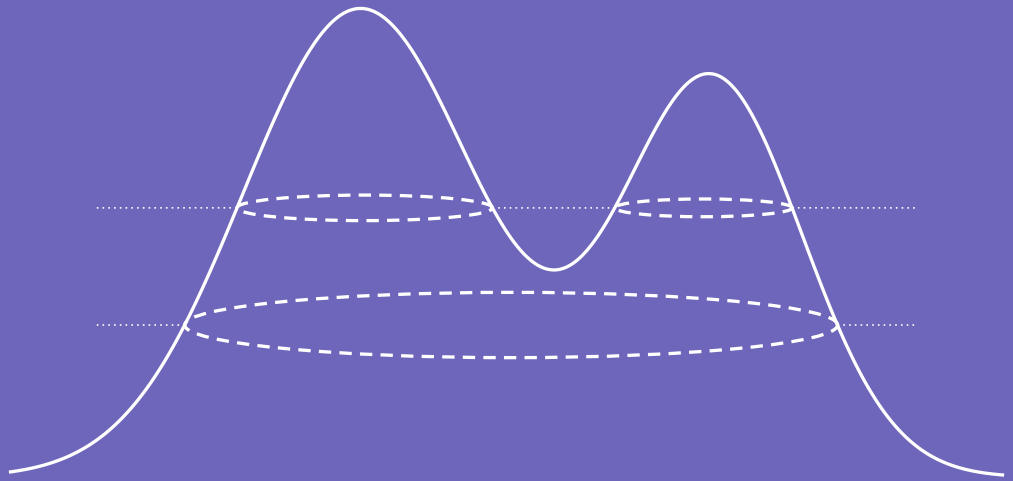
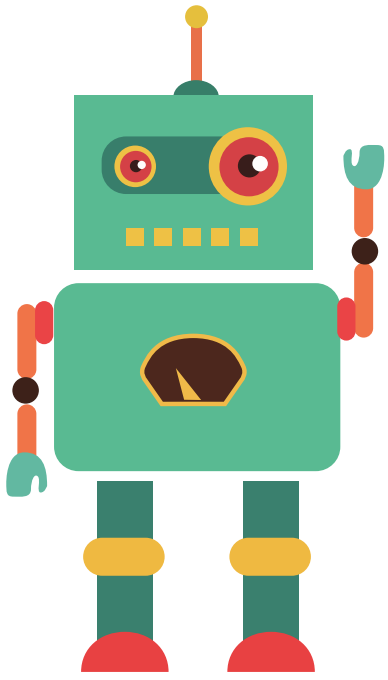
Clustering correctly

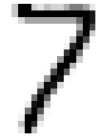


Justin Eldridge

The Ohio State University



?



?

one seven one seven one seven seven seven one

seven one seven one seven one one one seven

machine learning
without a teacher

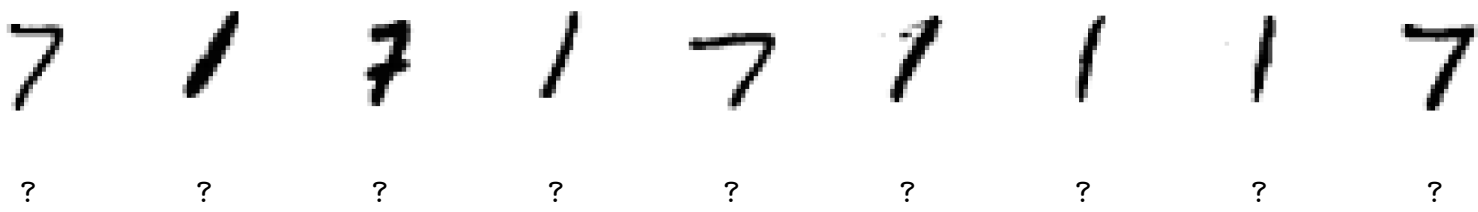
machine learning
without a teacher

- A teacher's time is expensive.

machine learning
without a teacher

- A teacher's time is expensive.
- A teacher may not exist.

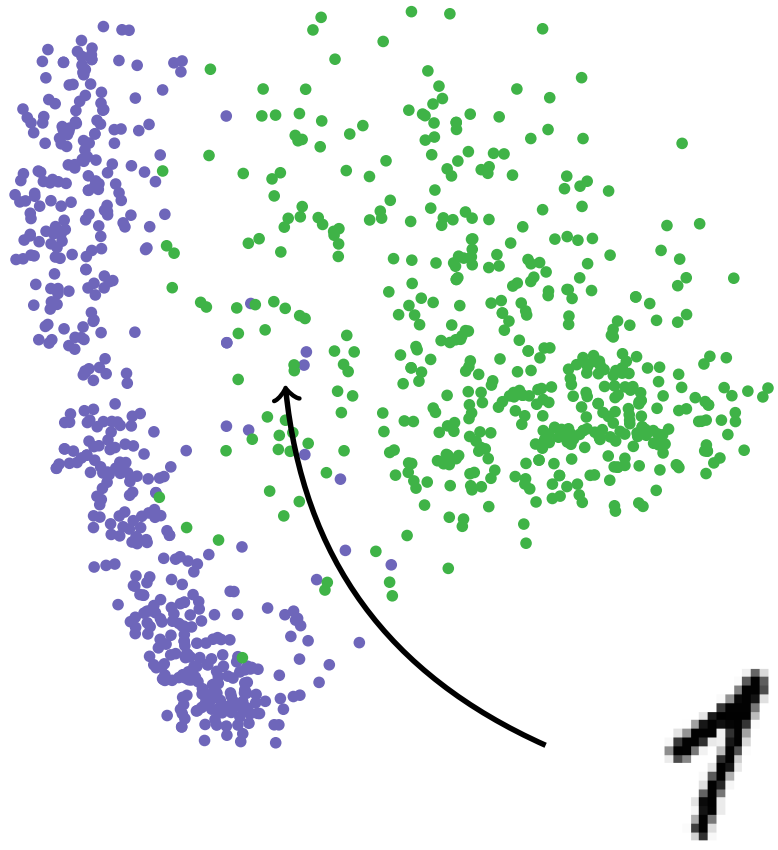
? ? ? ? ? ? ? ? ?



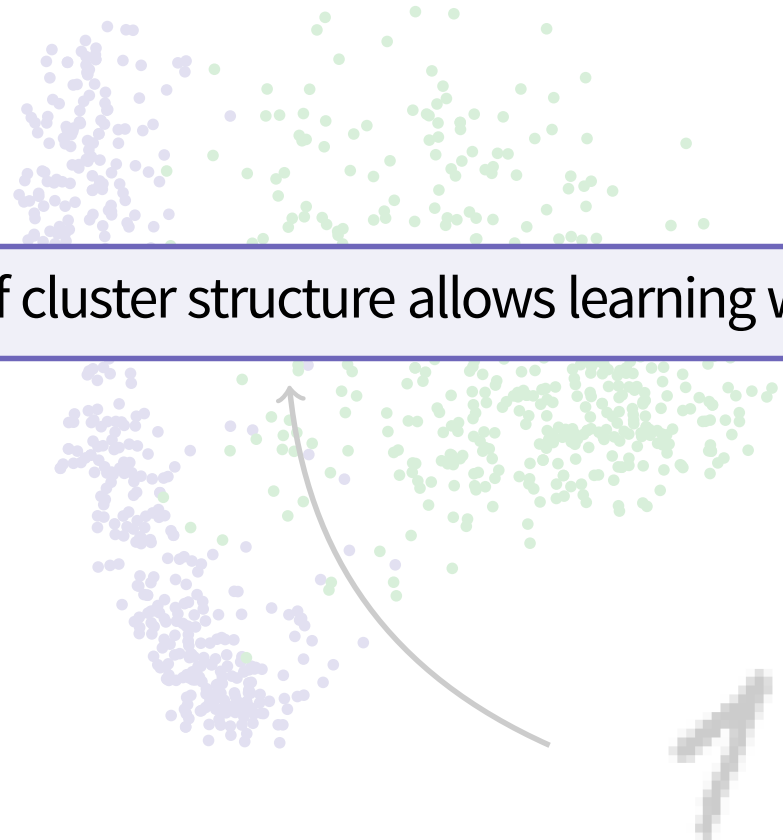
? ? ? ? ? ? ? ? ?

Treat image as a vector in $\mathbb{R}^{784}$,
project to $\mathbb{R}^2$ using principal components.

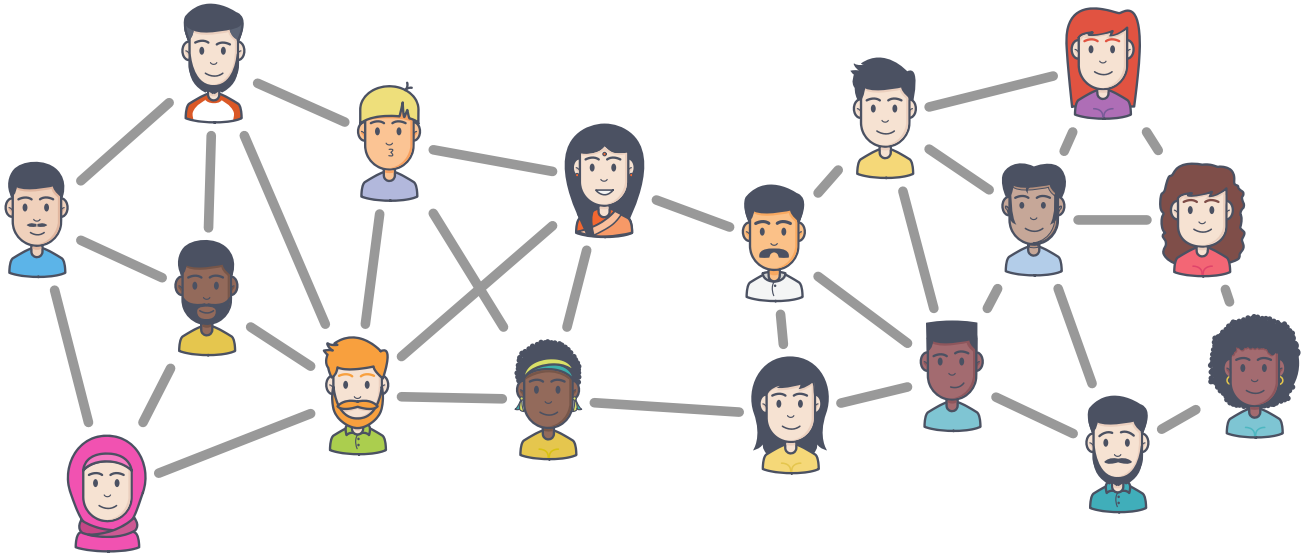
Treat image as a vector in $\mathbb{R}^{784}$,
project to $\mathbb{R}^2$ using principal components.

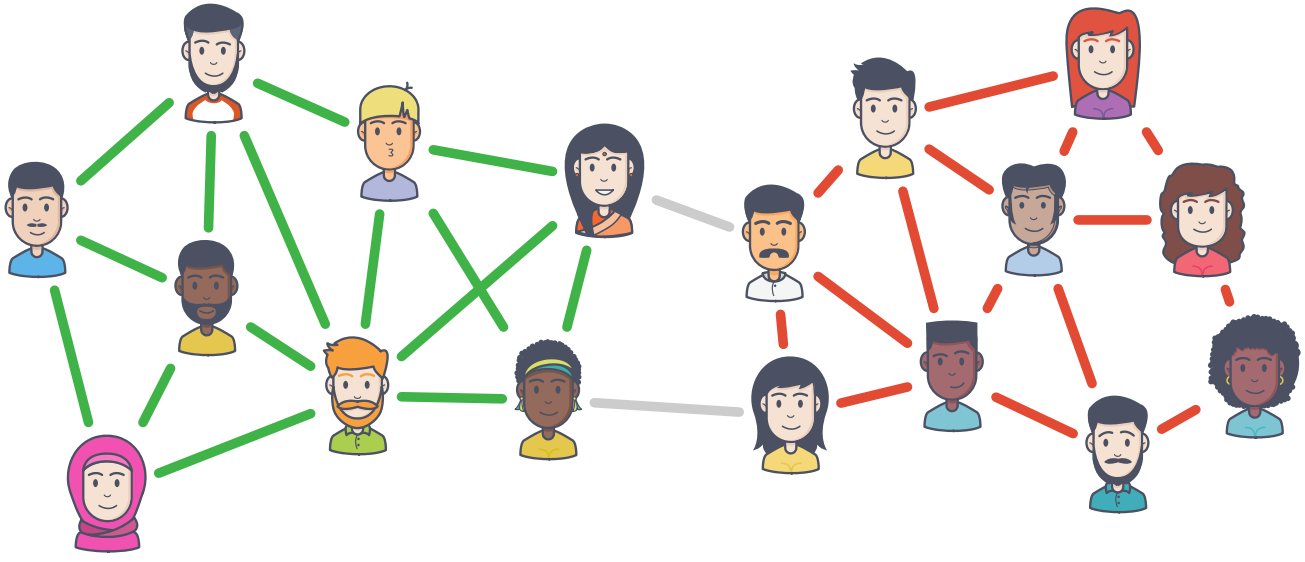


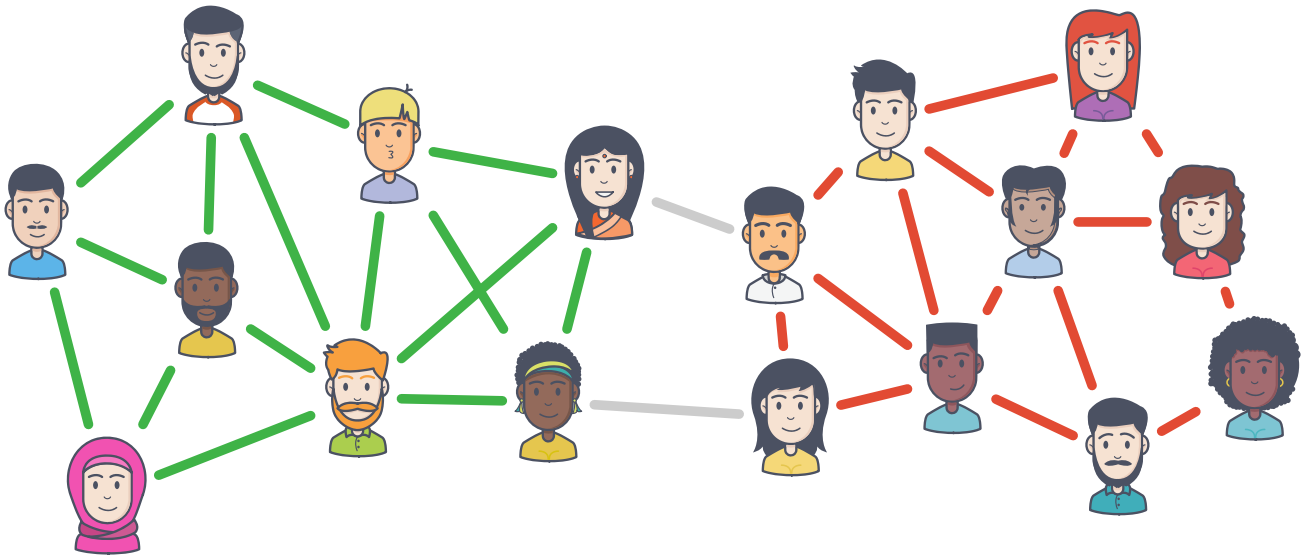
Treat image as a vector in $\mathbb{R}^{784}$,
project to $\mathbb{R}^2$ using principal components.



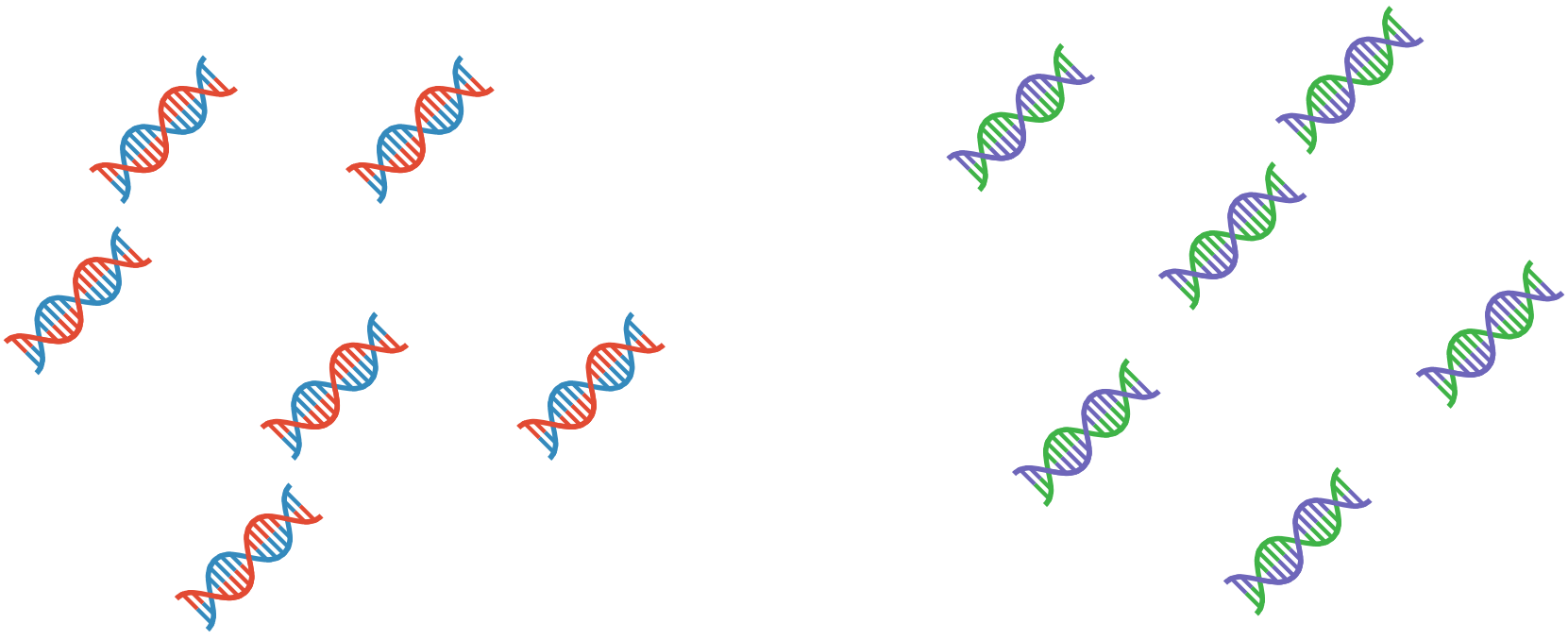
The phenomenon of cluster structure allows learning without a teacher.







Understanding cluster structure of social network helps in preventing spread of infectious diseases.



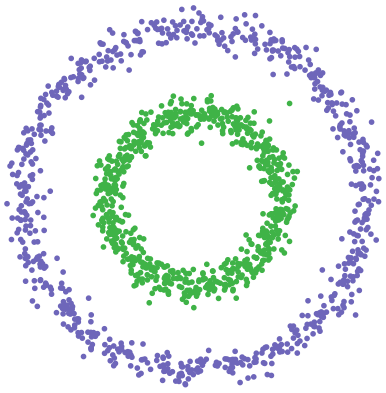
Identify subtypes of cancer via clustering of DNA microarrays.

Clustering customers based on past purchases allows accurate recommendation of new products.

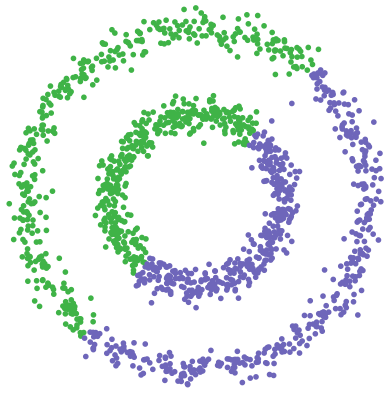
We can recover the structure of data without a teacher through clustering.

Problem: Different clustering methods recover different structures.

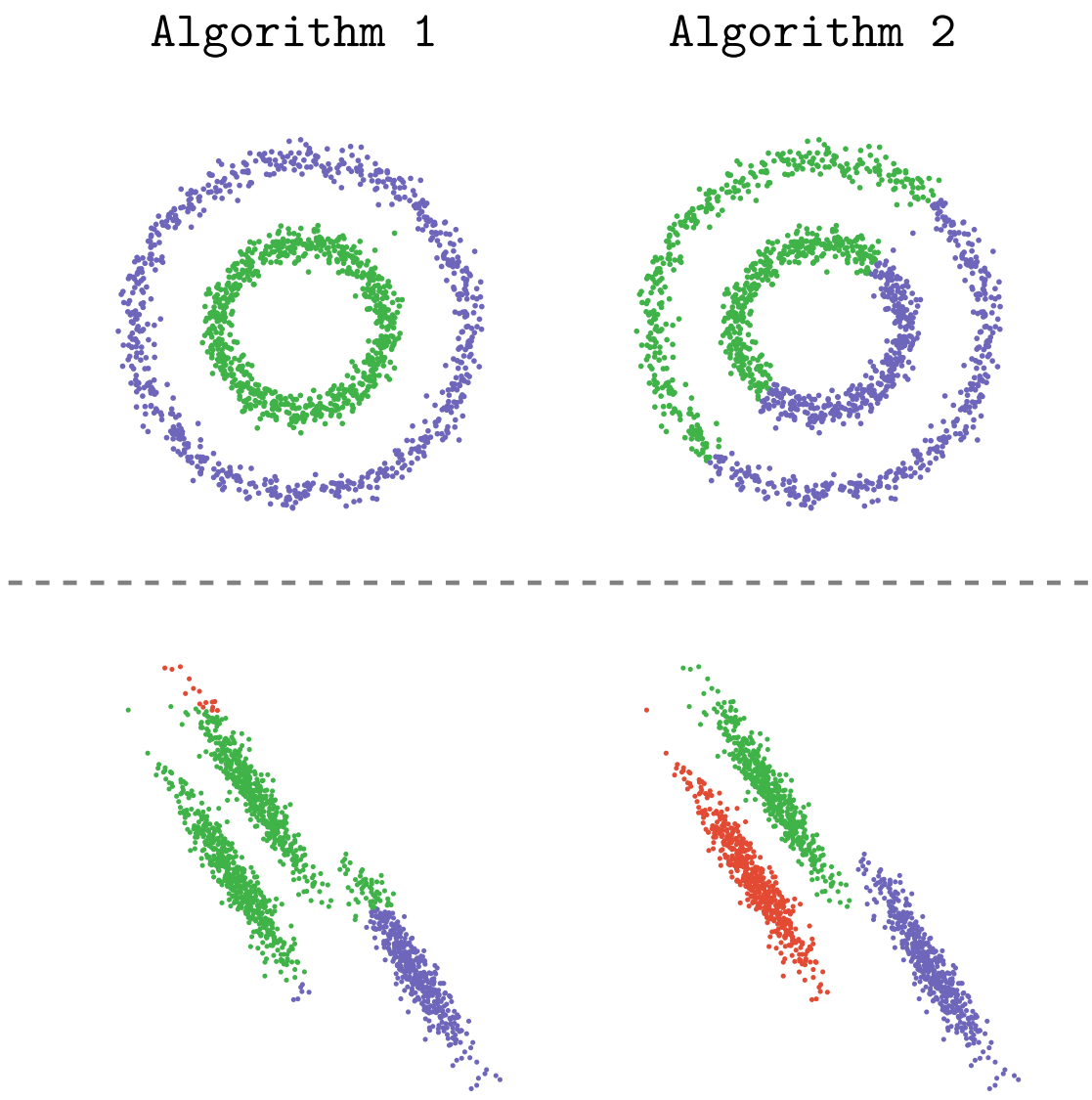
Algorithm 1



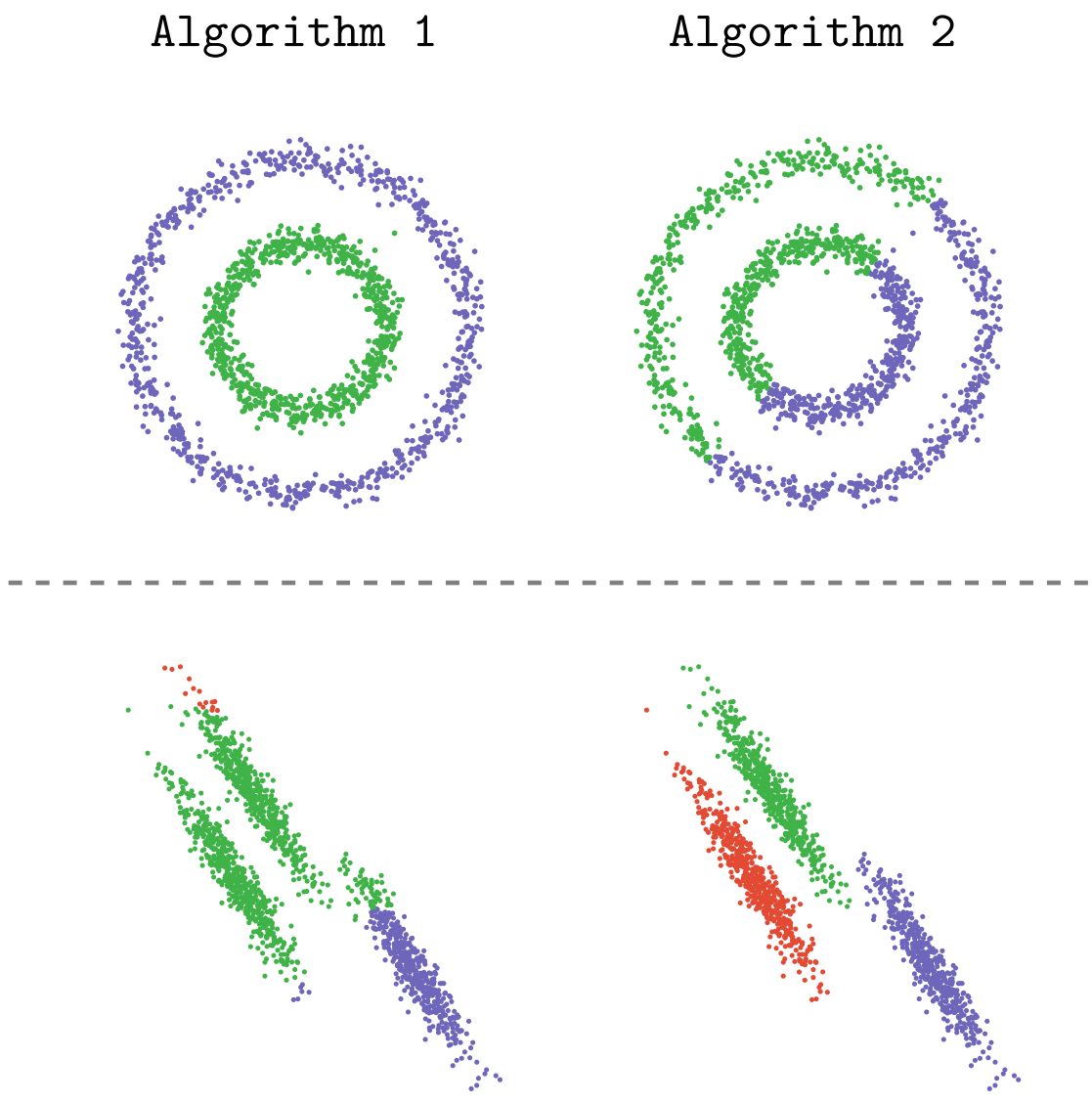
Algorithm 2



Problem: Different clustering methods recover different structures.



Problem: Each method has own internal idea of the “correct” clustering.



There is no “best” clustering method (no free lunch).
It is up to the user to pick an algorithm which matches their goals.

A theory of clustering:

- Formalize the goal: What is the correct clustering in this setting?
- What algorithm (if any) obtains it?

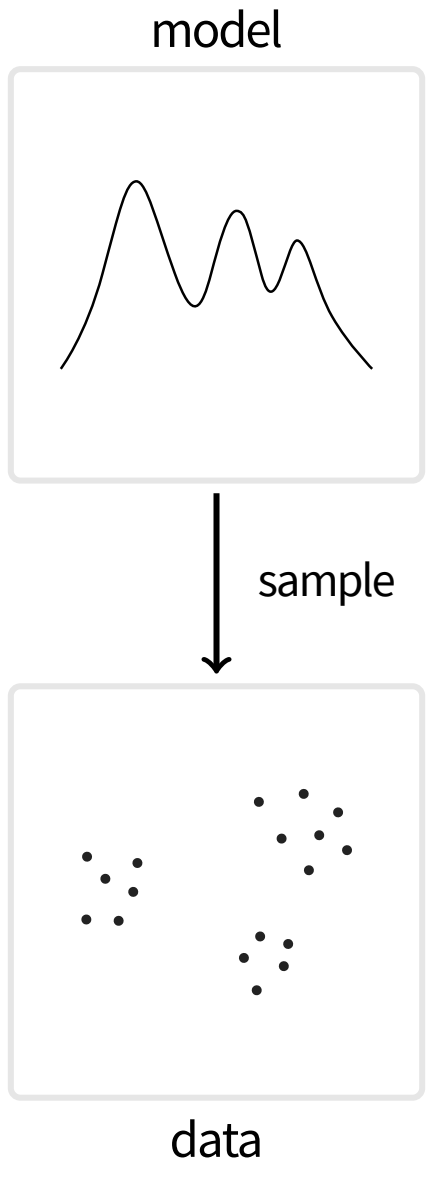
Without such a theory:

- Clustering is often *ad hoc*.
- Interpretation of clusters is unclear.

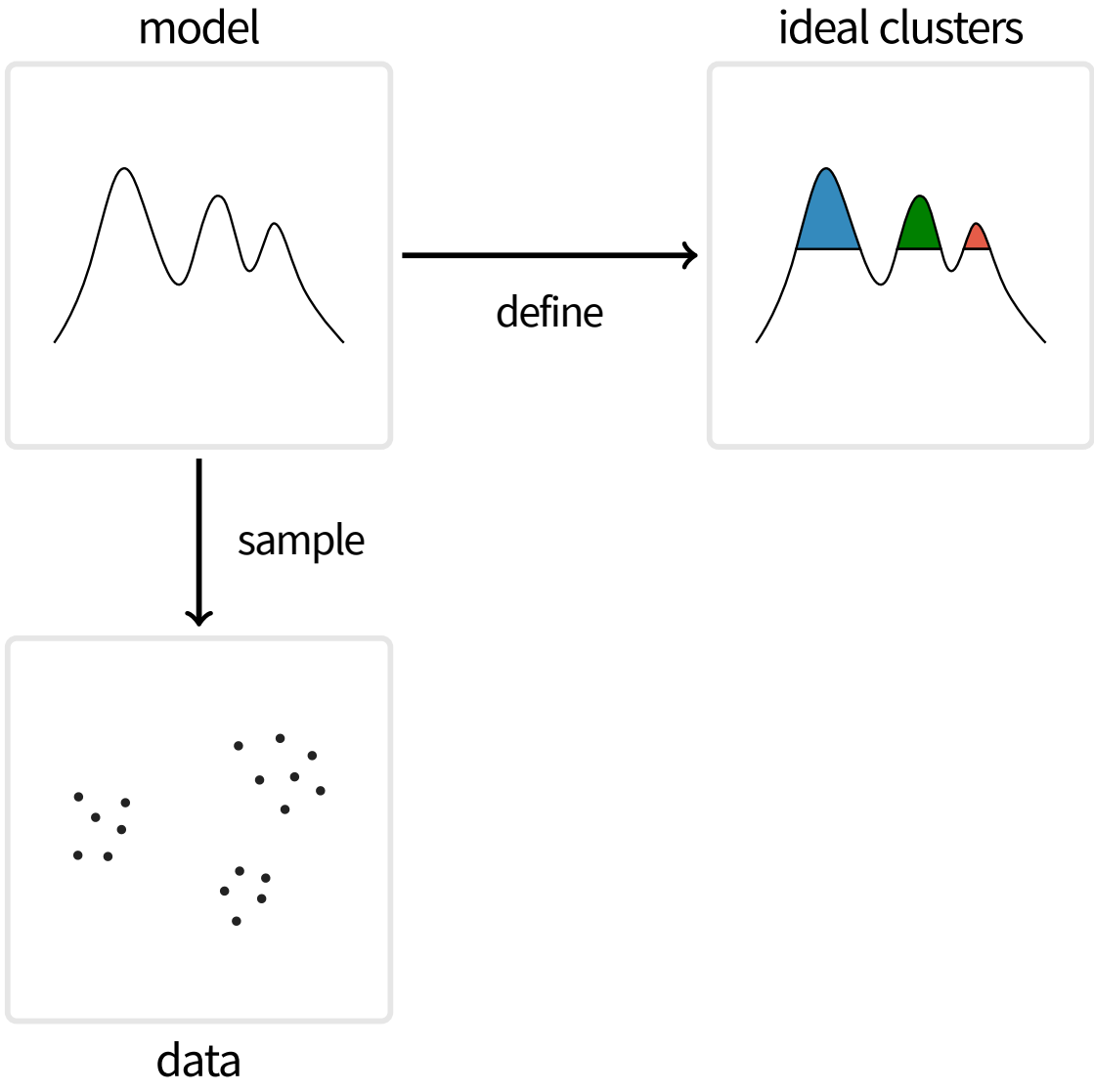
In this talk, we focus on

statistical theories of clustering.

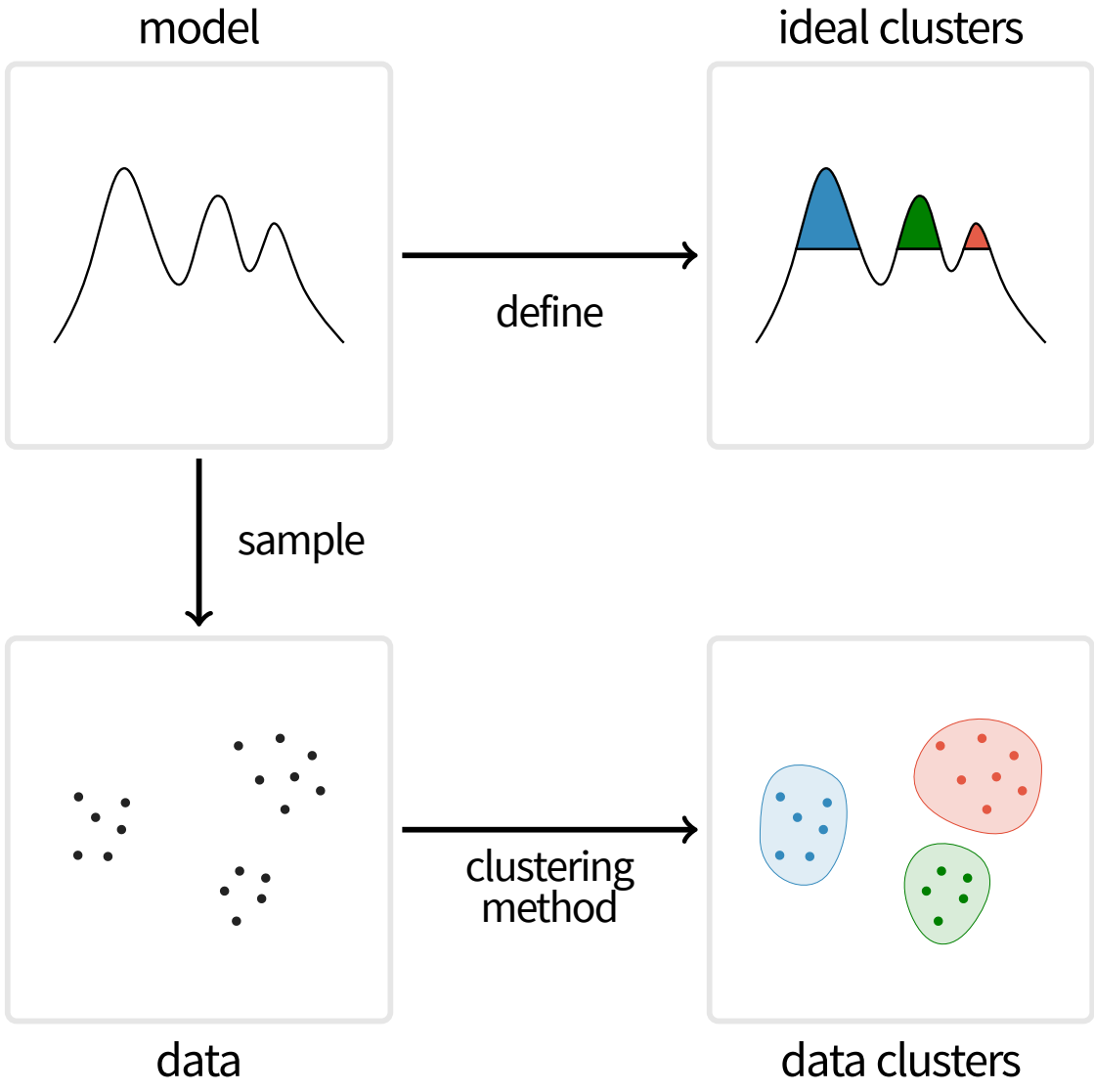
Step 0: Model the data-generating process.



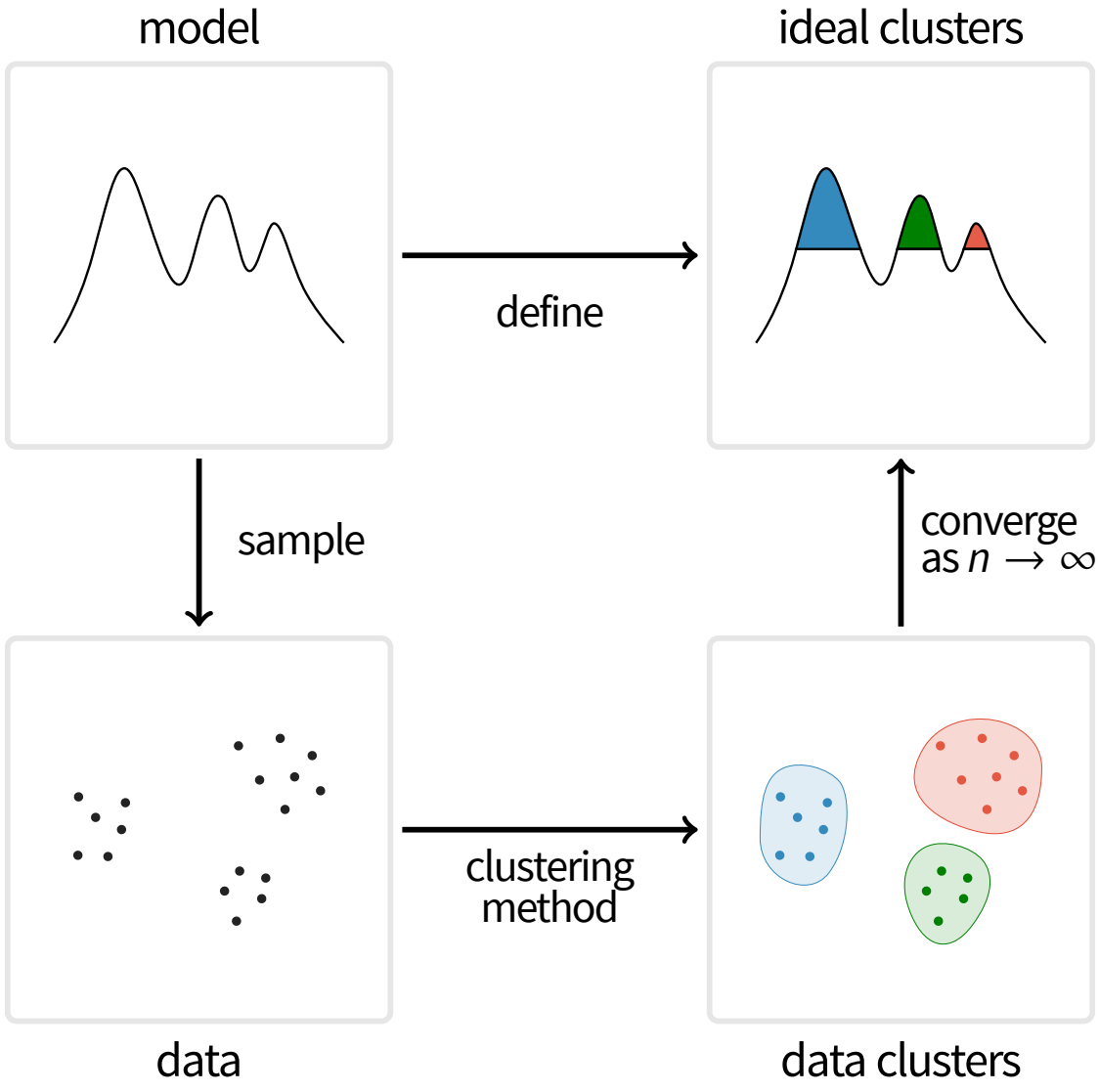
Step 1: Define the ideal clusters of the model.



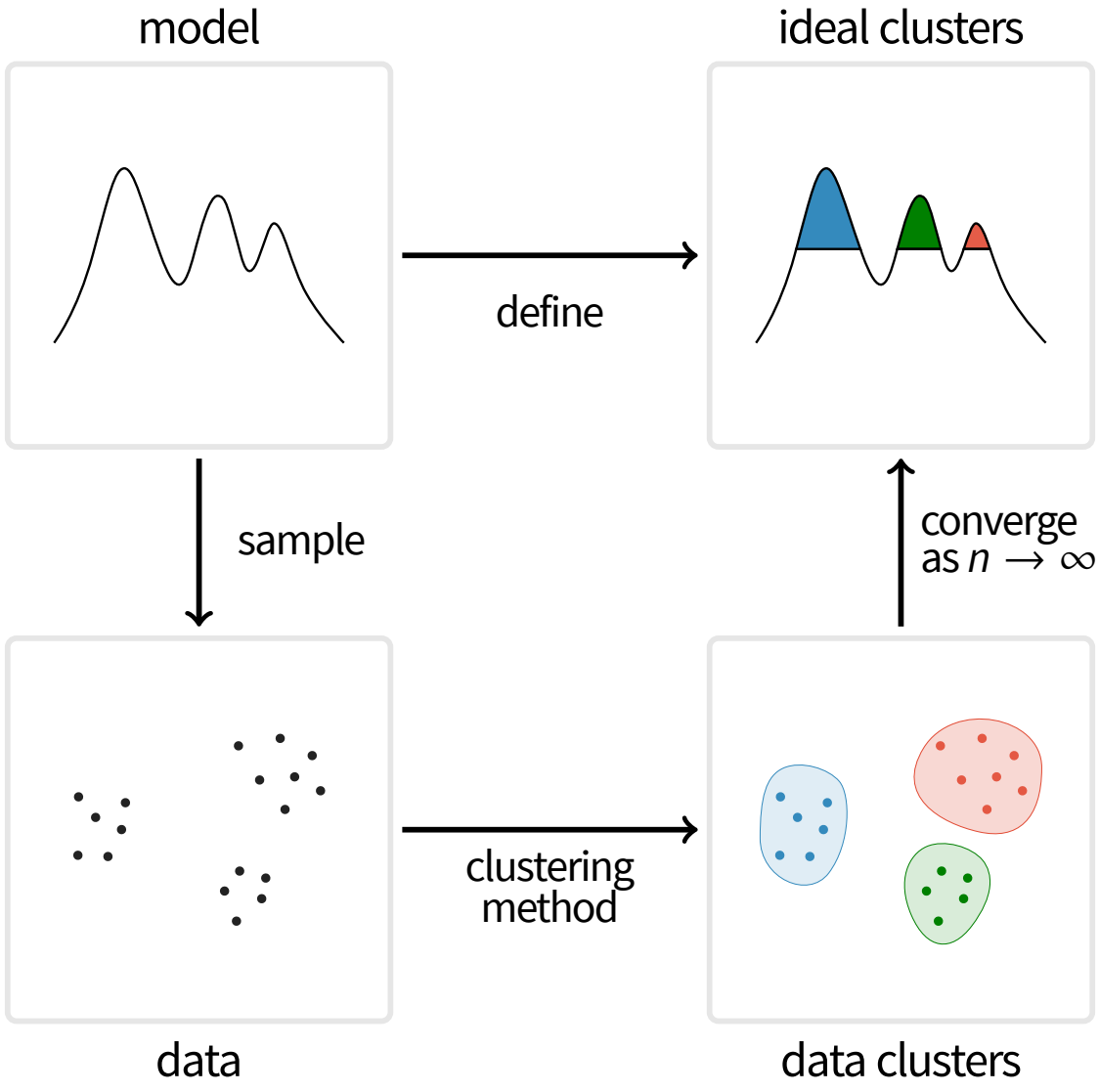
Step 2: Define convergence to the ideal clusters.



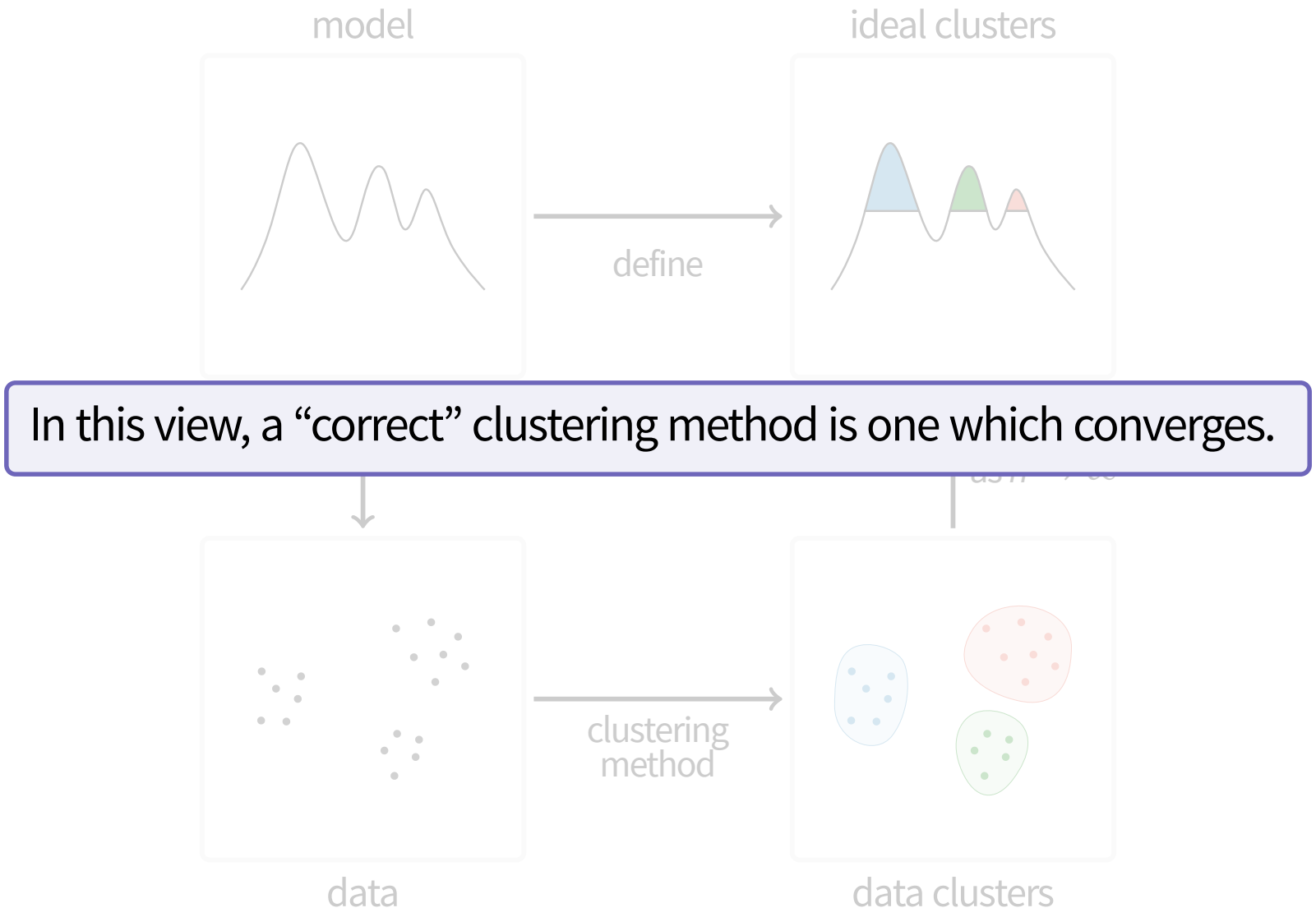
Step 2: Define convergence to the ideal clusters.



