Graphons, mergeons, and so on!

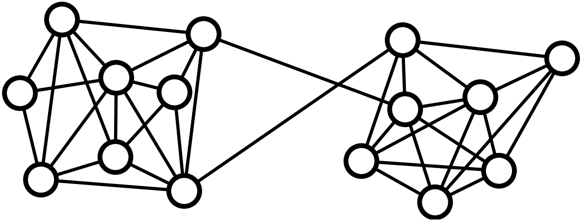
Justin Eldridge

with

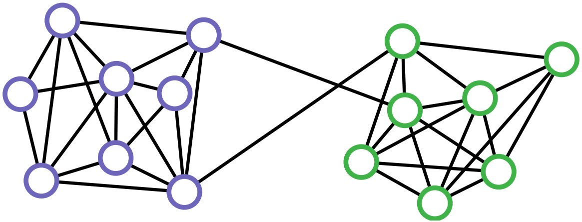
Mikhail Belkin, Yusu Wang

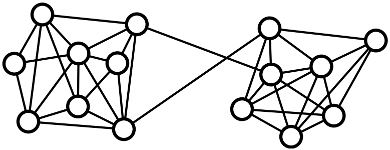
The Ohio State University

Graph clustering



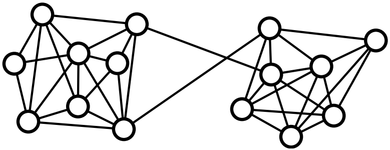
Graph clustering





Question 1

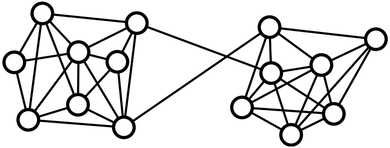
What is the "correct" clustering of the graph?



Question 1

What is the "correct" clustering of the graph?

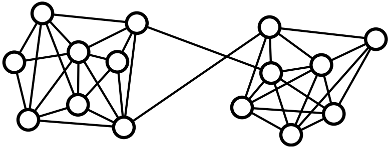
- There is no single answer.



Question 1

What is the "correct" clustering of the graph?

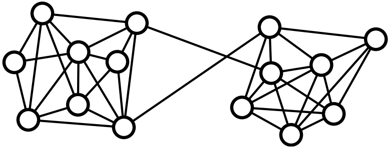
- There is no single answer.
- Right answer depends on nature of the data.



Question 1

What is the "correct" clustering of the graph?

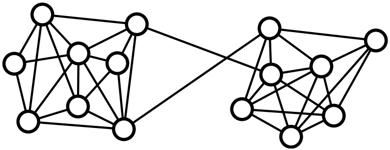
- There is no single answer.
- Right answer depends on nature of the data.
- When graph generated from a random graph model...



Question 1

What is the "correct" clustering of the graph?

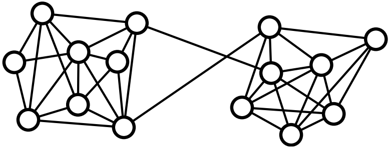
- There is no single answer.
- Right answer depends on nature of the data.
- When graph generated from a random graph model...
- Define the clusters of the model itself.



Question 1

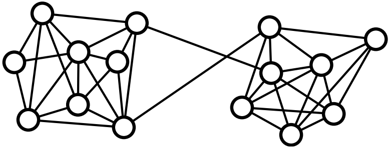
What is the "correct" clustering of the graph?

- There is no single answer.
- Right answer depends on nature of the data.
- When graph generated from a random graph model...
- Define the clusters of the model itself.
- Goal of clustering: recover the clusters of the model from a single graph.



Question 2

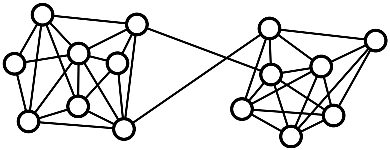
What does it mean to recover the “correct” clustering?



Question 2

What does it mean to recover the “correct” clustering?

- Need a notion of statistical consistency for the clusters of the random graph model.



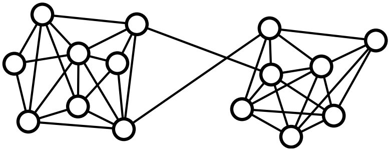
Question 2

What does it mean to recover the "correct" clustering?

- Need a notion of statistical consistency for the clusters of the random graph model.

Question 3

How do we recover the correct clustering?



Question 2

What does it mean to recover the “correct” clustering?

- Need a notion of statistical consistency for the clusters of the random graph model.

Question 3

How do we recover the correct clustering?

- Do correct algorithms exist?

In this talk...

We assume a very general and powerful random graph model called a graphon.

In this talk...

We assume a very general and powerful random graph model called a graphon.

Question 1: What is the “correct” clustering of a graphon?

- We introduce the graphon cluster tree.
- Introduce a useful encoding which we call a mergeon.

In this talk...

We assume a very general and powerful random graph model called a graphon.

Question 1: What is the "correct" clustering of a graphon?

- We introduce the graphon cluster tree.
- Introduce a useful encoding which we call a mergeon.

Question 2: What does it mean to recover the "correct" clustering?

- We develop a notion of statistical consistency for the graphon cluster tree using the mergeon.

In this talk...

We assume a very general and powerful random graph model called a graphon.

Question 1: What is the "correct" clustering of a graphon?

- We introduce the graphon cluster tree.
- Introduce a useful encoding which we call a mergeon.

Question 2: What does it mean to recover the "correct" clustering?

- We develop a notion of statistical consistency for the graphon cluster tree using the mergeon.

Question 3: How do we recover the graphon cluster tree?

- We give sufficient conditions under which a graphon estimator leads to a correct clustering method.
- We identify a practical, correct clustering algorithm.

What is a graphon?

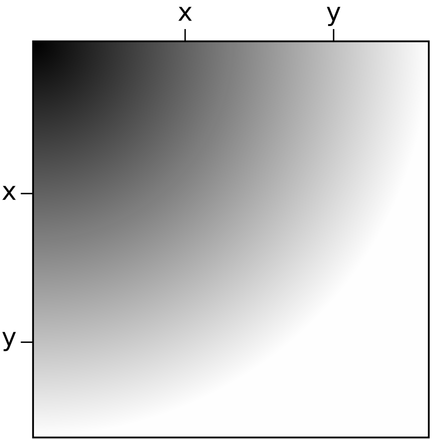
A graphon is a symmetric, measurable function $W : [0,1]^2 \to [0,1]$.

- Intuitively: the weight matrix of a graph on node set $[0,1]$.

What is a graphon?

A graphon is a symmetric, measurable function $W : [0,1]^2 \to [0,1]$.

- Intuitively: the weight matrix of a graph on node set $[0,1]$.

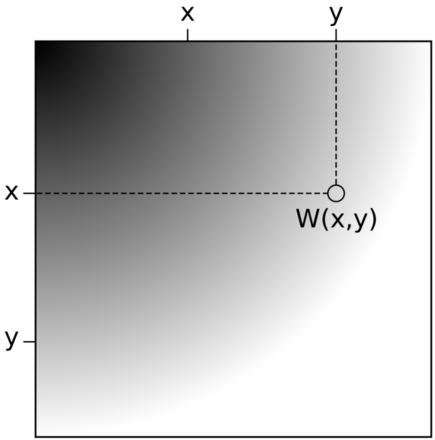


A graphon's "nodes" are points in $[0,1]$.

What is a graphon?

A graphon is a symmetric, measurable function $W : [0,1]^2 \to [0,1]$.

- Intuitively: the weight matrix of a graph on node set $[0,1]$.



The weight of the "edge" (x, y) is $W(x, y)$.

A graphon is a rich object...

Large networks and graph limits

László Lovász

Contents

Preface xi

Part 1. Large graphs: an informal introduction 1

Chapter 1. Very large networks 3
1.1. Huge networks everywhere 3
1.2. What to ask about them? 4
1.3. How to obtain information about them? 5
1.4. How to model them? 8
1.5. How to approximate them? 11
1.6. How to run algorithms on them? 18
1.7. Bounded degree graphs 22

Chapter 2. Large graphs in mathematics and physics 25
2.1. Extremal graph theory 25
2.2. Statistical physics 32

Part 2. The algebra of graph homomorphisms 35

Chapter 3. Notation and terminology 37
3.1. Basic notation 37
3.2. Graph theory 38
3.3. Operations on graphs 39

Chapter 4. Graph parameters and connection matrices 41
4.1. Graph parameters and graph properties 41
4.2. Connection matrices 42
4.3. Finite connection rank 45

Chapter 5. Graph homomorphisms 55
5.1. Existence of homomorphisms 55
5.2. Homomorphism numbers 56
5.3. What hom functions can express 62
5.4. Homomorphism and isomorphism 68
5.5. Independence of homomorphism functions 72
5.6. Characterizing homomorphism numbers 75
5.7. The structure of the homomorphism set 79

Chapter 6. Graph algebras and homomorphism functions 83
6.1. Algebras of quantum graphs 83
6.2. Reflection positivity 88

A graphon is a rich object...

Large networks and graph limits

László Lovász

6.3. Contractors and connectors 94
6.4. Algebras for homomorphism functions 101
6.5. Computing parameters with finite connection rank 106
6.6. The polynomial method 108

Part 3. Limits of dense graph sequences 113

Chapter 7. Kernels and graphons 115
7.1. Kernels, graphons and stepfunctions 115
7.2. Generalizing homomorphisms 116
7.3. Weak isomorphism I 121
7.4. Sums and products 122
7.5. Kernel operators 124

Chapter 8. The cut distance 127
8.1. The cut distance of graphs 127
8.2. Cut norm and cut distance of kernels 131
8.3. Weak and L_1-topologies 138

Chapter 9. Szemerédi partitions 141
9.1. Regularity Lemma for graphs 141
9.2. Regularity Lemma for kernels 144
9.3. Compactness of the graphon space 149
9.4. Fractional and integral overlays 151
9.5. Uniqueness of regularity partitions 154

Chapter 10. Sampling 157
10.1. W-random graphs 157
10.2. Sample concentration 158
10.3. Estimating the distance by sampling 160
10.4. The distance of a sample from the original 164
10.5. Counting Lemma 167
10.6. Inverse Counting Lemma 169
10.7. Weak isomorphism II 170

Chapter 11. Convergence of dense graph sequences 173
11.1. Sampling, homomorphism densities and cut distance 173
11.2. Random graphs as limit objects 174
11.3. The limit graphon 180
11.4. Proving convergence 185
11.5. Many disguises of graph limits 193
11.6. Convergence of spectra 194
11.7. Convergence in norm 196
11.8. First applications 197

Chapter 12. Convergence from the right 201
12.1. Homomorphisms to the right and multicuts 201
12.2. The overlay functional 205
12.3. Right-convergent graphon sequences 207
12.4. Right-convergent graph sequences 211

A graphon is a rich object...

Large networks and graph limits

László Lovász

Chapter 13. On the structure of graphons 217
13.1. The general form of a graphon 217
13.2. Weak isomorphism III 220
13.3. Pure kernels 222
13.4. The topology of a graphon 225
13.5. Symmetries of graphons 234

Chapter 14. The space of graphons 239
14.1. Norms defined by graphs 239
14.2. Other norms on the kernel space 242
14.3. Closures of graph properties 247
14.4. Graphon varieties 250
14.5. Random graphons 256
14.6. Exponential random graph models 259

Chapter 15. Algorithms for large graphs and graphons 263
15.1. Parameter estimation 263
15.2. Distinguishing graph properties 266
15.3. Property testing 268
15.4. Computable structures 276

Chapter 16. Extremal theory of dense graphs 281
16.1. Nonnegativity of quantum graphs and reflection positivity 281
16.2. Variational calculus of graphons 283
16.3. Densities of complete graphs 285
16.4. The classical theory of extremal graphs 293
16.5. Local vs. global optima 294
16.6. Deciding inequalities between subgraph densities 299
16.7. Which graphs are extremal? 307

Chapter 17. Multigraphs and decorated graphs 317
17.1. Compact decorated graphs 318
17.2. Multigraphs with unbounded edge multiplicities 325

Part 4. Limits of bounded degree graphs 327

Chapter 18. Graphings 329
18.1. Borel graphs 329
18.2. Measure preserving graphs 332
18.3. Random rooted graphs 338
18.4. Subgraph densities in graphings 344
18.5. Local equivalence 346
18.6. Graphings and groups 349

Chapter 19. Convergence of bounded degree graphs 351
19.1. Local convergence and limit 351
19.2. Local-global convergence 360

Chapter 20. Right convergence of bounded degree graphs 367
20.1. Random homomorphisms to the right 367
20.2. Convergence from the right 375

A graphon is a rich object...

