Clustering Consistently

Dissertation

Presented in Partial Fulfillment of the Requirements for the Degree
Doctor of Philosophy
in the Graduate School of The Ohio State University

By
Justin Eldridge, M.S.

Graduate Program in Computer Science and Engineering
The Ohio State University
2017

Dissertation Committee:
Mikhail Belkin, Advisor
Yusu Wang, Advisor
Facundo Mémoli
Vincent Vu

© Justin Eldridge, 2017

Abstract

Clustering is the task of organizing data into natural groups, or *clusters*. A central goal in developing a theory of clustering is the derivation of correctness guarantees which ensure that clustering methods produce the right results. In this dissertation, we analyze the setting in which the data are sampled from some underlying probability distribution. In this case, an algorithm is "correct" (or *consistent*) if, given larger and larger data sets, its output converges in some sense to the ideal cluster structure of the distribution.

In the first part, we study the setting in which data are drawn from a probability density supported on a subset of a Euclidean space. The natural cluster structure of the density is captured by the so-called *high density cluster tree*, which is due to Hartigan (1981). Hartigan introduced a notion of convergence to the density cluster tree, and recent work by Chaudhuri and Dasgupta (2010) and Kpotufe and Luxburg (2011) has contructed algorithms which are consistent in this sense.

We will show that Hartigan's notion of consistency is in fact not strong enough to ensure that an algorithm recovers the density cluster tree as we would intuitively expect. We identify the precise deficiency which allows this, and introduce a new, stronger notion of convergence which we call *consistency in merge distortion*. Consistency in merge distortion implies Hartigan's consistency, and we prove that the algorithm of Chaudhuri and Dasgupta (2010) satisfies our new notion.

In the sequel, we consider the clustering of graphs sampled from a very general, non-parametric random graph model called a *graphon*. Unlike in the density setting, clustering in the graphon model is not well-studied. We therefore rigorously analyze the cluster structure of a graphon and formally define the *graphon cluster tree*. We adapt our notion of consistency in merge distortion to the graphon setting and identify efficient, consistent algorithms.

Acknowledgements

I am lucky and grateful to have had a pair of excellent advisors in Mikhail Belkin and Yusu Wang. I could not have asked for better mentors. Thank you for reminding me to keep things as simple as possible, but no simpler.

I would like to thank my committee members, Facundo Mémoli and Vincent Vu, for their very helpful comments on the work herein.

To my friends: Thank you for making Ohio State my home for 10 years.

To my brother: Thank you for putting up with me when I was consumed with work.

To my family, and especially my parents: Thank you for fostering in me the curiosity, patience, and determination necessary to complete a Ph.D. It is because of your hard work and support that I have been free to sit and think about such frivolities as clustering.

Vita

2017	Presidential Fellow The Ohio State University
2015	M.S., Computer Science The Ohio State University
2011	B.S., Physics B.S., Applied Mathematics The Ohio State University

Publications

Eldridge, Justin, Mikhail Belkin, and Yusu Wang. 2016. "Graphons, mergeons, and so on!" *Advances in Neural Information Processing Systems*.

Eldridge, Justin, Mikhail Belkin, and Yusu Wang. 2015. "Beyond Hartigan Consistency: Merge Distortion Metric for Hierarchical Clustering." In *Proceedings of The 28th Conference on Learning Theory*, 588–606.

Eldridge, Justin, Alison E. Lane, Mikhail Belkin, and Simon Dennis. 2014. "Robust Features for the Automatic Identification of Autism Spectrum Disorder in Children." *Journal of Neurodevelopmental Disorders* 6 (1): 12.

Fields of Study

Major Field: Computer Science and Engineering

Machine learning, artificial intelligence.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction: clustering formalized

The world around us is increasingly data-driven. Scientific hypotheses, medical diagnoses, business decisions, and engineering designs are made by gathering and analyzing a wealth of data in search of meaningful and predictive patterns. As such, *machine learning* algorithms – methods capable of automatically identifying trends in data – have been the subject of intense recent study. This dissertation concerns fundamental questions about the capabilities and limitations of such learning algorithms.

In particular, this work studies algorithms which recover *cluster structure*; i.e., methods which find natural groups or *clusters* in data. Cluster structure is frequently evident in real-world information. The brain of *C. elegans*, the power grid of the western U.S., and the collaboration network of film actors each have interesting and interpretable group structure (Watts and Strogatz, 1998). Moreover, finding the clusters in data is crucial in many applications. Preventing the spread of infectious diseases in urban environments is aided by clustering the population according to social interactions (Eubank et al., 2004). Retailers make product recommendations by grouping customers according to the similarity of their previous purchases (Ungar and Foster, 1998). And important advances in the study and diagnosis of cancer have been made through clustering microRNA samples (Lu et al., 2005).

It is striking that Nature should so often give rise to cluster structure, and it is fascinating that such structure should be useful for understanding and predicting Nature itself. Given the effectiveness of cluster analysis, we are motivated to ask the fundamental questions: What sort of cluster structure can feasibly be recovered from data? How do we interpret the clusters returned by a clustering method? What does the "correct" clustering look like,

and does an algorithm exist which produces it? It is perhaps surprising that answers are often scarce and significantly limited in scope. Without understanding these questions, the usage of clustering methods in practice can be distressingly *ad hoc*.

The aim of this dissertation is to improve the theoretical footing of clustering by proving strong correctness results for algorithms. In order to do so, we first seek a more formal definition of clustering. In their seminal book, Jain and Dubes (1988) define *clustering* as the "process of classifying objects into subsets that have meaning in the context of a particular problem." But more precisely, what sort of mathematical object is a clustering, and how do we rigorously define the goal of clustering in any particular application?

1.1 Flat vs. hierarchical

As a mathematical object, a clustering of a finite set of objects $\mathcal{X}$ is a collection of subsets of $\mathcal{X}$. It is often the case that this clustering is *flat*:

Definition 1.1. A *flat clustering* of a finite set $\mathcal{X}$ is a partition of $\mathcal{X}$. That is, it is a collection $\mathcal{C}$ of non-empty subsets of $\mathcal{X}$ such that $\bigcup \mathcal{C} = \mathcal{X}$ and any two distinct elements $C, C' \in \mathcal{C}$ are disjoint. An element $C \in \mathcal{C}$ is called a *cluster*.

Alternatively, we may consider clusterings whose clusters are nested, thereby capturing group structure at several scales simultaneously. We say that such clusterings are *hierarchical*:

Definition 1.2. A *hierarchical clustering* (or *cluster tree*) of a finite set $\mathcal{X}$ is a collection $\mathcal{C}$ of non-empty subsets of $\mathcal{X}$ such that $\mathcal{X} \in \mathcal{C}$ and for any distinct $C, C' \in \mathcal{C}$, either $C \subset C'$, $C' \subset C$, or $C \cap C' = \emptyset$. An element $C \in \mathcal{C}$ is called a *cluster*.

We note that our definition of a hierarchical clustering differs from some sources in that it does not require each object $x \in \mathcal{X}$ to appear as a singleton cluster $\{x\}$ in $\mathcal{C}$.

There is a natural order on the clusters in a hiearchical clustering which allows us to interpret it as a directed tree. The root of this tree corresponds to the cluster $\mathcal{X}$. There is

a directed edge from cluster C to cluster C' if and only if $C' \subset C$. We will often emphasize this interpretation of a hierarchical clustering $\mathcal{C}$ by calling $\mathcal{C}$ a *cluster tree*.

1.1.1 Single-linkage hierarchical clustering

This dissertation will focus on the analysis of hierarchical clustering methods. Of particular interest will be the *single-linkage* hierarchical clustering, defined as follows. Let $\mathcal{X}$ be a finite collection of objects, and let $\omega : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a symmetric function. We will interpret ω as a measure of either the *similarity* or the *dissimilarity* between objects in $\mathcal{X}$. If ω is a *dissimilarity*, we define the *dissimilarity graph* as follows:

Definition 1.3 (Dissimilarity graph)**.** The *dissimilarity graph* of $(\mathcal{X}, \omega)$ is a function H, defined on the reals, such that for any $\lambda \in \mathbb{R}$, $H(\lambda)$ is the graph on $\mathcal{X}$ in which the edge (x, y) occurs if and only if $\omega(x, y) \leq \lambda$.

The definition of the *similarity graph* is symmetric; an edge (x, y) occurs if and only if $\omega(x, y) \geq \lambda$. The *single-linkage clustering* is naturally defined in terms of these graphs:

Definition 1.4 (Single-linkage clustering)**.** Let H be a dissimilarity (or similarity) graph of $(\mathcal{X}, \omega)$. The *single-linkage clusters at level* λ are the connected components of $H(\lambda)$. The *single-linkage clustering* $\mathcal{C}$ is the collection of all clusters from any level, i.e.:

$$\mathcal{C} = \{C \in 2^{\mathcal{X}} : C \text{ is a connected component of } H(\lambda) \text{ for some } \lambda \in \mathbb{R}\}.$$

The single-linkage hierarchical clustering is efficiently computable. A common approach is to build the clustering agglomeratively using Kruskal's algorithm for minimum spanning trees. If a disjoint set forest data structure is used in the implementation, this approach yields a time complexity of $O(n^2 \log n)$, where n is the number of data points (Cormen et al., 2001). Quadratic time complexity is attainable using faster methods for computing minimum spanning trees, such as Prim's algorithm with Fibonacci heaps, or the optimal SLINK method of Sibson (1973).

1.2 Objective minimization and k-means

We now turn to the problem of rigorously defining the goal of the clustering procedure. In the objective minimization approach, a cost function is defined over the space of valid clusterings, such that the mission of the clustering algorithm is to return the optimal clustering – or at least a clustering with low cost.

Perhaps the most familiar scheme within this family is k-means. Let $(\mathcal{X}, d)$ be a finite metric space embedded within a larger (not necessarily finite) metric space $(\mathcal{X}', d)$, such that $\mathcal{X} \subset \mathcal{X}'$. Suppose that $\mathcal{C}$ is a flat clustering of $\mathcal{X}$ into k clusters. Given a cluster $C \in \mathcal{C}$, its *centroid* $\mu(C)$ is defined to be:

$$\mu(C) = \arg\min_{x' \in \mathcal{X}'} \sum_{x \in C} d(x, x')^2.$$

That is, the centroid of a cluster C is the point in the ambient metric space which minimizes the sum of squared distances to points in C. The k-means cost of the clustering $\mathcal{C}$ is then:

$$\psi_{k\text{-means}}(\mathcal{C}) = \sum_{C \in \mathcal{C}} \sum_{x \in C} d(x, \mu(C))^2.$$

The k-means objective function has special significance in the task of *vector quantization*, where the goal is to find a *codebook* of k vectors which summarize the full set of vectors $\mathcal{X}$. A codebook which minimizes the k-means objective is optimal in the sense that it minimizes the sum of squared errors between vectors in $\mathcal{X}$ and their closest representative in the codebook.

In practice, finding the global minimum of the k-means objective function is typically not feasible. In the familiar Euclidean setting, the problem is known to be NP-hard even when $k = 2$ (Dasgupta and Freund, 2009) and, in general, NP-hard to approximate to within a constant factor (Awasthi et al., 2015). Therefore, an approximation algorithm is typically used in practice, the most famous being Lloyd's iterative algorithm (Lloyd, 1982).

Given a set of points $\mathcal{X}$ in m-dimensional Euclidean space and a parameter k corresponding to the number of clusters, Lloyd's algorithm begins by arbitrarily selecting initial centroids $\mu_1, \ldots, \mu_k$. Next, each point $x \in \mathcal{X}$ is associated with its closest centroid according to the Euclidean distance. The location of each centroid is updated by averaging all of the points in $\mathcal{X}$ which correspond to it. This process of assigning points to centroids and updating the centroid locations is continued until convergence.

While Lloyd's algorithm is quite popular in practice, the theory surrounding it is bleak. It may converge to something other than a local minimum of the k-means objective function, and there is no known useful upper bound on the approximation error (Shalev-Shwartz and Ben-David, 2014). Even the number of iterations until convergence is not well-bounded, as Vattani (2011) shows that there exist data configurations for which the running time is exponential in the number of points.

More broadly, it is debatable whether minimization of the k-means objective function is an appropriate goal in clustering settings outside of vector quantization. Consider, for instance, a set of points $\mathcal{X}$ sampled from a density f supported on Euclidean space. A natural goal of clustering in this setting is to recover some finite summary of the underlying density. But it is unclear how the clusters of the k-means-optimal clustering recover the structure of the density f.

There are many other objective minimization approaches to clustering apart from k-means, including the related k-median cost for vector clustering, and the RatioCut cost for graph clustering whose relaxation leads to spectral clustering with the graph Laplacian (von Luxburg, 2007). A notable recent addition to this family is a cost function for hierarchical clustering, introduced by Dasgupta (2015). It can be shown that this objective function assigns intuitively-reasonable costs in several canonical clustering tasks. While minimizing the objective is NP-hard, Charikar and Chatziafratis (2017) show that it is efficient to approximate.

1.3 Axiomatic approaches

An alternative approach to formalizing the goal of clustering is to identify *axioms* which describe the desirable behavior of a clustering method. Having made the axioms rigorous, we ask what sort of methods exist which satisfy them. In this section, we review axiomatic approaches to analyzing both flat and hierarchical clustering.

1.3.1 Kleinberg's impossibility results

Kleinberg (2003) formalizes three natural properties which any flat clustering method should have: *scale-invariance*, *richness*, and *consistency*. More precisely, let f be a clustering algorithm which takes as input a finite metric space $(\mathcal{X}, d_{\mathcal{X}})$ and outputs a flat clustering $\mathcal{C}$ of $\mathcal{X}$. That is, $f(\mathcal{X}, d_{\mathcal{X}}) = \mathcal{C}$. Kleinberg's three axioms are as follows:

1. *Scale-invariance*: Scaling the finite metric space shouldn't alter the clustering. That is, if $d'_{\mathcal{X}}$ is defined such that $d'_{\mathcal{X}}(x, x') = \alpha \cdot d_{\mathcal{X}}(x, x')$ for any $\alpha > 0$ and $x, x' \in \mathcal{X}$, then $f(\mathcal{X}, d'_{\mathcal{X}}) = f(\mathcal{X}, d_{\mathcal{X}})$.

2. *Richness*: We should be able to recover any particular clustering with an appropriate choice of metric. Concretely, fix some partition $\mathcal{C}$ of $\mathcal{X}$. There must exist a metric $d_{\mathcal{X}}$ on $\mathcal{X}$ such that $f(\mathcal{X}, d_{\mathcal{X}}) = \mathcal{C}$.

3. *Consistency*: Shrinking clusters and moving them apart should not change the clustering. To be precise, suppose we apply our clustering algorithm to $(\mathcal{X}, d_{\mathcal{X}})$ and obtain $\mathcal{C}$, that is: $f(\mathcal{X}, d_{\mathcal{X}}) = \mathcal{C}$. Now let $d'_{\mathcal{X}}$ be any metric such that, for all $x, x' \in \mathcal{X}$,

   (a) if x and x' are in the same block of $\mathcal{C}$, then $d'_{\mathcal{X}}(x, x') \leq d_{\mathcal{X}}(x, x')$,

   (b) if x and x' are in *different* blocks of $\mathcal{C}$, then $d'_{\mathcal{X}}(x, x') \geq d_{\mathcal{X}}(x, x')$.

   Then $f(\mathcal{X}, d_{\mathcal{X}}) = f(\mathcal{X}, d'_{\mathcal{X}})$.

Each of these properties is natural, and we might require that any reasonable clustering algorithm have them. Kleinberg's (surprising) result, however, is that no algorithm can

exist which satisfies all three axioms simultaneously. Kleinberg proceeds to show that for any pair of axioms an algorithm exists which satisfies both (but not the third). Interestingly, any center-based clustering method such as k-means does not satisfy *consistency*.

Shalev-Shwartz and Ben-David (2014) argue that Kleinberg's impossibility result should be interpreted as a no-free-lunch theorem for clustering; i.e., there is no single flat clustering method which produces the "correct" clustering of all data sets. Rather, the correctness of a clustering algorithm is limited to a particular application. On the other hand, Ben-David and Ackerman (2009) argue that it is measures of the quality of a clustering that should be axiomatized, rather than the clustering methods themselves. In this view, Kleinberg's negative result is a consequence of his formalism, and reformulating his axioms in the framework of clustering quality measures does not necessarily lead to impossibility.

1.3.2 Characterization of single-linkage

While Kleinberg (2003) achieves an impossibility result in his study of flat clustering, Carlsson and Mémoli (2010) analyze hierarchical clustering and find uniqueness. In particular, the authors examine so-called *ultrametric* clustering methods. Recall that a *finite metric space* is a pair $(\mathcal{X}, d)$, where $\mathcal{X}$ is a finite collection of objects and $d : \mathcal{X} \times \mathcal{X} \to [0, \infty)$ is a *metric* satisfying the metric properties for all $x, y, z \in \mathcal{X}$:

1. $d(x, y) = d(y, x)$;
2. $d(x, y) = 0 \Leftrightarrow x = y$; and
3. $d(x, y) \leq d(x, z) + d(z, y)$.

A metric space $(\mathcal{X}, u)$ is a *finite ultrametric space* if, in addition, for all $x, y, z \in \mathcal{X}$, $\max\{u(x, y), u(x, z)\} \geq u(x, z)$. This inequality is known as the *strong triangle inequality* or the *ultrametric inequality*. A consequence of this inequality is that all triangles are isosceles in an ultrametric space.

Carlsson and Mémoli (2010) analyze methods which take a finite metric space $(\mathcal{X}, d)$ to a finite ultrametric space $(\mathcal{X}, u)$. The natural hierarchical clustering associated with a finite ultrametric space $(\mathcal{X}, u)$ is most easily written in terms of the equivalence relation $\sim_\lambda$, defined as $x \sim_\lambda y \Leftrightarrow u(x, y) \leq \lambda$. The equivalence classes of $\sim_\lambda$ are called the *clusters* at level λ. The set of all equivalence classes for all levels $\lambda \in [0, \infty)$ is a *hierarchical clustering* of $\mathcal{X}$ as we have formalized in Definition 1.2. Therefore, any map from a metric space $(\mathcal{X}, d)$ to an ultrametric space $(\mathcal{X}, u)$ induces a hierarchical clustering on $\mathcal{X}$.

A natural and well known ultrametric map is the one induced by the *single-linkage* clustering described in Definition 1.3 and Definition 1.4 above. Given a finite metric space $(\mathcal{X}, d)$, let H be the corresponding dissimilarity graph; i.e., H is the function mapping a scalar λ to the graph which contains the edge (x, y) if and only if $d(x, y) \leq \lambda$. The *single-linkage ultrametric* u_{SL} is defined by

$$u_{\mathrm{SL}}(x, y) = \min\{\lambda : x \text{ and } y \text{ are connected in } H(\lambda)\}.$$

The single-linkage construction is only one possible map from a metric space to an ultrametric space; other commonly-used maps include *average-linkage* and *complete-linkage*. Carlsson and Mémoli (2010) show, however, that single-linkage is the unique ultrametric clustering method satisfying a set of natural axioms. In particular, suppose f is an ultrametric clustering method such that:

1. $f(\{x, y\}, \left(\begin{smallmatrix} 0 & \delta \\ \delta & 0 \end{smallmatrix}\right)) = (\{x, y\}, \left(\begin{smallmatrix} 0 & \delta \\ \delta & 0 \end{smallmatrix}\right)$ for any $\delta > 0$;
2. whenever $(\mathcal{X}, d_{\mathcal{X}})$ and $(\mathcal{Y}, d_{\mathcal{Y}})$ are two finite metric spaces, and $\varphi : \mathcal{X} \to \mathcal{Y}$ such that $d_{\mathcal{X}}(x, x') \geq d_{\mathcal{Y}}(\varphi(x), \varphi(x'))$ for all $x, x' \in \mathcal{X}$, then $u_{\mathcal{X}}(x, x') \geq u_{\mathcal{Y}}(\varphi(x), \varphi(x'))$ is true for all $x, x' \in \mathcal{X}$, where $f(\mathcal{X}, d_{\mathcal{X}}) = (\mathcal{X}, u_{\mathcal{X}})$ and $f(\mathcal{Y}, d_{\mathcal{Y}}) = (\mathcal{Y}, u_{\mathcal{Y}})$; and
3. for any finite metric space $(\mathcal{X}, d)$, $u(x, x') \geq \min_{y \neq y'} d(y, y')$ where $f(\mathcal{X}, d) = (\mathcal{X}, u)$.

Then $f : (\mathcal{X}, d) \mapsto (\mathcal{X}, u_{\mathrm{SL}})$; i.e., f is exactly single-linkage.

Like Kleinberg's axioms, the above properties are quite natural, especially when $u(x, y)$ is intrepreted as the "effort" required to cluster x and y together. The first property ensures that the effort required to bring two objects together in a finite metric space consisting of only those two objects is exactly the distance between them. The last property enforces our intuition that the effort required to bring any two objects together is at least as large as the distance between the closest pair of points. The second axiom, called *functoriality*, encodes our belief that shrinking the distance between a pair of points can only make them easier to cluster together. Unlike Kleinberg's axioms, however, the above three properties do not lead to impossibility, but rather characterize the single-linkage algorithm.

Linkage-based ultrametric clustering methods are commonly used in the biological sciences, where they fall under the name of "numerical taxonomy" (Sneath et al., 1962). It is interesting to note that, despite its theoretical footing, single-linkage is generally disfavored in comparison to the complete- and average-linkage methods. This is because single-linkage exhibits the so-called "chaining" effect, in which distinct regions of high density are nevertheless clustered together at low levels of the tree due to the presence of a thin, sparse chain of closely-spaced points (Lance and Williams, 1967).

1.4 Model-based methods

The objective function and axiomatic approaches to the formalization of clustering are typically agnostic as to the source of the data. In a third approach, we assume that the data are generated by an underlying statistical model. In this view, the goal of clustering is the recovery of the natural structure of the model from a finite sample. In particular, we are concerned with *consistent* methods which converge to the underlying structure as the size of the data grows.

The formulation of a model-based theory of statistically-consistent clustering has four major components:

Model: First, we model the process which generates the data.

Define clusters: Second, we rigorously define the cluster structure of the model. This is the structure which we hope to recover through clustering.

Define consistency: Third, we adopt a precise notion of statistical consistency which formalizes the sense in which the output of a clustering method can be said to converge to the structure of the model as the size of the data set grows.

Prove existence: Finally, we demonstrate that consistent clustering algorithms exist.

For our theory to have practical relevance, our model of the data generating process must adequately approximate the way in which real data are generated. This suggests the analysis of rich statistical models. On the other hand, the complexity of a model typically increases with its richness. Complex models are often hard to analyze, which presents an obstacle in developing the remaining parts of the clustering theory. Many clustering consistency results are situated in simpler models precisely because they are more tractable.

However, a statistical-based theory of clustering formulated in a sufficiently-rich model has an advantage over the axiomatic and objective function theories: the meaning of an individual data cluster is clearer. Clustering methods are often applied in the interest of *data exploration* where the goal is to provide results which are interpretable by a human analyst. The fact that a cluster appears in a clustering which minimizes an objective function is perhaps not of much use in interpreting the results. On the other hand, the appearance of a cluster in the output of a model-based method reveals some aspect of the underlying distribution, which is often the aim of data analysis.

In this dissertation, we develop statistical theories of clustering in rather general, non-parametric models. In particular, Chapter 2 studies the *density* setting, while Chapter 3 examines the clustering of graphs generated from a *graphon* – a powerful random graph model of much recent interest in the statistics and mathematics literature. In each case, the canonical cluster structure of the model will turn out to be hierarchical. We therefore study

the sense in which hierarchical clustering algorithms converge to the infinite tree underlying the distribution. We will see that, in both settings, single-linkage clustering as applied to an appropriately-defined pre-processing of the data will yield a consistent algorithm.

Chapter 2

The density model

In this chapter, we study clustering under the assumption that the data are samples from a probability density. A natural goal of clustering in this setting is to recover "islands" of high probability; that is, the output of a clustering algorithm should identify the distinct peaks of the density landscape. Wishart (1969) and Hartigan (1975) made this notion precise by defining the *high-density clusters* of a density $f : \mathcal{X} \to \mathbb{R}^+$ to be the connected components of the level set $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for some λ. The collection of all high density clusters of f at any level λ has hierarchical structure, in the sense that clusters from higher levels nest within clusters from lower levels. As a result, this collection is known as the *density cluster tree* of f. In this view, the goal of clustering is to recover the density cluster tree from data.

We are interested in proving the correctness of clustering algorithms which are density tree estimators. In particular, we will study the sense in which the output of a hierarchical clustering algorithm converges to the infinite underlying density cluster tree as the size of the data grows. In order to do so, we must adopt a formal notion of convergence. One natural approach is to require that any two distinct high density clusters are separated by the clustering algorithm given enough samples. This notion was introduced by Hartigan (1981) and is known as *Hartigan consistency*. While Hartigan consistency is a desirable property of any density tree estimator, it is well known that it does not fully capture the properties of convergence that one would *a priori* expect. In particular, it does not exclude clusterings which are very different in structure from the underlying probability distribution.

In this chapter, we identify two distinct undesirable clustering configurations permitted by Hartigan consistency – *over-segmentation* (identified as the problem of *false clusters* by

Chaudhuri et al., 2014) and *improper nesting*. We observe that both configurations result from clusters merging at the wrong level. To ensure that clusters merge at the correct level, we propose two basic limit properties sufficient for hierarchical cluster convergence: *minimality* and *separation*. Together, these properties are strictly stronger than Hartigan consistency and in fact rule out the aforementioned "improper" clusterings.

Furthermore, we introduce the *merge distortion metric* as a quantitative measure of the distance between two cluster trees. We show that if a sequence of clusterings converges to the density cluster tree of f in the sense of merge distortion, the sequence necessarily satisfies the above properties of minimality and separation. Conversely, we show that slightly stronger versions of minimality and separation imply convergence in merge distortion, and are therefore equivalent to convergence.

Still, attempts to formulate some intuitively desirable properties of clustering have led to well-known impossibility results, such as those proven by Kleinberg (2003). In order to show that our definitions correspond to actual objects, and, furthermore, to realistic algorithms, we analyze the robust single-linkage clustering proposed by Chaudhuri and Dasgupta (2010). We prove convergence of that algorithm under our merge distortion metric and hence show that it satisfies separation and minimality conditions.

2.1 Related work

The problem of devising an algorithm which provably converges to the true density cluster tree in the sense of Hartigan has a long history. Hartigan (1981) proved that single linkage clustering is *not* consistent in dimensions larger than one. Previous to this, Wishart (1969) had introduced a more robust version of single linkage, but its consistency had not been known. Stuetzle and Nugent (2010) introduced another generalization of single-linkage designed to estimate the density cluster tree, but again consistency was not established.

Recently, however, two distinct consistent algorithms have been introduced: The robust single linkage algorithm of Chaudhuri and Dasgupta (2010), and the tree pruning method of Kpotufe and Luxburg (2011). Both algorithms are analyzed together, along with a pruning extension, in Chaudhuri et al. (2014). The robust single linkage algorithm was generalized in Balakrishnan et al. (2013) to densities supported on a Riemannian submanifold of $\mathbb{R}^d$. We analyze the algorithm of Chaudhuri and Dasgupta (2010) in Section 2.6. Chaudhuri and Dasgupta (2010) provide several theorems which make precise the sense in which clusters are connected and separated at each step of the robust single linkage algorithm. This work translates their results to our formalism, thereby proving that robust single linkage converges to the density cluster tree in the merge distortion metric.

A central contribution of this chapter will be to introduce notions which extend Hartigan consistency and are desirable properties of any algorithm which estimates the density cluster tree. In a related direction, Kleinberg (2003) outlined three desirable properties of a clustering method, and proved that no method satisfying all three exists. Ben-David and Ackerman (2009) argued that the impossibility result of Kleinberg is tied to his formalism, and showed that axioms similar to his can be made consistent by axiomatizing clustering quality measures as opposed to clustering functions themselves. Zadeh and Ben-David (2009) and Ackerman et al. (2010) presented axiomatic characterizations of linkage-based clustering algorithms. Similarly, Carlsson and Mémoli (2010) introduced *functoriality* as one of three axioms related to Kleinberg's and showed that single linkage agglomerative clustering is the only method which simultaneously satisfies each.

2.2 Preliminaries and definitions

Given a density f supported on $\mathcal{X} \subset \mathbb{R}^d$, a natural way to cluster $\mathcal{X}$ is into regions of high density. Hartigan (1975) made this notion precise by defining a *high-density cluster* of f to be a connected component of the superlevel set $\{f \geq \lambda\} := \{x \in \mathcal{X} : f(x) \geq \lambda\}$ for

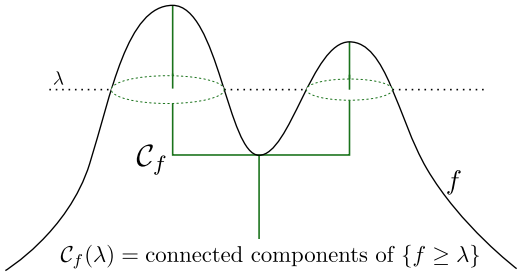


Figure 2.1: The high-density clusters of f at level λ.

any $\lambda \geq 0$; see Figure 2.1 for a depiction of this definition. It is clear that high-density clusters exhibit the nesting property: If C is a connected component of $\{f \geq \lambda\}$, and C' is a connected component of $\{f \geq \lambda'\}$, then either $C \subseteq C'$, $C' \subseteq C$, or $C \cap C' = \emptyset$. We can therefore interpret the set of all high-density clusters of a density f as a *cluster tree*:

Definition 2.1 (Density cluster tree of f)**.** Let $\mathcal{X} \subset \mathbb{R}^d$ and consider any $f : \mathcal{X} \to \mathbb{R}$. The *density cluster tree* of f, written $\mathcal{C}_f$, is the cluster tree whose nodes (clusters) are the connected components of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for some $\lambda \geq 0$.

We note that the density cluster tree of f is closely related to the so-called *split tree* studied in the computational geometry and topology literature as a variant of the *contour tree*; see e.g, (Carr et al., 2003).

In practice we do not have access to the true density f, but rather a finite collection of samples $X_n \subset \mathcal{X}$ drawn from f. We may attempt to recover the structure of the density cluster tree $\mathcal{C}_f$ by applying a hierarchical clustering algorithm to the sample, producing a discrete cluster tree $\hat{\mathcal{C}}_{f,n}$ whose clusters are subsets of X_n. In order to discuss the sense in which the discrete estimate $\hat{\mathcal{C}}_{f,n}$ is consistent with the density cluster tree $\mathcal{C}_f$ in the limit $n \to \infty$, Hartigan (1981) introduced a notion of convergence which has since been referred to as *Hartigan consistency*. We follow Chaudhuri and Dasgupta (2010) in defining Hartigan consistency in terms of the density cluster tree:

Definition 2.2 (Hartigan consistency)**.** Suppose a sample $X_n \subset \mathcal{X}$ of size n is used to construct a cluster tree $\hat{\mathcal{C}}_{f,n}$ that is an estimate of $\mathcal{C}_f$. For any sets $A, A' \subset \mathcal{X}$, let A_n

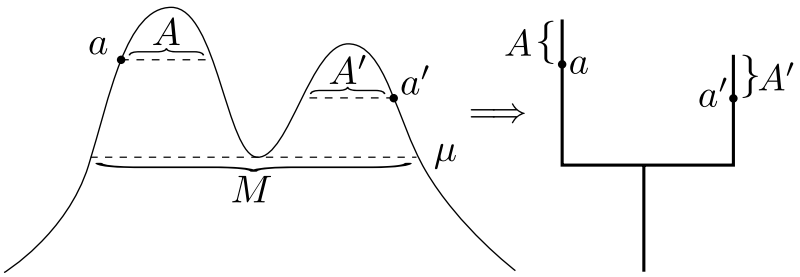


Figure 2.2: The merge height of a pair of points.

(respectively A'_n) denote the smallest cluster of $\hat{\mathcal{C}}_{f,n}$ containing $A \cap X_n$ (respectively, $A' \cap X_n$). We say $\hat{\mathcal{C}}_{f,n}$ is consistent if, whenever A and A' are different connected components of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for some $\lambda > 0$, $\mathbb{P}(A_n \text{ is disjoint from } A'_n) \to 1$ as $n \to \infty$.

A major goal of this chapter is to develop notions of consistency which are stronger than Hartigan's. For this, it will be useful to talk about the "height" at which two points in a clustering merge. To motivate our definition, consider the two points a and a' which sit on the surface of the density depicted in Figure 2.2. Intuitively, a sits at height $f(a)$ on the surface, while a' sits at $f(a')$. If we look at the superlevel set $\{f \geq f(a')\}$, we see that a and a' lie in two different high-density clusters. As we sweep $\lambda < f(a')$, the disjoint components of $\{f \geq \lambda\}$ containing a and a' grow, until they merge at height μ. We therefore say that the *merge height* of a and a' is μ.

We may also interpret the situation depicted in Figure 2.2 in the language of the density cluster tree. Let A be the connected component of $\{f \geq f(a)\}$ which contains a, and let A' be the component of $\{f \geq f(a')\}$ containing a'. Recognize that A and A' are nodes in the density cluster tree. As we walk the unique path from A to the root, we eventually come across a node M which contains both a and a'. Note that M is a connected component of the superlevel set $\{f \geq \mu\}$. It is desirable to assign a height to the entire cluster M, and a natural choice is therefore μ.

We extend this intuition to cluster trees which may not, in general, be associated with a density f by introducing the concept of a height function:

Definition 2.3 (Cluster tree with height function)**.** A cluster tree with a height function is a triple $\mathsf{C} = (X, \mathcal{C}, h)$, where X is a set of objects, $\mathcal{C}$ is a cluster tree of X, and $h : X \to \mathbb{R}$ is a height function mapping each point in X to a "height". Furthermore, we define the height of a cluster $C \in \mathcal{C}$ to be the lowest height of any point in the cluster. That is, $h(C) = \inf_{x \in C} h(x)$. Note that the nesting property of $\mathcal{C}$ implies that if C' is a descendant of C in the cluster tree, then $h(C') \geq h(C)$.

We will be consistent in using C_f to denote the density cluster tree of f equipped with height function f. That is, $\mathsf{C}_f = (\mathcal{X}, \mathcal{C}_f, f)$. Armed with these definitions, we may precisely discuss the sense in which points – and, by extension, clusters – are connected at some level of a tree:

Definition 2.4. Let $\mathsf{C} = (X, \mathcal{C}, h)$ be a hierarchical clustering of X equipped with height function h.

1. Let $x, x' \in X$. We say that x and x' are *connected at level* λ if there exists a $C \in \mathcal{C}$ with $x, x \in C$ such that $h(C) \geq \lambda$. Otherwise, x and x' are *separated at level* λ.

2. A subset $S \subset X$ is *connected at level* λ if for any $s, s' \in S$, s and s' are connected at level λ.

3. Let $S \subset X$ and $S' \subset X$. We say that S and S' are *separated at level* λ if for any $s \in S$, $s' \in S'$, s and s' are separated at level λ.

We can now formalize the notion of *merge height*:

Definition 2.5 (Merge height)**.** Let $\mathsf{C} = (X, \mathcal{C}, h)$ be a hierarchical clustering equipped with a height function. Let $x, x' \in X$, and suppose that M is the smallest cluster of $\mathcal{C}$ containing both x and x'. That is, if $M' \in \mathcal{C}$ is a proper sub-cluster of M, then $x \notin M'$ or $x' \notin M'$. We define the *merge height* of x and x' in C, written $m_{\mathsf{C}}(x, x')$, to be the height of the cluster M in which the two points merge, i.e., $m_{\mathsf{C}}(x, x') = h(M)$. If $S \subset X$, we define the *merge height* of S to be the $\inf_{(s,s') \in S \times S} m_{\mathsf{C}}(s, s')$.

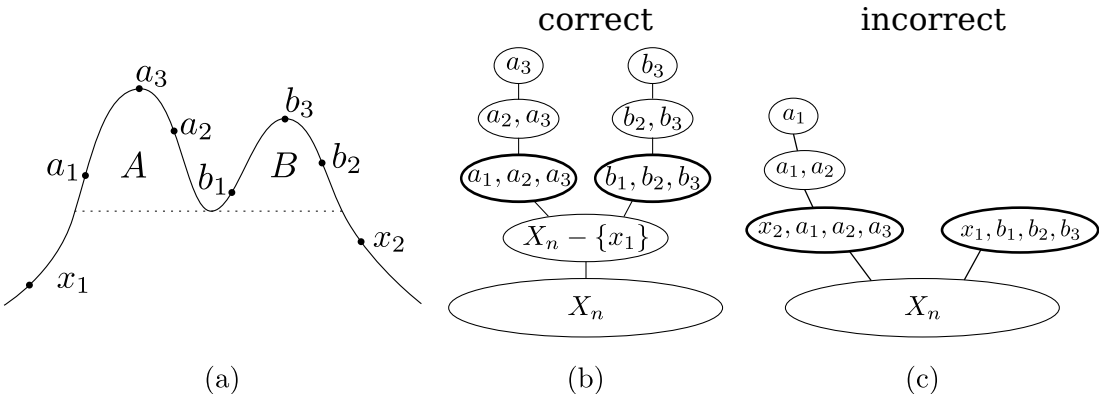


Figure 2.3: Undesirable clusterings permitted by Hartigan consistency.