Step 3: Identify algorithms which converge.



Step 3: Identify algorithms which converge.



Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

with Mikhail Belkin and Yusu Wang

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

- Well-studied since the 1960's.
- Existing notion of clusters, convergence (Hartigan)
- Proving convergence of algorithms took 30 years.

- We show that Hartigan's notion is *very weak*.
- Introduce stronger notion of convergence.
- Prove that algorithms converge.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

- Graph clustering.
- Much of existing theory in, e.g., blockmodel.
- The graphon represents a much richer model.
- Graphon was clustering virtually unstudied.

- We define clusters of the graphon.
- Define a strong notion of convergence.
- Introduce a new graph clustering algorithm
- Prove its convergence.

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

Context

Approaches to “correct” clustering.

- Optimization
- Axiomatic
- Statistical

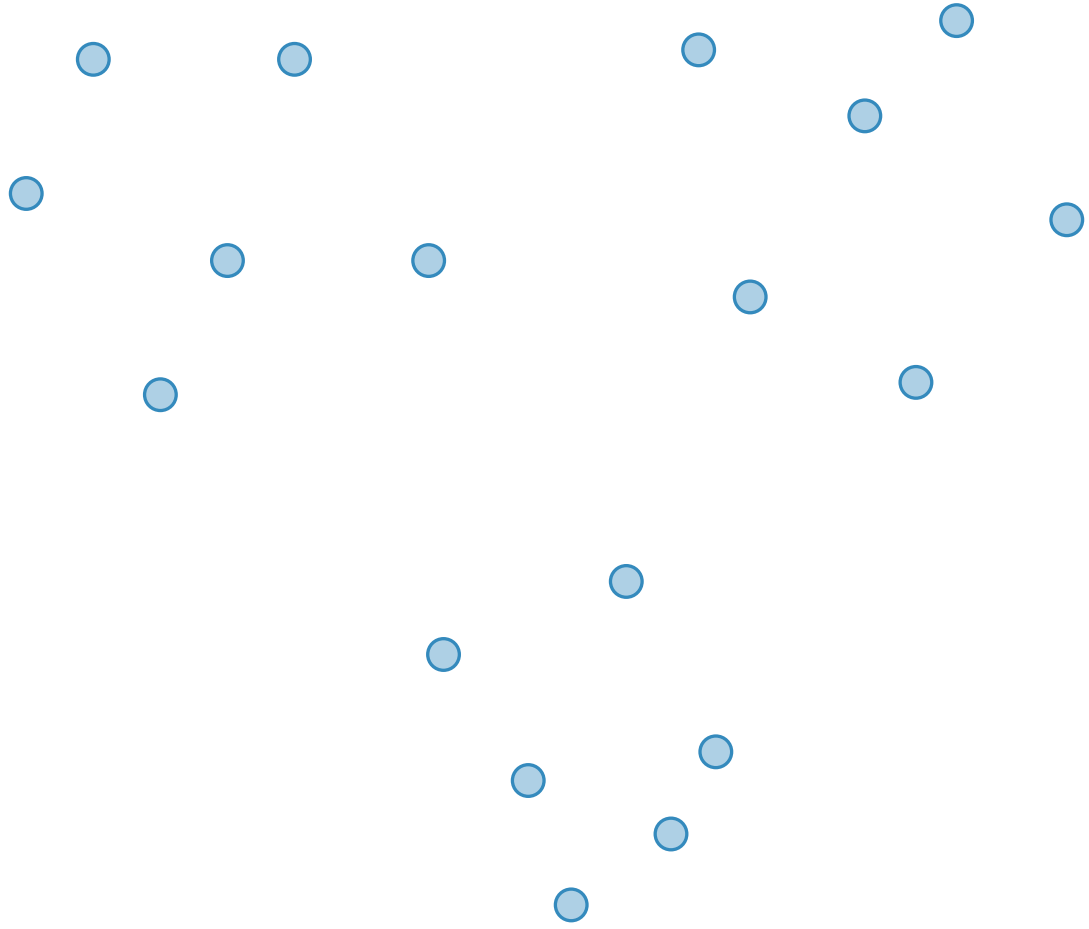
Part I: Density

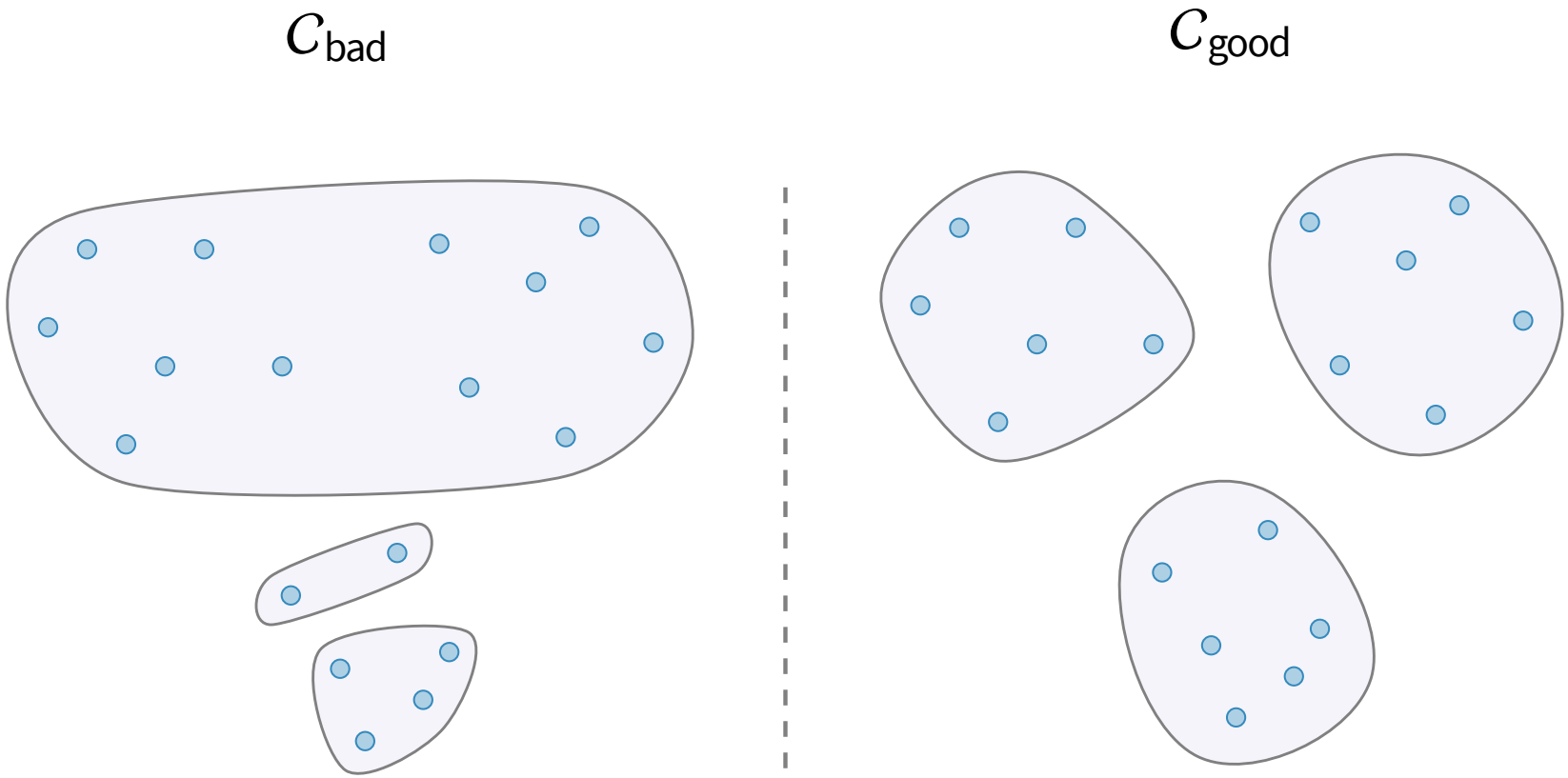
Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

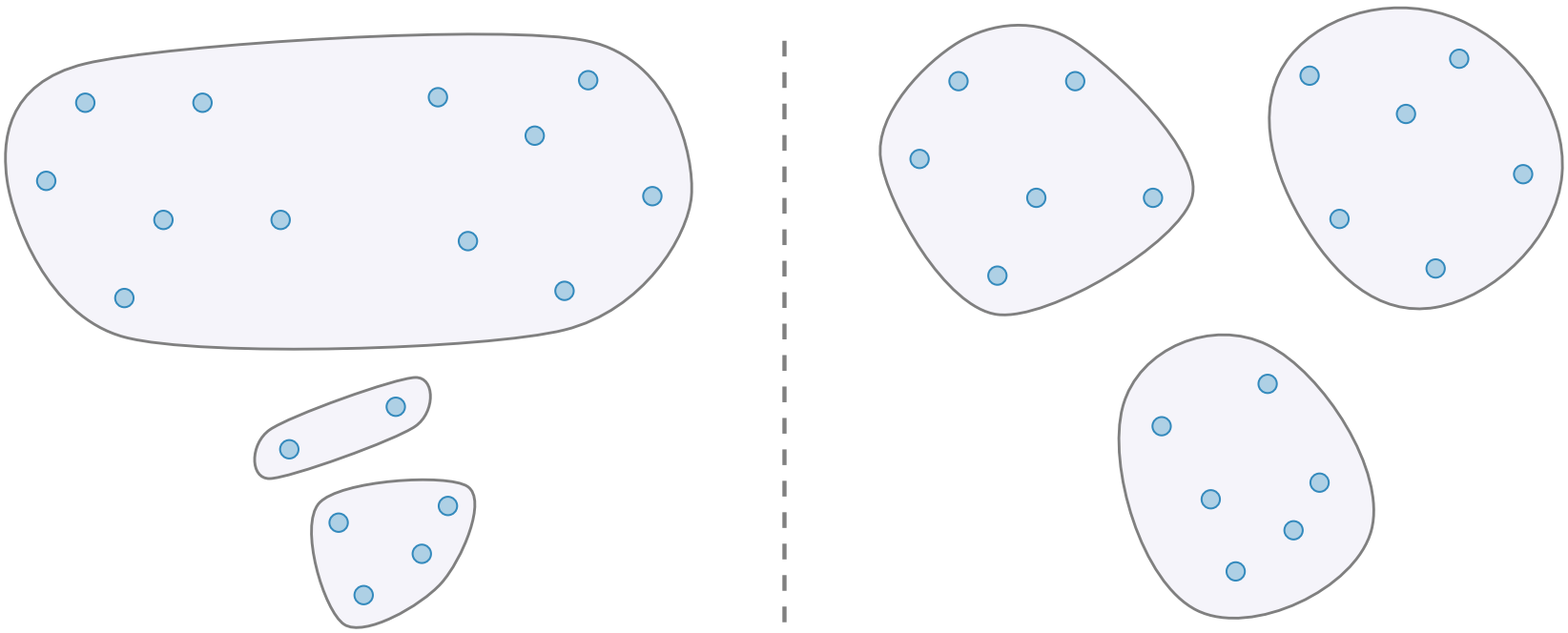
Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

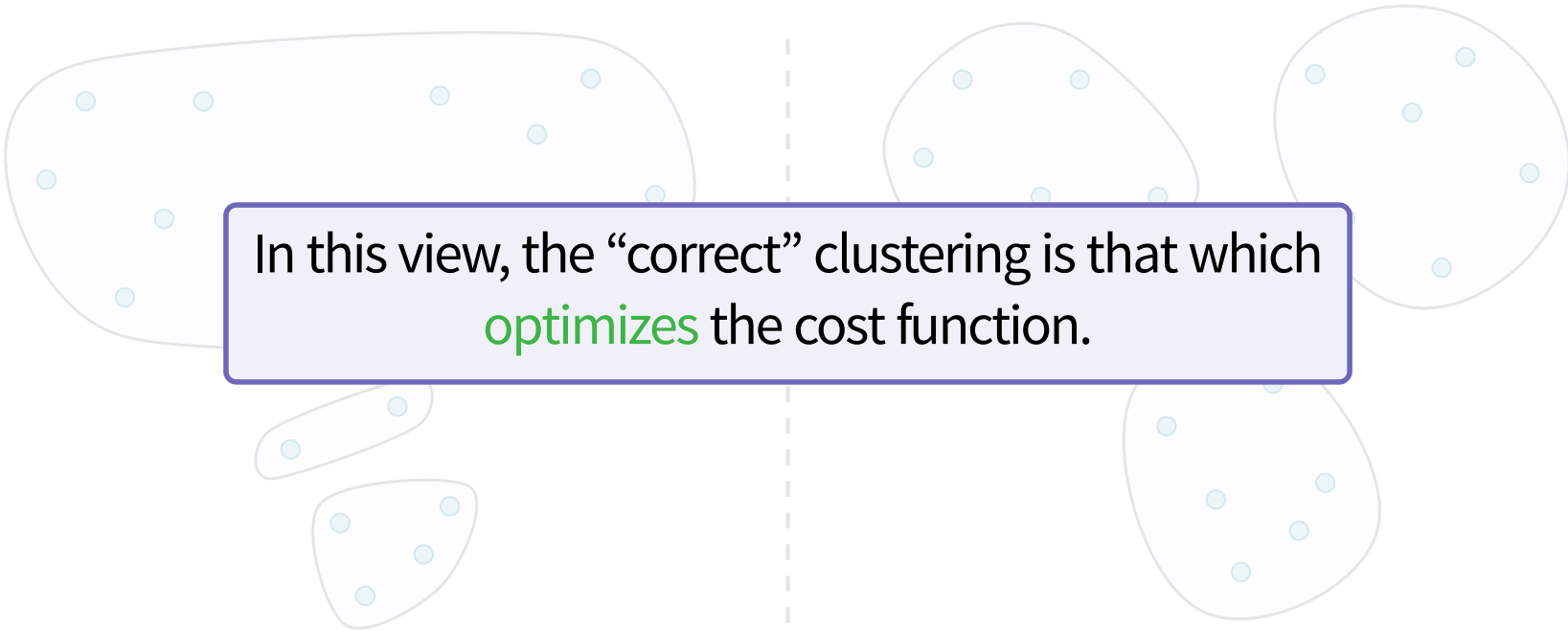



C_{bad}
C_{good}

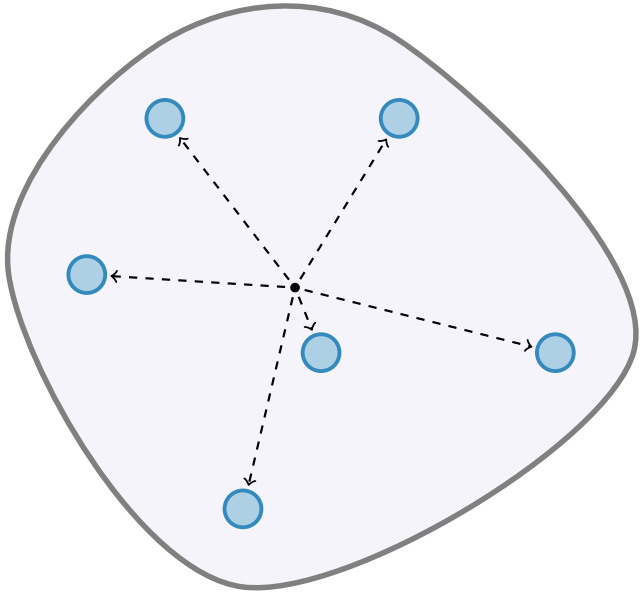
$\text{Cost}(C_{\text{bad}})$ > $\text{Cost}(C_{\text{good}})$

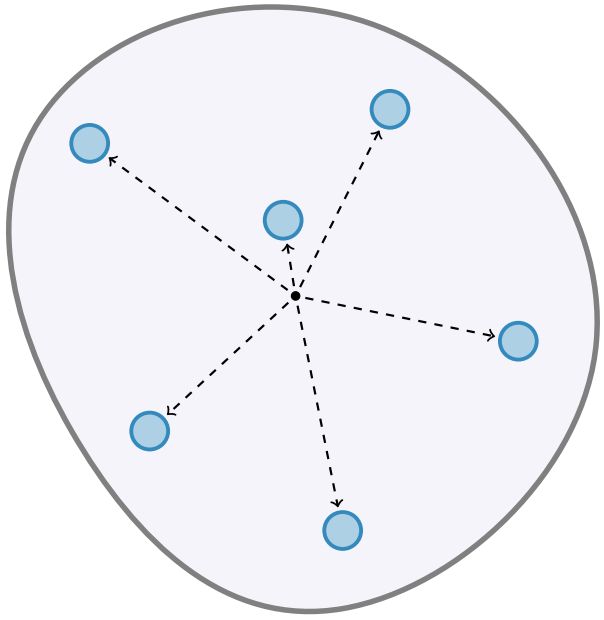


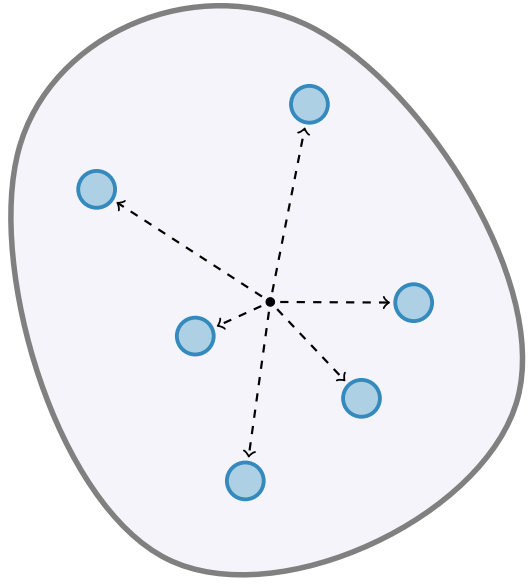
$\text{Cost}(C_{\text{bad}}) > \text{Cost}(C_{\text{good}})$



In this view, the “correct” clustering is that which optimizes the cost function.







k-Means-Cost($\mathcal{C}$)

$$= \sum_{C \in \mathcal{C}} \text{SSE}(C)$$

The “correct” clustering is that which minimizes the k-means cost.

Does an algorithm exist which obtains the correct clustering?

Optimization problem:
For fixed k, optimize k-Means-Cost over k-clusterings.

The "correct" clustering is that which minimizes the k-means cost.

Does an algorithm exist which obtains the correct clustering?

Optimization problem:
For fixed k, optimize k-Means-Cost over k-clusterings.

Dasgupta (2009): NP-Hard, even for $k = 2$.

The “correct” clustering is that which minimizes the k-means cost.

Does an algorithm exist which obtains the correct clustering?

Optimization problem:
For fixed k, optimize k-Means-Cost over k-clusterings.

Dasgupta (2009): NP-Hard, even for $k = 2$.

The "correct" clustering is that which minimizes the k-means cost.

Does an algorithm exist which obtains the correct clustering?

Optimization problem:
For fixed k, optimize k-Means-Cost over k-clusterings.

Dasgupta (2009): NP-Hard, even for $k = 2$.

Approximation algorithm:
Lloyd's method (a.k.a., the k-means algorithm)

The "correct" clustering is that which minimizes the k-means cost.

Does an algorithm exist which obtains the correct clustering?

Optimization problem:
For fixed k, optimize k-Means-Cost over k-clusterings.

Dasgupta (2009): NP-Hard, even for $k = 2$.

Approximation algorithm:
Lloyd's method (a.k.a., the k-means algorithm)

Kanungo (2003): Unbounded approximation ratio.

Advantages of optimization approach:

- Some clustering tasks are naturally quantifiable: e.g., compression.
- Straightforward to incorporate constraints.
- Easy to compare different clusterings of the same data.
- Use well-weathered machinery of optimization.

Disadvantages:

- Common cost functions are NP-Hard to optimize.
- Unclear how the optimum relates to data-generation process.

Context

Approaches to “correct” clustering.

- Optimization
- Axiomatic
- Statistical

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

Idea: establish rules (axioms) for the behavior of clustering methods.

In this view, a “correct” clustering is one produced by an algorithm which obeys the axioms.

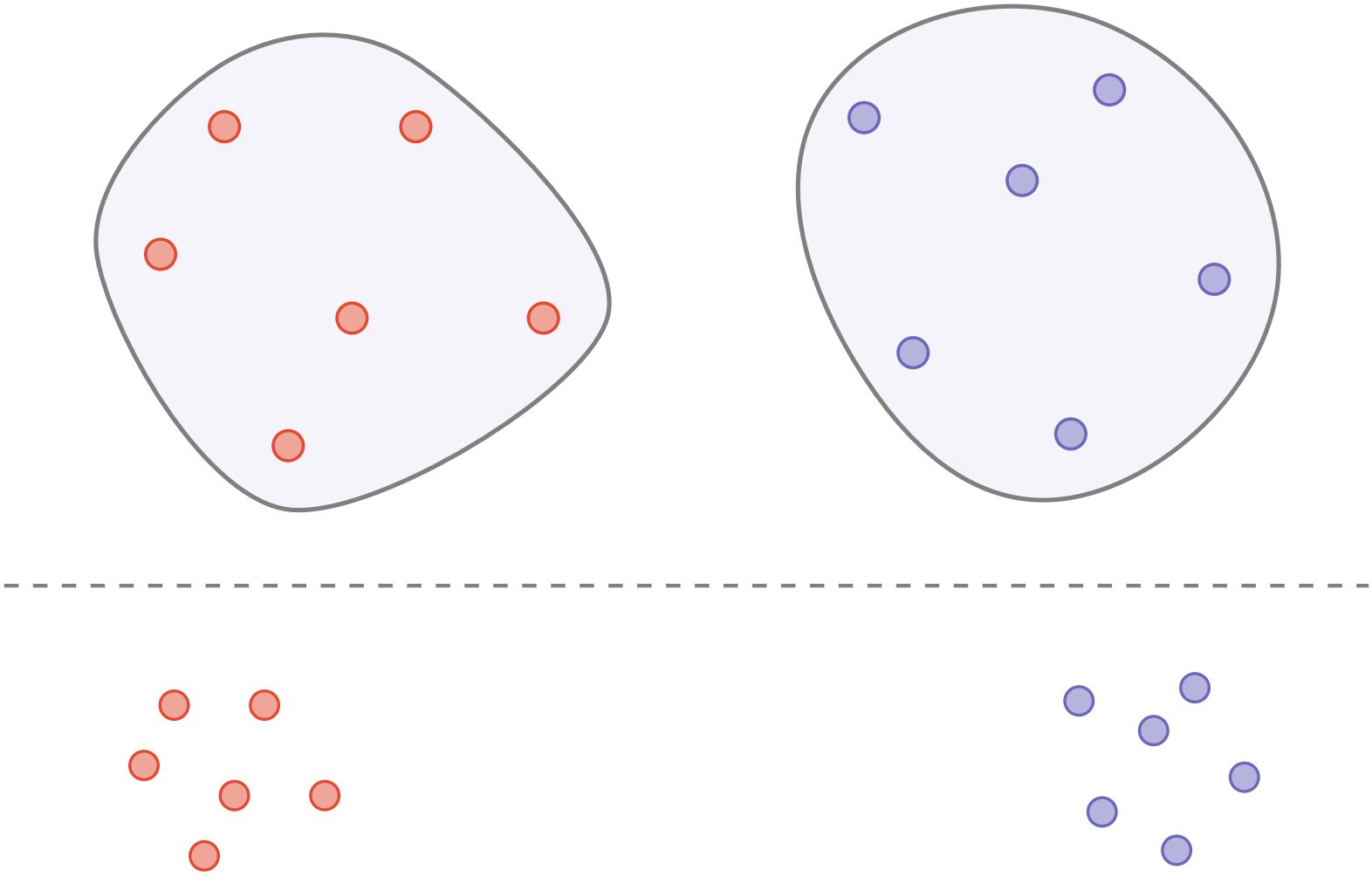
Idea: establish rules (axioms) for the behavior of clustering methods.

In this view, a “correct” clustering is one produced by an algorithm which obeys the axioms.

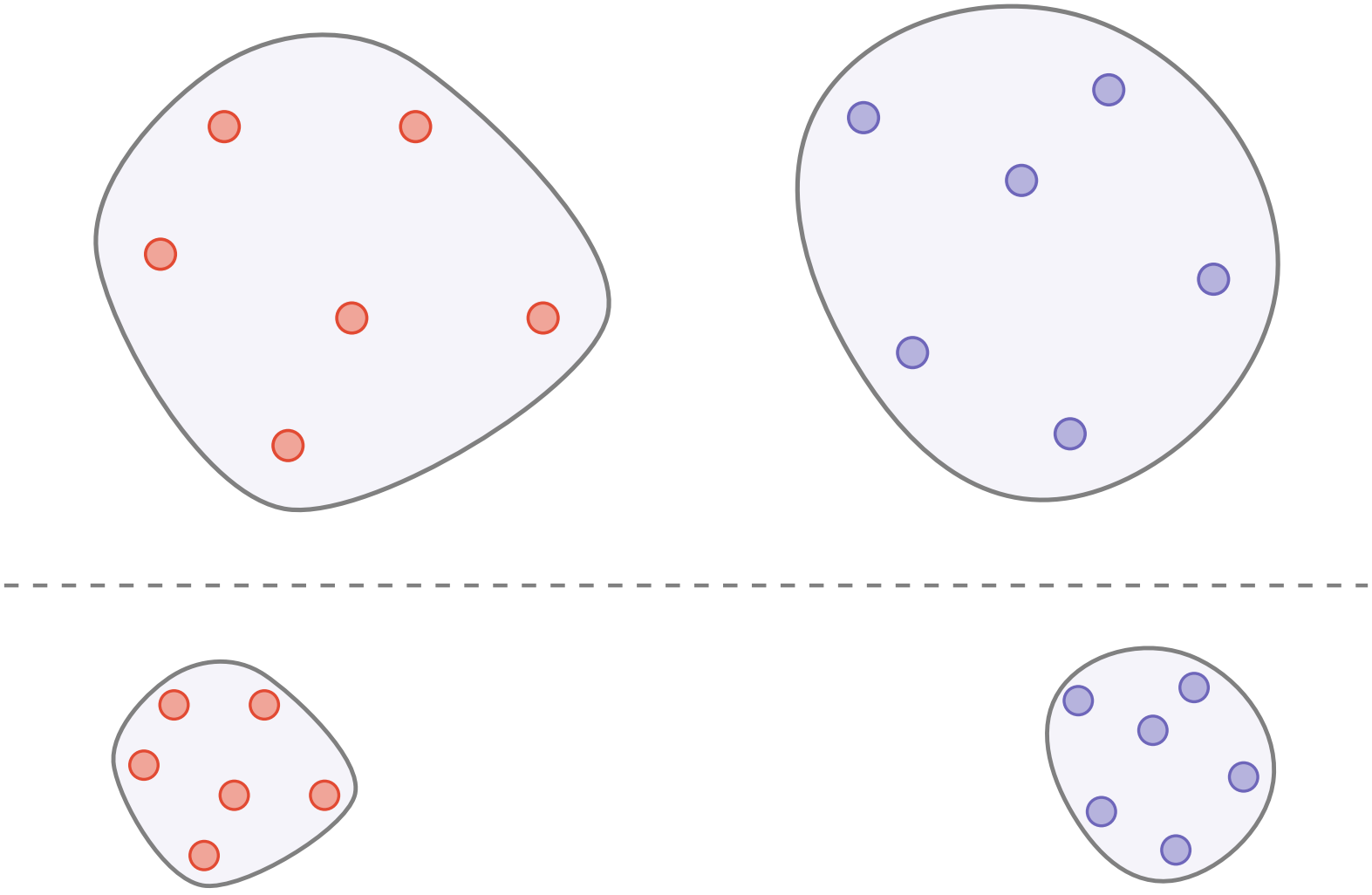
J. Kleinberg (2003) studied three axioms:

1. scale-invariance
2. consistency
3. richness

J. Kleinberg's consistency axiom



J. Kleinberg's consistency axiom



Three natural clustering axioms:

1. scale-invariance
2. consistency
3. richness

Three natural clustering axioms:

1. scale-invariance
2. consistency
3. richness

J. Kleinberg (2003): a method satisfying all three is impossible.

Three natural clustering axioms:

1. scale-invariance
2. consistency
3. richness

J. Kleinberg (2003): a method satisfying all three is impossible.

For positive results, see:

- (Ben-Davis & Ackerman, 2009)
- (Carlsson & Mémoli, 2010)

Advantages of axiomatic approach:

- Can be useful when stability is important.
- Or when certain invariances must hold.
- Better understand all methods by which axioms they fail to satisfy.

Disadvantages:

- Impossibility result.
- Unclear how axioms relate to data-generation process.

Context

Approaches to “correct” clustering.

- Optimization
- Axiomatic
- Statistical

Part I: Density

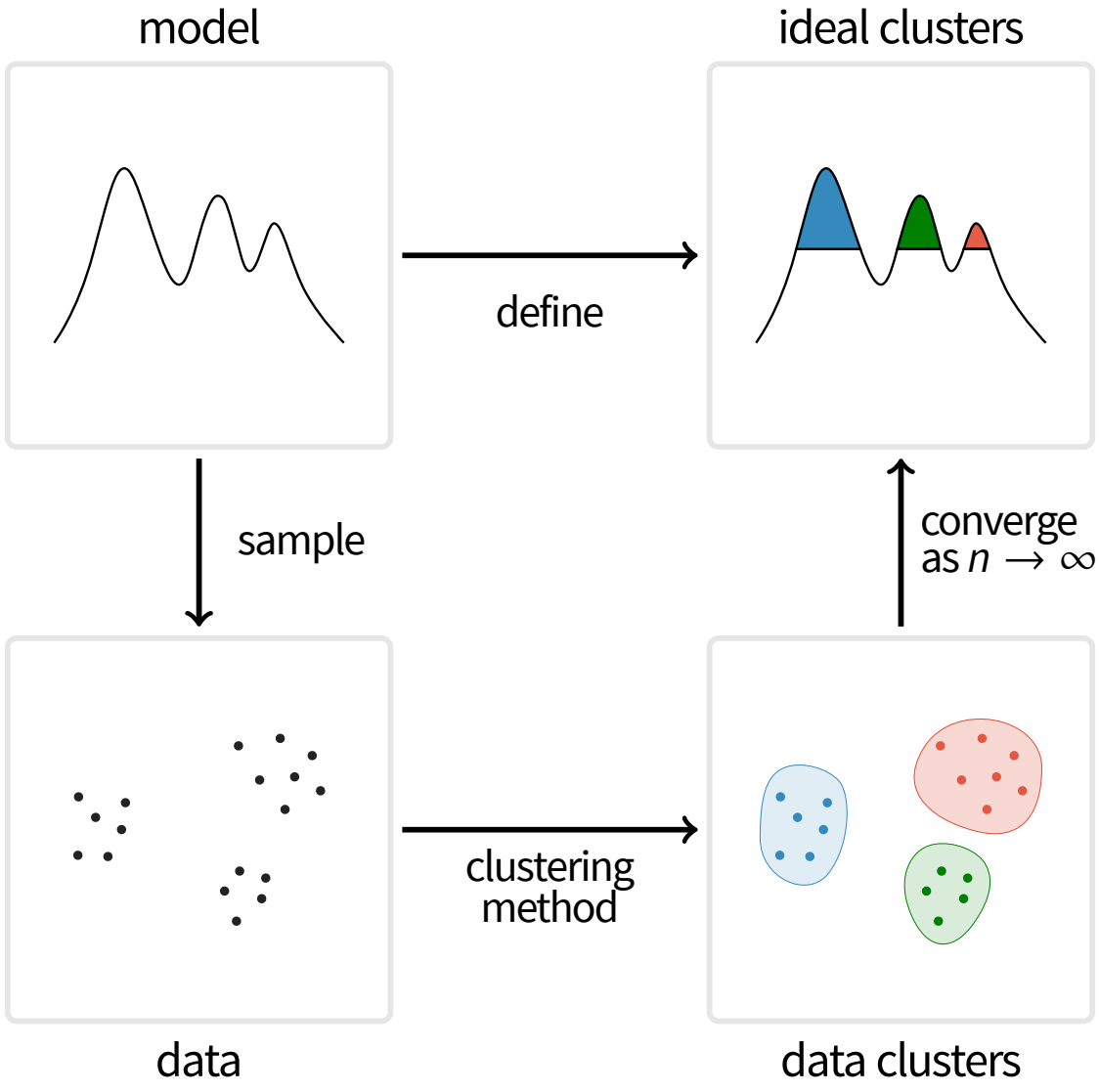
Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

A statistical approach to clustering.



Advantages of statistical theories:

- Clusters have explicit interpretation w.r.t. the model.
- Can talk about statistical significance of clusters.
- Encode domain knowledge within model.
- (Often) tractable.

(Potential) disadvantages:

- Model must be rich enough to describe reality.
- But it can be difficult to develop theory in a rich model.

Context

Approaches to “correct” clustering.

Part I: Density

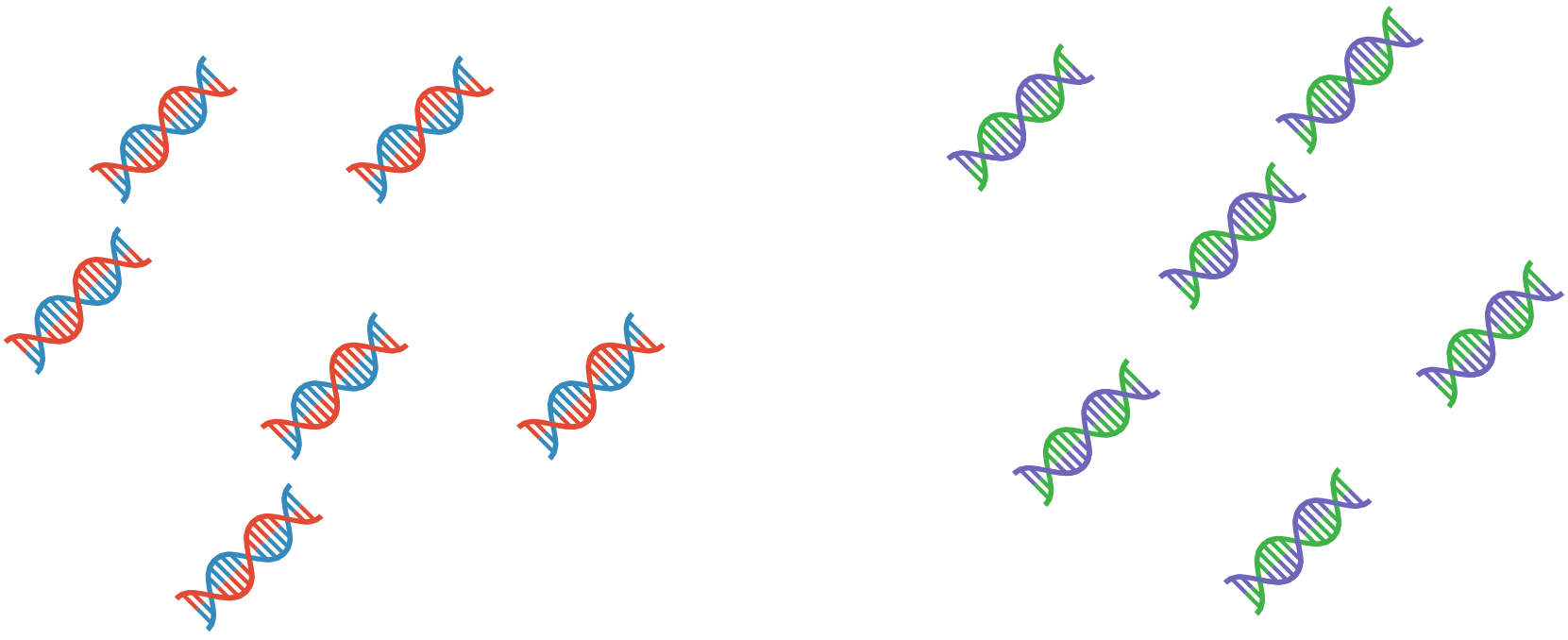
Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

- Existing theory of Hartigan consistency
- The weakness of Hartigan’s notion
- Our strong notion of convergence

Part II: Graphon

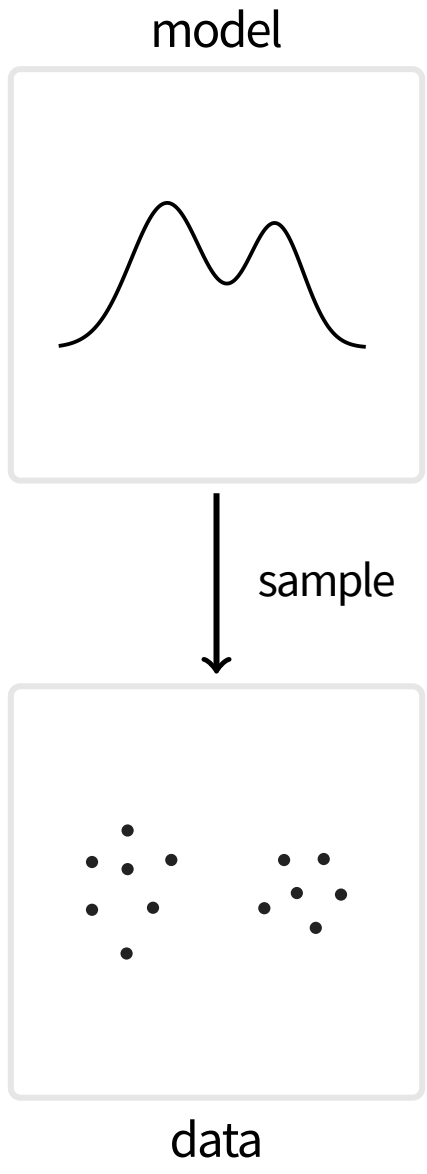
Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

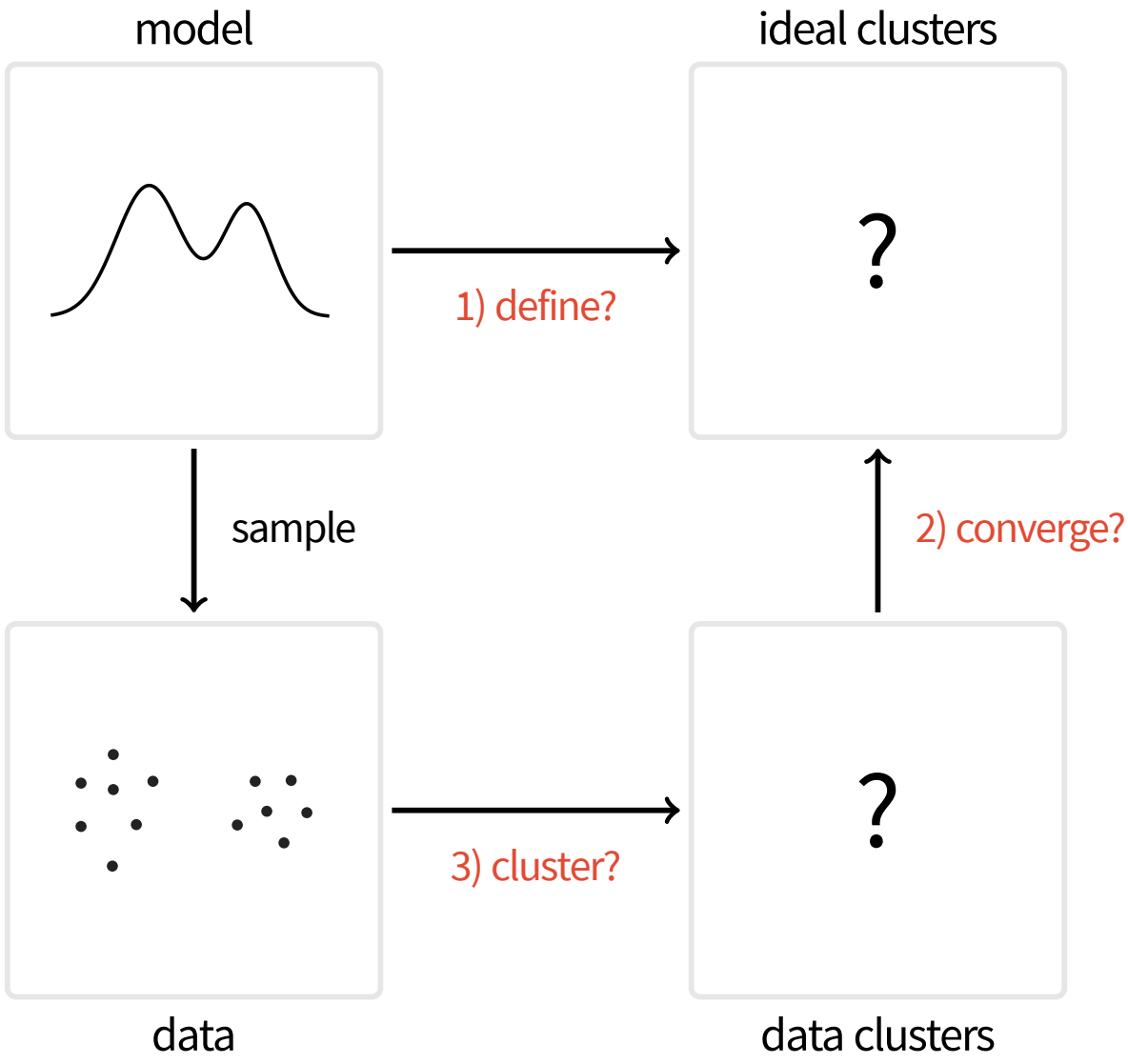


Identify subtypes of cancer via clustering of DNA microarrays.

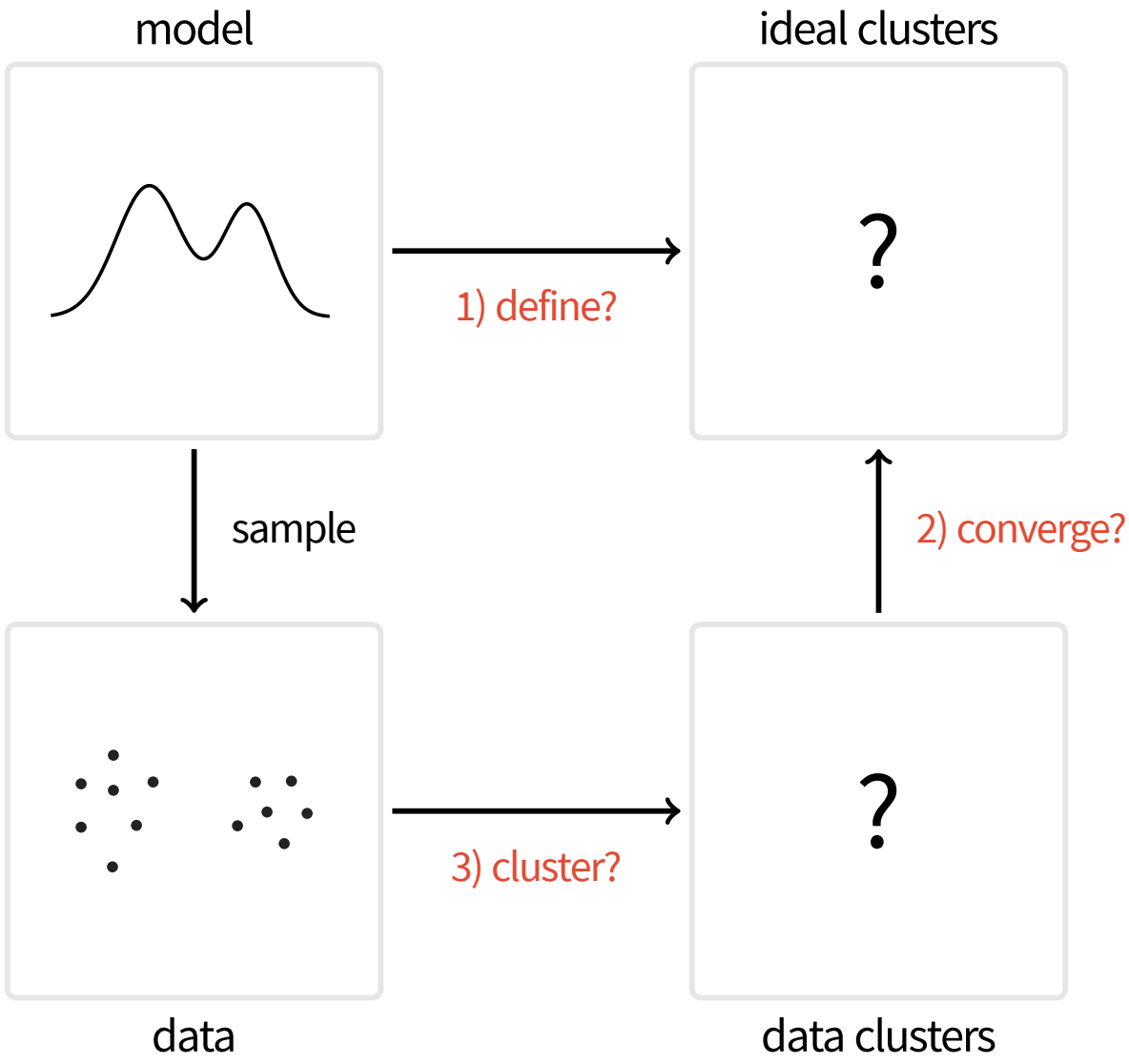
Clustering in the density model.



