is $\delta > 0$ such that, if $\gamma_s(r) < \delta$ for every $r \in I_\gamma$, then $||h^{\overline{\gamma}} - f||_\infty < \varepsilon$.*

Proof Using Lemma 96, let δ be such that if $s < \delta$, then, for all $x \in \mathcal{S}(f)$, we have $|f_s(x)/v_s - f(x)| < \varepsilon$. By definition of $\overline{\gamma}$, we have $h^{\overline{\gamma}}(x) = \varphi(h^\gamma(x))$. Using Lemma 99, this implies that, for all $x \in \mathcal{S}(f)$,

$$h^{\overline{\gamma}}(x) = \varphi(h^\gamma(x)) = \frac{\gamma_k(h^\gamma(x))}{v_{\gamma_s(h^\gamma(x))}} = \frac{f_{\gamma_s(h^\gamma(x))}(x)}{v_{\gamma_s(h^\gamma(x))}},$$

So, if $\gamma_s(r) < \delta$ for every $r \in I_\gamma$, then $|h^{\overline{\gamma}}(x) - f(x)| < \varepsilon$ as $\gamma_s(h^\gamma(x)) < \delta$. ■

Now, let T be a topological space, and let $\mathcal{U} = \{U_i\}_{i=1}^n$ be an open cover of T, with $U_i \neq \emptyset$ for all i. Consider the graph $G_\mathcal{U}$ with vertex set $\{1, \ldots, n\}$, and with an edge (i, j) if $U_i \cap U_j \neq \emptyset$.

Lemma 101 *If T is a connected topological space, and $\mathcal{U} = \{U_i\}_{i=1}^n$ is a finite open cover of T with $U_i \neq \emptyset$ for all i, then the graph $G_\mathcal{U}$ is connected.*

Proof We use induction on n. We assume the statement for $n-1$, and prove it for n. If $U_1 \cap U_i = \emptyset$ for all $1 < i \leq n$, then we can write $T = U_1 \sqcup (\cup_{i=2}^n U_i)$, contradicting the assumption that T is connected. So, we can choose $1 < j \leq n$ such that $U_1 \cap U_j \neq \emptyset$.

Let $\mathcal{U}' = \{U_1 \cup U_j, U_2, \ldots, \hat{U_j}, \ldots, U_n\}$, where $\hat{U_j}$ indicates that we remove U_j. Then $\mathcal{U}'$ is an open cover of T with $n-1$ elements, so by induction, $G_{\mathcal{U}'}$ is connected. Now, $G_{\mathcal{U}'}$ is obtained by contracting the edge $\{1, j\}$ of $G_{\mathcal{U}}$. Thus, $G_{\mathcal{U}}$ is connected. ■

Lemma 102 *Let $\varepsilon > 0$ and let γ be a slicing curve that is covering. There is $\delta > 0$ such that, if $\gamma_s(r), \gamma_t(r) < \delta$ for every $r \in I_\gamma$, then $\mathrm{L}(f)^{\overline{\gamma}}$ and $H(f)$ are ε-interleaved.*

Proof Using the fact that f is uniformly continuous, Lemma 100, and Lemma 96, choose $\delta > 0$ such that, for all $x, y \in \mathbb{R}^d$, if $||x - y|| < \delta$, then $|f(x) - f(y)| < \varepsilon/2$, and such that, if $\gamma_s(r) < \delta$ for all $r \in I_\gamma$, then $||h^{\overline{\gamma}} - f||_\infty < \varepsilon/2$, and such that, if $s < \delta$, then $||f_s/v_s - f||_\infty < \varepsilon/2$. Let γ be a slicing curve that is covering, and such that $\gamma_s(r), \gamma_t(r) < \delta$ for every $r \in I_\gamma$. We show that we have $\mathrm{L}(f)^{\overline{\gamma}}(r) \preceq H(f)(r-\varepsilon)$ and $H(f)(r) \preceq \mathrm{L}(f)^{\overline{\gamma}}(r-\varepsilon)$ in $\mathsf{C}(\mathcal{S}(f))$ for all $r > \varepsilon$.

By Lemma 100, for any $x \in \mathcal{S}(f)$, if x is contained in a cluster of $\mathrm{L}(f)^{\overline{\gamma}}(r)$, then x is contained in a cluster of $H(f)(r-\varepsilon)$; and if x is contained in a cluster of $H(f)(r)$, then x is contained in a cluster of $\mathrm{L}(f)^{\overline{\gamma}}(r-\varepsilon)$. Next, say $x, y \in C \in \mathrm{L}(f)^{\overline{\gamma}}(r)$ for $r > \varepsilon$. We show that x and y belong to the same cluster of $H(f)(r-\varepsilon)$. Let $r_0 = \varphi^{-1}(r)$, $s_0 = \gamma_s(r_0)$, $t_0 = \gamma_t(r_0)$, and $k_0 = \gamma_k(r_0)$. So, by the definition of φ, we have $r = k_0/v_{s_0}$. Note we have $t_0, s_0 < \delta$. As $x, y \in C$, there is a chain $x_0, \ldots, x_n \in \mathcal{S}(f)$ with $x = x_0, y = x_n$, such that $f_{s_0}(x_i) \geq k_0$ and $||x_i - x_{i+1}|| \leq t_0$ for all i. Dividing by v_{s_0}, we have $f_{s_0}(x_i)/v_{s_0} \geq k_0/v_{s_0} = r$. Let $0 \leq i \leq n-1$, and let $\alpha_i : [0,1] \to \mathbb{R}^d$ parameterize the straight-line path from x_i to x_{i+1}. Let $q \in [0,1]$. Because $||x_i - \alpha_i(q)|| \leq t_0 < \delta$, we have $|f(x_i) - f(\alpha_i(q))| < \varepsilon/2$. As $s_0 < \delta$, we have $|f_{s_0}(x_i)/v_{s_0} - f(x_i)| < \varepsilon/2$. So, we have $f(\alpha_i(q)) > r - \varepsilon$. The concatenation of the α_i is therefore a path in $\mathcal{S}(f)$ from x to y such that $f(p) > r - \varepsilon$ for all points p on the path. So, x and y belong to the same cluster of $H(f)(r-\varepsilon)$.

Finally, let $x, y \in C \in H(f)(r)$. We show that x and y belong to the same cluster of $\mathrm{L}(f)^{\overline{\gamma}}(r-\varepsilon)$ for $r > \varepsilon$. Write $t_\varepsilon = \gamma_t(\varphi^{-1}(r-\varepsilon)) > 0$; we will show that there is a t_ε-chain $(x = x_0, \ldots, x_n = y) \in C$. Let $\{P_i\}_{i \in I}$ be the set of path components of C. For each $i \in I$, let $P_i^{t_\varepsilon} = \cup_{a \in P_i} B_C(a, t_\varepsilon)$. Then, $\{P_i^{t_\varepsilon}\}_{i \in I}$ is an open cover of C. Since C is compact, there is a finite $J \subseteq I$ such that $\mathcal{U} = \{P_i^{t_\varepsilon}\}_{i \in J}$ is an open cover of C.

Now, say $i, j \in J$, and $P_i^{t_\varepsilon} \cap P_j^{t_\varepsilon} \neq \emptyset$. We show that for any $a \in P_i$ and any $b \in P_j$, there is a t_ε-chain in C connecting a and b. Choose $w \in P_i^{t_\varepsilon} \cap P_j^{t_\varepsilon}$; by definition, there is $w_i \in P_i$ and $w_j \in P_j$ such that $||w - w_i|| < t_\varepsilon$, and $||w - w_j|| < t_\varepsilon$. Then, there is a t_ε-chain in P_i connecting a to w_i, and a t_ε-chain in P_j connecting b to w_j, which together give a t_ε-chain in C connecting a and b. By Lemma 101, the graph $G_{\mathcal{U}}$ is connected. So, there is a t_ε-chain in C connecting x and y. ■

Definition 103 *A slicing curve $\gamma = (\gamma_s, \gamma_t, \gamma_k) : I_\gamma \to \mathbb{R}^3_{>0}$ is **non-singular in each component** if it is continuously differentiable and the derivatives γ_s', γ_t', and γ_k' never vanish.*

Definition 104 *We say that a family $\{\gamma^\theta\}_{\theta \in \Theta}$ of slicing curves is an **admissible family** if each γ^θ is covering and non-singular in each component, and if, for every $b > 0$, there is $\theta \in \Theta$ such that for all $r \in I_{\gamma^\theta}$, we have $\gamma_s^\theta(r), \gamma_t^\theta(r) < b$.*

Lemma 105 *Let γ be a slicing curve that is non-singular in each component. Let H and E be $\mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}^{\mathrm{op}}$-hierarchical clusterings of sets X and Y respectively, and let $R \subseteq X \times Y$ be a correspondence. Assume there exists $r \in I_\gamma$ with $H^\gamma(r) = \emptyset$. For every $\varepsilon > 0$ there is $\delta > 0$ such that, if H and E are (δ, δ, δ)-interleaved with respect to R, then H^γ and E^γ are ε-interleaved with respect to R.*

Proof Choose $r_0 \in I_\gamma$ such that $H^\gamma(r_0) = \emptyset$ and let $\varepsilon > 0$. Without loss of generality, suppose that $\varepsilon < r_0$. Choose $r_1, r_2 \in I_\gamma$ such that $r_0 < r_1 < r_2 < \max_\gamma$. Let $c = \min_{r \in [\varepsilon/3, r_2]} \{\min(|\gamma_s'(r)|, |\gamma_t'(r)|, \gamma_k'(r))\} > 0$. Let $\delta = \min(c(r_1 - r_0), c\varepsilon/3)$. We show that if H and E are (δ, δ, δ)-interleaved w.r.t. R, then H^γ and E^γ are ε-interleaved w.r.t. R.

Let $H^\gamma_{\varepsilon/3}$ be the I_γ-hierarchical clustering of X with $H^\gamma_{\varepsilon/3}(r) = H^\gamma(r)$ for $r \in I_\gamma$ with $r > \varepsilon/3$, and $H^\gamma_{\varepsilon/3}(r) = \{X\}$ else. Define $E^\gamma_{\varepsilon/3}$ in the same way. Since $H^\gamma_{\varepsilon/3}$ and H^γ are $\varepsilon/3$-interleaved, and similarly for E^γ, it suffices to show that $H^\gamma_{\varepsilon/3}$ and $E^\gamma_{\varepsilon/3}$ are $\varepsilon/3$-interleaved w.r.t. R. We first show that $E^\gamma(r_1) = \emptyset$. It follows that $E^\gamma(r) = H^\gamma(r) = \emptyset$ for all $r \geq r_1$. By the definition of c, we have $\gamma_s(r_1) + c(r_1 - r_0) \leq \gamma_s(r_0)$, $\gamma_t(r_1) + c(r_1 - r_0) \leq \gamma_t(r_0)$, and $\gamma_k(r_1) - c(r_1 - r_0) \geq \gamma_k(r_0)$. Using these equations, the (δ, δ, δ)-interleaving between H and E, and the assumption that $\delta \leq c(r_1 - r_0)$, we have $\pi_Y^*(E^\gamma)(r_1) \preceq \pi_X^*(H^\gamma)(r_0) = \emptyset$, and thus $E^\gamma(r_1) = \emptyset$. Now, to show that $H^\gamma_{\varepsilon/3}$ and $E^\gamma_{\varepsilon/3}$ are $\varepsilon/3$-interleaved w.r.t. R, it suffices to show that, for $r \in (2\varepsilon/3, r_1)$, we have $\pi_X^*(H^\gamma_{\varepsilon/3})(r) \preceq \pi_Y^*(E^\gamma_{\varepsilon/3})(r - \varepsilon/3)$, and $\pi_Y^*(E^\gamma_{\varepsilon/3})(r) \preceq \pi_X^*(H^\gamma_{\varepsilon/3})(r - \varepsilon/3)$. Again by the definition of c, we have $\gamma_s(r) + c(\varepsilon/3) \leq \gamma_s(r - \varepsilon/3)$, $\gamma_t(r) + c(\varepsilon/3) \leq \gamma_t(r - \varepsilon/3)$, and $\gamma_k(r) - c(\varepsilon/3) \geq \gamma_k(r - \varepsilon/3)$ for any $r \in (2\varepsilon/3, r_2)$. Using these equations, the (δ, δ, δ)-interleaving between H and E, and the assumption that $\delta \leq c\varepsilon/3$, we obtain the desired relations. ■