In what follows, we argue that a natural and advantageous definition of convergence to the true density cluster tree is one which requires that, for any two points x, x', the merge height of x and x' in an estimate, $m_{\hat{\mathsf{C}}_{f,n}}(x, x')$, approaches the true merge height $m_{\mathsf{C}_f}(x, x')$ in the limit $n \to \infty$.

2.3 The weakness of Hartigan consistency

In this section we demonstrate that while Hartigan consistency is a desirable property, it is not sufficient to guarantee that an estimate captures the true cluster tree in a sense that matches our intuition. That is, an algorithm which is Hartigan consistent can nevertheless produce results which are quite different than the true cluster tree. Figure 2.3 illustrates the issue. Figure 2.3(a) depicts a two-peaked density f from which the finite sample X_n is drawn. The two disjoint clusters A and B are also shown. The two trees to the right represent possible outputs of clustering algorithms attempting to recover the hierarchical structure of f. Figure 2.3(b) depicts what we would intuitively consider to be an ideal clustering of X_n, whereas Figure 2.3(c) shows an undesirable clustering which does not match our intuition behind the density cluster tree of f.

First, note that while the two clusterings are very different, both satisfy Hartigan consistency. Hartigan's notion requires only separation: The smallest empirical cluster con-

taining $A \cap X_n$ must be disjoint from the smallest empirical cluster containing $B \cap X_n$ in the limit. The smallest empirical cluster containing $A \cap X_n$ in the undesirable clustering is $A_n := \{x_2, a_1, a_2, a_3\}$, whereas the smallest containing $B \cap X_n$ is $B_n := \{x_1, b_1, b_2, b_3\}$. A_n and B_n are clearly disjoint, and so Hartigan consistency is not violated. In fact, the undesirable tree separates any pair of disjoint clusters of f, and therefore represents a possible output of an algorithm which is Hartigan consistent despite being quite different from the true tree.

We will show that the undesirable configurations of Figure 2.3(c) arise because Hartigan consistency does not place strong demands on the level at which a cluster should be connected. Consider a cluster A occurring at level λ of the true density, and let A_n be the smallest empirical cluster containing all of $A \cap X_n$. In the ideal case, an algorithm would perfectly recover A such that $A_n = A \cap X_n$. It is much more likely, however, that A_n contains "extra" points from outside of A. Hartigan consistency places one constraint on the nature of these extra points: They may not belong to some other disjoint cluster of f. However, Hartigan's notion allows A_n to contain points from clusters which are *not* disjoint from A. By their nature, these points must be of density less than λ. If A_n contains such extra points, then $A \cap X_n$ is *separated* at level λ, and in fact only becomes connected at level $\min_{a \in A_n} f(a) < \delta$. Therefore, permitting $A \cap X_n$ to become connected at a level lower than λ is equivalent to allowing "extra" points of density $< \lambda$ to be contained within A_n.

The undesirable configurations depicted in Figure 2.3(c) can be divided into two distinct categories, which we term *over-segmentation* and *improper nesting*. Either of these issues may exist independently of the other, and both are symptoms of allowing clusters to become connected at lower levels than what is appropriate.

2.3.1 Over-segmentation

Over-segmentation occurs when an algorithm fragments a true cluster, returning empirical clusters which are disjoint at level λ but are in actuality part of the same connected

component of $\{f \geq \lambda\}$. The problem is recognized in the literature by Chaudhuri et al. (2014), who refer to it as the presence of *false clusters*. Figure 2.3(c) demonstrates over-segmentation by including the clusters $A_n := \{x_2, a_1, a_2, a_3\}$ and $B_n := \{x_1, b_1, b_2, b_3\}$. A_n and B_n are disjoint at level $f(x_1)$, though both are in actuality contained within the same connected component of $\{f \geq f(x_1)\}$.

It is clear that over-segmentation is a direct result of clusters connecting at the incorrect level. The severity of the issue is determined by the difference between the levels at which the cluster connects in the density cluster tree and the estimate. That is, if A is connected at λ in the density cluster tree, but $A \cap X_n$ is only connected at $\lambda - \delta$ in the empirical clustering, then the larger δ the greater the extent to which A is over-segmented.

2.3.2 Improper nesting

Improper nesting occurs when an empirical cluster C_n is the smallest cluster containing a point x, and $f(x) > \min_{c \in C_n} f(c)$. The clustering in Figure 2.3(c) displays two instances of improper nesting. First, the left branch of the cluster tree has improperly nested the cluster $\{a_1, a_2\}$, as it is the smallest cluster containing a_2, yet $f(a_1) < f(a_2)$. The right branch of the same tree has also been improperly nested in a decidedly "lazier" fashion: the cluster $\{x_1, b_1, b_2, b_3\}$ is the smallest empirical cluster containing each of b_1, b_2, and b_3, despite each being of density greater than $f(x_1)$. Improper nesting is considered undesirable because it breaks the intuition we have about the containment of clusters in the density cluster tree; Namely, if $A \subset A'$ and $a \in A$, $a' \in A'$, then $f(a) \geq f(a')$.

Note that like over-segmentation, improper nesting is caused by a cluster becoming connected at a lower level than is appropriate. For instance, suppose C_n is improperly nested; That is, it is the smallest empirical cluster containing some point x such that $f(x) > \min_{c \in C_n} f(c)$. Let $\tilde{C}$ be the connected component of $\{f \geq f(x)\}$ which contains x, and let $\tilde{C}_n$ be the smallest empirical cluster containing all of $\tilde{C} \cap X_n$. Then $C_n \subset \tilde{C}_n$ such that $\min_{c \in \tilde{C}_n} f(c) < f(x)$. In other words, $\tilde{C} \cap X_n$ is connected only below $f(x)$.

2.4 Stronger properties for consistency

We have seen that Hartigan consistency does not ensure that a clustering captures the shape of the underlying density as one would intuitively expect. In particular, we have identified two senses – *over-segmentation* and *improper nesting* – in which a hierarchical clustering method can produce results which are inconsistent with the density cluster tree, but which are not prevented by Hartigan consistency. We have shown that both are symptoms of clusters becoming connected at the wrong level. In this section, we introduce two new limit properties, termed *minimality* and *separation*, which ensure that clusters indeed merge at the correct level in the empirical cluster tree.

2.4.1 Minimality

As previously mentioned, it is not reasonable to demand that a cluster A of the density f be perfectly recovered by a clustering algorithm. Rather, if A is connected at level λ in the density cluster tree, we should allow $A \cap X_n$ to be first connected at a level $\lambda - \delta$ in the estimate, for some small positive δ. We now introduce the notion of δ-*minimality* in order to measure the sense in which A is connected at the correct level in an empirical cluster tree.

In the following, let $\mathcal{C}_f$ be the ideal cluster tree of the density f. Let X be a finite subset of the support of f, and let $\hat{\mathcal{C}}_f$ be a hierarchical clustering of X. Denote by $\hat{\mathsf{C}}_f$ the cluster tree $\hat{\mathcal{C}}_f$ equipped with f as height function.

Definition 2.6 (δ-minimal)**.** Let A be a connected component of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$. A is δ-*minimal* in $\hat{\mathsf{C}}_f$ if $A \cap X$ is connected at level $\lambda - \delta$ in $\hat{\mathsf{C}}_{f,n}$.

Intuitively, if an empirical cluster tree truly resembles the ideal cluster tree of f, then each cluster of f should be δ-minimal in the empirical tree for some small δ. For example, consider again the situation depicted in Figure 2.3 on page 18. It is easy to see that every cluster of the density f is 0-minimal in the ideal clustering shown in Figure 2.3(b).

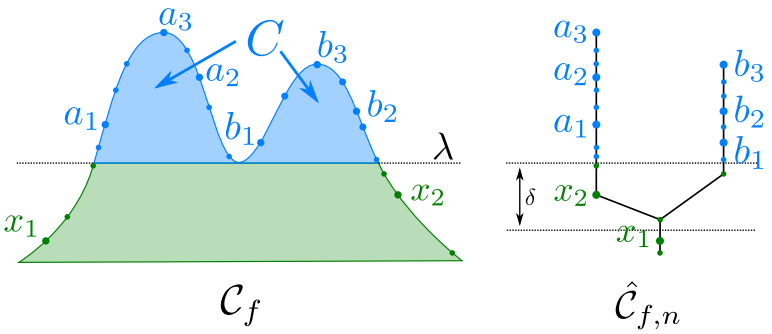


Figure 2.4: Minimality.

Indeed, the δ-minimality of a cluster C quantifies the extent to which it possibly exhibits over-segmentation or improper nesting.

Our notion of δ-minimality applies to a fixed cluster of f and a single instance of a cluster tree estimate, $\hat{\mathsf{C}}_{f,n}$. We now formally state the corresponding limit property for a sequence of estimated cluster trees $\{\hat{\mathsf{C}}_{f,n}\}$ and a sequence of random data sets $\{X_n\}$, each indexed by n. We call this notion *minimality*.

Definition 2.7 (Minimality)**.** We say that $\hat{\mathsf{C}}_{f,n}$ ensures *minimality* if given any connected component A of the superlevel set $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for some $\lambda > 0$,

$$\mathbb{P}\left(A \cap X_n \text{ is connected at level } \lambda - \delta \text{ in } \hat{\mathsf{C}}_{f,n}\right) \to 1$$

for any $\delta > 0$ as $n \to \infty$.

Figure 2.4 depicts *minimality*. The ideal cluster C is connected at level λ in the density cluster tree, but at some level $\lambda - \delta$ in the empirical cluster tree. This is an instance of *over-segmentation*. Minimality ensures that $\delta \to 0$ as $n \to \infty$, thereby limiting the magnitude of the over-segmentation.

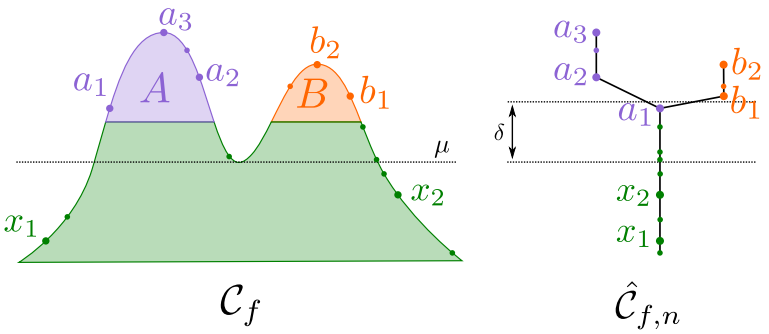


Figure 2.5: Separation.

2.4.2 Separation

Minimality concerns the level at which a cluster is connected – it says nothing about the ability of an algorithm to distinguish pairs of disjoint clusters. For this, we must complement minimality with an additional notion of consistency which ensures separation. Hartigan consistency is sufficient, but does not explicitly address the level at which two clusters are separated. We will therefore introduce a slightly different notion, which we term *separation*:

Definition 2.8 (Separation)**.** We say that $\hat{\mathsf{C}}_{f,n}$ ensures *separation* if when A and B are two disjoint connected components of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ merging at $\mu = m_{\mathsf{C}_f}(A \cup B)$,

$$\mathbb{P}\left(A \cap X_n \text{ and } B \cap X_n \text{ are separated at level } \mu + \delta \text{ in } \hat{\mathsf{C}}_{f,n}\right) \to 1$$

for any $\delta > 0$ as $n \to \infty$.

It is interesting to note that Hartigan consistency contains some weak notion of connectedness, as it requires the two sets $A \cap X_n$ and $B \cap X_n$ to be connected into clusters A_n and B_n at the same level at which they are separated. Our notion only requires that $A \cap X_n$ and $B \cap X_n$ be disjoint at this level. We "factor out" Hartigan consistency's idea of connectedness, leaving separation, and replace it with a stronger notion of minimality.

Figure 2.5 depicts *separation*. The ideal clusters merge at level μ in the density, but at a higher level $\mu + \delta$ in the empirical cluster tree. In this case, the clustering algorithm

has merged clusters too aggressively. Separation requires that as $n \to \infty$, $\delta \to 0$, thereby ensuring that the clustering is not over-connected.

2.4.3 Proof of strength

Together, minimality and separation imply Hartigan consistency.

Theorem 2.1 (Minimality and separation $\implies$ Hartigan consistency)**.** *If a hierarchical clustering method ensures both separation and minimality, then it is Hartigan consistent.*

Proof. Let A and A' be disjoint connected components of the superlevel set $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ merging at level μ. Pick any $\lambda - \mu > \delta > 0$. Let E_1 be the event that $A \cap X_n$ and $A' \cap X_n$ are separated in $\hat{\mathsf{C}}_{f,n}$ at level $\mu + \delta$, let E_2 be the event that $A \cap X_n$ is connected at level $\mu + \delta$, and let E_3 be the event that $A' \cap X_n$ is connected at level $\mu + \delta$.

Suppose all three events hold. Let A_n be the smallest cluster containing all of $A \cap X_n$, and A'_n be the smallest cluster containing all of $A' \cap X_n$. Suppose for a contradiction that there is some $x \in X_n$ such that $x \in A_n \cap A'_n$. Then either $A_n \subset A'_n$ or $A'_n \subset A_n$. Without loss of generality, assume $A_n \subset A'_n$.

By the assumption that event E_3 holds, $A \cap X_n$ is connected at level $\mu + \delta$. Since A'_n is the smallest empirical cluster containing all of $A \cap X_n$, it follows that $h(A'_n) \geq \mu + \delta$. This means that there exists a cluster at or above level $\mu + \delta$ containing all of $A \cap X_n$ and $A' \cap X_n$; namely A'_n. This contradicts the assumption that $A \cap X_n$ and $A' \cap X_n$ are separated at level $\mu + \delta$. Hence $A_n \cap A'_n = \emptyset$.

In other words, if events E_1, E_2, and E_3 hold, then A_n and A'_n are disjoint. It follows from minimality and separation that $\mathbb{P}(E_1 \wedge E_2 \wedge E_3) \to 1$ as $n \to \infty$. Hence $\mathbb{P}(A_n \cap A'_n = \emptyset) \to 1$ as $n \to \infty$; i.e., the method is Hartigan consistent. ■

2.4.4 Uniform minimality and separation

Minimality and separation have been defined as properties which are true for all clusters in the limit. In addition, we may define stronger versions of these concepts which require that all clusters approach minimality and separation simultaneously:

Definition 2.9 (Uniform minimality)**.** Fix some $\delta > 0$, and let E_δ be the event that for all clusters A of f simultaneously, $A \cap X_n$ is connected in $\hat{\mathsf{C}}_{f,n}$ at level $m_{\mathsf{C}_f}(A) - \delta$. We say that $\hat{\mathsf{C}}_{f,n}$ ensures *uniform minimality* if $\mathbb{P}(E_\delta) \to 1$ as $n \to \infty$ for any $\delta > 0$.

Definition 2.10 (Uniform separation)**.** Fix some $\delta > 0$, and let E_δ be the event that for all disjoint clusters A, A' of f simultaneously, $A \cap X_n$ and $A' \cap X_n$ are separated in $\hat{\mathsf{C}}_{f,n}$ at level $m_{\mathsf{C}_f}(A \cap A') + \delta$. We say that $\hat{\mathsf{C}}_{f,n}$ ensures *uniform separation* if $\mathbb{P}(E_\delta) \to 1$ as $n \to \infty$ for any $\delta > 0$.

The uniform versions of minimality and separation are equivalent to the weaker versions under some assumptions on the density, as the following lemma shows:

Lemma 2.1. *Let f be a density supported on $\mathcal{X}$, and let $\{\hat{\mathsf{C}}_{f,n}\}$ be a sequence of cluster trees computed from finite samples $X_n \subset \mathcal{X}$. Suppose $f \leq M$ for some $M \in \mathbb{R}$, and that for any λ, $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ contains finitely many connected components. Then*

1. *If $\{\hat{\mathsf{C}}_{f,n}\}$ ensures minimality for f, it ensures uniform minimality.*
2. *If $\{\hat{\mathsf{C}}_{f,n}\}$ ensures separation for f, it ensures uniform separation.*

Proof. We will prove the first case, in which $\hat{\mathsf{C}}_{f,n}$ ensures minimality. The proof of uniform separation follows closely, and is therefore omitted.

Pick $\delta > 0$. Let $\mathsf{C}_f(\lambda)$ denote the (finite) set of connected components of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$. Consider the collection of connected components of superlevel sets spaced $\delta/2$ apart:

$$\mathcal{D} = \bigcup_{n=0}^{\lfloor 2M/\delta \rfloor} \mathsf{C}_f(n\delta/2)$$

The fact that $\hat{\mathsf{C}}_{f,n}$ ensures minimality implies that for each $C \in \mathcal{D}$ there exists an $N(C)$ such that for all $n \geq N(C)$, $C \cap X_n$ is connected at level $h(C) - \delta/2$. Let $N = \max_{C \in \mathcal{D}} N(C)$. This is well-defined, as $\mathcal{D}$ is a finite set.

Let A be a connected component of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for an arbitrary λ. Let $\lambda' = \lfloor 2\lambda/\delta \rfloor \frac{\delta}{2}$, i.e., λ' is the largest multiple of $\delta/2$ such that $\lambda' \leq \lambda$. Then A is a subset of some connected component A' of $\{x \in \mathcal{X} : f(x) \geq \lambda'\}$. Note that $A' \in \mathcal{D}$, so that $A' \cap X_n$ is connected at level $\lambda' - \delta/2$. Therefore $A \cap X_n$ is connected at level $\lambda' - \delta/2 > (\lambda - \delta/2) - \delta/2 = \lambda - \delta$. Since A was arbitrary, and the choice of N depended only upon δ, it follows that $\hat{\mathsf{C}}_{f,n}$ ensures uniform minimality. ■

2.5 The merge distortion

The previous section introduced the notions of minimality and separation, which are desirable properties for a hierarchical clustering algorithm estimating the density cluster tree. Like Hartigan consistency, minimality and separation are limit properties, and do not directly quantify the disparity between the true density cluster tree and an estimate. We now introduce the *merge distortion* between cluster trees (equipped with height functions) which will allow us to do just that. A key property of our definitions is that, under mild conditions, convergence in merge distortion is equivalent to minimality and separation.

2.5.1 Motivation

Our concept of *merge distortion* is motivated by the existing literature surrounding *dendrogram* clustering. A *dendrogram* is a hierarchical clustering induced by a scale function. More formally, let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be a collection of objects. We follow Carlsson and Mémoli (2010) in defining a *dendrogram* to be a pair $(\mathcal{X}, \theta)$ where the scale function $\theta : [0, \infty) \to 2^{\mathcal{X}}$ maps each scale t to a partition of $\mathcal{X}$ such that:

1. at scale zero, each object is in its own singleton cluster, i.e., $\theta(0) = \{\{x_1\}, \{x_2\}, \ldots, \{x_n\}\}$;

2. at large enough scales, there is only one cluster, and it contains all object, i.e., there exists a t_0 such that $\theta(t) = \{\mathcal{X}\}$ for all $t \geq t_0$;

3. if $s \leq t$, then $\theta(s)$ is a refinement of the partition $\theta(t)$.[1]

We think of the partition $\theta(t)$ as being the set of clusters at scale t.

A dendrogram $D = (\mathcal{X}, \theta)$ is naturally associated with a function u_D which captures the first scale at which objects are clustered together. In particular, for any pair of objects x, x', let $u_D(x, x') = \min\{t \geq 0 : x, x' \text{ are in the same cluster of } \theta(t)\}$. As defined, u_D is a metric on the set of objects $\mathcal{X}$. More precisely, the pair $(\mathcal{X}, u_D)$ is a finite *ultrametric* space[2]. It is well-known that the set of dendrograms on $\mathcal{X}$ and the set of ultrametric spaces on $\mathcal{X}$ are in bijection, and so we may use a dendrogram D or its associated ultrametric u_D interchangeably.

It was observed by Carlsson and Mémoli (2010) that this ultrametric representation suggests a particularly convenient distance on the space of dendrograms. This distance makes use of ideas from metric geometry. First, recall the definition of a *correspondence*:

Definition 2.11. A *correspondence* γ between sets S and S' is a subset of $S \times S'$ such that for $\forall s \in S, \exists s' \in S'$ such that $(s, s') \in \gamma$, and $\forall s' \in S', \exists s \in S$ such that $(s, s') \in \gamma$.

Next, recall the definition of the *metric distortion*:

Definition 2.12. Let (X, d_X) and (Y, d_Y) be metric spaces, and let γ be a correspondence between X and Y. The *distortion* with respect to γ, written $\Delta_\gamma((X, d_X), (Y, d_Y))$ is given by

$$\Delta_\gamma((X, d_X), (Y, d_Y)) = \sup_{(x,y),(x',y') \in \gamma} |d_X(x, x') - d_Y(y, y')|.$$

Optimizing over the choice of correspondence leads to the *Gromov-Hausdorff distance* between metric spaces (Burago et al., 2001). Carlsson and Mémoli (2010) recognize that

[1] Carlsson and Mémoli (2010) add a further technical condition: for all s there exists $\epsilon > 0$ such that $\theta(s) = \theta(t)$ for all $t \in [s, s + \epsilon]$.

[2] Recall that a finite metric space $(\mathcal{X}, d)$ is an *ultrametric* space if it is a metric space and d satisfies the *strong triangle inequality*: for any $x, y, z \in \mathcal{X}$, $d(x, y) \leq \min\{d(x, z), d(y, z)\}$.

the metric distortion and Gromov-Hausdorff distance between *ultrametric* spaces provides a means for measuring the difference between dendrograms. They use this notion to study the stability and convergence of dendrogram clustering algorithms; in particular, they show that single-linkage clustering converges in Gromov-Hausdorff to the hierarchical structure of the *support* of the underlying density.

2.5.2 Definition

In this paper, we are interested in quantifying the distance between the density cluster tree and a finite estimate output by a clustering algorithm. The estimate, however, is not assumed to be a dendrogram, and so it has no natural ultrametric associated with it; As a result, we cannot directly apply the notion of metric distortion described above.

We will instead use the fact that the points in the finite clustering are naturally associated with a *height* – namely, a point x is associated with the value of the density at x, i.e., $f(x)$. By equipping the output of the clustering algorithm with this height, we can use the concept of the *merge height* as given in Definition 2.5 in place of the ultrametric in computing the distortion between clusterings. In particular, let $\mathsf{C} = (\mathcal{X}, \mathcal{C}, h)$ be a cluster tree equipped with height function h. Let x, x' be any pair of points in $\mathcal{X}$, and let $C_{x,x'}$ be the smallest cluster in $\mathcal{C}$ which contains them. Recall that the merge height of x, x', written $m_{\mathsf{C}}(x, x')$, is defined to be $\min_{s \in C_{x,x'}} h(s)$. Alternatively, we have:

$$m_{\mathsf{C}}(x, x') = \max_{\substack{C \in \mathcal{C} \\ x, x' \in C}} \left\{ \min_{s \in C} h(s) \right\}.$$

We note that this is not the only way to define the merge height on a cluster tree equipped with a height function, but it is particularly natural and will in fact be instrumental in the equivalence between minimality and separation and the notion of consistency we will develop below.

We now define the *merge distortion*. Let $\mathsf{C}_1 = (\mathcal{X}_1, \mathcal{C}_1, h_1)$ and $\mathsf{C}_2 = (\mathcal{X}_2, \mathcal{C}_2, h_2)$ be two cluster trees equipped with height functions. Let m_{C_1} and m_{C_2} be their respective merge height functions. We define the merge distortion between C_1 and C_2 in terms of the distortion between merge heights. In general, C_1 and C_2 cluster different sets of objects, so we will use the distortion with respect to a *correspondence* between these sets:

Definition 2.13 (Merge distortion)**.** Let $\mathsf{C}_1 = (X_1, \mathcal{C}_1, h_1)$ and $\mathsf{C}_2 = (X_2, \mathcal{C}_2, h_2)$ be two hierarchical clusterings equipped with height functions. Let $S_1 \subset X_1$ and $S_2 \subset X_2$. Let $\gamma \subset S_1 \times S_2$ be a correspondence between S_1 and S_2. The merge distortion between C_1 and C_2 with respect to γ is defined as

$$d_\gamma(\mathsf{C}_1, \mathsf{C}_2) = \sup_{(x_1,x_2),(x_1',x_2')\in\gamma} |m_{\mathsf{C}_1}(x_1, x_1') - m_{\mathsf{C}_2}(x_2, x_2')|.$$

We note that the novelty in our definition of the merge distortion is in the particular choice of merge height functions.

We now use the merge distortion in defining the sense in which a sequence of finite estimates converges to the density cluster tree $\mathsf{C}_f = (\mathcal{X}, \mathcal{C}_f, f)$. Suppose we run a hierarchical clustering algorithm on a sample $X_n \subset \mathcal{X}$ of size n drawn from f, obtaining a cluster tree $\hat{\mathcal{C}}_{f,n}$. Denote by $\hat{\mathsf{C}}_{f,n} = (X_n, \hat{\mathcal{C}}_{f,n}, f)$ the cluster tree equipped with height function f. The natural correspondence is induced by identity in X_n: That is, $\hat{\gamma}_n = \{(x, x) : x \in X_n\}$. We then define our notion of convergence to the density cluster tree with respect to this correspondence:

Definition 2.14 (Convergence to the density cluster tree)**.** We say that a sequence of cluster trees $\{\hat{\mathsf{C}}_{f,n}\}$ converges to the high density cluster tree C_f of f, written $\hat{\mathsf{C}}_{f,n} \to \mathsf{C}_f$, if for any $\varepsilon > 0$ there exists an N such that for all $n \geq N$, $d_{\hat{\gamma}_n}(\hat{\mathsf{C}}_{f,n}, \mathsf{C}_f) < \varepsilon$.

Convergence in merge distortion is depicted in Figure 2.6. As shown, the empirical cluster tree exhibits over-segmentation. As a consequence, the points a and b merge at a height $\hat{m}(a, b)$ well below their correct merge height of $m(a, b)$. Convergence in merge

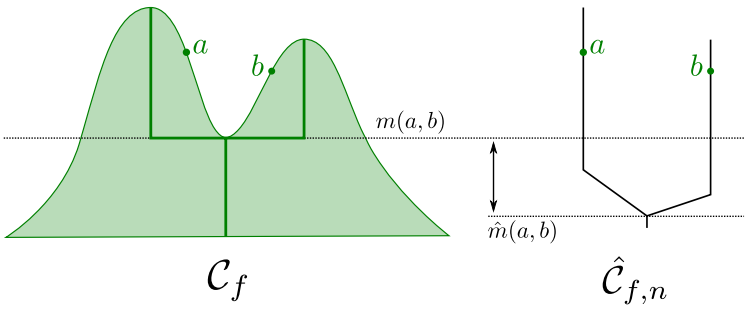


Figure 2.6: Convergence in merge distortion ensures that $\hat{m}(a,b) \to m(a,b)$.

distortion ensures that $\hat{m}(a,b) \to m(a,b)$ for all pairs of data points, thereby limiting the over-segmentation which can occur.

2.5.3 Properties

We now prove various useful properties of the merge distortion. First, we show that convergence in merge distortion to the density cluster tree is equivalent to the combination of uniform minimality and uniform separation. We then discuss stability properties of the distortion.

Equivalence with minimality and separation.

Intuitively, our property of *minimality* ensures that the empirical merge height $\hat{m}(a,b)$ converges to the true merge height from below. Similarly, *separation* ensures that the empirical merge height converges to the true merge height from above. The uniform versions of these properties together imply convergence to the density cluster tree in merge distortion.

Theorem 2.2. *If* $\hat{\mathsf{C}}_{f,n}$ *ensures uniform separation and uniform minimality, then* $\hat{\mathsf{C}}_{f,n} \to \mathsf{C}_f$.

Proof. Take any $\delta > 0$. Uniform separation and minimality imply that there exists an N such that for all λ any cluster $A \in \{x \in \mathcal{X} : f(x) \geq \lambda\}$ is connected at level $\lambda - \delta$, and for all μ any two disjoint clusters B, B' merging at μ are separated at level $\mu + \delta$. Assume

$n \geq N$, and consider any $x, x' \in X_n$. W.L.O.G., assume $f(x') \geq f(x)$. We will show that $|m_{\hat{\mathsf{C}}_{f,n}}(x, x') - m_{\mathsf{C}_f}(x, x')| \leq \delta$.

Let A be the connected component of $\{f \geq f(x)\}$ containing x, and let A' be the connected component of $\{f \geq f(x')\}$ containing x'. There are two cases: either $A' \subseteq A$, or $A \cap A' = \emptyset$.

Case I: $A' \subseteq A$. Then $m_{\mathsf{C}_f}(x, x') = f(x)$. Minimality implies that $A \cap X_n$ is connected at level $f(x) - \delta$, and therefore $m_{\hat{\mathsf{C}}_{f,n}}(x, x') \geq f(x) - \delta$. On the other hand, clearly $m_{\hat{\mathsf{C}}_{f,n}}(x, x') \leq f(x)$. Hence $|m_{\hat{\mathsf{C}}_{f,n}}(x, x') - m_{\mathsf{C}_f}(x, x')| \leq \delta$.

Case II: $A \cap A' = \emptyset$. Let $\mu := m_{\mathsf{C}_f}(x, x')$ be the merge height of x and x' in the density cluster tree of f, and suppose that M is the connected component of $\{f \geq \mu\}$ containing x and x'. Then separation implies that x and x' are separated at level $\mu + \delta$, such that $m_{\hat{\mathsf{C}}_{f,n}}(x, x') < \mu + \delta$. On the other hand, minimality implies that $M \cap X_n$ is connected at level $\mu - \delta$, so that $m_{\hat{\mathsf{C}}_{f,n}}(x, x') \geq \mu - \delta$. Therefore $|m_{\hat{\mathsf{C}}_{f,n}}(x, x') - m_{\mathsf{C}_f}(x, x')| \leq \delta$. ■

We now show that the converse is also true, and so convergence in merge distortion is equivalent to the combination of uniform separation and uniform minimality.

Theorem 2.3. $\hat{\mathsf{C}}_{f,n} \to \mathsf{C}_f$ *implies 1) uniform minimality and 2) uniform separation.*

Proof. Our proof consists of two parts.

Part I: $\hat{\mathsf{C}}_{f,n} \to \mathsf{C}_f$ implies uniform minimality. Pick any $\delta > 0$ and let n be large enough that $d(\mathsf{C}_f, \hat{\mathsf{C}}_{f,n}) < \delta$. Let A be a connected component of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for arbitrary λ. Let $a, a' \in A \cap X_n$. Then $m_{\hat{\mathsf{C}}_{f,n}}(a, a') > m_{\mathsf{C}_f}(a, a') - \delta$. But a and a' are elements of A, such that $m_{\mathsf{C}_f}(a, a') \geq \lambda$. Hence $m_{\hat{\mathsf{C}}_{f,n}}(a, a') > \lambda - \delta$. Since a and a' were arbitrary, it follows that $A \cap X_n$ is connected at level $\lambda - \delta$.

Part II: $\hat{\mathsf{C}}_{f,n} \to \mathsf{C}_f$ implies uniform separation. Pick any $\delta > 0$ and let n be large enough that $d(\mathsf{C}_f, \hat{\mathsf{C}}_{f,n}) < \delta$. Let A and A' be disjoint connected components of $\{x \in \mathcal{X} : f(x) \geq \lambda\}$ for arbitrary λ. Let $\mu := m_{\mathsf{C}_f}(A \cup A')$ be the merge height of A and A' in the density cluster tree. Take any $a \in A \cap X_n$ and $a' \in A' \cap X_n$. Then $m_{\hat{\mathsf{C}}_{f,n}}(a, a') < m_{\mathsf{C}_f}(a, a') + \delta = \mu + \delta$.

Therefore a and a' are separated at level $\mu + \delta$. Since a and a' were arbitrary, it follows that $A \cap X_n$ and $A' \cap X_n$ are separated at level $\mu + \delta$. ■

Stability.

An important property to study for a distance measure is its stability; namely, to quantify how much the cluster tree varies as its input is perturbed. We provide two such results.

The first result says that the density cluster tree induced by a density function is stable under the merge distortion with respect to L_∞-perturbation of the density function.

Theorem 2.4 (L_∞-stability of true cluster tree)**.** *Given a density function $f : \mathcal{X} \to \mathbb{R}$ supported on $\mathcal{X} \subset \mathbb{R}^d$, and a perturbation $\tilde{f} : \mathcal{X} \to \mathbb{R}$ of f, let $\mathcal{C}_f$ and $\mathcal{C}_{\tilde{f}}$ be the resulting density cluster trees as defined in Definition 2.1, and let $\mathsf{C}_f := (\mathcal{X}, \mathcal{C}_f, f)$ and $\mathsf{C}_{\tilde{f}} := (\mathcal{X}, \mathcal{C}_{\tilde{f}}, \tilde{f})$ denote the cluster trees equipped with height functions. We have $d_\gamma(\mathsf{C}_f, \mathsf{C}_{\tilde{f}}) \leq \|f - \tilde{f}\|_\infty$, where $\gamma \subset \mathcal{X} \times \mathcal{X}$ is the natural correspondence induced by identity $\gamma = \{(x, x) \mid x \in \mathcal{X}\}$.*

Proof. Set $\delta = \|f - \tilde{f}\|_\infty$. Let x, x' be two arbitrary points from X. We need to show that $|m_{\mathsf{C}_f}(x, x') - m_{\mathsf{C}_{\tilde{f}}}(x, x')| \leq 4\delta$, which will then implies the theorem. In what follows, we prove that $m_{\mathsf{C}_f}(x, x') \leq m_{\mathsf{C}_{\tilde{f}}}(x, x') + 4\delta$.

Let $m = m_{\mathsf{C}_f}(x, x')$ denote the merge height of x and x' w.r.t. C_f. This means that there exists a connected component $C \in \{y \in \mathcal{X} \mid f(y) \geq m\}$ such that $x, x' \in C$. Since $\|f - \tilde{f}\|_\infty = \delta$, we have that for any point $y \in C$, $|\tilde{f}(y) - f(y)| \leq \delta$ and thus $\tilde{f}(y) \geq m - \delta$. Hence all points in C must belong to the same connected component, call it $\tilde{C}(\supseteq C) \in \{y \in \mathcal{X} \mid \tilde{f}(y) \geq m - \delta\}$ with respect to the clustering $\mathcal{C}_{\tilde{f}}$. It then follows that the merge height $m_{\mathsf{C}_{\tilde{f}}}(x, x') \geq m - \delta$. Combining this with that $\|f - \tilde{f}\|_\infty = \delta$, we have:

$$\begin{aligned} m_{\mathsf{C}_{\tilde{f}}}(x, x') &= \tilde{f}(x) + \tilde{f}(x') - 2m_{\mathsf{C}_{\tilde{f}}}(x, x') \\ &\leq f(x) + \delta + f(x') + \delta - 2m + 2\delta = m_{\mathsf{C}_f}(x, x') + 4\delta. \end{aligned}$$

The proof for $m_{\mathsf{C}_f}(x, x') \leq m_{\mathsf{C}_{\tilde{f}}}(x, x') + \delta$ is symmetric. The theorem then follows. ■

The second result states that given a fixed hierarchical clustering, the cluster tree is stable w.r.t. small changes of the height function it is equipped with.

Theorem 2.5 (L_∞-stability w.r.t. f)**.** *Given a cluster tree $(X, \mathcal{C})$, let $\mathsf{C}_1 = (X, \mathcal{C}, f_1)$ and $(C)_2 = (X, \mathcal{C}, f_2)$ be the hierarchical clusterings equipped with two height function f_1 and f_2, respectively. Let $\gamma : X \times X$ be the natural correspondence induced by identity on X; that is, $\gamma = \{(x, x) \mid x \in X\}$. We then have $d_\gamma(\mathsf{C}_1, \mathsf{C}_2) \leq 2\|f_1 - f_2\|_\infty$.*

Proof. Set $\delta := \|f_1 - f_2\|_\infty$. Let x, x' be two arbitrary points from X. We need to show that $|m_{\mathsf{C}_1}(x, x') - m_{\mathsf{C}_2}(x, x')| \leq 4\delta$, which will then implies the theorem. In what follows, we prove that $m_{\mathsf{C}_2}(x, x') \leq m_{\mathsf{C}_1}(x, x') + 4\delta$.