Large networks and graph limits

László Lovász

Chapter 21. On the structure of graphings 383
21.1. Hyperfiniteness 383
21.2. Homogeneous decomposition 393

Chapter 22. Algorithms for bounded degree graphs 397
22.1. Estimable parameters 397
22.2. Testable properties 402
22.3. Computable structures 405

Part 5. Extensions: a brief survey 413

Chapter 23. Other combinatorial structures 415
23.1. Sparse (but not very sparse) graphs 415
23.2. Edge-coloring models 416
23.3. Hypergraphs 421
23.4. Categories 425
23.5. And more... 429

Appendix A. Appendix 433
A.1. Möbius functions 433
A.2. The Tutte polynomial 434
A.3. Some background in probability and measure theory 436
A.4. Moments and the moment problem 441
A.5. Ultraproduct and ultralimit 444
A.6. Vapnik–Chervonenkis dimension 445
A.7. Nonnegative polynomials 446
A.8. Categories 447

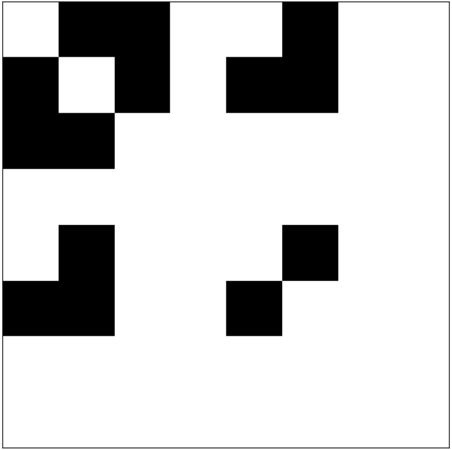
Bibliography 451

Author Index 465

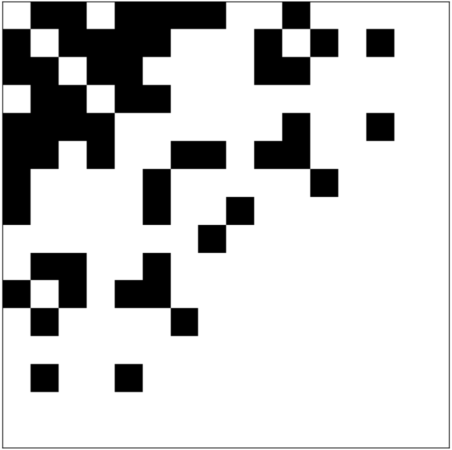
Subject Index 469

Notation Index 473

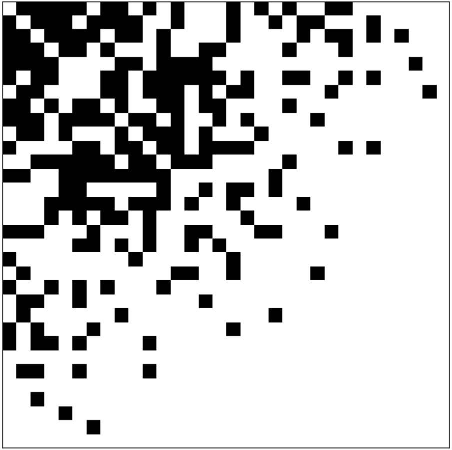
A graphon is a graph limit.



A graphon is a graph limit.

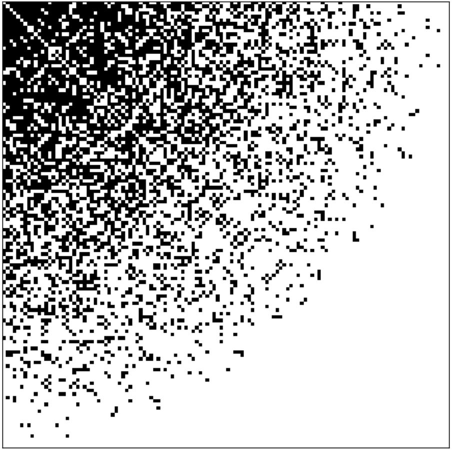


A graphon is a graph limit.

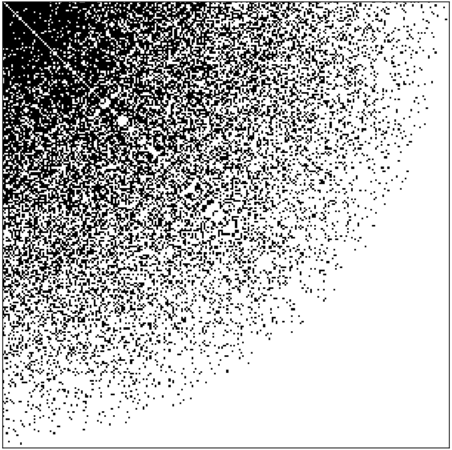


A graphon is a graph limit.

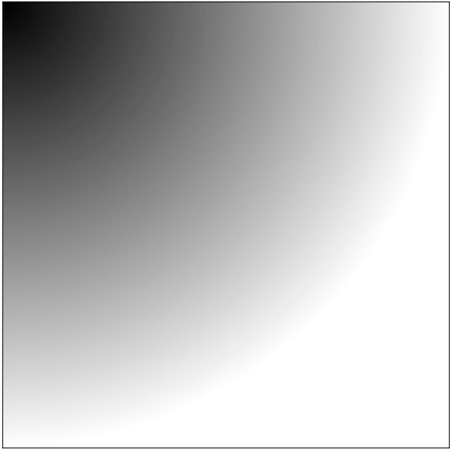
A graphon is a graph limit.



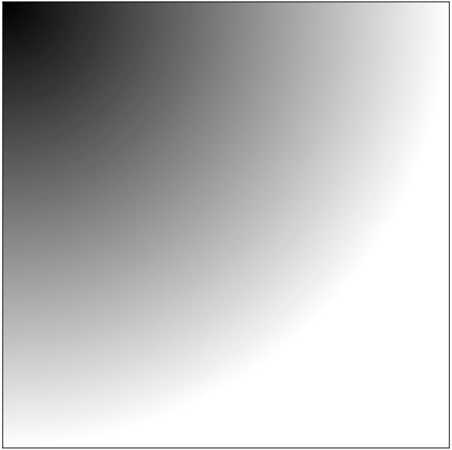
A graphon is a graph limit.



A graphon is a graph limit.

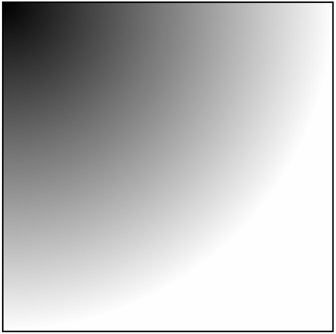


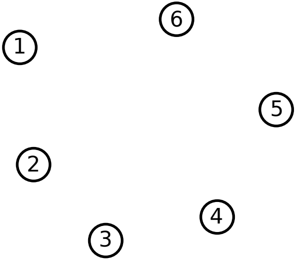
A graphon is a random graph model.



When $W(x, y)$ is interpreted as a probability.

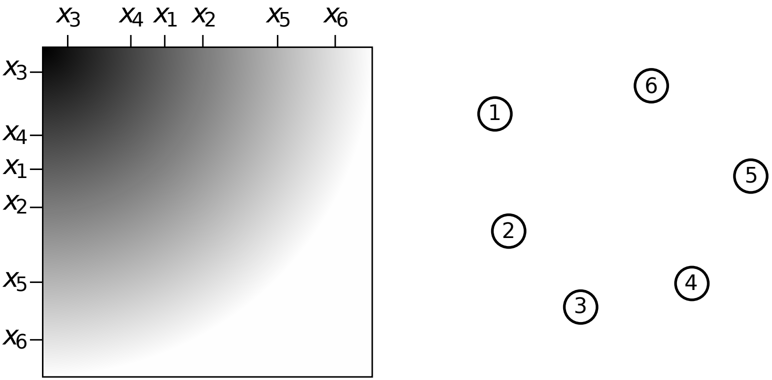
Sampling from a graphon





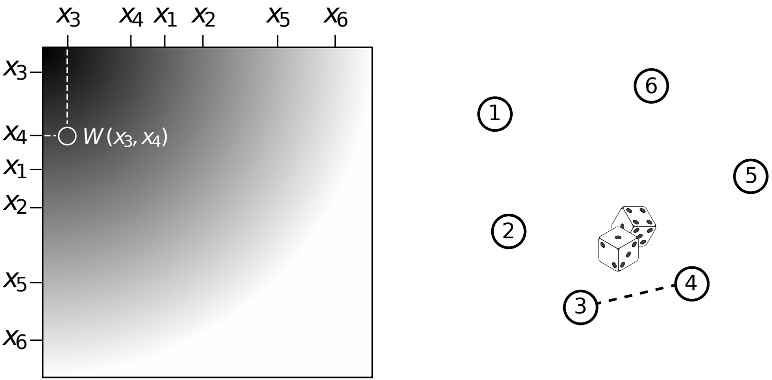
Start with a graph with integer node set [n].

Sampling from a graphon



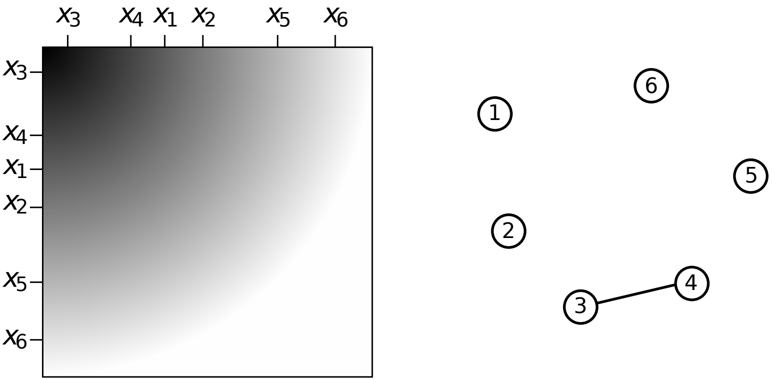
Draw n points $\{x_1, \ldots, x_n\}$ from Unif([0, 1]).

Sampling from a graphon



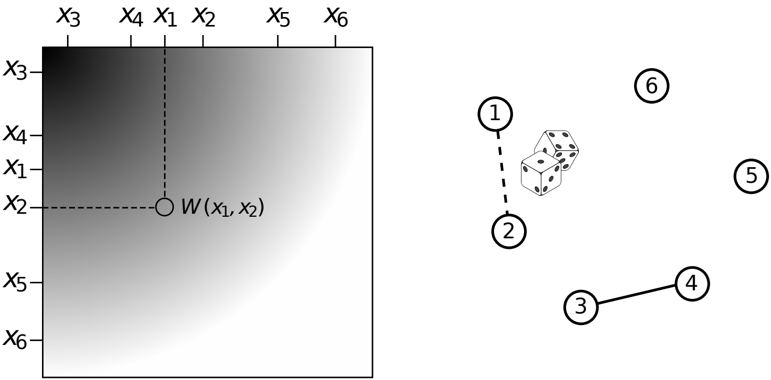
Connect nodes 3 and 4 with probability $W(x_3, x_4)$.