Lemma 106 *Let $\{\gamma^\theta\}_{\theta \in \Theta}$ be an admissible family of slicing curves, and let X_n be a sample of f. For every $\varepsilon > 0$ there exist $\theta \in \Theta$ and $\delta > 0$ such that, if $d_{\mathrm{P}}(\mu_n, \mu_f), d_{\mathrm{H}}(X_n, \mathcal{S}(f)) < \delta$, then $\mathrm{L}(X_n)^{\overline{\gamma^\theta}}$ and $H(f)$ are ε-interleaved with respect to the closest point correspondence $R_c \subseteq X_n \times \mathcal{S}(f)$.*

Proof By Lemma 102, and the fact that the family $\{\gamma^\theta\}_{\theta \in \Theta}$ is admissible, we can fix the parameter θ so that $H(f)$ and $\mathrm{L}(f)^{\overline{\gamma^\theta}}$ are $\varepsilon/2$-interleaved. It is then enough to show that we can choose $\delta > 0$ such that, if $d_{\mathrm{P}}(\mu_n, \mu_f) < \delta$ and $d_{\mathrm{H}}(X_n, \mathcal{S}(f)) < \delta$, then $\mathrm{L}(f)^{\overline{\gamma^\theta}}$ and $\mathrm{L}(X_n)^{\overline{\gamma^\theta}}$ are $\varepsilon/2$-interleaved with respect to R_c. To see that this can be done, note that the operation $\mathrm{L}(-)^{\overline{\gamma^\theta}}$ is the composite of $\mathrm{L}(-)$ and slicing by $\overline{\gamma^\theta}$, and apply Theorem 38 (note that the interleaving constructed in the proof is with respect to R_c) and Lemma 105, where the last result applies by Lemma 99, since $\overline{\gamma^\theta}$ is covering. ■

Lemma 107 *Let $(\mathcal{M}, d, \mu)$ be a compact metric probability space with full support and let X_n be an i.i.d. n-sample of $\mathcal{M}$, seen as a subspace of $\mathcal{M}$. Let $\varepsilon > 0$. Then, the probability that $\max(d_{\mathrm{P}}(\mu_n, \mu), d_{\mathrm{H}}^{\mathcal{M}}(X_n, \mathcal{M})) > \varepsilon$ goes to 0 as $n \to \infty$. Here μ_n is the empirical measure given by the sample X_n.*

Proof We show that $P(d_{\mathrm{H}}^{\mathcal{M}}(X_n, \mathcal{M}) > \varepsilon) \to 0$ as $n \to \infty$. Since $\mathcal{M}$ is compact, for any $\varepsilon > 0$ we can cover $\mathcal{M}$ with finitely many ε-balls, all of which have positive measure, by assumption. This implies that the probability that there is a sample point inside of each of these goes to 1 as n goes to ∞. We conclude by showing that $P(d_{\mathrm{P}}(\mu_n, \mu) > \varepsilon) \to 0$ as $n \to \infty$. This follows from the facts that i.i.d. samples of a separable metric space with the empirical measure converge weakly to the sampled space, in probability (Parthasarathy, 1972, Chapter II, Theorem 7.1), and that weak convergence implies convergence in the Prokhorov distance (Billingsley, 1999, Section 6). ■

Theorem 108 *Let $\{\gamma^\theta\}_{\theta\in\Theta}$ be an admissible family of slicing curves. The hierarchical clustering algorithm $\overline{\gamma}$-*`link`* with parameter space Θ, defined using any kernel K, is CI-consistent with respect to any continuous, compactly supported probability density function $f : \mathbb{R}^d \to \mathbb{R}$.*

Proof The theorem follows from Lemma 106 and the fact that samples converge to the space being sampled, Lemma 107. ■

A.4 Details from Section 5

Lemma 109 *Let $I \subseteq \mathbb{R}$ be an interval, let H be an I-hierarchical clustering, and let $\mathbf{C} < \mathbf{D} \in \mathsf{PC}(H)$. Then, there exists $\mathbf{E} \in \mathsf{PC}(H)$ such that $\mathbf{E} < \mathbf{D}$ and $\mathbf{C}$ and $\mathbf{E}$ are incomparable.*

Proof Let $r < s \in I$ and $C \in H(r)$ and $D \in H(s)$ such that $[C] = \mathbf{C}$ and $[D] = \mathbf{D}$. Since $\mathbf{C} \neq \mathbf{D}$, there must exist $t \in [r, s]$ and $E \in H(t)$ such that $E \subseteq D$ but $C \not\subseteq E$. Then, the persistent cluster $\mathbf{E} \in \mathsf{PC}(H)$ satisfies the required conditions. ■

Lemma 110 *Let $I \subseteq \mathbb{R}$ be an interval, H be an I-hierarchical clustering, and let $\mathbf{C}, \mathbf{D} \in \mathsf{PC}(H)$. If $U(\mathbf{C}) \cap U(\mathbf{D}) \neq \emptyset$, then $U(\mathbf{C}) \subseteq U(\mathbf{D})$ or $U(\mathbf{D}) \subseteq U(\mathbf{C})$. Moreover, if $U(\mathbf{C}) = U(\mathbf{D})$, then $\mathbf{C} = \mathbf{D} \in \mathsf{PC}(H)$.*

Proof Say $U(\mathbf{C}) \cap U(\mathbf{D}) \neq \emptyset$, and let $x \in U(\mathbf{C}) \cap U(\mathbf{D})$. Choose $r \in \mathsf{life}(\mathbf{C})$ with $x \in \mathbf{C}(r)$ and $r' \in \mathsf{life}(\mathbf{D})$ with $x \in \mathbf{D}(r')$. Without loss of generality, $r \leq r'$, and thus $\mathbf{C}(r) \subseteq \mathbf{D}(r')$. We show $U(\mathbf{C}) \subseteq U(\mathbf{D})$. Say $y \in U(\mathbf{C})$. Choose $i \in \mathsf{life}(\mathbf{C})$ with $y \in \mathbf{C}(i)$. We may assume $i \geq r$. If $r' \in \mathsf{life}(\mathbf{C})$, then $\mathbf{C}(r') = \mathbf{D}(r')$, and thus $\mathbf{C} = \mathbf{D}$. So, assume $r' \notin \mathsf{life}(\mathbf{C})$. As $\mathsf{life}(\mathbf{C})$ is an interval, $i < r'$, and thus $\mathbf{C}(i) \subseteq \mathbf{D}(r')$. So, $y \in \mathbf{D}(r') \subseteq U(\mathbf{D})$.

Now, assume $U(\mathbf{C}) = U(\mathbf{D})$. As in the first step above, we have without loss of generality $\mathbf{C}(r) \subseteq \mathbf{D}(r')$ for some $r \leq r'$. We show $\mathbf{C}(r) \sim \mathbf{D}(r')$. Let $r'' \in [r, r']$, and let $A, A' \in H(r'')$ with $A, A' \subseteq \mathbf{D}(r')$. We have $A, A' \subseteq U(\mathbf{D}) = U(\mathbf{C})$. Let $B \in H(r'')$ be the cluster such that $\mathbf{C}(r) \subseteq B$. It is straightforward to show $A = B$, and similarly $A' = B$. Thus $A = A'$, finishing the proof. ■

Lemma 111 *Let $I \subseteq \mathbb{R}$ be an interval, let H be an I-hierarchical clustering, and let $\mathbf{C} < \mathbf{D} \in \mathsf{PC}(H)$. Then, the set $\{\mathbf{D} \in \mathsf{PC}(H) : \mathbf{D} \geq \mathbf{C}\}$ is linearly ordered in $\mathsf{PC}(H)$.*

Proof Let $r, r' \in I$, $D \in H(r)$, $D' \in H(r')$ such that $[D] = \mathbf{D}$ and $[D'] = \mathbf{D}'$. Assume, without loss of generality, that $r \leq r'$. Since $\mathbf{C} \leq \mathbf{D}, \mathbf{D}'$, we have $U(\mathbf{C}) \subseteq D$ and $U(\mathbf{C}) \subseteq D'$. If follows that $U(\mathbf{D}) \cap U(\mathbf{D}') \neq \emptyset$ and thus $\mathbf{D} \leq \mathbf{D}'$ or $\mathbf{D}' \leq \mathbf{D}$, by Lemma 110. ■

Definition 112 *Let H be a one-parameter hierarchical clustering and let $\mathbf{C} \in \mathsf{PC}(H)$. The **successor** of $\mathbf{C}$ is, if it exists, the unique minimal element of $\{\mathbf{D} \in \mathsf{PC}(H) : \mathbf{D} > \mathbf{C}\}$, where uniqueness follows from the fact that $\{\mathbf{D} \in \mathsf{PC}(H) : \mathbf{D} > \mathbf{C}\}$ is linearly ordered (Lemma 111).*

We now recall the terminology of persistence modules; we refer the interested reader to Chazal et al. (2016); Bauer and Lesnick (2015) for details. Fix a field $\mathbb{F}$. Let $I \subseteq \mathbb{R}$ be an interval, and let H be an I-hierarchical clustering of a set X. For each $r \in I$, consider the vector space $\mathbb{F}H(r)$ that is freely generated by the clusters of $H(r)$. For $r \leq r'$, there is a linear map $\mathbb{F}H(r) \to \mathbb{F}H(r')$ defined on the basis given above by $H(r \leq r')$. This data gives a functor $\mathbb{F}H : I \to \mathbb{F}$-**vec**, where $\mathbb{F}$-**vec** is the category of vector spaces over $\mathbb{F}$. Analogously to Definition 4, we call such a functor an ***I*-persistence module**.

Let $I \subseteq \mathbb{R}$ be an interval and let $M : I \to \mathbb{F}$-**vec** be an I-persistence module. Given $r \leq r' \in I$, define the **rank invariant** (Zomorodian and Carlsson, 2005) of M at $r \leq r'$ as $\mathrm{rk}(M)(r \leq r') = \mathrm{rk}(M(r) \to M(r'))$. Note that $\mathrm{rk}(H) = \mathrm{rk}(\mathbb{F}H)$.

For our purposes, given a set A, a **multiset** of elements of A consists of an indexing set J and a function $a : J \to A$. We usually denote such a multiset by $\{a_j\}_{j \in J}$. We say that two multisets $\{a_j\}_{j \in J}$ and $\{b_k\}_{k \in K}$ of elements of A are **equal** if there exists a bijection $\beta : J \to K$ such that $a_j = b_{\beta(j)}$ for all $j \in J$.

Let $B \subseteq I \subseteq \mathbb{R}$ be inclusions of intervals. The I-persistence module $\mathbb{F}_B : I \to \mathbb{F}$-**vec** is the functor taking the value $\mathbb{F}$ on every $r \in B$ and the value 0 elsewhere, with structure morphism being the identity $\mathbb{F} \to \mathbb{F}$ whenever that is possible. The following theorem is due to Crawley-Boevey.

Theorem 113 (Crawley-Boevey 2015, Theorem 1.1) *Let $\mathbb{F}$ be a field, let $I \subseteq \mathbb{R}$ be an interval, and let $M : I \to \mathbb{F}$-**vec** be an I-persistence module. If M is pointwise finite-dimensional, then there exists a unique multiset of intervals $\{B_j \subseteq I\}_{j \in J}$ such that M is isomorphic to $\bigoplus_{j \in J} \mathbb{F}_{B_j}$.*

Lemma 114 *Let $I \subseteq \mathbb{R}$ be an interval. Let H be a pointwise finite I-hierarchical clustering and let $\mathbb{F}$ be any field. Then $\mathcal{B}(H)$ is the unique multiset $\{B_j \subseteq I\}_{j \in J}$ such that $\mathbb{F}H \cong \bigoplus_{j \in J} \mathbb{F}_{B_j}$.*

Proof By definition, the I-persistence module $\mathbb{F}H : I \to \mathbb{F}$-**vec** is pointwise finite dimensional, in the sense of Crawley-Boevey (2015). Thus, by Theorem 113, there exist a unique multiset of intervals $\{B_j \subseteq I\}_{j \in J}$ such that $\mathbb{F}H \cong \bigoplus_{j \in J} \mathbb{F}_{B_j}$. By unfolding definitions, we get

$$\mathrm{rk}\left(\bigoplus_{j \in J} \mathbb{F}_{B_j}\right)(r \leq r') = \sum_{j \in J} \mathrm{rk}\left(\mathbb{F}_{B_j}\right)(r \leq r') = |\{j \in J : r, r' \in B_j\}|,$$

so $\{B_j \subseteq I\}_{j\in J}$ is a barcode for H.