Let $m_1 = m_{\mathsf{C}_1}(x, x')$ denote the merge height of x and x' w.r.t. C_1. This means that there exists a cluster $C \in \mathcal{C}$ such that $x, x' \in C$ and $f_1(C) = m_1$. Since $f_i(C) = \min_{y \in C} f_i(y)$, for $i = 1, 2$, we thus have that $f_2(C) \in [m_1 - \delta, m_1 + \delta]$. It then follows that $m_{\mathsf{C}_2}(x, x') \geq f_2(C) \geq m_1 - \delta$. Combining with that $\|f_1 - f_2\|_\infty = \delta$, we have:

$$\begin{aligned} m_{\mathsf{C}_2}(x, x') &= f_2(x) + f_2(x') - 2m_{\mathsf{C}_2}(x, x') \\ &\leq f_1(x) + \delta + f_1(x') + \delta - 2m_1 + 2\delta = m_{\mathsf{C}_1}(x, x') + 4\delta. \end{aligned}$$

The proof for $m_{\mathsf{C}_1}(x, x') \leq m_{\mathsf{C}_2}(x, x') + \delta$ is symmetric. The theorem then follows. ■

Theorem 2.5 in particular leads to the following: Given a density $f : \mathcal{X} \to \mathbb{R}$ supported on $\mathcal{X} \subset \mathbb{R}^d$, suppose we have a hierarchical clustering $\hat{\mathcal{C}}_n$ constructed from a sample $X_n \subset \mathcal{X}$. However, we do not know the true density function f. Instead, suppose we have a density estimator producing an empirical density function $\tilde{f}_n : X_n \to \mathbb{R}$. Set $\hat{\mathsf{C}}_{f,n} = (X_n, \hat{\mathcal{C}}_n, f)$ as before, and $\tilde{\mathsf{C}}_{\tilde{f},n} = (X_n, \hat{\mathcal{C}}_n, \tilde{f}_n)$. Theorem 2.5 implies that $d(\hat{\mathsf{C}}_{f,n}, \tilde{\mathsf{C}}_{\tilde{f},n}) \leq \|f - \tilde{f}_n\|_\infty$. By

the triangle inequality, this further bounds

$$d_{\hat{\gamma}_n}(\mathcal{C}_f, \tilde{\mathsf{C}}_{\tilde{f},n}) \leq d_{\hat{\gamma}_n}(\mathcal{C}_f, \hat{\mathsf{C}}_{f,n}) + \|f - \tilde{f}_n\|_\infty. \tag{2.1}$$

Assuming that the density estimator is consistent, we note that the cluster tree $\tilde{\mathsf{C}}_{\tilde{f},n}$ also converges to $\mathcal{C}_f$ if $\hat{\mathsf{C}}_{f,n}$ converges to $\mathcal{C}_f$.

This has an important implication from a practical point of view. Imagine that we are given a sequence of more and more samples $X_{n_1}, X_{n_2}, \ldots$, and we construct a sequence of hierarchical clusterings $\hat{\mathcal{C}}_{n_1}, \hat{\mathcal{C}}_{n_2}, \ldots$. In practice, in order to test whether the current hierarchical clustering converges or not, one may wish to compare two consecutive clusterings $\hat{\mathcal{C}}_{n_i}$ and $\hat{\mathcal{C}}_{n_{i+1}}$ and measure their distance. However, since the true density is not available, one cannot compute the cluster tree distance $d_{\gamma_{n_i}}(\hat{\mathsf{C}}_{f,n_i}, \hat{\mathsf{C}}_{f,n_{i+1}})$, where the correspondence is induced by the natural inclusion from $X_{n_i} \subseteq X_{n_{i+1}}$, that is, $\gamma_{n_i} = \{(x,x) \mid x \in X_{n_i}\}$. Equation (2.1) justifies the use of a consistent empirical density estimator and computing $d_{\gamma_{n_i}}(\tilde{\mathsf{C}}_{\tilde{f},n_i}, \tilde{\mathsf{C}}_{\tilde{f},n_{i+1}})$ instead.

2.6 Strong consistency of robust single-linkage

We now analyze the robust single-linkage algorithm of Chaudhuri and Dasgupta (2010) in the context of our formalism. Chaudhuri and Dasgupta (2010) as well as Chaudhuri et al. (2014) studied the sense in which robust single-linkage ensures that clusters are separated and connected at the appropriate levels of the empirical tree. Our analysis translates their results to our definitions of minimality and separation, thereby reinterpreting the convergence of robust single-linkage in terms of our merge distortion metric.

2.6.1 Description of the algorithm

We first briefly describe the robust single-linkage algorithm, and refer readers to the work of Chaudhuri and Dasgupta (2010) for details. The algorithm operates as follows. Given a

sample X_n of n points drawn from a density f supported on $\mathcal{X}$, let $d(x, x')$ be the distance between $x, x' \in X_n$. Fix parameters α and k, and perform the following steps:

1. For each $x_i \in X_n$, set $r_k(x_i) = \min\{r : B(x_i, r) \text{ contains } k \text{ points}\}$, where $B(x_i, r)$ is the ball of radius r around x_i.

2. As r grows from 0 to ∞:

   (a) Construct a graph G_r with nodes $\{x_i : r_k(x_i) \leq r\}$. Include edge (x_i, x_j) if $d(x_i, x_j) \leq \alpha r$.

   (b) Let $\mathcal{C}_n(r)$ be the connected components of G_r.

The clustering produced by the algorithm is the collection of all connected components $\mathcal{C}_n(r)$ for any r. The clusters exhibit hierarchical structure, and can be interpreted as a cluster tree. We may therefore discuss the sense in which this discrete tree converges to the ideal density cluster tree.

As an aside, we note that the robust single-linkage algorithm can be implemented by applying the usual single-linkage algorithm to a transformed distance matrix. In particular, let

$$\tilde{d}(x, x') = \max\{r_k(x),\, r_k(x'),\, \alpha \cdot d(x, x')\}.$$

Let $\tilde{\mathcal{C}}_n$ be the single-linkage dissimilarity clustering of $(X_n, \tilde{d})$, and let $\mathcal{C}_n$ be the clustering as returned by the robust single-linkage algorithm described above. It is easy to see that if C is a cluster of $\tilde{\mathcal{C}}_n$ with two or more elements, then C is also a cluster in $\mathcal{C}_n$. Moreover, the other direction also holds: if C' is a non-singleton cluster in $\mathcal{C}_n$, then it is also a cluster in $\tilde{\mathcal{C}}_n$. Hence there is a bijection between the non-singleton clusters of $\mathcal{C}_n$ and those of $\tilde{\mathcal{C}}_n$.

The robust single-linkage clustering $\mathcal{C}_n$ differs from $\tilde{\mathcal{C}}_n$ only in the presence of singleton clusters. In the single-linkage clustering of $\tilde{d}$, each point is in some cluster at level $\lambda = 0$; typically, each point x is in its own singleton cluster $\{x\}$ unless it is degerate in the sense

that there exists a point x' such that $\tilde{d}(x, x') = 0$. In the robust single-linkage algorithm, however, a point x does not enter the clustering until the level $r_k(x)$.

Therefore, in order to recover the clustering produced by the algorithm of Chaudhuri and Dasgupta (2010), we need only process the singleton clusters of $\tilde{\mathcal{C}}_n$. For a singleton cluster $S = \{x\}$, let P be its *parent* – the smallest distinct cluster in $\tilde{\mathcal{C}}_n$ which contains S. Let λ be the level at which the parent P was created; i.e., the smallest level λ at which P is a connected component of the single linkage dissimilarity graph of $\tilde{d}$. Such information is often retained by off-the-shelf single-linkage implementations. We observe that $r_k(x)$ cannot be greater than λ, since x appears in P. If $r_k(x) < \lambda$, then the singleton cluster is valid and should be kept. On the other hand, if $r_k(x) = \lambda$, the singleton cluster is spurious and should be deleted.

2.6.2 Proof of consistency

We now demonstrate the strong consistency of robust single-linkage clustering. In what follows, assume that the density f is: 1) c-Lipschitz; 2) compactly supported (and hence bounded from above); and 3) such that $\{f \geq \lambda\}$ has finitely-many connected components for any λ. We will prove that the algorithm ensures minimality and separation. This, together with the assumptions on f and Theorem 2.2, will imply convergence in the merge distortion distance.

Suppose we run the robust single-linkage algorithm on a sample of size n. Denote by v_d the volume of the d-dimensional unit hypersphere, and let $B(x, r)$ the closed ball of radius r around x in $\mathbb{R}^d$. We will write $f(B(x, r))$ to denote the probability of $B(x, r)$ under f. Define $r(\lambda)$ to be the value of r such that $v_d r^d \lambda = \frac{k}{n} + \frac{C_\delta}{n}\sqrt{kd \log n}$. Here, k is a parameter of the algorithm which we will constrain, and C_δ is the constant appearing in the Lemma IV.1 of Chaudhuri et al. (2014). First, we must show that in the limit, $G_{r(\lambda)}$ contains no points of density less than $\lambda - \epsilon$, for arbitrary ϵ.

Lemma 2.2. *Fix $\epsilon > 0$ and $\lambda \geq 0$. Then if $\alpha \geq \sqrt{2}$ and $k \geq (8C_\delta\lambda/\epsilon)^2\, d \log n$, there exists an N such that for all $n \geq N$, if $x \in G_{r(\lambda)}$, then $f(x) > \lambda - \epsilon$.*

Proof. Define $\tilde{r} = r(\lambda - \epsilon/2)$. There exists an N such that for any $n \geq N$, $\tilde{r}\mathsf{c} \leq \epsilon/4$. Consider any point $x \in G_{\tilde{r}}$. By virtue of x's membership in the graph, X_n contains k points within $B(x, \tilde{r})$. Lemma IV.1 in (Chaudhuri et al., 2014) implies that $f(B(x,\tilde{r})) > \frac{k}{n} - \frac{C_\delta}{n}\sqrt{kd\log n}$. From our Lipschitz assumption, we have $v_d\tilde{r}^d(f(x) + \tilde{r}\mathsf{c}) \geq f(B(x,\tilde{r})) > \frac{k}{n} - \frac{C_\delta}{n}\sqrt{kd\log n}$. Multiplying both sides by $\lambda - \epsilon/2$ and substituting gives:

$$\begin{aligned} v_d\tilde{r}^d(\lambda - \epsilon/2)(f(x) + \tilde{r}\mathsf{c}) &= \left(\frac{k}{n} + \frac{C_\delta}{n}\sqrt{kd\log n}\right)(f(x) + \tilde{r}\mathsf{c}), \\ &> (\lambda - \epsilon/2)\left(\frac{k}{n} - \frac{C_\delta}{n}\sqrt{kd\log n}\right). \end{aligned}$$

Therefore:

$$\begin{aligned} f(x) &> (\lambda - \epsilon/2)\left(\frac{k - C_\delta\sqrt{kd\log n}}{k + C_\delta\sqrt{kd\log n}}\right) - \tilde{r}\mathsf{c}, \\ &\geq \left(1 - 2\frac{C_\delta\sqrt{d\log n}}{\sqrt{k}}\right)(\lambda - \epsilon/2) - \epsilon/4, \\ &\geq \left(1 - \frac{\epsilon}{4\lambda}\right)(\lambda - \epsilon/2) - \epsilon/4, \\ &\geq \lambda - \epsilon. \end{aligned}$$

Hence for any point $x \in G_{\tilde{r}}$, $f(x) > \lambda - \epsilon$. Note that $\tilde{r} > r(\lambda)$, implying that any point in $G_{r(\lambda)}$ is also in $G_{\tilde{r}}$. Therefore if $x \in G_{r(\lambda)}$, $f(x) > \lambda - \epsilon$. ■

We now make our claim. We will use the following fact without proof: For any $A \in \{f \geq \lambda\}$ and $\delta > 0$, there exists an N such that for all $n \geq N$, if $A \cap X_n \neq \emptyset$, there is at least one point $x \in A \cap X_n$ with $f(x) < \lambda + \delta$. This follows immediately from the continuity of f and the inequalities in the Lemma IV.1 of Chaudhuri et al. (2014).

Theorem 2.6. *Robust single-linkage converges in probability to the density cluster tree C_f in the merge distortion distance.*

Proof. It is sufficient to prove minimality and separation, as then Theorem 2.2 will imply convergence. Fix any $\varepsilon > 0$, and let A be a connected component of $\{f \geq \lambda\}$. Define $\sigma = \varepsilon/(2\mathsf{c})$, and let A_σ be the set A thickened by closed balls of radius σ. Define $\lambda' := \inf_{x \in A_\sigma} f(x) \geq \lambda - \varepsilon/2$. Theorem IV.7 in (Chaudhuri et al., 2014) implies that there exists an N_1 such that for all $n \geq N_1$, $A \cap X_n$ is connected in $G_{r(\lambda')}$. Take $\epsilon = \varepsilon/2$ in our Lemma 2.2; there exists an N_2 above which each point x in $G_{r(\lambda')}$ has density $f(x) > \lambda' - \epsilon \geq (\lambda - \varepsilon/2) - \varepsilon/2 = \lambda - \varepsilon$. Then for all $n \geq \max\{N_1, N_2\}$, $A \cap X_n$ is connected in $G_{r(\lambda')}$ at level no less than $\lambda - \varepsilon$. This proves minimality.

Again fix $\varepsilon > 0$ and let A and A' be connected components of $\{f \geq \lambda\}$ merging at some height $\mu = m_{\mathsf{C}_f}(A \cup A')$. Let $\tilde{A}$ and $\tilde{A}'$ be the connected components of $\{f \geq \mu + \varepsilon/2\}$ containing A and A', respectively. Define $\sigma = \varepsilon/(4\mathsf{c})$, and let $\tilde{A}_\sigma$ (resp. $\tilde{A}'_\sigma$) be the set $\tilde{A}$ (resp. $\tilde{A}'$) thickened by closed balls of radius σ. Define $\mu' := \inf_{x \in \tilde{A}_\sigma \cup \tilde{A}'_\sigma} f(x) \geq \mu + \varepsilon/4$. Then Lemma IV.3 in (Chaudhuri et al., 2014) implies[3] that there exists some N_1 such that for all $n \geq N_1$, $\tilde{A} \cap X_n$ and $\tilde{A}' \cap X_n$, are disconnected in $G_r(\mu')$ and individually connected. Let N_2 be large enough that there exists a point $x_1 \in \tilde{A} \cap X_n$ with $f(x_1) < \mu + \varepsilon$. Then for all $n \geq \max\{N_1, N_2\}$, $A \cap X_n$ and $A' \cap X_n$ are separated at level $\mu + \varepsilon$. This proves separation. ■

[3] More precisely, Lemma IV.3 requires A and A' to be so-called (σ, ϵ)-separated, for some σ and ϵ. It follows from the Lipschitz-continuity of f that there is some ϵ so that A and A' are (σ, ϵ)-*separated* for this choice of σ.

Chapter 3

The graphon model

In this chapter, we consider the setting in which the objects to be clustered are the vertices of a graph sampled from a *graphon* – a very general random graph model of significant recent interest. As in the previous chapter, we develop a statistical theory of graph clustering in the graphon model using the core idea of the merge distortion. The specific contributions of this chapter are threefold. First, we define the clusters of a graphon. Our definition results in a graphon having a tree of clusters, which we call its *graphon cluster tree*. We introduce an object called the *mergeon* which is a particular representation of the graphon cluster tree that encodes the heights at which clusters merge. Second, we develop a notion of consistency for graph clustering algorithms in which a method is said to be consistent if its output converges to the graphon cluster tree. Here the graphon setting poses subtle yet fundamental challenges which differentiate it from classical clustering models, and which must be carefully addressed. Third, we prove the existence of consistent clustering algorithms. In particular, we provide sufficient conditions under which a graphon estimator leads to a consistent clustering method. We then identify a specific practical algorithm which satisfies these conditions, and in doing so present a simple graph clustering algorithm which provably recovers the graphon cluster tree.

Notation. We will use $[n]$ to denote the set $\{1, \ldots, n\}$, $\triangle$ for the symmetric difference, μ for the Lebesgue measure on $[0, 1]$, and bold letters to denote random variables.

3.1 Related work

Graphons are objects of significant recent interest in graph theory, statistics, and machine learning. The theory of graphons is rich and diverse; A graphon can be interpreted as a generalization of a weighted graph with uncountably many nodes, as the limit of a sequence of finite graphs, or, more importantly for the present work, as a very general model for generating unweighted, undirected graphs. Conveniently, any graphon can be represented as a symmetric, measurable function $W : [0, 1]^2 \to [0, 1]$, and it is this representation that we use throughout this chapter.

The graphon as a graph limit was introduced in recent years by Lovász and Szegedy (2006), Borgs et al. (2008), and others. The interested reader is directed to the book by Lovász (2012) on the subject. There has also been a considerable recent effort to produce consistent estimators of the graphon, including the work of Wolfe and Olhede (2013), Chan and Airoldi (2014), Airoldi et al. (2013), Rohe et al. (2011), and others. We will analyze a simple modification of the graphon estimator proposed by Zhang et al. (2015) and show that it leads to a graph clustering algorithm which is a consistent estimator of the graphon cluster tree.

Much of the previous statistical theory of graph clustering methods assumes that graphs are generated by the so-called *stochastic blockmodel*. The simplest form of the model generates a graph with n nodes by assigning each node, randomly or deterministically, to one of two communities. An edge between two nodes is added with probability α if they are from the same community and with probability β otherwise. A graph clustering method is said to achieve *exact recovery* if it identifies the true community assignment of every node in the graph with high probability as $n \to \infty$. The blockmodel is a special case of a graphon model, and our notion of consistency will imply exact recovery of communities.

Stochastic blockmodels are widely studied, and it is known that, for example, spectral methods like that of McSherry (2001) are able to recover the communities exactly as $n \to \infty$,

provided that α and β remain constant, or that the gap between them does not shrink too quickly. For a summary of consistency results in the blockmodel, see Abbe et al. (2015), which also provides information-theoretic thresholds for the conditions under which exact recovery is possible. In a related direction, Balakrishnan et al. (2011) examines the ability of spectral clustering to withstand noise in a hierarchical block model.

3.2 In relation to the density setting

As discussed in the previous chapter of this dissertation, the problem of defining the underlying cluster structure of a probability distribution goes back to Hartigan (1981) who considered the setting in which the objects to be clustered are points sampled from a density $f : \mathcal{X} \to \mathbb{R}^+$. In this case, the *high density clusters* of f are defined to be the connected components of the upper level sets $\{x : f(x) \geq \lambda\}$ for any $\lambda > 0$. The set of all such clusters forms the so-called *density cluster tree*. Hartigan (1981) defined a notion of consistency for the density cluster tree, and proved that single-linkage clustering is *not* consistent. In recent years, Chaudhuri and Dasgupta (2010) and Kpotufe and Luxburg (2011) have demonstrated methods which *are* Hartigan consistent. The previous chapter introduced a distance between a clustering of the data and the density cluster tree, called the *merge distortion metric*. A clustering method is said to be *consistent* if the trees it produces converge in merge distortion to density cluster tree. It was shown that convergence in merge distortion is stronger than Hartigan consistency, and that the method of Chaudhuri and Dasgupta (2010) is consistent in this stronger sense.

In the present setting, we will be motivated by the approach taken in Hartigan (1981) and the previous chapter. We note, however, that there are significant and fundamental differences between the density case and the graphon setting. Specifically, it is possible for two graphons to be equivalent in the same way that two graphs are: up to a relabeling of the vertices. As such, a graphon W is a representative of an equivalence class of graphons

modulo appropriately defined relabeling. It is therefore necessary to define the clusters of W in a way that does not depend upon the particular representative used. A similar problem occurs in the density setting when we wish to define the clusters not of a single density function, but rather of a *class* of densities which are equal almost everywhere; Steinwart (2011) provides an elegant solution. But while the domain of a density is equipped with a meaningful metric – the mass of a ball around a point x is the same under two equivalent densities – the ambient metric on the vertices of a graphon is not useful. As a result, approaches such as that of Steinwart (2011) do not directly apply to the graphon case, and we must carefully produce our own. Additionally, we will see that the procedure for sampling a graph from a graphon involves latent variables which are in principle unrecoverable from data. These issues have no analogue in the classical density setting, and present very distinct challenges.

3.3 Measure theory preliminaries

As we will see, the graphon is a measure-theoretic object. As such, we will make use of various notions and results from measure theory throughout this chapter. We collect some of these here for convenience.

In what follows, we will often deal with collections of sets which differ only by sets of zero measure. More precisely, let (Ω, Σ, μ) be a measure space. Let A, A' be any measurable sets and define $\sim_\emptyset$ to be the relation $A \sim_\emptyset A' \Leftrightarrow \mu(A \triangle A') = 0$; that is, two measurable sets are equivalent under $\sim_\emptyset$ if they differ by a null set. Write $\Sigma/\sim_\emptyset$ for the quotient space of measurable sets by $\sim_\emptyset$, and denote by $[A]_\emptyset$ the equivalence class containing the set A. Throughout, we use script letters such as $\mathscr{A}$ to denote these equivalence classes of measurable sets modulo null sets.

We can often use the normal set notation to manipulate such classes without ambiguity. For instance, if $\mathscr{A}$ and $\mathscr{A}'$ are two classes in $\Sigma/\sim_\emptyset$, we define $\mathscr{A} \cup \mathscr{A}'$ to be $[A \cup A']_\emptyset$,

where A and A' are arbitrary members of $\mathscr{A}$ and $\mathscr{A}'$. $\mathscr{A} \cap \mathscr{A}'$ and $\mathscr{A} \setminus \mathscr{A}'$ are defined similarly. We can define $\mathscr{A} \times \mathscr{A}'$ in this manner too; note that the result is an equivalence class in $\Sigma \times \Sigma / \sim_\emptyset$, where the relation $\sim_\emptyset$ is implicitly assumed to be with respect to the product measure $\mu \times \mu$. Similarly, we can unambiguously order such equivalence classes. For example, we write $\mathscr{A} \subset \mathscr{A}'$ to denote $\mu(\mathscr{A} \setminus \mathscr{A}') = 0$.

In some instances it will be more convenient to work with sets as opposed to equivalence classes of sets. In such cases we will use a *section* map ρ which returns an (often arbitrary) member of the class, $\rho(\mathscr{A})$.

We will also work with collections of measurable sets. If the collection is closed under countable unions, we may speak about the "largest" measurable sets in the collection and the "smallest", as the following definition makes precise:

Definition 3.1 (Essential maxima/minima)**.** Let (Ω, Σ, μ) be a measure space, with μ a finite measure (i.e., $\mu(\Omega) < \infty$). Let $\mathfrak{A} \subset \Sigma$ be closed under countable unions. Define the set of *essential maxima* of $\mathfrak{A}$ by

$$\operatorname{ess\,max} \mathfrak{A} = \{M \in \mathfrak{A} : \mu(A \setminus M) = 0 \quad \forall A \in \mathfrak{A}\}.$$

Likewise, define the set of *essential minima* of $\mathfrak{A}$ by

$$\operatorname{ess\,min} \mathfrak{A} = \{M \in \mathfrak{A} : \mu(M \setminus A) = 0 \quad \forall A \in \mathfrak{A}\}.$$

The essential maxima and minima satisfy the following claim:

Claim 3.1. *Let (Ω, Σ, μ) be a measure space with μ a finite measure. Let $\mathfrak{A} \subset \Sigma$ be closed under countable unions. If $\mathfrak{A}$ is nonempty,* $\operatorname{ess\,max} \mathfrak{A}$ *and* $\operatorname{ess\,min} \mathfrak{A}$ *are nonempty. Furthermore, for any $M, M' \in \operatorname{ess\,max} \mathfrak{A}$, $\mu(M \triangle M') = 0$; the same is true for $M, M' \in \operatorname{ess\,min} \mathfrak{A}$.*

Proof. The claim holds trivially if $\mathfrak{A}$ is empty, so suppose it is not. Let $\tau = \sup_{A \in \mathfrak{A}} \mu(A)$, and note that τ is finite since $\mu(\Omega)$ is finite. Then for every $n \in \mathbb{N}^+$, there exists a set $A_n \in \mathfrak{A}$ such that $\tau - \mu(A_n) < 1/n$. Construct the sequence $\langle B_n \rangle_{n \in \mathbb{N}^+}$ by defining $B_n = \bigcup_{i=1}^{n} A_i$. Then $B_n \in \mathfrak{A}$ for every n, since it is the countable union of sets in $\mathfrak{A}$. Furthermore, $B_n \subseteq B_{n+1}$, and $\lim_{n \to \infty} \mu(B_n) = \tau$. Define $M = \bigcap_{n=1}^{\infty} B_n = \bigcap_{n=1}^{\infty} A_n$. Then $M \in \mathfrak{A}$ since it is a countable union of elements of $\mathfrak{A}$, and by continuity of measure $\mu(M) = \tau$.

First we show that for any set $A \in \mathfrak{A}$, $\mu(A \setminus M) = 0$. Suppose for a contradiction that $\mu(A \setminus M) \neq 0$. We have $A \cup M = (A \setminus M) \cup M$, such that $\mu(A \cup M) = \mu(A \setminus M) + \mu(M) > \tau$. But $A \cup M$ is in $\mathfrak{A}$, since $\mathfrak{A}$ is closed under unions. This violates the fact that τ is the supremal measure of any set in $\mathfrak{A}$, and hence it must be that $\mu(A \setminus M) = 0$. Therefore $M \in \operatorname{ess\,max} \mathfrak{A}$.

Now suppose M' is an arbitrary element of $\operatorname{ess\,max} \mathfrak{A}$. We have just seen that $\mu(M' \setminus M)$ must be zero, since $M' \in \mathfrak{A}$. Likewise, $\mu(M \setminus M') = 0$. Therefore

$$\mu(M \triangle M') = \mu((M \setminus M') \cup (M' \setminus M)) = \mu(M \setminus M') + \mu(M' \setminus M) = 0$$

where we used the fact that μ is an additive set function and $M \setminus M'$ and $M' \setminus M$ disjoint. It is also clear that if $M \in \operatorname{ess\,max} \mathfrak{A}$ and N is any null set, then $M \cup N$ and $M \setminus N$ are also essential maxima. ■

3.4 The graphon model

In order to discuss the statistical properties of a graph clustering algorithm, we must first model the process by which graphs are generated. Formally, a *random graph model* is a sequence of random variables $\mathbf{G}_1, \mathbf{G}_2, \ldots$ such that the range of $\mathbf{G}_n$ consists of undirected, unweighted graphs with node set $[n]$, and the distribution of $\mathbf{G}_n$ is invariant under relabeling of the nodes – that is, isomorphic graphs occur with equal probability. A random graph model of considerable recent interest is the *graphon model*, in which the distribution over

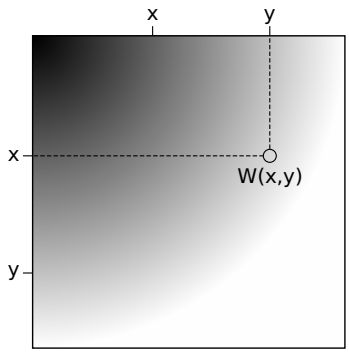


Figure 3.1: A graphon W.

graphs is determined by a symmetric, measurable function $W : [0,1]^2 \to [0,1]$ called a *graphon*. Informally, a graphon W may be thought of as the weight matrix of an infinite graph whose node set is the continuous unit interval, so that $W(x,y)$ represents the weight of the edge between nodes x and y.

An example of a graphon W is shown in Figure 3.1; in this particular case, the graphon function is $W(x,y) = \sqrt{1-x^2-y^2}$. It is conventional to plot the graphon as one typically plots an adjacency matrix: with the origin in the upper-left corner. Darker shades correspond to higher values of W.

Interpreting $W(x,y)$ as a probability suggests the following graph sampling procedure: To draw a graph with n nodes, we first select n points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ at random from the uniform distribution on $[0,1]$ – we can think of these $\mathbf{x_i}$ as being random "nodes" in the graphon. We then sample a random graph $\mathbf{G}$ on node set $[n]$ by admitting the edge (i,j) with probability $W(\mathbf{x}_i, \mathbf{x}_j)$; by convention, self-edges are not sampled. It is important to note that while we begin by drawing a set of nodes $\{\mathbf{x_i}\}$ from the graphon, the graph as given to us is labeled by integers. Therefore, the correspondence between node i in the graph and node $\mathbf{x_i}$ in the graphon is latent.

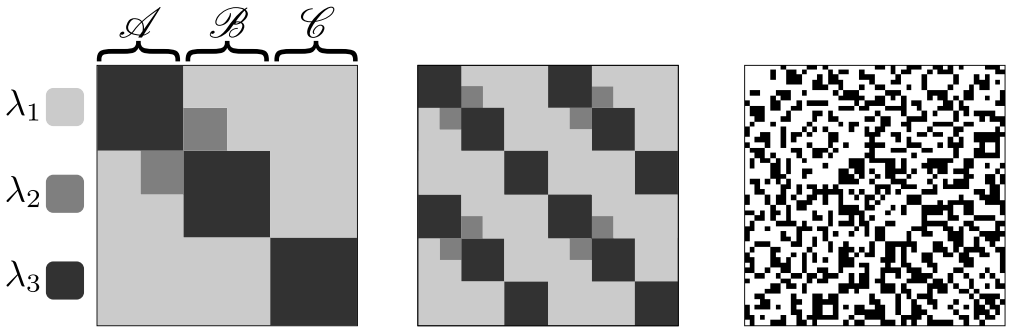
It can be shown that this sampling procedure defines a distribution on finite graphs, such that the probability of graph $G = ([n], E)$ is given by

$$\mathbb{P}_W(\mathbf{G} = G) = \int_{[0,1]^n} \prod_{(i,j)\in E} W(x_i, x_j) \prod_{(i,j)\notin E} [1 - W(x_i, x_j)] \prod_{i\in[n]} dx_i. \tag{3.1}$$

For a fixed choice of $x_1, \ldots, x_n \in [0, 1]$, the integrand represents the likelihood that the graph G is sampled when the probability of the edge (i, j) is assumed to be $W(x_i, x_j)$. By integrating over all possible choices of $x_1, \ldots, x_n$, we obtain the probability of the graph.

A very general class of random graph models may be represented as graphons. In particular, a random graph model $\mathbf{G}_1, \mathbf{G}_2, \ldots$ is said to be *consistent* if the random graph $\mathbf{F}_{k-1}$ obtained by deleting node k from $\mathbf{G}_k$ has the same distribution as $\mathbf{G}_k$. A random graph model is said to be *local* if whenever $S, T \subset [k]$ are disjoint, the random subgraphs of $\mathbf{G}_k$ induced by S and T are independent random variables. A result of Lovász and Szegedy (2006) is that any consistent, local random graph model is equivalent to the distribution on graphs defined by $\mathbb{P}_W$ for some graphon W; the converse is true as well. That is, any such random graph model is equivalent to a graphon.

A particular random graph model is not uniquely defined by a graphon – it is clear from Equation (3.1) that two graphons W_1 and W_2 which are equal almost everywhere (i.e., differ on a set of measure zero) define the same distribution on graphs. In fact, the distribution defined by W is unchanged by "relabelings" of W's nodes. More formally, if Σ is the sigma-algebra of Lebesgue measurable subsets of $[0, 1]$ and μ is the Lebesgue measure, we say that a relabeling function $\varphi : ([0, 1], \Sigma) \to ([0, 1], \Sigma)$ is *measure preserving* if for any measurable set $A \in \Sigma$, $\mu(\varphi^{-1}(A)) = \mu(A)$. We define the relabeled graphon W^φ by $W^\varphi(x, y) = W(\varphi(x), \varphi(y))$. By analogy with finite graphs, we say that graphons W_1 and W_2 are *weakly isomorphic* if they are equivalent up to relabeling, i.e., if there exist measure preserving maps φ_1 and φ_2 such that $W_1^{\varphi_1} = W_2^{\varphi_2}$ almost everywhere. Weak isomorphism is an equivalence relation, and most of the important properties of a graphon in fact belong to



(a) Graphon W.

(b) W^{φ}, weakly isomorphic to W.

(c) An instance of a graph adjacency sampled from W.

Figure 3.2: Example graphons and adjacencies.

its equivalence class. For instance, a powerful result of Lovász (2012) is that two graphons define the same random graph model if and only if they are weakly isomorphic.

Figure 3.2a illustrates a graphon W, while Figure 3.2b depicts a graphon W^{φ} which is weakly isomorphic to W. In particular, W^{φ} is the relabeling of W by the measure preserving transformation $\varphi(x) = 2x \mod 1$. As such, the graphons shown in Figures 3.2a and 3.2b define the same distribution on graphs. Figure 3.2c shows the adjacency matrix A of a graph of size $n = 50$ sampled from the distribution defined by the equivalence class containing W and W^{φ}. Note that it is in principle not possible to determine from A alone which graphon W or W^{φ} it was sampled from, or to what node in W a particular column of A corresponds to.

3.5 The clusters of a graphon

We now identify the cluster structure of a graphon. We will define a graphon's clusters such that they are analogous to the maximally-connected components of a finite graph. A consequence of our definition is that the clusters of equivalent graphons are related in the natural way.

3.5.1 Connectedness

Consider a finite weighted graph. It is natural to cluster the graph into connected components. In fact, because of the weighted edges, we can speak of the clusters of the graph at various levels. More precisely, we say that a set of nodes A is *internally connected* – or, from now on, just *connected* – at level λ if for every pair of nodes in A there is a path between them such that every node along the path is also in A, and the weight of every edge in the path is at least λ. Equivalently, A is *connected* at level λ if and only if for every partitioning of A into disjoint, non-empty sets A_1 and A_2 there is an edge of weight λ or greater between A_1 and A_2. The clusters at level λ are then the largest connected components at level λ.

A graphon is, in a sense, an infinite weighted graph, and we will define the clusters of a graphon using the example above as motivation. Our first step is to adopt a notion of connectedness for graphons. In doing so, we must be careful to make our notion robust to changes to the graphon on a set of zero measure, as such changes do not affect the graph distribution induced by the graphon. We base our definition on that of Janson (2008), who defined what it means for a graphon to be connected as a whole. We extend the definition in (Janson, 2008) to speak of the connectivity of subsets of the graphon's nodes at a particular height. Our definition is analogous to the notion of internal connectedness in finite graphs.

Definition 3.2 (Connectedness)**.** Let W be a graphon, and let $A \subset [0, 1]$ be a set of positive measure. We say that A is *disconnected at level* λ if there exists a measurable $S \subset A$ such that $0 < \mu(S) < \mu(A)$, and $W < \lambda$ almost everywhere on $S \times (A \setminus S)$. Otherwise, we say that A is *connected at level* λ.

Our definition of connectedness is illustrated in Figure 3.3. The figure depicts a piecewise-constant graphon W. Let the set A correspond to the purple bar, let the set S be the green bar, and let the set $A \setminus S$ be the yellow bar. The set A is *disconnected* at any level $\lambda > \lambda_2$ since $W \leq \lambda_2$ almost everywhere on $S \times (A \setminus S)$ (the blue region). The set A is *connected*

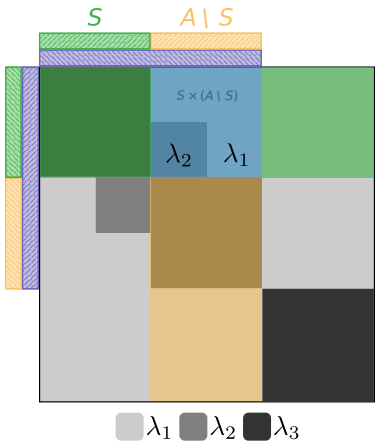


Figure 3.3: Graphon connectedness.

at any level $\lambda \leq \lambda_2$, since for any way of dividing A into non-null S' and $A \setminus S'$, $W \geq \lambda_2$ on some non-null subset of $S' \times (A \setminus S')$.

An important consequence of this definition is that if two overlapping sets are connected, their union must also be connected:

Claim 3.2. *Let W be a graphon, and suppose A and A' are measurable sets with positive measure, and that each is connected at level λ in W. If $\mu(A \cap A') > 0$, then $A \cup A'$ is connected at level λ.*

Proof. Suppose $\mu(A \cap A') > 0$ and that $A \cup A'$ is disconnected at level λ. Then, by definition, there exists a measurable set $S \subset A \cup A'$ such that $0 < \mu(S) < \mu(A \cup A')$ and $W < \lambda$ almost everywhere on $S \times ((A \cup A') \setminus S)$.

It is either the case that $0 < \mu(A \cap S) < \mu(A)$ or $0 < \mu(A' \cap S) < \mu(A')$, as otherwise we would have $\mu(S) = \mu(A \cup A')$. If $0 < \mu(A \cap S) < \mu(A)$, then $A \setminus S$ is of positive measure. Since $W < \lambda$ almost everywhere on $S \times ((A \cup A') \setminus S)$, it follows that $W < \lambda$ almost everywhere on $S \times (A \setminus S)$. This implies that A is disconnected at level λ. Likewise, if $0 < \mu(A' \cap S) < \mu(A')$, it follows that $W < \lambda$ almost everywhere on $S \times (A' \setminus S)$, and hence A' is disconnected at level λ. Both cases lead to contradictions, and so it must be that $A \cup A'$ is connected at level λ. ■

3.5.2 Clusters as connected components

A novel contribution of this chapter is the definition of a graphon cluster, which we now make precise. Motivated by the density setting and Hartigan's notion of high density clusters, we frame our definition in terms of maximally-connected components. We begin by gathering all subsets of $[0, 1]$ which belong to some cluster at level λ. Naturally, if a set is connected at level λ, it is also in a cluster at level λ; for technical reasons[1], we will also say that a set which is connected at all levels $\lambda' < \lambda$ (though perhaps not at λ) is contained in a cluster at level λ, as well. That is, for any λ, the collection $\mathfrak{A}_\lambda$ of sets which are in some cluster at level λ is $\mathfrak{A}_\lambda = \{ A \in \Sigma : \mu(A) > 0$ and A is connected at every level $\lambda' < \lambda\}$. Now suppose $A_1, A_2 \in \mathfrak{A}_\lambda$, and that there is a set $A \in \mathfrak{A}_\lambda$ such that $A \supset A_1 \cup A_2$. Naturally, the cluster to which A belongs should also contain A_1 and A_2, since both are subsets of A. We will therefore consider A_1 and A_2 to be equivalent, in the sense that they are contained in the same cluster at level λ.