Sampling from a graphon



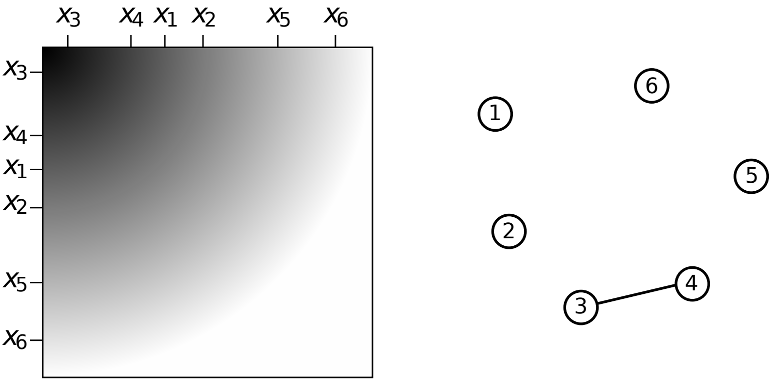
By chance, edge $(3, 4)$ is included.

Sampling from a graphon



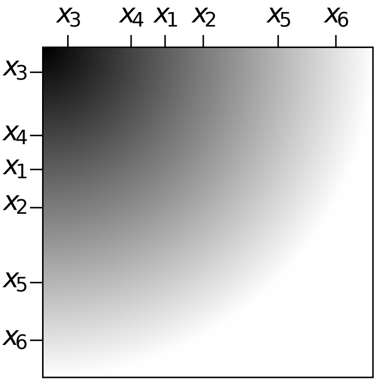
Connect nodes 1 and 2 with probability $W(x_1, x_2)$.

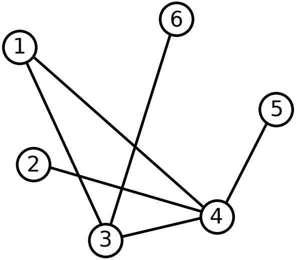
Sampling from a graphon



By chance, edge $(1, 2)$ is omitted.

Sampling from a graphon





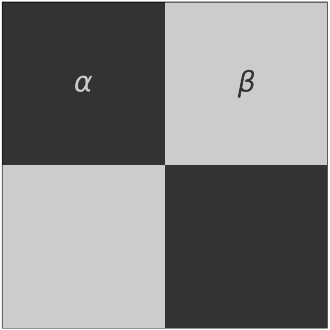
Repeat for all edges, resulting in a randomly-generated graph.

The stochastic blockmodel

Much theory of graph clustering assumes a stochastic blockmodel. Example:

- Randomly assign each of n nodes to one of two communities.
- Add edge between two nodes with probability:
  - α, if in the same community,
  - β, otherwise.

This is a special case of a graphon.



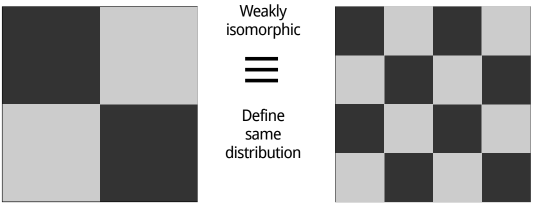
Equivalent graphons

- Any graphon W defines a distribution on graphs.
- Not uniquely! Many graphons define the same distribution.

Equivalent graphons

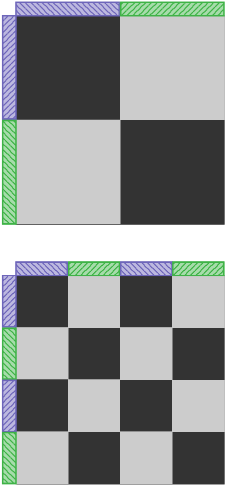
- Any graphon W defines a distribution on graphs.
- Not uniquely! Many graphons define the same distribution.
- Graphs are isomorphic if they are equivalent up to relabeling.
- Graphons are weakly isomorphic if they are equivalent up to a relabeling $\varphi : [0, 1] \to [0, 1]$.
  - φ must be measure preserving.
- An equivalence class of graphons under weak isomorphism uniquely defines distribution.

W $\qquad$ $W^{\varphi}, \varphi = 2x \mod 1$

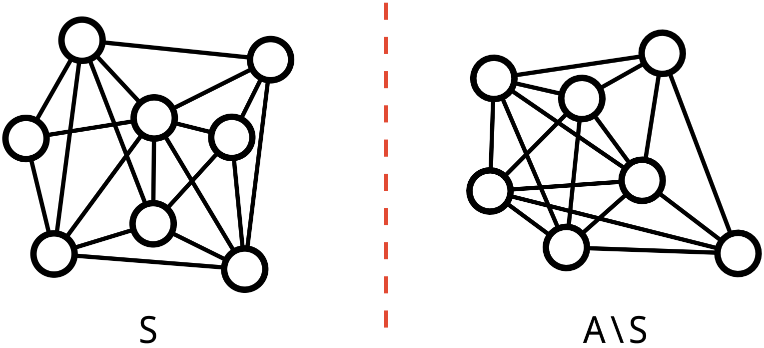


Question 1: What are the clusters of a graphon?

- It is natural to define clusters in terms of connected components.
- Carefully define connectivity for graphons.
- Intuitively: blockmodel graphon should have the two clusters shown.
- Clusters should be:
  - robust to changes to W on a set of zero measure,
  - preserved under relabeling of W.

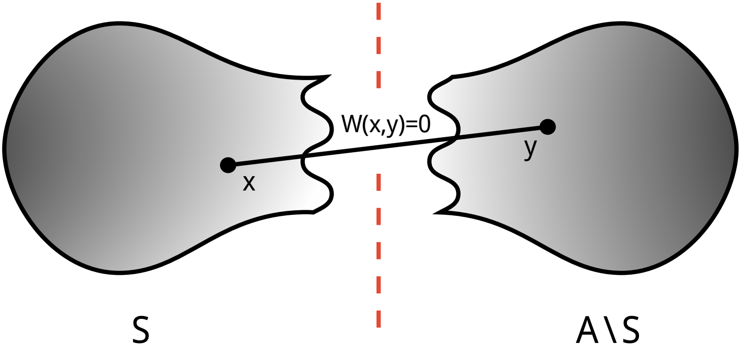


Connectedness in graphs



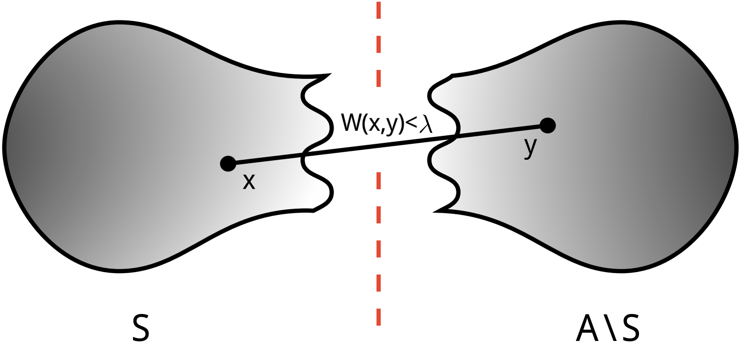
A is disconnected if it can be partitioned into S and A \ S with no crossing edge. Otherwise it is connected.

Connectedness in graphons

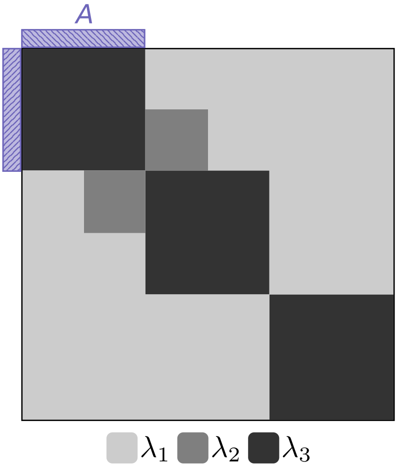


(Janson, 2008): A is disconnected if it can be partitioned into S and A \ S such that the weight of almost every crossing edge is zero. Otherwise it is connected.

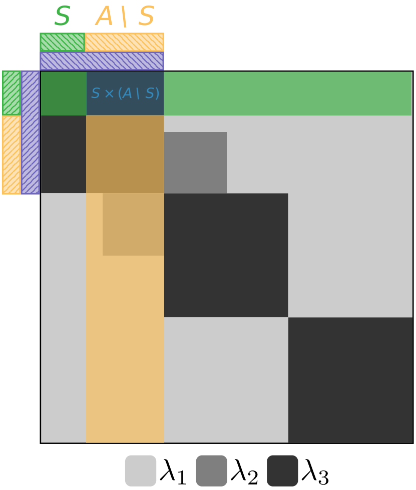
Connectedness in graphons



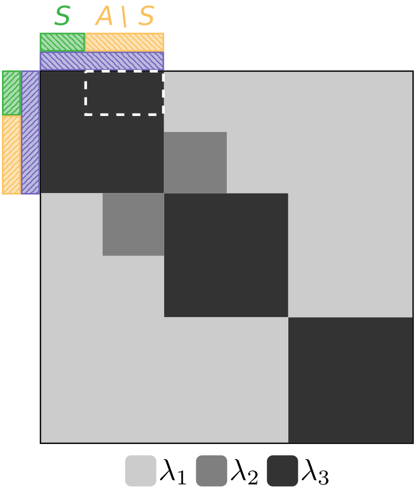
A is disconnected at level λ if it can be partitioned into S and A \ S such that the weight of almost every crossing edge is less than λ. Otherwise it is connected at level λ.



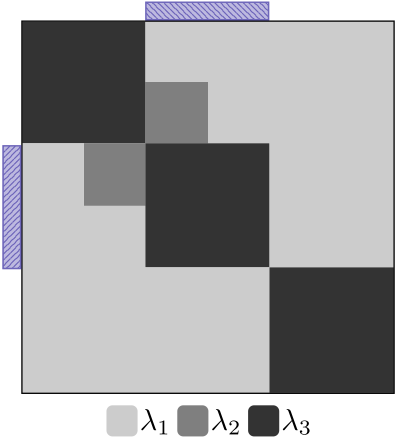
Connected at level λ_3.



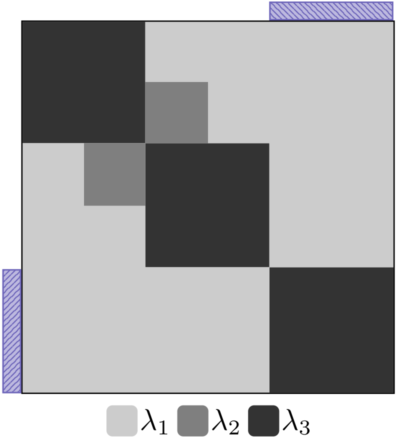
Connected at level λ_3.



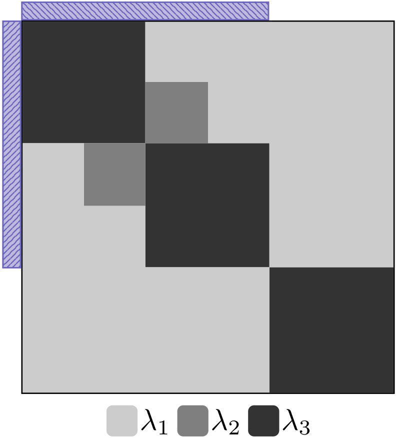
Connected at level λ_3.



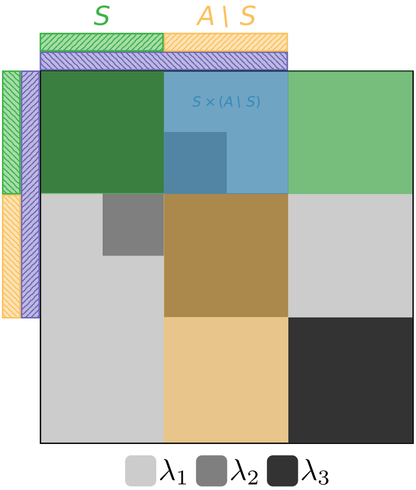
Connected at level λ_3.