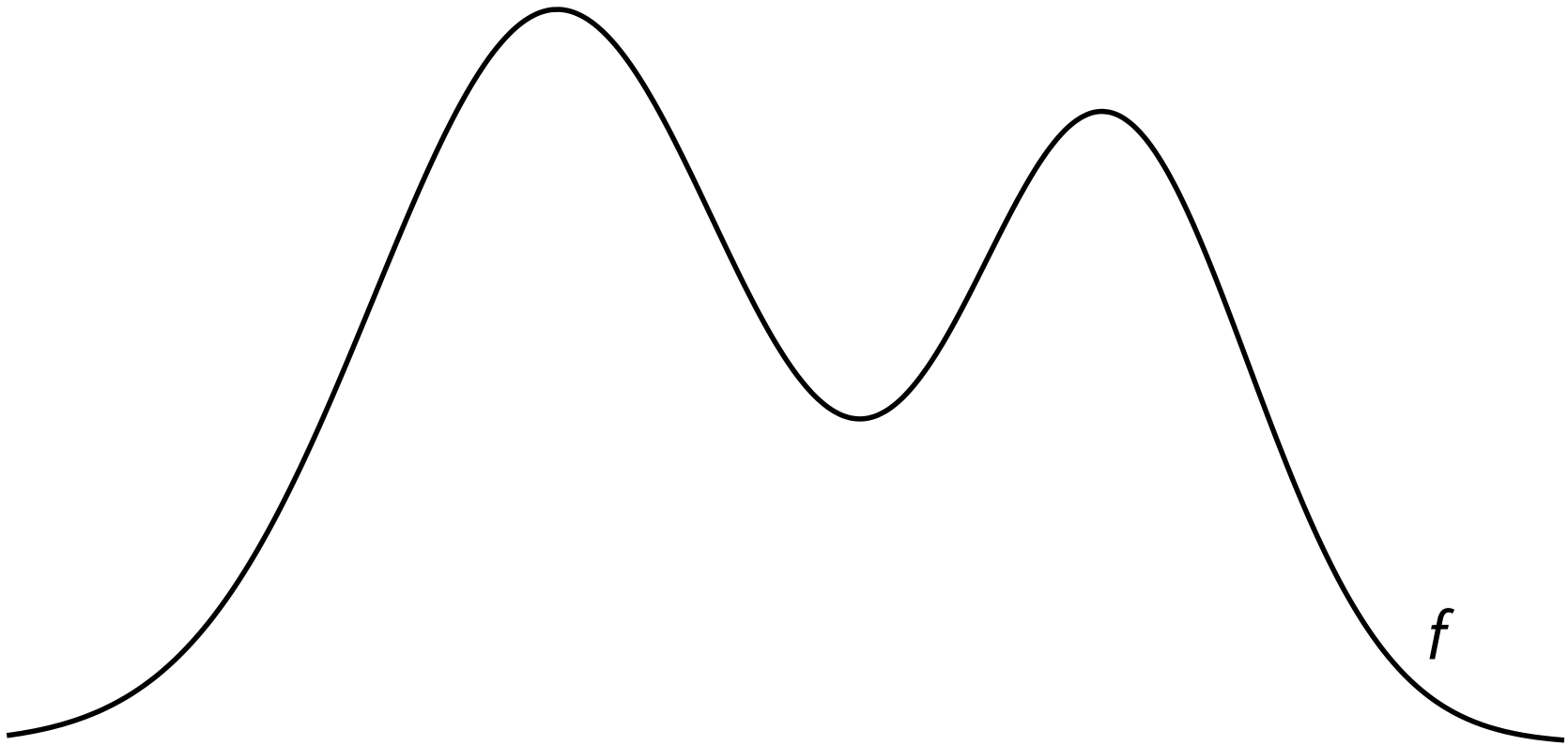
Clustering in the density model.



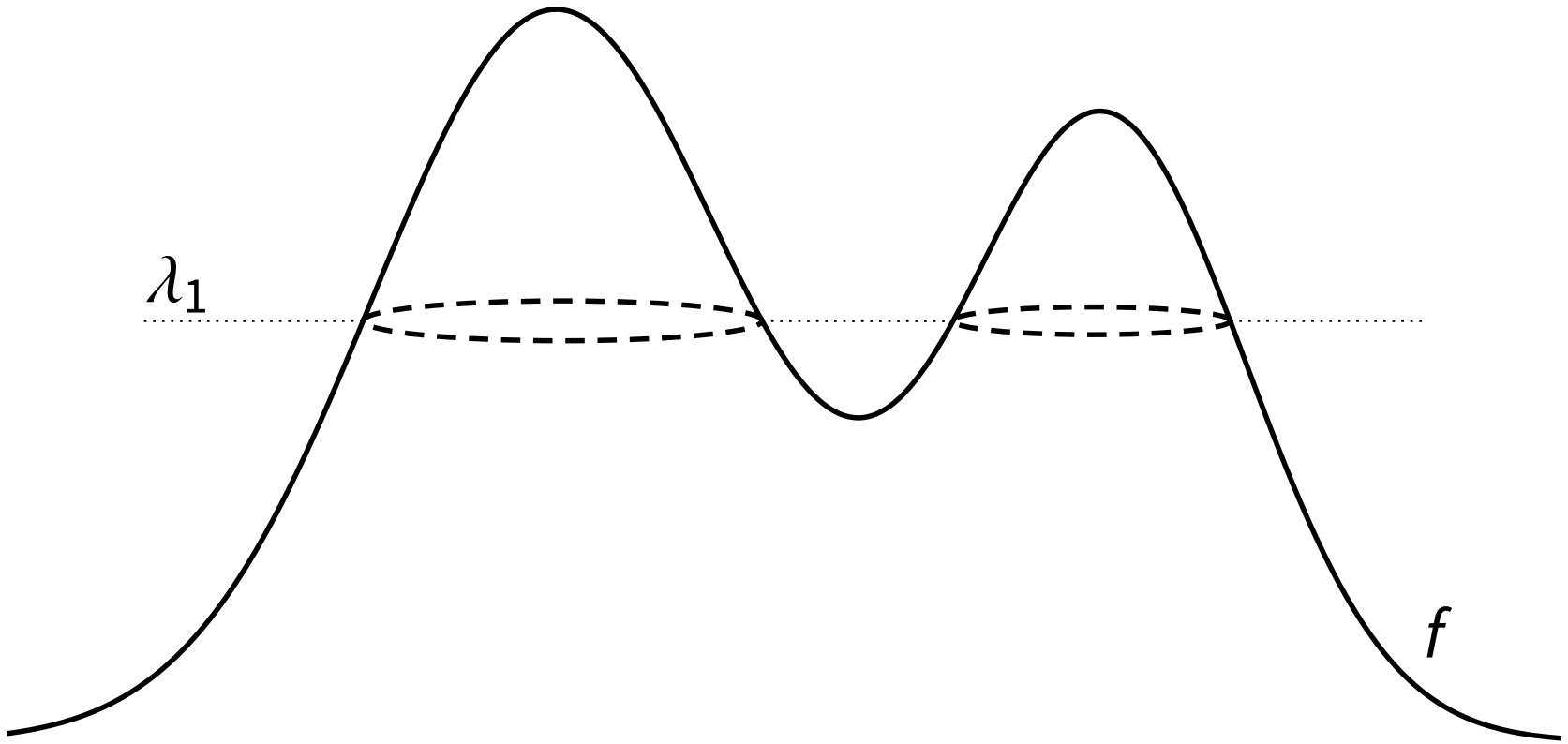
Step 1: What are the clusters of the density?



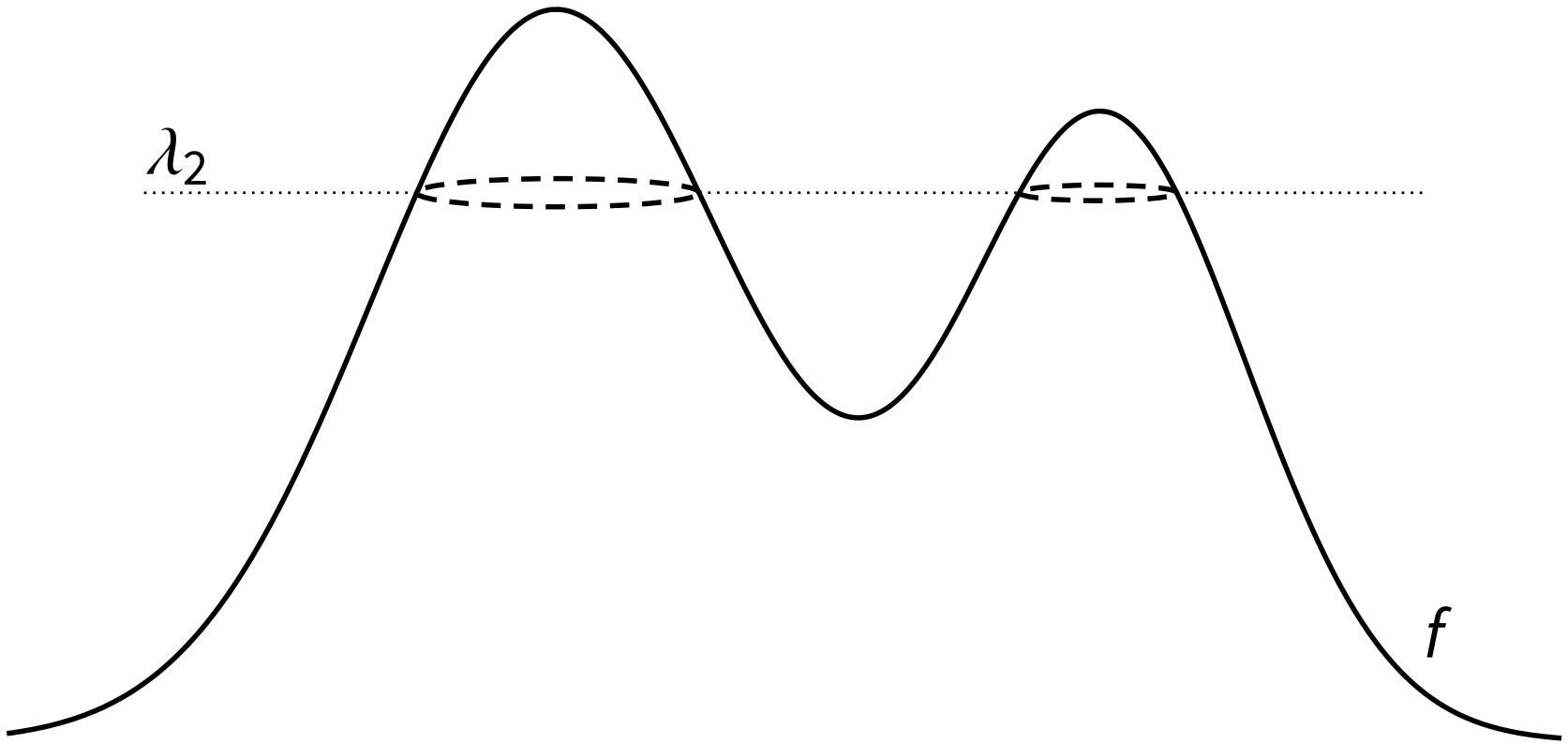
Intuition: a cluster is a region of high probability.



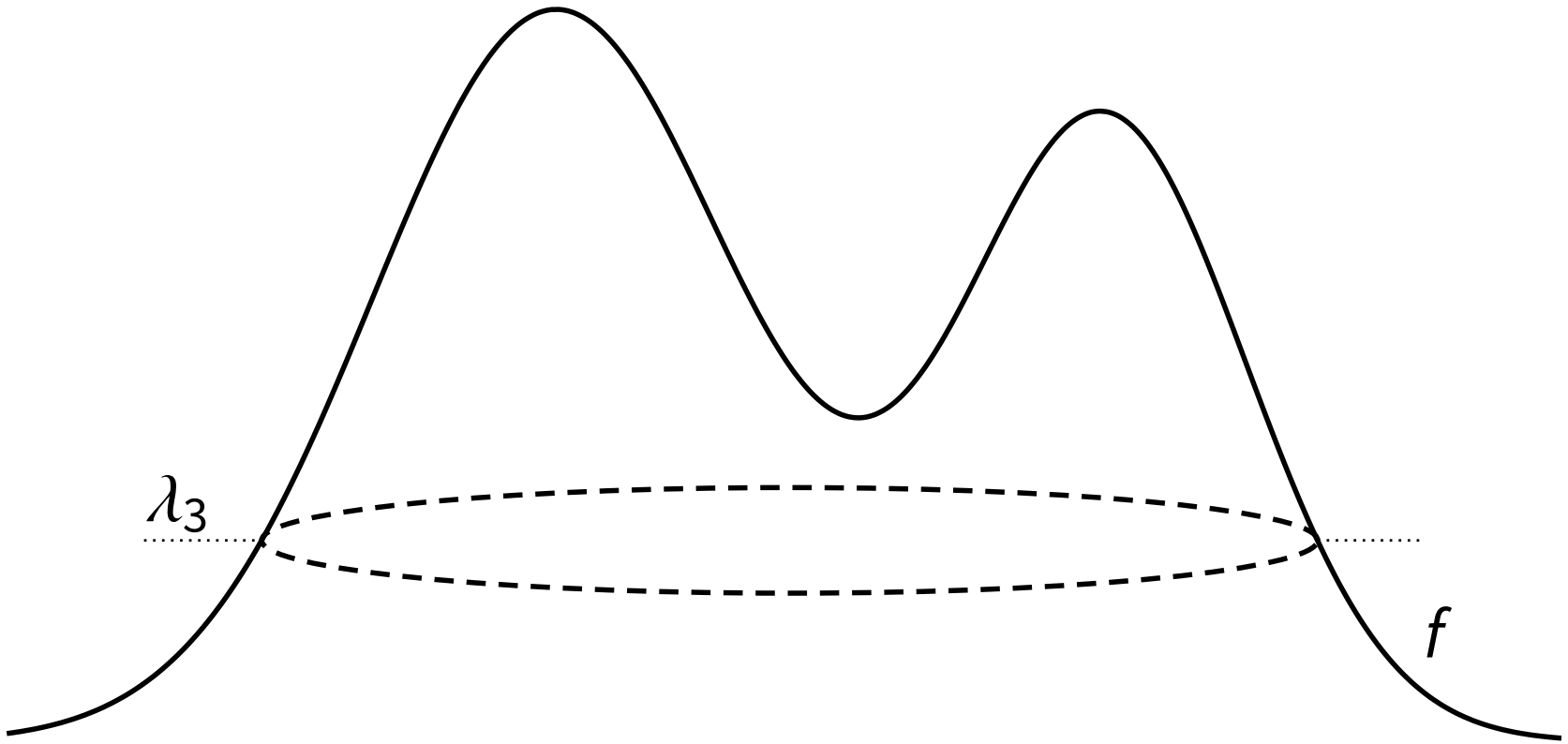
Connected component of $\{f \geq \lambda_1\}$?



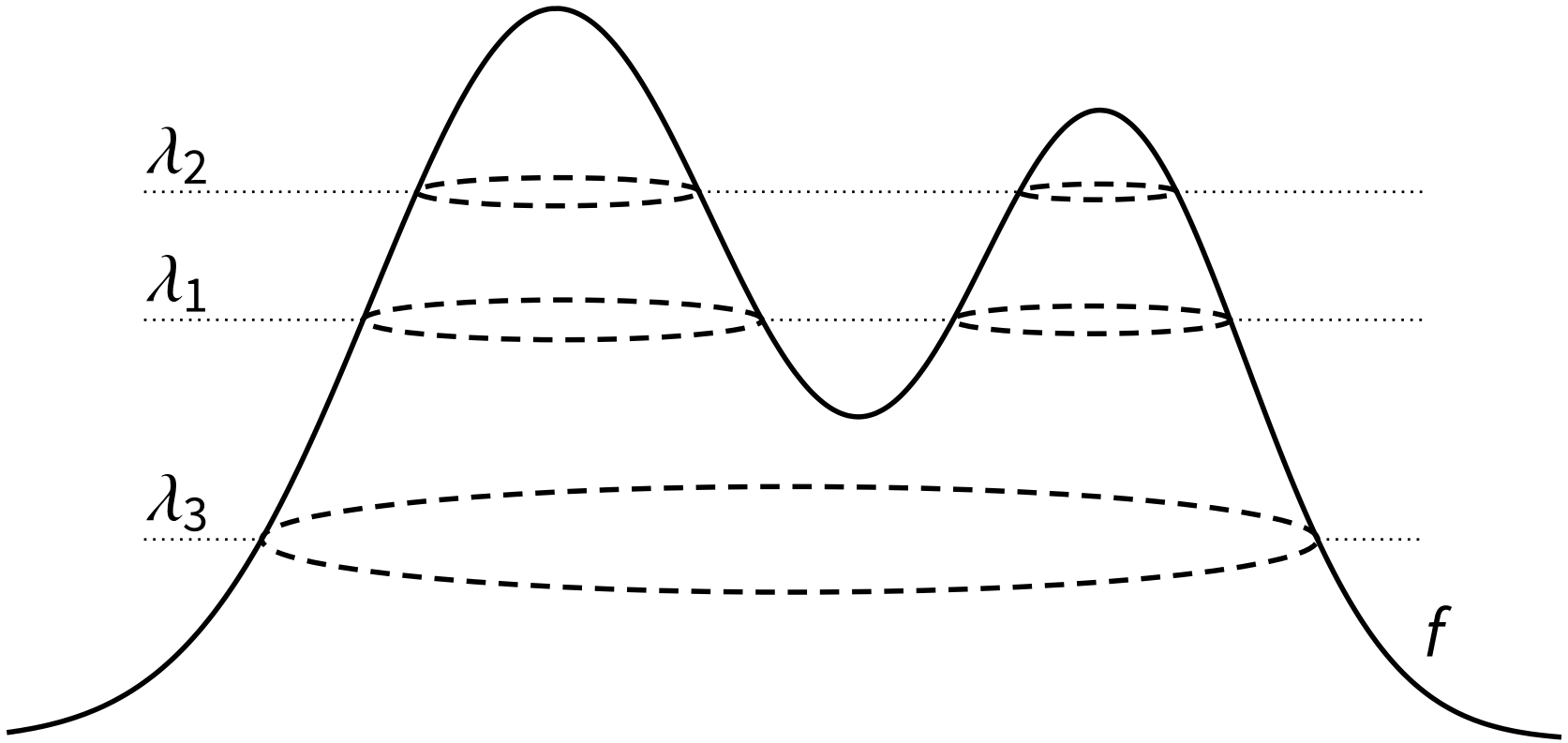
Connected component of $\{f \geq \lambda_2\}$?



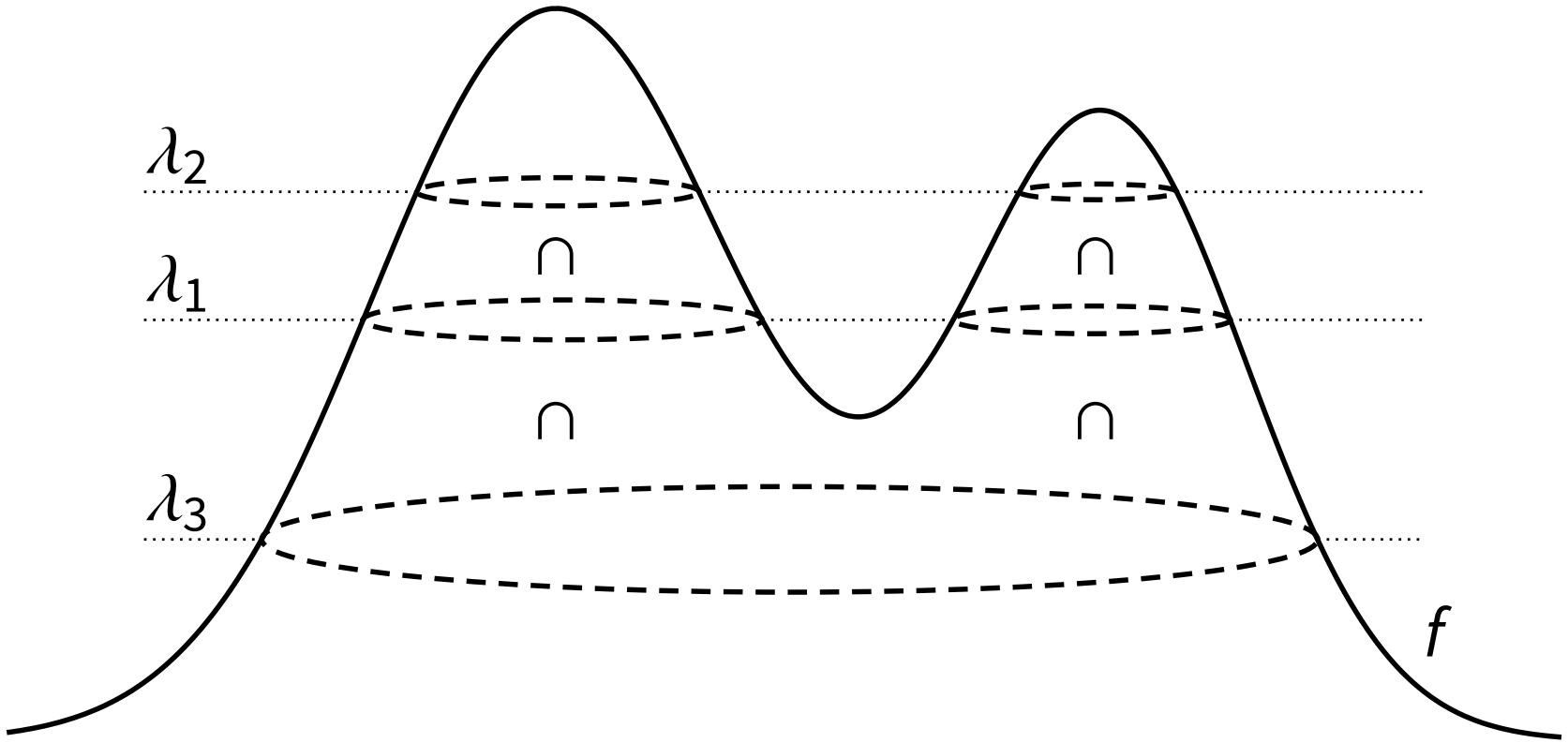
Connected component of $\{f \geq \lambda_3\}$?



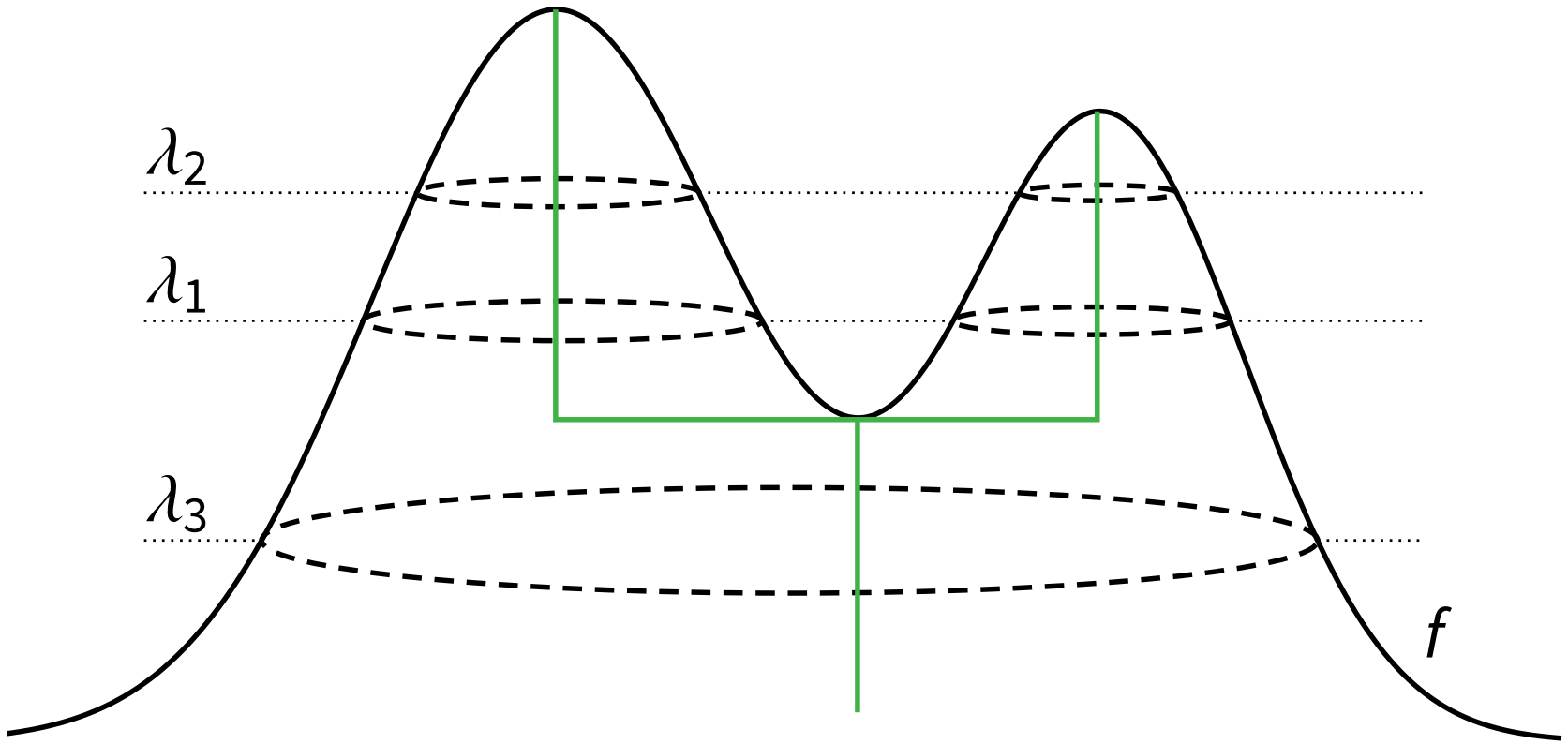
Hartigan (1981): A cluster is a connected component of $\{f \geq \lambda\}$, for any $\lambda > 0$.



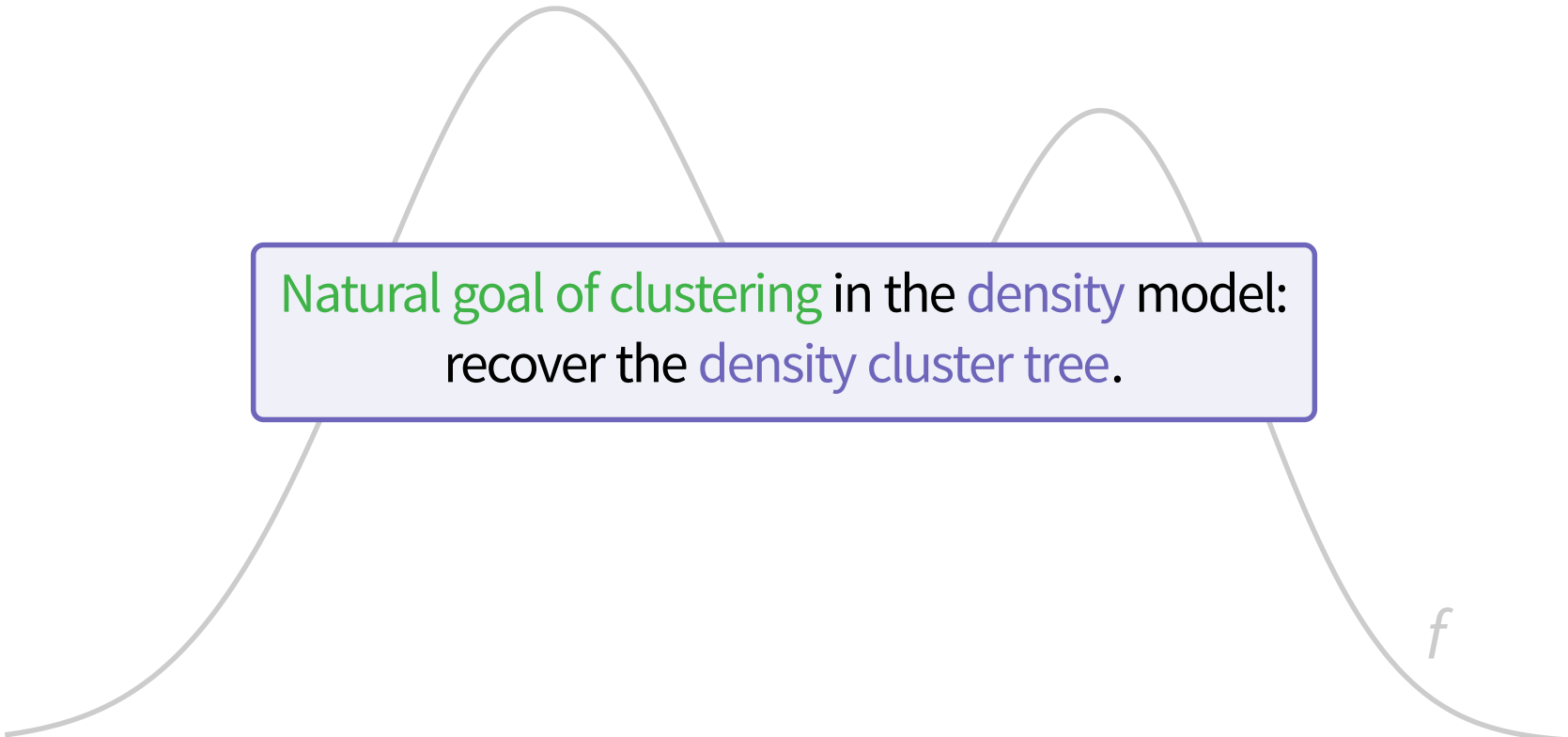
Hartigan (1981): A cluster is a connected component of $\{f \geq \lambda\}$, for any $\lambda > 0$.



Clusters form the density cluster tree of f.

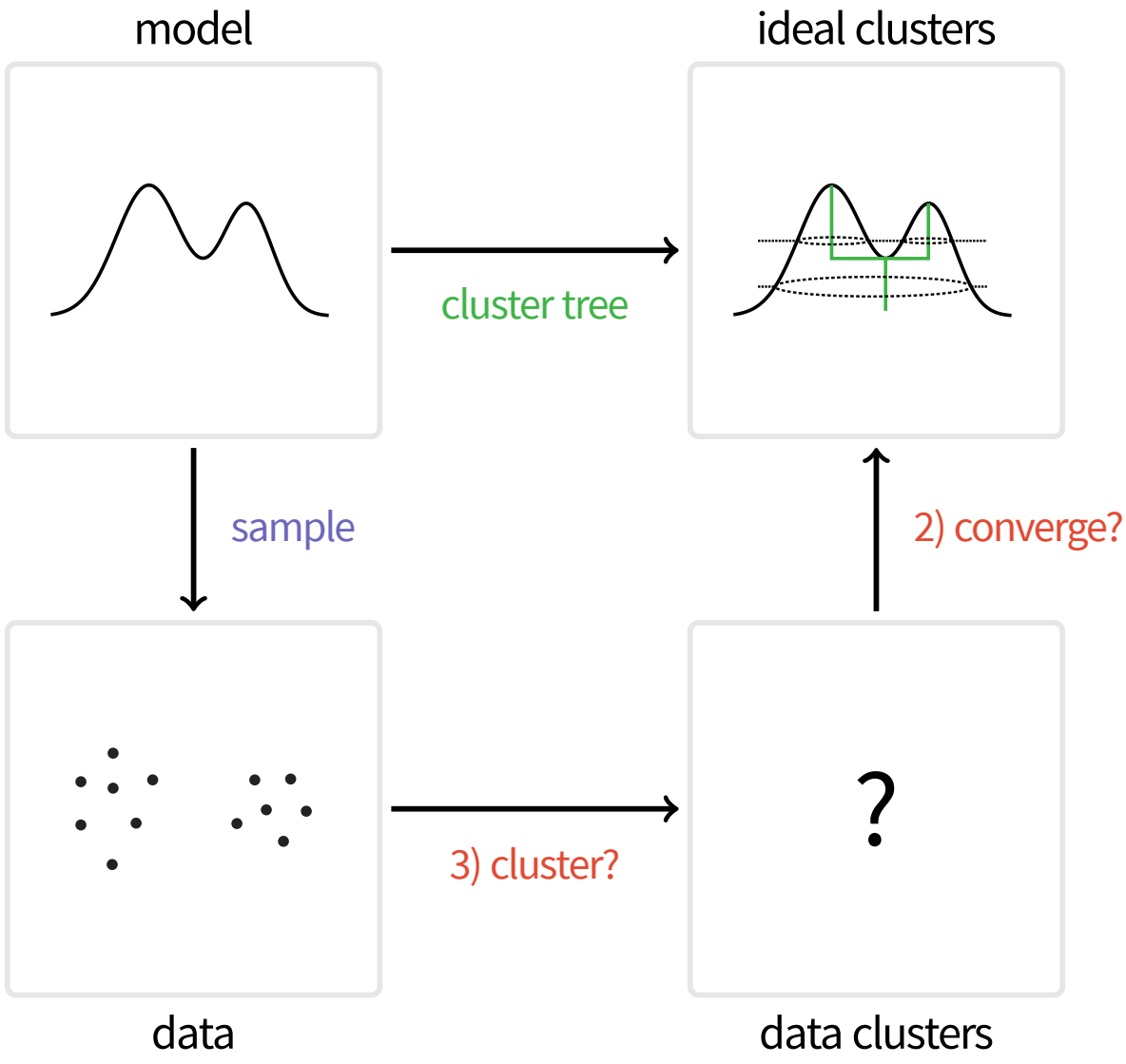


Clusters form the density cluster tree of f.

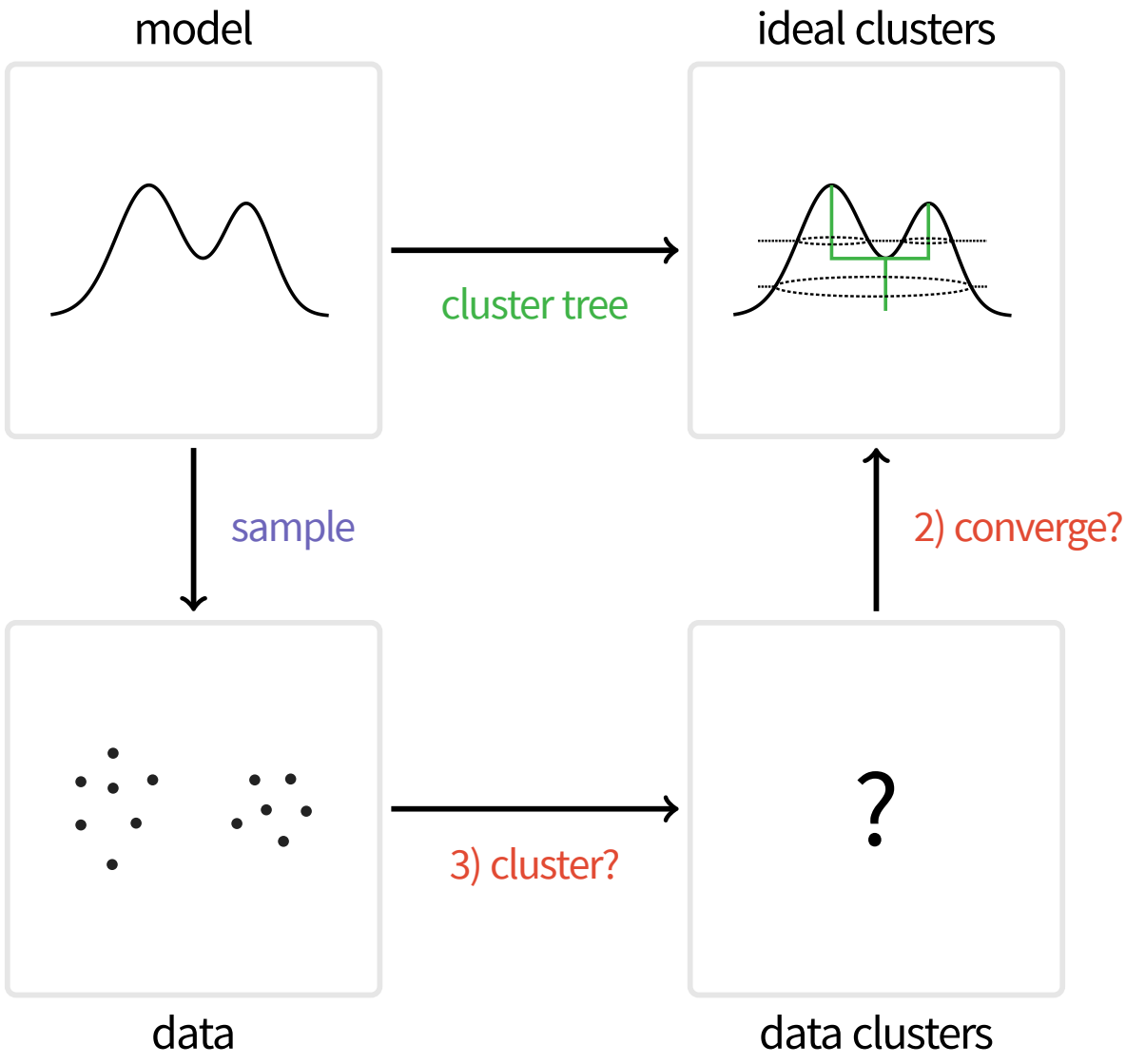


Natural goal of clustering in the density model:
recover the density cluster tree.

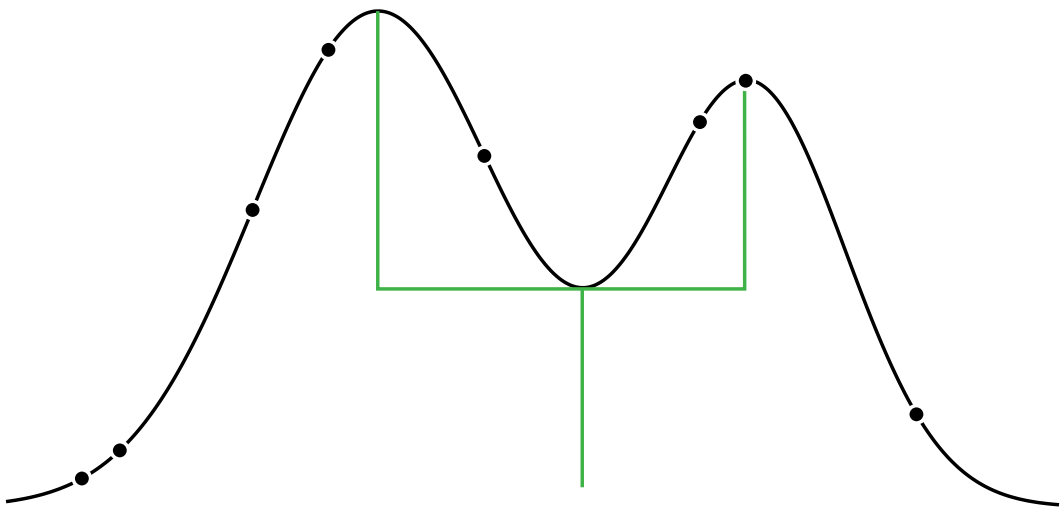
The density cluster tree is the ideal clustering.



Step 2: How do we define convergence to the density cluster tree?

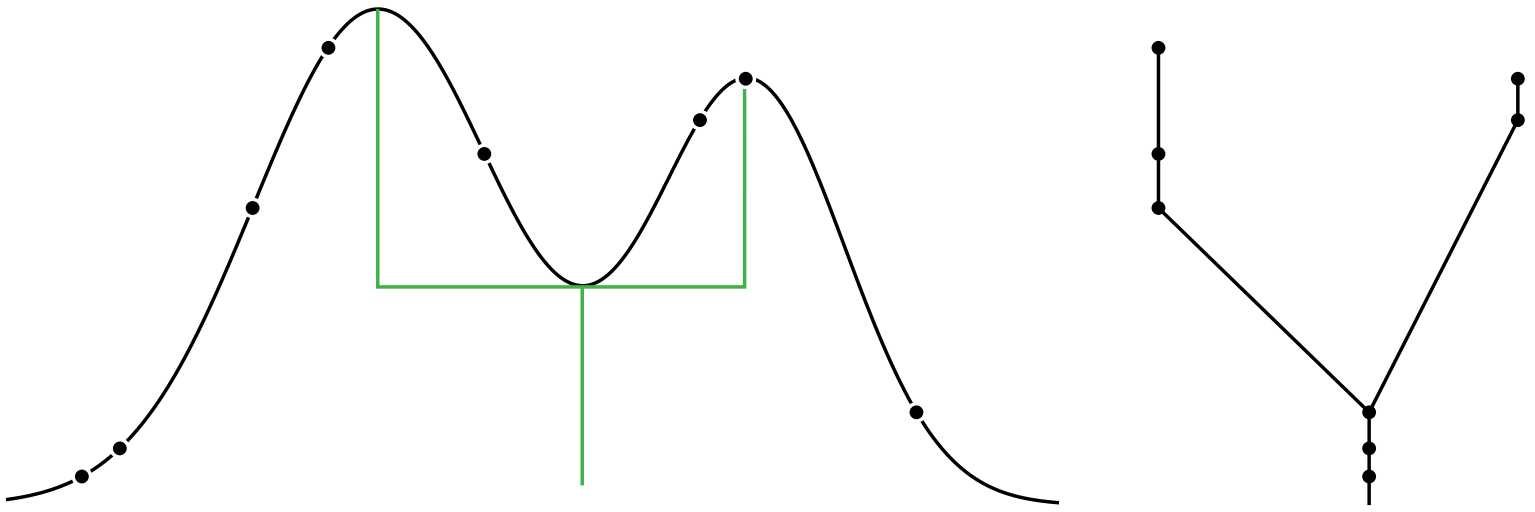


Need a formal notion of convergence to the density cluster tree.



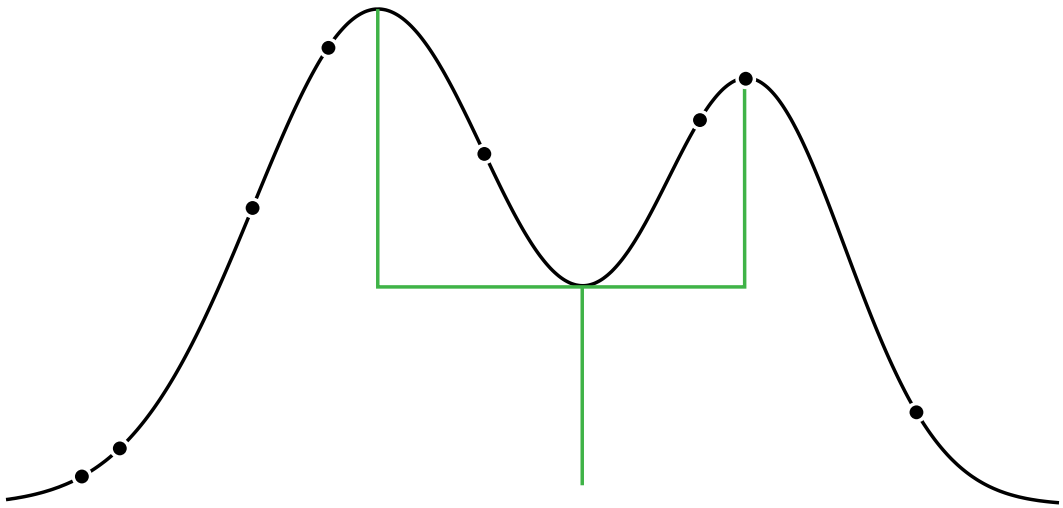
Sample n points from density.

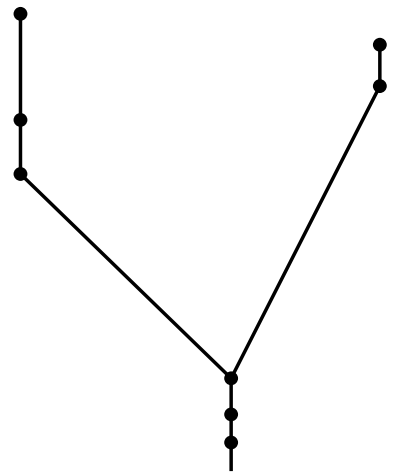
Need a formal notion of convergence to the density cluster tree.



Apply hierarchical clustering algorithm.

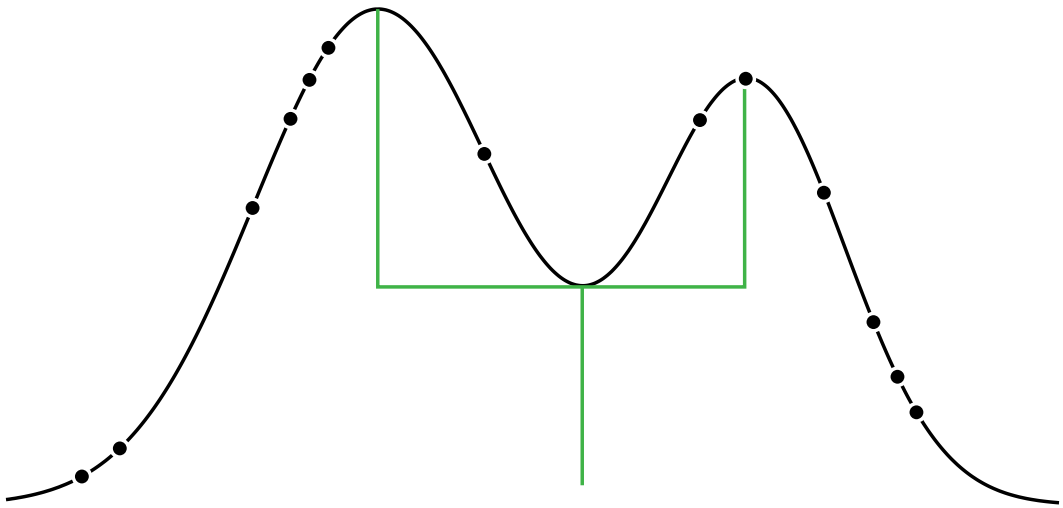
Need a formal notion of convergence to the density cluster tree.

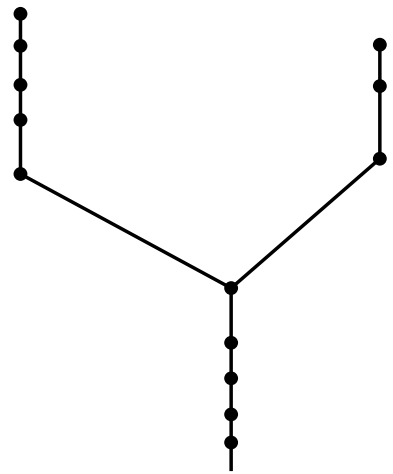




As $n \to \infty$...