More formally, we define a relation $\multimap_\lambda$ on $\mathfrak{A}_\lambda$ by $A_1 \multimap_\lambda A_2 \iff \exists A \in \mathfrak{A}_\lambda$ s.t. $A \supset A_1 \cup A_2$. It can be verified that $\multimap_\lambda$ is in fact an equivalence relation on $\mathfrak{A}_\lambda$:

Claim 3.3. *The relation $\multimap_\lambda$ is an equivalence relation on $\mathfrak{A}_\lambda$.*

Proof. The symmetry and reflexive properties of $\multimap_\lambda$ are clear. We need only prove that $\multimap_\lambda$ is transitive. Suppose $A_1 \multimap_\lambda A_2$ and $A_2 \multimap_\lambda A_3$. By the definition of $\multimap_\lambda$, there exist measurable sets $B_{12}, B_{23} \in \mathfrak{A}_\lambda$ such that $B_{12} \supset A_1 \cup A_2$ and $B_{23} \supset A_2 \cup A_3$. Since both B_{12} and B_{23} contain A_2, a set of positive measure, their intersection is not null. Furthermore, B_{12} and B_{23} are each connected at every level $\lambda' < \lambda$, by virtue of being in $\mathfrak{A}_\lambda$. Hence we can use Claim 3.2 above to conclude that their union $B_{12} \cup B_{23}$ is connected at every level $\lambda' < \lambda$, and is hence an element of $\mathfrak{A}_\lambda$. Since $A_1 \cup A_3 \subset B_{12} \cup B_{23}$, we have $A_1 \multimap_\lambda A_3$. ■

[1] In what follows, we will define the *merge height* of a pair of clusters $\mathscr{C}, \mathscr{C}'$. Making this technical assumption will allow us to say that if $\mathscr{C}$ and $\mathscr{C}'$ merge at some level λ, then there is a cluster $\tilde{\mathscr{C}}$ at level λ which contains both $\mathscr{C}$ and $\mathscr{C}'$.

Each equivalence class $\mathscr{A}$ in the quotient space $\mathfrak{A}_\lambda/\multimap_\lambda$ consists of connected sets which should intuitively be clustered together at level λ. Naturally, we will define the clusters to be the largest elements of each class; in some sense, these are the maximally-connected components at level λ. More precisely, suppose $\mathscr{A}$ is such an equivalence class. It is clear that in general no single member $A \in \mathscr{A}$ can contain all other members of $\mathscr{A}$, since adding a null set (i.e., a set of measure zero) to A results in a larger set A' which is nevertheless still a member of $\mathscr{A}$. However, we can find a member $A^* \in \mathscr{A}$ which contains all but a null set of every other set in $\mathscr{A}$. More formally, we say that A^* is an *essential maximum* of the class $\mathscr{A}$ if $A^* \in \mathscr{A}$ and for every $A \in \mathscr{A}$, $\mu(A \setminus A^*) = 0$; see Definition 3.1. A^* is of course not unique, but it is unique up to a null set; i.e., for any two essential maxima A_1, A_2 of $\mathscr{A}$, we have $\mu(A_1 \triangle A_2) = 0$. We will write the set of essential maxima of $\mathscr{A}$ as $\operatorname{ess\,max} \mathscr{A}$. Naturally, we define clusters as the maximal members of each equivalence class in $\mathfrak{A}_\lambda/\multimap_\lambda$:

Definition 3.3 (Clusters)**.** The set of clusters at level λ in W, written $\mathsf{C}_W(\lambda)$, is defined to be the countable collection $\mathsf{C}_W(\lambda) = \{ \operatorname{ess\,max} \mathscr{A} : \mathscr{A} \in \mathfrak{A}_\lambda/\multimap_\lambda \}$.

We must be careful to ensure that our notion of a cluster is well-defined. That this is so is a consequence of the definition of connectedness in Definition 3.2 above.

Claim 3.4. *Let $\mathscr{C}$ be an equivalence class in $\mathfrak{A}_\lambda/\multimap_\lambda$. Then $\operatorname{ess\,max} \mathscr{C}$ is well-defined and non-empty.*

Proof. We will invoke Claim 3.1 to show that $\operatorname{ess\,max} \mathscr{C}$ has the desired properties. To do so, we need only show that $\mathscr{C}$ is closed under countable unions. Let $\mathscr{F} \subset \mathscr{C}$ be a countable subset of $\mathscr{C}$, and let $F = \bigcup \mathscr{F}$. We will show that F is connected at every level $\lambda' < \lambda$ and is thus contained in $\mathscr{F}$ by its definition.

Suppose F is disconnected at some level $\lambda' < \lambda$. Then there exists a set $S \subset F$ such that $0 < \mu(S) < \mu(F)$ and $W < \lambda'$ almost everywhere on $S \times (F \setminus S)$. Now, there must exist sets $F_1, F_2 \in \mathscr{F}$ such that $\mu(S \cap F_1) > 0$ and $\mu((F \setminus S) \cap F_2) > 0$. Let $F_{12} = F_1 \cup F_2$. Note that $\mathscr{F}$ is closed under finite union by the definition of $\multimap_\lambda$, and so $F_{12} \in \mathscr{F}$, meaning that

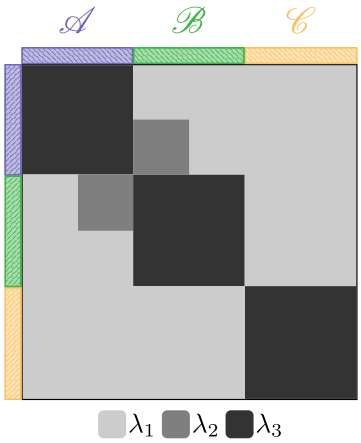


Figure 3.4: The clusters of a graphon at level λ_3.

F_{12} is connected at λ'. Furthermore, $F_{12} \subset F$, so that $(F_{12} \cap S) \cup (F_{12} \cap (F \setminus S)) = F_{12}$. But by assumption $W < \lambda'$ almost everywhere on $S \times (F \setminus S)$, and so in particular $W < \lambda'$ almost everywhere on $(F_{12} \cap S) \times (F_{12} \cap (F \setminus S))$. But this implies that F_{12} is disconnected at level λ', which is a contradiction. It must therefore be the case that F is connected at every $\lambda' < \lambda$, and so $F \in \mathfrak{A}_\lambda$.

It is clearly true that for all $F' \in \mathscr{F}$, $F' \multimap_\lambda F$, since $F = F \cup F'$. We therefore have that $F \in \mathscr{F}$. ■

Note that a cluster $\mathscr{C}$ of a graphon is not a subset of the unit interval per se, but rather an *equivalence class* of subsets which differ only by null sets. It is often possible to treat clusters as sets rather than equivalence classes, and we may write $\mu(\mathscr{C})$, $\mathscr{C} \cup \mathscr{C}'$, etc., without ambiguity. In addition, if $\varphi : [0,1] \to [0,1]$ is a measure preserving transformation, then $\varphi^{-1}(\mathscr{C})$ is well-defined.

For a concrete example of our notion of a cluster, consider the graphon W depicted in Figure 3.4. A, B, and C represent sets of the graphon's nodes. By our definitions there are three clusters at level λ_3: $\mathscr{A}$, $\mathscr{B}$, and $\mathscr{C}$. Clusters $\mathscr{A}$ and $\mathscr{B}$ merge into a cluster $\mathscr{A} \cup \mathscr{B}$ at level λ_2, while $\mathscr{C}$ remains a separate cluster. Everything is joined into a cluster $\mathscr{A} \cup \mathscr{B} \cup \mathscr{C}$ at level λ_1.

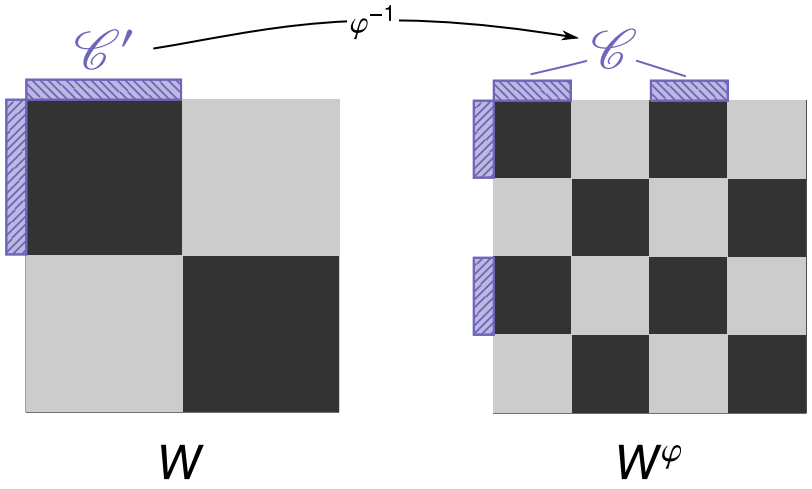


Figure 3.5: Clusters of weakly-isomorphic graphons

3.5.3 Clusters of weakly-isomorphic graphons

We have taken care to define the clusters of a graphon in such a way as to be robust to changes of measure zero to the graphon itself. In fact, clusters are also robust to measure preserving transformations:

Theorem 3.1. *Let W be a graphon and φ a measure preserving transformation. Then $\mathscr{C}$ is a cluster of W^φ at level λ if and only if there exists a cluster $\mathscr{C}'$ of W at level λ such that $\mathscr{C} = \varphi^{-1}(\mathscr{C}')$.*

Figure 3.5 illustrates the above theorem. W^φ is the relabeling of the graphon W by the measure preserving transformation $\varphi(x) = 2x \mod 1$. $\mathscr{C}'$ is a cluster of W at some level λ. It follows from Theorem 3.1 that $\varphi^{-1}(\mathscr{C})$ is a cluster of W^φ at that same level. Moreover, the theorem states that this correspondence is in fact a bijection.

The proof of Theorem 3.1 will comprise the remainder of the section. It is made non-trivial by the fact that a measure preserving transformation is in general not injective. For instance, $\varphi(x) = 2x \mod 1$ defines a measure preserving transformation, but is not an injection. Even worse, it is possible for a measure preserving transformation to map a set

of zero measure to a set of positive measure – it is only the measure of the *preimage* which must be preserved.

Twins and non-separating sets.

We will mitigate the fact that φ may not be injective by working whenever possible with sets whose image is necessarily stable under even non-injective measure preserving transformations, in the sense that $\varphi^{-1}(\varphi(A)) = A$. We will show that such stability is a property of sets which contain all of their so-called *twin* points, defined as follows (Lovász, 2012):

Definition 3.4 (Twin points)**.** Two points x and x' are *twins* in W if $W(x, y) = W(x', y)$ for almost every $y \in [0, 1]$. We say that a set A *separates* twins if there exist twins x and x' such that $x \in A$ and $x' \notin A$. The relation of being twins is an equivalence relation on $[0, 1]$.

We will define a probability space on the equivalence classes of the twin relation as follows – see the book by Lovász (2012) for the full construction:

Definition 3.5. Let W be a graphon. The *twin measure space* $(\Omega_W, \Sigma_W, \mu_W)$ is defined as follows. Let Ω_W be the set of equivalence classes under the twin relation in W, and let $\psi_W(x)$ denote the equivalence class in Ω_W containing x. Note that $\psi_W^{-1}(\psi_W(x))$ is simply the set of twins of x. If Σ is the sigma-algebra of Lebesgue measurable subsets of $[0, 1]$, create a new sigma-algebra by defining

$$\Sigma_W = \{\psi_W(X) : X \in \Sigma, X \text{ does not separate twins in } W\}.$$

Observe that for any $A \in \Sigma_W$, $\psi_W^{-1}(A)$ does not separate twins. Furthermore, we define the measure $\mu_W(A) = \mu(\psi_W^{-1}(A))$ for $A \in \Sigma_W$. It can be shown that, with this measure, ψ_W is measure preserving.

We note in passing that the random graph model defined by any graphon W can also be represented as a Σ_W-measurable function $W_T : \Omega_W \times \Omega_W \to [0, 1]$ defined on the

probability space $(\Omega_W, \Sigma_W, \mu_W)$, as is shown by Lovász (2012). W_T is called a "twin-free" graphon, since no two points in Ω_W are twins in W_T. In this representation, two twin-free graphons are weakly isomorphic if there exists a measure preserving *bijection* relating them. Our definitions of connectedness, clusters, mergeons, etc. can be formulated for twin-free graphons with minor modifications, and the existence of the measure preserving bijection between twin-free graphons means that clusters transfer trivially between weakly isomorphic graphons. In a sense, the twin-free setting is a more natural one for the considerations of the current section; We leave a more in-depth investigation of this direction to future work.

We first observe some useful properties of the map ψ_W.

Claim 3.5. *Suppose $A \subset [0,1]$ does not separate twins in W. Then $\psi_W^{-1}(\psi_W(A)) = A$ and A is Σ-measurable.*

Proof. It is clear that $A \subset \psi_W^{-1}(\psi_W(A))$. Now let $x \in \psi_W^{-1}(\psi_W(A))$. Then there exists a $y \in A$ such that $\psi_W(x) = \psi_W(y)$. But then x and y are twins in W^φ. Since A does not separate twins, $x \in A$, proving that $A = \psi_W^{-1}(\psi_W(A))$.

Now we prove that A is Σ-measurable. ψ_W is a measurable function, and so the inverse image of any Σ_W-measurable set is Σ-measurable. We have that $\psi_W(A)$ is Σ_W-measurable, since A does not separate twins. Hence $\psi_W^{-1}(\psi_W(A)) = A$ is Σ-measurable. ■

Claim 3.6. *Let W be a graphon and let $A \subset [0,1]$. Then $\psi_W^{-1}(\psi_W(A))$ is Σ-measurable.*

Proof. It is clear that $\psi_W^{-1}(\psi_W(A))$ does not separate twins in W. Hence it is Σ-measurable by the previous claim. ■

Sets which do not separate twins are particularly nice, as the following formalizes:

Lemma 3.1. *Let W be a graphon and φ a measure preserving transformation. Suppose A does not separate twins in W^φ. Then*

1. *$\varphi^{-1}(\varphi(A)) = A$, and*

2. $\mu(\varphi(A)) = \mu(A)$.

Proof. For the first claim, we know that $A \subseteq \varphi^{-1}(\varphi(A))$. Now we show the other inclusion. Let $x \in \varphi^{-1}(\varphi(A))$. Then there exists an x' in $\varphi(A)$ such that $\varphi(x) = \varphi(x')$. But then x and x' are twins, such that x and x' are both in A. Hence $x \in A$, proving the claim. The second claim follows immediately since φ is measure preserving. That is, $\mu(\varphi^{-1}(\varphi(A))) = \mu(\varphi(A))$, but since $\varphi^{-1}(\varphi(A)) = A$, $\mu(\varphi(A)) = \mu(A)$. ■

Non-separating cluster representatives.

A graphon cluster $\mathscr{C}$ is an equivalence class of sets modulo null sets. When working with clusters, it is often useful to take a particular member of the equivalence class as a representative. We will now show that we may always find a cluster representative which does not separate twins, and is therefore "nice" in the sense made precise above.

To begin, consider an arbitrary measurable set C which is not necessarily a cluster representative. In general C may separate twins in W, but we can always find a larger set which contains almost all of C, but which does not separate twins. In fact, there may be many such sets; we call the collection of them which has minimal measure the *family* of C:

Definition 3.6. Let W be a graphon and let $(\Omega_W, \Sigma_W, \mu_W)$ be the corresponding twin measure space for W. For any Σ-measurable set C, construct the collection

$$\mathcal{F}_C = \{A \in \Sigma_W : \mu(C \setminus \psi_W^{-1}(A)) = 0\}.$$

Observe that for any $A \in \mathcal{F}_C$, $\psi_W^{-1}(A)$ contains almost all of C and does not separate twins in W. Let $\operatorname{ess\,min} \mathcal{F}_C$ be as in Definition 3.1. We define the *family* of C, written $\mathsf{Fam}_W C$, as

$$\mathsf{Fam}_W C = \{\psi_W^{-1}(X) : X \in \operatorname{ess\,min} \mathcal{F}_C\}.$$

It is clear that $\mathsf{Fam}_W C$ cannot be empty, as $\psi_W^{-1}(\Omega_W)$ must contain almost all of C. To be rigorous, we must argue that $\mathcal{F}_C$ is closed under countable intersections so that it has a well-defined set of essential minima. To see this, let $\mathcal{F}$ be any countable subset of $\mathcal{F}_C$. Define $D = \bigcap \mathcal{F}$. Then $D \in \Sigma_W$ since it is a sigma-algebra, and we have

$$\begin{aligned}
\mu\left(C \setminus \psi_W^{-1}(D)\right) &= \mu\left(C \setminus \psi_W^{-1}\left(\bigcap \mathcal{F}\right)\right), \\
&= \mu\left(C \setminus \bigcap_{F \in \mathcal{F}} \psi_W^{-1}(F)\right), \\
&= \mu\left(\bigcup_{F \in \mathcal{F}} C \setminus F\right), \\
&= 0,
\end{aligned}$$

where the last step follows because each F has the property that $C \setminus F$ is a null set, and the union of countably many null sets is null. Hence $D \in \mathcal{F}_C$.

In general, if $A \in \mathsf{Fam}_W C$, it may be the case that A is much larger than C in the sense that $\mu(A \setminus C) > 0$. However, if C is in fact a graphon cluster we can show that A and C differ only by null sets. As a consequence, any element of $\mathsf{Fam}_W C$ is a cluster representative which does not separate twins, as the following lemma makes precise:

Lemma 3.2. *Let W be a graphon and suppose $\mathscr{C}$ is a cluster at level λ in W. Let C be an arbitrary representative of the cluster $\mathscr{C}$. Let $\bar{C} \in \mathsf{Fam}_W C$. Then $\mu(C \triangle \bar{C}) = 0$, and hence $\bar{C}$ is a representative of the cluster $\mathscr{C}$ which does not separate twins.*

Proof. We know that $C \setminus \bar{C}$ is a null set since $\bar{C}$ must contain all but a null set of C by the definition of Fam_W, so we need only show that $\bar{C} \setminus C$ is null. Suppose otherwise. That is, let $R = \bar{C} \setminus C$ and suppose $\mu(R) > 0$. Let A be any subset of R with positive measure. There are two cases: (1) For some $\lambda' < \lambda$, $W < \lambda'$ almost everywhere on $A \times (\bar{C} \setminus A)$, or (2) for every $\lambda' < \lambda$, $W \geq \lambda'$ on some subset of $A \times (\bar{C} \setminus A)$ of positive measure.

Suppose case (1) holds for some λ'. Then for almost all $a \in A$ it is true that $W(a, y) < \lambda'$ for almost every $y \in \bar{C} \setminus A$. That is, let

$$\hat{A} = \{a \in A : W(a, y) < \lambda' \text{ for almost every } y \in \bar{C} \setminus A\}.$$

Then $\mu(\hat{A}) = \mu(A)$ and $W < \lambda$ almost everywhere on $\hat{A} \times (\bar{C} \setminus \hat{A})$. Define $\bar{A} = \psi_W^{-1}(\psi_W(\hat{A}))$. There are two subcases: Either (1a) $\bar{A} \cap C$ is null, or (1b) it is of positive measure.

Consider the first subcase. Define $D = \psi_W(\bar{C}) \setminus \psi_W(\bar{A})$. We will show that $\psi_W^{-1}(D)$ contains C except for a set of zero measure, and so $\psi_W^{-1}(D) \in \mathsf{Fam}_W\, C$. But as we will see, $\mu(\psi_W^{-1}(D)) < \mu(\bar{C})$, which cannot be. We have

$$\begin{aligned}
\psi_W^{-1}(D) &= \psi_W^{-1}(\psi_W(\bar{C}) \setminus \psi_W(\bar{A})) \\
&= \psi_W^{-1}(\psi_W(\bar{C})) \setminus \psi_W^{-1}(\psi_W(\bar{A})) \\
&= \bar{C} \setminus \bar{A}
\end{aligned}$$

where the last step follows since $\bar{C}$ and $\bar{A}$ do not separate twins. Therefore,

$$\begin{aligned}
C \cap \psi_W^{-1}(D) &= C \cap (\bar{C} \setminus \bar{A}) \\
&= (C \cap \bar{C}) \cup (C \cap \bar{A})
\end{aligned}$$

But $C \cap \bar{A}$ is a null set, so $\mu(C \cap \psi_W^{-1}(D)) = \mu(C \cap \bar{C}) = \mu(C)$. This implies that $\mu(C \setminus \psi_W^{-1}(D)) = 0$, and hence $\psi_W^{-1}(D) \in \mathsf{Fam}_W\, C$. But $\mu_W(D) = \mu_W(\psi_W(\bar{C}) \setminus \psi_W(\bar{A}))$, and $\psi_W(\bar{A}) \subset \psi_W(\bar{C})$ with $\mu_W(\psi_W(\bar{A})) = \mu(\bar{A}) > 0$. Therefore, $\mu_W(D) < \mu_W(\psi_W(\bar{C}))$, and so $\mu(\psi_W^{-1}(D)) < \mu(C)$. This cannot be, since all elements of $\mathsf{Fam}_W\, C$ differ only by null sets. Hence it cannot be that $\bar{A} \cap C$ is null.

Suppose case (1b) holds, then. That is, suppose $\bar{A} \cap C$ is not null. Then for every $x \in \bar{A} \cap C$ it is true that $W(x, y) < \lambda'$ for almost all $y \in \bar{C} \setminus A$. In particular, since $C \setminus (A \cap C) \subset \bar{C} \setminus A$, we have that $W < \lambda'$ almost everywhere on $(\bar{A} \cap C) \times (C \setminus \bar{A})$. This

means that C is disconnected at level λ', which violates the assumption that C is a cluster at $\lambda > \lambda'$.

Both subcases lead to contradictions, and so (1) cannot hold. Therefore, it must be that case (2) holds: For every $\lambda' < \lambda$, $W \geq \lambda'$ on some subset of $A \times (\bar{C} \setminus A)$. Furthermore, this must hold for arbitrary $A \subset R$ with positive measure. This implies that $\bar{C}$ is connected at every level $\lambda' < \lambda$, and hence part of a cluster at level λ. To see this, let $S, T \subset \bar{C}$ such that S has positive measure and $S \cup T = \bar{C}$. Without loss of generality, assume $A \cap S$ is not null – if it is, swap S and T. Then $T \cap (\bar{C} \setminus A)$ is not null. Therefore $W \geq \lambda$ on some subset of $S \times T$ with positive measure – namely, $(S \cap A) \times (T \cap (\bar{C} \setminus A))$. Since this holds for arbitrary S and T, $\bar{C}$ is connected.

Therefore, both cases lead to contradictions under the assumption that $\mu(R) > 0$. Hence $\mu(R) = 0$, and $\mu(C \triangle \bar{C}) = 0$. $\bar{C}$ does not separate twins since $\bar{C} \in \mathsf{Fam}_W(C)$. Furthermore, $\mu(C \triangle \bar{C}) = 0$ implies that $\bar{C}$ is a representative of $\mathscr{C}$. Therefore the claim is proven. ■

The above result allows us to conclude that any cluster $\mathscr{C}$ of a graphon W has a representative C such that $\varphi(\varphi^{-1}(C)) = C$, as the following key lemma shows.

Lemma 3.3. *Let W be a graphon and φ a measure preserving transformation. Suppose $\mathscr{C}$ is a cluster of W. Then there exists a representative C of $\mathscr{C}$ such that $\varphi(\varphi^{-1}(C)) = C$.*

Proof. In the following, let $\bar{C}$ be a representative of $\mathscr{C}$ which does not separate twins; the existence of such a representative follows from Lemma 3.2.

$\varphi^{-1}(\bar{C})$ does not separate twins in W^{φ}, and so by Lemma 3.1, $\mu(\varphi(\varphi^{-1}(\bar{C}))) = \mu(\bar{C})$. But $\bar{C} \supset \varphi(\varphi^{-1}(\bar{C}))$, such that $\bar{C} \triangle \varphi(\varphi^{-1}(\bar{C})) = 0$. Hence $\varphi(\varphi^{-1}(\bar{C}))$ is a representative of the cluster $\mathscr{C}$. Furthermore, $\varphi^{-1}(\varphi(\varphi^{-1}(\bar{C}))) = \varphi^{-1}(\bar{C})$, such that, defining $C = \varphi(\varphi^{-1}(\bar{C}))$, we have $\varphi(\varphi^{-1}(C)) = C$, as claimed. ■

Connectedness under measure-preserving transformations.

In order to prove that the clusters of weakly-isomorphic graphons are related in the natural way, we must analyze how the connectedness of a set is affected by measure preserving transformations. Recall that a set C is disconnected at level λ in a graphon W if there exists a measurable set $A \subset C$ such that $\mu(A) > 0$ and $W < \lambda$ almost everywhere on $A \times (C \setminus A)$. As such, proving that a set C is disconnected amounts to finding a separating set A.

We would like to say that if a set C is disconnected in W, then the corresponding set C' is disconnected in a weakly-isomorphic graphon W'. To do so, we will show that if A is a separating set for C in W, then the corresponding set A' is a separating set of C' in W'. As before, we will prefer to work with sets which do not separate twins. The following claim shows that if C is a disconnected set that does not separate twins, we can always find a separating set A which also does not separate twins.

Claim 3.7. *Let W be a graphon and suppose that C is a set of positive measure that does not separate twins. If C is disconnected at level λ in W, then either $W < \lambda$ almost everywhere on $C \times C$, or there exists a set $\bar{A} \subset C$ such that $\bar{A}$ does not separate twins, $0 < \mu(\bar{A}) < \mu(C)$, and $W < \lambda$ almost everywhere on $\bar{A} \times (C \setminus \bar{A})$.*

Proof. Since C is disconnected at level λ, there exists a subset $S \subset C$ such that $0 < \mu(S) < \mu(C)$ and $W < \lambda$ almost everywhere on $S \times (C \setminus S)$. Define

$$\hat{S} = \{x \in S : W(x, y) < \lambda \text{ for almost every } y \in C \setminus S\}.$$

Since $W < \lambda$ almost everywhere on $S \times (C \setminus S)$, it must be that $\mu(\hat{S}) = \mu(S)$; This is an application of Fubini's theorem. Let $\bar{S} = \psi_W^{-1}(\psi_W(\hat{S}))$. It follows that $\bar{S} \subset C$, and for every $x \in \bar{S}$, $W(x, y) < \lambda$ for almost every $y \in C \setminus S$. Furthermore, $\bar{S}$ does not separate twins, and $\bar{S}$ contains $\hat{S}$ – which is S, less a null set – so $\mu(S \setminus \bar{S}) = 0$.

There are two cases: $\mu(\bar{S}) < \mu(C)$, or $\mu(\bar{S}) = \mu(C)$. Suppose the first case holds. Then, since $\mu((C \setminus S) \setminus (C \setminus \bar{S})) = 0$, we have that for every $x \in \bar{S}$, $W(x, y) < \lambda$ for almost every $y \in C \setminus \bar{S}$. Therefore, $W < \lambda$ almost everywhere on $\bar{S} \times (C \setminus \bar{S})$. This proves the claim for the first case, as we may take $\bar{A} = \bar{S}$.

Now suppose $\mu(\bar{S}) = C$, which is to say that $\bar{S}$ differs from C by a null set. Since $W < \lambda$ almost everywhere on $\bar{S} \times (C \setminus S)$, it follows that $W < \lambda$ almost everywhere on $C \times (C \setminus S)$. By symmetry of W, we have $W < \lambda$ almost everywhere on $(C \setminus S) \times C$. This means that $W < \lambda$ almost everywhere on $(C \times C) \setminus (S \times S)$.

Let $T = C \setminus S$. Then $W < \lambda$ almost everywhere on $T \times C = (C \setminus S) \times C$. Define

$$\hat{T} = \{x \in T : W(x, y) < \lambda \text{ for almost every } y \in C \}.$$

Let $\bar{T} = \psi_W^{-1}(\psi_W(\hat{T}))$. Then, by a similar argument used above for $\bar{S}$, $\mu(T \setminus \bar{T}) = 0$, $\bar{T}$ does not separate twins, and $W < \lambda$ almost everywhere on $\bar{T} \times C$.

There are two subcases: First, it may be that $\mu(\bar{T}) = \mu(C)$. If so, then $W < \lambda$ almost everywhere on $C \times C$, which proves the claim. Second, it may be that $\mu(\bar{T}) < \mu(C)$. In this case, we have $W < \lambda$ almost everywhere on $\bar{T} \times (C \setminus \bar{T})$, and so taking $\bar{A} = \bar{T}$ proves the claim. ■

We may now prove the key lemmas to Theorem 3.1. Together, these show that if $\mathscr{C}$ is a cluster of a graphon W at level λ, then its corresponding set in a weakly-isomorphic graphon is connected at every level $\lambda' < \lambda$, and is therefore part of some cluster at level λ.

Lemma 3.4. *Let W be a graphon and φ a measure preserving transformation. If $\mathscr{C}$ is a cluster at level λ in W, then $\varphi^{-1}(\mathscr{C})$ is connected at every level $\lambda' < \lambda$ in W^{φ}.*

Proof. For simplicity, we will work with an representative C of the cluster $\mathscr{C}$. As Lemma 3.3 shows, we may take C to be a representative such that $\varphi(\varphi^{-1}(C)) = C$.

Suppose for a contradiction that $\varphi^{-1}(C)$ is disconnected in W^{φ} at some level $\lambda' < \lambda$. Then by Claim 3.7 either $W^{\varphi} < \lambda'$ almost everywhere on $\varphi^{-1}(C) \times \varphi^{-1}(C)$, or there exists

a set $\bar{A} \subset \varphi^{-1}(C)$ such that $0 < \mu(\bar{A}) < \mu(\varphi^{-1}(C))$, $W^{\varphi} < \lambda'$ almost everywhere on $\bar{A} \times (\varphi^{-1}(C) \setminus \bar{A})$, and $\bar{A}$ does not separate twins.

In the first case, $W^{\varphi} < \lambda'$ almost everywhere on $\varphi^{-1}(C) \times \varphi^{-1}(C)$ implies that $W < \lambda'$ almost everywhere on $C \times C$, which contradicts the fact that C is the representative of a cluster at level λ' in W.

Suppose the second case, then, where $W^{\varphi} < \lambda'$ almost everywhere on $\bar{A} \times (\varphi^{-1}(C) \setminus \bar{A})$. Then $W < \lambda'$ almost everywhere on $\varphi(\bar{A}) \times \varphi(\varphi^{-1}(C) \setminus \bar{A})$. We now claim that $\varphi(\varphi^{-1}(C) \setminus \bar{A}) = \varphi(\varphi^{-1}(C)) \setminus \varphi(\bar{A}) = C \setminus \bar{A}$. To see this, note that $\varphi(\varphi^{-1}(C) \setminus \bar{A}) \supset \varphi^{-1}(\varphi(C)) \setminus \varphi(\bar{A})$. However, we have chosen C to be a representative such that $\varphi(\varphi^{-1}(C)) = C$, and so we obtain

$$\varphi(\varphi^{-1}(C) \setminus \bar{A}) \supset C \setminus \bar{A}.$$

On the other hand, suppose $y \in \varphi(\varphi^{-1}(C) \setminus \bar{A})$. This means that there is some $x \in \varphi^{-1}(C) \setminus \bar{A}$ such that $\varphi(x) = y$. But $\varphi^{-1}(C) \setminus \bar{A}$ does not separate twins, so there cannot be an $x' \in \bar{A}$ such that $\varphi(x') = \varphi(x) = y$. Therefore, $y \in \varphi(\varphi^{-1}(C) \setminus \bar{A})$ if $y \in C$ and there is no $a \in \bar{A}$ such that $\varphi(a) = y$. That is, $\varphi(\varphi^{-1}(C) \setminus \bar{A}) \subset C \setminus \bar{A}$. Hence $\varphi(\varphi^{-1}(C) \setminus \bar{A}) = C \setminus \varphi(\bar{A})$.

Therefore $W < \lambda$ almost everywhere on $\varphi(\bar{A}) \times (C \setminus \varphi(\bar{A}))$. Since $\mu(\varphi(\bar{A})) = \mu(\bar{A}) < \mu(C)$ by Lemma 3.1, this implies that C is disconnected at level λ' in W. Hence C is not the representative of a cluster at level λ, and so we have derived a contradiction.

Both cases lead to contradictions, and so it must be that $\varphi^{-1}(C)$ is connected in W^{φ} at every level $\lambda' < \lambda$. ■

Lemma 3.5. *Let W be a graphon and φ be a measure preserving transformation. Suppose $\mathscr{C}$ is a cluster of W^{φ} at level λ. Let $C \in \mathsf{Fam}_W \mathscr{C}$. Then $\varphi(C)$ is connected at every level $\lambda' < \lambda$ in W.*

Proof. Suppose for a contradiction that $\varphi(C)$ is not connected at some level $\lambda' < \lambda$ in W. Then there exists a set $S \subset \varphi(C)$ such that $0 < \mu(S) < \mu(\varphi(C))$ and $W < \lambda'$ almost everywhere on $S \times (\varphi(C) \setminus S)$. Hence $W^{\varphi} < \lambda'$ almost everywhere on $\varphi^{-1}(S) \times$

$\varphi^{-1}(\varphi(C) \setminus S) = \varphi^{-1}(S) \times (\varphi^{-1}(\varphi(C)) \setminus \varphi^{-1}(S))$. Since C does not separate twins in W^{φ}, we have by Lemma 3.1 that $\varphi^{-1}(\varphi(C)) = C$, and so $W^{\varphi} < \lambda'$ almost everywhere on $\varphi^{-1}(S) \times (C \setminus \varphi^{-1}(S))$.

Consider $\varphi^{-1}(S)$. We have $C = \varphi^{-1}(\varphi(C))$, and since $S \subset \varphi(C)$, it follows that $\varphi^{-1}(S) \subset C$. Moreover, $\mu(\varphi^{-1}(S)) = \mu(S)$, since φ is measure preserving, and $0 < \mu(S) < \mu(\varphi(C)) = \mu(C)$, where the last equality comes from Lemma 3.1. Hence C is disconnected at level λ' in W. This contradicts the fact that C is a representative of a cluster at level λ in W. Hence it must be that $\varphi(C)$ is connected at every level $\lambda' < \lambda$ in W. ■

Proof of Theorem 3.1.

The two previous claims are sufficient to prove the main result of this section, restated below:

Theorem 3.1. *Let W be a graphon and φ a measure preserving transformation. Then $\mathscr{C}$ is a cluster of W^{φ} at level λ if and only if there exists a cluster $\mathscr{C}'$ of W at level λ such that $\mathscr{C} = \varphi^{-1}(\mathscr{C}')$.*

Proof. Suppose $\mathscr{C}$ is a cluster of W at level λ and let C be a representative of $\mathscr{C}$. Then according to Lemma 3.4, $\varphi^{-1}(C)$ is connected at every level $\lambda' < \lambda$ in W^{φ}. Hence, by the definition of a graphon cluster, there exists a cluster $\mathscr{C}'$ at level λ in W^{φ} which contains $\varphi^{-1}(C)$. Then by Lemma 3.5, there is a representative C' of $\mathscr{C}'$ such that C' does not separate twins and $\varphi(C')$ is connected at every level $\lambda' < \lambda$ in W. Hence there is a cluster $\mathscr{C}''$ of W at level λ such that $\mathscr{C}''$ contains $\varphi(C')$. However, it must be that $\mathscr{C}'' = \mathscr{C}$. To see this, note that we have $\varphi^{-1}(C \cap \varphi(C')) = \varphi^{-1}(C) \cap \varphi^{-1}(\varphi(C')) = \varphi^{-1}(C) \cap C'$, where we used Lemma 3.1 in replacing $\varphi^{-1}(\varphi(C'))$ with C'. Since φ is measure preserving, it follows that $\mu(C \cap \varphi(C')) = \mu(\varphi^{-1}(C) \cap C')$, but $C' \triangle \varphi^{-1}(C)$ is a null set and so $\mu(C \cap \varphi(C')) = \mu(C)$. Thus $\mu(\mathscr{C}') = \mu(\varphi^{-1}(C))$, and so $\varphi^{-1}(C)$ is a representative of the cluster $\mathscr{C}'$. Hence $\varphi^{-1}(\mathscr{C})$ is a cluster at level λ of W^{φ}.