The fact that the barcode is unique is a particular case of (Botnan et al., 2022, Proposition 2.8), where the poset P is taken to be I and the collections of intervals $\widehat{\mathcal{I}}$ and $\mathcal{I}$ are all intervals included in I. ■

Proof (of Theorem 67) This follows at once from Theorem 113 and Lemma 114. ■

Lemma 115 *Let H be a finite I-hierarchical clustering. Then*

$$|\mathsf{leaves}(H)| = \max\{k \in \mathbb{N} : \exists\{r_j \in I\}_{1\leq j\leq k}, \{D_j \in H(r_j)\}_{1\leq j\leq k} \text{ s.t. } D_i \subseteq D_j \Rightarrow i = j\},$$

and any set $\{D_j \in H(r_j)\}_{1\leq j\leq k}$ attaining the maximum must be such that $\{[D_j] \in \mathsf{PC}(H)\} = \mathsf{leaves}(H)$.

Proof The inequality ($\leq$) follows immediately by taking the set of clusters $\{D_j\}$ to be a set of representatives of the leaves of H. The inequality ($\geq$) follows from the fact that, given a set $\{D_j \in H(r_j)\}_{1\leq j\leq k}$ as in the statement, the set $\{[D_j] \in \mathsf{PC}(H)\}_{1\leq j\leq k}$ forms an antichain of $\mathsf{PC}(H)$ and thus its cardinality is bounded above by the cardinality of the set of minimal elements of $\mathsf{PC}(H)$, which is, by definition, the set of leaves. To prove the last claim, note that any antichain of $\mathsf{PC}(H)$ of cardinality $|\mathsf{leaves}(H)|$ must necessarily be the set $\mathsf{leaves}(H)$ itself, since $\mathsf{PC}(H)$ forms a forest. ■

Lemma 116 *Let H be a finite I-hierarchical clustering of a set X. Assume that H is not constantly empty and let $\mathbf{C} \in \mathsf{leaves}(H)$. Then*

1. *$H \setminus \mathbf{C}$ is a finite I-hierarchical clustering;*
2. *$|\mathsf{leaves}(H \setminus \mathbf{C})| = |\mathsf{leaves}(H)| - 1$;*
3. *If $|\mathsf{leaves}(H)| \geq 2$, then $\min_{\mathbf{D}\in\mathsf{leaves}(H)} \mathsf{length}(\mathbf{D}) \leq \min_{\mathbf{D}'\in\mathsf{leaves}(H\setminus\mathbf{C})} \mathsf{length}(\mathbf{D}')$.*

Proof Note first that if $D_1 \in H(r_1)$ and $D_2 \in H(r_2)$ are such that $[D_1] = [D_2]$ in $\mathsf{PC}(H)$ and $[D_1] \neq \mathbf{C}$, then $[D_1] = [D_2]$ in $\mathsf{PC}(H \setminus \mathbf{C})$. Using this fact, we can define a function $\varphi : \mathsf{PC}(H) \setminus \mathbf{C} \to \mathsf{PC}(H \setminus \mathbf{C})$ as follows. For $\mathbf{D} \in \mathsf{PC}(H) \setminus \mathbf{C}$, pick any $D \in \mathbf{D}$, and let $\varphi(\mathbf{D}) = [D] \in \mathsf{PC}(H \setminus \mathbf{C})$. We now prove that φ is surjective, which implies the first statement of the lemma. Let $\mathbf{D} \in \mathsf{PC}(H \setminus \mathbf{C})$, and let $D \in \mathbf{D}$. Then $[D] \neq \mathbf{C}$ in $\mathsf{PC}(H)$ by definition of $H \setminus \mathbf{C}$. So, $\varphi([D]) = \mathbf{D}$.

Next, we show that, when φ is restricted to $\mathsf{leaves}(H) \setminus \mathbf{C}$, φ is a bijection between $\mathsf{leaves}(H) \setminus \mathbf{C}$ and $\mathsf{leaves}(H \setminus \mathbf{C})$; this proves the second statement of the lemma. For this, we will use the following fact, which is straightforward to check. If G is an I-hierarchical clustering, and $\mathbf{A} \in \mathsf{PC}(G)$, then $\mathbf{A}$ is a leaf if and only if for any $r \in \mathsf{life}(\mathbf{A})$ and for any $r' \in I$ with $r' < r$, there is at most one cluster $B \in H(r')$ with $B \subseteq \mathbf{A}(r)$. Now, let $\mathbf{D} \in \mathsf{leaves}(H) \setminus \mathbf{C}$. We need to show $\varphi(\mathbf{D})$ is a leaf. Let $D = \mathbf{D}(r)$ for some r, so that $\varphi(\mathbf{D}) = [D]$. If there is $r' < r$ and $B, B' \in (H \setminus \mathbf{C})(r')$ with $B, B' \subseteq D$, then as $\mathbf{D}$ is a

leaf of H, we have $B = B'$. Let $s \in \mathsf{life}(\varphi(\mathbf{D}))$, and let $s' \in I$ with $s' < s$. If $s' < r$, then we have shown that there is at most one cluster in $(H \setminus \mathbf{C})(s')$ contained in $\varphi(\mathbf{D})(s)$. If $r \leq s'$, then as $\varphi(\mathbf{D})(r) \sim \varphi(\mathbf{D})(s)$, there is exactly one cluster in $(H \setminus \mathbf{C})(s')$ contained in $\varphi(\mathbf{D})(s)$. So, $\varphi(\mathbf{D})$ is a leaf.

We show that $\varphi|_{\mathsf{leaves}(H)\setminus\mathbf{C}}$ is injective. Say $\mathbf{D}, \mathbf{E} \in \mathsf{leaves}(H) \setminus \mathbf{C}$ and $\mathbf{D} \neq \mathbf{E}$. Let $D = \mathbf{D}(r)$ and $E = \mathbf{E}(r')$ for some r, r'. As $\mathbf{D}$ and $\mathbf{E}$ are distinct leaves, Lemma 110 implies $D \cap E = \emptyset$. As $D = \varphi(\mathbf{D})(r)$ and $E = \varphi(\mathbf{E})(r')$, we have $\varphi(\mathbf{D}) \neq \varphi(\mathbf{E})$.

Next we show that the image of $\varphi|_{\mathsf{leaves}(H)\setminus\mathbf{C}}$ is $\mathsf{leaves}(H \setminus \mathbf{C})$. Say $\mathbf{D} \in \mathsf{leaves}(H \setminus \mathbf{C})$. Let $D = \mathbf{D}(r)$ for some r. If $[D] \in \mathsf{leaves}(H)$, then we are done, since $\varphi([D]) = \mathbf{D}$. So, say $[D]$ is not a leaf of H. Then there is $r' \in I$ with $r' < r$ and $B, B' \in H(r')$ with $B \neq B'$ and $B, B' \subseteq D$. Without loss of generality, $B \notin (H \setminus \mathbf{C})(r')$, so $[B] = \mathbf{C}$ in $\mathsf{PC}(H)$. Thus, $[B'] \neq \mathbf{C}$, so that $B' \in (H \setminus \mathbf{C})(r')$. As $\mathbf{D}$ is a leaf of $H \setminus \mathbf{C}$, we have $[B'] = \mathbf{D}$ in $\mathsf{PC}(H \setminus \mathbf{C})$, so that $\varphi([B']) = \mathbf{D}$. We show that $[B'] \in \mathsf{leaves}(H) \setminus \mathbf{C}$. Say there is $r'' < r'$ and $A, A' \in H(r'')$ with $A, A' \subseteq B'$. Then $A \cap B = \emptyset$ and $A' \cap B = \emptyset$, so that $[A] \neq \mathbf{C}$ and $[A'] \neq \mathbf{C}$ in $\mathsf{PC}(H)$. Thus, $A, A' \in (H \setminus \mathbf{C})(r'')$, and since $\mathbf{D}$ is a leaf of $H \setminus \mathbf{C}$, $A = A'$.

We have shown that $\varphi|_{\mathsf{leaves}(H)\setminus\mathbf{C}}$ is a bijection between $\mathsf{leaves}(H) \setminus \mathbf{C}$ and $\mathsf{leaves}(H \setminus \mathbf{C})$, proving the second statement of the lemma. We will also use this fact to prove the third statement of the lemma. Note first that if $\mathbf{D} \in \mathsf{PC}(H) \setminus \mathbf{C}$, then $\mathsf{life}(\mathbf{D}) \subseteq \mathsf{life}(\varphi(\mathbf{D}))$, and so $\mathsf{length}(\mathbf{D}) \leq \mathsf{length}(\varphi(\mathbf{D}))$. Now, say $|\mathsf{leaves}(H)| \geq 2$, so that $|\mathsf{leaves}(H \setminus \mathbf{C})| \geq 1$. Choose $\mathbf{E}' \in \mathsf{leaves}(H \setminus \mathbf{C})$ such that $\mathsf{length}(\mathbf{E}') = \min_{\mathbf{D}' \in \mathsf{leaves}(H\setminus\mathbf{C})} \mathsf{length}(\mathbf{D}')$. Let $\mathbf{E} \in \mathsf{leaves}(H) \setminus \mathbf{C}$ be such that $\varphi(\mathbf{E}) = \mathbf{E}'$. Then $\min_{\mathbf{D} \in \mathsf{leaves}(H)} \mathsf{length}(\mathbf{D}) \leq \mathsf{length}(\mathbf{E}) \leq \mathsf{length}(\mathbf{E}') = \min_{\mathbf{D}' \in \mathsf{leaves}(H\setminus\mathbf{C})} \mathsf{length}(\mathbf{D}')$. ■

Lemma 117 *Let H be a finite I-hierarchical clustering that is not constantly empty, and let $\mathbf{C}$ be a minimal leaf of H. Then $\mathbb{F}H \cong \mathbb{F}(H \setminus \mathbf{C}) \oplus \mathbb{F}_{\mathsf{life}(\mathbf{C})}$.*

Proof Let H be a finite I-hierarchical clustering of a set X. For convenience, denote $H' = H \setminus \mathbf{C}$. We start by defining a morphism of I-persistence modules $\mathbb{F}H' \oplus \mathbb{F}_{\mathsf{life}(\mathbf{C})} \to \mathbb{F}H$ as a sum of two morphisms $\phi : \mathbb{F}H' \to \mathbb{F}H$ and $\psi : \mathbb{F}_{\mathsf{life}(\mathbf{C})} \to \mathbb{F}H$. Let $\phi : \mathbb{F}H' \to \mathbb{F}H$ be given by mapping the basis element of $\mathbb{F}H'(r)$ corresponding to $D \in H'(r)$ to the basis element of $\mathbb{F}H(r)$ corresponding to the same cluster $D \in H(r)$.

To define ψ, we consider two cases. If $\mathbf{C}$ has no successor in $\mathsf{PC}(H)$, then the morphism $\psi : \mathbb{F}_{\mathsf{life}(\mathbf{C})} \to \mathbb{F}H$ defined by mapping the basis element $1 \in \mathbb{F} = \mathbb{F}_{\mathsf{life}(\mathbf{C})}(r)$ to the basis element of $\mathbb{F}H(r)$ corresponding to $\mathbf{C}(r) \in H(r)$ is well-defined.