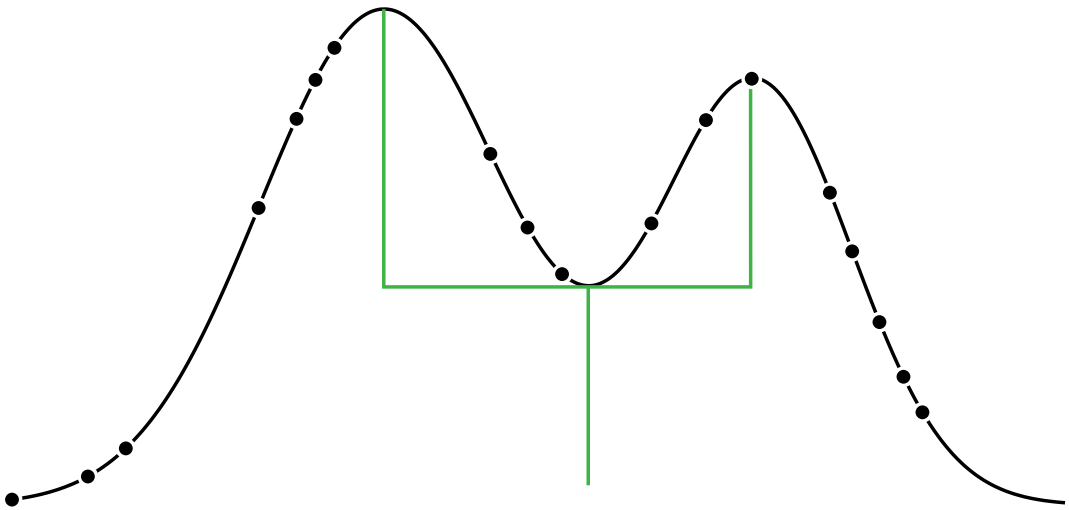
Need a formal notion of convergence to the density cluster tree.

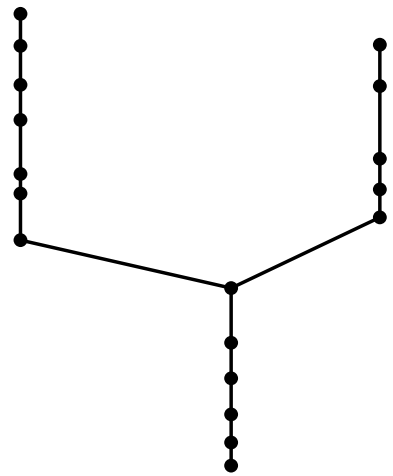




As $n \to \infty$...

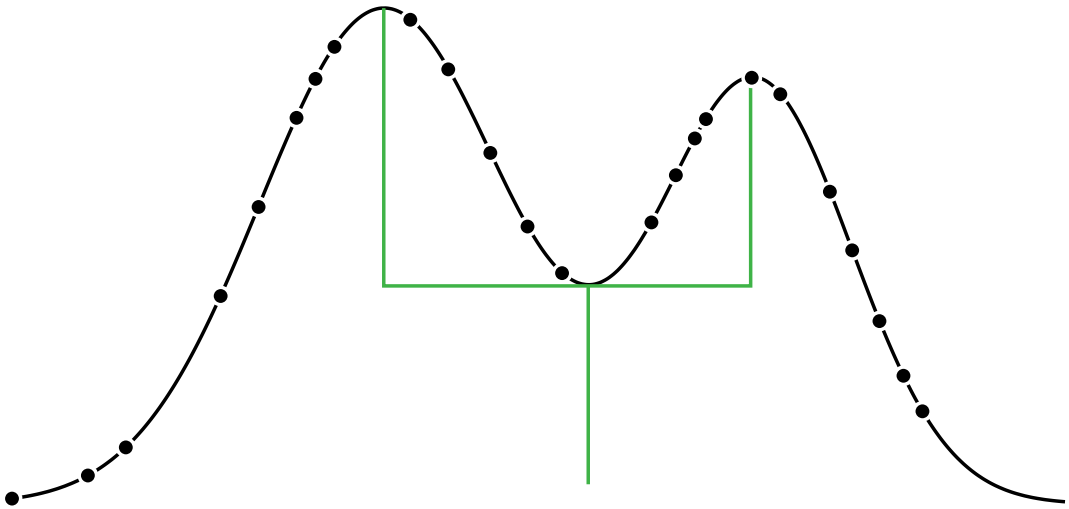
Need a formal notion of convergence to the density cluster tree.

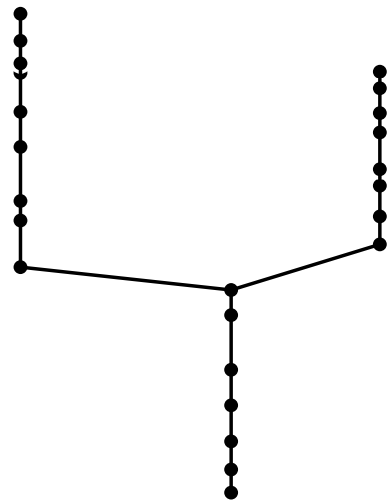




As $n \to \infty$...

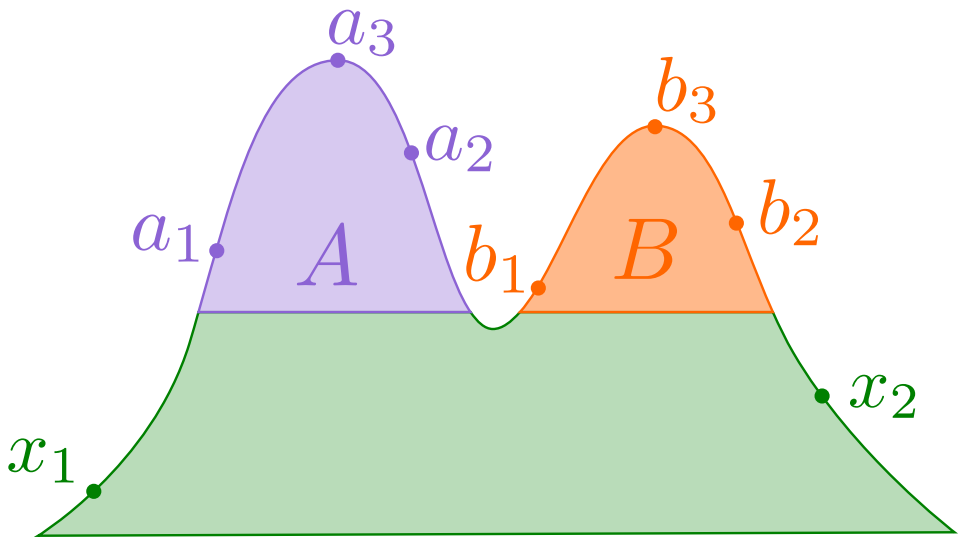
Need a formal notion of convergence to the density cluster tree.



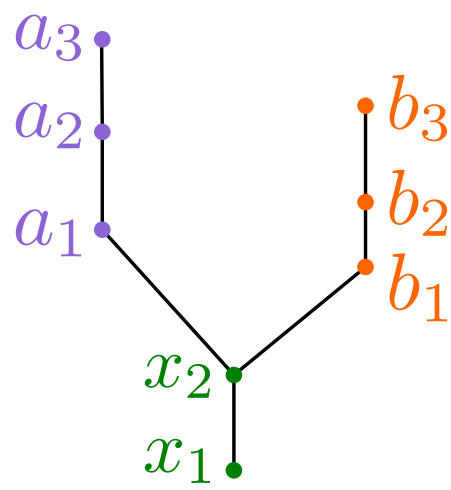


As $n \to \infty$...

Hartigan (1981): In the limit, clusters disjoint in true tree should be disjoint in empirical tree.

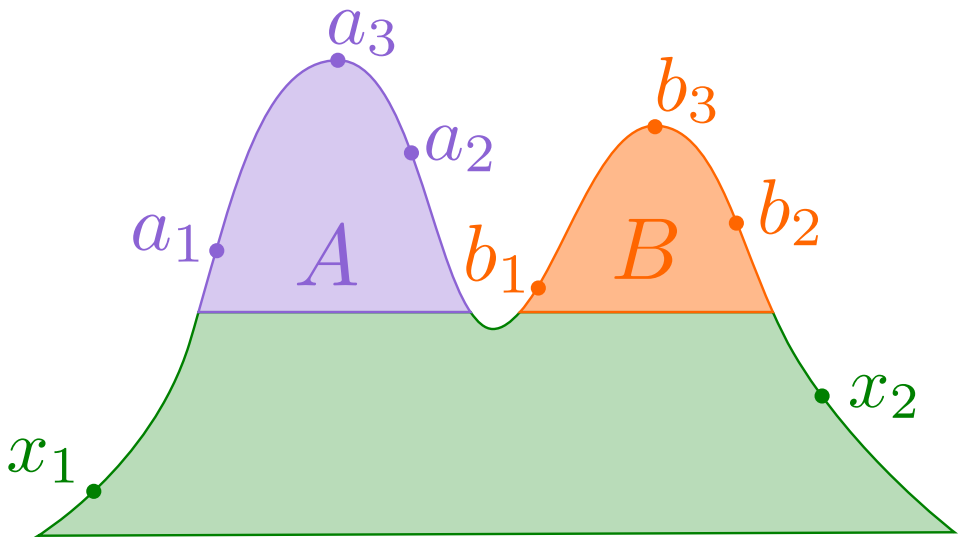


$\mathcal{C}_f$

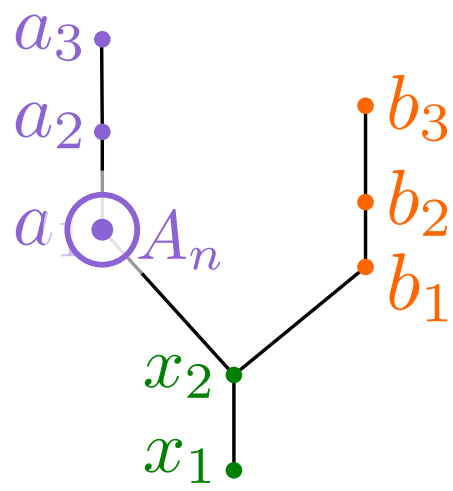


$\hat{\mathcal{C}}_{f,n}$

Find $A_n := $ the smallest empirical cluster containing $A \cap X_n$.

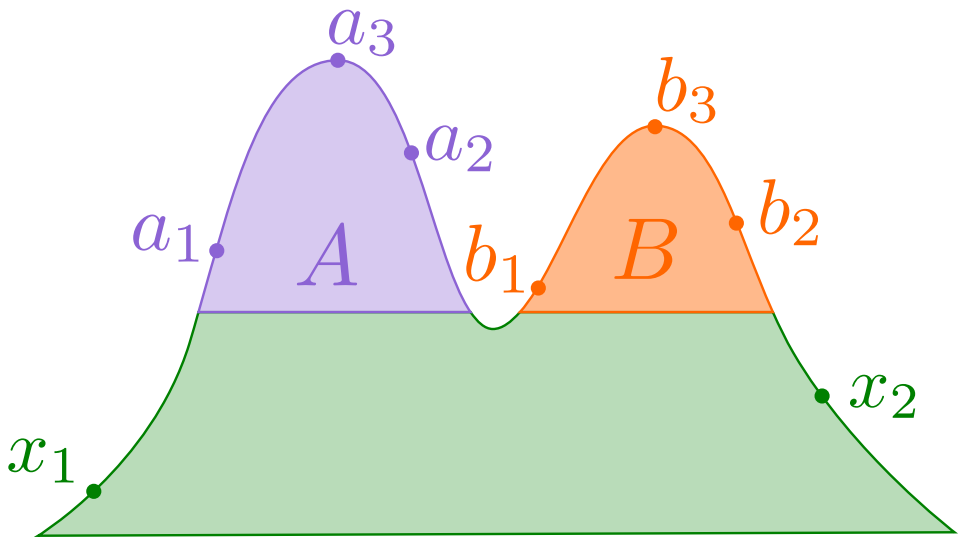


$\mathcal{C}_f$

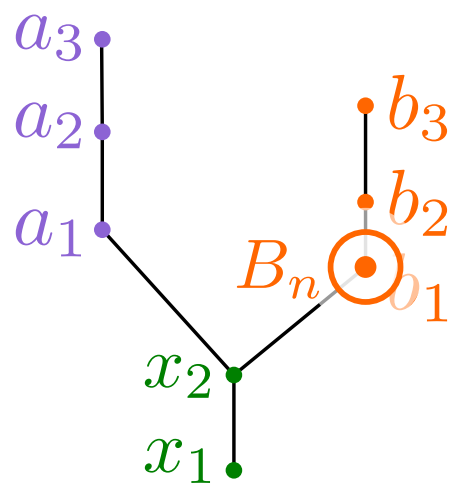


$\hat{\mathcal{C}}_{f,n}$

Find $B_n :=$ the smallest empirical cluster containing $B \cap X_n$.

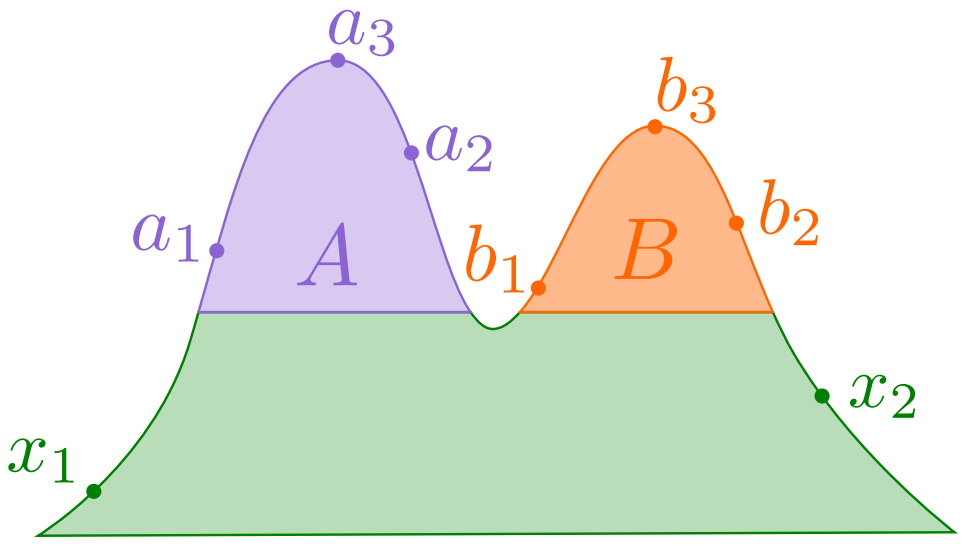


$\mathcal{C}_f$

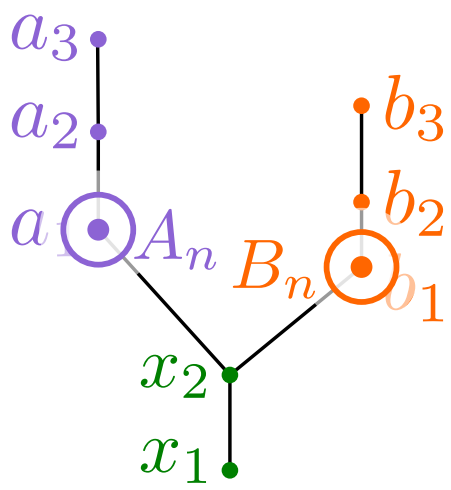


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

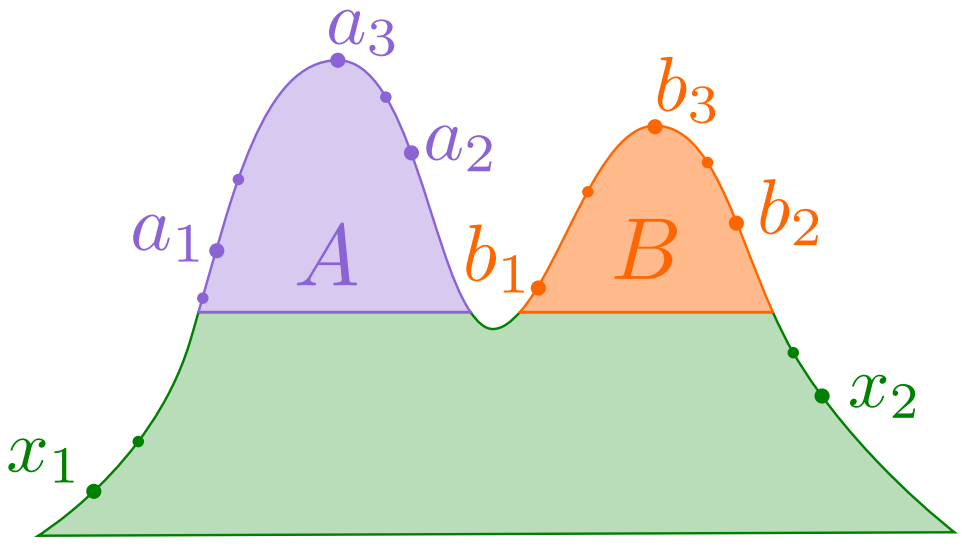


$\mathcal{C}_f$

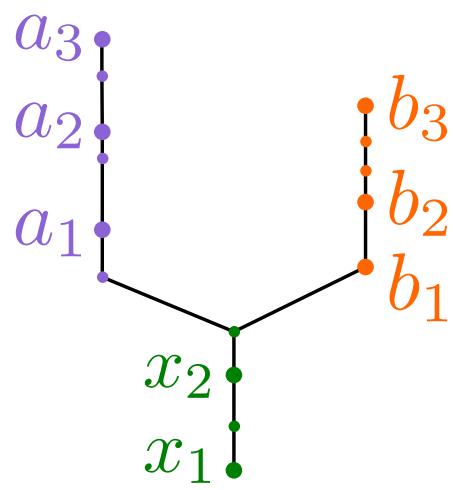


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

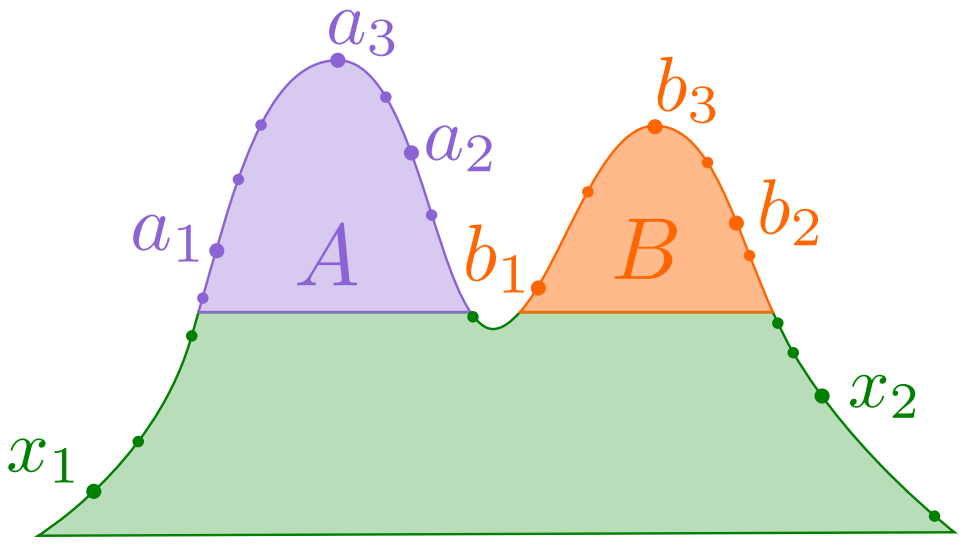


$\mathcal{C}_f$

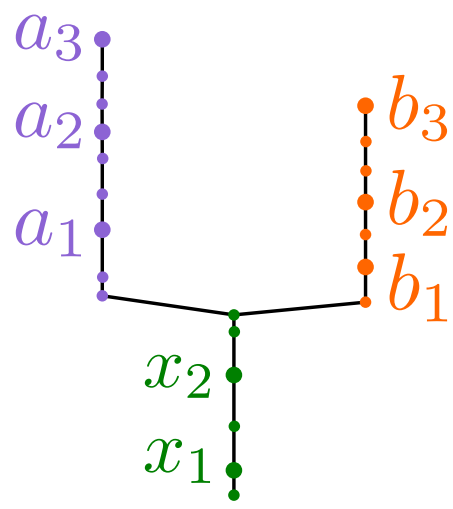


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

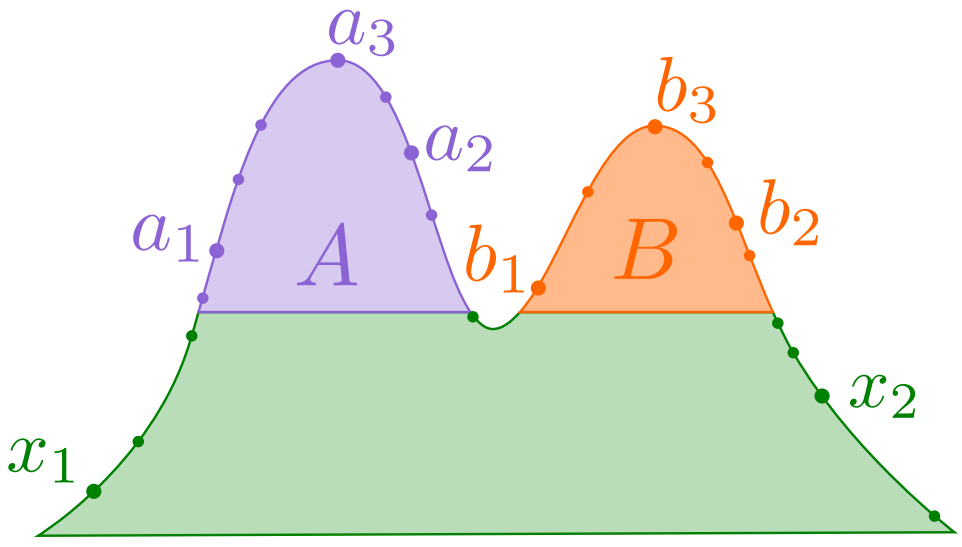


$\mathcal{C}_f$

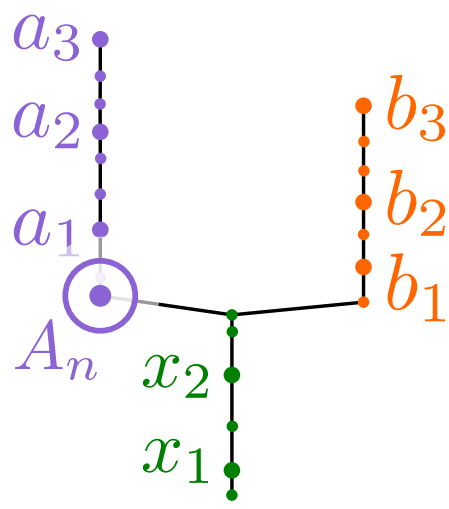


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

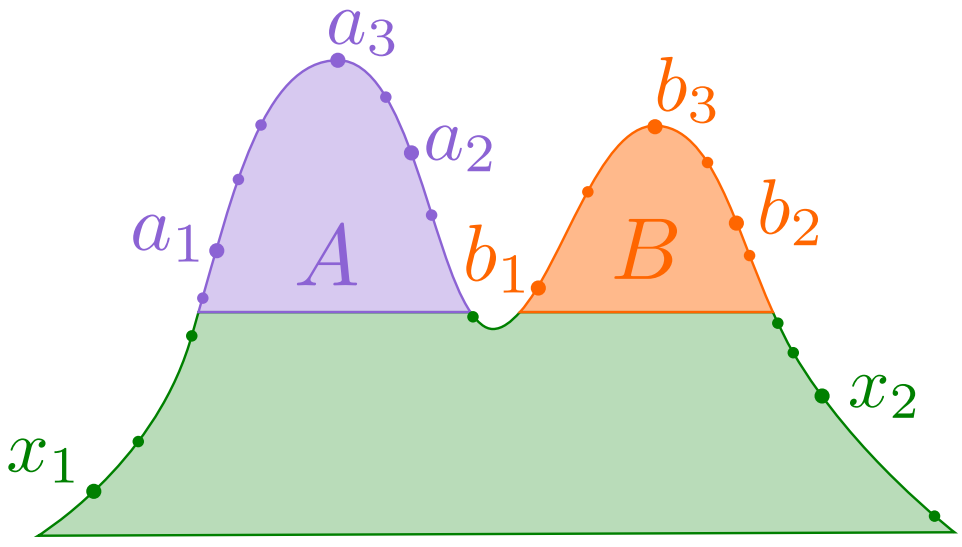


$\mathcal{C}_f$

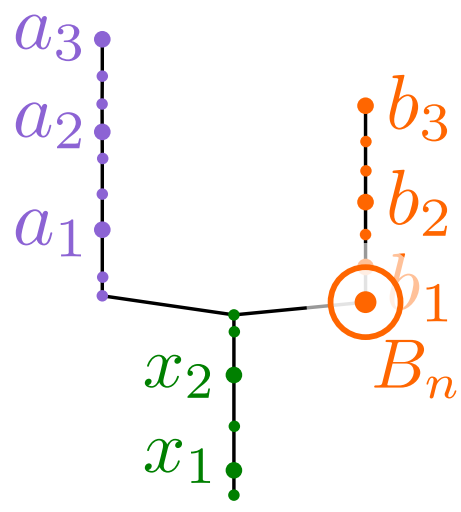


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

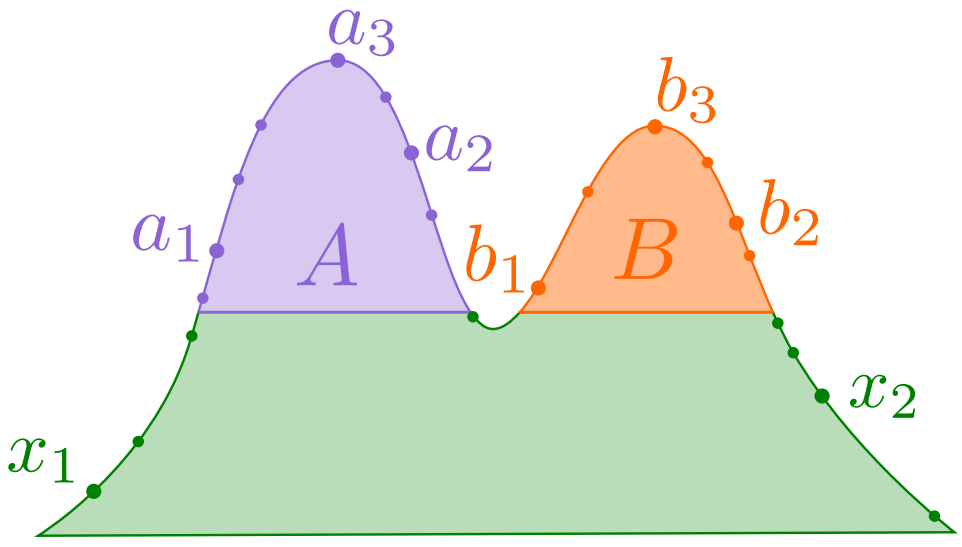


$\mathcal{C}_f$

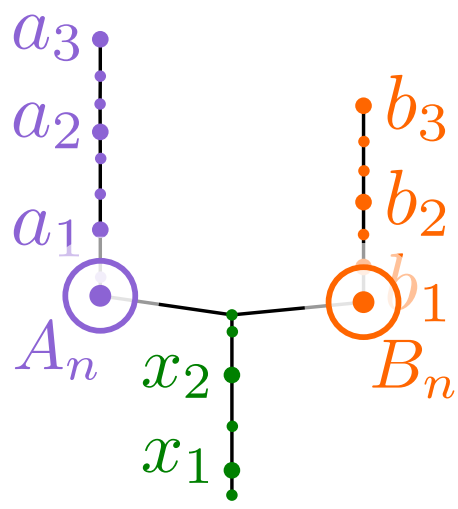


$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency: As $n \to \infty$, $\Pr(A_n \text{ is disjoint from } B_n) \to 1$.

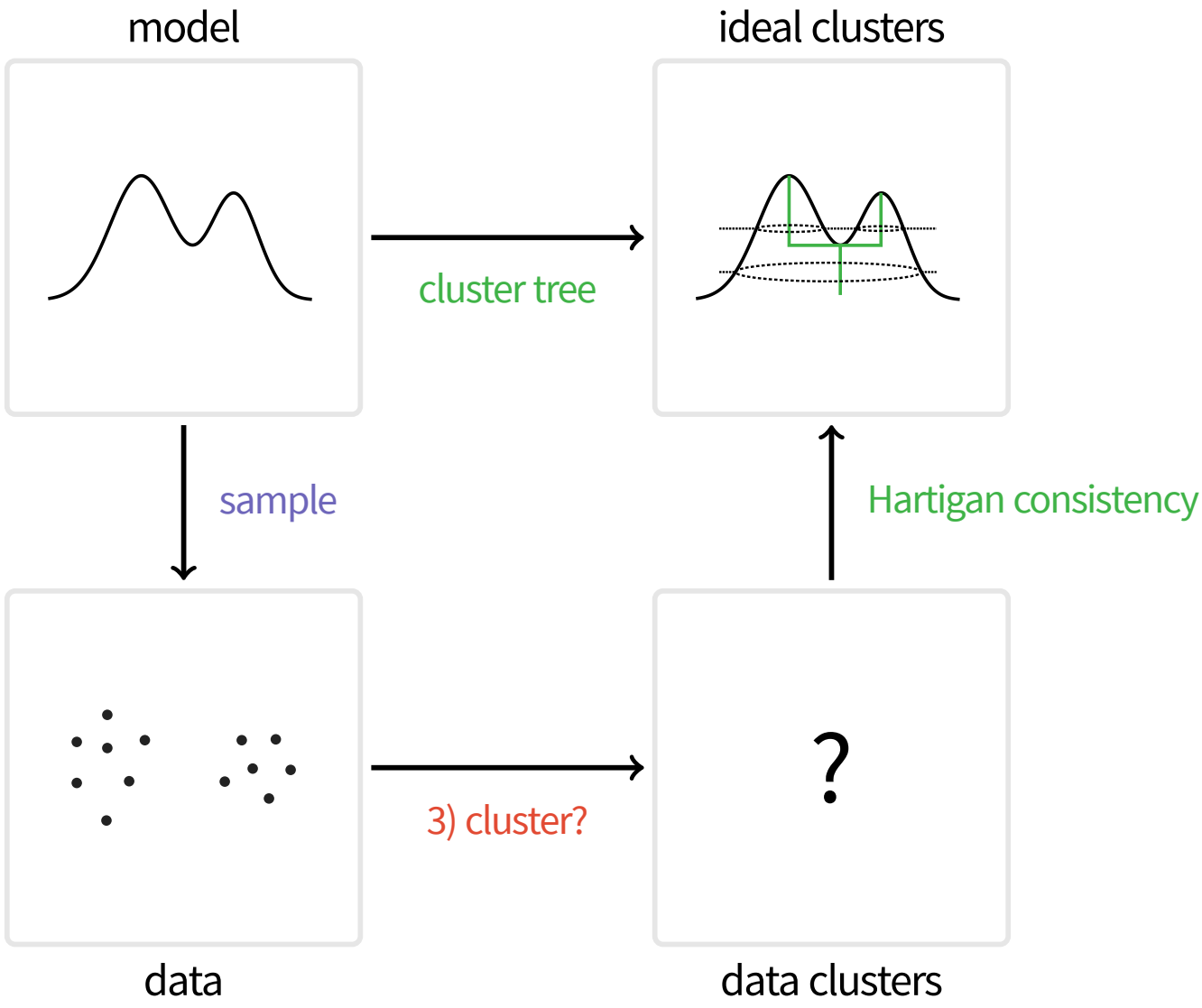


$\mathcal{C}_f$

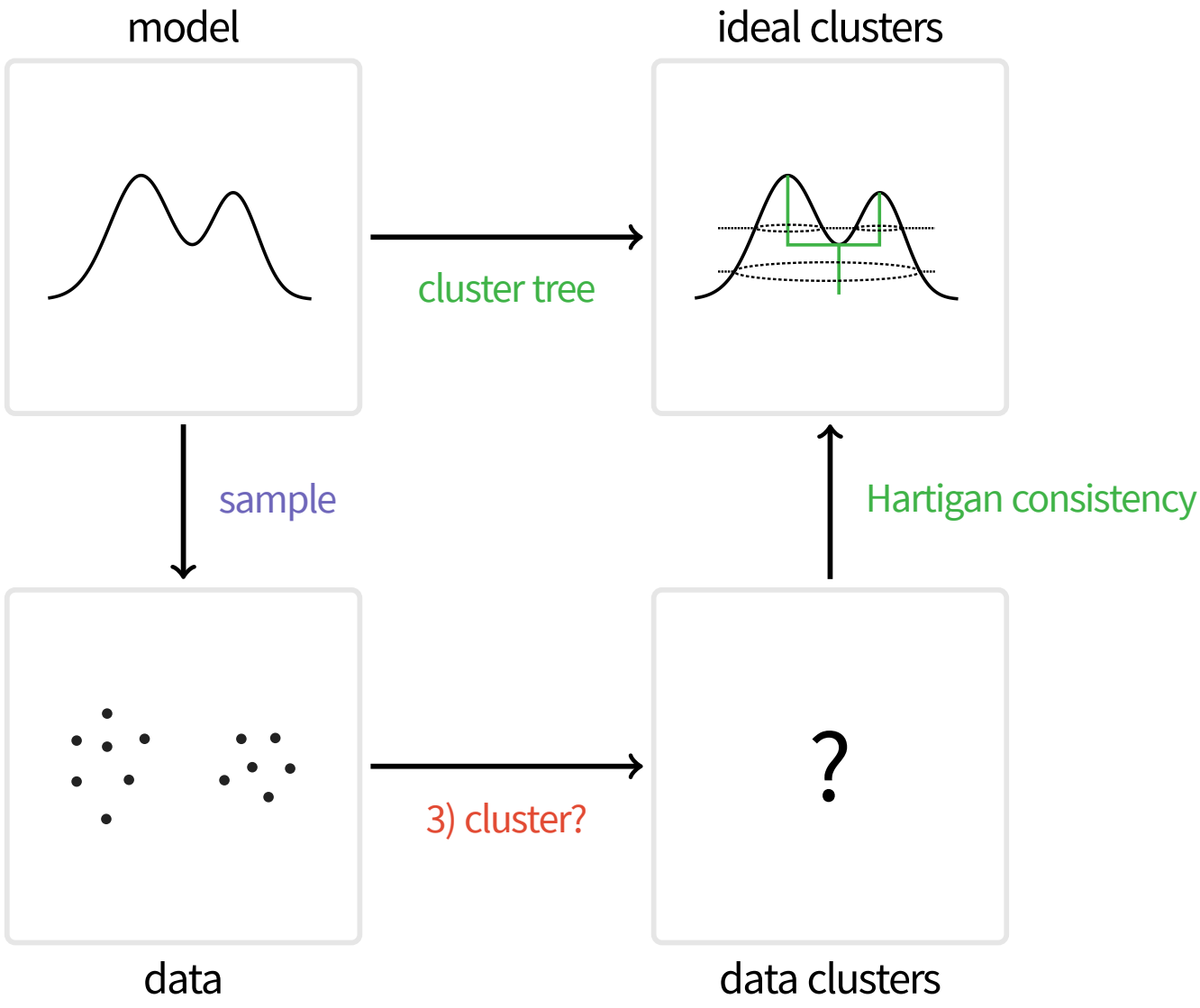


$\hat{\mathcal{C}}_{f,n}$ ✓

Hartigan consistency is a notion of convergence.

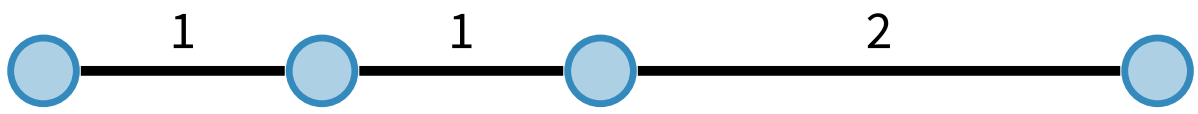


Step 3: Does a Hartigan consistent algorithm exist?



Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:

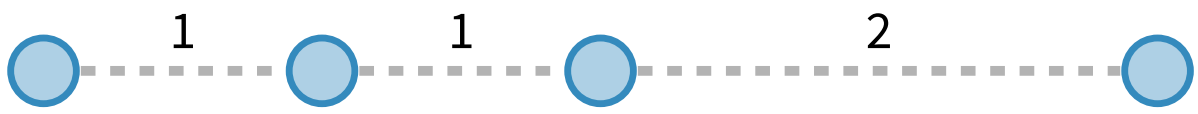


Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



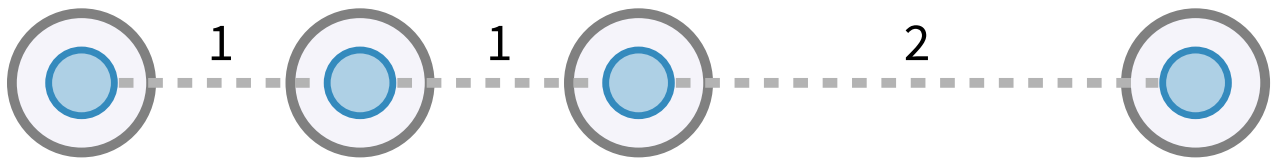
Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$$\lambda < 1$$

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



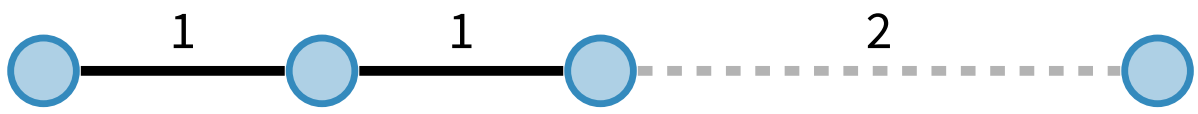
Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$\lambda < 1$

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



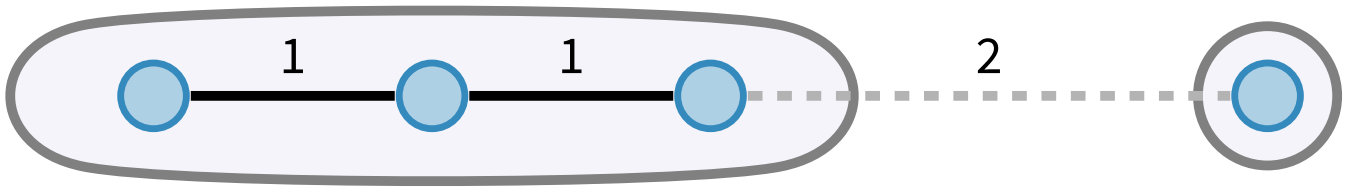
Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$$1 \leq \lambda < 2$$

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



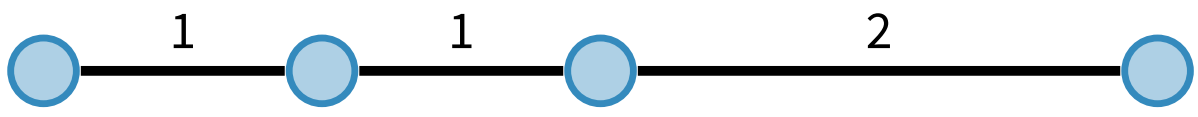
Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$$1 \leq \lambda < 2$$

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



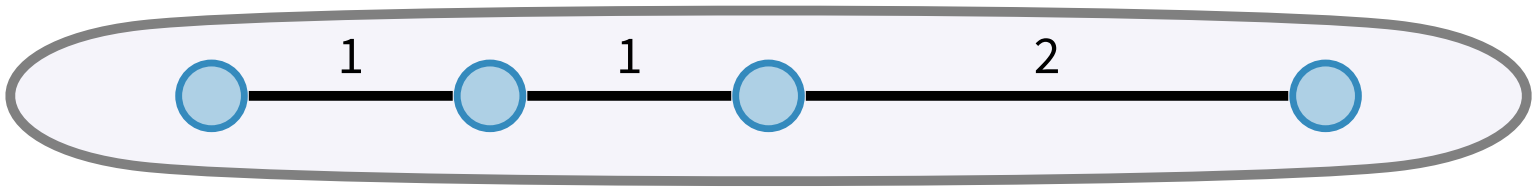
Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$$2 \leq \lambda$$

Hartigan (1981) analyzed single-linkage clustering.

Given a weighted graph:



Clusters are components of "sub-level graphs":

1. Fix level λ, cut all edges of weight $> \lambda$.
2. Clusters at level λ are the connected components.

$$2 \leq \lambda$$

(1981): Single-linkage is not Hartigan consistent.

(1981): Single-linkage is not* Hartigan consistent.

*In dimensions > 1.

(1981): Single-linkage is not* Hartigan consistent.

30 years pass...

*In dimensions > 1.

(1981): Single-linkage is not* Hartigan consistent.

30 years pass...

Proven Hartigan consistent:

(2010): Robust single-linkage of Chaudhuri & Dasgupta
(2011): Tree pruning of Kpotufe & von Luxburg

*In dimensions > 1.

Robust single-linkage of Chaudhuri & Dasgupta:

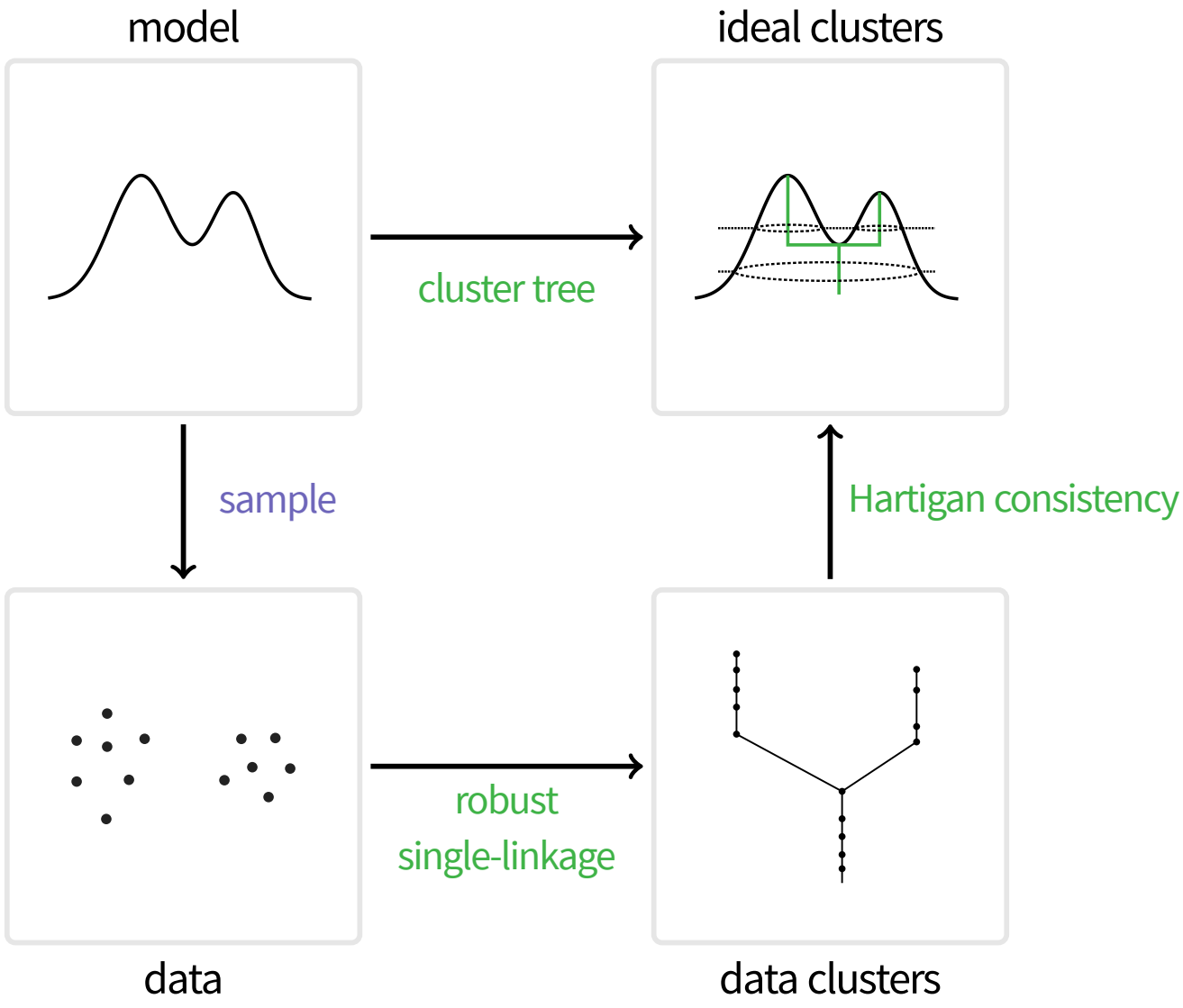
1. Fix level λ, cut all edges of weight $> \alpha\lambda$.
2. Remove low density nodes.
3. Clusters at level λ are the connected components.

Robust single-linkage of Chaudhuri & Dasgupta:

1. Fix level λ, cut all edges of weight $> \alpha\lambda$.
2. Remove low density nodes.
3. Clusters at level λ are the connected components.

Chaudhuri & Dasgupta (2010) prove Hartigan consistency.

Robust single-linkage converges.



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

- Existing theory of Hartigan consistency
- The weakness of Hartigan’s notion
- Our strong notion of convergence

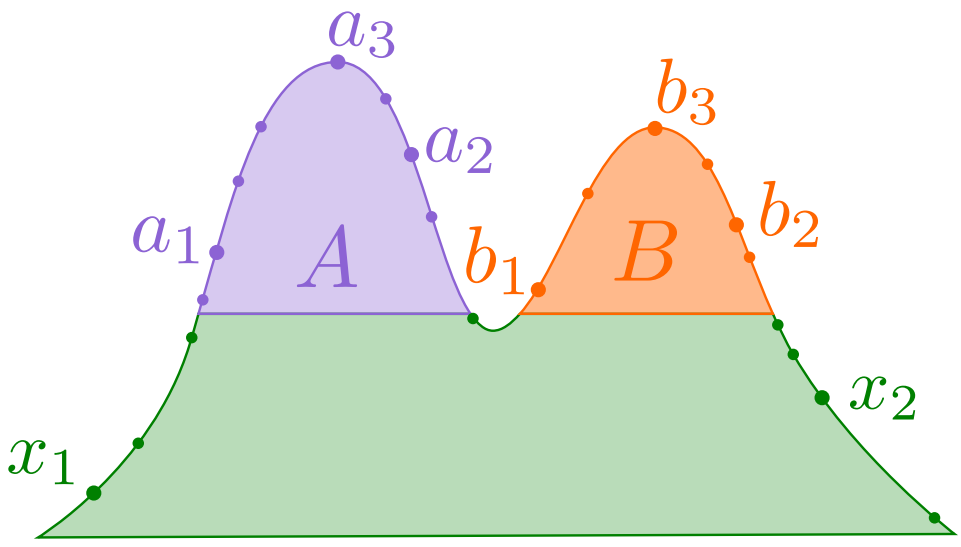
Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

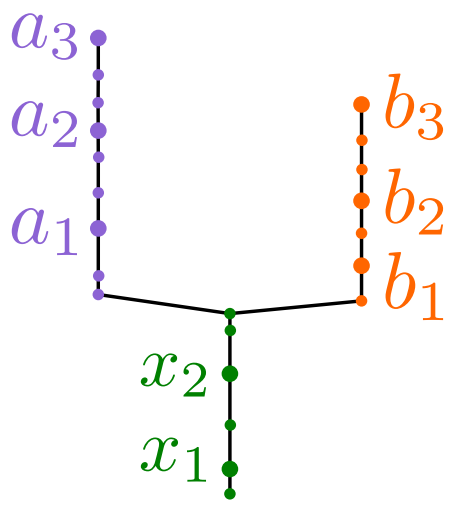
Applications & Directions

Hartigan consistency is insufficient

Hartigan lacks a strong notion of connectedness.