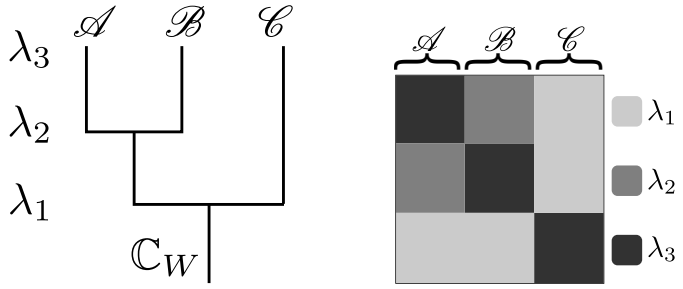
Now suppose $\mathscr{C}$ is a cluster of W^φ at level λ and let C be a representative of $\mathscr{C}$ such that $C \in \mathsf{Fam}_W(\mathscr{C})$. Then according to Lemma 3.5, $\varphi(C)$ is connected at every level $\lambda' < \lambda$ in W, and hence there exists a cluster $\mathscr{C}'$ in W at level λ which contains $\varphi(C)$. By the previous argument, $\varphi^{-1}(\mathscr{C}')$ is a cluster of W^φ at level λ. Since $C \in \mathsf{Fam}_W \mathscr{C}$, C does not separate twins in W^φ, and so $\varphi^{-1}(\varphi(C)) = C$, and thus C is contained in $\varphi^{-1}(\mathscr{C}')$. Since C is a cluster representative, and thus maximal, it must be that $\varphi^{-1}(\mathscr{C}') = \mathscr{C}$. ■

3.6 Mergeons

The set of all clusters of a graphon at any level has hierarchical structure in the sense that given any pair of distinct clusters $\mathscr{C}_1$ and $\mathscr{C}_2$, either one is "essentially" contained within the other, i.e., $\mathscr{C}_1 \subset \mathscr{C}_2$, or $\mathscr{C}_2 \subset \mathscr{C}_1$, or they are "essentially" disjoint, i.e., $\mu(\mathscr{C}_1 \cap \mathscr{C}_2) = 0$, as is proven by Claim 3.2 on page 49. Because of this hierarchical structure, we call the set C_W of all clusters from any level of the graphon W the *graphon cluster tree* of W. It is this tree that we hope to recover by applying a graph clustering algorithm to a graph sampled from W.

We may naturally speak of the height at which pairs of distinct clusters merge in the cluster tree. For instance, let $\mathscr{C}_1$ and $\mathscr{C}_2$ be distinct clusters of C_W. We say that the *merge height* of $\mathscr{C}_1$ and $\mathscr{C}_2$ is the level λ at which they are joined into a single cluster, i.e., $\max\{\lambda : \mathscr{C}_1 \cup \mathscr{C}_2 \in \mathsf{C}_W(\lambda)\}$. However, while the merge height of clusters is well-defined, the merge height of individual points is not. This is because the cluster tree is not a collection of sets, but rather a collection of equivalence classes of sets, and so a point does not belong to any one cluster more than any other. Note that this is distinct from the classical density case considered in Hartigan (1981), Chaudhuri and Dasgupta (2010), Abbe et al. (2015), and the previous chapter, where the merge height of any pair of points is well-defined.

Nevertheless, consider a measurable function $M : [0,1]^2 \to [0,1]$ which assigns a merge height to every pair of points. While the value of M on any given pair is arbitrary, the



(a) Cluster tree C_W of W.

(b) Mergeon M of C_W.

Figure 3.6: A graphon cluster tree and its mergeon.

value of M on sets of positive measure is constrained. Intuitively, if $\mathscr{C}$ is a cluster at level λ, then we must have $M \geq \lambda$ almost everywhere on $\mathscr{C} \times \mathscr{C}$. If M satisfies this constraint for every cluster $\mathscr{C}$ we call M a *mergeon* for C_W, as it is a graphon which determines a particular choice for the merge heights of every pair of points in $[0,1]$. More formally:

Definition 3.7 (Mergeon)**.** A *mergeon* of a cluster tree C is a graphon M such that for all $\lambda \in [0,1]$,

$$M^{-1}[\lambda,1] \;\triangle \bigcup_{\mathscr{C} \in \mathsf{C}_W(\lambda)} \rho(\mathscr{C}) \times \rho(\mathscr{C})$$

is a null set, where $M^{-1}[\lambda,1] = \{(x,y) : M(x,y) \geq \lambda\}$, $\triangle$ is the symmetric difference operator, and ρ is an arbitrary section map[2]. Equivalently, a *mergeon* of C is a graphon M such that for all $\lambda \in [0,1]$,

$$[M^{-1}[\lambda,1]]_\emptyset = \bigcup_{\mathscr{C} \in \mathsf{C}_W(\lambda)} \mathscr{C} \times \mathscr{C},$$

where $[M^{-1}[\lambda,1]]_\emptyset$ is the equivalence class of measurable subsets of $[0,1] \times [0,1]$ modulo null sets which contains $M^{-1}[\lambda,1]$.

[2]Recall from Section 3.3 that a *section map* ρ is a function which returns an element of an equivalence class. Since a cluster $\mathscr{C}$ is an equivalence class of measurable sets modulo null sets, $\rho(\mathscr{C})$ is a measurable subset of $[0,1]$.

An example of a mergeon and the cluster tree it represents is shown in Figure 3.6. In fact, the cluster tree depicted is that of the graphon W from Figure 3.2a. The mergeon encodes the height at which clusters $\mathscr{A}$, $\mathscr{B}$, and $\mathscr{C}$ merge. In particular, the fact that $M = \lambda_2$ everywhere on $\mathscr{A} \times \mathscr{B}$ represents the merging of $\mathscr{A}$ and $\mathscr{B}$ at level λ_2 in W.

3.6.1 Properties

It is clear that in general there is no unique mergeon representing a graphon cluster tree, however, the above definition implies that two mergeons representing the same cluster tree are equal almost everywhere. As such there is a unique *equivalence class* of mergeons corresponding to a graphon cluster tree.

Moreover, as the mergeon itself is a graphon, it has a corresponding cluster tree. The cluster tree of the mergeon satisfies the following property:

Theorem 3.2. *Let C be a cluster tree, and suppose M is a mergeon representing C. Then $\mathscr{C} \in \mathsf{C}(\lambda)$ if and only if $\mathscr{C}$ is a cluster in M at level λ. In other words, the cluster tree C_M of M is exactly C.*

To prove this theorem, we need the following technical results:

Lemma 3.6. *Let W be a graphon and suppose M is a mergeon of W. Suppose A is connected at level λ in M. Then A is contained in some cluster C in W at level λ.*

Proof. First, it must be the case that $\mu\left(A \setminus \bigcup \mathsf{C}_W(\lambda)\right) = 0$. Suppose not. Let $R = A \setminus \bigcup \mathsf{C}_W(\lambda)$. Since A is connected, it follows that there is a subset $Q \subset R \times (A \setminus R)$ of positive measure such that $M \geq \lambda$ on Q. We then have that

$$Q \cap M^{-1}[\lambda, 1] = Q \cap \bigcup_{C \in \mathsf{C}_W(\lambda)} C \times C$$

is not null. Since $\mathsf{C}_W(\lambda)$ is a countable set, it follows that there must be a cluster $C \in \mathsf{C}_W(\lambda)$ such that $Q \cap (C \times C)$ is not null. But $Q \subset R \times (A \setminus R)$, so this implies that

$(R \times (A \setminus R)) \cap (C \times C)$ is not null. We have the identity:

$$(R \times (A \setminus R)) \cap (C \times C) = (R \cap C) \times ((A \setminus R) \cap C),$$

which then implies that $(R \cap C) \times ((A \setminus R) \cap C)$ is not null. However, this is a contradiction, since $R \cap C$ is necessarily a set of measure zero by the definition of R. Hence it must be that all of A excluding a null set is contained within $\bigcup \mathsf{C}_W(\lambda)$.

Let $\hat{A} = A \cap \bigcup \mathsf{C}_W(\lambda)$. Then $\hat{A}$ is equivalent to A in that it differs only by a set of measure zero, however, it is contained entirely within $\bigcup \mathsf{C}_W(\lambda)$. Since $\hat{A}$ is a set of positive measure, there must exist a $\hat{C} \in \mathsf{C}_W(\lambda)$ such that $\mu(\hat{A} \cap \hat{C}) > 0$. We will show that $\mu(\hat{A} \setminus \hat{C}) = 0$.

Let $S = \hat{A} \cap \hat{C}$, and let $T = \hat{A} \setminus S$. Note that $T \cap \hat{C}$ is null. Suppose for a contradiction that T is not null. Since T is contained within $\bigcup \mathsf{C}_W(\lambda)$, we may decompose it as the union

$$T = \bigcup_{C \in \mathsf{C}_W(\lambda)} T \cap C$$

hence we have

$$\begin{aligned} S \times T &= \bigcup_{C \in \mathsf{C}_W(\lambda)} S \times (T \cap C) \\ &= \bigcup_{C \in \mathsf{C}_W(\lambda)} (\hat{A} \cap \hat{C}) \times (T \cap C) \\ &= \bigcup_{C \in \mathsf{C}_W(\lambda)} (\hat{A} \times T) \cap (\hat{C} \times C). \end{aligned}$$

But $M < \lambda$ almost everywhere on $\hat{C} \times C$ whenever C and $\hat{C}$, are disjoint clusters. Hence $M^{-1}[\lambda, 1] \cap (S \times T)$ is equal, up to a null set, to the set $M^{-1}[\lambda, 1] \cap (\hat{A} \times T) \cap (\hat{C} \times \hat{C})$. Using the identity once again, this is the set $M^{-1}[\lambda, 1] \cap \Big[(\hat{A} \cap \hat{C}) \times (\hat{C} \cap T)\Big]$. But $\hat{C} \cap T$ is null, so that this set is null. This is a contradiction, as it implies that $M < \lambda$ almost

everywhere on $S \times T$, but $S \cup T = \hat{A}$ is connected at level λ. Therefore it must be that T is null, and hence $\mu(\hat{A} \setminus \hat{C}) = 0$. This implies that $\mu(A \setminus \hat{C}) = 0$, and so A is contained in some cluster of W at level λ, namely C. ■

Lemma 3.7. *Suppose a set A of positive measure is contained in a cluster at every level $\lambda' < \lambda$. Then A is contained in a cluster at level λ.*

Proof. We may construct a sequence $C_1, C_2, \ldots$ of clusters such that C_i is a cluster at level $\lambda - 1/n$, and C_i contains A. Then the intersection $C = \bigcap_{i=1}^{\infty} C_i$ is connected at all levels $\lambda' < \lambda$, as otherwise there would exist a $\lambda^* < \lambda$ at which C is disconnected, but this would imply that C_i is disconnected for any i such that $\lambda - 1/i > \lambda^*$. Furthermore, C has positive measure, since the measure of every C_i is at least $\mu(A)$. Therefore, C is contained in some cluster at level λ, and C contains A. Hence A is in some cluster at level λ. ■

With these results, we are now able to prove Theorem 3.2:

Proof of Theorem 3.2. Let C be an arbitrary representative of the cluster $\mathscr{C}$. By definition of the mergeon, all but a null set of $C \times C$ is contained within $M^{-1}[\lambda, 1]$, and therefore $M \geq \lambda$ almost everywhere on $C \times C$. This implies that C is connected at level λ in M, which in turn implies that C is contained in some cluster C' of M at level λ. By definition, C' is connected in M at every level $\lambda' < \lambda$, and so Lemma 3.6 implies that C' is contained in some cluster of W at every level $\lambda' < \lambda$. Lemma 3.7 then implies that C' is contained in some cluster of W at level λ. In other words, C is a cluster of W at level λ, and $C \subset C'$, so the fact that C' is contained in a cluster of W at level λ implies that C' differs from C by at most a null set. Hence C is a cluster of M.

Now suppose C is a cluster of M at level λ. Then C is connected in M at every level $\lambda' < \lambda$, and so Lemma 3.6 implies that C is contained in some cluster of W at every level $\lambda' < \lambda$. Lemma 3.7 then implies that C is contained in some cluster of W at level λ. Let C' be this cluster. Then the above implies that C' is a cluster at level λ in M. But $C \subset C'$,

and C is a cluster of M, so it must be that C and C' differ by a null set, and hence C is a cluster of W. ■

Additionally, we show that the mergeon transforms naturally under measure preserving transformations:

Theorem 3.3. *Let W be a graphon and M a mergeon of the cluster tree of W. If φ is a measure preserving transformation, then M^φ is a mergeon of the cluster tree of W^φ.*

Proof. One one hand, the function defined by

$$M_0^{-1}[\lambda, 1] = \bigcup_{C \in \mathsf{C}_W(\lambda)} \varphi^{-1}(C) \times \varphi^{-1}(C)$$

is a mergeon of W^φ, since C is a cluster of W if and only if $\varphi^{-1}(C)$ is a cluster of W^φ by Theorem 3.1 on page 53. Now consider the pullback M^φ and its upper level set

$$\begin{aligned}(M^\varphi)^{-1}[\lambda, 1] &= \{(x, y) : M^\varphi(x, y) \geq \lambda\} \\ &= \{(x, y) : M(\varphi(x), \varphi(y)) \geq \lambda\},\end{aligned}$$

which, by definition of the mergeon, is

$$= \left\{ (x, y) : (\varphi(x), \varphi(y)) \in \bigcup_{C \in \mathsf{C}_W(\lambda)} C \times C \right\}.$$

It is well-known that if φ is a measure preserving map, then the transformation defined by $\Phi : (x, y) \mapsto (\varphi(x), \varphi(y))$ is also measure preserving and measurable. Therefore we have

$$= \Phi^{-1} \left(\bigcup_{C \in \mathsf{C}_W(\lambda)} C \times C \right).$$

Since preimages commute with arbitrary unions:

$$= \bigcup_{C \in \mathsf{C}_W(\lambda)} \Phi^{-1}(C \times C)$$

Some thought will show that $\Psi^{-1}(C \times C) = \varphi^{-1}(C) \times \varphi^{-1}(C)$, such that:

$$= \bigcup_{C \in \mathsf{C}_W(\lambda)} \varphi^{-1}(C) \times \varphi^{-1}(C)$$

Comparing this to the definition of M_0 above, which was a mergeon of W^φ, we see that M^φ is a mergeon of W^φ. ■

3.6.2 Strict cluster trees and mergeons

A graphon cluster tree is a hierarchical collection of equivalence classes of sets. It is sometimes useful to instead to work with a hierarchical collection of subsets of $[0, 1]$. We may always do so by choosing a section map ρ and applying it to every cluster in the cluster tree. Though the choice of representative of a given cluster is arbitrary, it will sometimes be useful to choose it in such a way that the cluster tree has strictly nested structure, as made precise in the following definition.

Definition 3.8 (Strict section)**.** Let C be a cluster tree. A *strict section* $\tilde{\rho} : \mathsf{C} \to \Sigma$ is a function which selects a unique representative from each cluster $\mathscr{C}$ such that if:

1. $\mu(\mathscr{C} \cap \mathscr{C}') = 0 \Rightarrow \tilde{\rho}(\mathscr{C}) \cap \tilde{\rho}(\mathscr{C}) = \emptyset$,
2. $\mathscr{C} \subset \mathscr{C}' \Rightarrow \tilde{\rho}(\mathscr{C}) \subset \tilde{\rho}(\mathscr{C}')$, and
3. (Technical condition) $\tilde{\rho}(\mathscr{C}) = \bigcap\{\tilde{\rho}(\mathscr{C}') : \mathscr{C}' \supset \mathscr{C}\}$.

The *strict cluster tree* $\tilde{\mathsf{C}}$ is defined by $\tilde{\mathsf{C}}(\lambda) = \{\, \tilde{\rho}(\mathscr{C}) : \mathscr{C} \in \mathsf{C}(\lambda)\}$.

The following result shows that a strict section function always exists:

Theorem 3.4. *Let* C *be a cluster tree. There exists a section function* $\tilde{\rho}$ *on* C *such that if*

1. $\mu(\mathscr{C} \cap \mathscr{C}') = 0 \Rightarrow \tilde{\rho}(\mathscr{C}) \cap \tilde{\rho}(\mathscr{C}) = \emptyset$,
2. $\mathscr{C} \subset \mathscr{C}' \Rightarrow \tilde{\rho}(\mathscr{C}) \subset \tilde{\rho}(\mathscr{C}')$, *and*
3. *(Technical condition)* $\tilde{\rho}(\mathscr{C}) = \bigcap\{\tilde{\rho}(\mathscr{C}') : \mathscr{C}' \supset \mathscr{C}\}$.

Proof. We construct such a section function on the clusters at rational levels and extend it to $[0, 1]$. Let $\mathbb{Q}_{[0,1]} = \mathbb{Q} \cap [0, 1]$. Define

$$\hat{\mathsf{C}} = \{\mathscr{C} \in \mathsf{C}(\lambda) : \lambda \in \mathbb{Q}_{[0,1]}\}$$

that is, $\hat{\mathsf{C}}$ is the set of all clusters from every rational level. Note that this is a countable collection. For any cluster $\mathscr{C} \in \hat{\mathsf{C}}$, define $P_{\mathscr{C}}$ to be the set of clusters in $\hat{\mathsf{C}}$ which have null intersection with $\mathscr{C}$. That is:

$$P_{\mathscr{C}} = \{\mathscr{C}' \in \hat{\mathsf{C}} : \mu(\mathscr{C} \cap \mathscr{C}') = 0\}.$$

Let ρ_0 be an arbitrary section function, and define the section function $\rho_1 : \hat{\mathsf{C}} \to \Sigma$ as follows:

$$\rho_1(\mathscr{C}) = \rho_0(\mathscr{C}) \setminus \bigcup P_{\mathscr{C}}.$$

Furthermore, let $\mathscr{C}_0$ be the equivalence class of sets differing from $[0, 1]$ by a null set, and define $\rho_1(\mathscr{C}_0) = [0, 1]$; This will ensure that all pairs of points have a well-defined merge height. The intersection of $\rho_0(\mathscr{C})$ and $\bigcup P_{\mathscr{C}}$ is null, by definition of $P_{\mathscr{C}}$ and the fact that it is a countable set. Therefore, $\rho_1(\mathscr{C}) \triangle \rho_0(\mathscr{C})$ is null, and $\rho_1(\mathscr{C})$ is hence a valid representative of $\mathscr{C}$. Furthermore, for any $\mathscr{C}, \mathscr{C}' \in \hat{\mathsf{C}}$ such that $\mu(\mathscr{C} \cap \mathscr{C}') = 0$, we have $\rho_1(\mathscr{C}) \cap \rho_2(\mathscr{C}') = \emptyset$.

We now define the section function on all levels in $[0, 1]$. For a cluster $\mathscr{C}$ at any level, define its set of ancestors in $\hat{\mathsf{C}}$ to be

$$\mathfrak{A}_{\mathscr{C}} = \{\mathscr{C}' \in \hat{\mathsf{C}} : \mu(\mathscr{C}' \setminus \mathscr{C}) = 0\}.$$

Then define

$$\tilde{\rho}(\mathscr{C}) = \bigcap_{\mathscr{C}' \in \mathfrak{A}_{\mathscr{C}}} \rho_1(\mathscr{C}')$$

Hence $\tilde{\rho}$ trivially satisfies the third condition of the claim.

We must argue that $\tilde{\rho}(\mathscr{C})$ is a valid representative of $\mathscr{C}$. First, suppose that $\mathscr{C}$ is a cluster at level λ. Then $\mathfrak{A}_{\mathscr{C}}$ contains a cluster from every rational level below λ, and $\tilde{\rho}(\mathscr{C})$ is contained in a representative of each of them. It follows that $\tilde{\rho}(\mathscr{C})$ is contained in a cluster representative at every level $\lambda' < \lambda$. Hence, by Lemma 3.7, $\tilde{\rho}(\mathscr{C})$ is contained in a cluster representative at level λ. But $\mathscr{C}$ is essentially contained in all of its ancestors. Therefore, it must be that $\tilde{\rho}(\mathscr{C}) \triangle \mathscr{C}$ is null and so $\tilde{\rho}(\mathscr{C})$ is a valid representative of $\mathscr{C}$.

Now we show that $\tilde{\rho}$ has the desired properties. Suppose $\mathscr{C}$ and $\mathscr{C}'$ have null intersection, and without loss of generality, assume that they are clusters at the same level λ. Let $\lambda' < \lambda$ be the maximal level at which their intersection is not null. Then there is some rational level $\tilde{\lambda}$ between λ' and λ. Hence $\tilde{\rho}(\mathscr{C})$ is strictly contained in the representative $\rho_1(\tilde{\mathscr{C}})$ of some cluster $\tilde{\mathscr{C}}$ at level $\tilde{\lambda}$, and similarly, $\tilde{\rho}(\mathscr{C})$ is strictly contained in $\rho_1(\tilde{\mathscr{C}'})$ at the same level. Necessarily, $\tilde{\mathscr{C}}$ and $\tilde{\mathscr{C}'}$ have null intersection, and so $\rho_1(\tilde{\mathscr{C}})$ and $\rho_1(\tilde{\mathscr{C}'})$ are strictly disjoint. Therefore, so also are $\tilde{\rho}(\mathscr{C})$ and $\tilde{\rho}(\mathscr{C}')$.

Furthermore, suppose that $\mathscr{C}$ and $\mathscr{C}'$ are such that $\mu(\mathscr{C}' \setminus \mathscr{C}) = 0$. Suppose without loss of generality that $\lambda > \lambda'$ (if $\lambda = \lambda'$ then $\tilde{\rho}(\mathscr{C}) = \tilde{\rho}(\mathscr{C}')$). Then the ancestors of $\mathscr{C}$ include the ancestors of $\mathscr{C}'$, and so the intersection of the ancestors of $\mathscr{C}$ is a subset of the intersection of the ancestors of $\mathscr{C}'$. This proves that $\tilde{\rho}(\mathscr{C}) \subset \tilde{\rho}(\mathscr{C}')$. ■

Furthermore, given a cluster tree and a strict section, there is a *unique* mergeon representing the strict cluster tree, defined as follows:

Definition 3.9 (Strict mergeon)**.** Let C be a cluster tree, and suppose $\tilde{\rho}$ is a strict section for the clusters of C. Then M is a *strict mergeon* of the strict cluster tree induced by $\tilde{\rho}$ if, for every $\lambda \in [0,1]$,

$$M^{-1}[\lambda, 1] = \bigcup_{\mathscr{C} \in \mathsf{C}_W(\lambda)} \tilde{\rho}(\mathscr{C}) \times \tilde{\rho}(\mathscr{C}).$$

Because any two mergeons of the same cluster tree differ only on a null set, we are typically free to assume that a mergeon is strict without much loss. Making this assumption will simplify some statements and proofs.

3.7 Notions of consistency

We have so far defined the sense in which a graphon has hierarchical cluster structure and identified the *mergeon* as an object which encodes this hierarchy. We now turn to the problem of determining whether a clustering algorithm is able to recover this structure when applied to a graph sampled from a graphon. Our approach is to define a distance between the infinite graphon cluster tree and a finite clustering. We will then define consistency by requiring that a consistent method converge to the graphon cluster tree in this distance for all inputs minus a set of vanishing probability.

3.7.1 Merge distortion revisited

Recall from Definition 1.2 in Section 1.1 that a *hierarchical clustering* $\mathcal{C}$ of a set S – or, from now on, just a *clustering* of S – is a collection of non-empty subsets of S such that $S \in \mathcal{C}$ and for all $C, C' \in \mathcal{C}$, either $C \subset C'$, $C' \subset C$, or $C \cap C' = \emptyset$. Suppose $\mathcal{C}$ is a clustering of a finite set S consisting of graphon nodes; i.e, $S \subset [0,1]$. How might we measure the distance between this clustering and a graphon cluster tree C? Intuitively, the two trees

are close if every pair of points in S merges in $\mathcal{C}$ at about the same level as they merge in C. But this informal description faces two problems: First, C is a collection of equivalence classes of sets, and so the height at which any pair of points merges in C is not defined. Recall, however, that the cluster tree has an alternative representation as a *mergeon*. A mergeon *does* define a merge height for every pair of nodes in a graphon, and thus provides a solution to this first issue. Second, the clustering $\mathcal{C}$ is not equipped with a height function, and so the height at which any pair of points merges in $\mathcal{C}$ is also undefined. Following the previous chapter, our approach is to *induce* a merge height function on the clustering using the mergeon in the following way:

Definition 3.10 (Induced merge height)**.** Let M be a mergeon, and suppose S is a finite subset of $[0,1]$. Let $\mathcal{C}$ be a clustering of S. The *merge height function on $\mathcal{C}$ induced by M* is defined by $\hat{M}_{\mathcal{C}}(s,s') = \min_{u,v \in \mathcal{C}(s,s')} M(u,v)$, for every $s, s' \in S \times S$, where $\mathcal{C}(s,s')$ denotes the smallest cluster $C \in \mathcal{C}$ which contains both s and s'.

We measure the distance between a clustering $\mathcal{C}$ and the cluster tree C using the *merge distortion*:

Definition 3.11. Let M be a mergeon, S a finite subset of $[0,1]$, and $\mathcal{C}$ a clustering of S. The *merge distortion* is defined by $d_S(M, \hat{M}_{\mathcal{C}}) = \max_{s,s' \in S,\, s \neq s'} |M(s,s') - \hat{M}_{\mathcal{C}}(s,s')|$.

Defining the induced merge height and merge distortion in this way leads to an especially meaningful interpretation. In particular, if the merge distortion between $\mathcal{C}$ and C is ϵ, then any two clusters of C which are separated at level λ but merge below level $\lambda - \epsilon$ are correctly separated in the clustering $\mathcal{C}$. A similar result guarantees that a cluster in C is connected in $\mathcal{C}$ at within ϵ of the correct level. The following makes this precise:

Claim 3.8. *Let C be cluster tree, and let $\tilde{\mathbb{C}}$ be a strict cluster tree obtained by applying a strict section $\tilde{\rho}$ to each cluster of C as described in Section 3.6.2. Let M be the strict mergeon representing $\tilde{\mathbb{C}}$. Let $S = (x_1, \ldots, x_n)$ with each $x_i \in [0,1]$ and suppose $\mathcal{C}$ is a clustering of S. Let $\hat{M}$ be the induced merge height on $\mathcal{C}$. If $d_S(M, \hat{M}) < \epsilon$, we then have:*

1. Connectedness*: If C is a cluster of $\tilde{\mathbb{C}}$ at level λ and $|S \cap C| \geq 2$, then the smallest cluster in $\mathcal{C}$ which contains all of $S \cap C$ is contained within $C' \cap S$, where C' is the cluster of $\tilde{\mathbb{C}}$ at level $\lambda' = \lambda - \epsilon$ which contains C.*

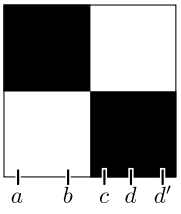
2. Separation*: If C_1 and C_2 are two clusters of $\tilde{\mathbb{C}}$ at level λ such that C_1 and C_2 merge at level $\lambda' < \lambda - \epsilon$, then if $|C_1 \cap S|, |C_2 \cap S| \geq 2$, the smallest cluster in $\mathcal{C}$ containing $C_1 \cap S$ and the smallest cluster containing $C_2 \cap S$ are disjoint.*

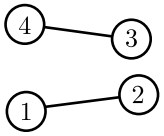
Proof. First we prove connectedness. Let $\hat{C}$ be the smallest cluster in $\tilde{\mathbb{C}}$ containing $C \cap S$. Suppose $\hat{C}$ contains a point y from outside of C'. Let x, x' be any two distinct points in $C \cap S$. Then necessarily $M(x, y) < \lambda' = \lambda - \epsilon$, as, since M is strict, $M(x, y) \geq \lambda$ if and only if x, y are in the same cluster of $\tilde{\mathbb{C}}$ at level λ. Hence the merge distortion is at least $M(x, x') - M(x, y) > \epsilon$, which is a contradiction.

Separation follows from connectedness. Let $\hat{C}_1$ be the smallest cluster in the clustering containing $C_1 \cap S$, and similarly for $\hat{C}_2$. Let $\tilde{C}_1$ and $\tilde{C}_2$ be the clusters at level $\lambda - \epsilon$ which contain C_1 and C_2. Then $\tilde{C}_1 \cap \tilde{C}_2 = \emptyset$, since C_1 and C_2 merge below $\lambda - \epsilon$. Furthermore, by connectedness, $C_1 \cap S \subset \tilde{C}_1$ and $C_2 \cap S \subset \tilde{C}_2$. Hence they are disjoint. ■

3.7.2 The label measure

We will use the merge distortion to measure the distance between $\mathcal{C}$, a hierarchical clustering of a graph, and C, the graphon cluster tree. Recall, however, that the nodes of a graph sampled from a graphon have integer labels. That is, $\mathcal{C}$ is a clustering of $[n]$, and not of a subset of $[0, 1]$. Hence, in order to apply the merge distortion, we must first relabel the nodes of the graph, placing them in direct correspondence to nodes of the graphon, i.e., points in $[0, 1]$.





Recall that we sample a graph of size n from a graphon W by first drawing n points $\mathbf{x_1}, \ldots, \mathbf{x_n}$ uniformly at random from the unit interval. We then generate a graph on node set $[n]$ by connecting nodes i and j

with probability $W(\mathbf{x_i}, \mathbf{x_j})$. However, the nodes of the sampled graph are not labeled by $\mathbf{x_1}, \ldots, \mathbf{x_n}$, but rather by the integers $1, \ldots, n$. Thus we may think of $\mathbf{x_i}$ as being the "true" latent label of node i. In general the latent node labeling is not recoverable from data, as is demonstrated by the figure to the right. We might suppose that the graph shown is sampled from the graphon above it, and that node 1 corresponds to a, node 2 to b, node 3 to c, and node 4 to d. However, it is just as likely that node 4 corresponds to d', and so neither labeling is more "correct". It is clear, though, that some labelings are less likely than others. For instance, the existence of the edge $(1, 2)$ makes it impossible that 1 corresponds to a and 2 to c, since $W(a, c)$ is zero.

Therefore, given a graph $G = ([n], E)$ sampled from a graphon, there are many possible relabelings of G which place its nodes in correspondence with nodes of the graphon, but some are more likely than others. The merge distortion depends which labeling of G we assume, but, intuitively, a good clustering of G will have small distortion with respect to highly probable labelings, and only have large distortion on improbable labelings. Our approach is to assign a probability to every pair (G, S) of a graph and possible labeling. We will thus be able to measure the probability mass of the set of pairs for which a method performs poorly, i.e., results in a large merge distortion.

More formally, let $\mathfrak{G}_n$ denote the set of all undirected, unweighted graphs on node set $[n]$, and let Σ^n be the sigma-algebra of Lebesgue-measurable subsets of $[0, 1]^n$. A graphon W induces a unique product measure $\Lambda_{W,n}$ defined on the product sigma-algebra $2^{\mathfrak{G}_n} \times \Sigma^n$ such that for all $\mathcal{G} \in 2^{\mathfrak{G}_n}$ and $\mathcal{S} \in \Sigma^n$:

$$\Lambda_{W,n}(\mathcal{G} \times \mathcal{S}) = \sum_{G \in \mathcal{G}} \left(\int_{\mathcal{S}} \mathcal{L}_W(S|G)\, dS \right),$$

where

$$\mathcal{L}_W(S \mid G) = \prod_{(i,j) \in E(G)} W(x_i, x_j) \prod_{(i,j) \notin E(G)} [1 - W(x_i, x_j)],$$

and $E(G)$ represents the edge set of the graph G. We recognize $\mathcal{L}_W(S \mid G)$ as the integrand in Equation (3.1) for the probability of a graph as determined by a graphon. If G is fixed, integrating $\mathcal{L}_W(S \mid G)$ over all $S \in [0,1]^n$ gives the probability of G under the model defined by W.

We now formally state our notion of consistency. A *hierarchical graph clustering method* f is a map from the set $\mathfrak{G}_n$ of all unweighted, undirected graphs on node set $[n]$ to the set of hierarchical clusterings of $[n]$. If $\mathcal{C}$ is a clustering of $[n]$ and $S = (x_1, \ldots, x_n)$, write $\mathcal{C} \circ S$ to denote the relabeling of $\mathcal{C}$ by S, in which i is replaced by x_i in every cluster. Then $f(G) \circ S$ is a hierarchical clustering of S, and $\hat{M}_{f(G)\circ S}$ denotes the merge function induced on $f(G) \circ S$ by M in the manner of Definition 3.10.

Definition 3.12 (Consistency)**.** Let W be a graphon and M be a mergeon of W. A hierarchical graph clustering method f is said to be a *consistent* estimator of the graphon cluster tree of W if for any fixed $\epsilon > 0$, as $n \to \infty$,

$$\Lambda_{W,n}\left(\left\{(G,S) : d_S(M, \hat{M}_{f(G)\circ S}) > \epsilon\right\}\right) \to 0.$$

The choice of mergeon for the graphon W does not affect consistency, as any two mergeons of the same graphon differ on a set of measure zero. Furthermore, consistency is with respect to the random graph model, and not to any particular graphon representing the model. The following theorem makes this precise.

Theorem 3.5. *Let W be a graphon and φ a measure preserving transformation. A clustering method f is a consistent estimator of the graphon cluster tree of W if and only if it is a consistent estimator of the graphon cluster tree of W^φ.*

Proof. Let M be a mergeon of the cluster tree of W and fix any $\epsilon > 0$. Consider the set

$$F = \left\{(G,S) : d_S\left(M, \hat{M}_{f(G)\circ S}\right) > \epsilon\right\},$$

which is the set of graph/sample pairs for which the merge distortion between the clustering and the mergeon M is greater than ϵ. Consistency with respect to the cluster tree of W requires that $\Lambda_{W,n}(F) \to 0$ as $n \to \infty$. Now recall that M^φ is a mergeon of the cluster tree of W^φ, and consider

$$F_\varphi = \left\{(G, S) : d_S\left(M^\varphi, \hat{M}^\varphi_{f(G)\circ S}\right) > \epsilon\right\}$$

where $\hat{M}^\varphi{}_{f(G)\circ S}$ is the merge height induced on the clustering $f(G)\circ S$ by the mergeon M^φ. F_φ is the set of graph/sample pairs for which the merge distortion between the clustering and the mergeon M^φ is greater than ϵ. Consistency with respect to the cluster tree of W^φ requires that $\Lambda_{W^\varphi,n}(F_\varphi) \to 0$ as $n \to \infty$. It will therefore be sufficient to show that $\Lambda_{W,n}(F) = \Lambda_{W^\varphi,n}(F_\varphi)$ to prove the claim.

Now we compute the measure under $\Lambda_{W^\varphi,n}$ of F_φ:

$$\Lambda_{W^\varphi,n}(F_\varphi) = \sum_{G\in\mathfrak{G}_n} \int_{F_\varphi(G)} \mathcal{L}_{W^\varphi}(S \mid G)\, dS,$$

where $F_\varphi(G)$ denotes the *section* of F_φ by graph G, that is, the set $F_\varphi(G) = \{S : (G, S) \in F_\varphi\}$. It is easy to see that $\mathcal{L}_{W^\varphi}(S \mid G) = \mathcal{L}_W(\varphi(S), G)$, such that:

$$\Lambda_{W^\varphi,n}(F_\varphi) = \sum_{G\in\mathfrak{G}_n} \int_{F_\varphi(G)} \mathcal{L}_W(\varphi(S) \mid G)\, dS,$$

Since $M^\varphi(x, y) = M(\varphi(x), \varphi(y))$, we have

$$d_S\left(M^\varphi, \hat{M}^\varphi_{f(G)\circ S}\right) = d_{\varphi(S)}\left(M, \hat{M}_{f(G)\circ\varphi(S)}\right)$$

such that

$$F_\varphi = \left\{(G, S) : d_{\varphi(S)}\left(M, \hat{M}_{f(G)\circ\varphi(S)}\right) > \epsilon\right\}.$$

Now consider the section of F by G, defined by $F(G) = \{S : (G, S) \in F\}$. It is clear that $F_\varphi(G) = \varphi^{-1}(F(G))$ for every graph G. Therefore,

$$\Lambda_{W^\varphi,n}(F_\varphi) = \sum_{G \in \mathfrak{G}_n} \int_{\varphi^{-1}(F(G))} \mathcal{L}_W(\varphi(S) \mid G)\, dS.$$

Now, it is a property of measure preserving maps that $\int_{\varphi^{-1}(A)} f(\varphi(x))\, d\mu(x) = \int_A f(x)\, d\mu(x)$; See, for example, (Ash and Doleans-Dade, 2000). Therefore, we have

$$\begin{aligned}
\Lambda_{W^\varphi,n}(F_\varphi) &= \sum_{G \in \mathfrak{G}_n} \int_{\varphi^{-1}(F(G))} \mathcal{L}_W(\varphi(S) \mid G)\, dS \\
&= \sum_{G \in \mathfrak{G}_n} \int_{F(G)} \mathcal{L}_W(S \mid G)\, dS \\
&= \Lambda_{W,n}(F)
\end{aligned}$$

which proves the claim. ■

3.7.3 Consistency and the blockmodel

If a graph clustering method is consistent in the sense defined above, it is also consistent in the stochastic blockmodel; i.e., it ensures strict recovery of the communities with high probability as the size of the graphs grow large. For instance, suppose W is a stochastic blockmodel graphon with α along the block-diagonal and β everywhere else. W has two clusters at level α, merging into one cluster at level β. When the merge distortion between the graphon cluster tree and a clustering is less than $\alpha - \beta$, which will eventually be the case with high probability if the method is consistent, the two clusters are totally disjoint in $\mathcal{C}$; this implication is made precise by Claim 3.8 on page 74.