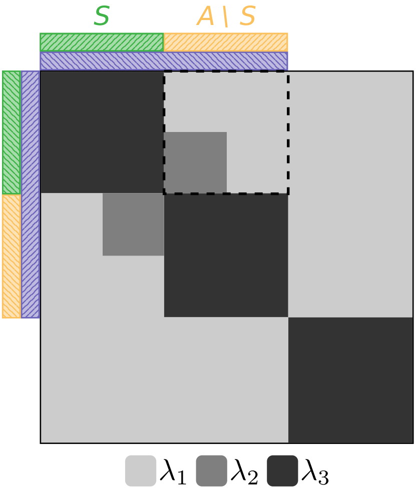
Connected at level λ_3.



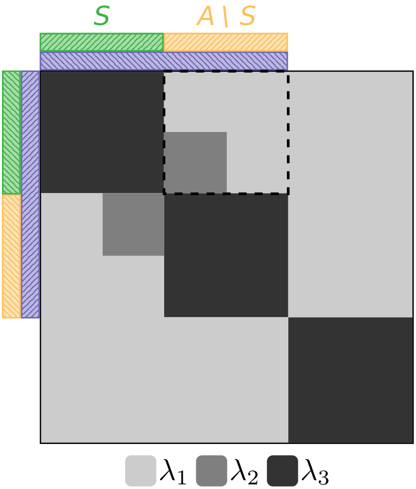
Disconnected at level λ_3.



Disconnected at level λ_3.



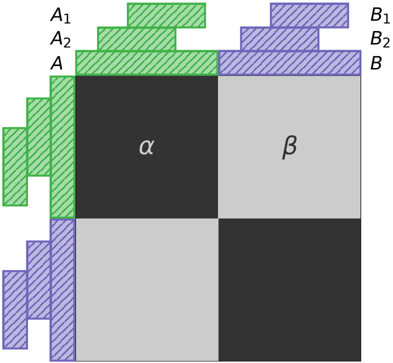
Disconnected at level λ_3.



Connected at level λ_2.

The clusters of a graphon

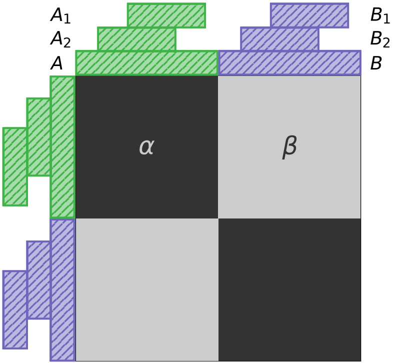
Naturally define clusters in terms of maximally connected sets:



The clusters of a graphon

Naturally define clusters in terms of maximally connected sets:

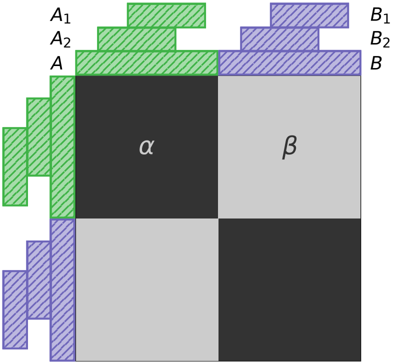
1. A set of nodes should be in a cluster at level λ if it is connected at level λ.



The clusters of a graphon

Naturally define clusters in terms of maximally connected sets:

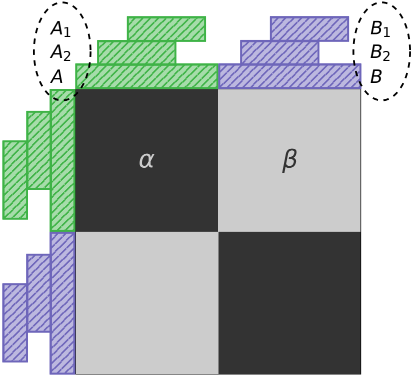
1. A set of nodes should be in a cluster at level λ if ~~it is connected at level λ~~ it is connected at all $\lambda' < \lambda$.



The clusters of a graphon

Naturally define clusters in terms of maximally connected sets:

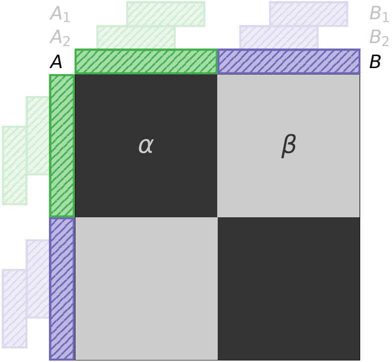
1. A set of nodes should be in a cluster at level λ if ~~it is connected at level λ~~ it is connected at all $\lambda' < \lambda$.
2. Group the sets which should be in the same cluster.



The clusters of a graphon

Naturally define clusters in terms of maximally connected sets:

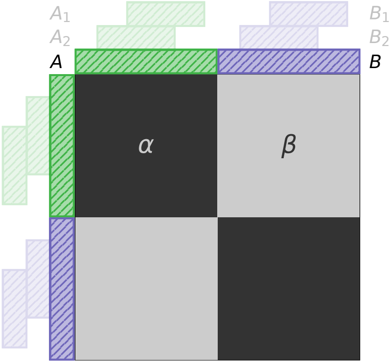
1. A set of nodes should be in a cluster at level λ if ~~it is connected at level λ~~ it is connected at all $\lambda' < \lambda$.
2. Group the sets which should be in the same cluster.
3. Define clusters to be the "largest" elements of each group.



The clusters of a graphon

Naturally define clusters in terms of maximally connected sets:

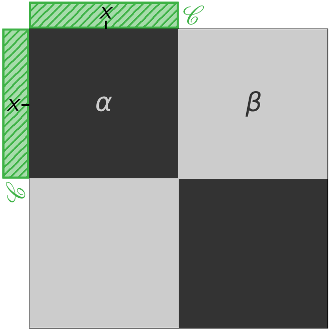
1. A set of nodes should be in a cluster at level λ if ~~it is connected at level λ~~ it is connected at all $\lambda' < \lambda$.
2. Group the sets which should be in the same cluster.
3. Define clusters to be the "largest" elements of each group.



We write $\mathbb{C}_W(\lambda)$ to denote the set of clusters of W at level λ.

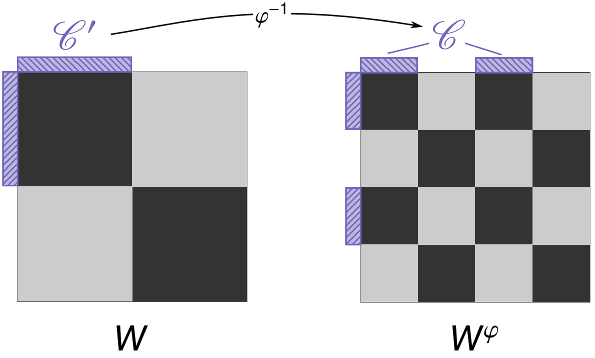
Properties of clusters

- Connectivity of a set is not changed by adding/removing sets of zero measure.
- Careful! A cluster $\mathscr{C} \in \mathbb{C}_W(\lambda)$ is not a subset of $[0,1]$!
  - It is an equivalence class of subsets equal up to null sets.
  - A single point $x \in [0,1]$ belongs to no cluster in particular.



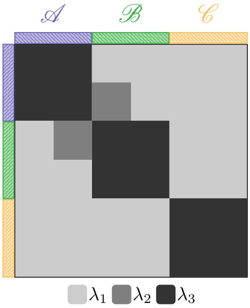
Properties of clusters

- Claim: Clusters are preserved under relabelings.
- I.e., the clusters of a graphon W and the clusters of its relabeling W^{φ} are in bijection.
  - Surprisingly non-trivial to show due to graphon subtleties.
- Example: $\varphi(x) = 2x \mod 1$



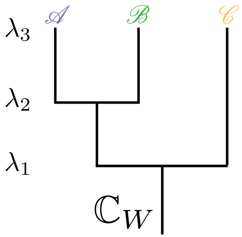
The graphon cluster tree

- The set of clusters $\mathbb{C}_W$ from all levels has hierarchical structure.
  - I.e., if C and C′ are clusters, then either $\mu(C \cap C') = 0$, $\mu(C \setminus C') = 0$, or $\mu(C' \setminus C) = 0$.
- We call $\mathbb{C}_W$ the graphon cluster tree of W.
- Claim: If two graphons are equivalent, their cluster trees are isomorphic.



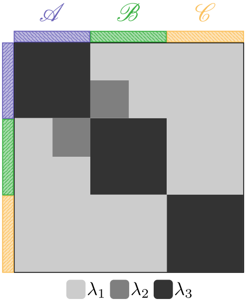
The goal of clustering (graphon setting)

- Given a graph sampled from W...
- recover the cluster tree of W.

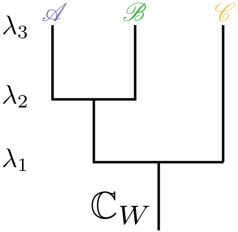


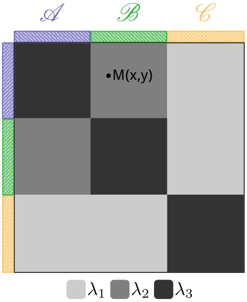
Mergeons

- We may naturally speak of the height at which clusters merge.
- But the merge height of any pair of nodes is undefined.
- Encode particular choice of merge heights with a mergeon.



Graphon W

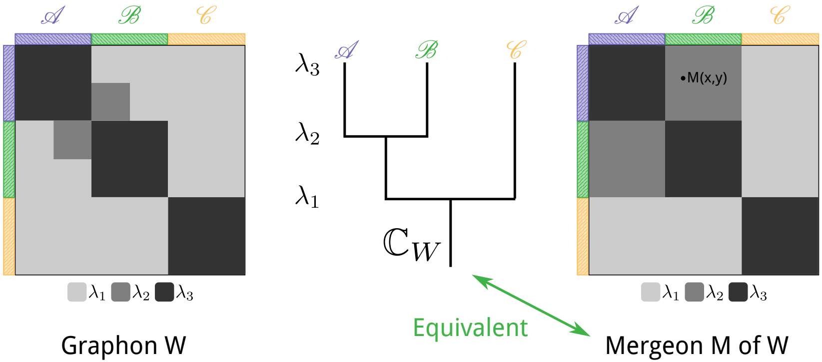


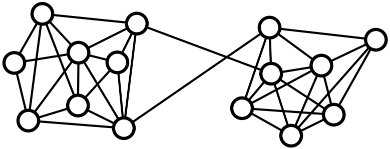


Mergeon M of W

Mergeons

- We may naturally speak of the height at which clusters merge.
- But the merge height of any pair of nodes is undefined.
- Encode particular choice of merge heights with a mergeon.



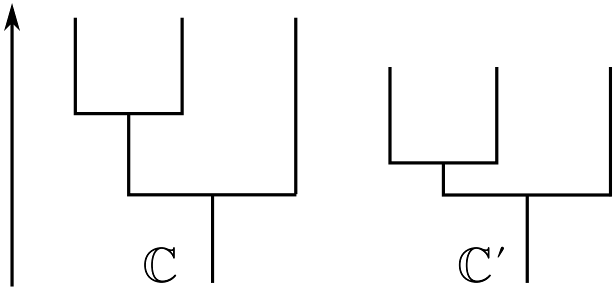


Question 1: What is the "correct" clustering of a graphon?
Answer: The mergeon or, equivalently, the graphon cluster tree.

Question 2: What does it mean to recover the "correct" clustering?

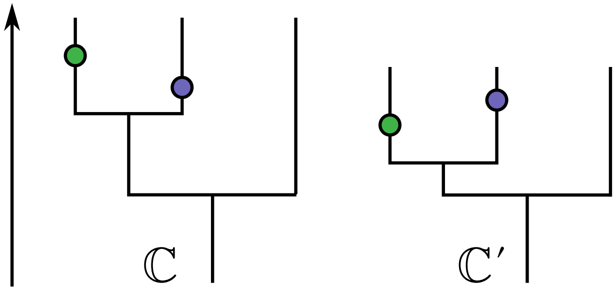
Question 3: How do we recover the correct clustering?