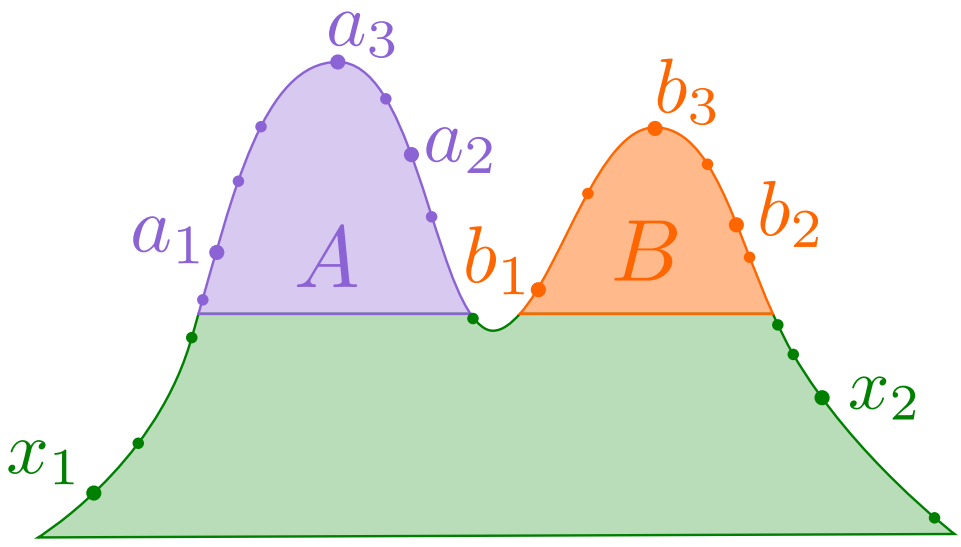
$\mathcal{C}_f$



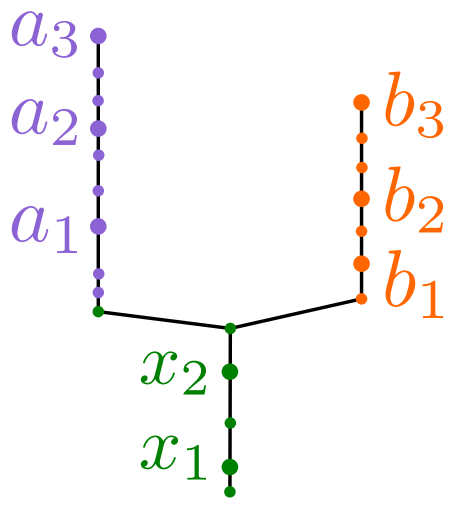
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

Hartigan lacks a strong notion of connectedness.



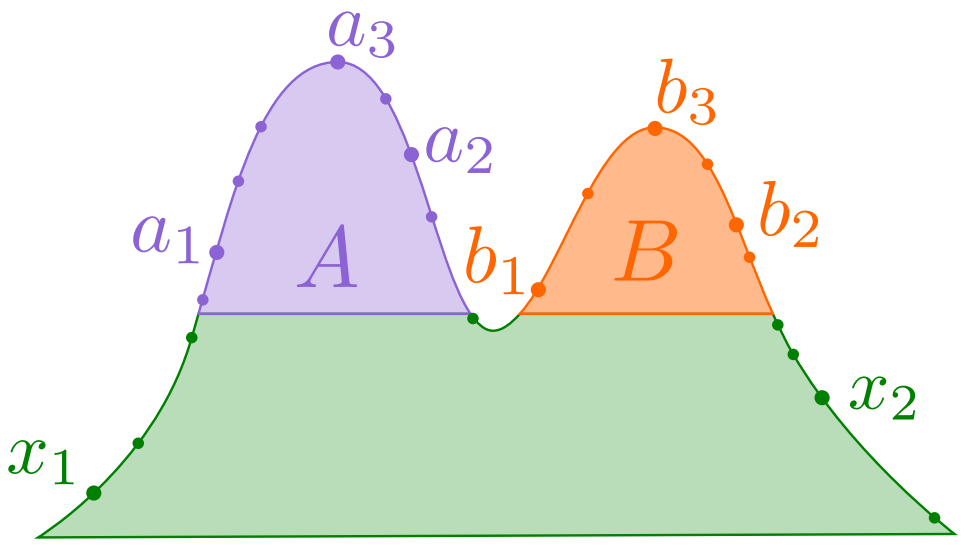
$\mathcal{C}_f$



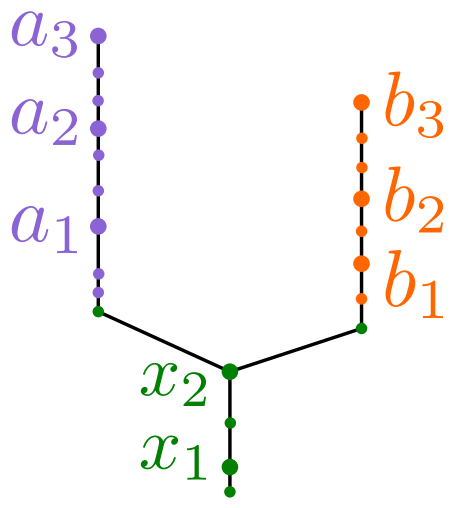
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

Hartigan lacks a strong notion of connectedness.



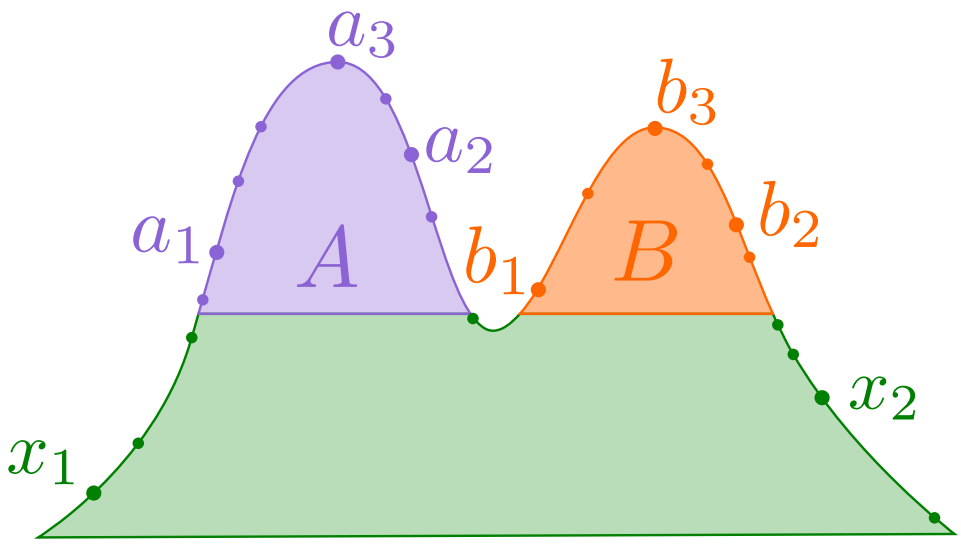
$\mathcal{C}_f$



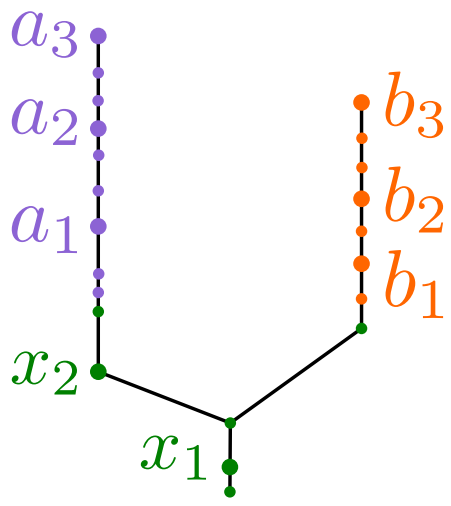
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

Hartigan lacks a strong notion of connectedness.



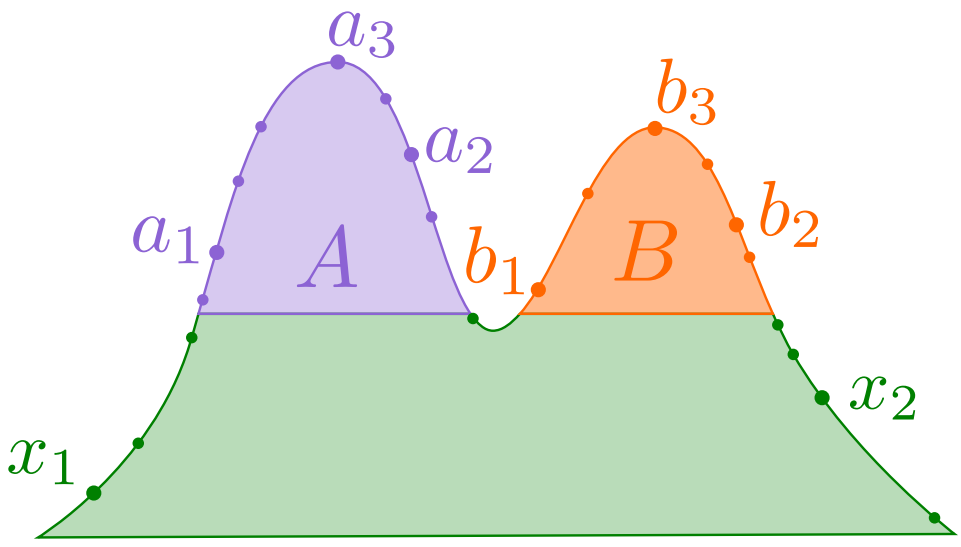
$\mathcal{C}_f$



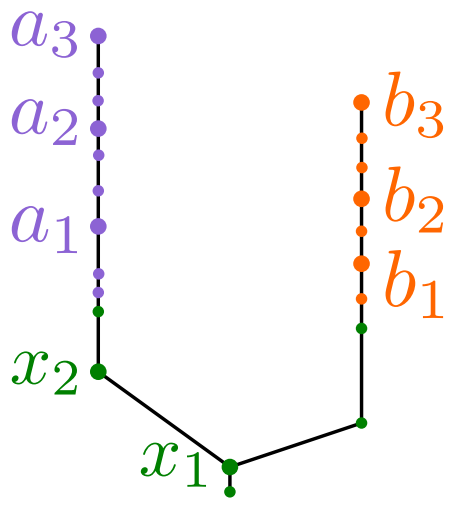
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

Hartigan lacks a strong notion of connectedness.



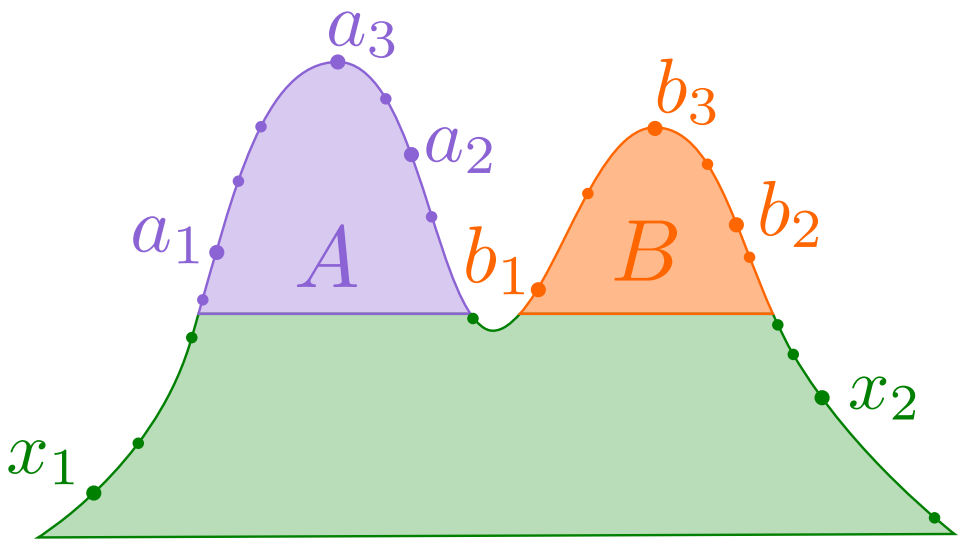
$\mathcal{C}_f$



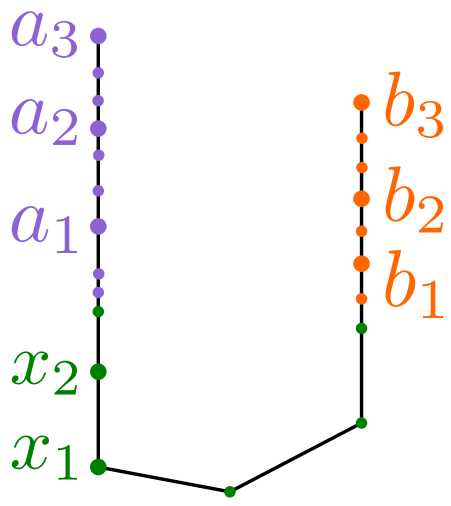
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

This tree does not violate Hartigan consistency!



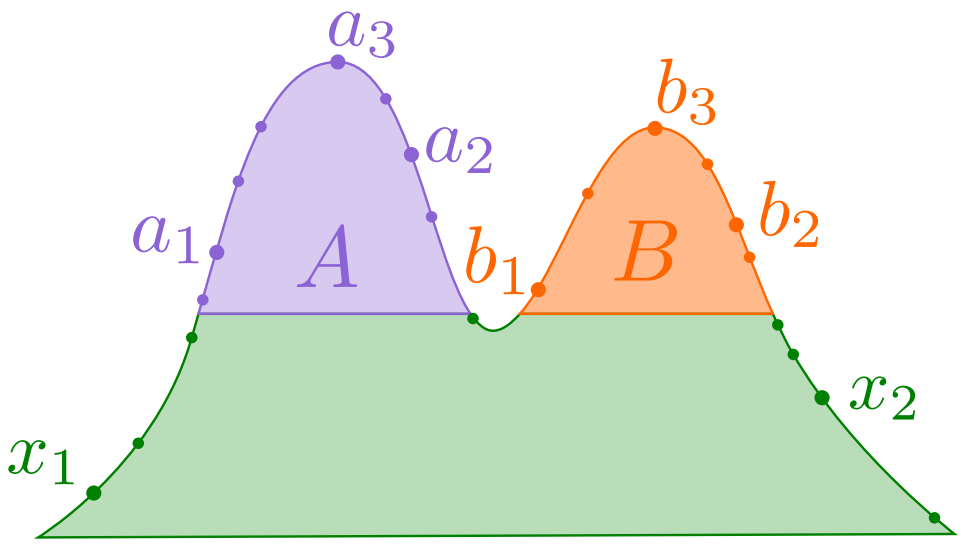
$\mathcal{C}_f$



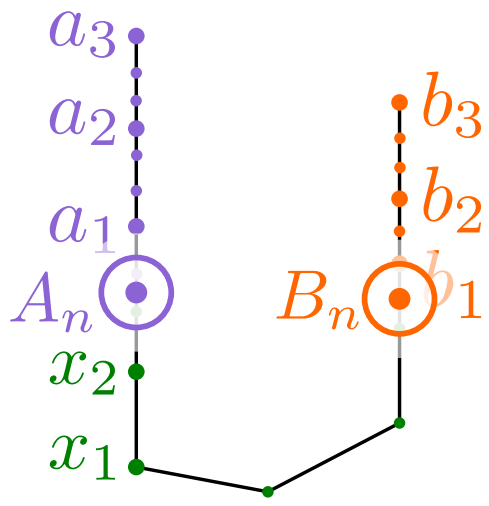
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

This tree does not violate Hartigan consistency!



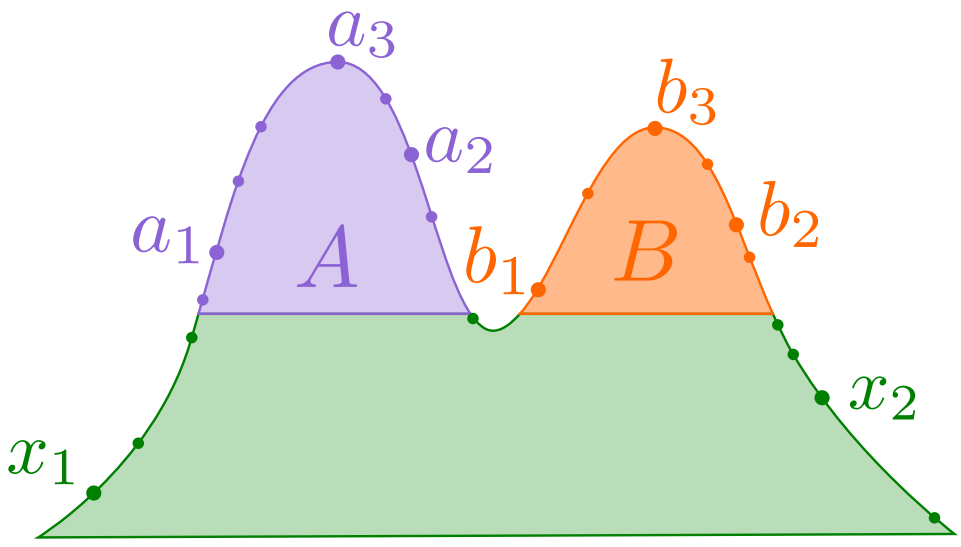
$\mathcal{C}_f$



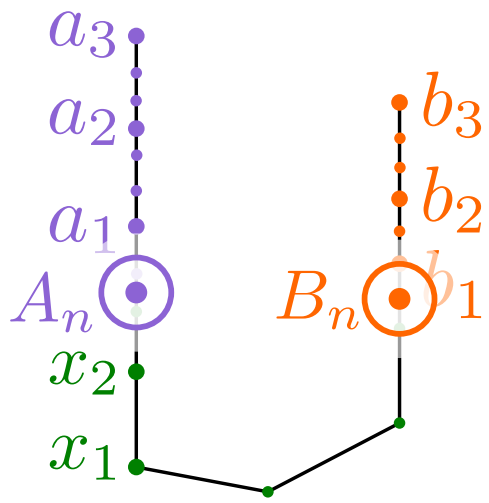
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

This tree does not violate Hartigan consistency!



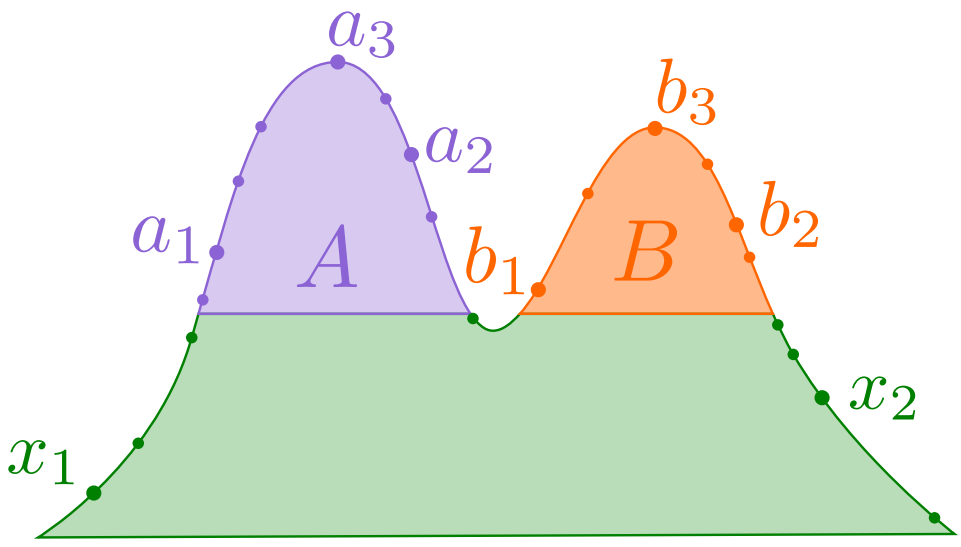
$\mathcal{C}_f$



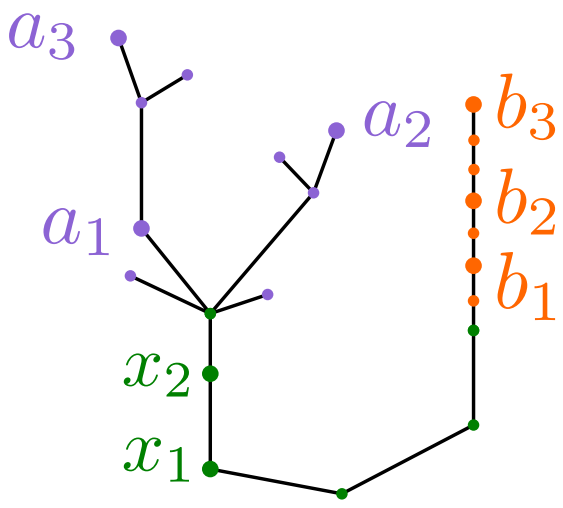
$\hat{\mathcal{C}}_{f,n}$ ✓

Hartigan consistency is insufficient

What about this tree?



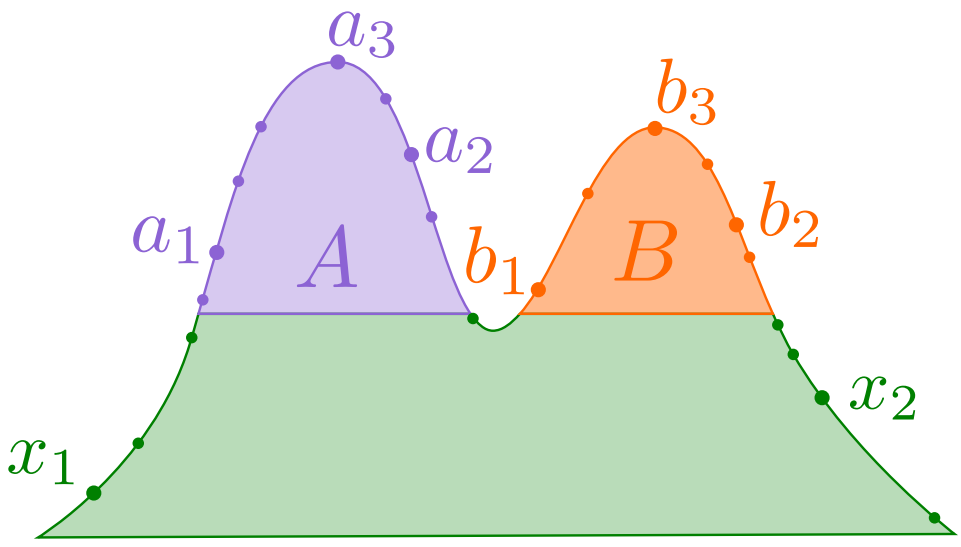
$\mathcal{C}_f$



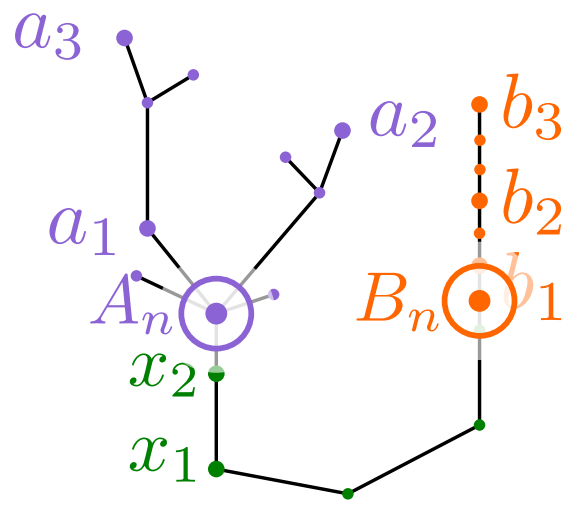
$\hat{\mathcal{C}}_{f,n}$

Hartigan consistency is insufficient

What about this tree? Also consistent!



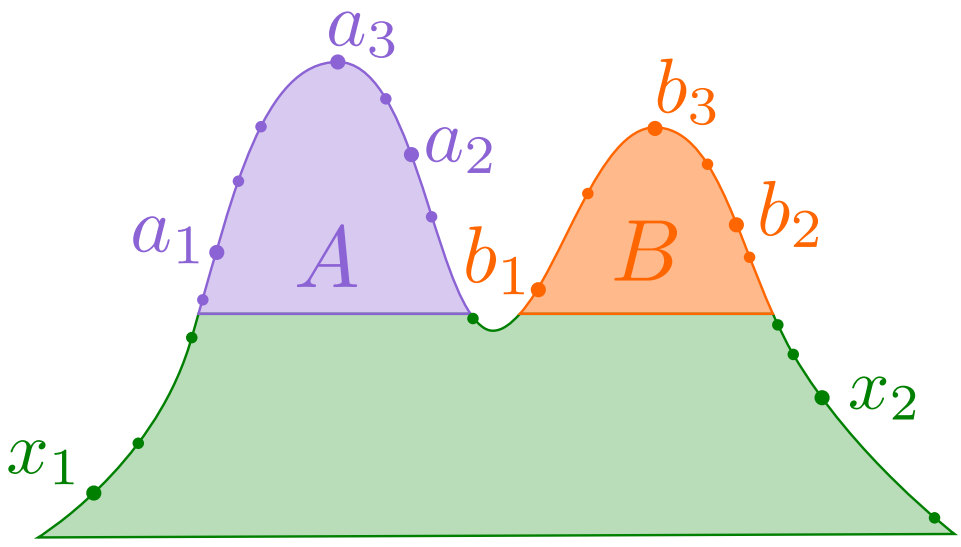
$\mathcal{C}_f$



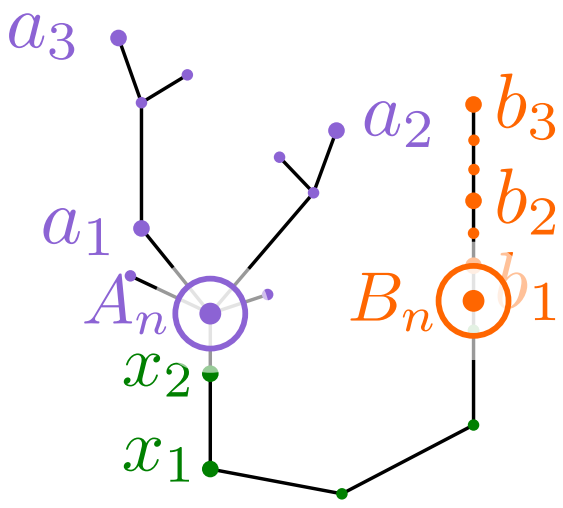
$\hat{\mathcal{C}}_{f,n}$ ✓

Hartigan consistency is insufficient

A tree can be Hartigan consistent yet very different from the true tree.



$\mathcal{C}_f$



$\hat{\mathcal{C}}_{f,n}$ ✓

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

- Existing theory of Hartigan consistency
- The weakness of Hartigan’s notion
- Our strong notion of convergence

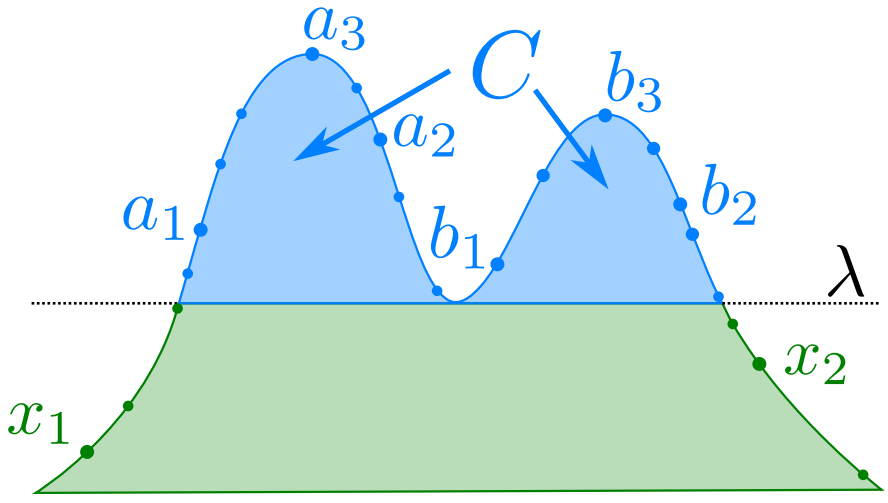
Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

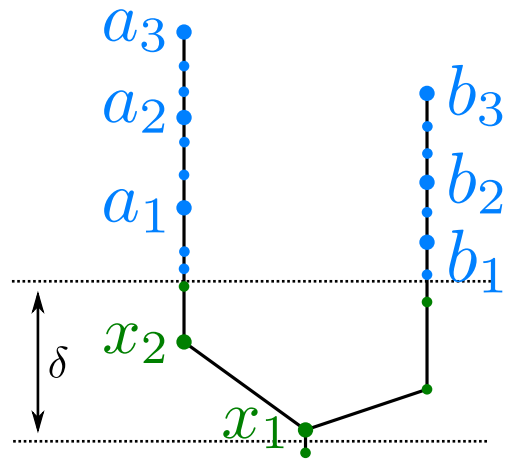
Applications & Directions

Hartigan consistency lacks minimality.

Suppose C is a cluster at level λ.
C_n should not contain points of density much less than λ.



$\mathcal{C}_f$

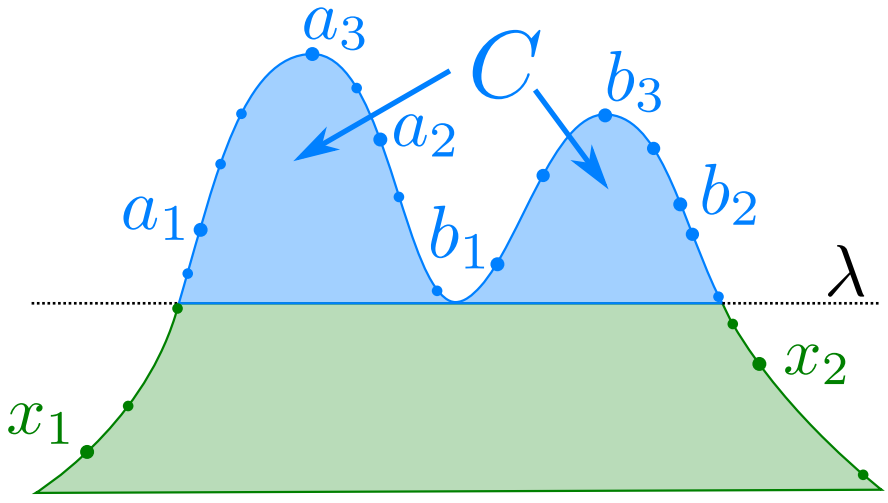


$\hat{\mathcal{C}}_{f,n}$

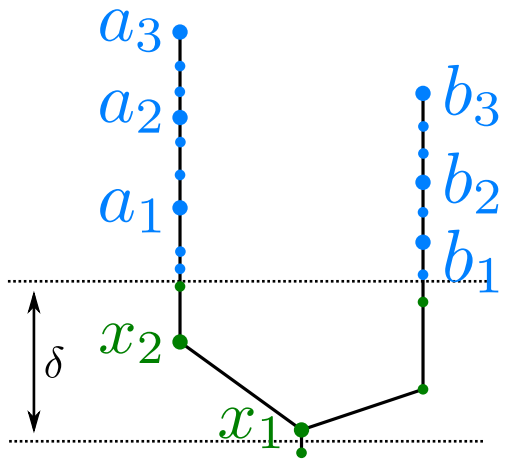
Hartigan consistency lacks minimality.

Definition (Eldridge, 2015):

Minimality: For cluster C at level λ, $\min_{x \in C_n} f(x) > \lambda - \delta$ with $\delta \to 0$.



$\mathcal{C}_f$

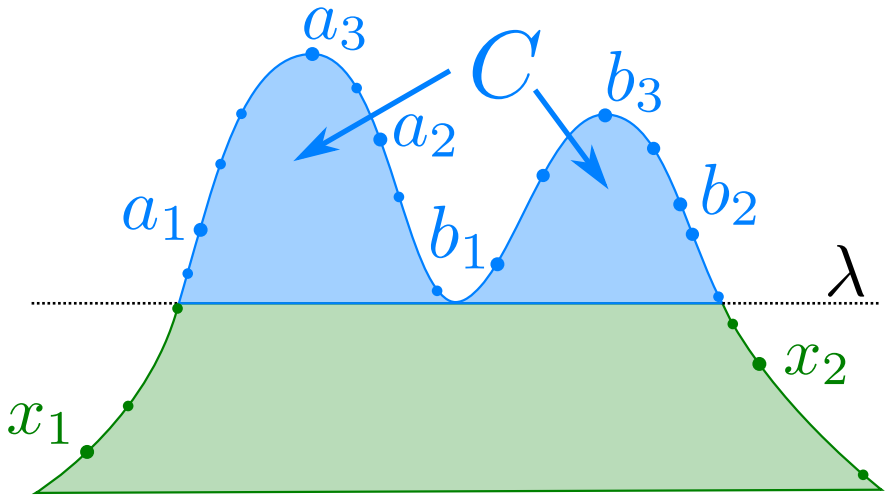


$\hat{\mathcal{C}}_{f,n}$

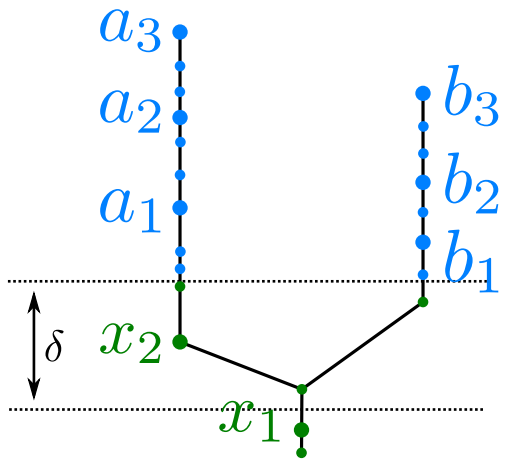
Hartigan consistency lacks minimality.

Definition (Eldridge, 2015):

Minimality: For cluster C at level λ, $\min_{x \in C_n} f(x) > \lambda - \delta$ with $\delta \to 0$.



$\mathcal{C}_f$

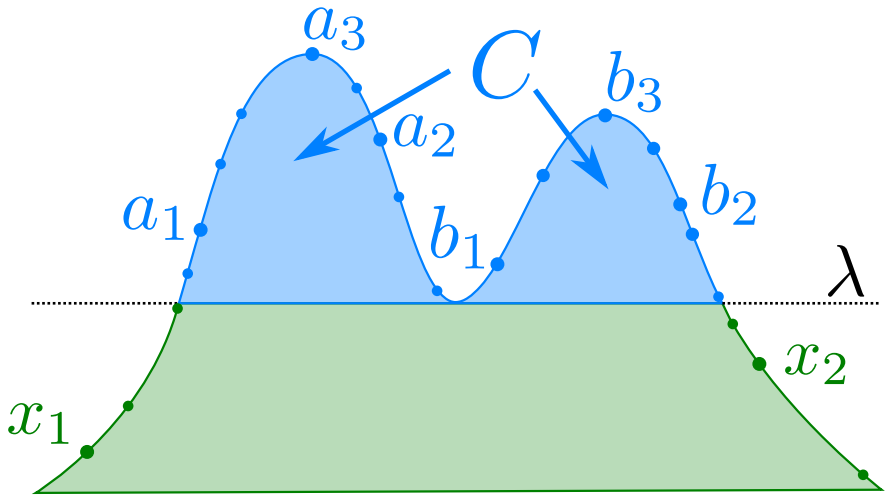


$\hat{\mathcal{C}}_{f,n}$

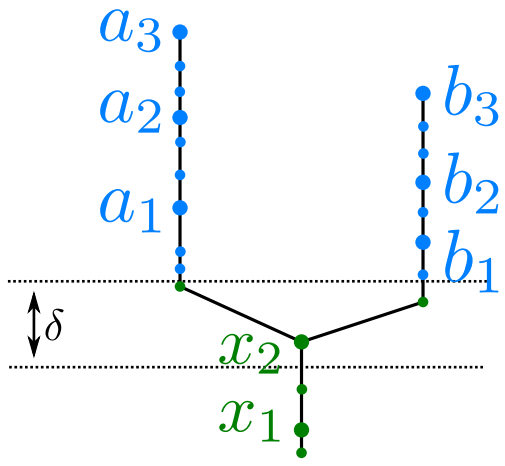
Hartigan consistency lacks minimality.

Definition (Eldridge, 2015):

Minimality: For cluster C at level λ, $\min_{x \in C_n} f(x) > \lambda - \delta$ with $\delta \to 0$.



$\mathcal{C}_f$

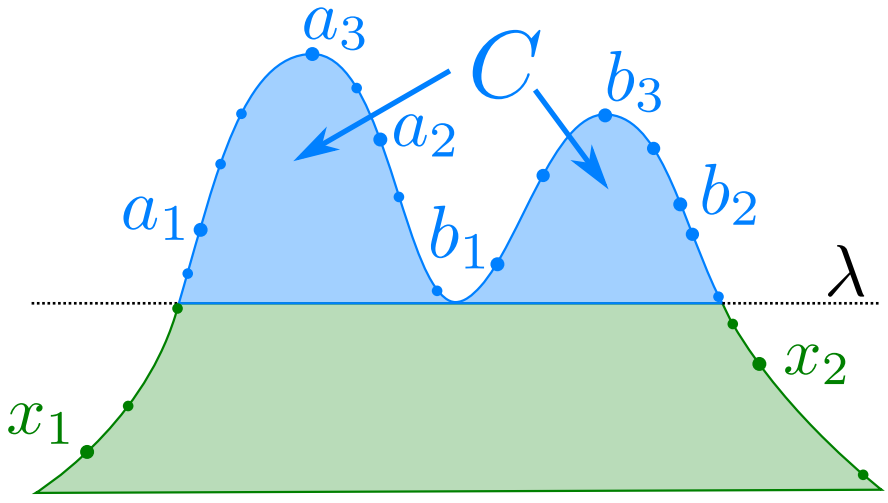


$\hat{\mathcal{C}}_{f,n}$

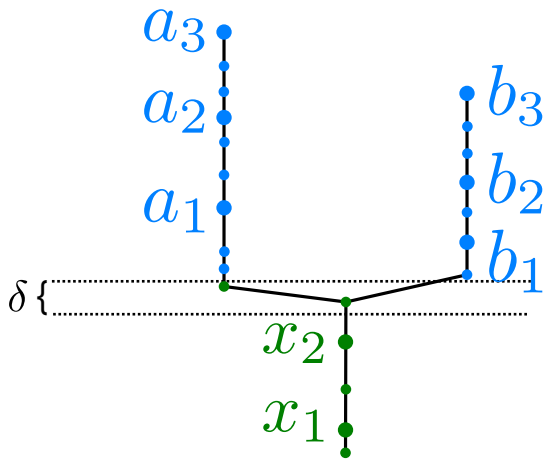
Hartigan consistency lacks minimality.

Definition (Eldridge, 2015):

Minimality: For cluster C at level λ, $\min_{x \in C_n} f(x) > \lambda - \delta$ with $\delta \to 0$.



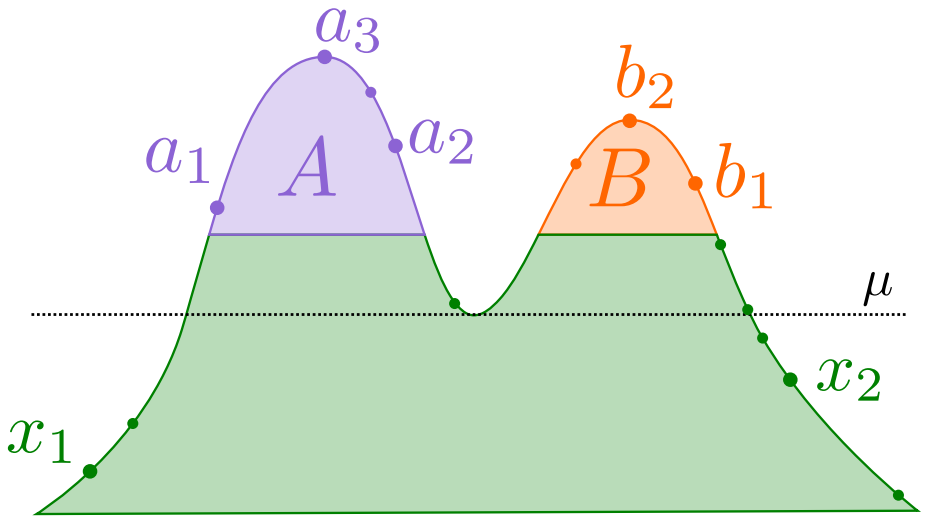
$\mathcal{C}_f$



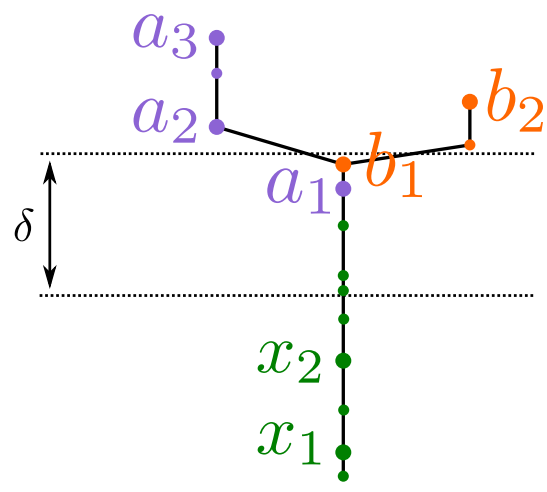
$\hat{\mathcal{C}}_{f,n}$

Definition (Eldridge, 2015):

Separation: If A and B are disjoint clusters merging at μ, $A \cap X_n$ and $B \cap X_n$ are separated at level $\mu + \delta$, with $\delta \to 0$.



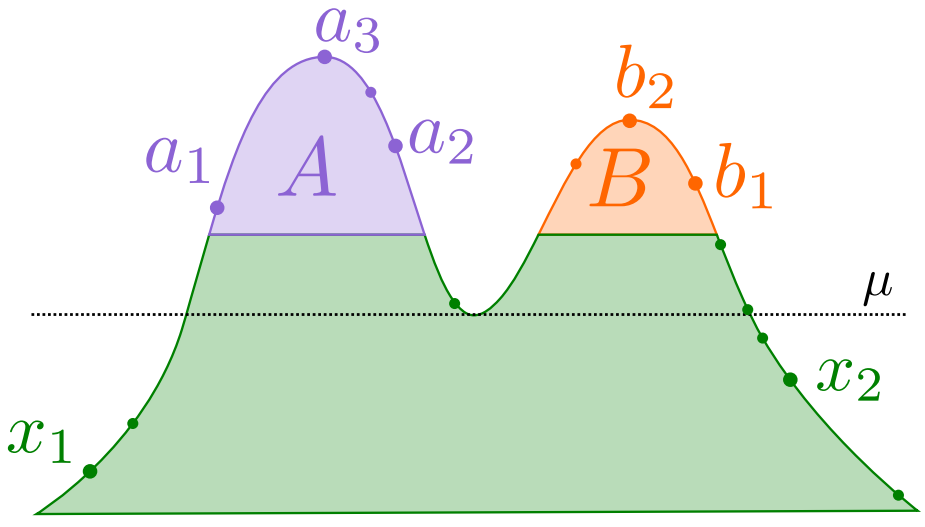
$\mathcal{C}_f$



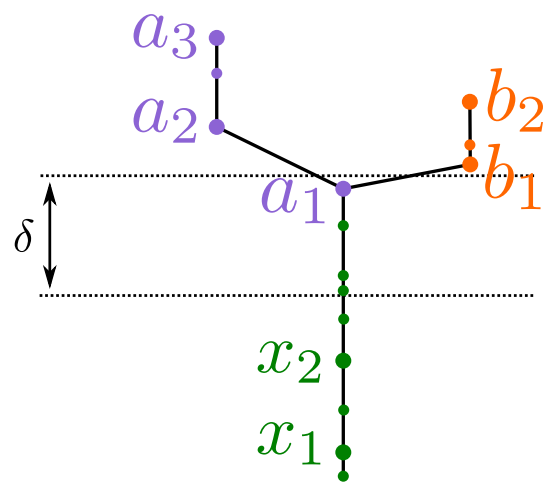
$\hat{\mathcal{C}}_{f,n}$

Definition (Eldridge, 2015):

Separation: If A and B are disjoint clusters merging at μ, $A \cap X_n$ and $B \cap X_n$ are separated at level $\mu + \delta$, with $\delta \to 0$.



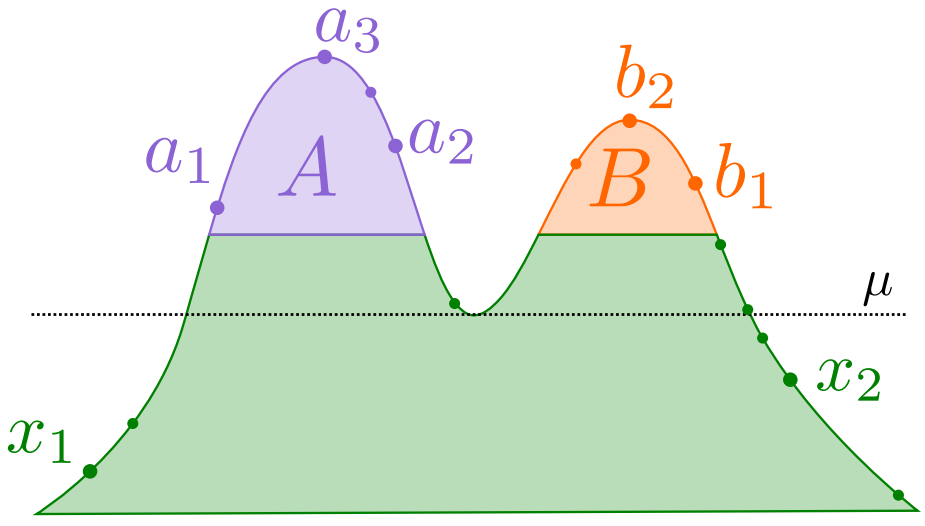
$\mathcal{C}_f$



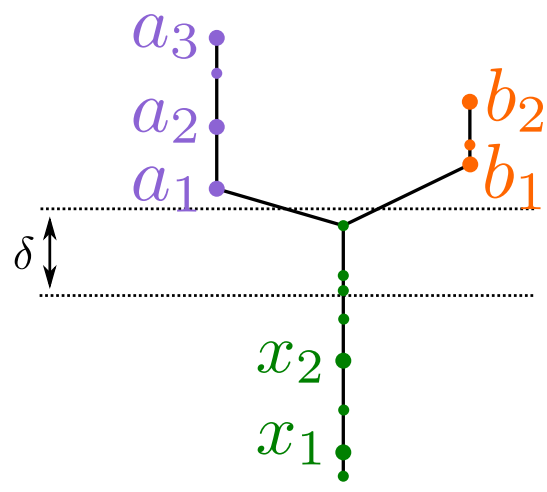
$\hat{\mathcal{C}}_{f,n}$

Definition (Eldridge, 2015):

Separation: If A and B are disjoint clusters merging at μ, $A \cap X_n$ and $B \cap X_n$ are separated at level $\mu + \delta$, with $\delta \to 0$.