3.8 Sufficient conditions for consistency

Having made our notion of consistency rigorous, we now ask whether consistent graphon clustering algorithms exist. In this section, we show that any method which is capable of consistently estimating the probability of each edge in a random graph leads to a consistent clustering algorithm; the next section will construct such an edge probability estimator.

3.8.1 The single-linkage clustering of edge probabilities

Our definition of a graphon cluster is motivated by interpreting the graphon function W as the weight matrix of an infinite weighted graph. In the case of a finite weighted graph H, we argued that a natural approach to defining a cluster is as a connected component of an appropriately-defined subgraph. In particular, let H_λ be the subgraph induced by removing all edges of weight less than λ from H. The *clusters* of H at level λ are defined to be the connected components of H_λ. Our definition of a graphon cluster can be seen as an extension of this notion to the setting of infinite weighted graphs.

In fact, the clusters of H are precisely those obtained by applying the familiar *single-linkage* clustering algorithm, using the matrix of edge weights as a similarity matrix. In this sense, the ideal clustering of a graphon, according to our definitions, is (informally-speaking) the *single-linkage* clustering obtained by interpreting $W(x, x')$ as a measure of the similarity between x and x'.

In clustering we are given an unweighted, unlabeled graph sampled from a graphon; our goal is to recover the graphon cluster tree. Each possible edge in this sampled graph is the result of a Bernoulli trial with a success probability that is latent. It is possible, however, to estimate this latent probability. Doing so for every pair of nodes results in a finite weighted graph in which the weight of each edge is an estimate of the edge's latent probability. Following the above discussion, the single-linkage clustering of this graph is quite natural. Intuitively, the quality of this clustering depends upon the accuracy of the

edge probability estimates. Furthermore, it is easy to see that changing the weight of a single edge can drastically alter the single-linkage clustering. As a result, we feel that *every* edge probability estimate must be accurate in order to guarantee that the single-linkage clustering is close to the graphon cluster tree.

We now formalize this intuition into a sufficiency result. We will work exclusively with graphons which are piecewise Lipschitz (or weakly isomorphic to a piecewise Lipschitz graphon). We follow Zhang et al. (2015) in defining a piecewise-Lipschitz graphon as follows:

Definition 3.13 (Piecewise Lipschitz)**.** We say that $\mathcal{B} = \{B_1, \ldots, B_k\}$ is a *block partition* if each B_i is an open, half-open, or closed interval in $[0, 1]$ with positive measure, $B_i \cap B_j$ is empty whenever $i \neq j$, and $\bigcup \mathcal{B} = [0, 1]$. We say that a graphon W is *piecewise* c*-Lipschitz* if there exists a set of blocks $\mathcal{B}$ such that for any (x, y) and (x', y') in $B_i \times B_j$, $|W(x, y) - W(x', y')| \leq \mathsf{c}(|x - x'| + |y - y'|)$.

Let $\mathbf{S} = (\mathbf{x_1}, \ldots, \mathbf{x_n})$ be an ordered set of n uniform random variables drawn from the unit interval. Fix a graphon W, and let $\mathbf{P}$ be the random matrix whose ij entry is given by $W(\mathbf{x_i}, \mathbf{x_j})$. We say that $\mathbf{P}$ is the random *edge probability matrix*. Assuming that W has structure, it is possible to estimate $\mathbf{P}$ from a single graph sampled from W. We say that an estimator $\hat{\mathbf{P}}$ of $\mathbf{P}$ is *consistent* in max-norm[3] if, for any $\epsilon > 0$, $\lim_{n\to\infty} \mathbb{P}(\max_{i\neq j} |\mathbf{P}_{ij} - \hat{\mathbf{P}}_{ij}| > \epsilon) = 0$. The following theorem, whose proof comprises the remainder of the section, states that any estimator which is consistent in this sense leads to a consistent clustering algorithm:

Theorem 3.6. *Let W be a piecewise c-Lipschitz graphon. Let $\hat{\mathbf{P}}$ be a consistent estimator of $\mathbf{P}$ in max-norm. Let f be the clustering method which performs single-linkage clustering using $\hat{\mathbf{P}}$ as a similarity matrix. Then f is a consistent estimator of the graphon cluster tree of W in the sense of merge distortion.*

[3]Note that our definition ignores the diagonal, and is therefore a slight abuse of terminology.

3.8.2 Proof

We begin our proof of Theorem 3.6 by adopting a deterministic setting in which the graphon nodes are fixed; we will later reïntroduce randomness when proving the main theorem.

The merge estimate matrix.

Consider a graphon W and a fixed sampled $S = (x_1, \ldots, x_n)$ whose elements are points in the unit interval. S induces a fixed edge probability matrix P whose entries are defined as $P_{ij} = W(x_i, x_j)$. Let $\hat{P}$ be an estimate of P. Let $\mathcal{C}_{\hat{P}}$ be the clustering of $[n]$ obtained by applying the single-linkage algorithm to $\hat{P}$, interpreting it as a similarity matrix.

Our first observation is that $\hat{P}$ being close to P does *not* imply that $\mathcal{C}_{\hat{P}}$ is close to the graphon cluster tree in the sense of merge distortion. This is because the sample S may not be sufficient to capture the structure of the graphon. For instance, consider a graphon which contains two large regions of high probability connected into a single cluster by a thin bridge. If S does not contain a sample from the bridge, neither it nor the cluster it joins can be detected. In this case, the single-linkage clustering of $\hat{P}$ may be very different from the structure of the underlying graphon, even though the estimate of each edge between the observed samples may be highly accurate.

We therefore work not with the edge probability estimates directly, but with a matrix of *merge estimates*, defined as follows. Let H be the finite weighted graph on $[n]$ in which the weight between node i and j is given by $\hat{P}_{ij}$. Let H_λ be the finite subgraph induced by removing any edge of weight less than λ from H. For any $i, j \in [n]$, define the *merge estimate* matrix Q by

$$\hat{Q}_{ij} = \max\{\lambda : \ i \text{ and } j \text{ are connected in } H_\lambda\}.$$

We recognize $\hat{Q}$ as the natural similarity matrix induced by the single-linkage clustering of $\hat{P}$. It follows that the single-linkage clustering of $\hat{Q}$, written $\mathcal{C}_{\hat{Q}}$, is identical to $\mathcal{C}_{\hat{P}}$, i.e., the

single-linkage clustering of $\hat{P}$.

As its name suggests, the *merge estimate* matrix can be interpreted as an estimate of the mergeon. If $\hat{Q}$ is close to M on the sample S, we can indeed show that the single-linkage clustering of the data is close to the graphon cluster tree in merge distortion, as the following lemma shows:

Lemma 3.8. *Let W be a graphon, M be a mergeon of W, and $S = (x_1, \ldots, x_n)$. Suppose $\max_{i \neq j} |M(x_i, x_j) - \hat{Q}_{ij}| < \epsilon$, and let $\mathcal{C}_{\hat{Q}}$ be the single-linkage clustering of the weighted graph H with weight matrix $\hat{Q}$. Let $\hat{M}$ be the merge height on $\mathcal{C}_{\hat{Q}}$ induced by M. Write d_S for the merge distortion w.r.t. the sample S. Then $d_S(M, \hat{M}) < 2\epsilon$.*

Proof. Take any arbitrary $i \neq j$ in the clustering $\mathcal{C}_{\hat{Q}}$. Let C be the smallest cluster containing both i and j. Then C is a cluster in H at level $\hat{Q}_{ij}$. Let $u, v \in C$, $u \neq v$ be such that $M(x_u, x_v) = \min_{u' \neq v' \in C} M(x_{u'}, x_{v'}) = \hat{M}_{ij}$. Then we have that $M(x_i, x_j) \geq M(x_u, x_v)$. On the other hand, u and v are members of C, which is a cluster at level $\hat{Q}_{ij}$, so that $\hat{Q}_{uv} \geq \hat{Q}_{ij}$. Hence $\hat{Q}_{uv} > M(x_i, x_j) - \epsilon$. But $\hat{Q}_{uv} < M(x_u, x_v) + \epsilon$. Therefore, $M(x_i, x_j) - M(x_u, x_v) < 2\epsilon$, and hence $M(x_i, x_j) - \hat{M}_{ij} < 2\epsilon$. This holds for all i and j simultaneously, since i and j were arbitrary. Hence the merge distortion is less than 2ϵ. ■

Accuracy of merge estimates.

We now prove that the merge estimate $\hat{Q}$ is close to the mergeon M on the sample S provided that the estimated edge probabilities $\hat{P}$ are accurate, and that the sample S is sufficient to capture the structure of W.

Recall that $\hat{Q}_{ij}$ is defined to be the largest value of λ such that nodes i and j are connected in the graph H_λ described above. There is an edge between nodes i and j in H_λ if and only if $\hat{P}_{ij} \geq \lambda$. Therefore, i and j are connected in H_λ if and only if there exists a path in H from node i to node j along which every edge has weight λ, or, equivalently, there exists a sequence of indices $p_1, \ldots, p_k$ such that $\hat{P}_{p_t, p_{t+1}} \geq \lambda$ for each $t \in \{1, \ldots, k-1\}$.

First, consider bounding $\hat{Q}_{ij}$ from below. We do show by showing that there must exist a sequence of points $x_i = x_{p_1}, \ldots, x_{p_k} = x_j$ such that the estimated edge probability between each consecutive pair of points is at least some λ, provided that $\hat{P}$ is accurate and S satisfies certain assumptions. If this is indeed the case, then i and j must be connected at level λ in H, and therefore $\hat{Q}_{ij} \geq \lambda$.

For this method of proof to succeed, the sample S must be sufficient to recover the fine structure of the graphon. In order formalize this, we will discretize the graphon by refining it into blocks on which it is approximately constant. We will then assume that the sample S contains at least one point from each block of the partition. Because the value of the graphon does not vary too much on each block of the partition, we may work with sequences of blocks instead of sequences of samples; this will allow us to use our notions of graphon connectedness which were defined for sets with positive measure, but not for individual points in a graphon.

More precisely, we define a *refinement* as follows:

Definition 3.14. A set of blocks $\mathcal{R} = \{R_i\}$ is a Δ-*refinement* of a block partition $\mathcal{B} = \{B_i\}$ if for every $R \in \mathcal{R}$, $\Delta \leq \mu(R) \leq 2\Delta$ and there exists some $B \in \mathcal{B}$ such that $B \supseteq R$.

We can think of the blocks in a refinement as being nodes in a weighted graph, such that the weight of the edge between blocks R and R' is approximately the value of W on $R \times R'$. As such, we define a *path* of blocks in a refinement as follows:

Definition 3.15 (λ-path)**.** Let $\mathcal{R}$ be a block partition of $[0, 1]$, and suppose $R, R' \in \mathcal{R}$. A λ-*path* from R to R' in a graphon W is a sequence $\langle R = R_1, \ldots, R_t = R' \rangle$ of blocks from $\mathcal{R}$ such that, for all $1 \leq i < t$, $W \geq \lambda$ almost everywhere on $R_i \times R_{i+1}$. The elements of the path need not be distinct.

We now prove a key lemma used in lower-bounding $\hat{Q}_{ij}$. Rather than directly finding a path of points between x_i and x_j, we instead find a sequence of refinement blocks between

the block containing x_i and the block containing x_j. This allows us to use our notions of graphon connectivity to lower-bound the edge weights along the path.

Lemma 3.9. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$. Let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$, and suppose $R, R' \in \mathcal{R}$ (possibly with $R = R'$). If there exists a cluster $\mathscr{C}$ at level λ such that $\mu(\mathscr{C} \cap R) > 0$ and $\mu(\mathscr{C} \cap R') > 0$, then there exists a $(\lambda' - 2\Delta\mathsf{c})$-path $(R = R_1, \ldots, R_t = R')$ between R and R', for any $\lambda' < \lambda$.*

Proof. To be precise, let $C = \rho(\mathscr{C})$ be any representative of the cluster $\mathscr{C}$. Fix a $\lambda' < \lambda$. Let

$$\mathcal{G} = \{R'' \in \mathcal{R} : \mu(R'' \cap C) > 0\}.$$

Then $\mathcal{G}$ contains, in particular, R_1 and R_t. Since $\mathscr{C}$ is connected at level λ, it is true that

$$\mu(W^{-1}[\lambda', 1] \cap (R_1 \cap C) \times (C \setminus R_1)) > 0.$$

Since $C \setminus R_1$ is a subset of $(\bigcup \mathcal{G}) \setminus R_1$, there must exist an $R_2 \in \mathcal{G}$ such that

$$\mu(W^{-1}[\lambda', 1] \cap R_1 \times R_2) > 0.$$

Consider W on R_2. From above, we know that there is a non-negligible subset of $R_1 \times R_2$ on which $W \geq \lambda'$. Hence there is some point in $R_1 \times R_2$ on which $W \geq \lambda'$. Therefore, due to the Lipschitz condition, we know that W is at least $\lambda' - 2\Delta\mathsf{c}$ everywhere on $R_1 \times R_2$.

Now let $S_2 = R_1 \cup (R_2 \cap C)$. Now, since $\mathscr{C}$ is connected at level λ, it is true that

$$\mu(W^{-1}[\lambda', 1] \cap S_2 \times (C \setminus S_2)) > 0.$$

By the same logic as above, there must exist an $R_3 \in \mathcal{G}$, $R_3 \neq R_2, R_1$ such that

$$\mu(W^{-1}[\lambda', 1] \cap S_2 \times R_3) > 0.$$

Hence it must be the case that either

$$\mu(W^{-1}[\lambda, 1] \cap R_1 \times R_3) > 0,$$

or

$$\mu(W^{-1}[\lambda, 1] \cap R_2 \times R_3) > 0.$$

In either case, it is true that between any pair chosen from R_1, R_2, R_3, there is a $\lambda - 2\Delta\mathsf{c}$ path. The process continues, choosing $R_4, R_5, \ldots$ and so on. This process must complete in a finite number of steps, since $\mathcal{G}$ is a finite set. At every step, there exists a $(\lambda - 2\Delta\mathsf{c})$-path between any two of the R_i. Hence we eventually construct a $(\lambda - 2\Delta\mathsf{c})$-path between R and R'. ■

To prove an upper-bound on $\hat{Q}_{ij}$, we use the fact that the existence of a λ-path of blocks implies that the path is connected at level λ in the graphon. Since the mergeon encodes the level at which this connection occurs, this gives a bound on λ.

Lemma 3.10. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$ and let M be a mergeon of W. Let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$. Let $\langle R_1, \ldots, R_t \rangle$ be a λ-path in $\mathcal{R}$. Let $C = R_1 \cup \ldots \cup R_t$. Then C is connected at level λ in W, and thus $M \geq \lambda$ almost everywhere on $C \times C = (R_1 \cup \ldots \cup R_t) \times (R_1 \cup \ldots \cup R_t)$.*

Proof. Let A be an arbitrary measurable subset of C such that $0 < \mu(A) < \mu(C)$. We will show that $W^{-1}[\lambda, 1] \cap A \times (C \setminus A)$ has positive measure, and therefore C is connected at level λ. Since C is connected at level λ in W, it must be part of some cluster at level λ, and so the mergeon is at least λ almost everywhere on $C \times C$.

There are two cases: Either 1) There exists a $j \in [t]$ such that $0 < \mu(R_j \cap A) < \mu(R_j)$, or 2) for all $i \in [t]$, either $\mu(R_i \cap A) = 0$ or $\mu(R_i \cap A) = \mu(R_i)$.

Assume the first case: there exists a j such that R_j contains some non-negligible part of A, but $\mu(A \cap R_j) < \mu(R_j)$. Since there are at least two elements in the path, there is a

j' such that $j' \in [t]$ and $|j - j'| = 1$, that is, $R_{j'}$ is either immediately before or after R_j in the λ-path. There are two sub-cases:

- $\mu(R_{j'} \cap A) = 0$, such that $R_{j'} \subseteq C \setminus A$. Then $(R_j \cap A) \times R_{j'} \subseteq A \times (C \setminus A)$. Since $\mu(R_j \cap A) > 0$ and $\mu(R_{j'}) > 0$, we have that $\mu((R_j \cap A) \times R_{j'}) > 0$, and since W is at least λ a.e. on $R_j \times R_{j'}$, we have that

$$\mu(W^{-1}[\lambda, 1] \cap A \times (C \setminus A)) \geq \mu(W^{-1}[\lambda, 1] \cap (R_j \cap A) \times R_{j'}) > 0.$$

- $\mu(R_{j'} \cap A) > 0$. Then $(R_{j'} \cap A) \times (R_j \setminus A) \subseteq A \times (C \setminus A)$ is a set of positive measure. Since W is at least λ a.e. on $R_{j'} \times R_j$, we have:

$$\mu(W^{-1}[\lambda, 1] \cap A \times (C \setminus A)) \geq \mu\left(W^{-1}[\lambda, 1] \cap (R_{j'} \cap A) \times (R_j \setminus A)\right) > 0.$$

Now consider the second case in which, for every $i \in [t]$, $\mu(R_i \cap A = 0)$ or $\mu(R_i \cap A) = \mu(R_i)$. There must exist a $j, j' \in [t]$ such that $|j - j'| = 1$, $\mu(R_j \cap A) = \mu(R_j)$, and $\mu(R_{j'} \cap A) = 0$. If this were not the case, then it would be that either $\mu(R_i \cap A) = \mu(R_i)$ for every $i \in [t]$, or $\mu(R_i \cap A) = 0$ for every $i \in [t]$. But the former of these would imply that $\mu(A) = \mu(C)$, and the latter would imply $\mu(A) = 0$, which we have assumed not to be the case.

Therefore, $R_j \times R_{j'} \subseteq A \times (C \setminus A)$, and this set is of positive measure. Since W is at least λ a.e. on $R_j \times R_{j'}$, we once again find

$$\mu(W^{-1}[\lambda, 1] \cap A \times (C \setminus A)) \geq \mu\left(W^{-1}[\lambda, 1] \cap R_j \times R_{j'}\right) > 0.$$

Hence, in every case it is true that $\mu(W^{-1}[\lambda, 1] \cap A \times (C \setminus A))$ has positive measure. Since A was arbitrary, C is connected at level λ. Hence $M \geq \lambda$ almost everywhere on $C \times C$. ■

In addition, we show that the mergeon does not vary too much on a block of the refinement:

Lemma 3.11. *Let $R, R' \in \mathcal{R}$. Let C be a cluster tree, and let λ be the greatest level at which there exists some cluster $\mathscr{C}$ containing a non-negligible piece of both R and R'. That is,*

$$\lambda = \sup\{\lambda' : \exists \mathscr{C} \in \mathsf{C}(\lambda') \text{ such that } \mu(R \cap \mathscr{C}) > 0 \text{ and } \mu(R' \cap \mathscr{C}) > 0.\}$$

Then $\lambda' - 2\Delta\mathsf{c} \leq M \leq \lambda$ almost everywhere on $R \times R'$.

Proof. By the definition of the mergeon it must be that $M \leq \lambda$ almost everywhere on $R \times R'$, since if there existed a $\lambda' > \lambda$ for which $M^{-1}[\lambda', 1] \cap R \times R'$ is not-null, this would imply that there exists some cluster at level λ' containing a non-negligible part of both R and R'.

Now, by Lemma 3.9, for any $\lambda' < \lambda$ there exists a $(\lambda' - 2\Delta\mathsf{c})$ path between R and R'. Hence, by Lemma 3.10, $M \geq \lambda' - 2\Delta\mathsf{c}$ almost everywhere on $R \times R'$ for any $\lambda' < \lambda$. ■

Putting these ideas together, we are able to bound the difference between the true merge height of points in a mergeon, and the merge estimate $\hat{Q}$.

Claim 3.9. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$ and let M be a mergeon of W. Let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$. Let $S = (x_1, \ldots, x_n)$ be an ordered set of elements of $[0, 1]$ such for any $R \in \mathcal{R}$, $R \cap S \neq \emptyset$. Let P be the edge probability matrix, i.e., the matrix whose (i, j) entry is given by $W(x_i, x_j)$, and suppose $\hat{P}$ is such that $\|\hat{P} - P\|_\infty < \epsilon$. Then $\max_{i \neq j} |M(x_i, x_j) - \hat{Q}_{ij}| \leq 4\Delta\mathsf{c} + \epsilon$.*

Proof. Consider an arbitrary $x_i, x_j \in S$. Let R_i and R_j be the blocks in $\mathcal{R}$ which contain x_i and x_j, respectively. Let λ^* be the greatest level at which there exists some cluster containing non-negligible parts of both R_i and R_j. Therefore, by Lemma 3.11, M is bounded below by $\lambda^* - 2\Delta\mathsf{c}$ and above by λ^* almost everywhere on $R_i \times R_j$.

First we bound $\hat{Q}_{ij}$ from below. By Lemma 3.9 there exists a $(\lambda' - 2\Delta\mathsf{c})$-path $\langle R_i = R_1, \ldots, R_t = R_j \rangle$ between R_i and R_j, for any $\lambda' < \lambda^*$. By the assumption on S, there

exists a sample from each element of the path, so that there is a path of samples $\langle x_i = x_{p_1}, \ldots, x_{p_k} = x_j \rangle$ with the property that, between any two consecutive elements in the path, we have $W(x_t, x_{t+1}) \geq \lambda' - 2\Delta\mathsf{c}$ for all $\lambda' < \lambda^*$. Hence $\hat{P}_{x_t, x_{t+1}} \geq \lambda^* - 2\Delta\mathsf{c} - \epsilon$. Therefore, there exists a path in H from x_i to x_j such that every edge has weight of at least $\lambda^* - 2\Delta - \epsilon$. As a result, $\hat{Q}_{ij} \geq \lambda^* - 2\Delta\mathsf{c} - \epsilon$.

We now bound $\hat{Q}_{ij}$ from above. Let $p = \langle x_i = x_1, \ldots, x_t = x_j \rangle$ be a path with cost $\hat{Q}_{ij}$. Let $\langle R_1, \ldots, R_t \rangle$ be the corresponding path of blocks from $\mathcal{R}$, such that $x_k \in R_k$. Then we have $\hat{P}_{x_k x_{k+1}} \geq \hat{Q}_{ij}$, so that $W(x_k, x_{k+1}) \geq \hat{Q}_{ij} - \epsilon$. Hence there is a point in $R_k \times R_{k+1}$ which is at least $\hat{Q}_{ij} - \epsilon$, and by smoothness it follows that $W \geq \hat{Q}_{ij} - 2\Delta c - \epsilon$ almost everywhere on $R_k \times R_{k+1}$. That is, $\langle R_1, \ldots, R_t \rangle$ is a $(\hat{Q}_{ij} - 2\Delta\mathsf{c} - \epsilon)$-path. Therefore, Lemma 3.10 implies that the mergeon M is at least $\hat{Q}_{ij} - 2\Delta\mathsf{c} - \epsilon$ almost everywhere on $R_i \times R_j$. However, by Lemma 3.11, $M \leq \lambda^*$ almost everywhere on $R_i \times R_j$. Therefore $\hat{Q}_{ij} \leq \lambda^* + 2\Delta\mathsf{c} + \epsilon$.

Combining the above bounds, we find that

$$|\hat{Q}_{ij} - \lambda^*| \leq 2\Delta\mathsf{c} + \epsilon.$$

The true merge height $M(x_i, x_j)$ is bounded between $\lambda^* - 2\Delta\mathsf{c}$ and λ^*, and so we have

$$|\hat{Q}_{ij} - M(x_i, x_j)| \leq 4\Delta\mathsf{c} + \epsilon.$$

■

Proof of Theorem 3.6.

We have bounded the difference between the merge estimate and the mergeon on a fixed sample S under the assumption that S contains a point from every block of a suitable refinement. We now reïntroduce randomness and show that such a sample occurs with high probability as $n \to \infty$. If the edge probability estimator $\hat{\mathbf{P}}$ is consistent in max-norm, it

follows that single-linkage clustering applied to $\hat{\mathbf{P}}$ is a consistent estimator of the graphon cluster tree.

Theorem 3.6. *Let W be a piecewise c-Lipschitz graphon. Let $\hat{\mathbf{P}}$ be a consistent estimator of $\mathbf{P}$ in max-norm. Let f be the clustering method which performs single-linkage clustering using $\hat{\mathbf{P}}$ as a similarity matrix. Then f is a consistent estimator of the graphon cluster tree of W in the sense of merge distortion.*

Proof. As stated, f is the clustering method which takes a graph G and returns the clustering $\mathcal{C}_{\hat{Q}}$ described at the beginning of the section – the single linkage clustering of the estimated edge probability matrix $\hat{\mathbf{P}}$. Let M be a mergeon of W. We will show that, for any $\epsilon > 0$.

$$\Lambda_{W,n}\left(\left\{(G,S) : d_S(M, \hat{M}_{f(G)\circ S}) > \epsilon\right\}\right) \to 0,$$

where $\hat{M}_{f(G)\circ S}$ is the merge height function induced on the clustering $f(G) \circ S$ by the mergeon M, and $\Lambda_{W,n}$ is the label measure as defined in Section 3.7.2.

First, fix any $\epsilon > 0$. Let $\tilde{\epsilon} = \epsilon/4$. Define

$$H_n = \left\{(G,S) \in \mathfrak{G}_n \times [0,1]^n : \max_{i \neq j} |\hat{P} - P| < \tilde{\epsilon}\right\},$$

where P is the edge probability matrix induced by S and $\hat{P}$ is the estimate of P computed from G. By the assumption that $\hat{\mathbf{P}}$ is consistent in ∞-norm, we have $\Lambda_{W,n}(H_n) \to 1$ as $n \to \infty$.

Now let $\Delta = \epsilon/16\mathsf{c}$. Let $\mathcal{B}$ be the block partition on which W is piecewise c-Lipschitz, and let $\mathcal{R}$ be an arbitrary Δ-refinement of $\mathcal{B}$. In order to apply Claim 3.9, we require that the labeling S satisfies the property that every block R in the refinement contains at least one point from S. The probability that a block R contains no points from a random sample $\mathbf{S}$ is $(1 - |R|)^n \leq (1 - \Delta/2)^n$, since $|R| \geq \Delta/2$. Now take a union bound over all blocks in the partition, of which there are at most $2/\Delta$. Hence the probability that there exists a

block in the partition that does not have a sample from S is at most $\frac{2}{\Delta}(1-\Delta/2)^n$. Let

$$F_n = \{(G,S) \in \mathfrak{G}_n \times [0,1]^n : |R \cap S| > 1 \text{ for all } R \in \mathcal{R}\}.$$

As per above, we have $\Lambda_{W,n}(F_n) = \frac{2}{\Delta}(1-\Delta/2)^n$, which tends to 0 as $n \to \infty$.

By Claim 3.9, for every $(G,S) \in H_n \setminus F_n$, we have that, for all $i \neq j \in [n] \times [n]$, writing $S = (x_1, \ldots, x_n)$:

$$|\hat{Q}_{ij} - M(x_i, x_j)| \leq 4\Delta\mathsf{c} + \tilde{\epsilon} = \epsilon/2,$$

where $\hat{Q}$ is the merge estimate between nodes i and j, described at the beginning of the section. The clustering method f uses $\hat{Q}$ to construct the clustering $\mathcal{C}_{\hat{Q}}$ Therefore, by Lemma 3.8, the merge distortion $d(M, \hat{M}_{f(G)\circ S})$ is bounded above by ϵ on the set $H_n \setminus F_n$. Since $\Lambda_{W,n}(H_n) \to 1$ and $\Lambda_{W,n}(F_n) \to 0$ as $n \to \infty$, we have $\Lambda_{W,n}(H_n \setminus F_n) \to 1$ as $n \to \infty$ and have thus proven the claim. ■

3.9 Consistency of neighborhood smoothing

In the previous section, sufficient conditions for a consistent graphon clustering algorithm were given. In particular, Theorem 3.6 shows that and edge probability estimator which is consistent in max-norm gives rise to a consistent clustering algorithm. In this section, we construct such a consistent estimator and thereby identify a consistent graph clustering algorithm.

Estimating the graphon W or the edge probability matrix $\mathbf{P}$ is an area of recent research. There are a number of methods in the literature; see, for instance, the works of Wolfe and Olhede (2013), Chan and Airoldi (2014), Airoldi et al. (2013), Rohe et al. (2011), and Zhang et al. (2015). To our knowledge, each work in this direction defines a slightly different sense in which the proposed estimator is consistent, but all use some variant of the mean squared error; i.e., estimators $\hat{\mathbf{P}}$ for which $1/n^2 \|\mathbf{P} - \hat{\mathbf{P}}\|_F^2 \to 0$ with high probability. Convergence in

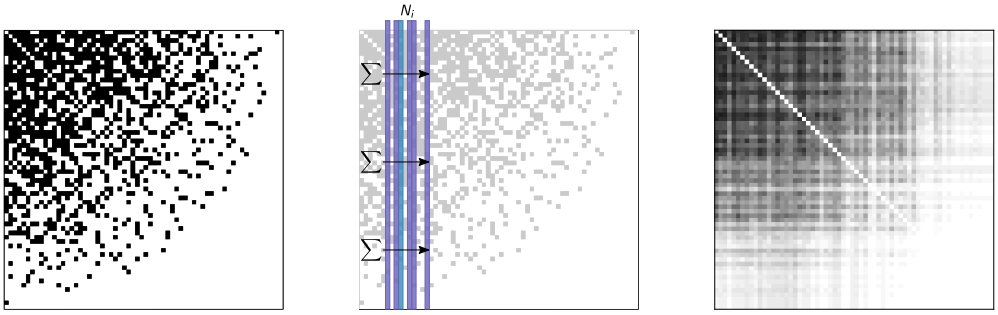


Figure 3.8: The neighborhood smoothing method of Zhang et al. (2015).

this norm ensures that the estimate is close to the true graphon in aggregate, but still allows the estimate to differ from the ground truth by a large amount on a set of small measure. Since our merge distortion is sensitive to the *largest* error, regardless of measure, consistency of graphon estimators as shown in the literature is not sufficient to show consistency in merge distortion.

One practical method of estimating $\mathbf{P}$ is the neighborhood smoothing algorithm of Zhang et al. (2015). The method constructs for each node i in the graph $\mathbf{G}$ a neighborhood of nodes $\mathbf{N_i}$ which are similar to i in the sense that for every $i' \in \mathbf{N_i}$, the corresponding column $\mathbf{A_{i'}}$ of the adjacency matrix is close to $\mathbf{A_i}$ in a particular distance. $\mathbf{A_{ij}}$ is clearly not a good estimate for the probability of the edge (i, j), as it is either zero or one, however, if the graphon is piecewise Lipschitz, the average of $\mathbf{A_{i'j}}$ over $i' \in \mathbf{N_{ij}}$ will intuitively tend to the true probability.

The neighborhood smoothing method is depicted in Figure 3.8. Figure (a) shows the input adjacency matrix; its entries are either zero or one. Figure (b) illustrates the smoothing process. The blue column represents the node i whose edge probabilities we would like to estimate. We first build a neighborhood of similar columns, represented by the purple bars in the image. We then average across this neighborhood to estimate the probability of an edge between node i and each of its neighbors. Figure (c) depicts the output of the algorithm: the smoothed edge probabilty estimate matrix whose entries are elements in the

unit interval.

Like other graphon estimators, the method of Zhang et al. (2015) is proven to be consistent in mean squared error. Since Theorem 3.6 requires consistency in max-norm, we analyze a modification of this algorithm and show that it consistently estimates $\mathbf{P}$ in this stronger sense. The proof of the following will constitute the remainder of the section:

Theorem 3.7. *If the graphon W is piecewise Lipschitz, the modified neighborhood smoothing algorithm in Section 3.9.2 is a consistent estimator of $\mathbf{P}$ in max-norm.*

This leads to the following practical and consistent graph clustering algorithm: first, we estimate the matrix $\hat{\mathbf{P}}$ of edge probabilities using the modified neighborhood smoothing method, then we apply single-linkage clustering to $\hat{\mathbf{P}}$. The pseudocode of this clustering algorithm is shown in Algorithm 1. As a corollary of Theorem 3.7 and Theorem 3.6, we find that Algorithm 1 is consistent:

Corollary 3.1. *If the graphon W is piecewise Lipschitz, Algorithm 1 is a consistent estimator of the graphon cluster tree of W.*

We now turn to proving Theorem 3.7.

3.9.1 The method of Zhang et al. (2015)

Theorem 3.6 states sufficient conditions under which an estimator $\hat{\mathbf{P}}$ of the edge probability matrix leads to a consistent clustering algorithm. In particular, if the graphon W is piecewise Lipschitz, and if for any $\epsilon > 0$,

$$\lim_{n\to\infty} \mathbb{P}(\max_{i\neq j} |\mathbf{P}_{ij} - \hat{\mathbf{P}}_{ij}| > \epsilon) = 0,$$

then one consistent clustering algorithm is that which applies single-linkage clustering to the estimate $\hat{\mathbf{P}}$. In this section, we analyze a modification of the edge probability estimator

Algorithm 1 Clustering by nbhd. smoothing

Require: Adjacency matrix A, $C \in (0,1)$
  % Step 1: Compute the estimated edge
  % probability matrix $\hat{P}$ using neighborhood
  % smoothing algorithm based on Zhang et al. (2015)
  $n \leftarrow \textsc{Size}(A)$
  $h \leftarrow C\sqrt{(\log n)/n}$
  for $i \neq j \in [n] \times [n]$ **do**
    $\hat{A} \leftarrow A$ after setting row/column j to zero
    for $i' \in [n] \setminus \{i, j\}$ **do**
      $d_j(i, i') \leftarrow \max_{k \neq i, i', j} |(\hat{A}^2/n)_{ik} - (\hat{A}^2/n)_{i'k}|$
    end for
    $q_{ij} \leftarrow h$th quantile of $\{d_j(i, i') : i' \neq i, j\}$
    $N_{ij} \leftarrow \{i' \neq i, j : d_j(i, i') \leq q_{ij}(h)\}$
  end for
  for $(i, j) \in [n] \times [n]$ **do**
    $\hat{P}_{ij} \leftarrow \frac{1}{2}\left(\frac{1}{N_{ij}} \sum_{i' \in N_{ij}} A_{i'j} + \frac{1}{N_{ji}} \sum_{j' \in N_{ji}} A_{ij'}\right)$
  end for
  % Step 2: Cluster $\hat{P}$ with single-linkage
  $\mathcal{C} \leftarrow$ the single linkage clusters of $\hat{P}$
  return $\mathcal{C}$

introduced in Zhang et al. (2015) and show that it satisfies the above condition. Combining this result with Theorem 3.6 shows that the single-linkage clustering applied this estimate of the edge probability matrix is a consistent clustering algorithm.

The aim of the neighborhood smoothing method of Zhang et al. (2015) is to estimate the random edge probability matrix $\mathbf{P}$. In particular, the method defines a distance $d(i, i')$ between the columns of the random adjacency matrix $\mathbf{A}$ as such:

$$d(i, i') = \frac{1}{n} \max_{k \neq i, i'} |\langle \mathbf{A}_i - \mathbf{A}_{i'}, \mathbf{A}_k \rangle| = \max_{k \neq i, i'} |(\mathbf{A}^2/n)_{ik} - (\mathbf{A}^2/n)_{i'k}|.$$

The neighborhood $\mathcal{N}_i(\mathbf{A})$ of node i then consists of all nodes i' such that $d(i, i')$ is below the h-th quantile of $\{d(i, k)\}_{k \neq i}$, where h is a parameter of the algorithm. Note that $\mathcal{N}_i(\mathbf{A})$ is a random set, as the neighborhood around node i depends on the random adjacency matrix $\mathbf{A}$. For simplicity, however, we will often omit the explicit dependence on $\mathbf{A}$.