If $\mathbf{C}$ does have a successor $\mathbf{D}$, let $\mathbf{A}$ denote any leaf of H smaller than $\mathbf{D}$ and different from $\mathbf{C}$, which must exist by Lemma 109. Since $\mathbf{C}$ is a minimal leaf, for each $r \in \mathsf{life}(\mathbf{C})$, there exists $r' \in \mathsf{life}(\mathbf{A})$ such that $r' \leq r$. Let $\mathbf{A}' : \mathsf{life}(\mathbf{C}) \to \mathsf{C}(X)$ denote the persistent cluster defined as $\mathbf{A}'(r) = H(r' \leq r)(\mathbf{A}(r')) \in H(r)$. Let $\psi : \mathbb{F}_{\mathsf{life}(\mathbf{C})} \to \mathbb{F}H$ be defined by mapping the basis element $1 \in \mathbb{F} = \mathbb{F}_{\mathsf{life}(\mathbf{C})}(r)$ to the subtraction of basis elements of $\mathbb{F}H(r)$ given by $\mathbf{C}(r) - \mathbf{A}'(r)$. In order to see that this is well-defined, note that, if $t \in \mathsf{life}(\mathbf{D})$, then $H(r \leq t)(\mathbf{C}(r)) = H(r \leq t)(\mathbf{A}'(r))$, since both $\mathbf{C}$ and $\mathbf{A}$ are smaller than $\mathbf{D}$ in $\mathsf{PC}(H)$, and thus $\mathbf{C}(r) - \mathbf{A}'(r)$ maps to 0 in $\mathbb{F}H(t)$ as soon as t is larger than all elements in $\mathsf{life}(\mathbf{C})$.

To conclude the proof, it is enough to prove that the morphism $\phi + \psi : \mathbb{F}H' \oplus \mathbb{F}_{\mathsf{life}(\mathbf{C})} \to \mathbb{F}H$ is an epimorphism, since, in that case, it must also by a monomorphism, by dimension-counting. Let $r \in I$; it is clear that all basis elements of $\mathbb{F}H(r)$ corresponding to clusters

$B \in H(r)$ such that $[B] \neq \mathbf{C}$ are in the image of $\phi+\psi$, since they are already in the image of ϕ, by construction. It is then enough to prove that basis elements of $\mathbb{F}H(r)$ corresponding to clusters $C \in H(r)$ such that $[C] = \mathbf{C}$ are in the image of $\phi+\psi$, and for this we necessarily have $r \in \mathsf{life}(\mathbf{C})$. To conclude, note that the element of $\mathbb{F}H(r)$ corresponding to $\mathbf{A}'(r)$ is in the image of ϕ, since it is a basis element of $\mathbb{F}H'(r)$, and the element corresponding to $\mathbf{C}(r) - \mathbf{A}'(r)$ is in the image of ψ, by construction. ■

Proof (of Proposition 71) Note that the process is well-defined and terminates in $|\mathsf{leaves}(H)|$ by Lemma 116. By induction and Lemma 117, we have $\mathbb{F}H = \bigoplus_{1\leq i\leq k} \mathbb{F}_{\mathsf{life}(\mathbf{C}_i)}$, so the claim follows from Lemma 114. ■

Corollary 118 *Let H be a finite hierarchical clustering. Then,*

$$\min_{\mathbf{C}\in\mathsf{leaves}(H)} \mathsf{length}(\mathbf{C}) = \min_{B\in\mathcal{B}(H)} \mathsf{length}(B).$$

Proof This follows from the first step in Proposition 71 and the third claim of Lemma 116. ■

Lemma 119 *For any $\lambda \in \{\lambda^{x,y}_{\mathrm{con}}\}_{x,y>0}$ and any compact metric probability space $\mathcal{M}$, the hierarchical clustering λ-$\mathtt{link}(\mathcal{M})$ is essentially finite. If $f : \mathbb{R}^d \to \mathbb{R}$ is continuous and compactly supported, then $H(f)$ is essentially finite.*

Proof Let H be an $\mathbb{R}$-hierarchical clustering. For the purposes of this proof, and in analogy with the notion of q-tameness introduced in Chazal et al. (2009), we say that H is **q-tame** if, for every $r < r' \in \mathbb{R}$, the cardinality of the image of the function $H(r \leq r') : H(r) \to H(r')$ is finite, and, we say that H is **bounded** if there exist $s \leq t \in \mathbb{R}$ such that H is constant on $(-\infty, s)$ and on (t, ∞). We start by proving that the hierarchical clusterings of interest are q-tame and bounded.

Let $\mathcal{M}$ be a compact metric probability space. Then, the extension of λ-$\mathtt{link}(\mathcal{M})$ to an $\mathbb{R}$-hierarchical clustering is q-tame since λ-$\mathtt{link}(\mathcal{M})$ is pointwise finite, as its values are a single-linkage clustering of a totally bounded metric space. The extension is also bounded, since it is an extension of a hierarchical clustering defined over a finite interval.

Let $f : \mathbb{R}^d \to \mathbb{R}$ be continuous and compactly supported. The hierarchical clustering $H(f)$ is q-tame by Cagliari and Landi (2011, Theorem 2) (or Chazal et al. 2016, Theorem 3.33), and it is bounded since f takes values in a compact set. Then, the result follows from the following claim.

Claim. A bounded $\mathbb{R}$-hierarchical clustering is q-tame if and only if it is essentially finite.

Proof of claim. Let H be a bounded $\mathbb{R}$-hierarchical clustering.

Assume that H is essentially finite. If $r < r' \in \mathbb{R}$, there exists $\tau > 0$ such that $r \leq r' - \tau$; thus $H(r \leq r') = H(r' - \tau \leq r') \circ H(r \leq r' - \tau)$. The image of the function $H(r' - \tau \leq r')$ is finite, since $H_{\geq\tau}$ is a finite hierarchical clustering. It follows that the image of $H(r \leq r')$ is finite, and thus that H is q-tame.

Now assume that H is q-tame, and, towards a contradiction, let $\tau > 0$, assume $\mathsf{PC}(H_{\geq\tau})$ is infinite, and let $\{\mathbf{C}_n\}_{n\geq 0}$ be a countably infinite family of elements of $\mathsf{PC}(H_{\geq\tau})$. Let $s \leq t \in \mathbb{R}$ be such that $H_{\geq\tau}$ is constant on $(-\infty, s)$ and on (t, ∞); such s and t must exist since H is bounded. Only finitely many elements of $\{\mathbf{C}_n\}$ can have support intersecting $(-\infty, s) \cup (t, \infty)$, since $H_{\geq\tau}$ is constant on $(-\infty, s)$ and on (t, ∞). Thus, after taking a subsequence of $\{\mathbf{C}_n\}$ we may assume that all intervals $\mathsf{life}(\mathbf{C}_n)$ are contained in $[s, t]$. Since $[s, t]$ is compact, after taking a subsequence of $\{\mathbf{C}_n\}$, we may assume that, as $n \to \infty$, we have $b_n := \mathsf{birth}(\mathbf{C}_n) \to b$ and $d_n := \mathsf{death}(\mathbf{C}_n) \to d$, with $b \leq d$. Let $s_n = (b_n + d_n)/2$ and let $s = (b + d)/2$. For each n, there is $C_n \in H_{\geq\tau}(s_n)$ with $[C_n] = \mathbf{C}_n$, and therefore there is $C'_n \in H(s_n - \tau)$ with $H(s_n - \tau \leq s_n)(C'_n) = C_n$. As $n \to \infty$, we have $s_n \to s$, so there exists $n_0 \in \mathbb{N}$ such that $s_n - \tau < s - \frac{2\tau}{3} < s - \tau/3 < s_n$ for all $n \geq n_0$. Thus, for $n \geq n_0$, the elements $H(s - \tau \leq s - \frac{\tau}{3})(C'_n)$ form an infinite subset of $H(s - \frac{2\tau}{3} \leq s - \frac{\tau}{3})$, a contradiction. This concludes the proof of the claim. ■

Proof (of Proposition 69) Note that, by Lemma 114, the barcode of a pointwise finite HC H is equal to the barcode of $\mathbb{F}H$ for any field $\mathbb{F}$. If H and E are ε-interleaved with respect to some correspondence, then $\mathbb{F}H$ and $\mathbb{F}E$ are ε-interleaved in the sense of Bauer and Lesnick (2015), so it follows from Bauer and Lesnick (2015, Theorem 6.4) that there exists an ε-matching between the barcodes of H and E. ■

Proof (of Lemma 73) As every permutation can be written as a composition of adjacent transpositions, it suffices to consider modifying an ordering $[\sigma_1, \ldots, \sigma_p]$ by transposing σ_i and σ_{i+1}. If this transposition results in an ordering satisfying the assumptions, then necessarily $f(\sigma_i) = f(\sigma_{i+1})$. There are three cases.

In Case 1, σ_i and σ_{i+1} are both vertices. In this case it is clear that the transposition does not effect the output. In Case 2, one of the simplices is a vertex x and the other an edge e. In this case, since it is admissible to order e before x, x is not an endpoint of e. Thus, processing e does not effect the connected component nor the bar introduced when processing x, so the transposition does not effect the output. In Case 3, $\sigma_i = \{x, y\}$ and $\sigma_{i+1} = \{a, b\}$ are both edges. Let (c_x, u_x) be the connected component containing x after processing σ_{i-1}, and similarly for y, a, b. If $c_x = c_y$ or $c_a = c_b$, then it is clear that the transposition has no effect on the output. So, we assume $c_x \neq c_y$ and $c_a \neq c_b$. If $\{c_x, c_y, c_a, c_b\}$ has 2 or 4 elements, then it is straightforward to check that the transposition has no effect on the output. Say $\{c_x, c_y, c_a, c_b\}$ has 3 elements. Without loss of generality, we assume $c_y = c_b$. Let $r = f(\{x, y\}) = f(\{a, b\})$ and let u_1, u_2, u_3 be u_x, u_y, u_a ordered from smallest to largest. Then, after processing σ_i and σ_{i+1} in either order, we have

$$\begin{aligned}
\texttt{conn_comp} &\leftarrow \big(\texttt{conn_comp} \setminus \big\{(c_x, u_x), (c_y, u_y), (c_a, u_a)\big\}\big) \cup \big\{(c_x \cup c_y \cup c_a, \min(u_x, u_y, u_a))\big\} \\
\texttt{barcode} &\leftarrow \big(\texttt{barcode} \setminus \big\{[u_x, \infty), [u_y, \infty), [u_a, \infty)\big\}\big) \cup \big\{[u_1, \infty), [u_2, r), [u_3, r)\big\}
\end{aligned}$$

So, the transposition does not effect the output. ■

Proof (of Proposition 74) Let X be the set of vertices of G, and let $H = H(G, f)$. By Example 65, H is a finite hierarchical clustering. We begin by noting some useful facts

about Algorithm 1. Note first that if $G = G_1 \cup \cdots \cup G_q$ is the decomposition of G into connected components, then the output of the algorithm on (G, f) is equal to the union of the output of the algorithm on $(G_i, f|_{G_i})$, and similarly the barcode of H is equal to the union of the barcodes of $H(G_i, f|_{G_i})$. So, if G is non-empty, we may assume it is connected.

Second, note that if T is a minimum spanning tree of (G, f), then we have $H(G, f) = H(T, f|_T)$. Furthermore, the output of the algorithm on (G, f) is equal to the output on $(T, f|_T)$. To see this, order the simplices of G such that, among all simplices σ with $f(\sigma) = r$, we first take all the vertices, then all the edges in T, then all the edges not in T. Now, if we run the algorithm on (G, f) and process an edge $e = \{x, y\}$ not in T, then the algorithm does nothing at this step, since T contains a path between x and y such that every edge on this path has f value less than or equal to $f(e)$. So, we may assume G is a tree.

Third, note that if G is a tree with an edge $e = \{x, y\}$ such that $f(e) = f(x)$, then the output of the algorithm on (G, f) is equal to the output on $(\tilde{G}, \tilde{f})$, where $\tilde{G} = G/e$ arises from contracting the edge e, and $\tilde{f}(\sigma) = f(\sigma)$ for $\sigma \notin \{x, y, e\}$, and $\tilde{f}(v_e) = f(y)$, where v_e is the vertex onto which e contracts. And, the barcodes of $H(G, f)$ and $H(\tilde{G}, \tilde{f})$ are equal; to see this, note that $H(G, f)$ and $H(\tilde{G}, \tilde{f})$ are 0-interleaved with respect to the correspondence that identifies x, y with v_e, so that there is a 0-matching between the barcodes of $H(G, f)$ and $H(\tilde{G}, \tilde{f})$ by Proposition 69. So, we may assume G contains no edges e such that $f(e) = f(v)$ for $v \in e$.