The merge distortion



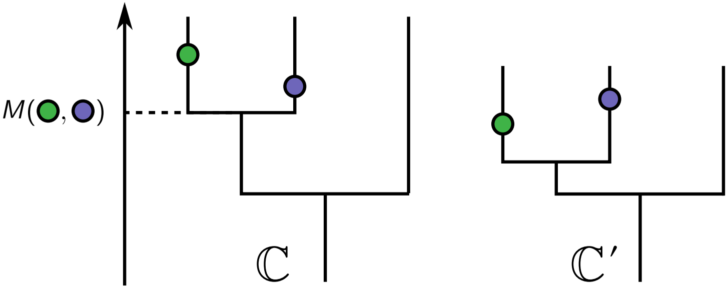
How "close" are $\mathbb{C}$ and $\mathbb{C}'$?

The merge distortion



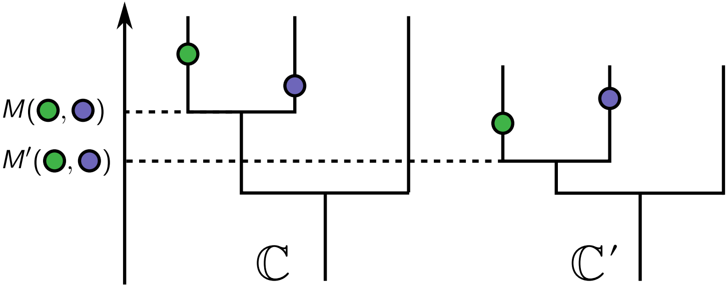
How “close” are $\mathbb{C}$ and $\mathbb{C}'$?
Compare merge heights using mergeons.

The merge distortion



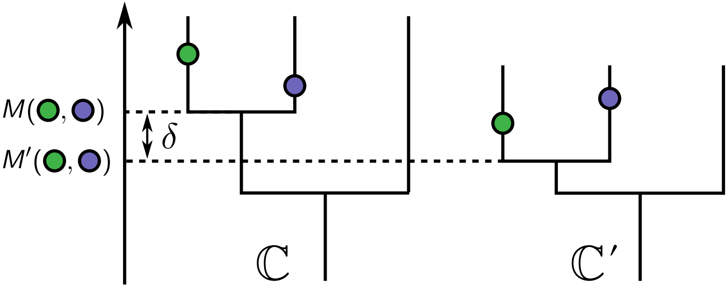
How "close" are $\mathbb{C}$ and $\mathbb{C}'$?
Compare merge heights using mergeons.

The merge distortion



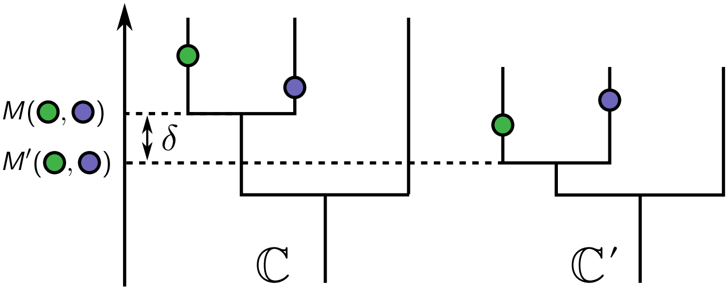
How "close" are $\mathbb{C}$ and $\mathbb{C}'$?
Compare merge heights using mergeons.

The merge distortion



How “close” are $\mathbb{C}$ and $\mathbb{C}'$?
Compare merge heights using mergeons.

The merge distortion



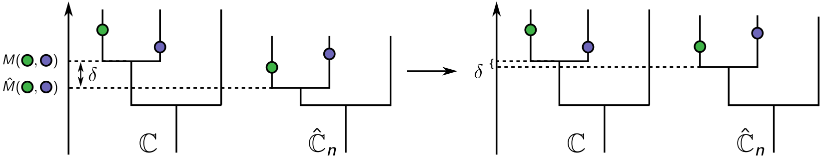
The merge distortion between $\mathbb{C}$ and $\mathbb{C}'$ with respect to a (finite) set S is:

$$d_S(\mathbb{C}, \mathbb{C}') = \max_{s_1 \neq s_2 \in S} \left| M(s_1, s_2) - M'(s_1, s_2) \right| .$$

Convergence in merge distortion

Definition

A sequence $\hat{\mathbb{C}}_n$ converges in merge distortion to $\mathbb{C}$ if $d(\mathbb{C}, \hat{\mathbb{C}}_n) \to 0$ as $n \to \infty$.



Consistent clustering methods

Question 2

What does it mean to recover the "correct" clustering?

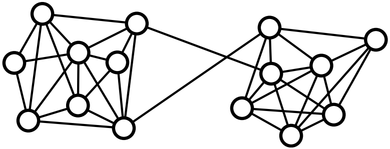
- A clustering method is consistent for the graphon W if its output converges in merge distortion to $\mathbb{C}_W$, w.h.p. as $n \to \infty$.

Consistent clustering methods

Question 2

What does it mean to recover the "correct" clustering?

- A clustering method is consistent for the graphon W if its output converges in merge distortion to $\mathbb{C}_W$, w.h.p. as $n \to \infty$.
- That is:
  - If G_n is a random graph of size n sampled from W,
  - $\hat{\mathbb{C}}_{G_n}$ is the output of the method given G_n as input,
  - then, for any fixed $\epsilon > 0$, $\mathbb{P}(d(\mathbb{C}_W, \hat{\mathbb{C}}_{G_n}) > \epsilon) \to 0$ as $n \to \infty$.
- Consistent methods recover the clusters of the graphon.



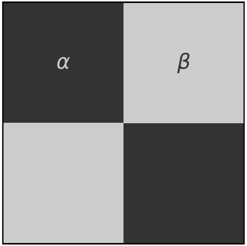
Question 1: What is the "correct" clustering of a graphon?
Answer: The graphon cluster tree or, equivalently, the mergeon.

Question 2: What does it mean to recover the "correct" clustering?
Answer: Convergence in merge distortion to graphon cluster tree.

Question 3: How do we recover the correct clustering?

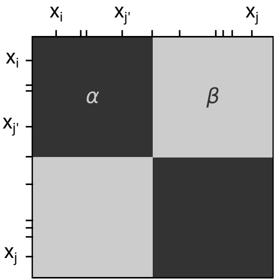
- I.e., do algorithms exist which are consistent in merge distortion?

Estimating edge probabilities

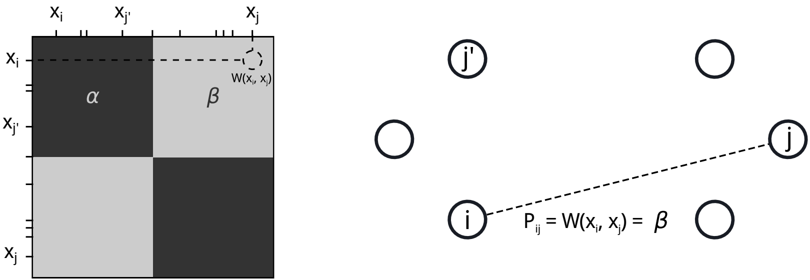


Consider sampling a graph from this graphon.

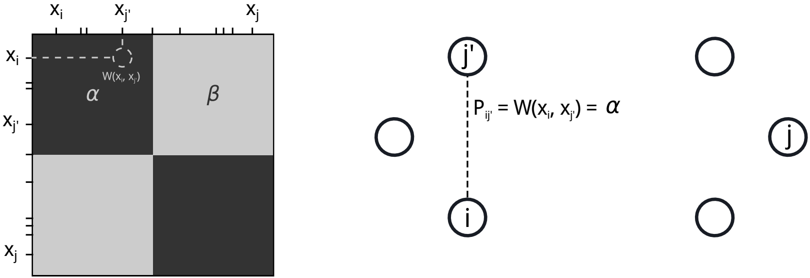
Estimating edge probabilities



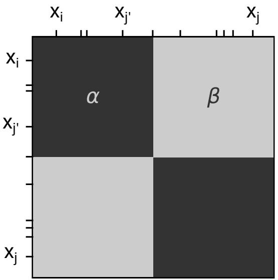
Estimating edge probabilities

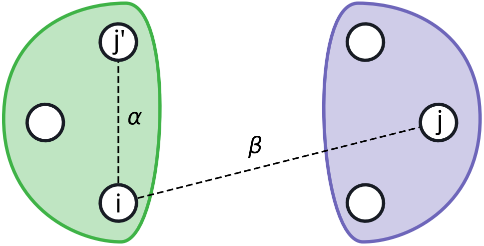


Estimating edge probabilities



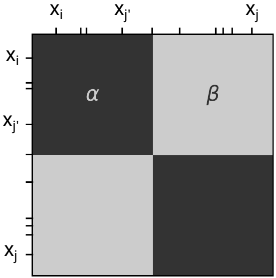
Estimating edge probabilities

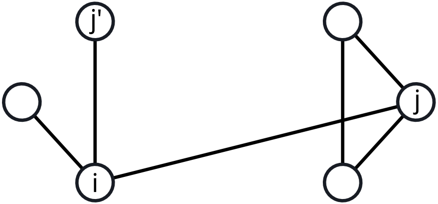




The correct clustering is determined by these edge probabilities.

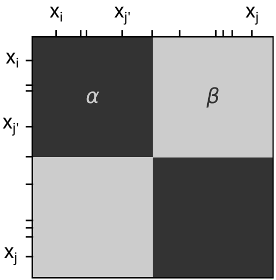
Estimating edge probabilities

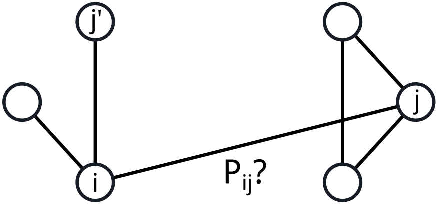




But the edge probabilities are unknown, and the presence of an edge (i, j) tells us little about P_{ij}.

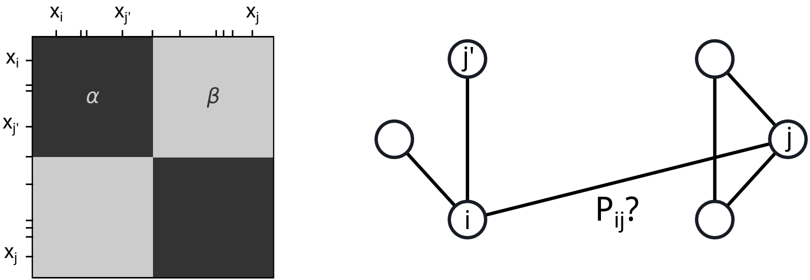
Estimating edge probabilities





Goal: Compute estimate $\hat{P}$ of edge probabilities from single graph.

Estimating edge probabilities



Goal: Compute estimate $\hat{P}$ of edge probabilities from single graph.

Theorem

If $\|P - \hat{P}\|_\infty \to 0$ in probability as $n \to \infty$, then single linkage clustering on $\hat{P}$ is a consistent clustering method.

Neighborhood smoothing

- To cluster consistently, it is sufficient to estimate P in ∞-norm.
- We now search for such an estimator...
- Zhang et al. (2015) propose neighborhood smoothing.