The estimate of the probability of the edge (i, j), written $\hat{\mathbf{P}}_{ij}$, is then computed by smoothing over the neighborhoods $\mathcal{N}_i$ and $\mathcal{N}_j$:

$$\hat{\mathbf{P}}_{ij} = \frac{1}{2}\left(\frac{1}{|\mathcal{N}_i|}\sum_{i' \in \mathcal{N}_i} \mathbf{A}_{i'j} + \frac{1}{|\mathcal{N}_j|}\sum_{j' \in \mathcal{N}_j} \mathbf{A}_{ij'}\right).$$

If it is assumed that $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$, and h is set to be $C_0\sqrt{\log n/n}$ for arbitrary constant C_0, where n is the size of the sampled graph, then the method is consistent in mean square error. That is, for any $\epsilon > 0$, as $n \to \infty$

$$\mathbb{P}\left(\frac{1}{n^2}\|\hat{\mathbf{P}} - \mathbf{P}\|_F^2 > \epsilon\right) \to 0$$

3.9.2 Our modification

In order to construct an algorithm which is a consistent estimator of the graphon cluster tree in the sense made precise above, we need for the edge probability estimator to be consistent in a stronger sense. In particular, we need that for any $\epsilon > 0$, as $n \to \infty$

$$\mathbb{P}\left(\max_{i \neq j} |\hat{\mathbf{P}}_{ij} - \mathbf{P}_{ij}| > \epsilon\right) \to 0.$$

In order to show that the neighborhood smoothing method satisfies such a notion of consistency, one might attempt to apply a concentration inequality to bound the difference between $\frac{1}{|\mathcal{N}_i|}\sum_{i' \in \mathcal{N}_i} \mathbf{A}_{i'j}$ and $\mathbf{P}_{ij}$. The difficulty with this approach, however, is that such concentration results require an assumption of statistical independence that is not satisfied by the neighborhoods as defined; that is, the terms of the sum $\sum_{i' \in \mathcal{N}_i} \mathbf{A}_{i'j}$ are not statistically independent. It is true that, unconditioned, $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$ are independent Bernoulli random variables. However, once we condition on the event $i' \in \mathcal{N}_i$ and $i'' \in \mathcal{N}_i$, the random variables $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$ are no longer independent.

More precisely, we are interested in

$$\mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j} \mid i', i'' \in \mathcal{N}_i) = \frac{\mathbb{P}(i', i'' \in \mathcal{N}_i \mid \mathbf{A}_{i'j}, \mathbf{A}_{i''j})\mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j})}{\mathbb{P}(i', i'' \in \mathcal{N}_i)}. \tag{3.2}$$

The denominator of the RHS is a normalization constant which does not depend on $\mathbf{A}_{i'j}$ or $\mathbf{A}_{i''j}$. Moreover, the entries of $\mathbf{A}$ are independent when unconditioned, and so $\mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j}) = \mathbb{P}(\mathbf{A}_{i'j})\mathbb{P}(\mathbf{A}_{i''j})$. The difficulty is in computing $\mathbb{P}(i', i'' \in \mathcal{N}_i \mid \mathbf{A}_{i'j}, \mathbf{A}_{i''j})$. Intuitively, the event $i' \in \mathcal{N}_i$ depends on $\mathbf{A}_{i'j}$, and, likewise, $i'' \in \mathcal{N}_i$ depends on $\mathbf{A}_{i''j}$. This is because $i' \in \mathcal{N}_i$ when $d(i, i')$ is small. But $d(i, i')$ depends on $\mathbf{A}_{i'j}$, since

$$\begin{aligned} d(i, i') &= \max_{k \neq i, i'} |(\mathbf{A}^2/n)_{ik} - (\mathbf{A}^2/n)_{i'k}|, \\ &= \frac{1}{n} \max_{k \neq i, i'} \left| \sum_{\ell=1}^{n} \mathbf{A}_{k\ell} \left(\mathbf{A}_{i\ell} - \mathbf{A}_{i'\ell} \right) \right|. \end{aligned}$$

and so $\mathbf{A}_{i'j}$ enters the sum and $d(i, i')$ depends on it. In the extreme case, suppose there are two nodes i' and i'' such that the row vectors $\mathbf{A}_{i'}$ and $\mathbf{A}_{i''}$ are identical except in their jth component. Then the only difference between $d(i, i')$ and $d(i, i'')$ comes from the difference in $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$. Hence it is clear that $d(i, i')$ and $d(i, i'')$ depend on the values of $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$, and, by extension, the events $i' \in \mathcal{N}_i$ and $i'' \in \mathcal{N}_i$ are *not* independent of $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$.

Our modification of the algorithm is to change the way in which neighborhoods are constructed so that statistical independence is ensured. Instead of constructing a neighborhood for each node i, we construct a neighborhood $\mathcal{N}_{i\backslash j}$ for each *ordered pair* (i, j) by using a parameterized distance function d_j which ignores all information about node j. More precisely, let $\partial_j \mathbf{A}$ represent the matrix obtained by setting the jth row and column of $\mathbf{A}$ to

zero. Then for every node j we define

$$\begin{aligned} d_j(i,i') &= \max_{k \neq i,i',j} |([\partial_j \mathbf{A}]^2/n)_{ik} - ([\partial_j \mathbf{A}]^2/n)_{i'k}|, \\ &= \frac{1}{n} \max_{k \neq i,i',j} \left| \sum_{\substack{\ell=1 \\ \ell \neq j}}^{n} \mathbf{A}_{k\ell} \left(\mathbf{A}_{i\ell} - \mathbf{A}_{i'\ell} \right) \right|. \end{aligned}$$

Observe that $\mathbf{A}_{i'j}$ does not appear in $d_j(i,i')$, and, since the other entries of $\mathbf{A}$ are independent of $\mathbf{A}_{i'j}$, we have that $d_j(i,i')$ is statistically independent of $\mathbf{A}_{i'j}$. Therefore the event $i' \in \mathcal{N}_{i\backslash j}$ is independent of $\mathbf{A}_{ij}$.

We are now interested in the quantity:

$$\mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j} \mid i', i'' \in \mathcal{N}_{i\backslash j}) = \frac{\mathbb{P}(i', i'' \in \mathcal{N}_{i\backslash j} \mid \mathbf{A}_{i'j}, \mathbf{A}_{i''j}) \mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j})}{\mathbb{P}(i', i'' \in \mathcal{N}_{i\backslash j})}, \tag{3.3}$$

where we are using the parameterized distance d_j to construct the neighborhood $\mathcal{N}_{i\backslash j}$. In this case, we apply the independence argument above to see that

$$\mathbb{P}(i', i'' \in \mathcal{N}_{i\backslash j} \mid \mathbf{A}_{i'j}, \mathbf{A}_{i''j}) = \mathbb{P}(i', i'' \in \mathcal{N}_{i\backslash j}).$$

Therefore the denominator cancels with the term in the numerator, and we have

$$\mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j} \mid i', i'' \in \mathcal{N}_{i\backslash j}) = \mathbb{P}(\mathbf{A}_{i'j}, \mathbf{A}_{i''j}) = \mathbb{P}(\mathbf{A}_{i'j}) \mathbb{P}(\mathbf{A}_{i''j}). \tag{3.4}$$

As a result, $\mathbf{A}_{i'j}$ and $\mathbf{A}_{i''j}$ are independent even when conditioning on the event $i' \in \mathcal{N}_{i\backslash j}$ and $i'' \in \mathcal{N}_{i\backslash j}$. This allows us to apply a concentration inequality to bound each entry of $\hat{\mathbf{P}} - \mathbf{P}$, and a max norm result follows after a simple union bound.

In total, the modified neighborhood smoothing procedure is as follows: Fix some neighborhood size parameter h and let $q_{i\backslash j}(h)$ denote the h-th quantile of the set $\{d_j(i,i') : i' \neq$

$i, j\}$. Construct the neighborhood $\mathcal{N}_{i\setminus j}$ by setting

$$\mathcal{N}_{i\setminus j} = \{i' \neq i, j : d_j(i, i') \leq q_{i\setminus j}(h)\}.$$

Then set

$$\hat{\mathbf{P}}_{ij} = \frac{1}{2}\left(\frac{1}{|\mathcal{N}_{i\setminus j}|}\sum_{i'\in\mathcal{N}_{i\setminus j}} \mathbf{A}_{i'j} + \frac{1}{|\mathcal{N}_{j\setminus i}|}\sum_{j'\in\mathcal{N}_{j\setminus i}} \mathbf{A}_{ij'}\right).$$

We will show that this estimator of the edge probability matrix is consistent in max-norm.

3.9.3 Proof

There are two major components to the analysis. First, we show that, with high probability, each neighborhood $\mathcal{N}_{i\setminus j}$ consists only of nodes i' for which $\|\mathbf{P}_i - \mathbf{P}_{i'}\|_\infty < \epsilon$, with $\epsilon \to 0$ as $n \to \infty$; The formal statement of this result is made in Lemmas 3.13 and 3.14 below. This is an extension of the analysis in Zhang et al. (2015), where it is shown that the neighborhood $\mathcal{N}_i$ consists only of nodes i' for which $1/n\,\|\mathbf{P}_i - \mathbf{P}_{i'}\|_2 < \epsilon$, with $\epsilon \to 0$ as $n \to \infty$. The procedure for proving this result parallels that of Zhang et al. (2015), however, the modifications we make to the algorithm – namely, the deletion of a node from the graph – mean that the claims in that paper do not directly transfer. Much of the analysis consists of making the minor changes necessary to show that analogous versions of the claims in Zhang et al. (2015) hold for our modified algorithm.

The second part of the analysis uses concentration inequalities to derive the consistency result. In particular, Lemma 3.15 shows that smoothing within neighborhoods produces an estimate of the edge probability matrix which is close within max-norm, provided that each neighborhood consists only of nodes which are sufficiently similar in the sense described above. Theorem 3.8 puts these two claims together to derive the main result.

We will make use of several minor technical results throughout the proof. For clarity, these results are collected in Section 3.9.4.

Sample requirements.

The analysis will require the notion of a block partition and Δ-refinement as defined in Definition 3.13 and Definition 3.14, respectively, both in Section 3.8. If $\mathcal{R}$ is a block partition and $x \in [0,1]$, we write $\mathcal{R}(x)$ to denote the block $R \in \mathcal{R}$ which contains x. Some of the following results will include an assumption that there are "enough" samples in each block of a partition. We formalize this notion as follows:

Definition 3.16. If $\mathbf{S}$ is an ordered set of random samples from the uniform distribution on the unit interval, and $\mathcal{B}$ is any block partition, we say that $\mathbf{S}$ is a *ρ-dense sample in $\mathcal{B}$* if for any block $B \in \mathcal{B}$,

$$\frac{|B \cap \mathbf{S}|}{n} > (1-\rho)\mu(B).$$

If we fix any ρ and a Δ-block partition, a random sample $\mathbf{S}$ will be ρ-dense with high probability as the size of the sample $n \to \infty$, as the following result shows:

Claim 3.10. *Let $\mathcal{B}$ be a Δ-block partition. Let $\rho < 1$. Then with probability $1 - \frac{2}{\Delta} e^{-2n\rho^2\Delta^2}$, $\mathbf{S}$ is a ρ-dense sample of $\mathcal{B}$. That is, for all $B \in \mathcal{B}$ simultaneously,*

$$\frac{|B \cap \mathbf{S}|}{n} > (1-\rho)|B|.$$

Proof. Let B be an arbitrary block in the partition $\mathcal{B}$. Since $\mathcal{B}$ is a Δ-partition, the size of any block is between $\Delta/2$ and Δ. Therefore there are at most $2/\Delta$ blocks in $\mathcal{B}$.

The membership of any given sample in B is a Bernoulli trial with probability $|B|$ of success. Applying Hoeffding's inequality:

$$\mathbb{P}\left(\left|\frac{1}{n}|B \cap \mathbf{S}| - |B|\right| > \epsilon\right) < e^{-2n\epsilon^2}.$$

Choose $\epsilon = \rho\Delta$. This gives

$$\mathbb{P}\left(\left|\frac{1}{n}\left|B \cap \mathbf{S}\right| - |B|\right| > \rho\Delta\right) < e^{-2n\rho^2\Delta^2},$$

which implies

$$\mathbb{P}\left(\frac{1}{n}\left|B \cap \mathbf{S}\right| > |B| - \rho\Delta\right) < e^{-2n\rho^2\Delta^2}.$$

Now, $|B| \leq \Delta$, so that for any arbitrary B it is true that

$$\mathbb{P}\left(\frac{1}{n}\left|B \cap \mathbf{S}\right| > |B| - \rho|B|\right) < e^{-2n\rho^2\Delta^2}.$$

The result follows by applying a union bound over all blocks of the partition, of which there are at most $2/\Delta$. ■

The adjacency column distance.

In the previous section detailing our modified neighborhood smoothing algorithm, we motivated a new distance $d_j(i, i')$ between columns of the adjacency matrix. This distance is computed by first deleting all information about node j from the adjacency matrix. We achieve this by setting the jth column and row of the adjacency matrix to zero. Because this will be a common operation in our proof, we define the following notation:

Definition 3.17. For a square matrix M, let $\partial_v M$ denote the matrix obtained by replacing the v-th row and column of the matrix M with zeros.

With this notation, the distance between node i and i' is written

$$d_j(i, i') = \max_{k \neq i, i'} \left| \left[\left(\partial_j \mathbf{A}\right)^2 / n \right]_{ik} - \left[\left(\partial_j \mathbf{A}\right)^2 / n \right]_{i'k} \right|.$$

This pattern – the maximum elementwise difference of normalized squared matrices – will reoccur in the analysis. We therefore make the following definition:

Definition 3.18. Let M_1 and M_2 be $n \times n$ matrices. We define

$$D(M_1, M_2) = \max_{i,j} \left| \left[M_1^2/n\right]_{ij} - \left[M_2^2/n\right]_{ij} \right|.$$

A key observation in the analysis of Zhang et al. (2015) is that if $\mathbf{A}$ is sampled from P, then for any fixed $\epsilon > 0$, $\mathbb{P}(D(\mathbf{A}, P) < \epsilon) \to 0$ as $n \to \infty$. In our analysis, however, we will work with $\partial_k \mathbf{A}$ and $\partial_k P$, which are the adjacency and edge probability matrices with the kth row and column set to zero. We therefore have a slightly modified claim:

Lemma 3.12. *Let P be an arbitrary $n \times n$ edge probability matrix. Let $C_2 > 0$ be an arbitrary constant and suppose n is large enough that $\sqrt{\frac{(C_2+2)\log n}{n}} \leq 1$. Then, with probability $1 - 2n^{-C_2/4}$ over random adjacency matrices $\mathbf{A}$ sampled from P, for all $k \in [n]$ simultaneously,*

$$D(\partial_k \mathbf{A}, \partial_k P) = \max_{i \neq j} \left| \left[\left(\partial_k \mathbf{A}\right)^2 / n\right]_{ij} - \left[\left(\partial_k P\right)^2 / n\right]_{ij} \right| \leq \sqrt{\frac{(C_2+2)\log n}{n}} + \frac{6}{n}.$$

Proof. The proof of Lemma 5.2 in Zhang et al. (2015) establishes that, given the above assumptions, with probability $1 - 2n^{C_2/4}$,

$$D(\mathbf{A}, P) = \max_{i \neq j} \left| \left[\mathbf{A}^2/n\right]_{ij} - \left[P^2/n\right]_{ij} \right| \leq \sqrt{\frac{(C_2+2)\log n}{n}} + \frac{4}{n}.$$

From Claim 3.13, for all k,

$$\left| \left[\mathbf{A}^2/n\right]_{ij} - \left[\left(\partial_k \mathbf{A}\right)^2 / n\right]_{ij} \right| \leq \frac{1}{n},$$
$$\left| \left[P^2/n\right]_{ij} - \left[\left(\partial_k P\right)^2 / n\right]_{ij} \right| \leq \frac{1}{n},$$

and so, with probability $1 - 2n^{-C_2/4}$,

$$\max_{i \neq j} \left| \left[(\partial_k \mathbf{A})^2 / n \right]_{ij} - \left[(\partial_k P)^2 / n \right]_{ij} \right| \leq \max_{i \neq j} \left| \left[\mathbf{A}^2 / n \right]_{ij} - \left[P^2 / n \right]_{ij} \right| + \frac{2}{n}$$
$$\leq \sqrt{\frac{(C_2 + 2) \log n}{n}} + \frac{6}{n}.$$

■

Composition of neighborhoods.

Another key step in the analysis of Zhang et al. (2015) is that, with high probability, for any i' in the neighborhood of node i, $1/n \left\| P_{i'} - P_i^2 \right\|_2 = O(\sqrt{\log n / n})$. We derive a similar result for our modified neighborhoods:

Lemma 3.13. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$ and let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$. Suppose S is a ρ-dense sample of $\mathcal{R}$ and let P be the induced edge probability matrix. Suppose A is an adjacency matrix such that $D(\partial_k A, \partial_k P) < \epsilon$ for every $k \in [n]$. Pick $0 < h \leq \rho\Delta$, and construct for every pair i, j a neighborhood $\mathcal{N}_{i \setminus j}$ as described above, including all nodes within the h-th quantile. Then for all i, j and any $i' \in \mathcal{N}_{i \setminus j}$ we have*

$$\frac{1}{n} \left\| P_i - P_{i'} \right\|_2^2 \leq 6\mathsf{c}\Delta + 8\epsilon + \frac{5}{n}.$$

Proof. We start by applying Claim 3.15, which yields

$$\frac{1}{n} \left\| P_i - P_{i'} \right\|_2^2 \leq 2 d_j(i, i') + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon.$$

We now upper bound $d_j(i, i')$. Since we have assumed that $h \leq \rho$, at least a fraction h of the nodes are within i's partition in the refinement. Therefore the distance between any two nodes in the neighborhood is bounded above by the maximum distance between two

nodes in this partition. This is computed in Claim 3.16, such that:

$$\begin{aligned}\frac{1}{n}\left\|P_i - P_{i'}\right\|_2^2 &\leq 2\left(\mathsf{c}\Delta + 2\epsilon + \frac{2}{n}\right) + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon \\ &= 6\mathsf{c}\Delta + 8\epsilon + \frac{5}{n}.\end{aligned}$$

■

Additionally, we prove that neighborhoods are composed of nodes whose corresponding columns of P are close in ∞-norm. This follows from the previous claim after leveraging the piecewise Lipschitz condition.

Lemma 3.14. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$ and let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$. Suppose S is a ρ-dense sample of $\mathcal{R}$. Then for any $\epsilon \geq 4\rho\Delta^3\mathsf{c}^2$, if $i \neq j$ are such that $\frac{1}{n}\left\|P_i - P_j\right\|_2^2 \leq \epsilon$, then $\left\|P_i - P_j\right\|_\infty^2 \leq \frac{4\epsilon}{\rho\Delta}$.*

Proof. Suppose $\epsilon \geq 4\rho\Delta^3\mathsf{c}^2$. Define $\alpha = \sqrt{4\epsilon/(\rho\Delta)}$ and suppose that $\left\|P_i - P_j\right\|_\infty > \alpha$. This implies that there exists a k such that $|P_{ik} - P_{jk}| > \alpha$. Consider any $k' \in \mathcal{R}(k)$. Then

$$\begin{aligned}|P_{ik'} - P_{jk'}| &= \left|(P_{ik'} - P_{ik}) + P_{ik} - (P_{jk'} - P_{jk}) - P_{jk}\right| \\ &= \left|(P_{ik'} - P_{ik}) + (P_{jk} - P_{jk'}) + (P_{ik} - P_{jk})\right|\end{aligned}$$

Since $|x_k - x_{k'}| < \Delta$ by virtue of being in the same block $\mathcal{R}(x_k)$, we have $|P_{ik'} - P_{ik}| \leq \Delta\mathsf{c}$. But by assumption, $\Delta \leq \alpha/(4\mathsf{c})$. Therefore, $|P_{ik'} - P_{ik}| \leq \alpha/4$. Similarly, $|P_{jk'} - P_{jk}| \leq \alpha/4$. The last term satisfies $|P_{ik} - P_{jk}| > \alpha$. Therefore the entire quantity must be at least:

$$> \alpha/2.$$

Now consider

$$\frac{1}{n}\|P_i - P_j\|_2^2 = \frac{1}{n}\sum_l (P_{il} - P_{jl})^2,$$
$$\geq \frac{1}{n}\sum_{k' \in \mathcal{R}(k)} (P_{ik'} - P_{jk'})^2.$$

But, as established above, each term in the sequence is at least $\alpha/2$, and so:

$$> \frac{1}{n}\sum_{k' \in \mathcal{R}(k)} \alpha^2/4.$$

Since S is assumed to be a ρ-dense sample of $\mathcal{R}$, there are at least $\rho\Delta n$ elements in $\mathcal{R}(k)$. Therefore:

$$\geq \frac{\rho\Delta\alpha^2}{4} = \epsilon.$$

The claim follows from the contrapositive. ■

Main result.

Intuitively, if every neighborhood $\mathcal{N}_{i\backslash j}$ is composed of nodes whose corresponding columns of P are close in ∞-norm, and whose jth elements are statistically independent, we may apply a concentration inequality to conclude that the estimate $\hat{\mathbf{P}}_{ij}$ is close to P_{ij}. The following claim makes this precise.

Lemma 3.15. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$ and let $\mathcal{R}$ be a Δ-refinement of $\mathcal{B}$. Let $S \in [0,1]^n$ be fixed, and let P be the edge probability matrix induced by S. Assume that with probability $1-\delta$ over graphs generated from P, that for all $i \neq j$ simultaneously, $\|P_i - P_{i'}\|_\infty < \epsilon$ for all $i' \in \mathcal{N}_{i\backslash j}$.*

Then with probability at least $(1-\delta)\left[1-2n(n-1)e^{-2hnt^2}\right]$,

$$\max_{ij}\left|\hat{\mathbf{P}}_{ij}-P_{ij}\right|<\epsilon+t.$$

Proof. Consider an arbitrary ordered pair of nodes $i \neq j$. The neighborhood $\mathcal{N}_{i\setminus j}$ is a random variable, since it depends on the random adjacency matrix $\mathbf{A}$. Define $\ell_{i\setminus j}$ to be the amount by which our smoothed estimate computed using $\mathcal{N}_{i\setminus j}$ differs from P_{ij}:

$$\ell_{i\setminus j}=\left|P_{ij}-\frac{1}{\mathcal{N}_{i\setminus j}}\sum_{i'\in\mathcal{N}_{i\setminus j}}\mathbf{A}_{i'j}\right|.$$

Note that $\ell_{i\setminus j}$ is itself a random variable, and we seek to compute

$$\mathbb{P}\left(\max_{i\neq j}\ell_{i\setminus j}<\tilde{\epsilon}\right),$$

where it will be assumed that $\tilde{\epsilon}>\epsilon$.

Denote by $\mathscr{N}_{i\setminus j}$ the subset of $2^{[n]}$ consisting of all possible values of the neighborhood $\mathcal{N}_{i\setminus j}$ over all graphs on $[n]$. Denote by $\mathscr{N}^{\epsilon}_{i\setminus j}$ the subset of $\mathscr{N}_{i\setminus j}$ consisting of neighborhoods with the property that that if i' is in the neighborhood, then $\|P_i-P_{i'}\|_\infty<\epsilon$. Then

$$\begin{aligned}\mathbb{P}\left(\max_{i\neq j}\ell_{i\setminus j}<\tilde{\epsilon}\right)&\geq\mathbb{P}\left(\max_{i\neq j}\ell_{i\setminus j}<\tilde{\epsilon}\,\middle|\,\forall\, i\neq j,\,\mathcal{N}_{i\setminus j}\in\mathscr{N}^{\epsilon}_{i\setminus j}\right)\mathbb{P}\left(\forall\, i\neq j,\,\mathcal{N}_{i\setminus j}\in\mathscr{N}^{\epsilon}_{i\setminus j}\right)\\&\geq\mathbb{P}\left(\max_{i\neq j}\ell_{i\setminus j}<\tilde{\epsilon}\,\middle|\,\forall\, i\neq j,\,\mathcal{N}_{i\setminus j}\in\mathscr{N}^{\epsilon}_{i\setminus j}\right)(1-\delta).\end{aligned}$$

We now lower bound the probability that an arbitrary pair $u \neq v$ is such that $\ell_{u\setminus v}<\tilde{\epsilon}$. The result will then follow from a union bound. That is, we would like to compute, for arbitrary $u \neq v$, the probability

$$\mathbb{P}\left(\ell_{u\setminus v}<\tilde{\epsilon}\,\middle|\,\forall\, i\neq j,\,\mathcal{N}_{i\setminus j}\in\mathscr{N}^{\epsilon}_{i\setminus j}\right)=\mathbb{P}\left(\ell_{u\setminus v}<\tilde{\epsilon}\,\middle|\,\mathcal{N}_{u\setminus v}\in\mathscr{N}^{\epsilon}_{u\setminus v}\right)$$

We decompose this quantity as a sum over all neighborhoods in $\mathscr{N}^{\epsilon}_{u\backslash v}$:

$$= \sum_{N \in \mathscr{N}^{\epsilon}_{u\backslash v}} \mathbb{P}\left(\ell_{u\backslash v} < \tilde{\epsilon} \,\middle|\, \mathcal{N}_{u\backslash v} = N\right) \mathbb{P}\left(\mathcal{N}_{u\backslash v} = N \,\middle|\, \mathcal{N}_{u\backslash v} \in \mathscr{N}^{\epsilon}_{u\backslash v}\right)$$

We now claim that, conditioned on a particular neighborhood N, the random variables $\mathbf{A}_{u_1 v}$ and $\mathbf{A}_{u_2 v}$ are independent. We may then apply Hoeffding's inequality to conclude:

$$\mathbb{P}\left(\left|\frac{1}{|N|}\sum_{u' \in N}(\mathbf{A}_{u'v} - P_{u'v})\right| > t\right) < e^{-2hnt^2}.$$

Where we have used the fact that there are at least hn nodes in the neighborhood N. By the assumption that $|P_{uv} - P_{u'v}| < \epsilon$ for any $u \in N$, we have:

$$\mathbb{P}\left(\left|P_{uv} - \frac{1}{|N|}\sum_{u' \in N}\mathbf{A}_{u'v}\right| > t + \epsilon\right) < e^{-2hnt^2}.$$

So that

$$\begin{aligned}
\mathbb{P}\left(\ell_{u\backslash v} < \tilde{\epsilon} \,\middle|\, \forall\, i \neq j,\, \mathcal{N}_{i\backslash j} \in \mathscr{N}^{\epsilon}_{i\backslash j}\right) &= \sum_{N \in \mathscr{N}^{\epsilon}_{u\backslash v}} \mathbb{P}\left(\ell_{u\backslash v} < \tilde{\epsilon} \,\middle|\, \mathcal{N}_{u\backslash v} = N\right) \mathbb{P}\left(\mathcal{N}_{u\backslash v} = N \,\middle|\, \mathcal{N}_{u\backslash v} \in \mathscr{N}^{\epsilon}_{u\backslash v}\right) \\
&> \left(1 - e^{-2hnt^2}\right) \sum_{N \in \mathscr{N}^{\epsilon}_{u\backslash v}} \mathbb{P}\left(\mathcal{N}_{u\backslash v} = N \,\middle|\, \mathcal{N}_{u\backslash v} \in \mathscr{N}^{\epsilon}_{u\backslash v}\right) \\
&= \left(1 - e^{-2hnt^2}\right)
\end{aligned}$$

Now, returning to:

$$\mathbb{P}\left(\max_{i \neq j} \ell_{i\backslash j} < \tilde{\epsilon}\right) \geq \mathbb{P}\left(\max_{i \neq j} \ell_{i\backslash j} < \tilde{\epsilon} \,\middle|\, \forall\, i \neq j,\, \mathcal{N}_{i\backslash j} \in \mathscr{N}^{\epsilon}_{i\backslash j}\right)(1 - \delta)$$

We apply a union bound over all $2n(n-1)$ ordered pairs to obtain:

$$> (1-\delta)\left(1 - 2n(n-1)\left[1 - e^{-2hnt^2}\right]\right).$$

■

We combine all of the previous results to derive our main result.

Theorem 3.8. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathfrak{c}}$. Let $\mathbf{P}$ be the random edge probability matrix arising by sampling a graph of size n from W according to the graphon sampling procedure, and denote by $\hat{\mathbf{P}}$ the estimated edge probability using our modified neighborhood smoothing method. Then*

$$\max_{i \neq j} \left|\hat{\mathbf{P}}_{ij} - \mathbf{P}_{ij}\right| = O_p\left(\left[\frac{\log n}{n}\right]^{1/6}\right).$$

Proof. The mechanism of the proof involves a translation from the L^2 result of Zhang et al. (2015) to our desired max-norm result. To accomplish this, we will make use of two discretizations at different scales. First, define arbitrary constants $\alpha_2, \alpha_\infty > 0$ and $0 < \rho < 1$ such that $\rho \cdot \alpha_2 > \frac{1}{2}$, and let

$$\Delta_2(n) = \alpha_2\sqrt{\frac{\log n}{n}}, \qquad \Delta_\infty(n) = \alpha_\infty\left(\frac{\log n}{n}\right)^{1/6},$$

for any $n \geq 2$. For each $n \geq 2$, let $\mathcal{R}_\infty(n)$ be an arbitrary $\Delta_\infty(n)$-refinement of $\mathcal{B}$, and let $\mathcal{R}_2(n)$ be an arbitrary $\Delta_2(n)$-refinement of $\mathcal{R}_\infty(n)$. In what follows we will drop the functional notation, as the dependence of these quantities on n should be clear.

Let $\mathbf{S}$ be a random sample of $[0,1]$. Then, according to Claim 3.10, $\mathbf{S}$ is ρ-dense in $\mathcal{R}_2$ with probability

$$\begin{aligned}
1 - \frac{2}{\Delta_2} e^{-2n\rho^2\Delta_2^2} &= 1 - 2\alpha_2 \sqrt{\frac{n}{\log n}} e^{-2n\rho^2\alpha_2^2 \frac{\log n}{n}}, \\
&= 1 - 2\alpha_2 \sqrt{\frac{n}{\log n}} n^{-2\alpha_2^2\rho^2}, \\
&\geq 1 - 2\alpha_2 \sqrt{n} \cdot n^{-2\alpha_2^2\rho^2}, \\
&= 1 - 2\alpha_2 n^{\frac{1}{2} - 2\alpha_2^2\rho^2}.
\end{aligned}$$

Since $\rho \cdot \alpha_2 > 1/2$ by assumption, this is a decreasing function in n.

We have so-far shown that a sample is "good" with high probability in the sense that it is ρ-dense in $\mathcal{R}_2$. We now show that, assuming the sample $\mathbf{S} = S$ is a fixed, ρ-dense sample of $\mathcal{R}_2$, the estimate $\hat{P}$ is good in max-norm with high probability over random graphs sampled according to the distribution induced by S.

We begin by showing that, with high probability, the neighborhood around node i contains only nodes i' such that P_i and $P_{i'}$ are close in 2-norm, which will follow from combining Lemmas 3.12 and 3.13. We will use this result to invoke Lemma 3.14, which says that, for i' in the neighborhood of i, P_i and $P_{i'}$ are close in ∞-norm. This will in turn satisfy the assumptions of Lemma 3.15, which shows that $\hat{P}$ is close to P.

First, we combine Lemmas 3.12 and 3.13 to show that, with high probability, $\frac{1}{n}\|P_i - P_{i'}\|_2^2$ is small when i' is in $\mathcal{N}_{i\setminus j}$. Fix an arbitrary constant $C_2 > 0$ and suppose that n is large enough that $\sqrt{(C_2+2)\log n/n} \leq 1$. Then Lemma 3.12 says that, with probability $1 - 2n^{-C_2/4}$ over random adjacency matrices $\mathbf{A}$ generated by P, for all $k \in [n]$ simultaneously,

$$D(\partial_k \mathbf{A}, \partial_k P) \leq \sqrt{\frac{(C_2+2)\log n}{n}} + \frac{6}{n}.$$

Using $\mathcal{R}_2$ as the partition in Lemma 3.13, we find that this implies that the adjacency matrix $\mathbf{A}$ is such that for any i, j and $i' \in \mathcal{N}_{i \setminus j}$,

$$\begin{aligned}
\frac{1}{n} \|P_i - P_{i'}\|_2^2 &\leq 6\mathsf{c}\Delta_2 + 8\left(\sqrt{\frac{(C_2+2)\log n}{n}} + \frac{6}{n}\right) + \frac{5}{n} \\
&\leq 6\mathsf{c}\alpha_2\sqrt{\frac{\log n}{n}} + 8\sqrt{\frac{(C_2+2)\log n}{n}} + \frac{53}{n} \\
&= \left(6\mathsf{c}\alpha_2 + 8\sqrt{C_2+2}\right)\sqrt{\frac{\log n}{n}} + \frac{53}{n} \\
&\leq \tilde{\alpha}_2\sqrt{\frac{\log n}{n}}
\end{aligned}$$

where $\tilde{\alpha}_2$ is an arbitrary constant greater than $6\mathsf{c}\alpha_2 + 8\sqrt{C_2+2}$, and assuming that n is large enough that $53/n \leq \tilde{\alpha}_2 - 6\mathsf{c}\alpha_2 + 8\sqrt{C_2+2}$.

Now we may invoke Lemma 3.14 using $\mathcal{R}_\infty$ as the refinement of $\mathcal{B}$. Define $\gamma = \max\{\tilde{\alpha}_2, 4\rho\alpha_\infty^3 c^2\}$ and let $\tilde{\epsilon} = \gamma\sqrt{\frac{\log n}{n}}$. Then, from the previous result, for any $i' \in \mathcal{N}_{i \setminus j}$, $\frac{1}{n}\|P_i - P_{i'}\|_2^2 \leq \gamma\sqrt{\frac{\log n}{n}}$. Furthermore,

$$\tilde{\epsilon} = \gamma\sqrt{\frac{\log n}{n}} \geq 4\rho\alpha_\infty^3 c^2\sqrt{\frac{\log n}{n}} = 4\rho c^2\left[\alpha_\infty\left(\frac{\log n}{n}\right)^{1/6}\right]^3 = 4\rho c^2\Delta_\infty^3$$

and so we may use the claim to conclude that, with probability at least $1 - 2n^{-C_2/4}$ over graphs generated from P, for all $i, j \in [n]$ and any $i' \in \mathcal{N}_{i \setminus j}$,

$$\|P_i - P_{i'}\|_\infty^2 \leq \frac{4\gamma}{\rho\Delta_\infty}\sqrt{\frac{\log n}{n}} = \frac{4\gamma}{\rho \cdot \alpha_\infty}\left(\frac{\log n}{n}\right)^{1/3}.$$

Now we may apply Lemma 3.15. Let α_t be an arbitrary constant, and choose

$$t = \alpha_t\left(\frac{\log n}{n}\right)^{1/6}.$$

Then, with probability

$$\left(1-2n^{-C_2/4}\right)\left(1-n^{2-2h\alpha_t\left(\frac{n}{\log n}\right)^{2/3}}\right),$$

it holds that

$$\max_{ij}\left|\hat{\mathbf{P}}_{ij}-P_{ij}\right|<\left(\alpha_t+\sqrt{\frac{4\gamma}{\rho\cdot\alpha_\infty}}\right)\left(\frac{\log n}{n}\right)^{1/6}.$$

The probability over all samples and graphs is therefore

$$\left(1-2n^{-C_2/4}\right)\left(1-n^{2-2h\alpha_t\left(\frac{n}{\log n}\right)^{2/3}}\right)\left(1-2\alpha_2\sqrt{\frac{n}{\log n}}n^{-2\alpha_2^2\rho^2}\right).$$

■

3.9.4 Supplementary claims

The following claims will be used in the proofs of Section 3.9.3, and are gathered here for convenience.