Fourth, say that G contains no edges e such that $f(e) = f(v)$ for $v \in e$. Then for $x \in X$, there is a leaf $\mathbf{C}$ of H such that $\mathbf{C}(\mathsf{birth}(\mathbf{C})) = \{x\}$. This defines a bijection $X \cong \mathsf{leaves}(H)$.

We prove the correctness of the algorithm by induction on the number of leaves of H. Consider the case $|\mathsf{leaves}(H)| = 0$. Then G is empty and the output of the algorithm is correct. Consider the case $|\mathsf{leaves}(H)| = 1$. We may assume G contains no edges e such that $f(e) = f(v)$ for $v \in e$, and thus $|X| = 1$. Then it's straightforward to check that the output of the algorithm is correct.

Consider the case $|\mathsf{leaves}(H)| \geq 2$. Let $\mathbf{C}$ be a minimal leaf of H. We may assume G is connected, so we have $\mathsf{death}(\mathbf{C}) < \infty$. We may assume G contains no edges e such that $f(e) = f(v)$ for $v \in e$, and thus $U(\mathbf{C}) = \{x\}$ for some $x \in X$. Let $f' : G \to \mathbb{R}$ coincide with f on $G \setminus \{x\}$ and have $f'(x) = \mathsf{death}(\mathbf{C})$. In this case, $H(G, f') = H \setminus \mathbf{C}$. By Lemma 116 and the inductive hypothesis, the algorithm is correct on the input (G, f'). So, by Proposition 71, it is enough to prove that the output of the algorithm on (G, f) is equal to the union of the output on (G, f') and the interval $\mathsf{life}(\mathbf{C})$. Say we run the algorithm on (G, f) using the ordering $[\sigma_1, \ldots, \sigma_p]$. Let $\sigma_i = \{x, y\}$ be the first edge adjacent to x in this ordering. We run the algorithm on (G, f') with the ordering obtained from $[\sigma_1, \ldots, \sigma_p]$ by moving x to the first position such that the ordering satisfies the assumptions of the algorithm. Let (d, v) be the connected component with $y \in d$ after processing σ_{i-1} (this is the same whether running the algorithm on (G, f) or (G, f')). We have $f(\sigma_i) = \mathsf{death}(\mathbf{C})$, and as $\mathbf{C}$ is a minimal leaf, we have $v \leq f(x)$, else H would contain a leaf shorter than $\mathbf{C}$. After processing σ_i on the input (G, f), `conn_comp` and `barcode` are updated as

$$\begin{aligned}
\texttt{conn_comp} &\leftarrow \big(\texttt{conn_comp} \setminus \big\{(\{x\}, f(x)), (d, v)\big\}\big) \cup \big\{(\{x\} \cup d, v)\big\} \\
\texttt{barcode} &\leftarrow \big(\texttt{barcode} \setminus \big\{[f(x), \infty)\big\}\big) \cup \big\{[f(x), \mathsf{death}(\mathbf{C}))\big\}
\end{aligned}$$

After processing σ_i on the input (G, f'), the variable `conn_comp` is updated in the same way, and `barcode` is updated as

$$\texttt{barcode} \leftarrow \big(\texttt{barcode} \setminus \big\{[\mathsf{death}(\mathbf{C}), \infty)\big\}\big) \cup \big\{[\mathsf{death}(\mathbf{C}), \mathsf{death}(\mathbf{C}))\big\}.$$

It follows that the output of the algorithm on (G, f) is equal to the union of the output on (G, f') and the interval $\mathsf{life}(\mathbf{C})$, as desired. ■

Lemma 120 *Let $P, Q : \mathbb{N} \to [0, \infty]$ be prominence diagrams and let $\sigma : \mathbb{N} \to \mathbb{N}$ be a bijection. Then,*

$$\sup_{i \in \mathbb{N}} |P(i) - Q(i)| \leq \sup_{i \in \mathbb{N}} |P(i) - Q(\sigma(i))|.$$

Proof Consider the bijection $\sigma_1 : \mathbb{N} \to \mathbb{N}$ given by

$$\sigma_1(i) = \begin{cases} 0 & \text{if } i = 0 \\ \sigma(0) & \text{if } i = \sigma^{-1}(0) \\ \sigma(i) & \text{otherwise.} \end{cases}$$

In words, the bijection σ_1 coincides with σ, except that it maps 0 to 0 and $\sigma^{-1}(0)$ to $\sigma(0)$. We then proceed inductively by considering a bijection σ_{j+1} which coincides with σ_j except that it maps j to j and $\sigma_j^{-1}(j)$ to $\sigma_j(j)$.

As $j \to \infty$, the bijection σ_j converges pointwise to the identity function $\mathbb{N} \to \mathbb{N}$. Since both $P(i)$ and $Q(i)$ converge to 0 as $i \to \infty$, it follows that $\sup_{i \in \mathbb{N}} |P(i) - Q(\sigma_j(i))|$ converges to $\sup_{i \in \mathbb{N}} |P(i) - Q(i)|$ as $j \to \infty$. Then, the result follows from the following claim.

Claim. For all $j \geq 1 \in \mathbb{N}$, we have

$$\sup_{i \in \mathbb{N}} |P(i) - Q(\sigma_{j+1}(i))| \;\leq\; \sup_{i \in \mathbb{N}} |P(i) - Q(\sigma_j(i))|.$$

Proof of claim. Since σ_j and σ_{j+1} coincide except on j and $\sigma_j^{-1}(j)$, it is sufficient to prove that

$$\begin{aligned} \max\Big(|P(j) - Q(\sigma_{j+1}(j))|, |P(\sigma_j^{-1}(j)) - Q(\sigma_{j+1}(\sigma_j^{-1}(j)))|\Big) \\ \leq \max\Big(|P(j) - Q(\sigma_j(j))|, |P(\sigma_j^{-1}(j)) - Q(j)|\Big) \end{aligned}$$

By definition of σ_{j+1}, the left-hand side of the inequality is equal to

$$\max\Big(|P(j) - Q(j)|, |P(\sigma_j^{-1}(j)) - Q(\sigma_j(j))|\Big).$$

Recall that, if $a \leq a' \in \mathbb{R}$ and $b \leq b' \in \mathbb{R}$ we have $\max(|a-b|, |a'-b'|) \leq \max(|a-b'|, |a'-b|)$. It is thus enough to prove that $P(j) \geq P(\sigma_j^{-1}(j))$ and $Q(j) \geq Q(\sigma_j(j))$. This follows from the fact that we have $\sigma_j(i) = i$ for all $i < j$ and thus $j \leq \sigma_j^{-1}(j)$ and $j \leq \sigma_j(j)$. ■

Proof (of Lemma 78) Let $\mathcal{B}(H) = \{A_0, \ldots, A_k\}$ and $\mathcal{B}(E) = \{B_0, \ldots, B_m\}$, such that $\mathsf{Pr}(H)(i) = \mathsf{length}(A_i)$ for $0 \leq i \leq k$ and $\mathsf{Pr}(E) = \mathsf{length}(B_i)$ for $0 \leq i \leq m$. By Proposition 69, there exists an ε-matching $f : \{0, \ldots, k\} \to \{0, \ldots, m\}$ between $\mathcal{B}(H)$ and $\mathcal{B}(E)$. We can thus extend the matching f to a bijection $f : \mathbb{N} \to \mathbb{N}$ and any such extension has the property that $|\mathsf{Pr}(H)(i) - \mathsf{Pr}(E)(f(i))| \leq 2\varepsilon$. This is because, if $i \in \{0, \ldots, m\}$ is in the domain of f, then $|\mathsf{length}(A_i) - \mathsf{length}(B_{f(i)})| \leq 2\varepsilon$, if $i \in \{0, \ldots, m\}$ is not in the domain of f, then $|\mathsf{length}(A_i)| \leq 2\varepsilon$, and if $j \in \{0, \ldots, m\}$ is not in the codomain of f, then $|\mathsf{length}(B_i)| \leq 2\varepsilon$. Now, the result follows from Lemma 120. ■

Proof (of Lemma 81) By Lemma 78, $d_\infty(\mathsf{Pr}(H_{\geq\tau}), \mathsf{Pr}(E_{\geq\tau})) \leq 2\, d_{\mathrm{CI}}(H_{\geq\tau}, E_{\geq\tau})$ for any $\tau > 0$. By Proposition 121, $d_{\mathrm{CI}}(H_{\geq\tau}, E_{\geq\tau}) \leq d_{\mathrm{CI}}(H, E)$. So, $d_\infty(\mathsf{Pr}(H_{\geq\tau}), \mathsf{Pr}(E_{\geq\tau})) \leq 2\, d_{\mathrm{CI}}(H, E)$. By definition, as $\tau \to 0$, we have $d_\infty(\mathsf{Pr}(H_{\geq\tau}), \mathsf{Pr}(H)) \to 0$. So, letting $\tau \to 0$ and using the triangle inequality, we have $d_\infty(\mathsf{Pr}(H), \mathsf{Pr}(E)) \leq 2\, d_{\mathrm{CI}}(H, E)$. ■

A.5 Details from Section 6

Proposition 121 *Let H and E be $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively, and let $\tau \geq 0$. The hierarchical clusterings $H_{\geq\tau}$ and H are τ-interleaved and, if H and E are ε-interleaved with respect to a correspondence $R \subseteq X \times Y$, then $H_{\geq\tau}$ and $E_{\geq\tau}$ are ε-interleaved with respect to R.*

Proof The fact that $H_{\geq\tau}$ and H are τ-interleaved follows immediately from the definitions. We show that $H_{\geq\tau}$ and $E_{\geq\tau}$ are ε-interleaved with respect to R. Let $r \in \mathbb{R}$ and let $C \in H_{\geq\tau}(r)$. As H and E are ε-interleaved with respect to R, there is $D \in E(r + \varepsilon)$ such that $\pi_X^{-1}(C) \subseteq \pi_Y^{-1}(D)$. As $C \in H_{\geq\tau}(r)$, there is $C' \in H(r - \tau)$ with $C' \subseteq C$. Again by the interleaving, there is $D' \in E(r - \tau + \varepsilon)$ such that $\pi_X^{-1}(C') \subseteq \pi_Y^{-1}(D')$. It follows that $D' \subseteq D$, and thus $D \in E_{\geq\tau}(r + \varepsilon)$, as desired. ■

Notation 122 *Let H and E be $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively. Let $\varepsilon \geq 0$, and let $R \subseteq X \times Y$ be a correspondence such that H and E are ε-interleaved with respect to R. For $r \in \mathbb{R}$, we write $R_X : H(r) \to E(r + \varepsilon)$ for the function such that $\pi_X^{-1}(A) \subseteq \pi_Y^{-1}(R_X(A))$ for all $A \in H(r)$.*

An **interval** of a poset P (Definition 2) consists of a subset $I \subseteq P$ such that whenever we have $x \preceq y \preceq z \in P$ with $x, z \in I$, we also have $y \in I$. A subset T of a poset P is **totally-ordered** if, for all $x, y \in T$, we have $x \preceq y$ or $y \preceq x$. Let $\mathsf{TOI}(P)$ denote the set of totally-ordered intervals of P. Let H and E be $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively, and assume there is $\varepsilon \geq 0$ such that H and E are ε-interleaved with respect to a correspondence $R \subseteq X \times Y$. Define a function $i_X : \mathsf{PC}(H) \to \mathsf{TOI}(\mathsf{PC}(E))$ by mapping a persistent cluster $\mathbf{C}$ to the totally-ordered interval $\{[R_X(\mathbf{C}(r))]\}_{r \in \mathsf{life}(\mathbf{C})}$. If H and E are finite, then we get an order-preserving function $m_X : \mathsf{PC}(H) \to \mathsf{PC}(E)$ by mapping $\mathbf{C}$ to $\min(i_X(\mathbf{C}))$. Note that this depends on ε.