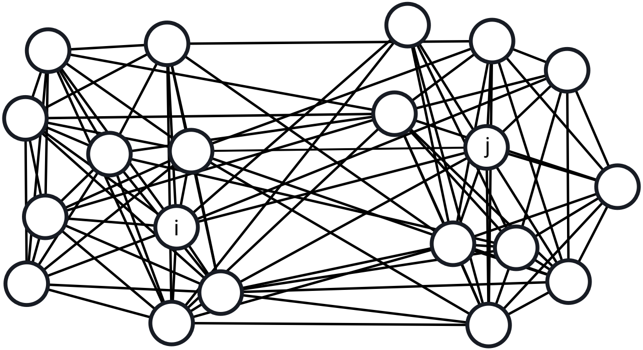
Neighborhood smoothing

- To cluster consistently, it is sufficient to estimate P in ∞-norm.
- We now search for such an estimator...
- Zhang et al. (2015) propose neighborhood smoothing.
- Motivation:
  - If we had many observations of random graph: estimate P_{ij} by counting those which contain (i, j).
  - But we have just one observation.

Neighborhood smoothing

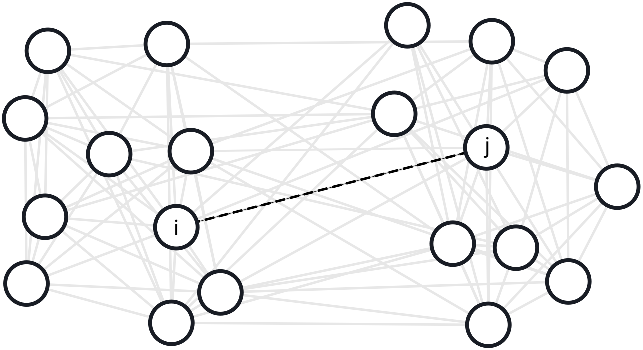
- To cluster consistently, it is sufficient to estimate P in ∞-norm.
- We now search for such an estimator...
- Zhang et al. (2015) propose neighborhood smoothing.
- Motivation:
  - If we had many observations of random graph: estimate P_{ij} by counting those which contain (i, j).
  - But we have just one observation.
- Approach:
  - For node i, build neighborhood N_i of similar nodes.
  - Think of $i' \in N_i$ as another observation of i.
  - To estimate P_{ij}: count number of edges between j and a node in N_i.

Neighborhood smoothing



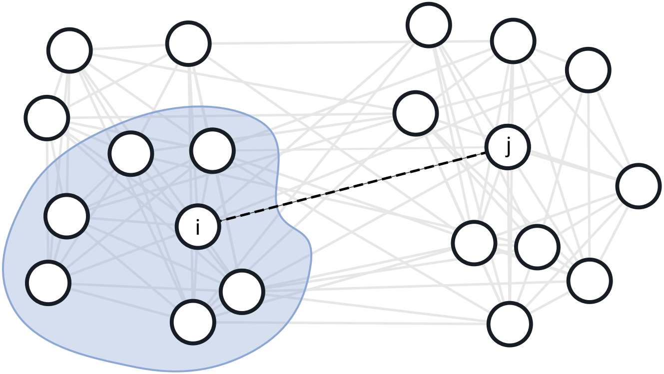
Given this graph...

Neighborhood smoothing



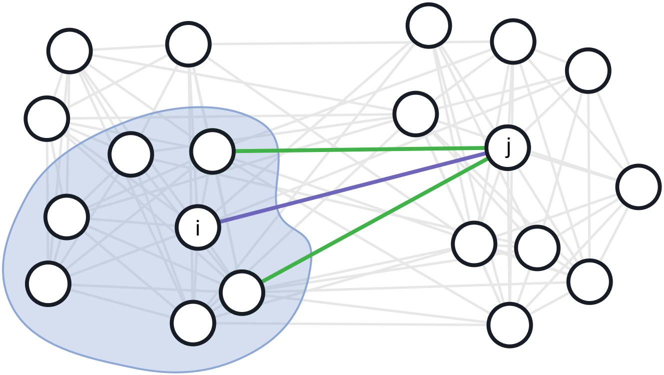
Given this graph... estimate P_{ij}.

Neighborhood smoothing



Build a neighborhood N_i of nodes with similar connectivity to that of i. I.e., close in the distance: $d(i, i') = \max_{k \neq i, i'} \left|(A^2)_{ik} - (A^2)_{i'k}\right|$.

Neighborhood smoothing



- Count number of edges from N_i to node j (excluding i): 2.
- Normalize by size of neighborhood: 6.
- Estimated edge probability: $\hat{P}_{ij} = {}^2/_6 = {}^1/_3$.

Consistency of neighborhood smoothing

- Zhang et al. (2015) prove that neighborhood smoothing is consistent in mean squared error:

$$\frac{1}{n^2}\|P - \hat{P}\|_F^2 = \frac{1}{n^2}\sum_{ij}(P_{ij} - \hat{P}_{ij})^2 \to 0 \quad \text{as} \quad n \to \infty, \text{ w.h.p.}.$$

- But convergence in this norm is too weak. We need convergence in ∞-norm.
- We modify neighborhood smoothing and analyze.

Consistency of neighborhood smoothing

- Zhang et al. (2015) prove that neighborhood smoothing is consistent in mean squared error:

$$\frac{1}{n^2}\|P - \hat{P}\|_F^2 = \frac{1}{n^2}\sum_{ij}(P_{ij} - \hat{P}_{ij})^2 \to 0 \quad \text{as} \quad n \to \infty, \text{ w.h.p..}$$

- But convergence in this norm is too weak. We need convergence in ∞-norm.
- We modify neighborhood smoothing and analyze.

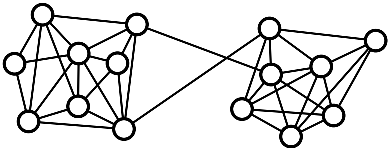
Theorem

The modified neighborhood smoothing estimator for P is consistent in ∞-norm.

Corollary

Performing single linkage on the modified neighborhood smoothing estimate of P is a consistent graphon clustering method.

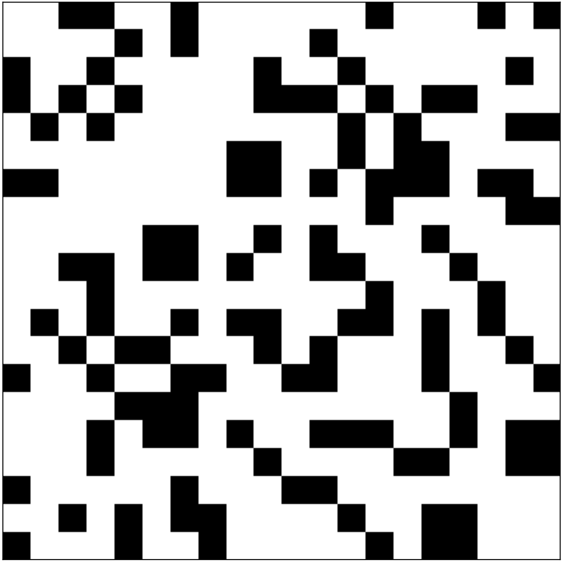
Summary

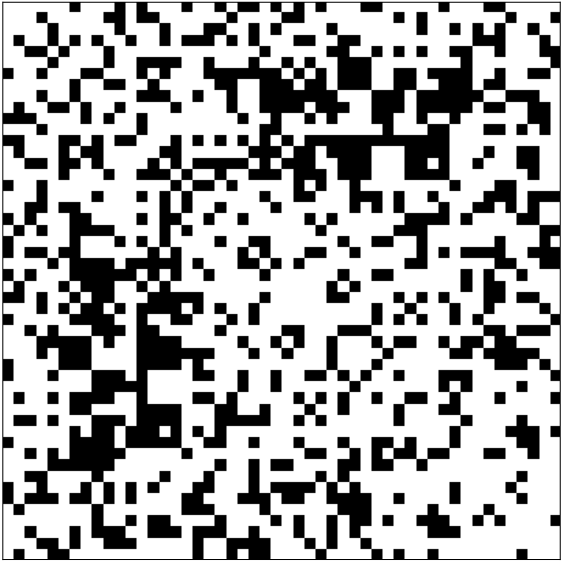


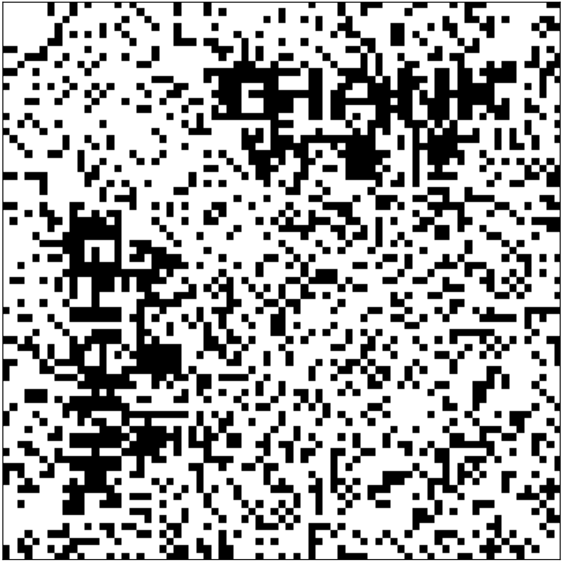
Question 1: What is the "correct" clustering of a graphon?
Answer: The graphon cluster tree or, equivalently, the mergeon.

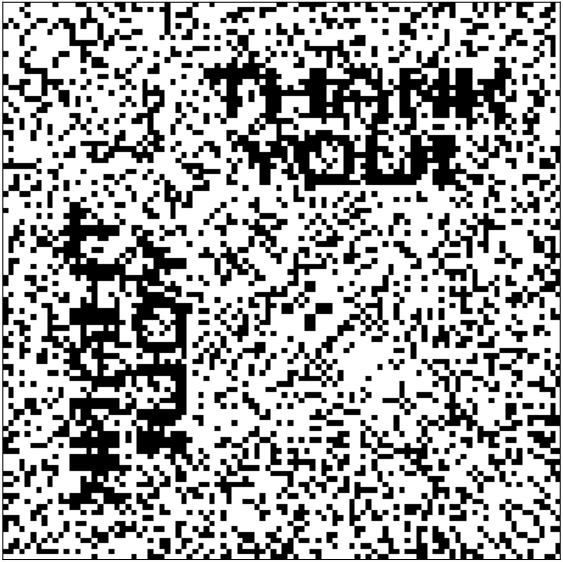
Question 2: What does it mean to recover the "correct" clustering?
Answer: Convergence in merge distortion to graphon cluster tree.

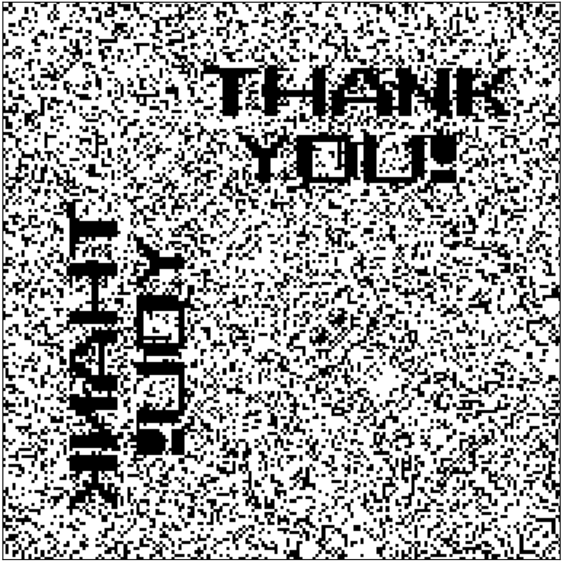
Question 3: How do we recover the correct clustering?
Answer: Modified neighborhood smoothing + single linkage clustering.










THANK YOU!


THANK YOU!
THANK YOU!

http://web.cse.ohio-state.edu/~eldridge/

Weak isomorphism

- Any graphon W defines a graph distribution.
- Not uniquely! Many graphons define the same distribution.
- The distribution is uniquely determined up to relabeling of W.

Weak isomorphism

- Any graphon W defines a graph distribution.
- Not uniquely! Many graphons define the same distribution.
- The distribution is uniquely determined up to relabeling of W.

Definition

A measure preserving transformation (i.e., graphon relabeling) $\varphi : [0,1] \to [0,1]$ is a Lebesgue-measurable function whose preimage preserves measure. That is, $\mu(\varphi^{-1}(A)) = \mu(A)$ for all measurable $A \subset [0,1]$.