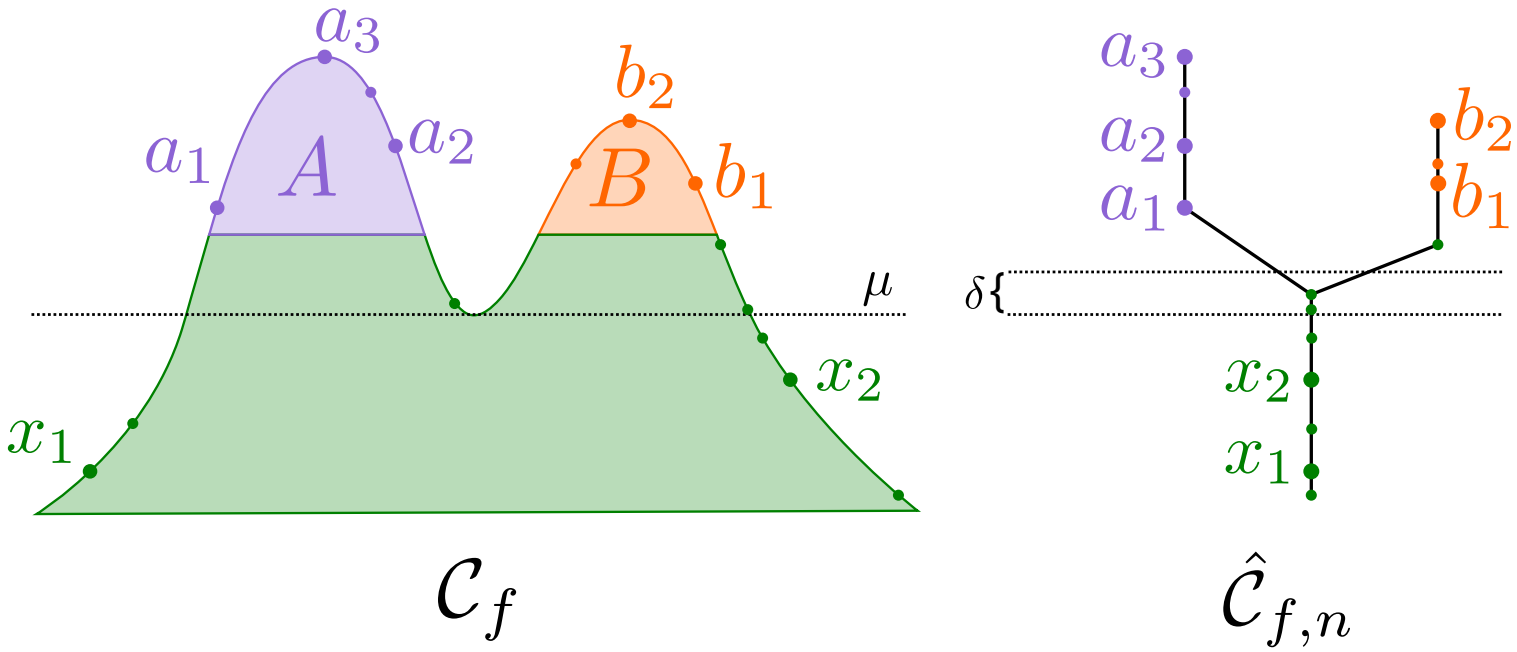
$\mathcal{C}_f$



$\hat{\mathcal{C}}_{f,n}$

Definition (Eldridge, 2015):

Separation: If A and B are disjoint clusters merging at μ, $A \cap X_n$ and $B \cap X_n$ are separated at level $\mu + \delta$, with $\delta \to 0$.



Theorem (Eldridge, 2015):

Minimality + Separation $\Longrightarrow$ Hartigan consistency

Theorem (Eldridge, 2015):

Minimality + Separation $\implies$ Hartigan consistency

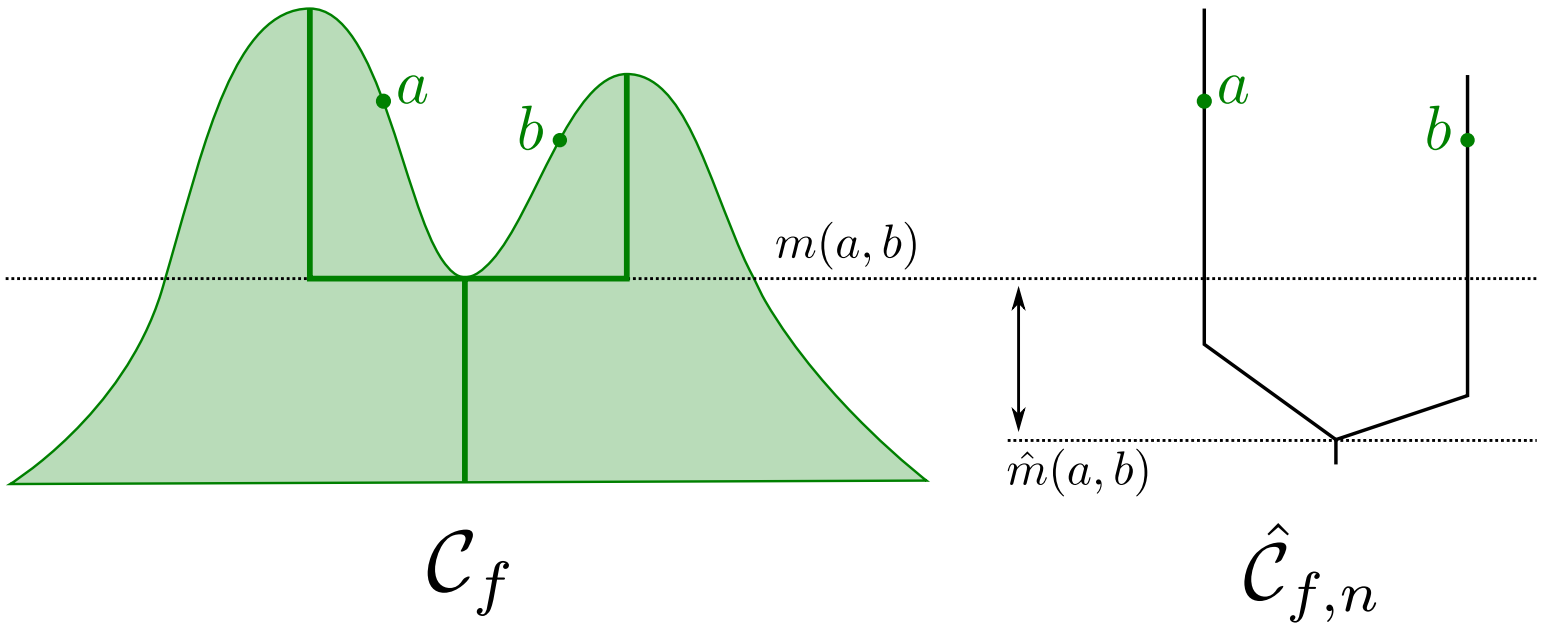
Hartigan consistency $\not\implies$ Minimality + Separation

Theorem (Eldridge, 2015):

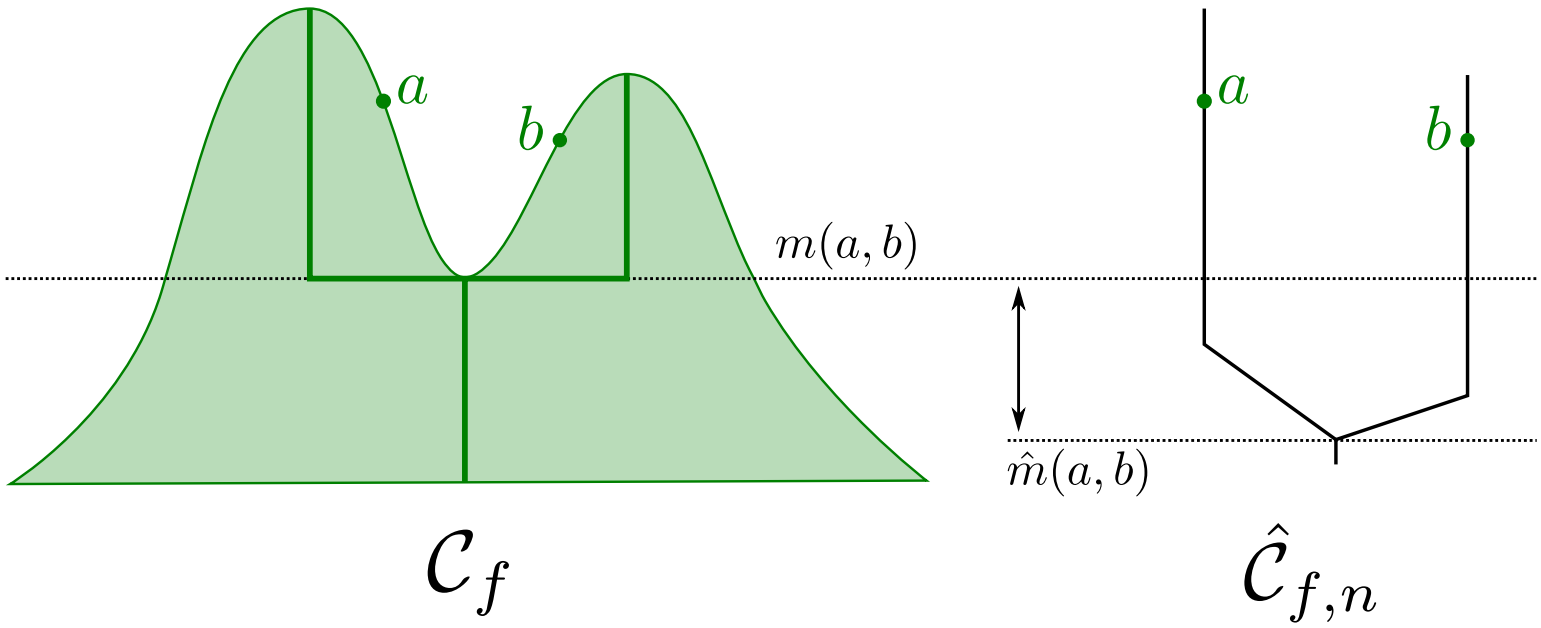
Minimality + Separation $\implies$ Hartigan consistency

Hartigan consistency $\not\implies$ Minimality + Separation

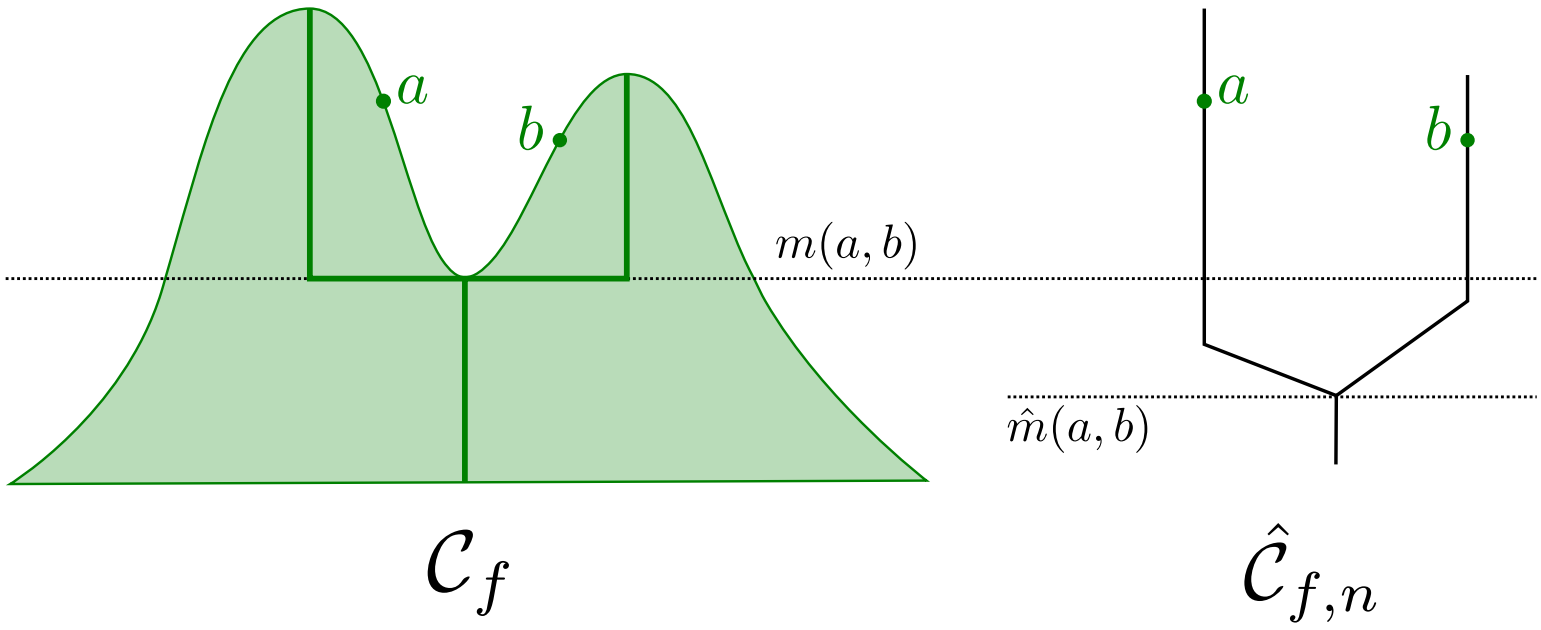
Minimality and Separation are limit properties.
Can we **quantify** how close a clustering is to the density cluster tree?



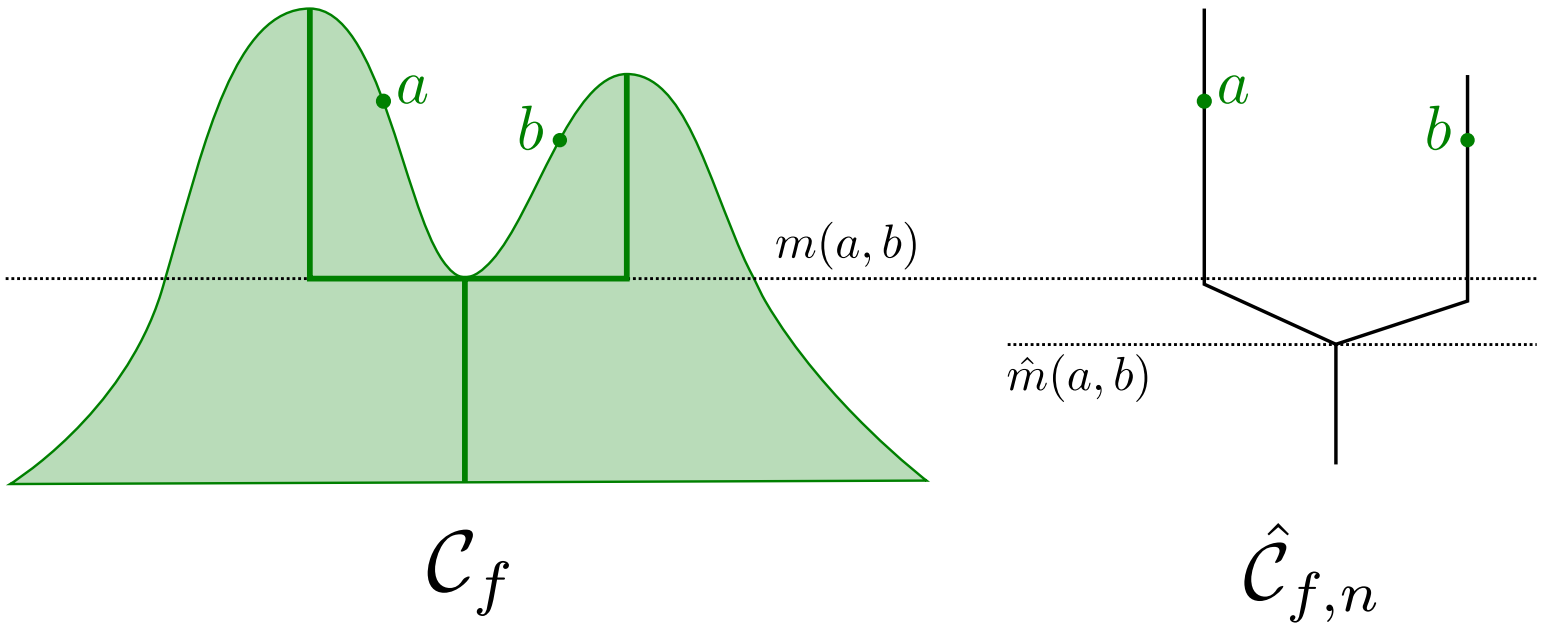
- The ideal merge height: $m(a, b)$
- The empirical merge height: $\hat{m}(a, b)$



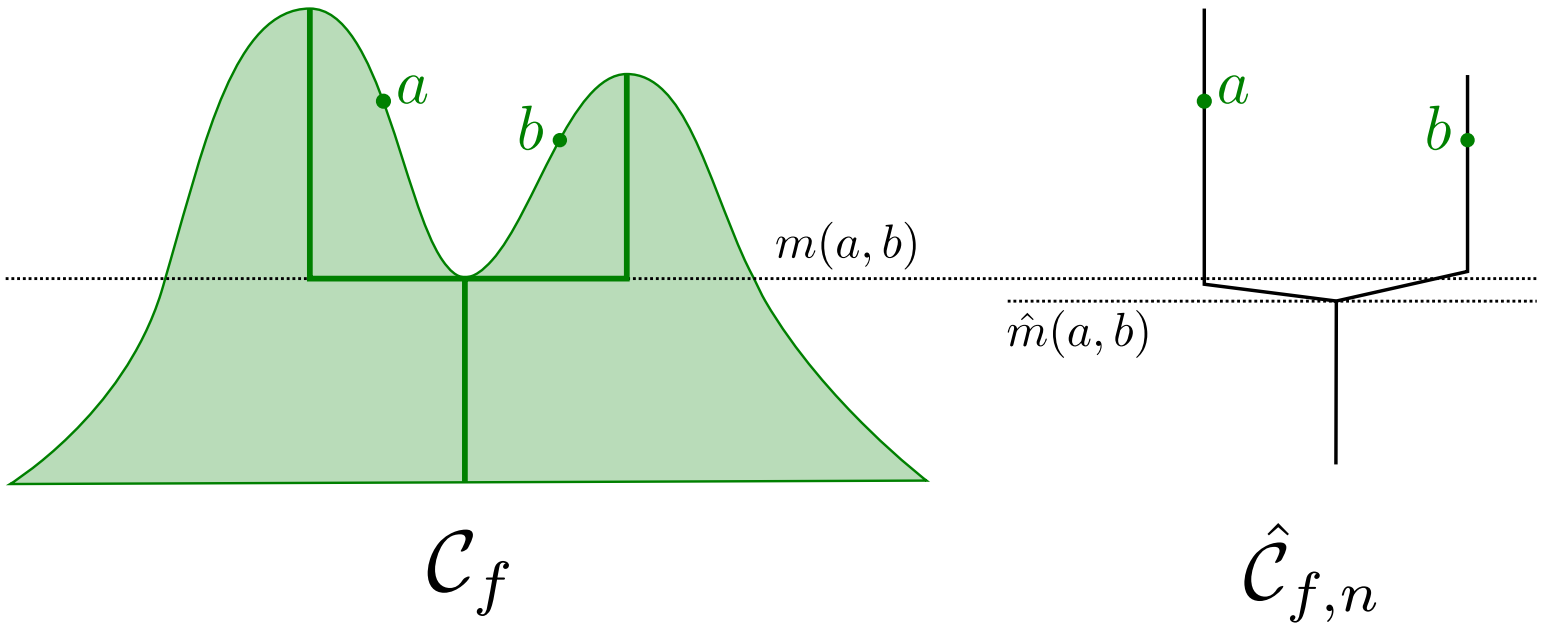
- The ideal merge height: $m(a, b)$
- The empirical merge height: $\hat{m}(a, b)$
- Minimality + Separation $\implies \hat{m}(a, b) \to m(a, b)$



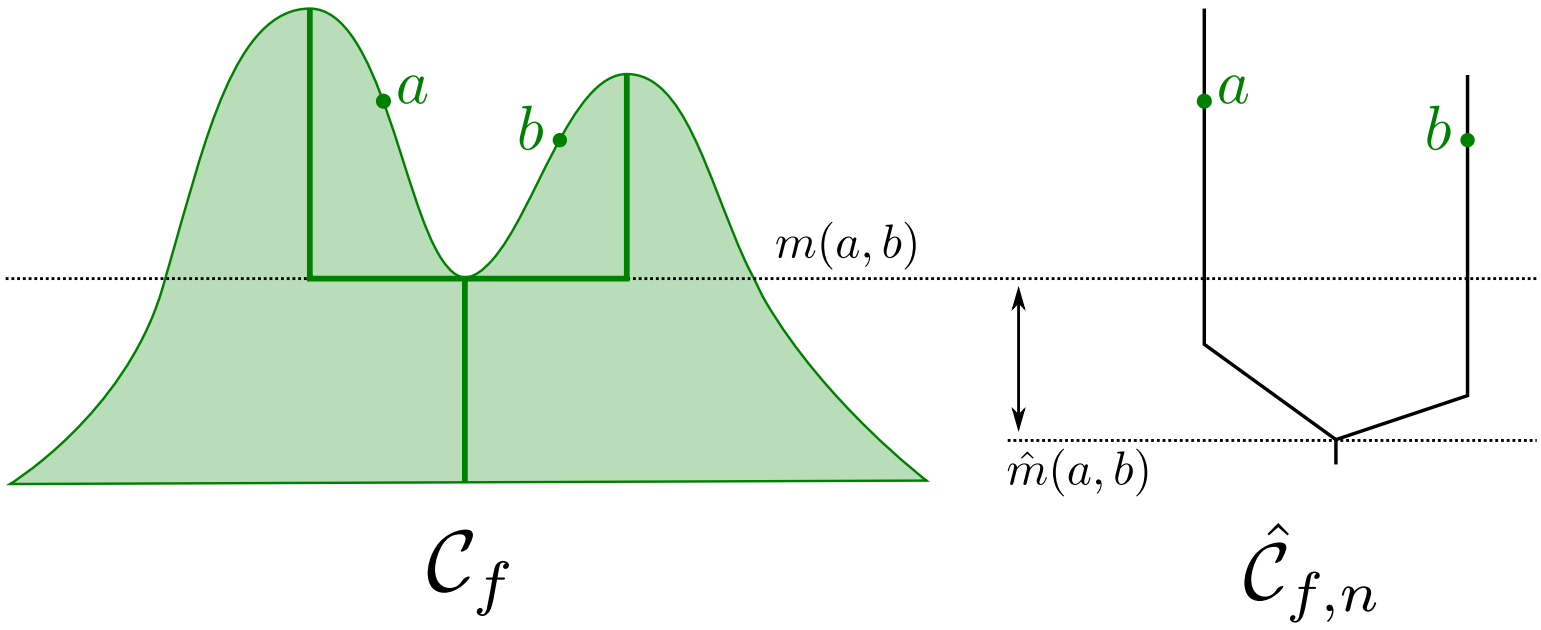
- The ideal merge height: $m(a, b)$
- The empirical merge height: $\hat{m}(a, b)$
- Minimality + Separation $\implies$ $\hat{m}(a, b) \to m(a, b)$



- The ideal merge height: $m(a, b)$
- The empirical merge height: $\hat{m}(a, b)$
- Minimality + Separation $\implies \hat{m}(a, b) \to m(a, b)$



- The ideal merge height: $m(a, b)$
- The empirical merge height: $\hat{m}(a, b)$
- Minimality + Separation $\implies \hat{m}(a, b) \to m(a, b)$



Definition (Eldridge, 2015):

The merge distortion between the cluster tree and its estimate is $d(\mathcal{C}_f, \hat{\mathcal{C}}_{f,n}) = \max_{(x,x') \in X_n} \left| m(x, x') - \hat{m}(x, x') \right|$.

Theorem (Eldridge, 2015):

Convergence in merge distortion $\iff$ Minimality + Separation

Corollary (Eldridge, 2015):

Convergence in merge distortion $\implies$ Hartigan consistency

Theorem (Eldridge, 2015):

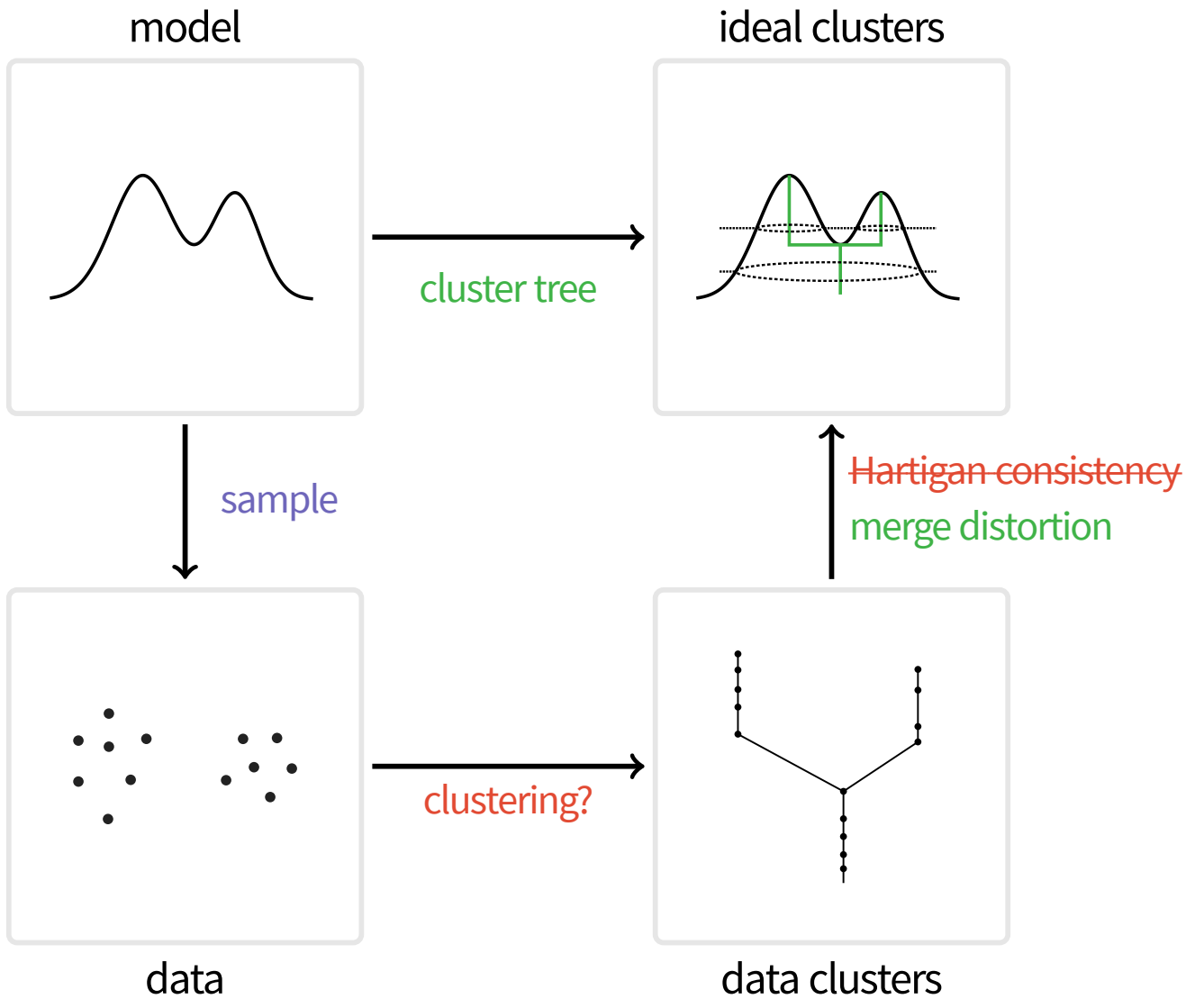
Convergence in merge distortion $\iff$ Minimality + Separation

Corollary (Eldridge, 2015):

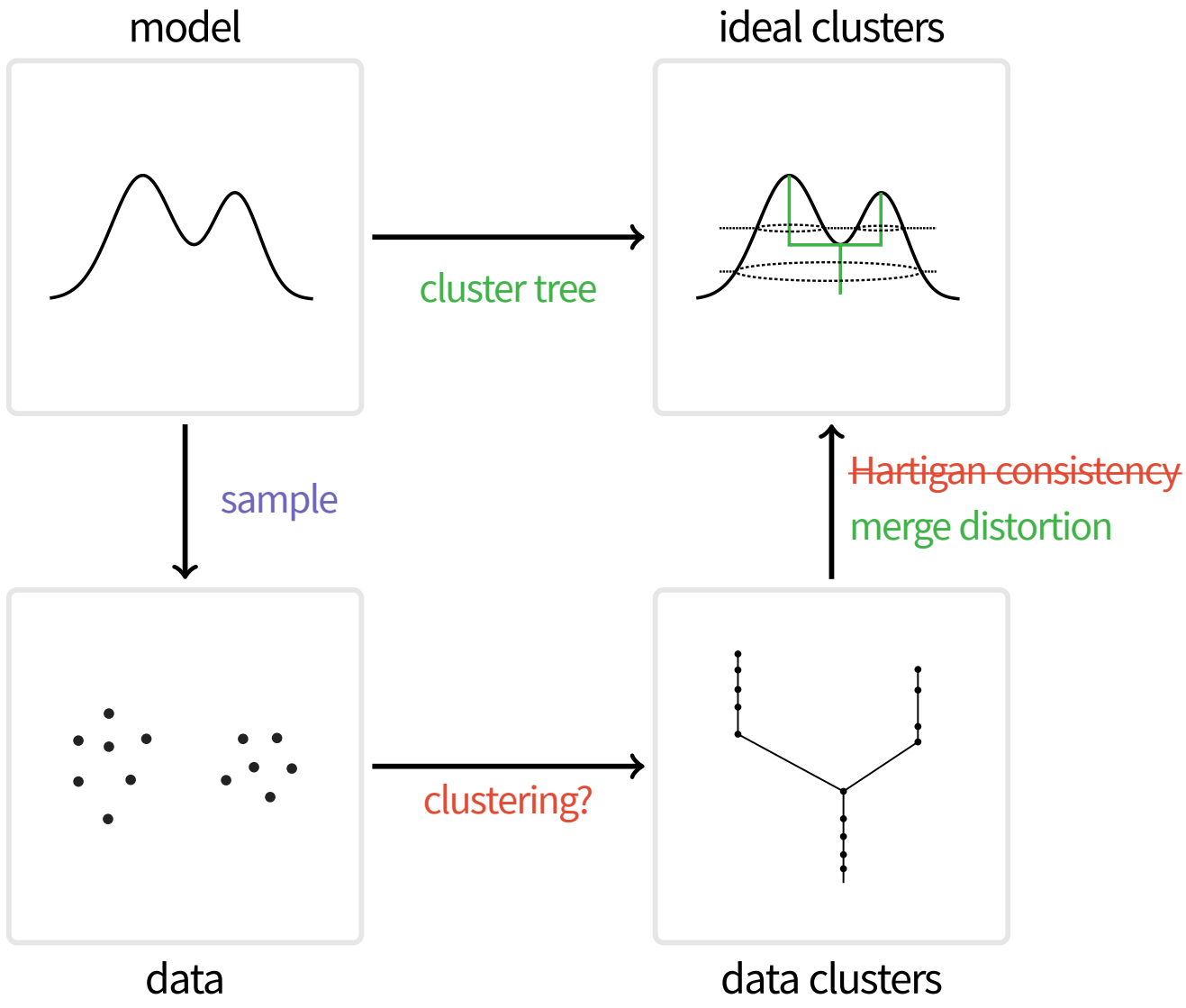
Convergence in merge distortion $\implies$ Hartigan consistency

Hartigan consistency $\not\implies$ convergence in merge distortion.

Merge distortion is stronger than Hartigan consistency.



Step 3: Does an algorithm exist which converges in merge distortion?



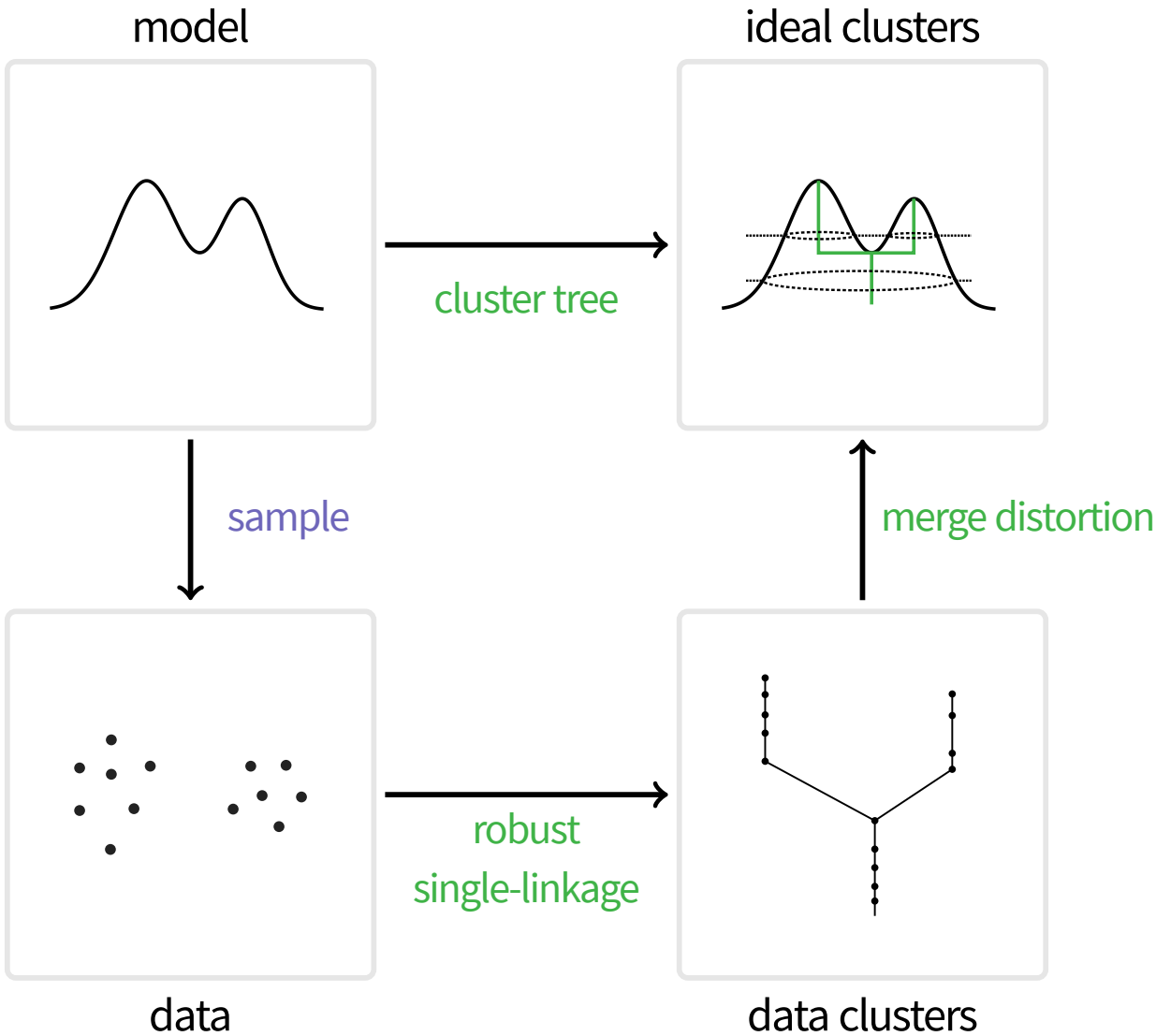
It took 30 years to prove Hartigan consistency...

It took 30 years to prove Hartigan consistency...

Theorem (Eldridge, 2015):

Assume the density f is Lipschitz and compactly supported. Then robust single-linkage converges in merge distortion to the density cluster tree.

A theory of convergence in merge distortion.



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

Recap:

- Hartigan consistency was too weak
- We replaced it with minimality and separation
- Introduced distance between clusterings
- Our notion of convergence is stronger
- We prove that robust single-linkage converges

Context

Approaches to “correct” clustering.

Part I: Density

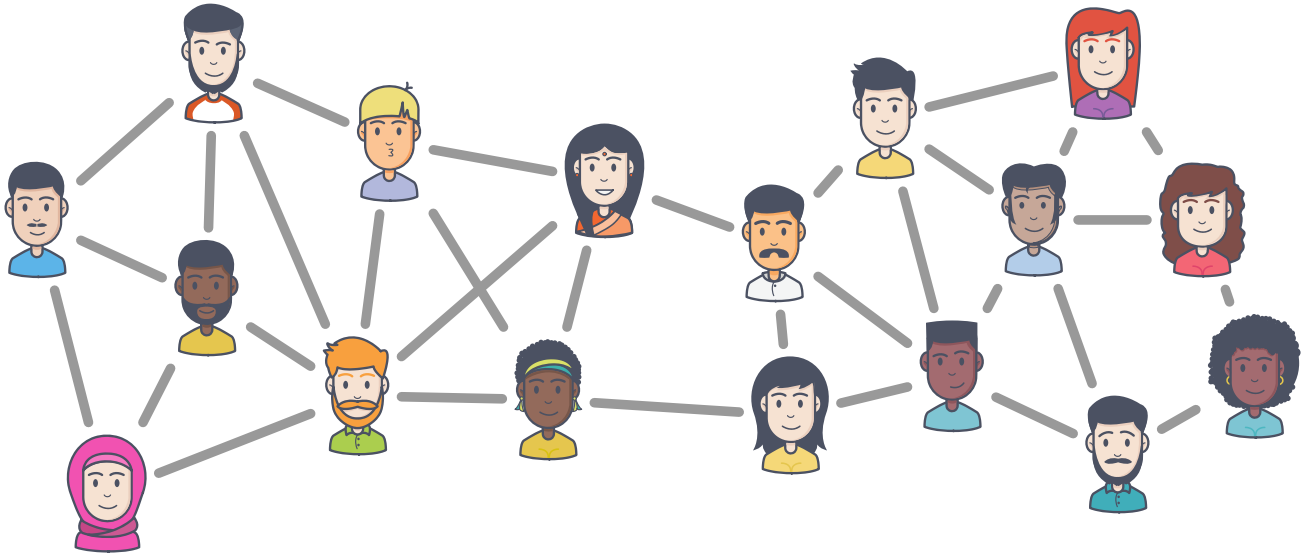
Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

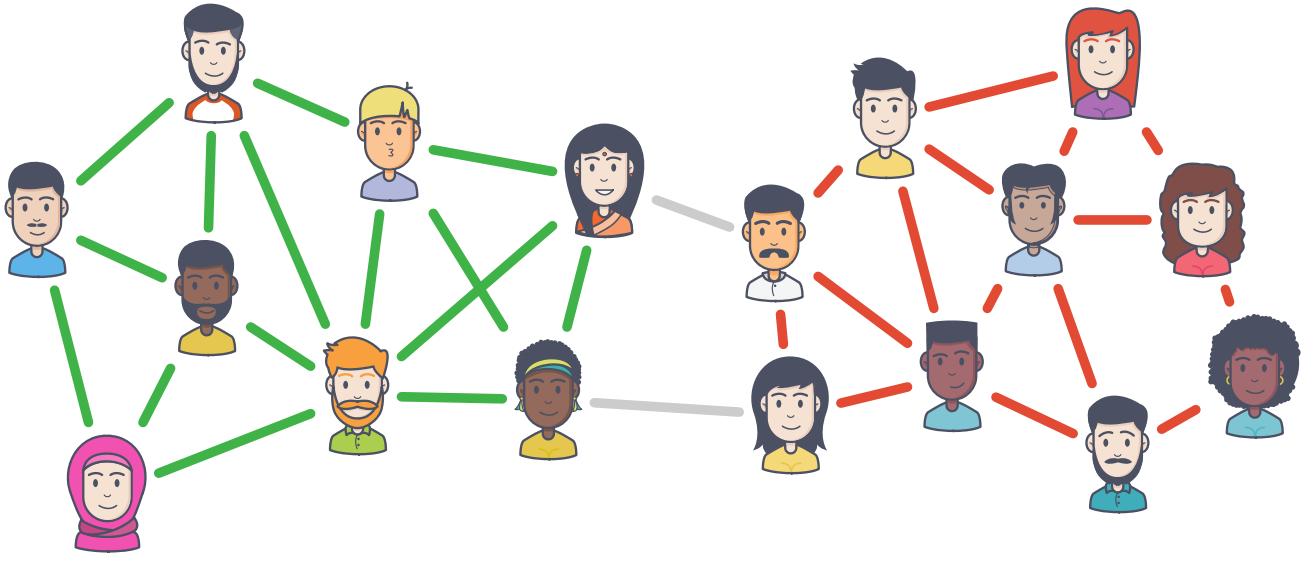
Part II: Graphon

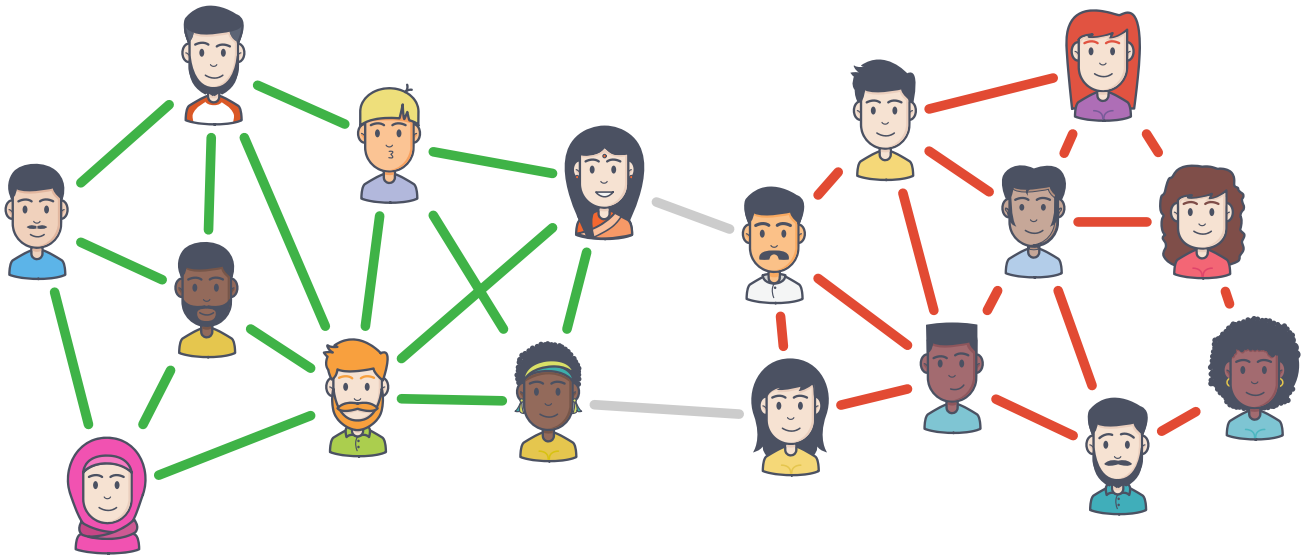
Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

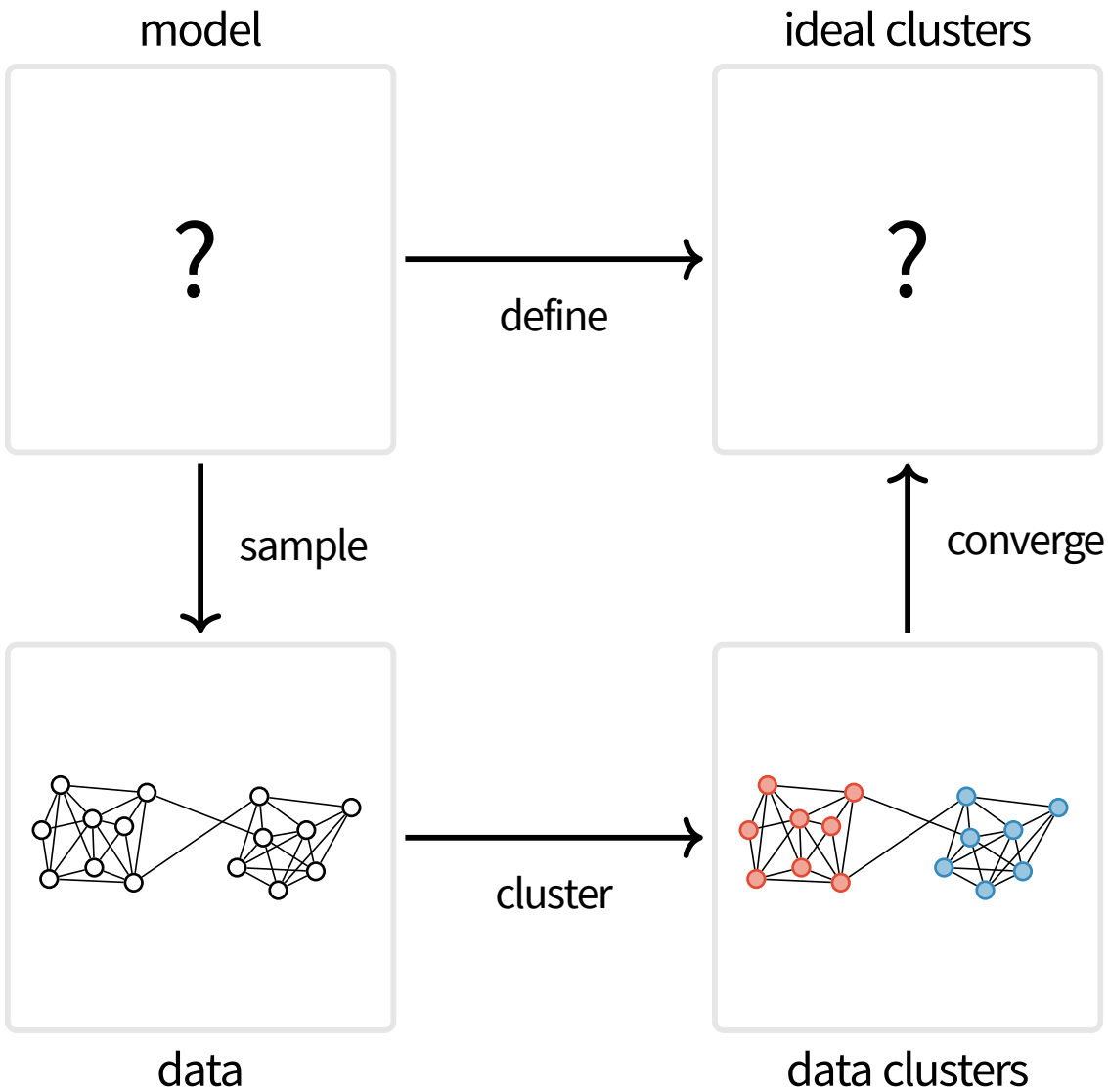
Applications & Directions

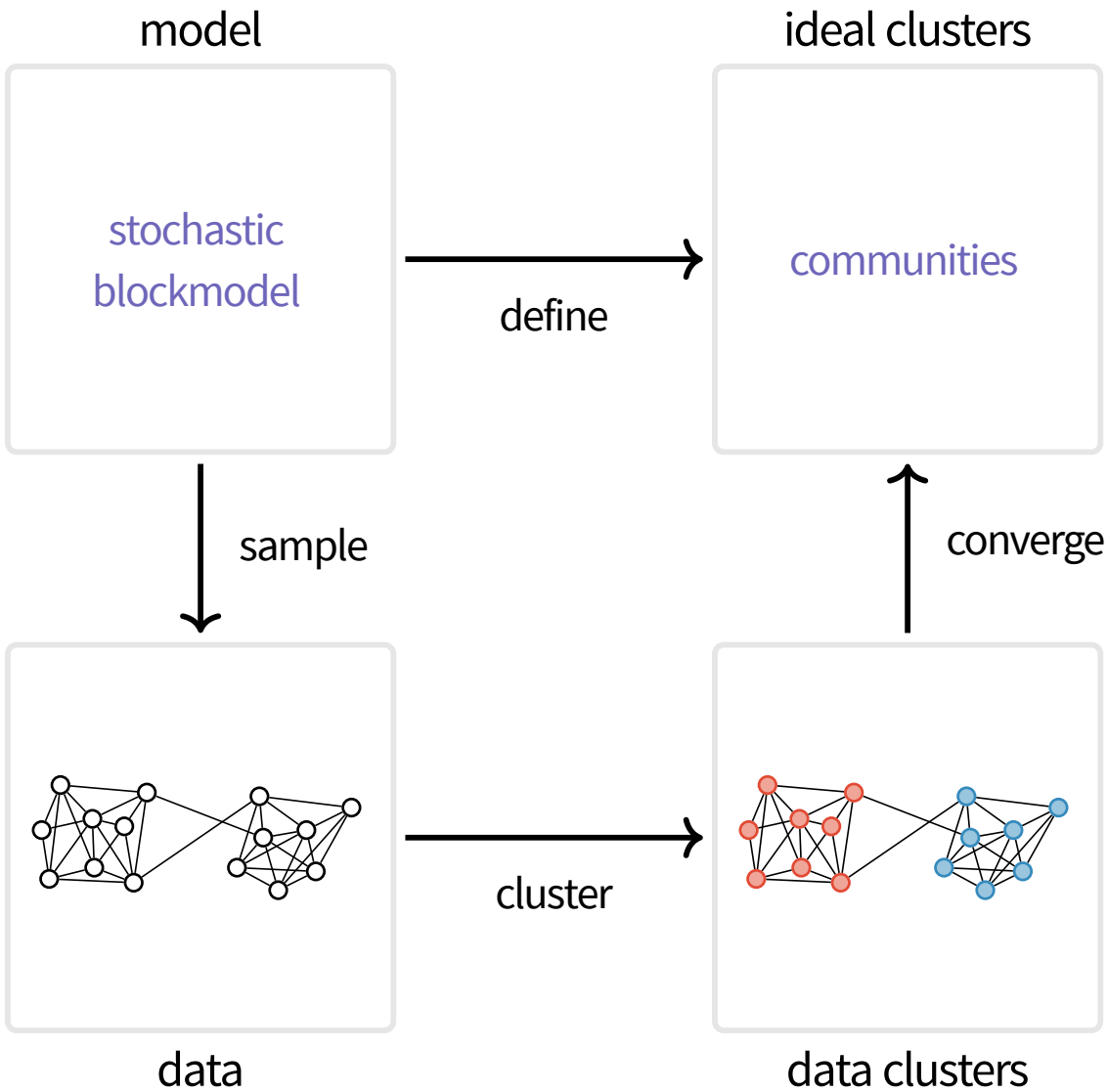






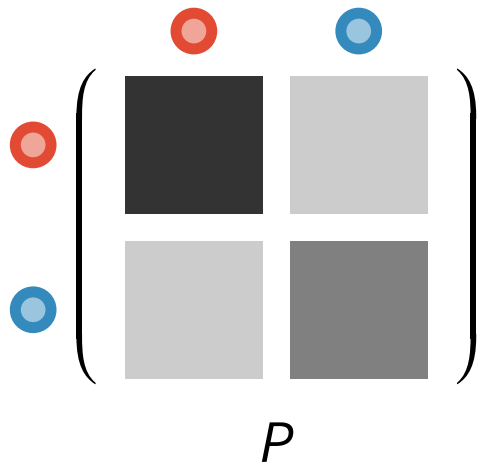
Example goal: recover communities in a social network.


model
ideal clusters
?
?
define
sample
converge
cluster
data
data clusters


model
ideal clusters
stochastic blockmodel
define
communities
sample
converge
cluster
data
data clusters

Background: the stochastic blockmodel.