Lemma 123 *Let H and E be finite $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively, and assume there is $\varepsilon \geq 0$ such that H and E are ε-interleaved with respect to a correspondence $R \subseteq X \times Y$. Let $\mathbf{C} \in \mathsf{PC}(H)$.*

1. *We have $\mathsf{birth}(m_X(\mathbf{C})) \leq \mathsf{birth}(\mathbf{C}) + \varepsilon$.*
2. *Assume all the leaves of H and E have length strictly greater than 2ε. If $\mathbf{C} \in \mathsf{leaves}(H)$ and $\mathbf{D} \in \mathsf{leaves}(E)$, then $\mathbf{D} \leq m_X(\mathbf{C})$ if and only if $\mathbf{C} \leq m_Y(\mathbf{D})$.*

Proof For part (1), we can choose $r > \mathsf{birth}(\mathbf{C})$ that is arbitrarily close to $\mathsf{birth}(\mathbf{C})$ such that $m_X(\mathbf{C}) = [R_X(\mathbf{C}(r))]$, and thus $r + \varepsilon \in \mathsf{life}(m_X(\mathbf{C}))$.

For part (2), say we have $\mathbf{C} \in \mathsf{leaves}(H)$ and $\mathbf{D} \in \mathsf{leaves}(E)$ with $\mathbf{D} \leq m_X(\mathbf{C})$. We show that $\mathbf{C} \leq m_Y(\mathbf{D})$. Choose $r_0 \in \mathsf{life}(\mathbf{C})$ such that $r_0 + 2\varepsilon \in \mathsf{life}(\mathbf{C})$ and such that $m_X(\mathbf{C}) = [R_X(\mathbf{C}(r_0))]$. As $\mathbf{D} \leq m_X(\mathbf{C})$, we can choose $r_1 \leq r_0 + \varepsilon$ with $r_1 \in \mathsf{life}(\mathbf{D})$ and such that $m_Y(\mathbf{D}) = [R_Y(\mathbf{D}(r_1))]$. We have $\mathbf{D}(r_1) \subseteq R_X(\mathbf{C}(r_0))$; choose $y \in \mathbf{D}(r_1)$. If $(x, y) \in R$, then as $y \in R_X(\mathbf{C}(r_0))$, we have $x \in R_Y(R_X(\mathbf{C}(r_0)))$. Meanwhile, as $r_0 + 2\varepsilon \in \mathsf{life}(\mathbf{C})$, we have $\mathbf{C} = [R_Y(R_X(\mathbf{C}(r_0)))]$, and so $x \in U(\mathbf{C})$. Now, as $y \in \mathbf{D}(r_1)$, $x \in R_Y(\mathbf{D}(r_1))$, and so $x \in U(m_Y(\mathbf{D}))$. We have therefore shown that $U(\mathbf{C}) \cap U(m_Y(\mathbf{D})) \neq \emptyset$, and thus $\mathbf{C}$ and $m_Y(\mathbf{D})$ are comparable in the poset $\mathsf{PC}(H)$, by Lemma 110. As $\mathbf{C}$ is a leaf, we must have $\mathbf{C} \leq m_Y(\mathbf{D})$. By a symmetric argument, $\mathbf{D} \leq m_X(\mathbf{C})$ if and only if $\mathbf{C} \leq m_Y(\mathbf{D})$. ■

Lemma 124 *Let H and E be finite $\mathbb{R}$-hierarchical clusterings of sets X and Y respectively, and assume there is $\varepsilon \geq 0$ such that H and E are ε-interleaved with respect to a correspondence $R \subseteq X \times Y$. If the leaves of H and E all have length strictly greater than 2ε, then m_X restricts to a bijection $\mathsf{leaves}(H) \to \mathsf{leaves}(E)$ such that $\mathbf{C}$ and $m_X(\mathbf{C})$ are ε-interleaved with respect to R for every $\mathbf{C} \in \mathsf{leaves}(H)$.*

Proof Let $\mathbf{C} \in \mathsf{leaves}(H)$. We start by proving that $m_X(\mathbf{C})$ is a leaf; a symmetric argument shows that m_Y sends leaves to leaves. Let $\mathbf{D} \in \mathsf{leaves}(E)$ with $\mathbf{D} \leq m_X(\mathbf{C})$; we have $\mathbf{C} \leq m_Y(\mathbf{D})$, by Lemma 123 (2). Towards a contradiction, say $\mathbf{D} \neq m_X(\mathbf{C})$. As $\mathsf{length}(\mathbf{D}) > 2\varepsilon$, we have $\mathsf{birth}(\mathbf{D}) + 2\varepsilon < \mathsf{birth}(m_X(\mathbf{C})) \leq \mathsf{birth}(\mathbf{C}) + \varepsilon \leq \mathsf{birth}(m_Y(\mathbf{D})) + \varepsilon$. So, we have $\mathsf{birth}(\mathbf{D}) + \varepsilon < \mathsf{birth}(m_Y(\mathbf{D}))$, which contradicts Lemma 123 (1). So, $\mathbf{D} = m_X(\mathbf{C})$, and thus $m_X(\mathbf{C})$ is a leaf. It follows that we have $m_Y(m_X(\mathbf{C})) = \mathbf{C}$ for any leaf $\mathbf{C}$, since $\mathbf{C} \leq m_Y(m_X(\mathbf{C}))$ and $\mathbf{C}$ and $m_Y(m_X(\mathbf{C}))$ are both leaves. Together with a symmetric argument, this shows that m_X and m_Y restrict to inverse bijections on leaves. The proof that $\mathbf{C}$ and $m_X(\mathbf{C})$ are ε-interleaved is straightforward. ■

Next, we need a lemma that describes the barcode of $H_{\geq\tau}$ in terms of the barcode of H. Let H be a pointwise finite I-hierarchical clustering with $I \subseteq \mathbb{R}$ an interval, and let $\mathcal{B}(H) = \{B_j\}_{j \in J}$. For $\tau > 0$, let $\mathcal{B}(H)_\tau = \{\tilde{B}_j\}_{j \in \tilde{J}}$, where for any $j \in J$, $\tilde{B}_j \subset B_j$ is the sub-interval $\tilde{B}_j = \{x \in B_j : x - \tau \in B_j\}$, and $\tilde{J} \subseteq J$ consists of j such that $\tilde{B}_j \neq \emptyset$.

Lemma 125 *Let H be a pointwise finite I-hierarchical clustering with $I \subseteq \mathbb{R}$ an interval. For any $\tau > 0$, $\mathcal{B}(H_{\geq\tau}) = \mathcal{B}(H)_\tau$.*

Proof We need to check that, for all $r \leq r'$, we have $\mathrm{rk}(H_{\geq\tau}) = |\{j \in \tilde{J} : r, r' \in \tilde{B}_j\}|$. As $H_{\geq\tau}(r) = \mathrm{Im}\, H(r - \tau \leq r)$, we have $\mathrm{Im}\, H_{\geq\tau}(r \leq r') = \mathrm{Im}\, H(r - \tau \leq r')$. And by definition of the barcode, we have $|\mathrm{Im}\, H(r - \tau \leq r')| = |\{j \in J : r - \tau, r' \in B_j\}|$. Now, for any $j \in J$, we have $r - \tau, r' \in B_j$ if and only if we have $r, r' \in \tilde{B}_j$. So, $|\{j \in J : r - \tau, r' \in B_j\}| = |\{j \in \tilde{J} : r, r' \in \tilde{B}_j\}|$. This finishes the proof. ■

Lemma 126 *Let H be an essentially finite I-hierarchical clustering with $I \subseteq \mathbb{R}$ an interval. Let $n \geq 1$ and assume $\mathsf{gap}_n(H)$ is non-empty. If $\tau \in \mathsf{gap}_n(H)$, then for every $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})$, we have $\mathsf{length}(\mathbf{C}) \geq \mathsf{Pr}(H)(n-1) - \tau$.*

Proof First, say H is finite. Then $\min_{\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})} \mathsf{length}(\mathbf{C}) = \min_{B \in \mathcal{B}(H_{\geq\tau})} \mathsf{length}(B)$ by Corollary 118. As H is finite, it is pointwise finite, so we can apply Lemma 125. Let $(\ell_0, \ldots, \ell_k)$ be the lengths of the intervals in $\mathcal{B}(H) = \{B_j\}_{j \in J}$ ordered from largest to smallest, as in Definition 77. We have $\mathcal{B}(H_{\geq\tau}) = \{\tilde{B}_j\}_{j \in \tilde{J}}$, and thus the lengths of the intervals in $\mathcal{B}(H_{\geq\tau})$, ordered from largest to smallest, are $(\tilde{\ell}_0, \ldots, \tilde{\ell}_{n-1})$, where $\tilde{\ell}_p = \ell_p - \tau$. Thus, $\min_{B \in \mathcal{B}(H_{\geq\tau})} \mathsf{length}(B) = \tilde{\ell}_{n-1} = \ell_{n-1} - \tau = \mathsf{Pr}(H)(n-1) - \tau$. Now we consider the case where H is essentially finite. Let $0 < \sigma < \tau$. As $\sigma \to 0$, we have $\mathsf{Pr}(H_{\geq\sigma})(n-1) \to \mathsf{Pr}(H)(n-1)$ and $\mathsf{Pr}(H_{\geq\sigma})(n) \to \mathsf{Pr}(H)(n)$. So, if σ is small enough, $\mathsf{gap}_n(H_{\geq\sigma})$ is non-empty and $\tau \in \mathsf{gap}_n(H_{\geq\sigma})$. We have $H_{\geq\tau} = (H_{\geq\sigma})_{\geq\tau-\sigma}$. Applying the finite case, we have, for every $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})$, $\mathsf{length}(\mathbf{C}) \geq \mathsf{Pr}(H_{\geq\sigma})(n-1) - (\tau - \sigma)$. As $\sigma \to 0$, we have $\mathsf{Pr}(H_{\geq\sigma})(n-1) - (\tau - \sigma) \to \mathsf{Pr}(H)(n-1) - \tau$, which finishes the proof. ■

Proof (of Proposition 83) Without loss of generality, $\tau < \tau'$. Let $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})$. By Lemma 126, $\mathsf{length}(\mathbf{C}) \geq \mathsf{Pr}(H)(n-1) - \tau$, and thus $\mathsf{length}(\mathbf{C}) > \tau' - \tau$. So, there is $r \in \mathbb{R}$ such that, with $C = \mathbf{C}(r)$, $C' := H(r < r + (\tau' - \tau))(C) \in \mathbf{C}$. By construction, $C' \in H_{\geq\tau'}(r + (\tau' - \tau))$. It is easy to check that $[C'] \in \mathsf{leaves}(H_{\geq\tau'})$. Define m by setting $m(\mathbf{C}) = [C']$. We show m is injective. Let $\mathbf{C}, \mathbf{D} \in \mathsf{leaves}(H_{\geq\tau})$, and say $[C'] = [D']$, using the notation from above. Without loss of generality, $C' \subseteq D'$. Then $U(\mathbf{C}) \cap U(\mathbf{D}) \neq \emptyset$, so by Lemma 110, $\mathbf{C} \leq \mathbf{D}$ or $\mathbf{D} \leq \mathbf{C}$. As $\mathbf{C}$ and $\mathbf{D}$ are leaves, it follows that $\mathbf{C} = \mathbf{D}$.