Notation: $W^{\varphi}(x,y) = W(\varphi(x), \varphi(y))$.

Weak isomorphism

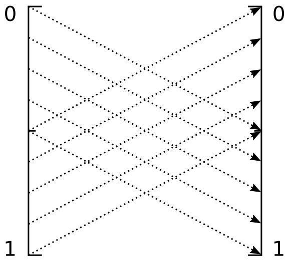
- Any graphon W defines a graph distribution.
- Not uniquely! Many graphons define the same distribution.
- The distribution is uniquely determined up to relabeling of W.

Definition

A measure preserving transformation (i.e., graphon relabeling) $\varphi : [0,1] \to [0,1]$ is a Lebesgue-measurable function whose preimage preserves measure. That is, $\mu(\varphi^{-1}(A)) = \mu(A)$ for all measurable $A \subset [0,1]$.

Notation: $W^{\varphi}(x,y) = W(\varphi(x), \varphi(y))$.

$$\varphi(x) = \begin{cases} x + 1/2 & x \leq 1/2, \\ x - 1/2 & x > 1/2 \end{cases}$$



Weak isomorphism

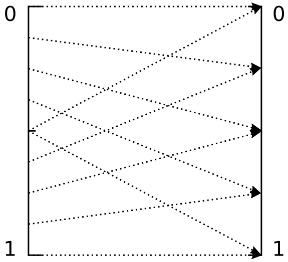
- Any graphon W defines a graph distribution.
- Not uniquely! Many graphons define the same distribution.
- The distribution is uniquely determined up to relabeling of W.

Definition

A measure preserving transformation (i.e., graphon relabeling) $\varphi : [0,1] \to [0,1]$ is a Lebesgue-measurable function whose preimage preserves measure. That is, $\mu(\varphi^{-1}(A)) = \mu(A)$ for all measurable $A \subset [0,1]$.

Notation: $W^{\varphi}(x,y) = W(\varphi(x), \varphi(y))$.

$$\varphi(x) = 2x \mod 1$$



Weak isomorphism

Definition (Lovász)

Two graphons W_1 and W_2 are weakly isomorphic if there exist measure preserving transformations φ_1 and φ_2 such that $W_1^{\varphi_1} \overset{\text{a.e.}}{=} W_2^{\varphi_2}$.

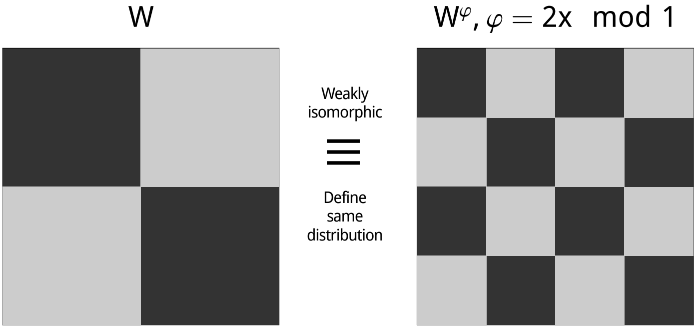
- W_1 and W_2 define the same distribution iff they are weakly isomorphic.

Weak isomorphism

Definition (Lovász)

Two graphons W_1 and W_2 are weakly isomorphic if there exist measure preserving transformations φ_1 and φ_2 such that $W_1^{\varphi_1} \stackrel{\text{a.e.}}{=} W_2^{\varphi_2}$.

- W_1 and W_2 define the same distribution iff they are weakly isomorphic.



The clusters of a graphon

1. Collect all subsets of $[0, 1]$ which should be clustered at λ:

$$\mathfrak{A}_\lambda = \{A \subset [0,1] : \mu(A) > 0 \text{ and A is connected } \forall\, \lambda' < \lambda.\}$$

2. If $A_1, A_2, A \in \mathfrak{A}_\lambda$, and $A_1 \cup A_2 \subset A$, then A_1, A_2, and A should all be in the same cluster at λ. Consider them equivalent.
   - Define equivalence relation on $\mathfrak{A}_\lambda$:

$$A_1 \multimap\!\!\circ_\lambda A_2 \iff \exists A \in \mathfrak{A}_\lambda, A \supset A_1 \cup A_2.$$

   - Read: A_1 is clustered with A_2 at level λ.
   - $\multimap\!\!\circ_\lambda$ partitions $\mathfrak{A}_\lambda$ into equivalence classes of sets which should be in the same cluster.

The clusters of a graphon

3. Define clusters to be "largest" element of each equivalence class.
   - Subtlety in defining "largest":
     - Suppose $\mathscr{A} \in \mathfrak{A}_\lambda / \multimap_\lambda$ is such an equivalence class.
     - Let A be any representative from $\mathscr{A}$, let Z be a set of zero measure.
     - $A' = A \cup Z$ is a representative of $\mathscr{A}$.
   - In general there is no representative of $\mathscr{A}$ which strictly contains all other representatives in $\mathscr{A}$
   - We can find reps. which contain every other rep. up to a null set, called the "essential maxima" of $\mathscr{A}$:

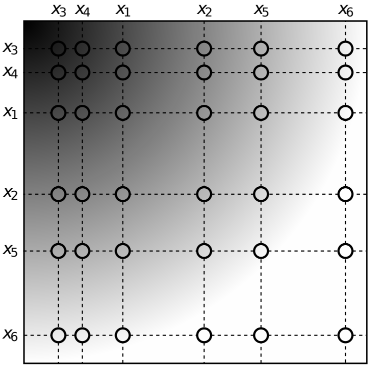
$$\operatorname{ess\,max} \mathscr{A} = \{A \in \mathscr{A} : \forall A' \in \mathscr{A}, \mu(A' \setminus A) = 0\}$$

   - The clusters of W at level λ are the essential maxima of each equivalence class:

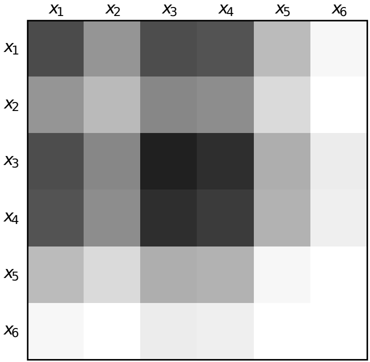
$$\mathbb{C}_W(\lambda) = \{\operatorname{ess\,max} \mathscr{A} : \mathscr{A} \in \mathfrak{A}_\lambda / \multimap_\lambda.\}$$

Consistent algorithms

- Intuitively, estimating the graphon is related to clustering.
- It suffices to estimate the so-called edge probability matrix.



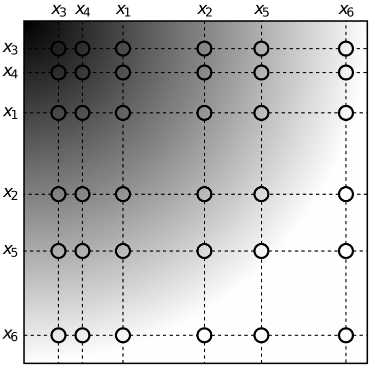
W



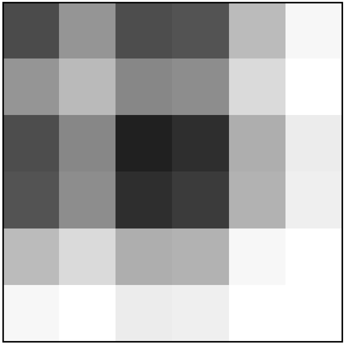
$P : P_{ij} = W(x_i, x_j)$

Consistent algorithms

- Intuitively, estimating the graphon is related to clustering.
- It suffices to estimate the so-called edge probability matrix.



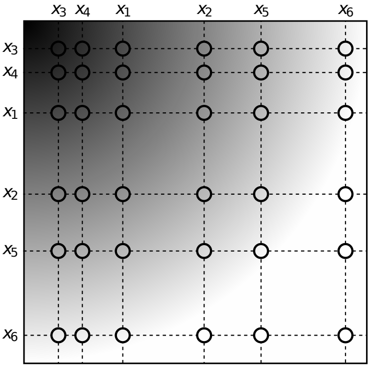
W



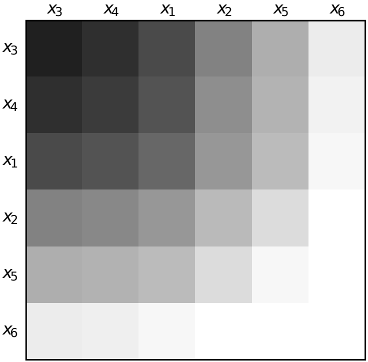
$P : P_{ij} = W(x_i, x_j)$

Consistent algorithms

- Intuitively, estimating the graphon is related to clustering.
- It suffices to estimate the so-called edge probability matrix.

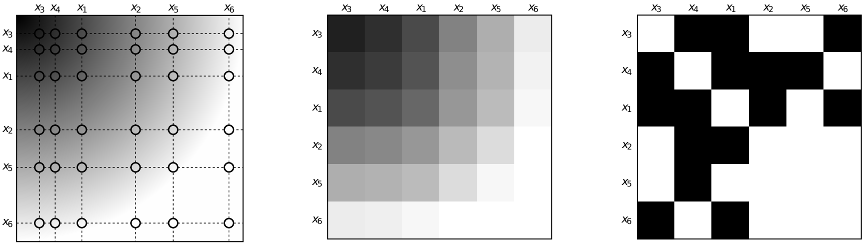


W



P (artificially permuted)

Sample an adjacency matrix A from P:

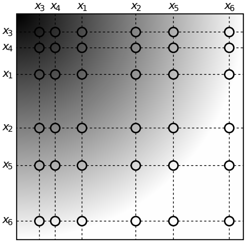


W

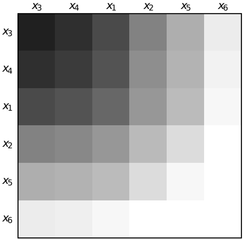
P
(artificially permuted)

A

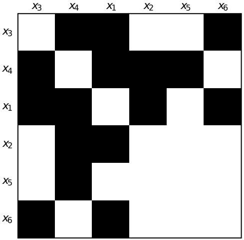
Sample an adjacency matrix A from P:



W



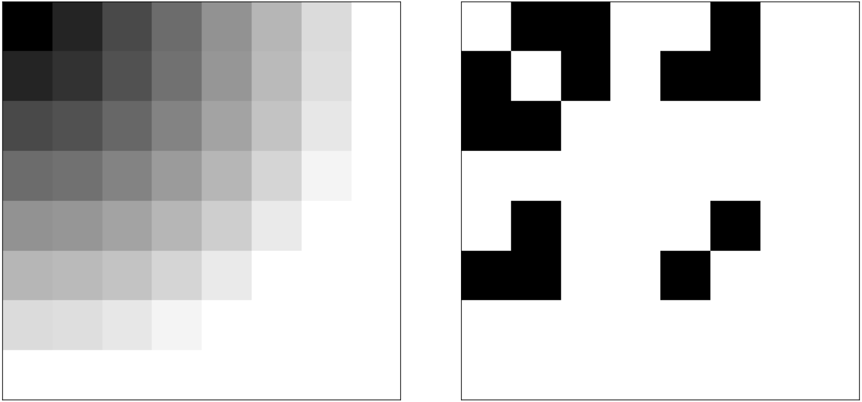
P
(artificially permuted)



A

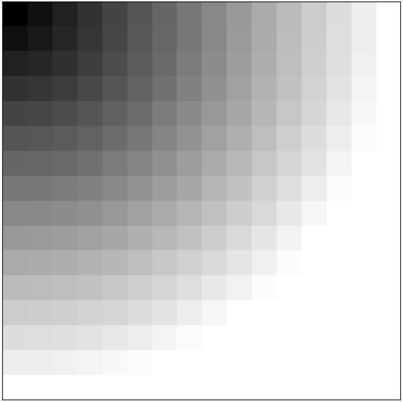
A is a poor estimate of P.