- Each graph node belongs to one of k blocks, or communities.
- Edge probabilities parameterized by symmetric $k \times k$ matrix P:
  - Prob. of edge between communities i and j given by P_{ij}.
- Example: 2-block model.
  - Social network of girls and boys at a school.

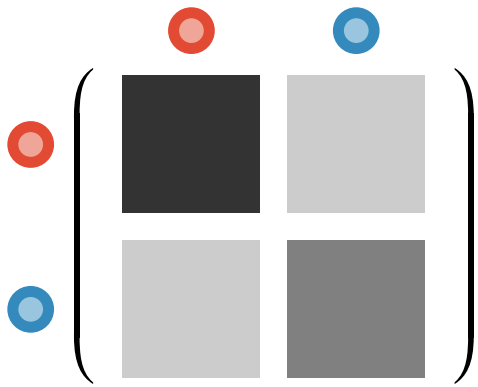


P

Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

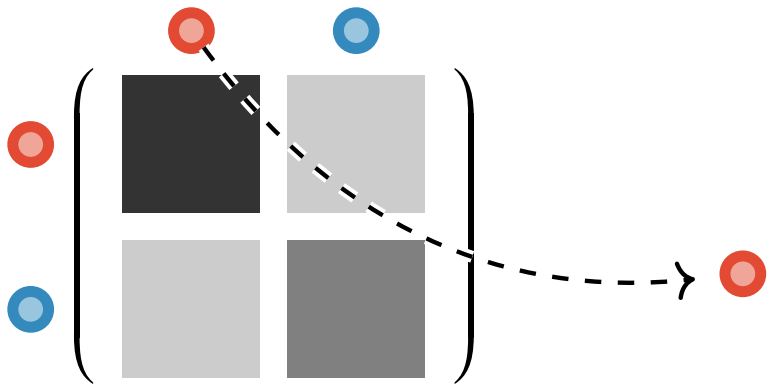
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

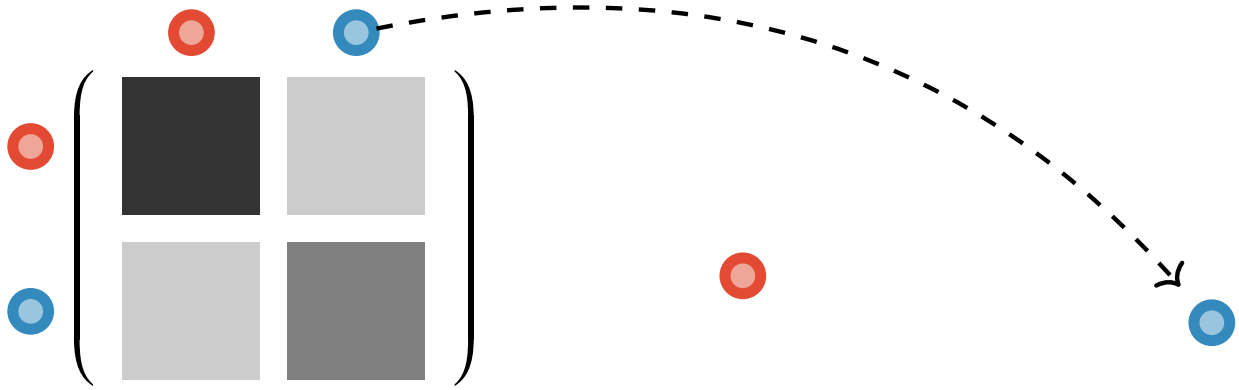
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

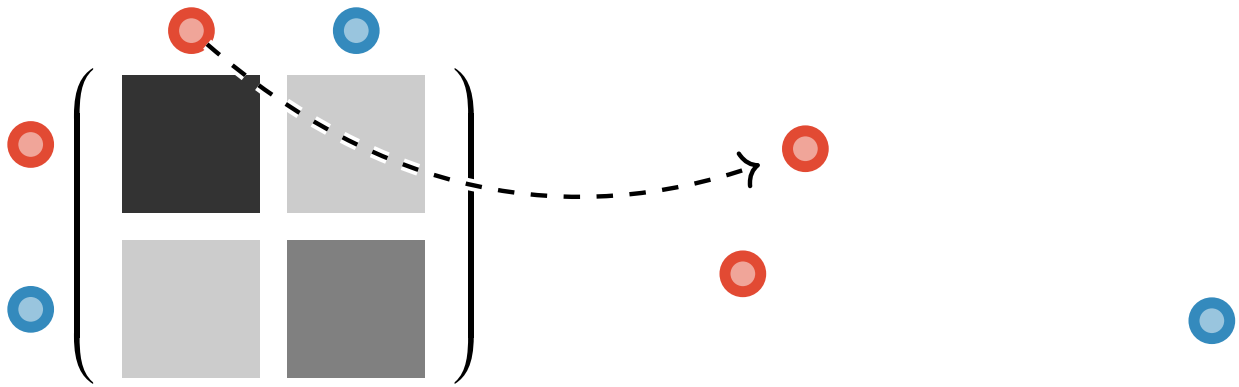
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

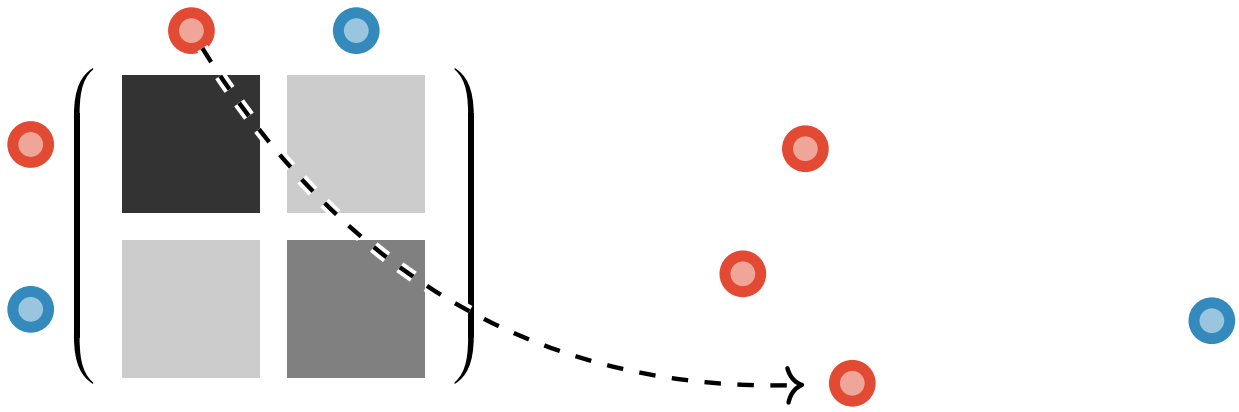
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

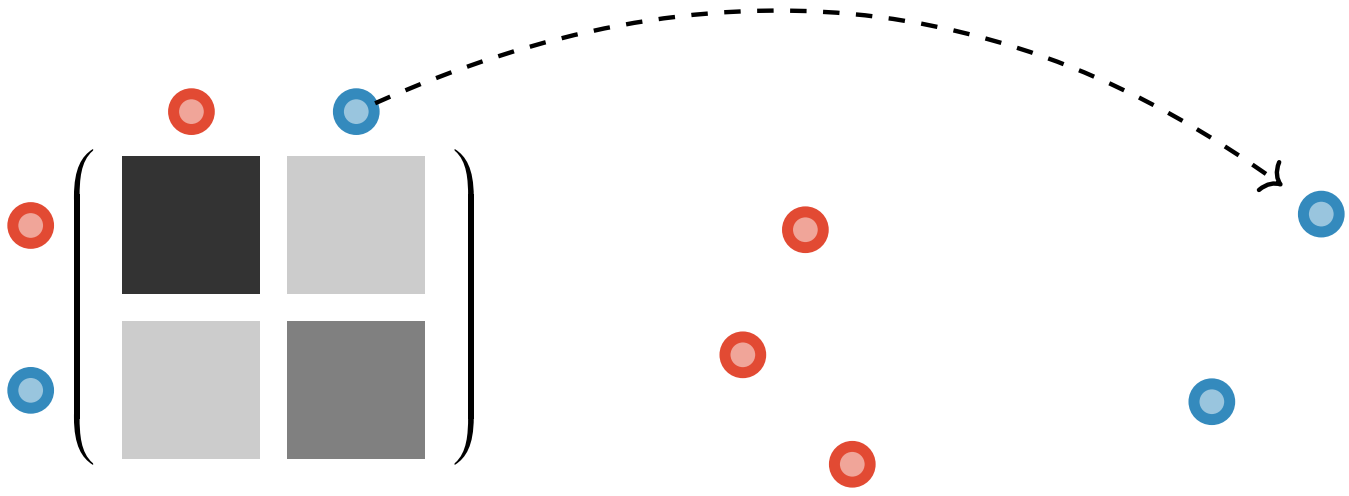
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

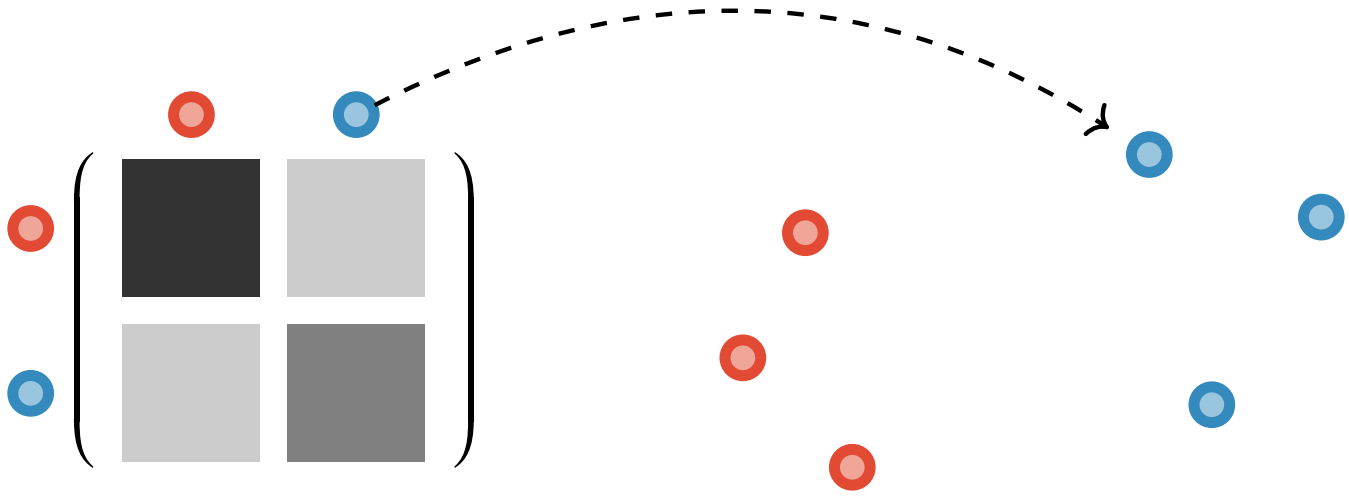
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

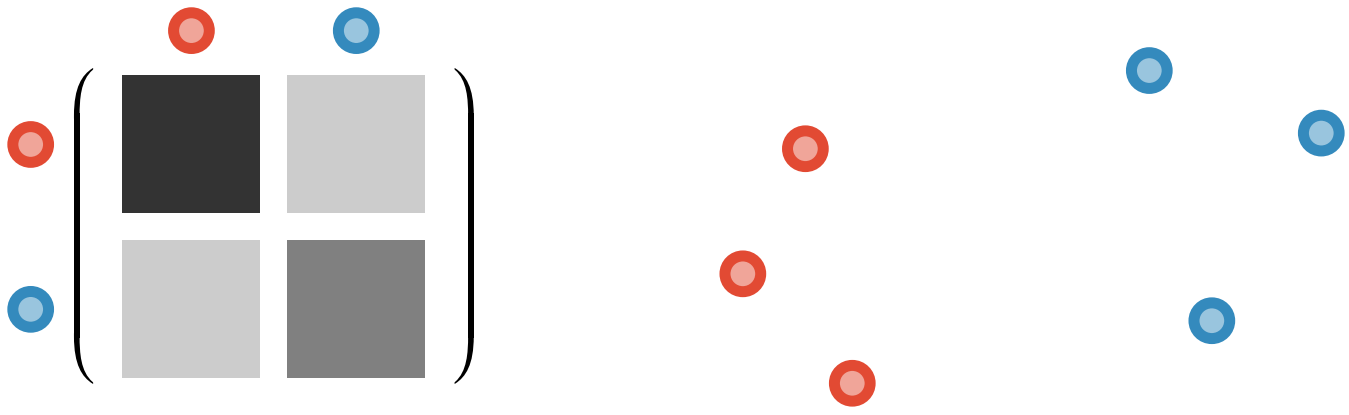
1. Sample communities uniformly with replacement.



Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

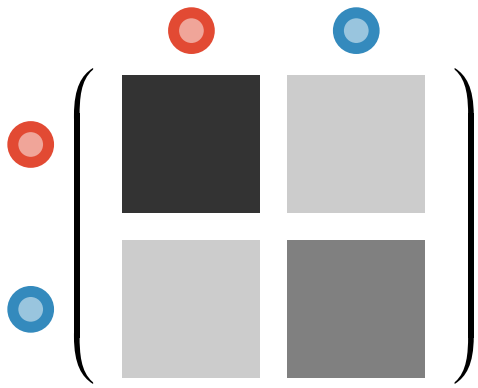
1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.

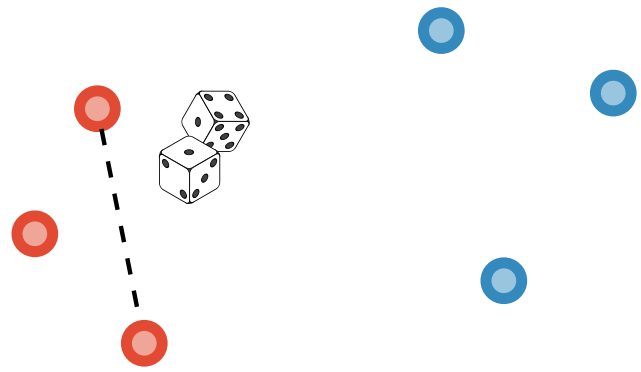


Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



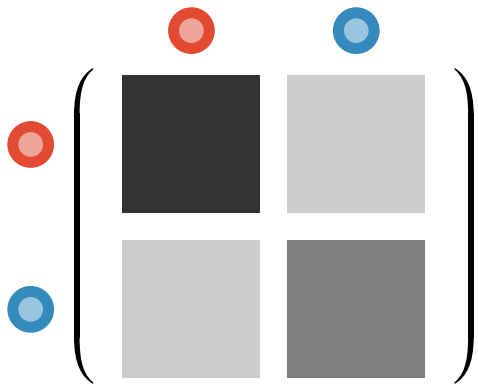


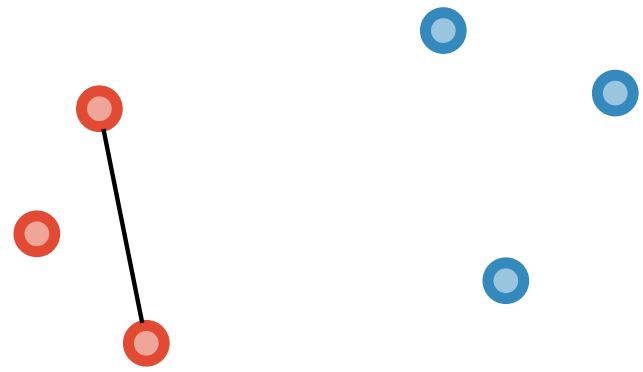
Add edge  with probability $P_{\bullet\bullet}$.

Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



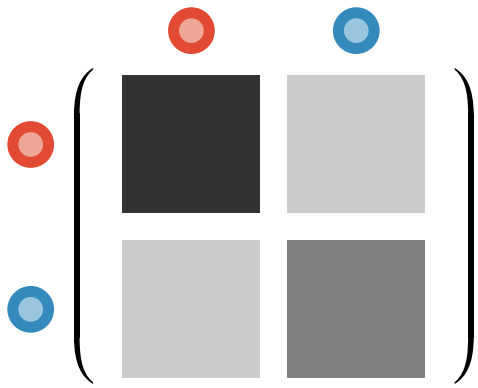


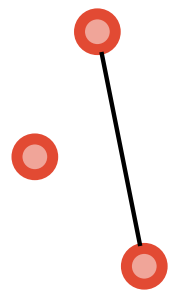
Add edge  with probability $P_{\circ\circ}$.

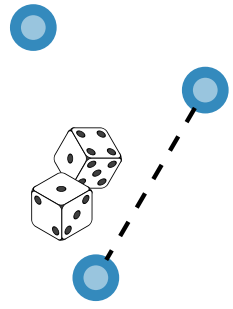
Sampling from a blockmodel.

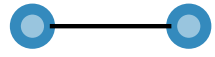
We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.





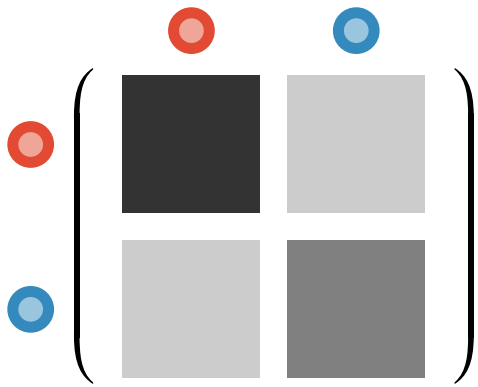


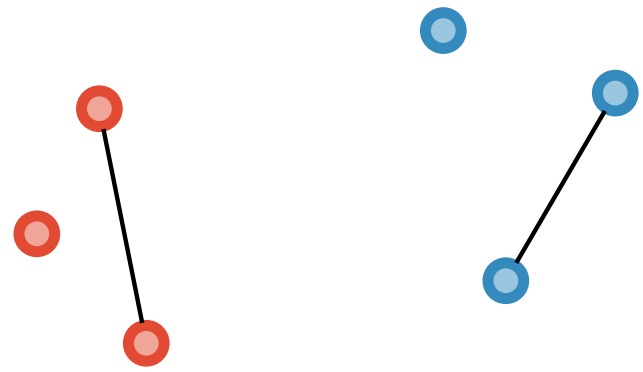
Add edge  with probability $P_{\circ\circ}$.

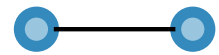
Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



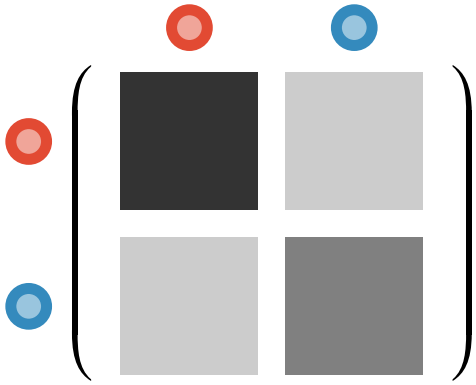


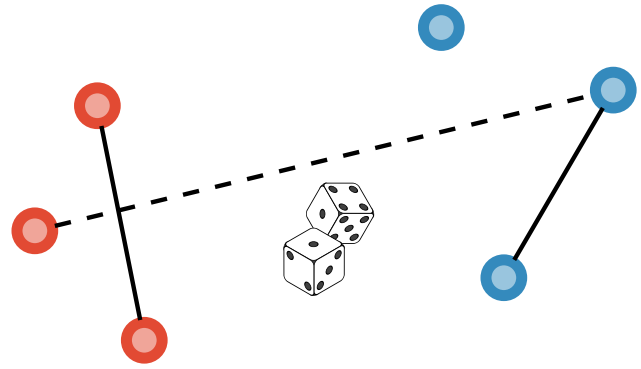
Add edge  with probability $P_{\circ\circ}$.

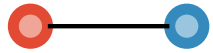
Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



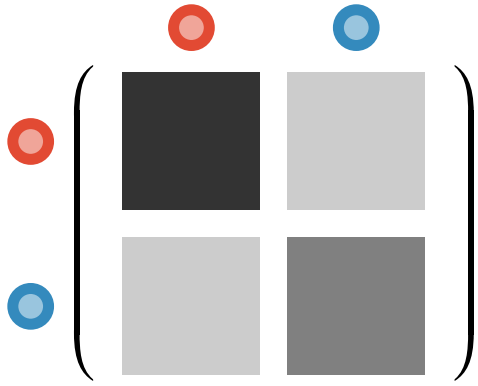


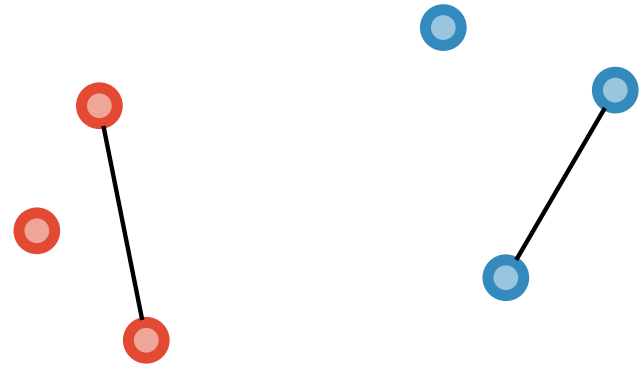
Add edge  with probability $P_{\color{red}\bullet\color{blue}\bullet}$.

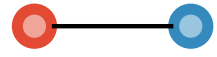
Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



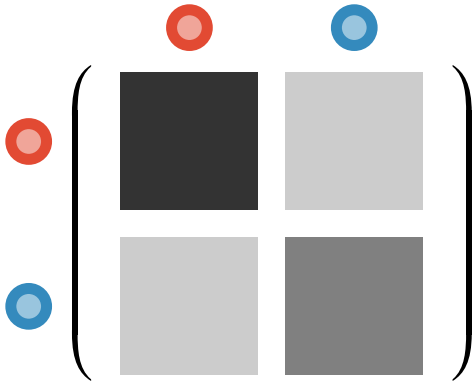


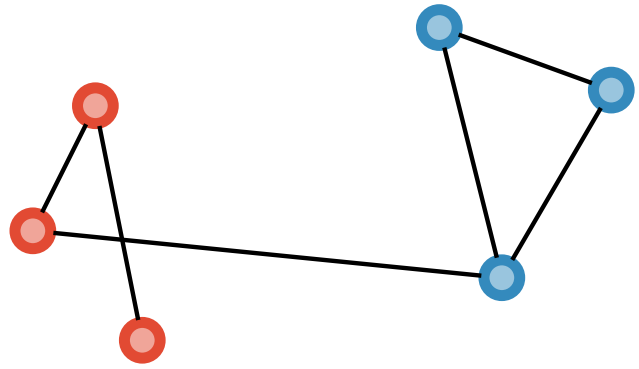
Add edge 
with probability $P_{\bullet\bullet}$.

Sampling from a blockmodel.

We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.



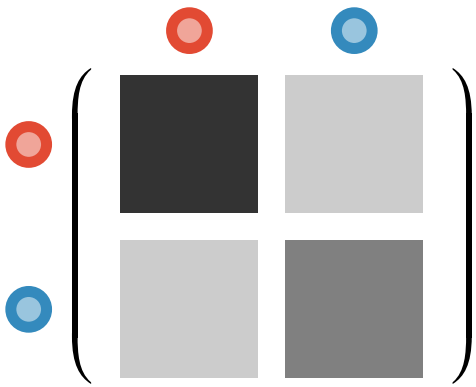


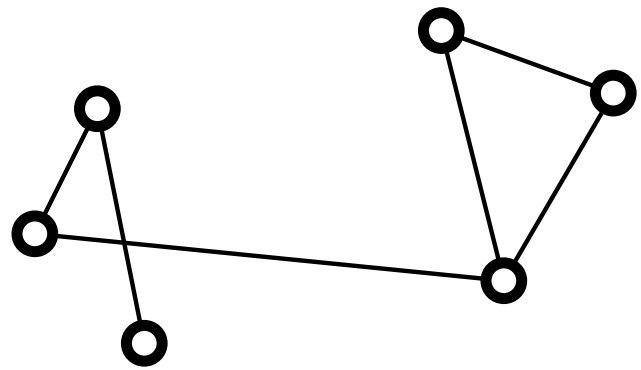
Repeat for all pairs of nodes.

Sampling from a blockmodel.

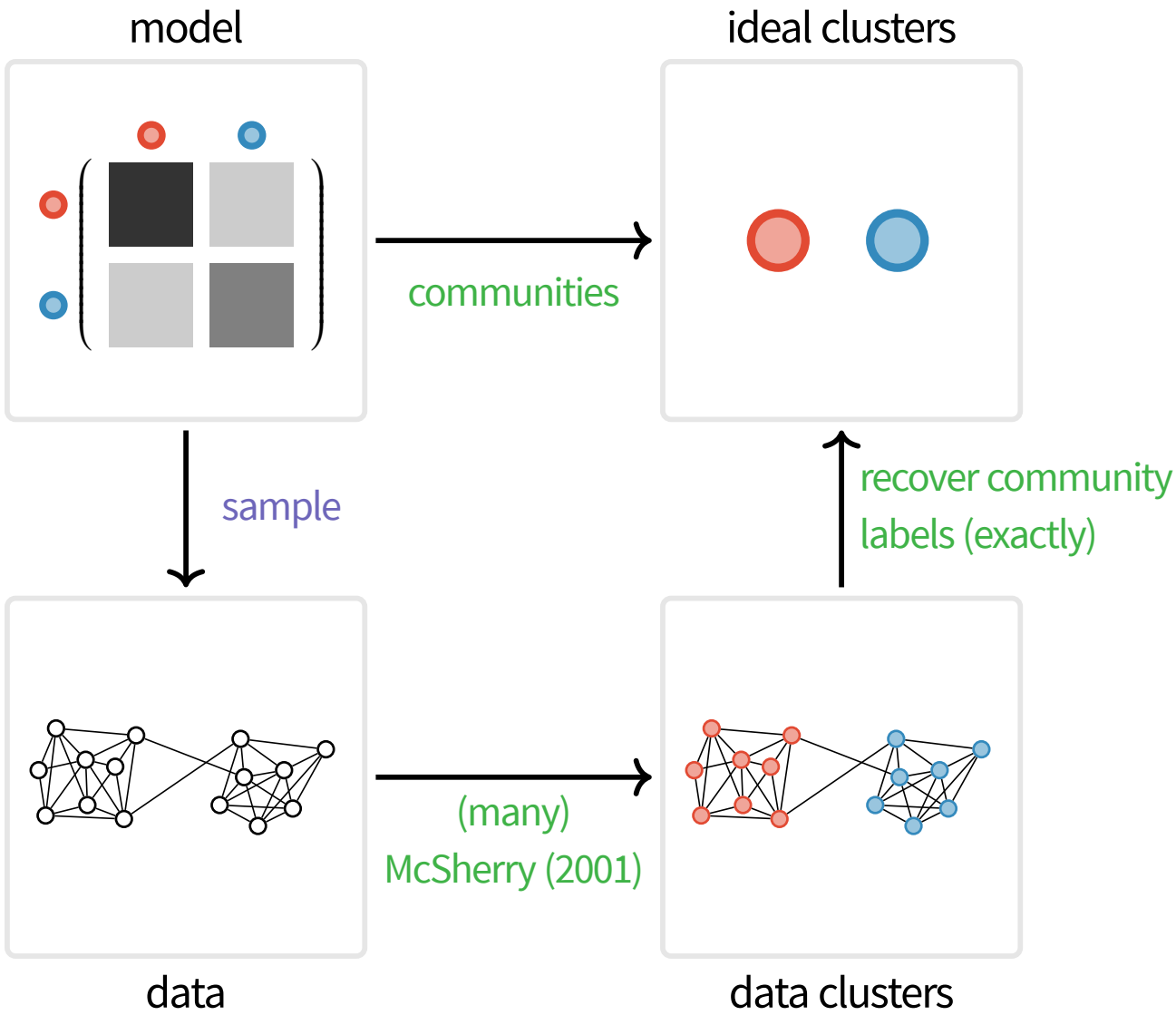
We can generate a random graph with n nodes from P as follows...

1. Sample communities uniformly with replacement.
2. Sample edges with probability according to P.
3. Forget community labels.

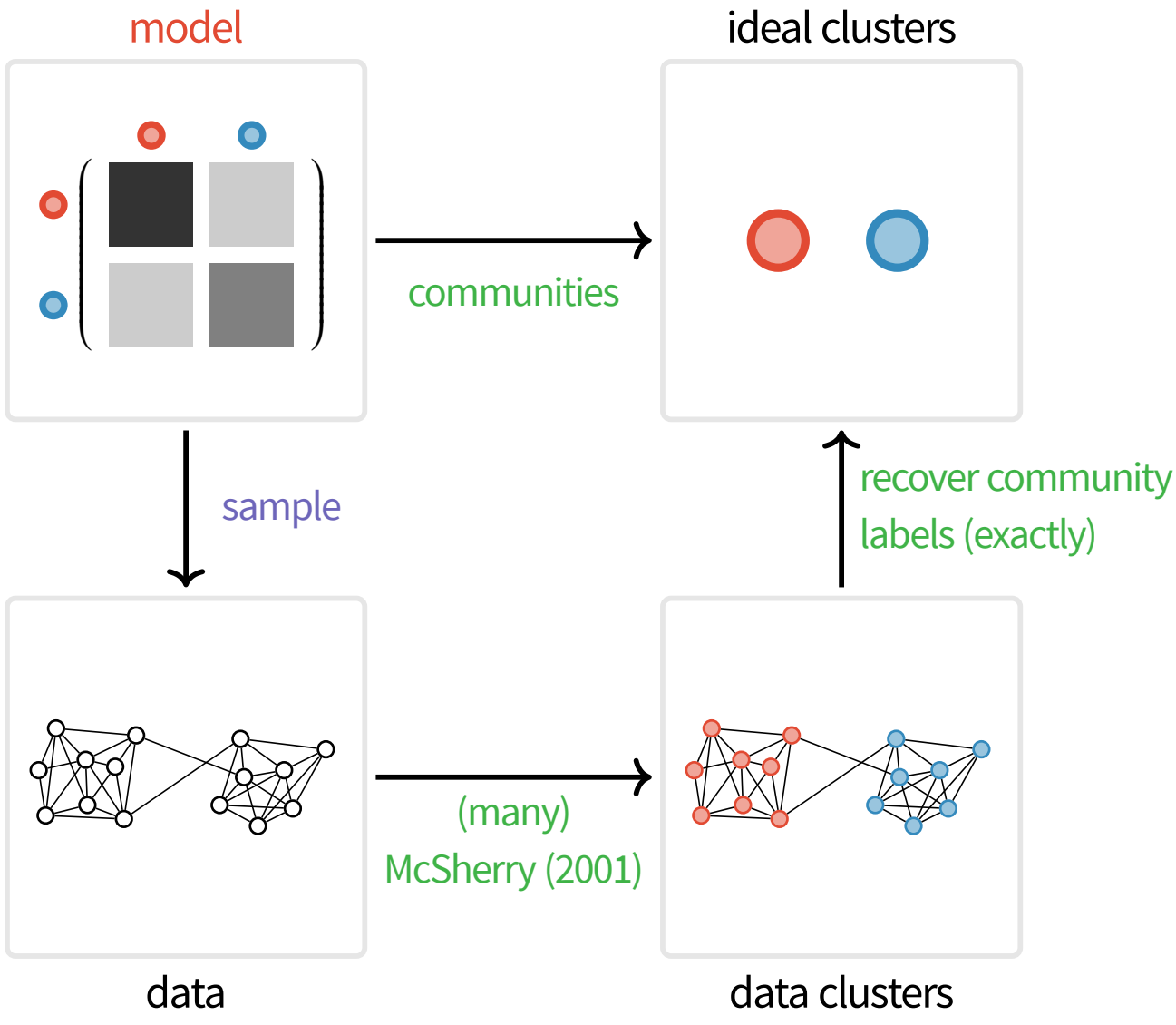




Clustering in the stochastic blockmodel.



Clustering in the stochastic blockmodel.



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

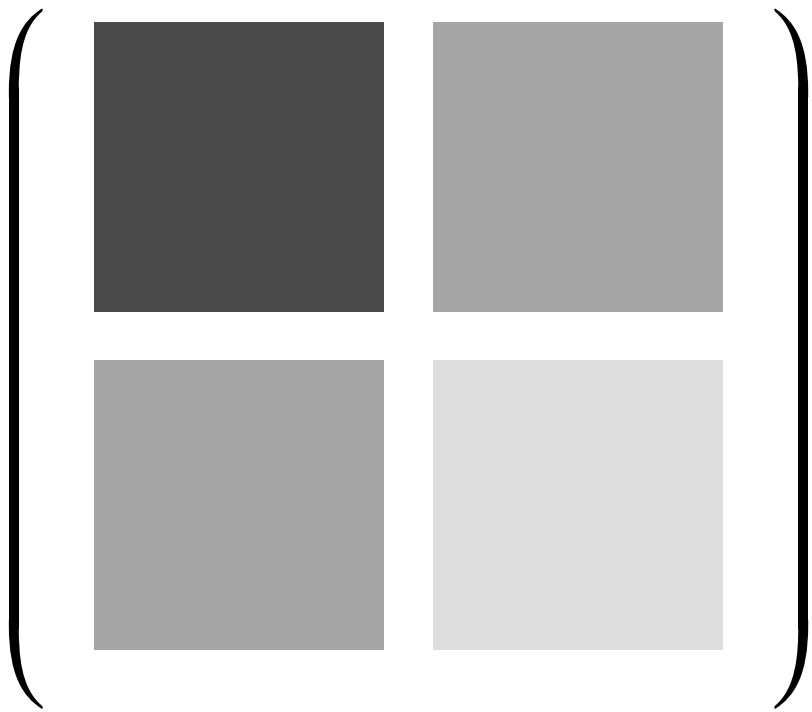
Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

Applications & Directions

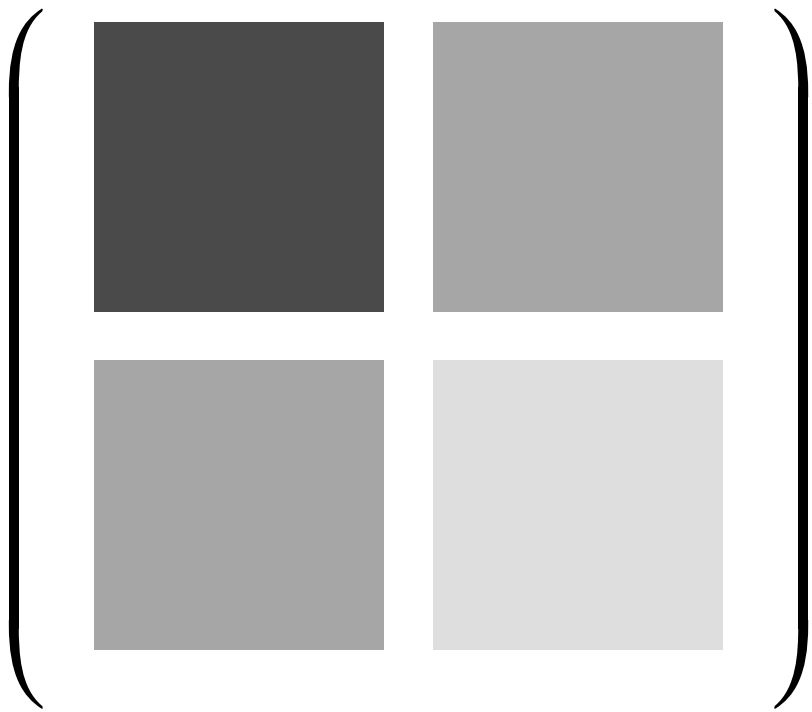
Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.



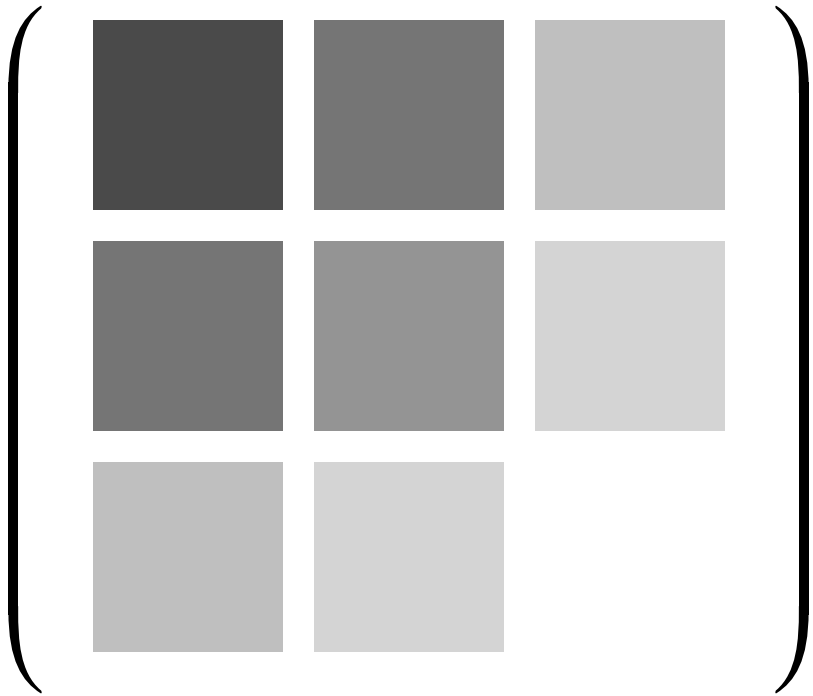
Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.
- Solution: Increase number of parameters, i.e., communities...



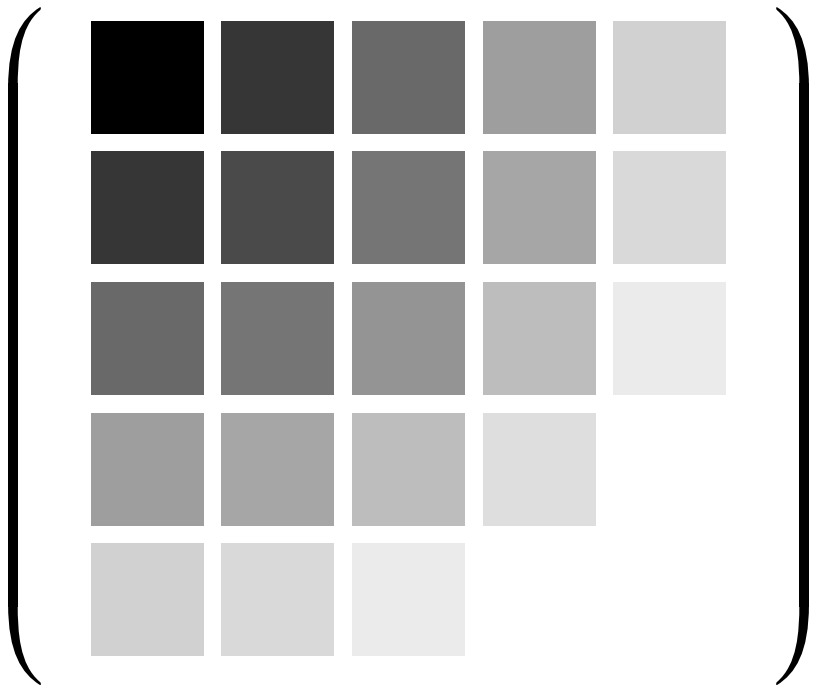
Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.
- Solution: Increase number of parameters, i.e., communities...



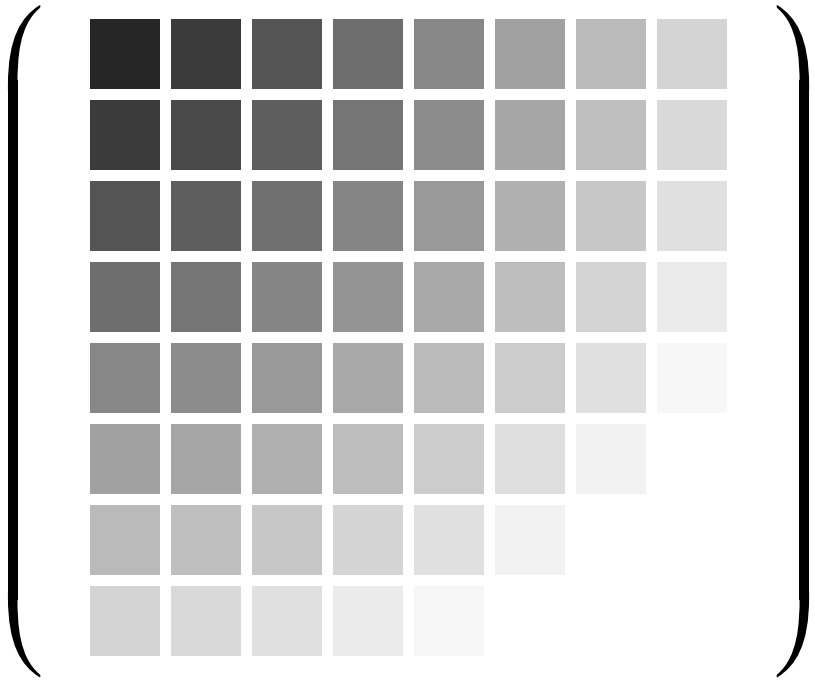
Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.
- Solution: Increase number of parameters, i.e., communities...