We show m is a bijection by showing $|\mathsf{leaves}(H_{\geq\tau})| = |\mathsf{leaves}(H_{\geq\tau'})|$. By Proposition 71, for any finite $\mathbb{R}$-hierarchical clustering E, $|\mathsf{leaves}(E)| = |\mathcal{B}(E)|$. By an argument like the last step of the proof of Lemma 126, we have $|\mathcal{B}(H_{\geq\tau})| = |\mathcal{B}(H_{\geq\tau'})|$. Thus, $|\mathsf{leaves}(H_{\geq\tau})| = |\mathsf{leaves}(H_{\geq\tau'})|$ and therefore m is a bijection. It is easy to check that for any $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau})$, $U(\mathbf{C}) = U(m(\mathbf{C}))$. ■

Proof (of Theorem 84) By definition, $\mathsf{PF}(H, n) = \mathsf{leaves}(H_{\geq\tau_H})$, where $\tau_H = (\mathsf{Pr}(H)(n-1) + \mathsf{Pr}(H)(n))/2$, and $\mathsf{PF}(E, n) = \mathsf{leaves}(E_{\geq\tau_E})$, where $\tau_E = (\mathsf{Pr}(E)(n-1) + \mathsf{Pr}(E)(n))/2$. By Proposition 121, $H_{\geq\tau_H}$ and $E_{\geq\tau_H}$ are ε-interleaved with respect to R, and $E_{\geq\tau_H}$ and $E_{\geq\tau_E}$ are $|\tau_H - \tau_E|$-interleaved. So, $H_{\geq\tau_H}$ and $E_{\geq\tau_E}$ are $(\varepsilon + |\tau_H - \tau_E|)$-interleaved with respect to R. By Lemma 81, $d_\infty(\mathsf{Pr}(H), \mathsf{Pr}(E)) \leq 2\, d_{\mathrm{CI}}(H, E) \leq 2\varepsilon$. So, $|\tau_H - \tau_E| \leq 2\varepsilon$, and thus $H_{\geq\tau_H}$ and $E_{\geq\tau_E}$ are 3ε-interleaved with respect to R.

For any $\mathbf{C} \in \mathsf{leaves}(H_{\geq\tau_H})$, Lemma 126 implies that $\mathsf{length}(\mathbf{C}) \geq \mathsf{Pr}(H)(n-1) - \tau_H = (\mathsf{Pr}(H)(n-1) - \mathsf{Pr}(H)(n))/2 > 8\varepsilon$. Similarly, for any $\mathbf{D} \in \mathsf{leaves}(E_{\geq\tau_E})$, we have $\mathsf{length}(\mathbf{D}) \geq$

$\mathsf{Pr}(E)(n-1)-\tau_E = (\mathsf{Pr}(E)(n-1)-\mathsf{Pr}(E)(n))/2 \geq ((\mathsf{Pr}(H)(n-1)-\mathsf{Pr}(H)(n))/2)-2\varepsilon > 6\varepsilon$. As $H_{\geq\tau_H}$ and $E_{\geq\tau_E}$ are finite hierarchical clusterings that are 3ε-interleaved with respect to R, and the leaves of $H_{\geq\tau_H}$ and $E_{\geq\tau_E}$ all have length strictly greater than 6ε, Lemma 124 implies that there is a bijection $m : \mathsf{PF}(H,n) \to \mathsf{PF}(E,n)$ such that for all $\mathbf{C} \in \mathsf{PF}(H,n)$, $\mathbf{C}$ and $m(\mathbf{C})$ are 3ε-interleaved with respect to R. ■

Remark 127 *We describe the relationship between* ExhaustivePF *(Algorithm 2) and ToMATo (Chazal et al., 2013, Algorithm 1). In line 10 of* ExhaustivePF *we use* $\leq$ *rather than* $<$ *in order to follow the behavior of the persistence-based flattening. In this remark, we consider the version of* ExhaustivePF *that uses* $<$ *in line 10. Say given the input* $(G, \tilde{f}, \tau)$ *to ToMATo. Extend the filtering function to edges, by setting* $\tilde{f}(\{x,y\}) = \min\{\tilde{f}(x), \tilde{f}(y)\}$. *Reverse the filtration, by setting* $f(\sigma) = -\tilde{f}(\sigma)$ *for* $\sigma \in G$. *Order the simplices of* G *as follows. Begin with any ordering of the vertices* $x_1, \ldots, x_q$ *such that* $f(x_i) \leq f(x_{i+1})$. *For each* i, *insert directly after* x_i *those edges* $\{x_j, x_i\}$ *with* $j < i$, *ordered by* j. *Now, the output of* ExhaustivePF *on this input agrees with the output of ToMATo on* $(G, \tilde{f}, \tau)$, *with one exception: ToMATo excludes clusters* C *such that* $\max_{x\in C} \tilde{f}(x) < \tau$.

A.6 Details from Section 7

k	Memory (GB)	Time (s)	Number of clusters
20	0.46	65	6524
40	0.82	58	2618
80	1.54	64	1057
160	2.99	68	473
320	5.88	79	230

Table 2: Evaluation of the HDBSCAN implementation of McInnes et al. (2017) on the Rideshare data set. The columns are the algorithm parameter $k = \mathsf{min\_samples}$, the peak memory usage in GB, the runtime in seconds, and the number of clusters. The algorithm parameter $\mathsf{min\_cluster\_size}$ is set equal to $\mathsf{min\_samples}$ (the default). The implementation includes several algorithms for computing HDBSCAN. We use the Dual-Tree Borůvka algorithm with kd-trees (the default choice for low-dimensional Euclidean data, as in this case), which is the only choice that gives reasonable performance on this data set. Memory usage scales linearly with k because the algorithm stores the k nearest neighbors of each data point.

	1	2	3	noise
South Italy	293			30
Sardinia		97		1
North Italy			118	33

Table 3: A confusion matrix comparing the Persistable clustering (clusters 1–3 and points labeled as noise), and the large area labels of the Olive oil data. The adjusted Rand index is 1.0, and 89% of the data is clustered.

		1	2	3	4	5	6	7	8	noise
South Italy	North Apulia	12								13
	Calabria		7	1						48
	South Apulia			100						106
	Sicily	3								33
Sardinia	Inland Sardinia				51					14
	Coast Sardinia					19				14
North Italy	East Liguria						14	1		35
	West Liguria							29		21
	Umbria								42	9

Table 4: A confusion matrix comparing the Persistable clustering (clusters 1–8 and points labeled as noise), and the region labels of the Olive oil data. The adjusted Rand index is 0.98, and 49% of the data is clustered.

		1	2	3	4	5	6	7	8	noise
South Italy	North Apulia	21	2							2
	Calabria		50	3			1			2
	South Apulia		1	196						9
	Sicily	6	20	7						3
Sardinia	Inland Sardinia				65					
	Coast Sardinia				2	31				
North Italy	East Liguria						41	2		7
	West Liguria							47		3
	Umbria						2		48	1

Table 5: A confusion matrix comparing the Persistable clustering, using the exhaustive persistence-based flattening (clusters 1–8 and points labeled as noise), and the region labels of the Olive oil data. The adjusted Rand index is 0.90, and 95% of the data is clustered.

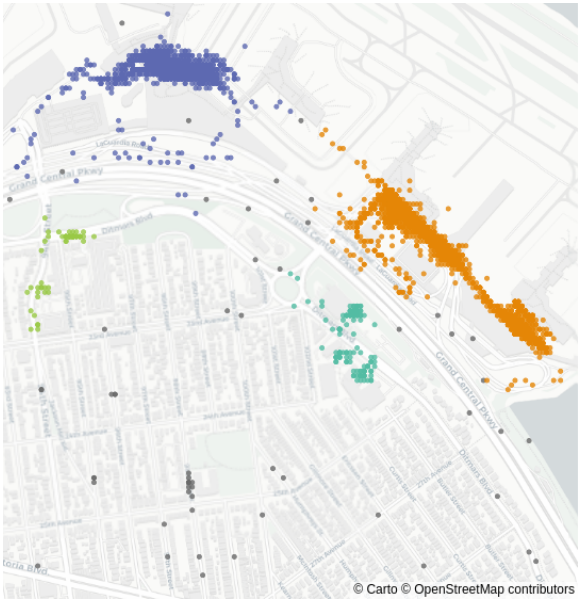

Figure 16: The Persistable clustering of the Rideshare data using the slice in Fig. 12, choosing the gap below the fourth vine. Gray points are unclustered.

References

Hirokazu Anai, Frédéric Chazal, Marc Glisse, Yuichi Ike, Hiroya Inakoshi, Raphaël Tinarrage, and Yuhei Umeda. DTM-Based Filtrations. In Gill Barequet and Yusu Wang, editors, *35th International Symposium on Computational Geometry (SoCG 2019)*, volume 129 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 58:1–58:15, 2019. doi: 10.4230/LIPIcs.SoCG.2019.58.

Bryon Aragam, Chen Dan, Eric P. Xing, and Pradeep Ravikumar. Identifiability of nonparametric mixture models and Bayes optimal clustering. *The Annals of Statistics*, 48 (4):2277 – 2302, 2020. doi: 10.1214/19-AOS1887.

Sivaraman Balakrishnan, Srivatsan Narayanan, Alessandro Rinaldo, Aarti Singh, and Larry Wasserman. Cluster trees on manifolds. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2679–2687. Curran Associates, Inc., 2013.

Ulrich Bauer and Michael Lesnick. Induced matchings and the algebraic stability of persistence barcodes. *Comput. Geom.*, 6:162–191, 2015. doi: 10.20382/jocg.v6i2a9.

Ulrich Bauer, Axel Munk, Hannes Sieling, and Max Wardetzky. Persistence barcodes versus Kolmogorov signatures: Detecting modes of one-dimensional signals. *Foundations of Computational Mathematics*, 17:1–33, 2017. doi: 10.1007/s10208-015-9281-9.

Ulrich Bauer, Magnus Botnan, Steffen Oppermann, and Johan Steen. Cotorsion torsion triples and the representation theory of filtered hierarchical clustering. *Advances in Mathematics*, 369:107171, 2020. doi: 10.1016/j.aim.2020.107171.

Silvia Biasotti, Andrea Cerri, Patrizio Frosini, Daniela Giorgi, and Claudia Landi. Multidimensional size functions for shape comparison. *Journal of Mathematical Imaging and Vision*, 32:161–179, 2008. doi: https://doi.org/10.1007/s10851-008-0096-z.

Gérard Biau, Benoît Cadre, and Bruno Pelletier. A graph-based estimator of the number of clusters. *ESAIM: Probability and Statistics*, 11:272–280, 6 2007. doi: https://doi.org/10.1051/ps:2007019.

Patrick Billingsley. *Convergence of Probability Measures, Second Edition*. Wiley Series in Probability and Statistics. Wiley, 1999. doi: 10.1002/9780470316962.

Andrew J. Blumberg and Michael Lesnick. Stability of 2-parameter persistent homology. *Foundations of Computational Mathematics*, 2022. doi: 10.1007/s10208-022-09576-6.

Omer Bobrowski, Sayan Mukherjee, and Jonathan E. Taylor. Topological consistency via kernel estimation. *Bernoulli*, 23(1):288–328, 2017. doi: 10.3150/15-BEJ744.

Magnus Bakke Botnan, Steffen Oppermann, and Steve Oudot. Signed Barcodes for Multi-Parameter Persistence via Rank Decompositions. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry (SoCG 2022)*, volume 224 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19:1–19:18, 2022. doi: 10.4230/LIPIcs.SoCG.2022.19.

Kevin Buchin, Maike Buchin, Marc van Kreveld, Bettina Speckmann, and Frank Staals. Trajectory grouping structure. *Journal of Computational Geometry*, 6, 2015. doi: https://doi.org/10.20382/jocg.v6i1a3.

Dmitri Burago, Yuri Burago, and Sergei Ivanov. *A Course in Metric Geometry*, volume 33 of *Graduate Studies in Mathematics*. American Mathematical Society, 2001. doi: http://dx.doi.org/10.1090/gsm/033.

Francesca Cagliari and Claudia Landi. Finiteness of rank invariants of multidimensional persistent homology groups. *Appl. Math. Lett.*, 24(4):516–518, 2011. ISSN 0893-9659. doi: 10.1016/j.aml.2010.11.004. URL `https://doi.org/10.1016/j.aml.2010.11.004`.

Francesca Cagliari, Barbara Di Fabio, and Massimo Ferri. One-dimensional reduction of multidimensional persistent homology. *Proc. Amer. Math. Soc.*, 138(8):3003–3017, 2010. ISSN 0002-9939. doi: 10.1090/S0002-9939-10-10312-8.

Chen Cai, Woojin Kim, Facundo Mémoli, and Yusu Wang. Elder-Rule-Staircodes for Augmented Metric Spaces. In Sergio Cabello and Danny Z. Chen, editors, *36th International Symposium on Computational Geometry (SoCG 2020)*, volume 164 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 26:1–26:17, 2020. doi: 10.4230/LIPIcs.SoCG.2020.26.