$$n = 8$$

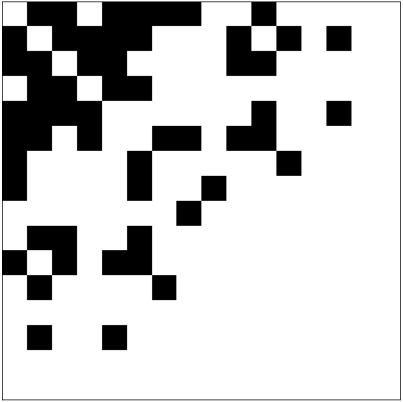


P A

$n = 16$

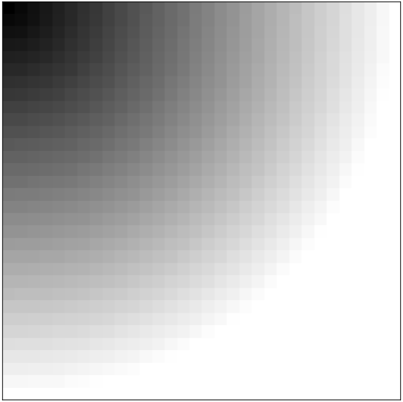


P

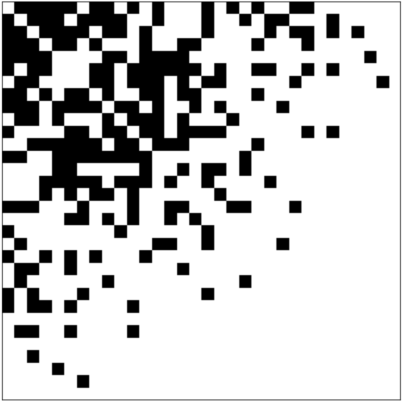


A

$n = 32$

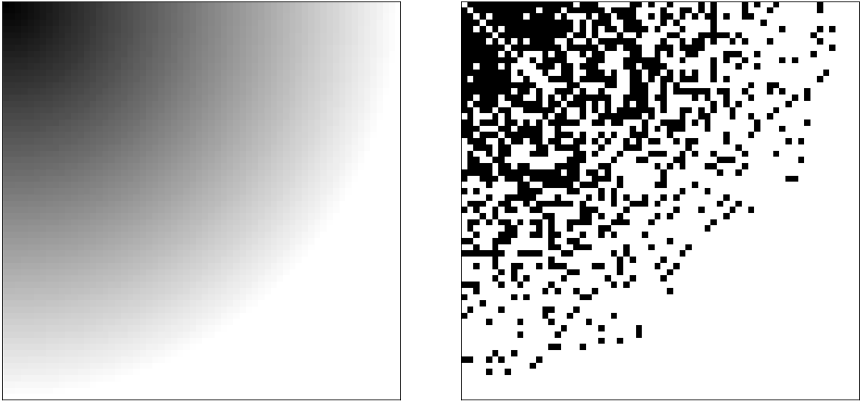


P



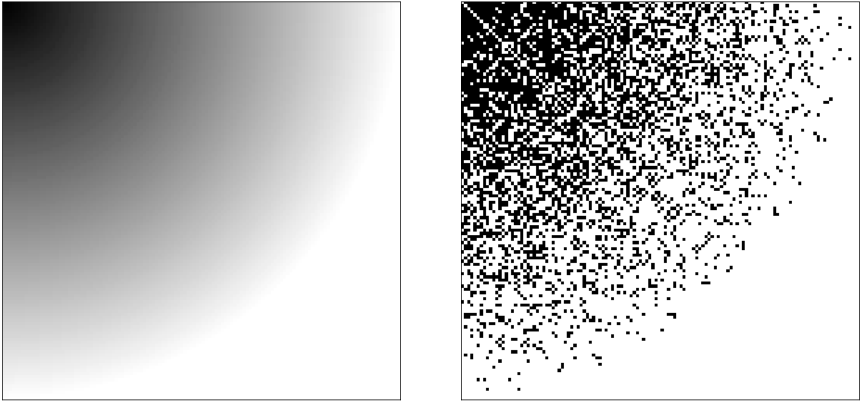
A

n = 64



P A

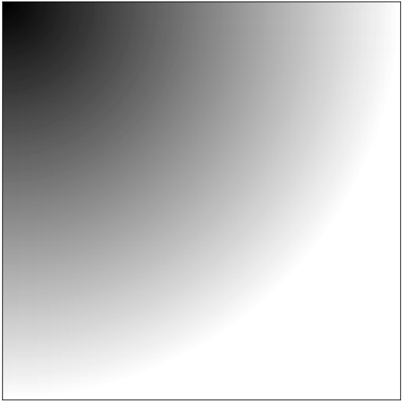
$n = 128$



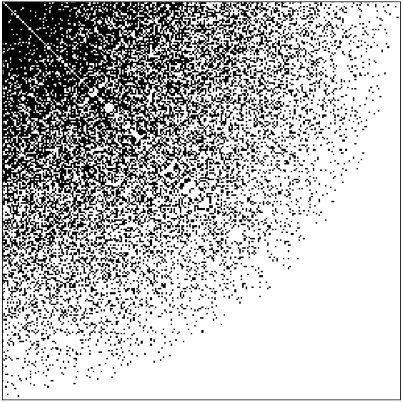
P

A

$n = 256$



P

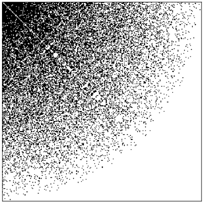


A

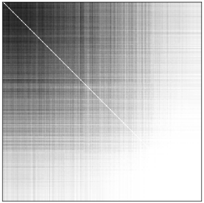
Edge probability estimation

Goal: Compute estimated edge probabilities $\hat{P}$ from A.

P



A



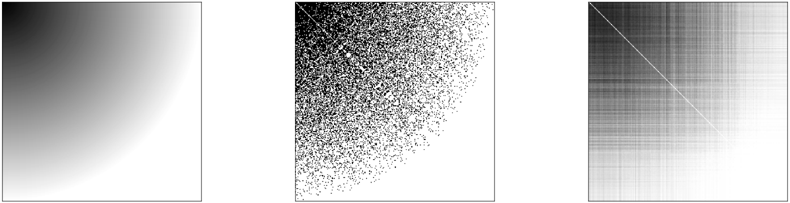
$\hat{P}$

Theorem

If $\|P - \hat{P}\|_\infty \to 0$ in probability as $n \to \infty$, then single linkage clustering on $\hat{P}$ is a consistent clustering method.

Edge probability estimation

Goal: Compute estimated edge probabilities $\hat{P}$ from A.



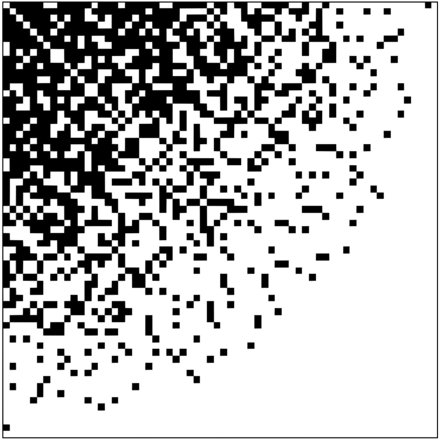
P $\qquad$ A $\qquad$ $\hat{P}$

Theorem

If $\|P - \hat{P}\|_\infty \to 0$ in probability as $n \to \infty$, then single linkage clustering on $\hat{P}$ is a consistent clustering method.

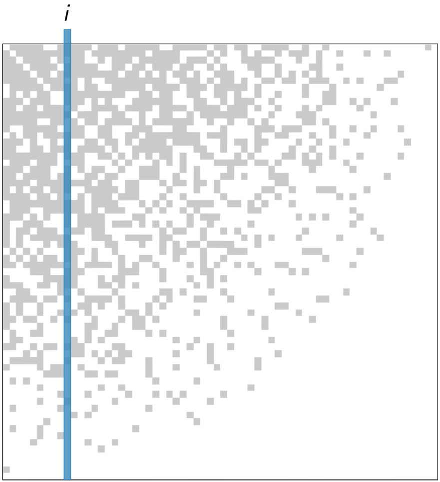
- We need a suitable estimator $\hat{P}$ of edge probabilities.
- Recently, Zhang et al. (2015) proposed neighborhood smoothing.

Neighborhood smoothing



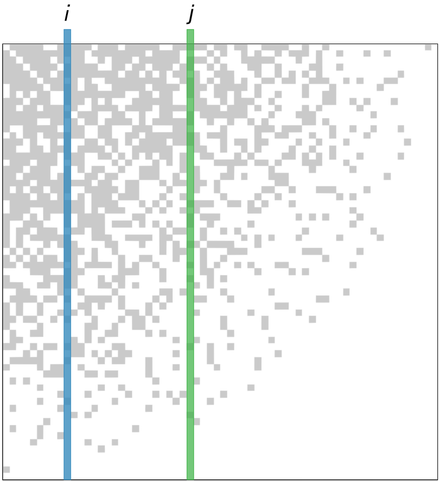
Given A, the adjacency matrix of a sampled graph...

Neighborhood smoothing



Consider a node i and its corresponding column of A.

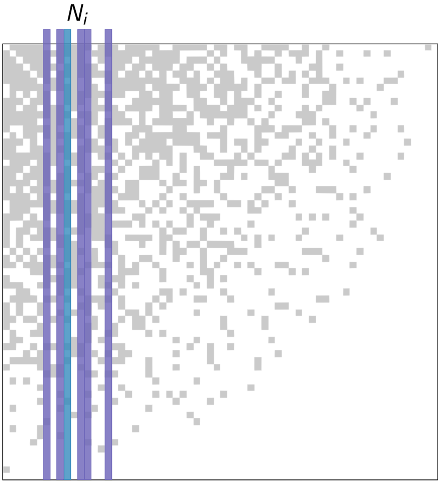
Neighborhood smoothing



Measure similarity to every other node j:

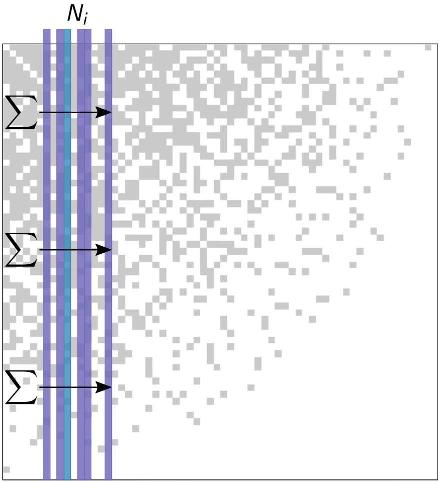
$$d(i,j) = \max_{k \neq i,j} \left| (A^2)_{ik} - (A^2)_{jk} \right|$$

Neighborhood smoothing



Form neighborhood N_i of nodes most similar to i.

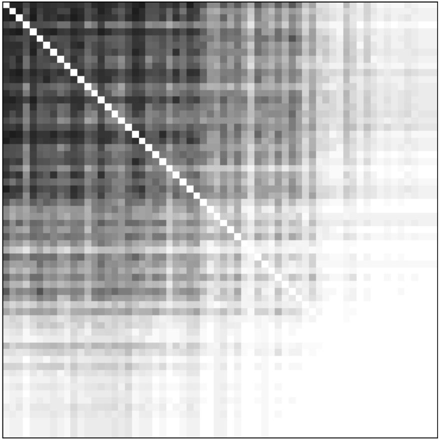
Neighborhood smoothing



Average within neighborhood to estimate edge probability:

$$\hat{P}_{ij} = \frac{1}{2|N_i|} \sum_{i' \in N_i} A_{i'j} + \frac{1}{2|N_j|} \sum_{j' \in N_j} A_{ij'}$$

Neighborhood smoothing



The result is a smoothed estimate $\hat{P}$ of edge probabilities.