Claim 3.11. *Let $\mathcal{R}_2$ be a block partition. Suppose $\mathcal{R}_1$ is a Δ-refinement of $\mathcal{R}_2$. If a S is a ρ-dense sample of $\mathcal{R}_1$, then it is also a ρ-dense sample of $\mathcal{R}_2$.*

Proof. Suppose S is a ρ-dense sample of $\mathcal{R}_1$. Take any block $R\in\mathcal{R}_2$. Then R is the disjoint union of blocks in $\mathcal{R}_1$:

$$R=R_1\cup\ldots\cup R_t$$

where $R_i\in\mathcal{R}_1$. Each R_i is such that $|R_i|\leq\Delta$. Therefore:

$$\frac{|R\cap S|}{n}=\sum_i\frac{|R_i\cap S|}{n}\geq\sum_i(1-\rho)|R_i|=(1-\rho)\sum_i|R_i|=(1-\rho)|R|.$$

Therefore S is a ρ-dense sample of $\mathcal{R}_2$. ■

Claim 3.12. *Let M be an $n \times n$ matrix with values in $[0,1]$. Then for any distinct $u, u', v \in [n]$,*

$$\|(\partial_v M)_u - (\partial_v M)_{u'}\|_2^2 \geq \|M_u - M_{u'}\|_2^2 - 1$$

Proof.

$$\begin{aligned}
\|(\partial_v M)_u - (\partial_v M)_{u'}\|_2^2 &= \sum_t \left((\partial_v M)_{ut} - (\partial_v M)_{u't}\right)^2 \\
&= \sum_t \left(M_{ut} - M_{u't}\right)^2 - \left(M_{uv} - M_{u'v}\right)^2 \\
&= \|M_u - M_{u'}\|_2^2 - \left(M_{uv} - M_{u'v}\right)^2 \\
&\geq \|M_u - M_{u'}\|_2^2 - 1
\end{aligned}$$

■

Claim 3.13. *Let M be an $n \times n$ symmetric matrix with values in $[0,1]$. Then for any distinct $i, j, k \in [n]$,*

$$\left[M^2\right]_{ij} - 1 \leq \left[(\partial_k M)^2\right]_{ij} \leq \left[M^2\right]_{ij}.$$

Proof. We have

$$\begin{aligned}
\left[(\partial_k M)^2\right]_{ij} &= \sum_{l \neq k} M_{il} M_{lj} \\
&= \sum_l M_{il} M_{lj} - M_{ik} M_{kj} \\
&= \left[M^2\right]_{ij} - M_{ik} M_{kj}
\end{aligned}$$

The product $M_{ik} M_{kj}$ is at most one and at least zero, which proves the claim. ■

Claim 3.14. *Let M be an $n \times n$ symmetric matrix. Then for any distinct $i, i' \in [n]$,*

$$\|M_i - M_{i'}\|_2^2 = \left(M^2\right)_{ii} - 2\left(M^2\right)_{ii'} + \left(M^2\right)_{i'i'}.$$

Proof. For any u, v we have

$$\left(M^2\right)_{uv} = \sum_k M_{uk} M_{vk}.$$

Therefore,

$$\begin{aligned}
\left(M^2\right)_{ii} - 2\left(M^2\right)_{ii'} + \left(M^2\right)_{i'i'} &= \sum_k M_{ik}^2 - 2\sum_k M_{ik} M_{i'k} + \sum_k M_{i'k}^2 \\
&= \sum_k \left(M_{ik} - M_{i'k}\right)^2 \\
&= \|M_i - M_{i'}\|_2^2.
\end{aligned}$$

■

Claim 3.15. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$. Suppose $\mathcal{R}$ is a Δ-refinement of $\mathcal{B}$. Let $S = (x_1, \ldots, x_n)$ be a fixed sample. Fix $\Delta > 0$ and assume that $|\mathcal{R}(x_i) \cap S| \geq 4$ for every $i \in [n]$. Let P be the edge probability matrix induced by S. Let A be an adjacency matrix, and suppose that A is such that $D(\partial_k A, \partial_k P) < \epsilon$ for all $k \in [n]$. Then for all $i \neq j \neq k$ simultaneously,*

$$2d_k(i,j) + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon \geq \frac{1}{n}\|P_i - P_j\|_2^2$$

for d_k computed w.r.t. A.

Proof. We may apply Claim 3.12 to obtain

$$\frac{1}{n}\|P_i - P_j\|_2^2 \leq \frac{1}{n}\|(\partial_k P)_i - (\partial_k P)_j\|_2^2 + \frac{1}{n}$$

which may be expanded using Claim 3.14, yielding:

$$= \left[(\partial_k P)^2 / n\right]_{ii} - 2\left[(\partial_k P)^2 / n\right]_{ij} + \left[(\partial_k P)^2 / n\right]_{jj} + \frac{1}{n}$$
$$\leq \left| \left[(\partial_k P)^2 / n\right]_{ii} - \left[(\partial_k P)^2 / n\right]_{ij} \right| + \left| \left[(\partial_k P)^2 / n\right]_{jj} - \left[(\partial_k P)^2 / n\right]_{ij} \right| + \frac{1}{n}$$

By virtue of the fact that every block $\mathcal{R}(x_i)$ in the refinement contains at least 4 points, we may find an $x_{\tilde{i}} \in \mathcal{R}(x_i) \cap S$ and $\tilde{j} \in \mathcal{R}(x_j) \cap S$ such that $\tilde{i} \neq i, k$ and $\tilde{j} \neq j, k$. It is clear that $\left[(\partial_k P)^2 / n\right]_{ii}$ differs from $\left[(\partial_k P)^2 / n\right]_{i\tilde{i}}$ by at most $\mathsf{c}\Delta$, and similarly for the other terms. Hence

$$\leq \left| \left[(\partial_k P)^2 / n\right]_{i\tilde{i}} - \left[(\partial_k P)^2 / n\right]_{\tilde{i}j} \right| + \left| \left[(\partial_k P)^2 / n\right]_{j\tilde{j}} - \left[(\partial_k P)^2 / n\right]_{i\tilde{j}} \right| + \frac{1}{n} + 4\mathsf{c}\Delta$$

Next we apply the assumption that $D(\partial_k A, \partial_k P) < \epsilon$:

$$\leq \left| \left[(\partial_k A)^2 / n\right]_{i\tilde{i}} - \left[(\partial_k A)^2 / n\right]_{\tilde{i}j} \right| + \left| \left[(\partial_k A)^2 / n\right]_{j\tilde{j}} - \left[(\partial_k A)^2 / n\right]_{i\tilde{j}} \right| + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon$$
$$\leq 2 \max_{l \neq i,j} \left| \left[(\partial_k A)^2 / n\right]_{il} - \left[(\partial_k A)^2 / n\right]_{jl} \right| + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon$$

We recognize this as:

$$= 2d_k(i,j) + \frac{1}{n} + 4\mathsf{c}\Delta + 4\epsilon.$$

■

Claim 3.16. *Let $W \in \mathscr{W}_{\mathcal{B}}^{\mathsf{c}}$. Fix a sample $S = (x_1, \ldots, x_n)$ and let P be the induced edge probability matrix. Suppose that $\mathcal{R}$ is a Δ-refinement of $\mathcal{B}$. Now suppose that nodes x_i and $x_{i'}$ are from the same $\mathcal{R}(x_{i''})$ for some i''. Furthermore, suppose that A is an adjacency*

matrix with the property that $D(\partial_j A, \partial_j P) \leq \epsilon$ *for all* $j \in [n]$. *Then for all* $j \neq i, i'$,

$$d_j(i, i') \leq \mathsf{c}\Delta + 2\epsilon + \frac{2}{n}.$$

Proof. We have

$$d_j(i, i') = \max_{k \neq i, i'} \left| \left[(\partial_j A)^2 / n \right]_{ik} - \left[(\partial_j A)^2 / n \right]_{i'k} \right|$$

Applying the fact that $D(\partial_j A, \partial_j P) \leq \epsilon$:

$$\leq \max_{k \neq i, i'} \left| \left[(\partial_j P)^2 / n \right]_{ik} - \left[(\partial_j P)^2 / n \right]_{i'k} \right| + 2\epsilon$$

Applying Claim 3.13 yields an additional two terms of $1/n$:

$$\leq \max_{k \neq i, i'} \left| \left[P^2/n \right]_{ik} - \left[P^2/n \right]_{i'k} \right| + 2\epsilon + \frac{2}{n}$$

The fact that x_i and $x_{i'}$ are from the same block of the Δ-refinement implies that $|x_i - x_{i'}| \leq \Delta$. Hence, by smoothness of W, we have that $|P_{ik} - P_{i'k}| \leq \mathsf{c}\Delta$ for every k. It is therefore the case that for any k $\left| \left[P^2/n \right]_{ik} - \left[P^2/n \right]_{i'k} \right| \leq \mathsf{c}\Delta$, as is shown in the proof of Lemma 5.2 in Zhang et al. (2015). Therefore:

$$\leq \mathsf{c}\Delta + 2\epsilon + \frac{2}{n}.$$

■

3.10 Experiments

In this section we apply the graph clustering method proposed in Algorithm 1 of Section 3.9 to real and synthetic data. The purpose of these experiments is to help the reader develop an

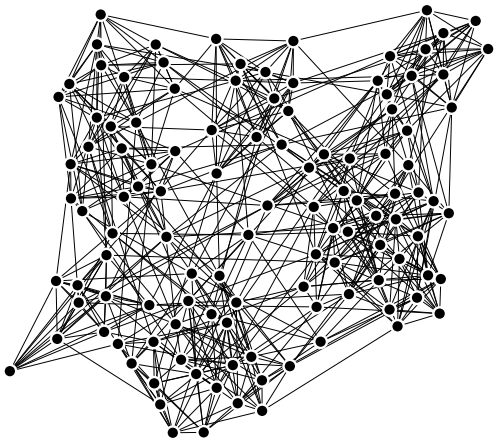


Figure 3.9: Network of college football games played during the 2000 regular season.

intuition for how the clustering method works, and not necessarily to demonstrate superior practical performance. As such, only limited comparisons are made to existing clustering methods.

3.10.1 Football dataset

We first apply Algorithm 1 to the football network from Girvan and Newman (2002). This is a undirected, unweighted graph representing the games played between all NCAA Division I-A American college football teams during the regular season in the year 2000. Each team appears as a node in the graph; an edge exists between two teams if and only if they played one another. The graph, shown in Figure 3.9, includes 115 nodes (teams) and 613 edges (games).

In this year, the teams in Division I-A were divided into eleven football conferences, excepting five "independent" teams which belonged to no conference in particular. The conferences and their associated teams[4] are shown in Table 3.1. In general, an American college football team will play the majority of its games against opponents belonging to its

[4] Note that the dataset from Girvan and Newman (2002) erroneously assigns Texas Christian to C-USA. Texas Christian was in fact in the WAC in the year 2000, and we have made this correction before performing our analysis.

ACC	Big 10	Big 12	Big East	C-USA	Independent
Clemson	Illinois	Baylor	BostonCollege	AlabamaBirmingham	CentralFlorida
Duke	Indiana	Colorado	MiamiFlorida	Army	Connecticut
FloridaState	Iowa	IowaState	Pittsburgh	Cincinnati	Navy
GeorgiaTech	Michigan	Kansas	Rutgers	EastCarolina	NotreDame
Maryland	MichiganState	KansasState	Syracuse	Houston	UtahState
NorthCarolina	Minnesota	Missouri	Temple	Louisville	
NorthCarolinaState	Northwestern	Nebraska	VirginiaTech	Memphis	
Virginia	OhioState	Oklahoma	WestVirginia	SouthernMississippi	
WakeForest	PennState	OklahomaState		Tulane	
	Purdue	Texas			
	Wisconsin	TexasA&M			
		TexasTech			

MAC	MW	Pac 10	SEC	Sunbelt	WAC
Akron	AirForce	Arizona	Alabama	ArkansasState	BoiseState
BallState	BrighamYoung	ArizonaState	Arkansas	Idaho	FresnoState
BowlingGreenState	ColoradoState	California	Auburn	LouisianaLafayette	Hawaii
Buffalo	NevadaLasVegas	Oregon	Florida	LouisianaMonroe	LouisianaTech
CentralMichigan	NewMexico	OregonState	Georgia	MiddleTennesseeState	Nevada
EasternMichigan	SanDiegoState	SouthernCalifornia	Kentucky	NewMexicoState	Rice
Kent	Utah	Stanford	LouisianaState	NorthTexas	SanJoseState
Marshall	Wyoming	UCLA	Mississippi		SouthernMethodist
MiamiOhio		Washington	MississippiState		TexasChristian
NorthernIllinois		WashingtonState	SouthCarolina		TexasElPaso
Ohio			Tennessee		Tulsa
Toledo			Vanderbilt		
WesternMichigan					

Table 3.1: Conference memberships in the football dataset.

own conference – though the team will not usually play every other conference member in the same season. The remaining games on the team's schedule are against out-of-conference opponents. For instance, Ohio State belongs to the Big 10 conference, and in this particular season played conference opponents Iowa, Illinois, Purdue, Michigan, Minnesota, Wisconsin, Michigan State, and Penn State, as well as out-of-conference opponents Miami of Ohio, Arizona, and Fresno State. Because of this connection between conference membership and the scheduling of games, it is reasonable to assume that the graph of football games will exhibit cluster structure. In particular, the clusters of the graph should roughly correspond to the eleven football conferences. As such, we apply the neighborhood smoothing and clustering method to this network and compare the resulting clusters to the eleven football conferences.

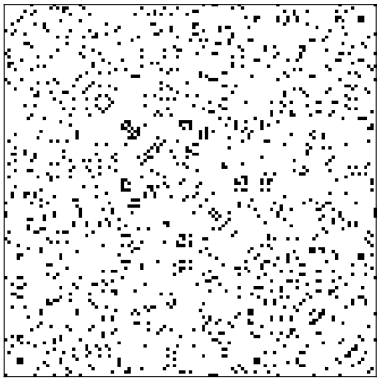
The input to the algorithm is the adjacency matrix of the football graph, shown in

Figure 3.10(a). Rearranging the rows and columns of the adjacency matrix according to conference membership as shown in Figure 3.10(b) reveals the network's cluster structure. Note that the algorithm *does not* have access to this rearranged adjacency or the conference membership of each team; it is shown here only for the convenience of the reader. Smoothing was performed with the neighborhood size parameter $C = 0.09$; the parameter was chosen by hand to produce a good clustering. The output $\hat{P}$ of the network smoothing step is shown in Figure 3.10(c); this matrix after rearranging by conference membership is shown in Figure 3.10(d).

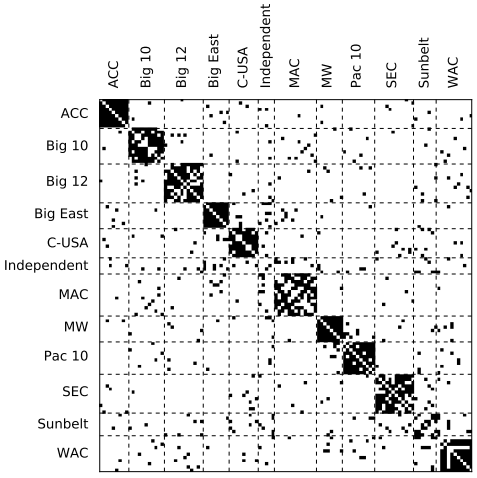
The effect of neighborhood smoothing is to propagate trends in scheduling to all teams within a conference. For instance, consider the ACC and Big East conferences. As Figure 3.10(b) shows, in this season there were five games played between these conferences. Most ACC teams played at least one Big East opponent, but some ACC teams played no Big East opponent. After applying neighborhood smoothing, however, the estimated probability that any ACC team should play any Big East team is uniformly nonzero, as shown in Figure 3.10. That is, even if an ACC team played no Big East opponent, the algorithm smooths the estimate of the probability of such a game to be consistent with the other teams in the conference.

In the clustering step, single-linkage clustering is applied to $\hat{P}$, interpreting it as a similarity matrix. The resulting dendrogram is shown in Figure 3.11. Nodes joining at higher levels of the tree are more similar. If all of the leaf nodes in a subtree belong to the same conference, every edge in the subtree is marked with the same color. Different colors are used to distinguish such subtrees, but the particular color used is not meaningful. The conference labels in the figure are used to show where the majority of that conference's teams are in the clustering. Not marked is the Sun Belt conference, the majority of whose teams are placed between the Big East and Big 12, and the independent teams which belong to no conference in particular.

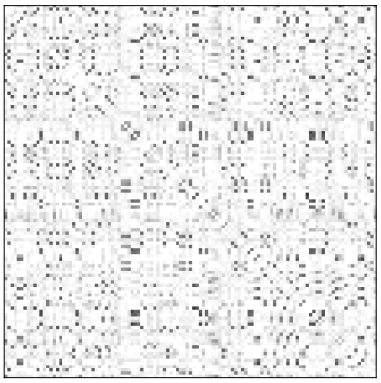
In general the clustering recovers the conferences with high accuracy. In addition, because the clustering is a tree and not a flat partitioning of the teams, more structure is evident. For instance, the clusters corresponding to the MW (Mountain West) conference and the Pac 10 are joined at a high level. This is because the Mountain West and Pac 10 are comprised of teams which are from roughly the same geographical area – the western United States.



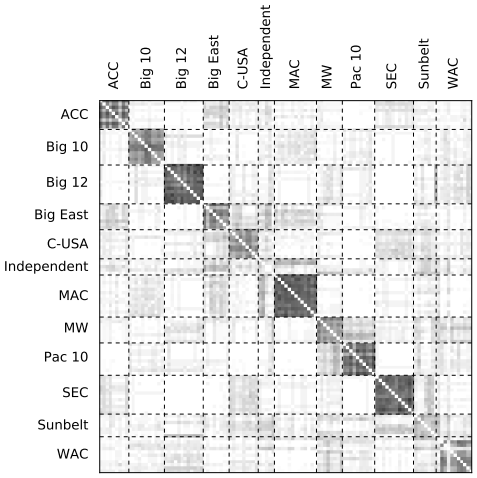
(a) The input adjacency matrix.



(b) The input adjacency matrix, rearranged according to conference membership.



(c) The result of neighborhood smoothing.



(d) The result of neighborhood smoothing, rearranged according to conference membership.

Figure 3.10: The neighborhood smoothing step as applied to the football network.

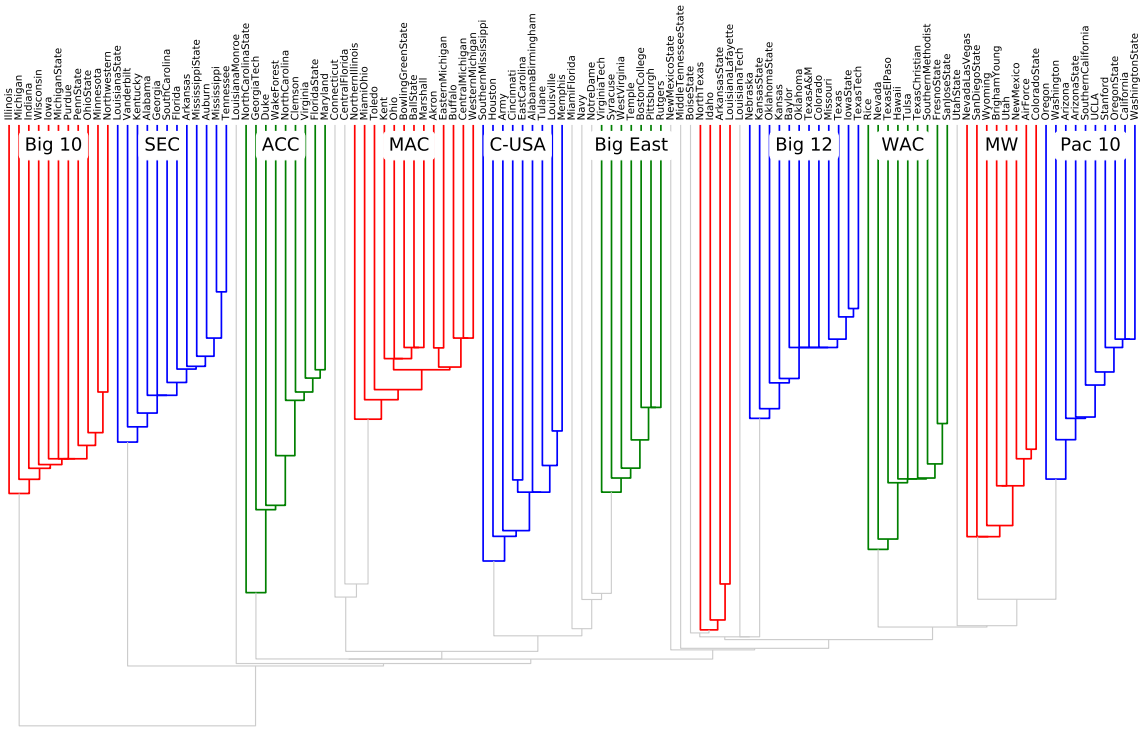


Figure 3.11: The clustering of the football network produced by Algorithm 1.

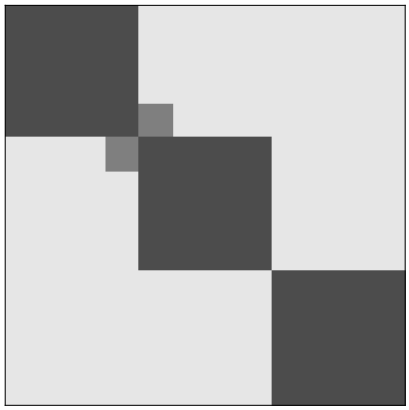
3.10.2 Synthetic network sampled from a graphon

In this experiment we apply Algorithm 1 to a network sampled from the graphon shown in Figure 3.12a. This graphon was chosen to demonstrate a non-trivial case where a simple clustering method may yield the incorrect result. The graphon consists of three large blocks along the diagonal which take value 0.7. The first two of these blocks are joined by a small region whose value is 0.5. As such, the cluster tree of this graphon is as shown in Figure 3.12b.

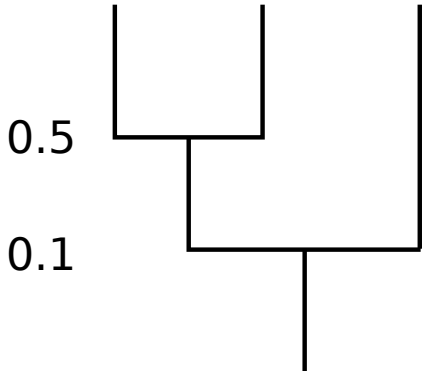
The adjacency matrix of a graph sampled from this graphon is shown in Figure 3.12c. The matrix in the figure has been rearranged in order to show the cluster structure of the graph; The matrix given as input to the smoothing algorithm is a permutation of this matrix. Smoothing was applied with a neighborhood size parameter of $C = 0.1$. The result is shown in Figure 3.12d.

In the cluster step, single linkage is applied to the smoothed estimate of edge probabilites. The resulting dendrogram is shown in Figure 3.13a. Three major clusters are evident in the tree, two of which are joined at a noticeably higher level. As we would expect from a consistent clustering method, the dendrogram resembles the ground-truth cluster tree shown in Figure 3.12b.

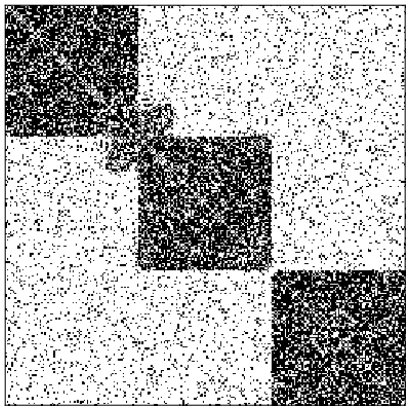
On the other hand, one simple approach to network clustering fails. In this approach, we use the pairwise distance between columns of the adjacency matrix as input to single-linkage clustering. That is, for every $i, j \in \{1, \ldots, n\}$, we use the matrix D whose i, j entry is $\|A_i - A_j\|$, where A_i and A_j are the ith and jth columns of A, respectively, and $\|\cdot\|$ is a suitable norm – here, we use the 2-norm. Such a simple approach can often work in practice; for example, this method works well on the football network in the previous section. However, as the results shown in Figure 3.13b demonstrate, it does not work as well for recovering the graphon cluster tree. Though the method appears to recover three clusters, it does not join two of them at a significantly higher level. Therefore the resulting tree does not resemble the ideal tree. In fact, is easily seen that this method is not consistent



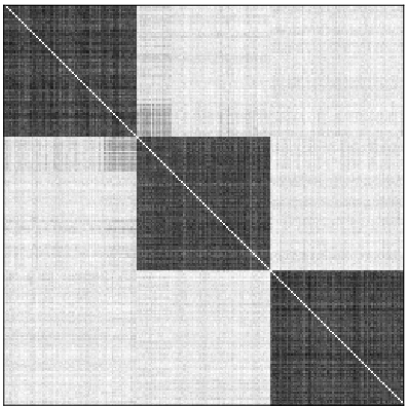
(a) The graphon used in the synthetic experiment. The graphon takes on three values: The darkest region has a height of 0.7; the small, medium-dark blocks are of height 0.5; the remaining light area has value 0.1.



(b) The cluster tree of this graphon. The two leftmost blocks join at a height of 0.5. These join with the remaining block at 0.1.

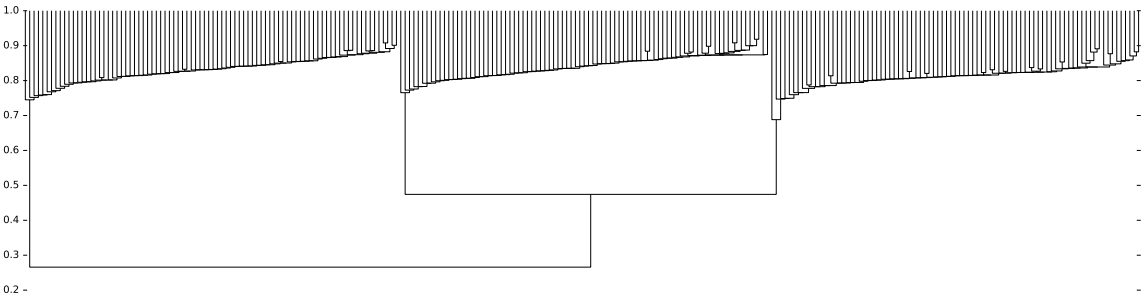


(c) An adjacency matrix sampled from the graphon, rearranged for the presentation (the algorithm receives a random permutation of this matrix.

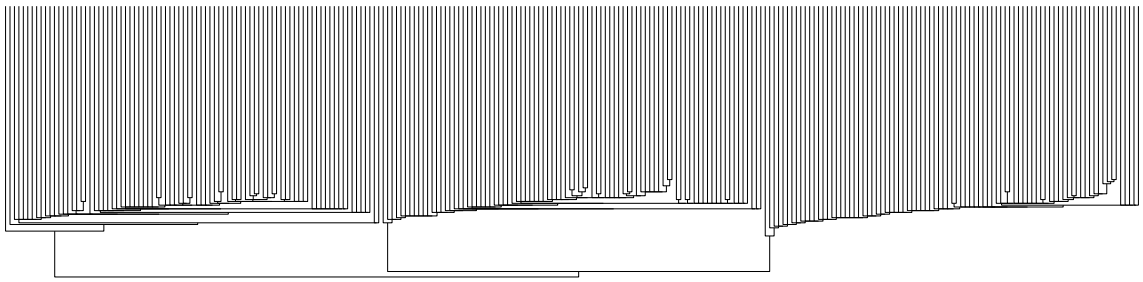


(d) The smoothed estimate of edge probabilities computed from the adjacency matrix at left.

Figure 3.12: The neighborhood smoothing step as applied to a synthetic network.



(a) The result of applying Algorithm 1 to the synthetic network generated from a graphon.



(b) The result of a simple, inconsistent clustering algorithm which applies single-linkage to the pairwise distances between the columns of the adjacency matrix.

Figure 3.13: Neighborhood smoothing compared to naïve single-linkage clustering.

in the sense described earlier.

Chapter 4

Conclusion

This dissertation has studied clustering under a statistical lens by assuming that the data come from an underlying generative model. In both the density and graphon models, we have identified the ideal hierarchical cluster structure of the distribution, defined a rigorous notion of convergence in merge distortion, and analyzed algorithms which converge. Our results are stronger than those which existed previously, and in the graphon setting provide a theory of clustering where little existed.

But the statistical approach to clustering is not the only one. As discussed in Chapter 1, there are numerous ways to formalize the goal of clustering, including framing it as an optimization problem or axiomatizing the behavior of algorithms. Which approach is best depends on the particular application, and arguably the most popular clustering method in use – k-means – makes no statistical assumptions about the source of the data.

Nevertheless, we may still ask statistical questions of clustering algorithms which are framed under a different paradigm. For example, suppose we apply an optimization-based clustering algorithm to points sampled from a density. What aspects of the density cluster tree does the optimal clustering recover? In general, such questions are difficult to answer except in rather simple settings, but some results are known, such that that of Chaudhuri et al. (2009) for k-means. Moreover, common cost functions for clustering are often NP-Hard to optimize, and we sometimes do not even have satisfactory convergence guarantees for approximation algorithms. Studying such algorithms under a statistical lens opens up the possibility that the optimization problem (or its analysis) becomes easier when the data

is generated from a distribution satisfying certain regularity conditions. As it stands, there is much work to be done in order to answer such fundamental questions.

At a higher level, framing clustering as, for example, an optimization problem is necessary because clustering is an unsupervised learning task, and so the correct clustering is *ill defined*. As a result, we must inject assumptions into clustering algorithms such that each has its own internal idea of what the correct clustering should be. It is then up to the data analyst to choose the algorithm whose assumptions match their own goals in clustering. A major reason for developing the sort of correctness results which appear in this dissertation is to provide the analyst with a better understanding of the capabilities of clustering algorithms which make statistical assumptions about the source of the data.

In a sense, the selection of a clustering algorithm and its parameters should be viewed as prescribing a particular a ground truth clustering without assigning a label to every data point. Unfortunately, this process is often rather opaque. Suppose, for instance, that a data analyst has decided that optimizing the k-means objective is the correct approach to clustering in their application. There still remains the problem of choosing the number of clusters k. In order to do so, the analyst will typically run the algorithm for several choices of the parameter and choose the output which looks reasonable.

Implicit in this procedure is the fact that the analyst has an internal idea of what a reasonable clustering looks like. Ideally, such a clustering is eventually produced by interacting with the algorithm through a parameter search. On the other hand, we may design clustering algorithms which explicitly incorporate interaction with the user in order to produce a reasonable clustering. For example, the algorithm may ask the user a limited number of questions, such as whether two objects should be clustered together or separated. In addition, human interaction can be much richer than the simple labeling of points; for instance, the analyst may provide a reason for why two objects should belong to the same cluster. Such *interactive clustering* is of growing interest in the literature; see, for example (Awasthi et al., 2013; Awasthi and Zadeh, 2010; Balcan and Blum, 2008). Nevertheless,

much work remains to be done in this direction, and many questions remain open.

The role of clustering will only grow as datasets increase in number and size; It is therefore important that our understanding of it grows correspondingly. This dissertation has provided correctness results for clustering algorithms, but has perhaps more importantly introduced new tools for discussing the convergence of clustering methods. The notion of *clustering consistently* developed in this work has the potential to be used in theories to come, and will hopefully play a role in improving our knowledge of this important class of algorithms.

References

Abbe, E., Bandeira, A. S., and Hall, G. (2015). Exact recovery in the stochastic block model. *IEEE Trans. Inf. Theory*, 62(1):471–487.

Ackerman, M., Ben-David, S., and Loker, D. (2010). Characterization of linkage-based clustering. *Proceedings of the Conference on Learning Theory*, page 270–281.

Airoldi, E. M., Costa, T. B., and Chan, S. H. (2013). Stochastic blockmodel approximation of a graphon: Theory and consistent estimation. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 26*, pages 692–700. Curran Associates, Inc.

Ash, R. B. and Doleans-Dade, C. (2000). *Probability and measure theory*. Academic Press.

Awasthi, P., Balcan, M.-F., and Voevodski, K. (2013). Local algorithms for interactive clustering. 1312.6724.

Awasthi, P., Charikar, M., Krishnaswamy, R., and Sinop, A. K. (2015). The hardness of approximation of Euclidean k-means. *arXiv*, 1502.03316.

Awasthi, P. and Zadeh, R. B. (2010). Supervised clustering. In Lafferty, J. D., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. S., and Culotta, A., editors, *Advances in Neural Information Processing Systems 23*, pages 91–99. Curran Associates, Inc.

Balakrishnan, S., Narayanan, S., Rinaldo, A., Singh, A., and Wasserman, L. (2013). Cluster Trees on Manifolds. In *Advances in Neural Information Processing Systems*, pages 2679–2687.

Balakrishnan, S., Xu, M., Krishnamurthy, A., and Singh, A. (2011). Noise thresholds for spectral clustering. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 24*, pages 954–962. Curran Associates, Inc.

Balcan, M.-F. and Blum, A. (2008). Clustering with interactive feedback. In Freund, Y., Györfi, L., Turán, G., and Zeugmann, T., editors, *Algorithmic Learning Theory*, volume

5254 of *Lecture Notes in Computer Science*, pages 316–328. Springer Berlin Heidelberg, Berlin, Heidelberg.

Ben-David, S. and Ackerman, M. (2009). Measures of clustering quality: A working set of axioms for clustering. *Advances in Neural Information Processing Systems 21*, page 121–128.

Borgs, C., Chayes, J. T., Lovász, L., Sós, V. T., and Vesztergombi, K. (2008). Convergent sequences of dense graphs I: Subgraph frequencies, metric properties and testing. *Adv. Math.*, 219(6):1801–1851.

Burago, D., Burago, Y., and Ivanov, S. (2001). *A course in metric geometry*, volume 33. American Mathematical Society.

Carlsson, G. and Mémoli, F. (2010). Characterization, stability and convergence of hierarchical clustering methods. *The Journal of Machine Learning Research*, 11:1425–1470.

Carr, H., Snoeyink, J., and Axen, U. (2003). Computing contour trees in all dimensions. *Comput. Geom*, 24(2):75–94.

Chan, S. and Airoldi, E. (2014). A consistent histogram estimator for exchangeable graph models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 208–216.

Charikar, M. and Chatziafratis, V. (2017). Approximate hierarchical clustering via sparsest cut and spreading metrics. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 841–854, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

Chaudhuri, K. and Dasgupta, S. (2010). Rates of convergence for the cluster tree. In *Advances in Neural Information Processing Systems*, pages 343–351.

Chaudhuri, K., Dasgupta, S., Kpotufe, S., and von Luxburg, U. (2014). Consistent procedures for cluster tree estimation and pruning. *IEEE Transactions on Information Theory*, 60(12):7900–7912.

Chaudhuri, K., Dasgupta, S., and Vattani, A. (2009). Learning mixtures of gaussians using the k-means algorithm. 0912.0086.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2001). *Introduction to Algorithms*. The MIT Press, third edition.

Dasgupta, S. (2015). A cost function for similarity-based hierarchical clustering. 1510.05043.

Dasgupta, S. and Freund, Y. (2009). Random projection trees for vector quantization. *IEEE Trans. Inf. Theory*, 55(7):3229–3242.

Eubank, S., Guclu, H., Kumar, V. S. A., Marathe, M. V., Srinivasan, A., Toroczkai, Z., and Wang, N. (2004). Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988):180–184.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proc. Natl. Acad. Sci. U. S. A.*, 99(12):7821–7826.

Hartigan, J. A. (1975). *Clustering Algorithms*. John Wiley and Sons, Inc., New York, NY, USA, 99th edition.

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

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice-Hall, Inc.

Janson, S. (2008). Connectedness in graph limits. *arXiv*, 0802.3795.

Kleinberg, J. (2003). An impossibility theorem for clustering. *Advances in Neural Information Processing Systems*, pages 463–470.

Kpotufe, S. and Luxburg, U. V. (2011). Pruning nearest neighbor cluster trees. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 225–232, New York, NY, USA. ACM.

Lance, G. N. and Williams, W. T. (1967). A general theory of classificatory sorting strategies 1. hierarchical systems. *Comput. J.*, 9(4):373–380.

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Trans. Inf. Theory*, 28(2):129–137.

Lovász, L. (2012). *Large networks and graph limits*, volume 60. American Mathematical Soc.

Lovász, L. and Szegedy, B. (2006). Limits of dense graph sequences. *J. Combin. Theory Ser. B*, 96(6):933–957.

Lu, J., Getz, G., Miska, E. A., Alvarez-Saavedra, E., Lamb, J., Peck, D., Sweet-Cordero, A., Ebert, B. L., Mak, R. H., Ferrando, A. A., and et al. (2005). MicroRNA expression profiles classify human cancers. *Nature*, 435(7043):834–838.

McSherry, F. (2001). Spectral partitioning of random graphs. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 529–537.

Rohe, K., Chatterjee, S., and Yu, B. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *Ann. Stat.*, 39(4):1878–1915.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.

Sibson, R. (1973). SLINK: An optimally efficient algorithm for the single-link cluster method. *Comput. J.*, 16(1):30–34.

Sneath, P., Sokal Nature, R. R., and 1962 (1962). Numerical taxonomy. *Springer*.

Steinwart, I. (2011). Adaptive density level set clustering. In *Proceedings of The 24th Conference on Learning Theory*, pages 703–737.

Stuetzle, W. and Nugent, R. (2010). A Generalized Single Linkage Method for Estimating the Cluster Tree of a Density. *Journal of Computational and Graphical Statistics*, 19(2):397–418.

Ungar, L. H. and Foster, D. P. (1998). Clustering methods for collaborative filtering. *AAAI workshop on recommendation systems*.

Vattani, A. (2011). k-means requires exponentially many iterations even in the plane. *Discrete Comput. Geom.*, 45(4):596–616.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Stat. Comput.*, 17(4):395–416.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. *Nature*, 393(6684):440–442.

Wishart, D. (1969). Mode analysis: A generalization of nearest neighbor which reduces chaining effects. *Numerical taxonomy*, 76(282-311):17.

Wolfe, P. J. and Olhede, S. C. (2013). Nonparametric graphon estimation. *arXiv:1309.5936*, 1309.5936.

Zadeh, R. B. and Ben-David, S. (2009). A uniqueness theorem for clustering. *Proceedings of the twenty-fifth Conference on Uncertainty in Artificial Intelligence*, page 639–646.

Zhang, Y., Levina, E., and Zhu, J. (2015). Estimating network edge probabilities by neighborhood smoothing. *arXiv*, 1509.08588.