Ricardo J. G. B. Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining*, volume 7819 of *Lecture Notes in Computer Science*, pages 160–172. Springer, 2013. doi: https://doi.org/10.1007/978-3-642-37456-2_14.

Gunnar Carlsson and Facundo Mémoli. Characterization, stability and convergence of hierarchical clustering methods. *J. Mach. Learn. Res.*, 11:1425–1470, 2010a.

Gunnar Carlsson and Facundo Mémoli. Multiparameter hierarchical clustering methods. In Hermann Locarek-Junge and Claus Weihs, editors, *Classification as a Tool for Research*, pages 63–70. Springer Berlin Heidelberg, 2010b. ISBN 978-3-642-10745-0.

Gunnar Carlsson and Afra Zomorodian. The theory of multidimensional persistence. *Discrete & Computational Geometry*, 42(1):71 – 93, 2009. doi: 10.1007/s00454-009-9176-0.

Gunnar Carlsson, Afra Zomorodian, Anne Collins, and Leonidas Guibas. Persistence barcodes for shapes. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, page 124–135, 2004. doi: 10.1145/1057432.1057449.

Mathieu Carrière and Andrew J. Blumberg. Multiparameter persistence images for topological machine learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Andrea Cerri, Barbara Di Fabio, Massimo Ferri, Patrizio Frosini, and Claudia Landi. Multidimensional persistent homology is stable, 2009. arXiv:0908.0064.

Kamalika Chaudhuri and Sanjoy Dasgupta. Rates of convergence for the cluster tree. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 343–351. Curran Associates, Inc., 2010. URL `http://papers.nips.cc/paper/4068-rates-of-convergence-for-the-cluster-tree.pdf`.

Frédéric Chazal, David Cohen-Steiner, Marc Glisse, Leonidas J. Guibas, and Steve Y. Oudot. Proximity of persistence modules and their diagrams. In *Proceedings of the Twenty-Fifth Annual Symposium on Computational Geometry*, SCG '09, page 237–246, 2009. doi: 10.1145/1542362.1542407.

Frédéric Chazal, Leonidas J. Guibas, Steve Y. Oudot, and Primoz Skraba. Persistence-based clustering in Riemannian manifolds. *J. ACM*, 60(6), 2013. doi: 10.1145/2535927.

Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. *The structure and stability of persistence modules*. SpringerBriefs in Mathematics. Springer, [Cham], 2016. ISBN 978-3-319-42543-6; 978-3-319-42545-0. doi: 10.1007/978-3-319-42545-0.

Simon G. Chiossi. *Essential Mathematics for Undergraduates: A Guided Approach to Algebra, Geometry, Topology and Analysis*. Springer Cham, 2021. doi: https://doi.org/10.1007/978-3-030-87174-1.

David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and vineyards by updating persistence in linear time. In *Symposium on Computational Geometry*, pages 119–126. ACM, New York, 2006. doi: 10.1145/1137856.1137877.

David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete Comput Geom*, 37:103–120, 2007. doi: 10.1007/s00454-006-1276-5.

David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Yuriy Mileyko. Lipschitz functions have L_p-stable persistence. *Foundations of Computational Mathematics*, 10:127–139, 2010. doi: 10.1007/s10208-010-9060-6.

René Corbet, Ulderico Fugacci, Michael Kerber, Claudia Landi, and Bei Wang. A kernel for multi-parameter persistent homology. *Computers & Graphics: X*, 2:100005, 2019. doi: https://doi.org/10.1016/j.cagx.2019.100005.

William Crawley-Boevey. Decomposition of pointwise finite-dimensional persistence modules. *J. Algebra Appl.*, 14(5):1550066, 8, 2015. doi: 10.1142/S0219498815500668.

Ákos Császár. *General Topology*. Adam Hilger Ltd, Bristol, 1978. doi: https://doi.org/10.1017/S0013091500027905.

Antonio Cuevas and Alberto Rodríguez-Casal. On boundary estimation. *Advances in Applied Probability*, 36(2):340–354, 2004. doi: 10.1239/aap/1086957575.

Antonio Cuevas, Manuel Febrero, and Ricardo Fraiman. Estimating the number of clusters. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 28(2):367–382, 2000. doi: https://doi.org/10.2307/3315985.

Justin Curry. The fiber of the persistence map for functions on the interval. *J. Appl. Comput. Topol.*, 2(3-4):301–321, 2018. ISSN 2367-1726. doi: 10.1007/s41468-019-00024-z.

Michele d'Amico, Patrizio Frosini, and Claudia Landi. Optimal matching between reduced size functions. *DISMI, Univ. di Modena e Reggio Emilia, Italy, Technical report*, 35, 2003.

Tamal Krishna Dey and Yusu Wang. *Computational Topology for Data Analysis*. Cambridge University Press, 2022. doi: 10.1017/9781009099950.

Richard M. Dudley. The Speed of Mean Glivenko-Cantelli Convergence. *The Annals of Mathematical Statistics*, 40(1):40 – 50, 1969. doi: 10.1214/aoms/1177697802.

Richard M. Dudley. *Real analysis and probability*, volume 74 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2002. ISBN 0-521-00754-2. doi: 10.1017/CBO9780511755347. Revised reprint of the 1989 original.

Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Society, 2010. doi: 10.1007/978-3-540-33259-6_7.

Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511 – 533, 2002. doi: 10.1007/s00454-002-2885-2.

Justin Eldridge, Mikhail Belkin, and Yusu Wang. Beyond Hartigan consistency: Merge distortion metric for hierarchical clustering. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 588–606, Paris, France, 03–06 Jul 2015. PMLR. URL `http://proceedings.mlr.press/v40/Eldridge15.html`.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231. AAAI Press, 1996.

Brittany Terese Fasy, Fabrizio Lecci, Alessandro Rinaldo, Larry Wasserman, Sivaraman Balakrishnan, and Aarti Singh. Confidence sets for persistence diagrams. *The Annals of Statistics*, 42(6):2301 – 2339, 2014. doi: 10.1214/14-AOS1252.

FiveThirtyEight. Uber TLC FOIL response data, 2015. data retrieved from `https://github.com/fivethirtyeight/uber-tlc-foil-response`.

Gerald B. Folland. *Real Analysis: Modern Techniques and Their Applications*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2013. ISBN 9781118626399.

Michele Forina, C. Armanino, Sergio Lanteri, and E. Tiscornia. Classification of olive oils from their fatty acid composition. In H. Martens and H. Russwurm Jr, editors, *Food research and data analysis : proceedings from the IUFoST Symposium*, pages 189–214. ill., maps, Oslo, Norway, 1983.

Patrizio Frosini. A distance for similarity classes of submanifolds of a Euclidean space. *Bull. Austral. Math. Soc.*, 42(3):407–416, 1990. ISSN 0004-9727. doi: 10.1017/S0004972700028574. URL https://doi.org/10.1017/S0004972700028574.

Patrizio Frosini. Discrete computation of size functions. *J. Combin. Inform. System Sci.*, 17(3-4):232–250, 1992a. ISSN 0250-9628.

Patrizio Frosini. Measuring shapes by size functions. In *Intelligent Robots and Computer Vision X: Algorithms and Techniques*, volume 1607, pages 122–133. SPIE, 1992b.

Robert Ghrist. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, 45:61–75, 2008. doi: 10.1090/S0273-0979-07-01191-3.

Alison L. Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International Statistical Review / Revue Internationale de Statistique*, 70(3):419–435, 2002. URL http://www.jstor.org/stable/1403865.

Mikhail Gromov. *Metric Structures for Riemannian and Non-Riemannian Spaces*. Modern Birkhäuser Classics. Birkhäuser Boston, MA, 2007. doi: 10.1007/978-0-8176-4583-0.

John A. Hartigan. *Clustering algorithms*. John Wiley & Sons, New York-London-Sydney, 1975. Wiley Series in Probability and Mathematical Statistics.

John A. Hartigan. Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association*, 76(374):388–394, 1981. doi: 10.1080/01621459.1981.10477658.

John F. Jardine. Persistent homotopy theory, 2020a. arXiv:2002.10013.

John F. Jardine. Stable components and layers. *Canadian Mathematical Bulletin*, 63(3):562–576, 2020b. doi: 10.4153/S000843951900064X.

Jisu Kim, Yen-Chi Chen, Sivaraman Balakrishnan, Alessandro Rinaldo, and Larry Wasserman. Statistical inference for cluster trees. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1839–1847. Curran Associates, Inc., 2016.

Woojin Kim and Facundo Mémoli. Formigrams: Clustering summaries of dynamic data. In Stephane Durocher and Shahin Kamali, editors, *Proceedings of the 30th Canadian Conference on Computational Geometry, CCCG 2018, August 8-10, 2018, University of Manitoba, Winnipeg, Manitoba, Canada*, pages 180–188, 2018.

Claudia Landi. The rank invariant stability via interleavings. In Erin Wolf Chambers, Brittany Terese Fasy, and Lori Ziegelmeier, editors, *Research in Computational Topology*, pages 1–10, Cham, 2018. Springer International Publishing. doi: 10.1007/978-3-319-89593-2_1.

Michael Lesnick and Matthew Wright. Interactive visualization of 2-D persistence modules. *arXiv:1512.00180*, 2015.

Leland McInnes and John Healy. Accelerated hierarchical density based clustering. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, volume 00, pages 33–42, Nov. 2018. doi: 10.1109/ICDMW.2017.12.

Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017. doi: 10.21105/joss.00205.

Marina Meilă. Comparing clusterings—an information based distance. *J. Multivariate Anal.*, 98(5):873–895, 2007. ISSN 0047-259X. doi: 10.1016/j.jmva.2006.11.013.

Grégory Miermont. Tessellations of random maps of arbitrary genus. *Ann. Sci. Éc. Norm. Supér. (4)*, 42(5):725–781, 2009. ISSN 0012-9593. doi: 10.24033/asens.2108.

Kalyanapuram Rangachari Parthasarathy. *Probability Measures on Metric Spaces*. AMS Chelsea Publishing Series. Acad. Press, 1972. ISBN 9780821869420.

Alessandro Rinaldo and Larry Wasserman. Generalized density clustering. *Ann. Statist.*, 38(5):2678–2722, 10 2010. doi: 10.1214/10-AOS797.

Alessandro Rinaldo, Aarti Singh, Rebecca Nugent, and Larry Wasserman. Stability of density-based clustering. *J. Mach. Learn. Res.*, 13(1):905–948, April 2012.

Luis Scoccola. *Locally Persistent Categories And Metric Properties Of Interleaving Distances*. PhD thesis, University of Western Ontario, 2020.

Luis Scoccola and Alexander Rolle. Persistable: persistent and stable clustering. *Journal of Open Source Software*, 8(83):5022, 2023. doi: 10.21105/joss.05022.

Dan Shiebler. Flattening multiparameter hierarchical clustering functors, 2021. arXiv:2104.14734.

Robin Sibson. SLINK: An optimally efficient algorithm for the single-link cluster method. *Comput. J.*, 16:30–34, 1973. doi: https://doi.org/10.1093/comjnl/16.1.30.

Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, 1986. doi: https://doi.org/10.1201/9781315140919. Monographs on Statistics and Applied Probability.

Werner Stuetzle and Rebecca Nugent. A generalized single linkage method for estimating the cluster tree of a density. *Journal of Computational and Graphical Statistics*, 19(2): 397–418, 2010. doi: 10.1198/jcgs.2009.07049.

Robert Endre Tarjan. *Data Structures and Network Algorithms*. Society for Industrial and Applied Mathematics, 1983. doi: 10.1137/1.9781611970265.

The RIVET Developers. RIVET. 1.1.0, 2020. URL `https://github.com/rivetTDA/rivet/`.

Cédric Villani. *Optimal Transport: Old and New*. Springer Berlin, Heidelberg, 2009. doi: 10.1007/978-3-540-71050-9. Grundlehren der mathematischen Wissenschaften.

Oliver Vipond. Multiparameter persistence landscapes. *J. Mach. Learn. Res.*, 21(61):1–38, 2020.

Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249 – 274, 2005. doi: 10.1007/s00454-004-1146-y.