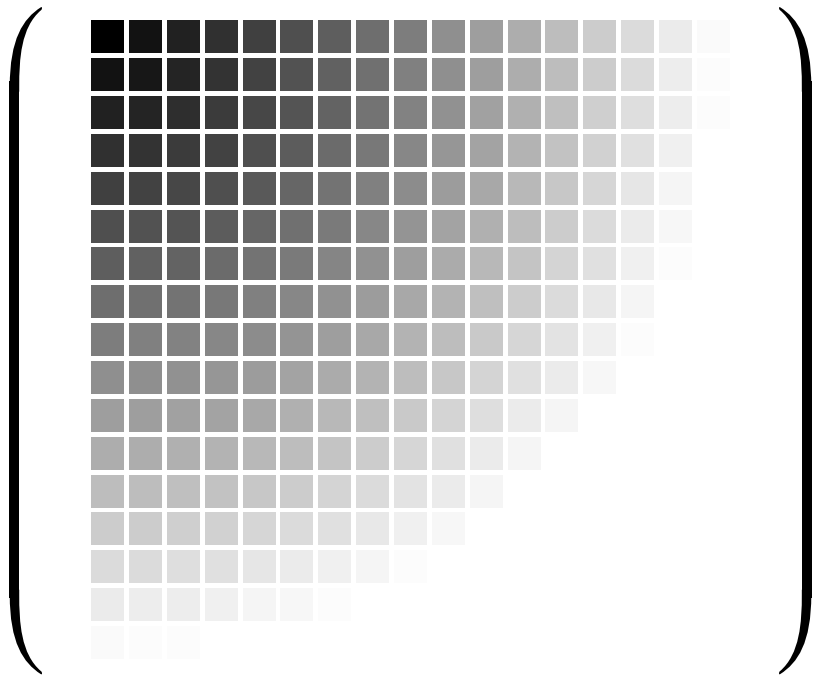
Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.
- Solution: Increase number of parameters, i.e., communities...

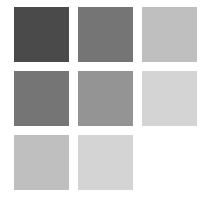
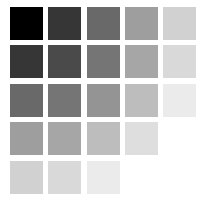
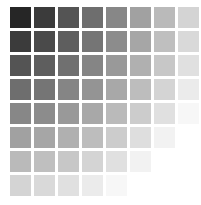
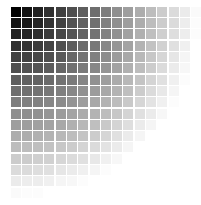


Problem: Many real-world networks not well-fit by blockmodel.

- Large networks (Facebook, LinkedIn, etc.) are complicated.
- The 2-blockmodel is very simple.
- Solution: Increase number of parameters, i.e., communities...



The limit of a blockmodel is...

$$\lim_{k \to \infty} \left(\text{  } \right), \left(\text{  } \right), \left(\text{  } \right), \left(\text{  } \right), \ldots$$

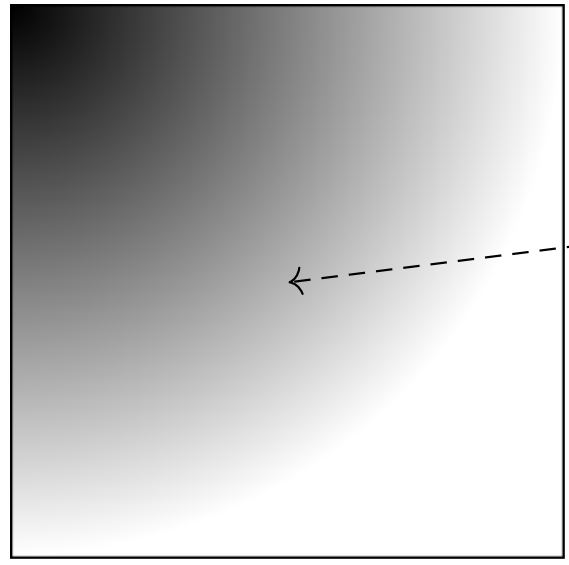
$$=$$

?

The limit of a blockmodel is...

$$\lim_{k\to\infty} \left(\quad\right), \left(\quad\right), \left(\quad\right), \left(\quad\right), \ldots$$

$$=$$



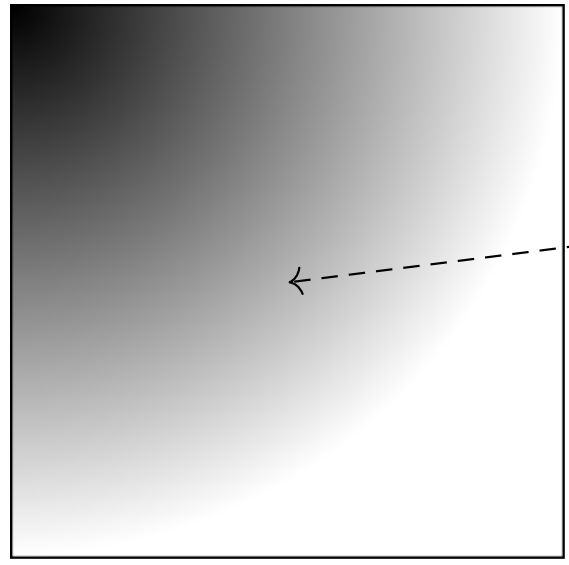
...a graphon!

symmetric function $W : [0,1]^2 \to [0,1]$

The limit of a blockmodel is...

$$\lim_{k\to\infty}{}^{\dagger} \left(\right), \left(\right), \left(\right), \left(\right), \ldots$$

$$=$$

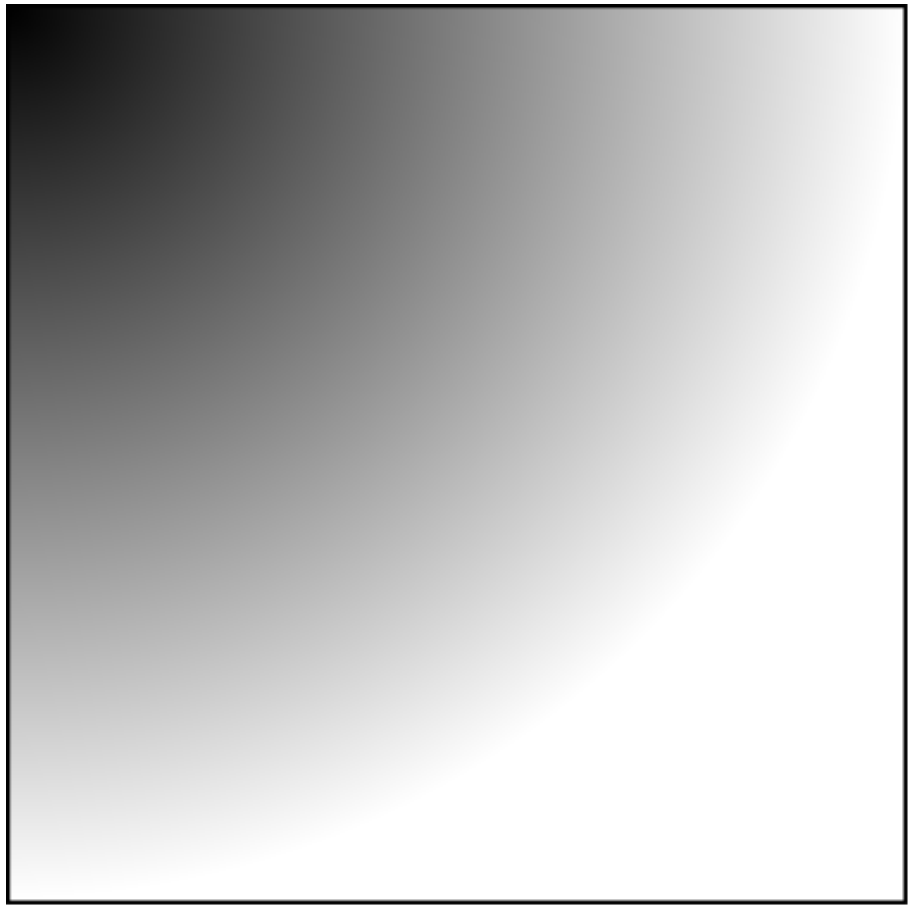


...a graphon!

symmetric function $W : [0,1]^2 \to [0,1]$

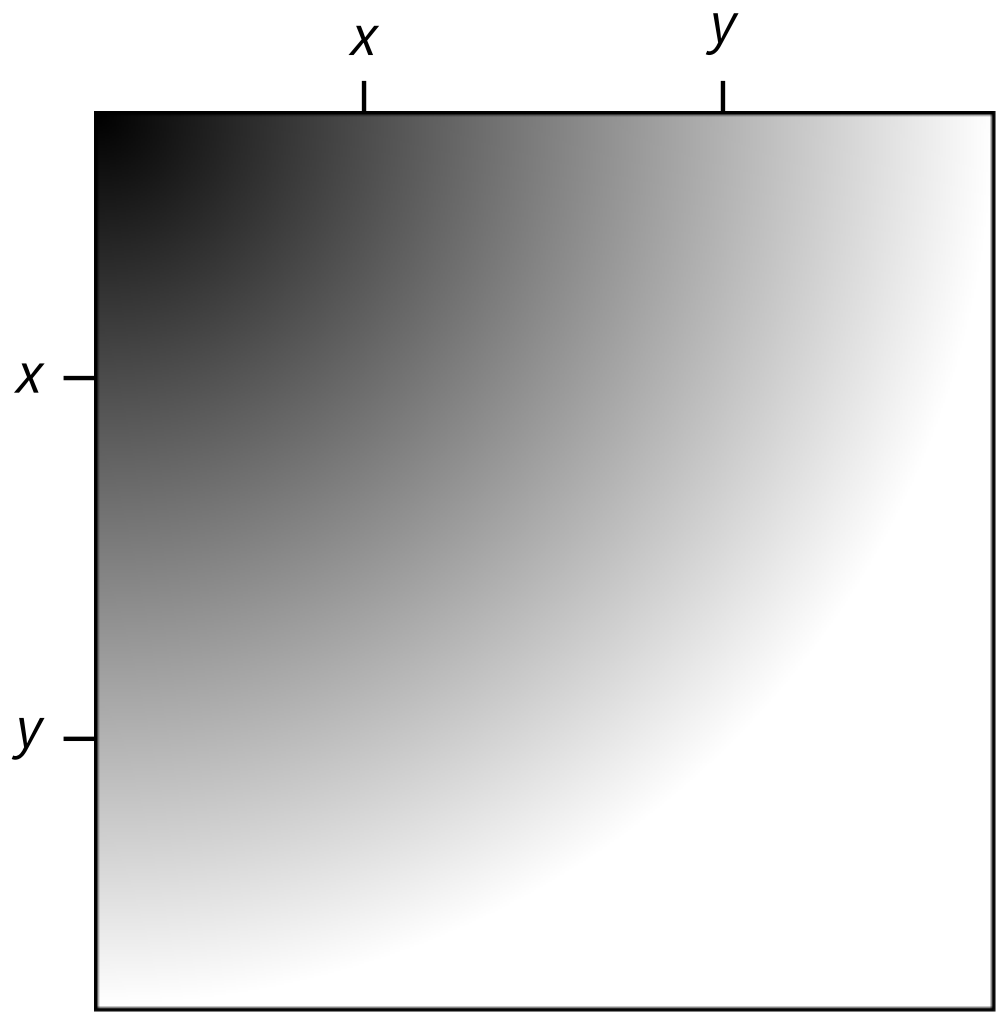
† Convergence in so-called cut metric, (Lovász, 2012).

Interpretation: The adjacency of an infinite weighted graph.



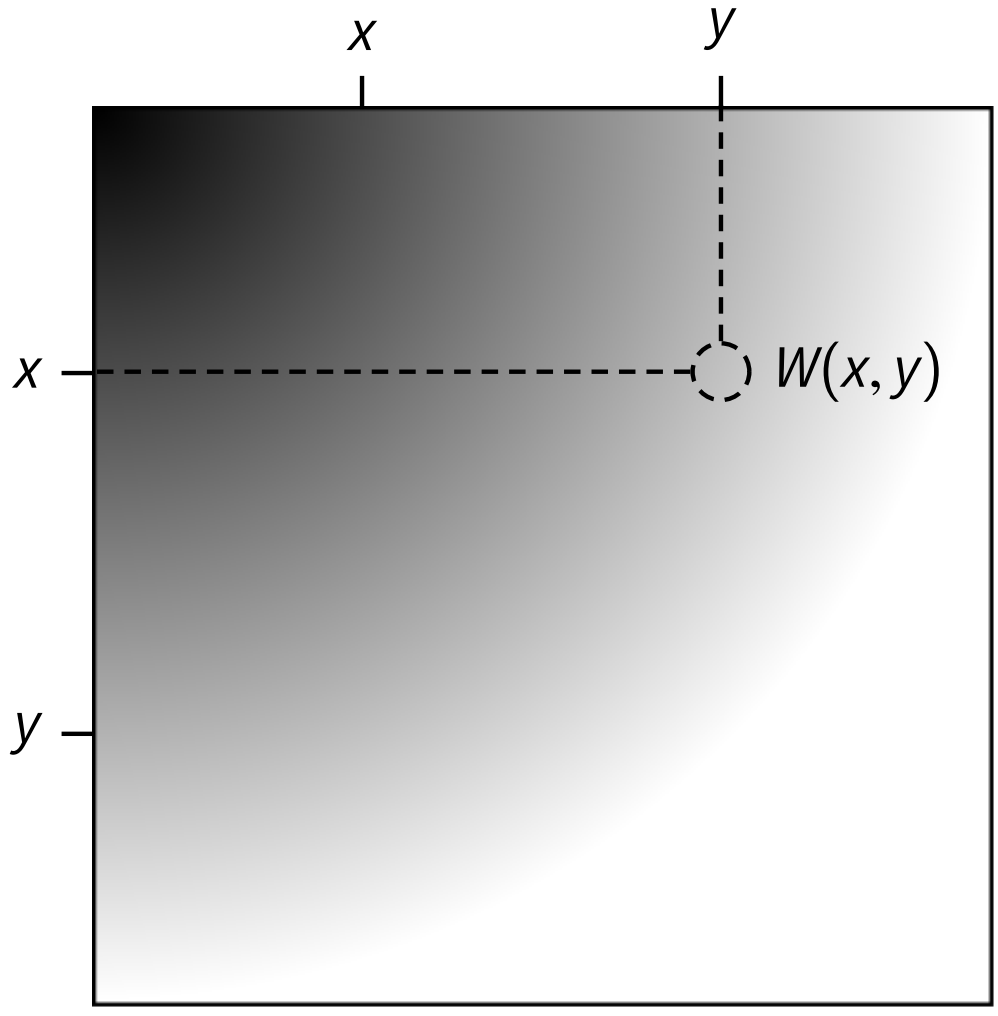
Interpretation: The adjacency of an infinite weighted graph.

Graphon "nodes" are points $x, y \in [0, 1]$.



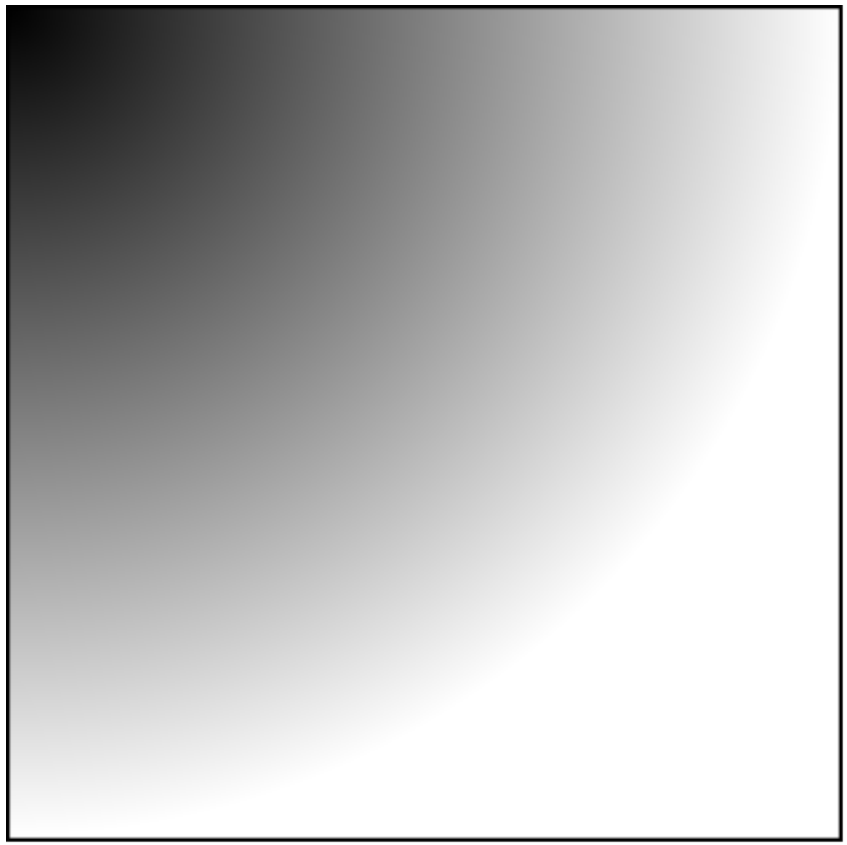
Interpretation: The adjacency of an infinite weighted graph.

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



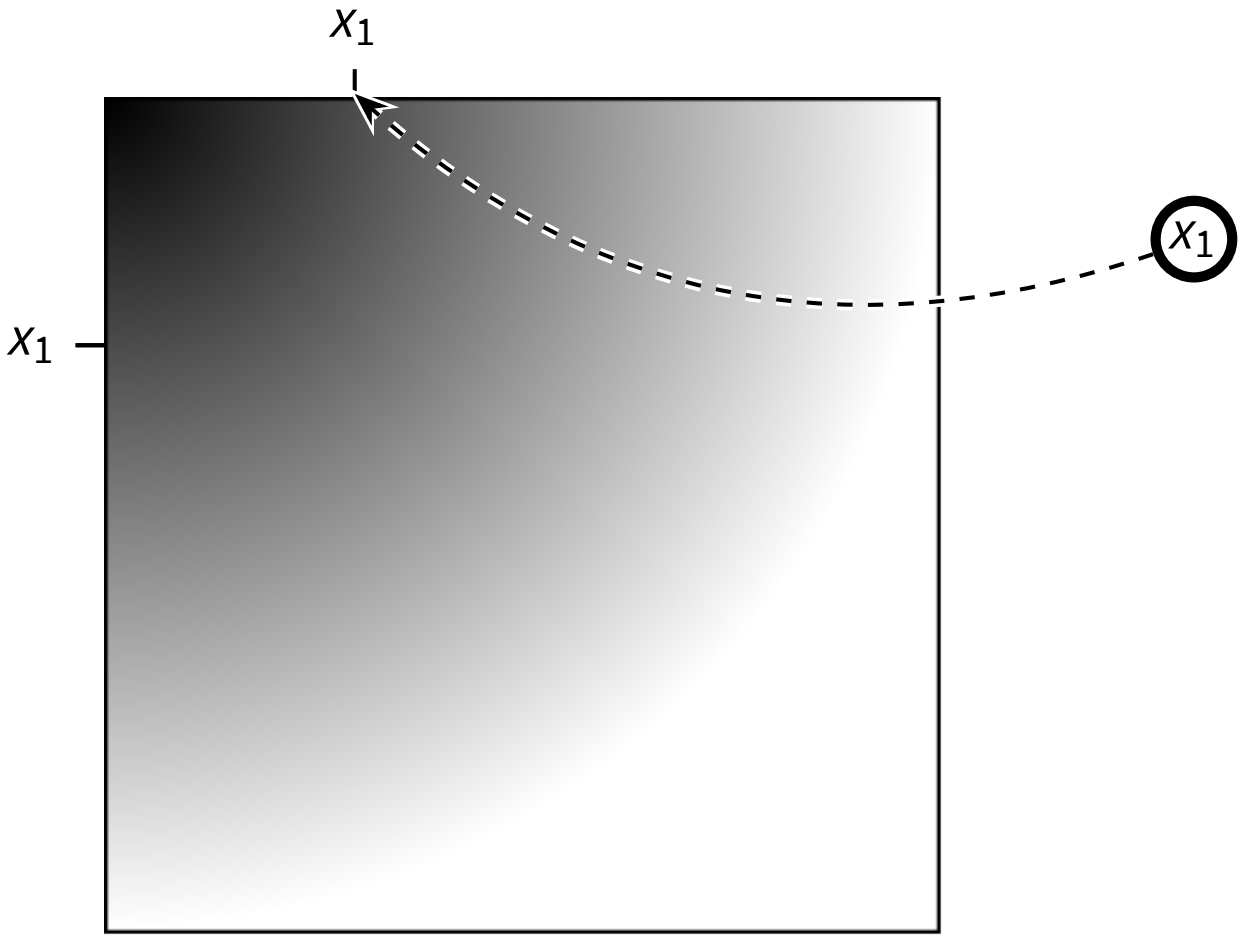
Sampling a graph from W.

Graphon sampling is analogous to sampling from a blockmodel.



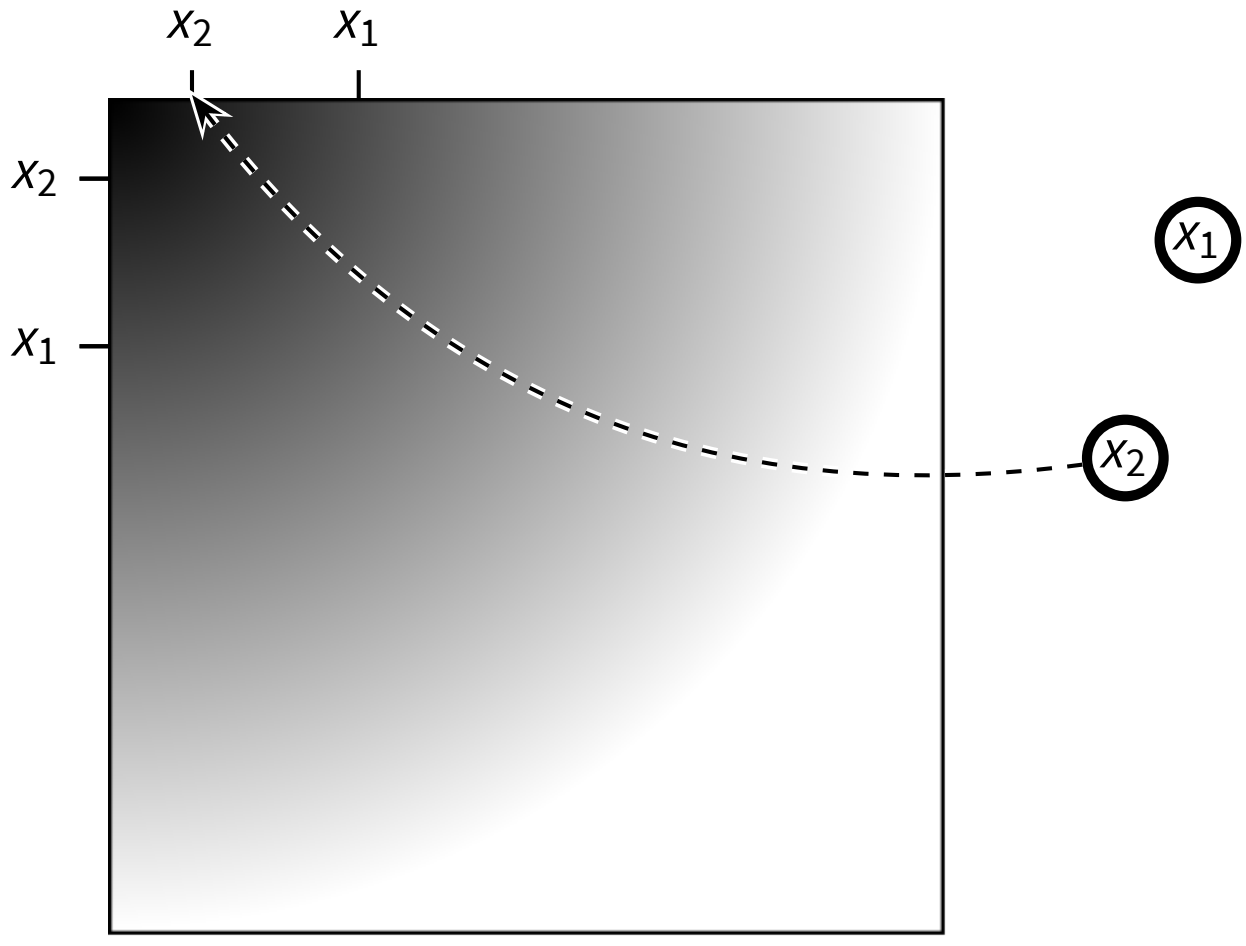
Sampling a graph from W.

First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



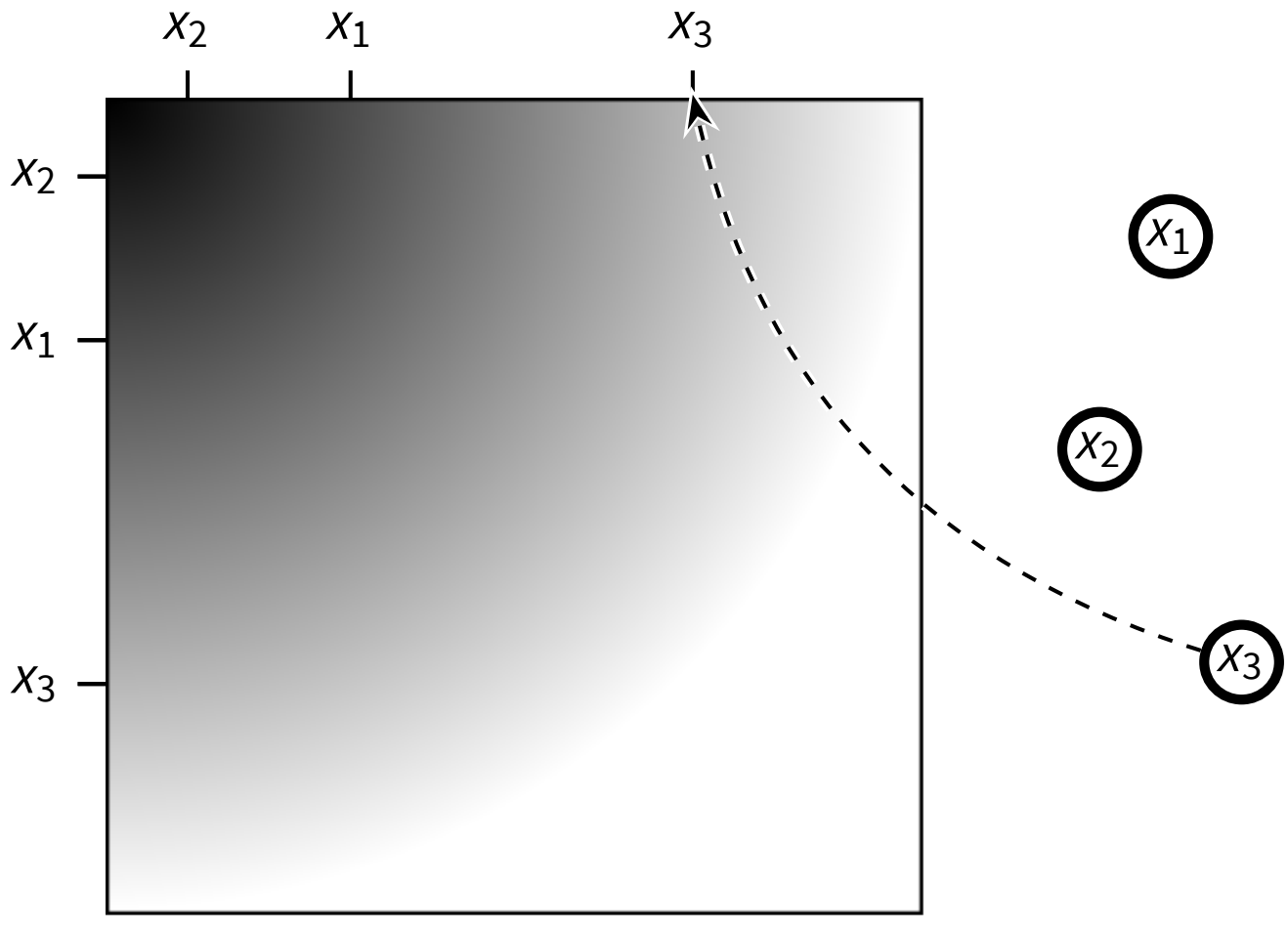
Sampling a graph from W.

First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



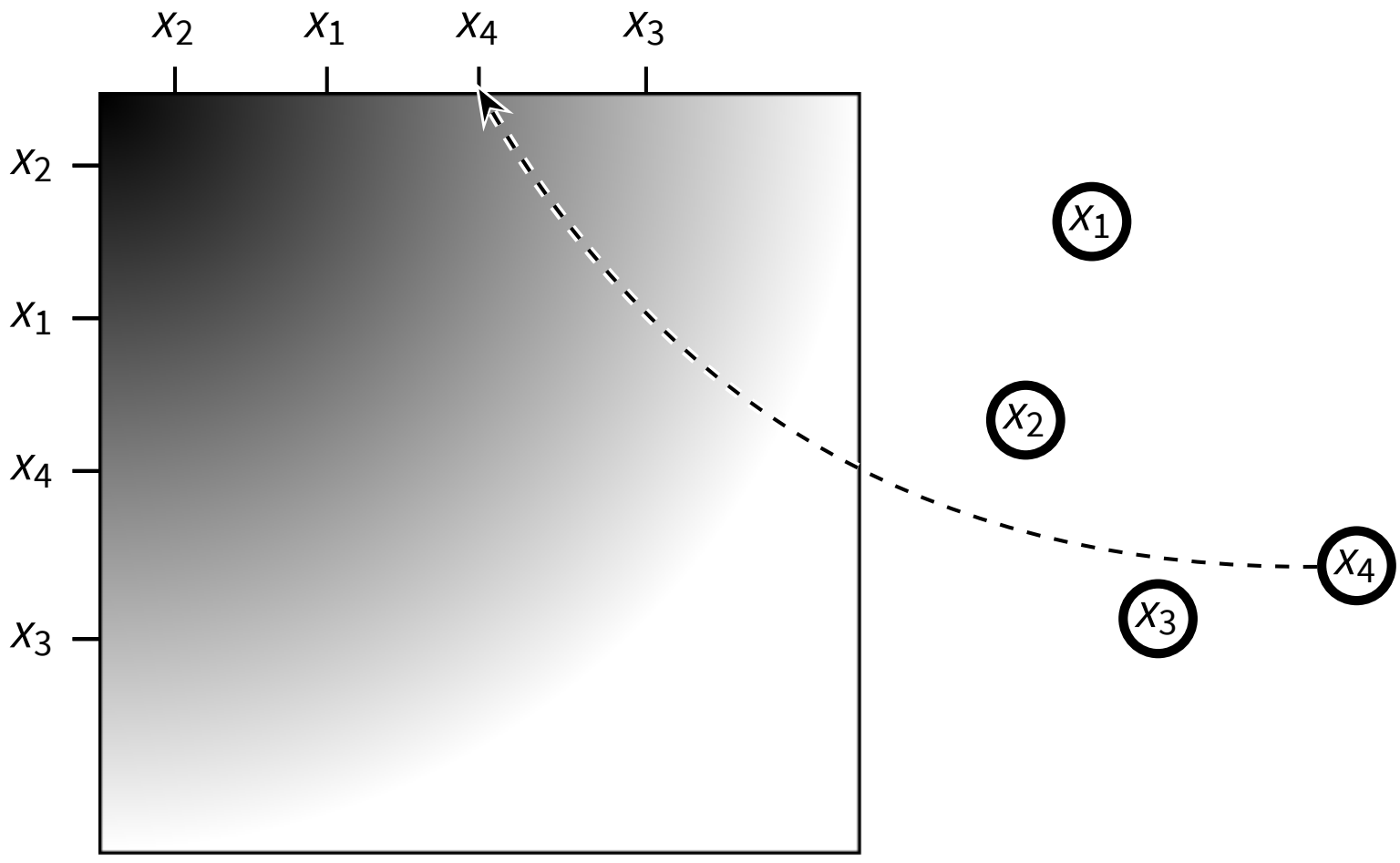
Sampling a graph from W.

First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



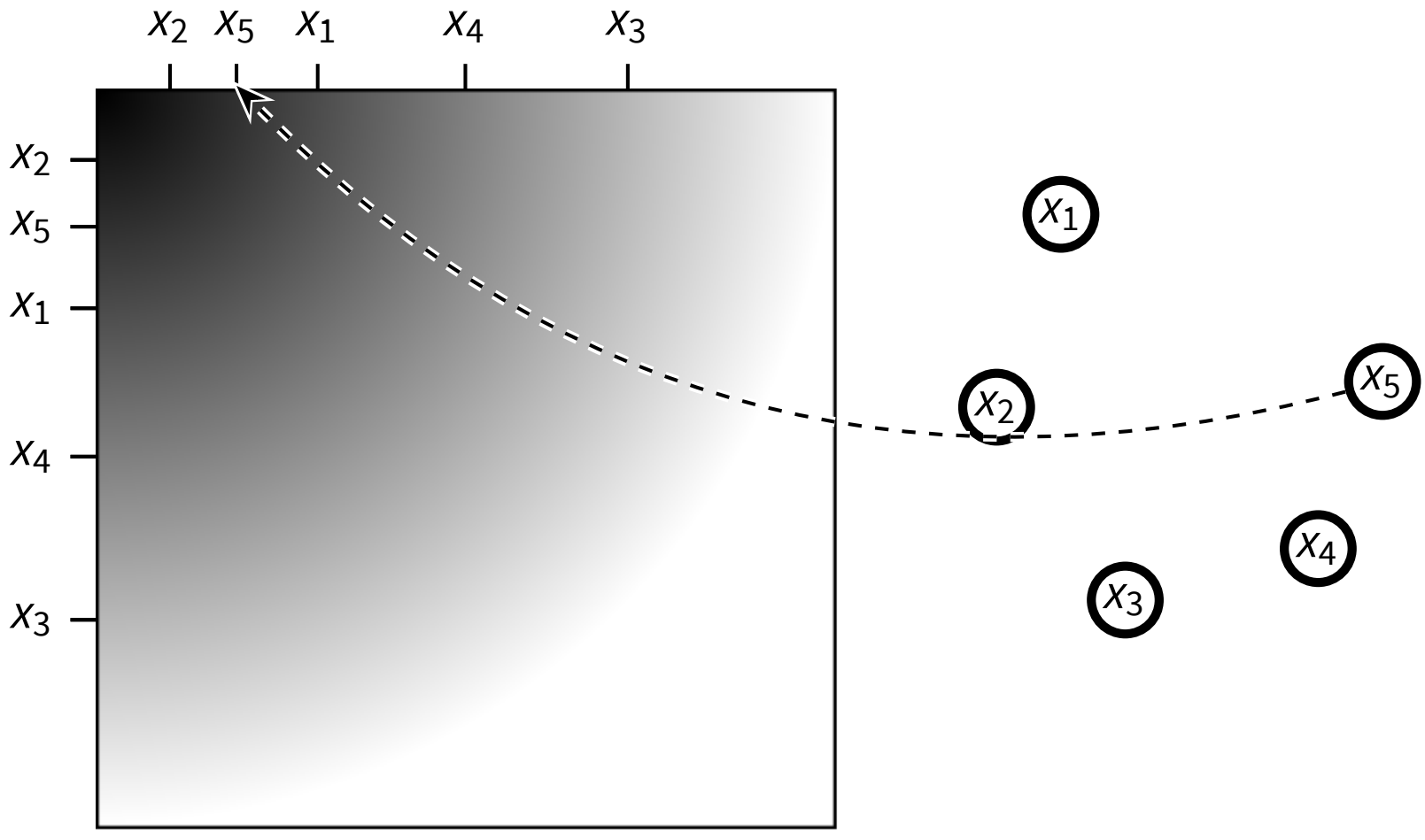
Sampling a graph from W.

First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



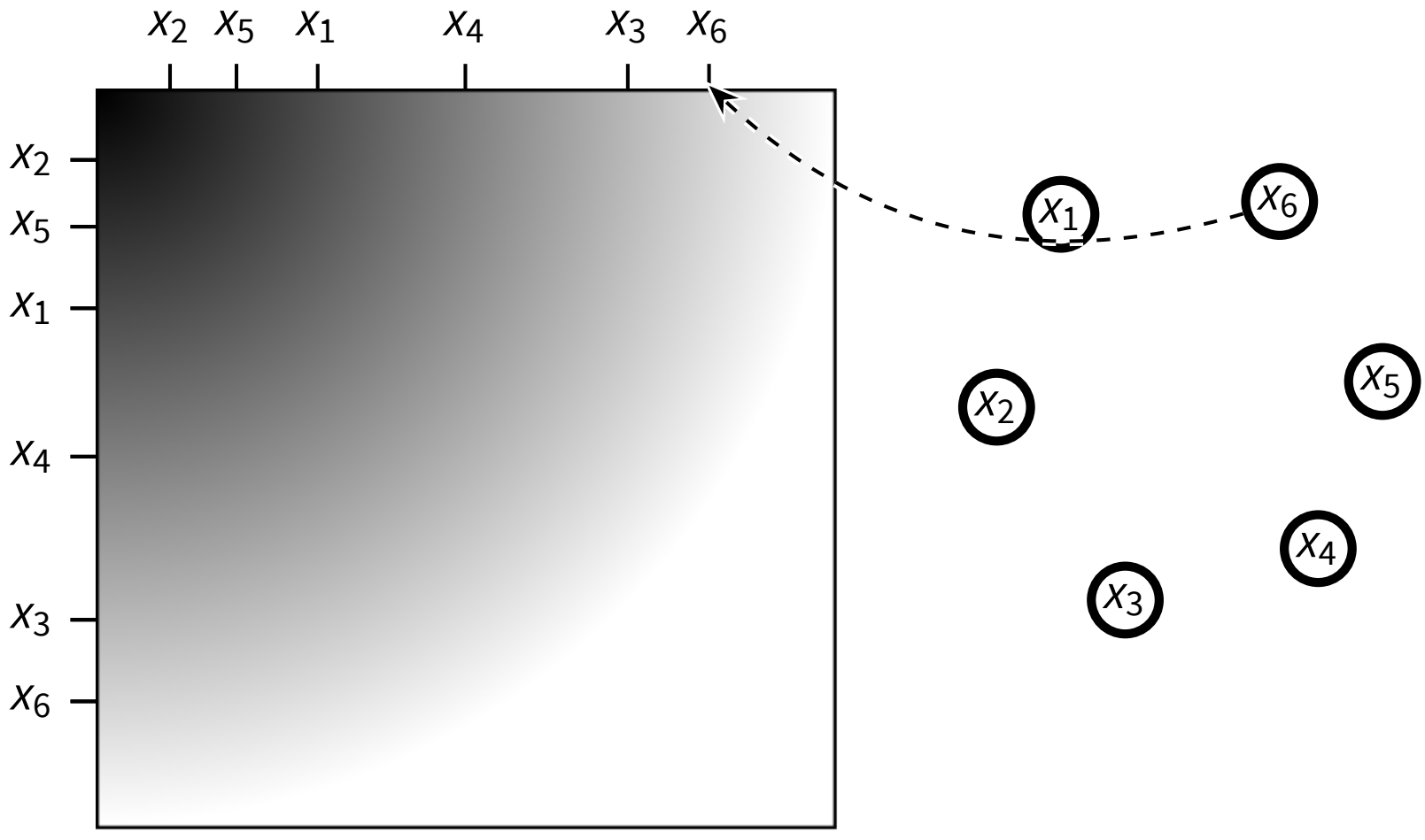
Sampling a graph from W.

First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



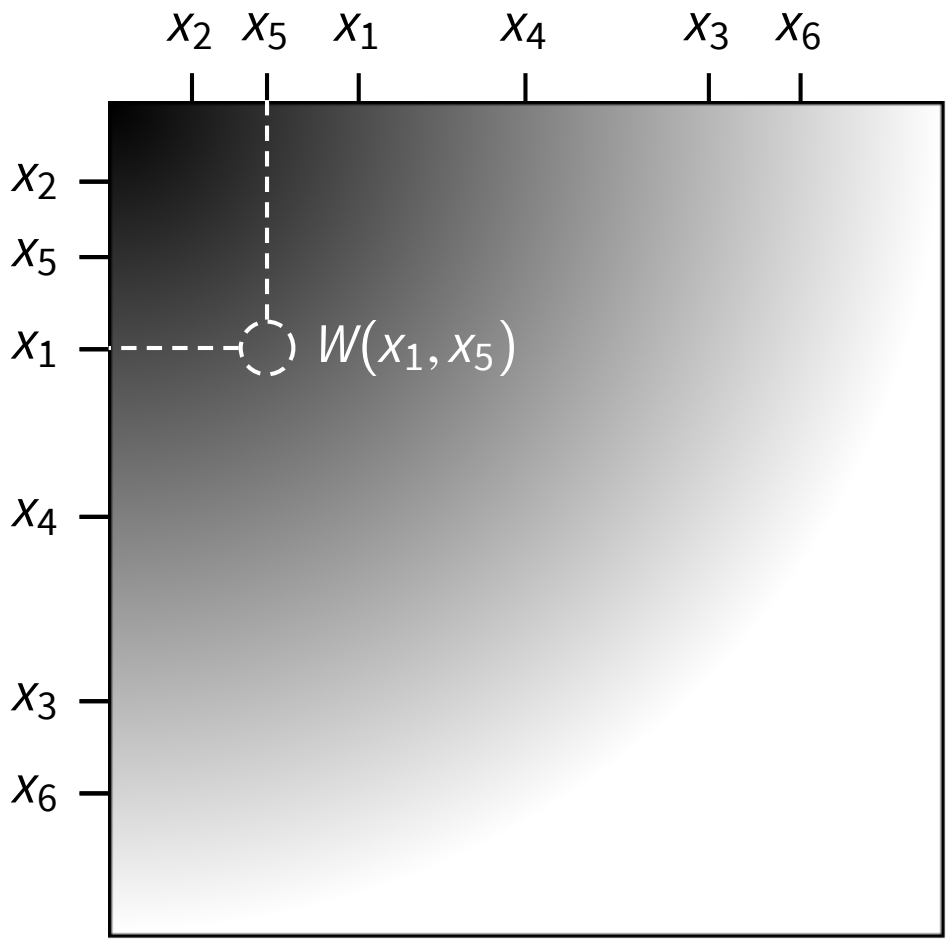
Sampling a graph from W.

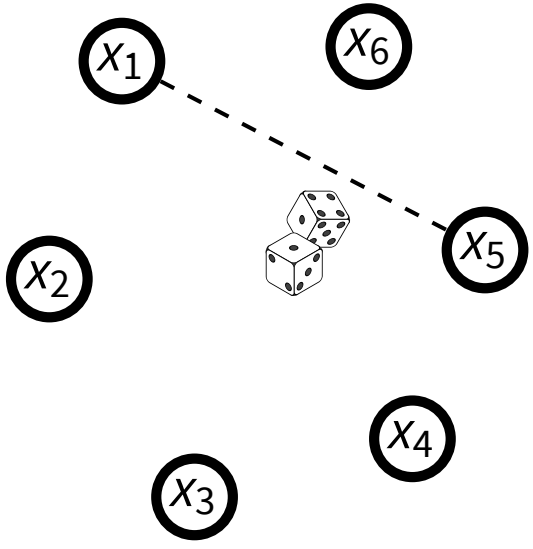
First, sample n graphon nodes, i.e., points from $\mathrm{Unif}[0, 1]$.



Sampling a graph from W.

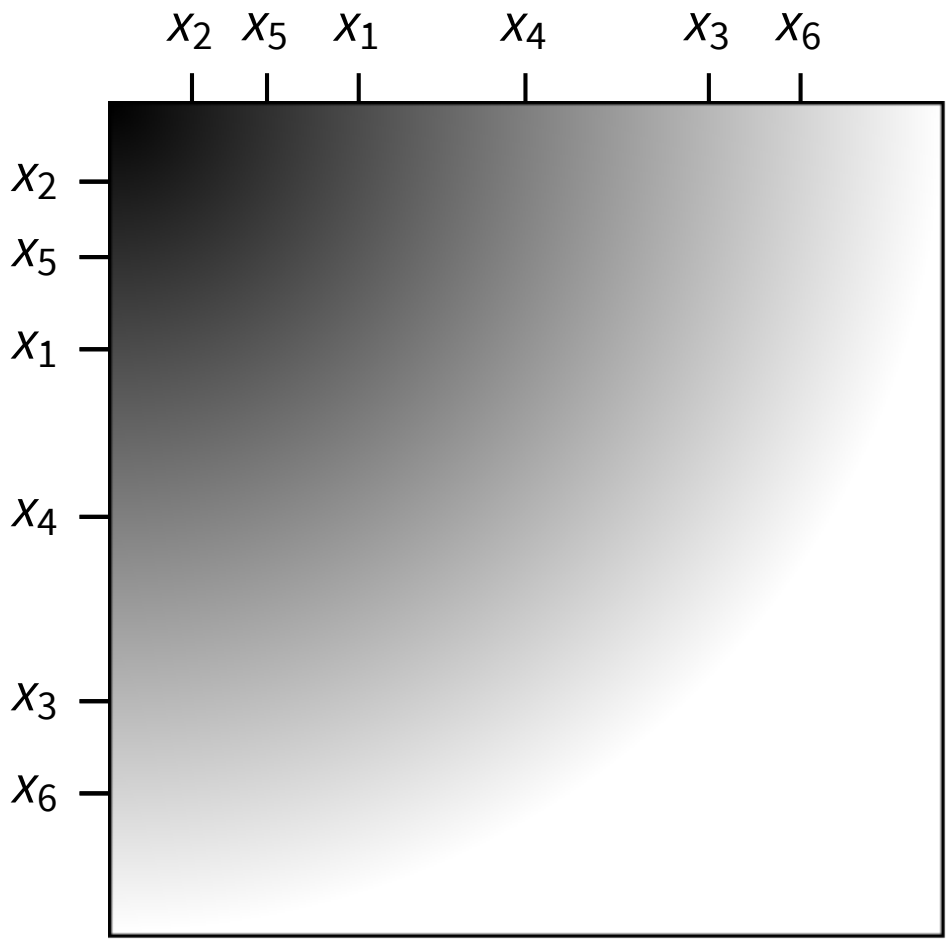
Include edge (x_1, x_5) with probability $W(x_1, x_5)$.

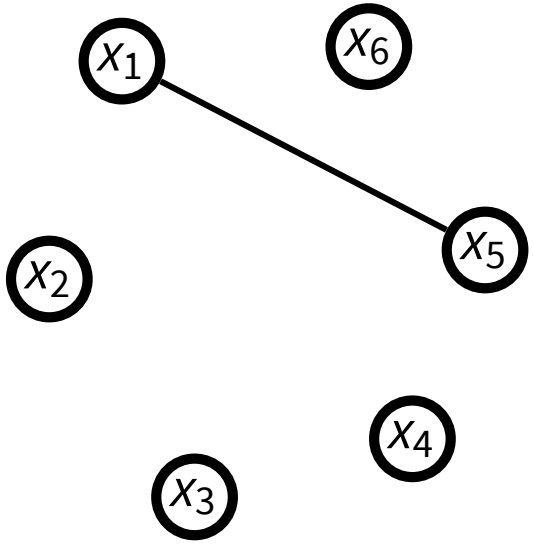




Sampling a graph from W.

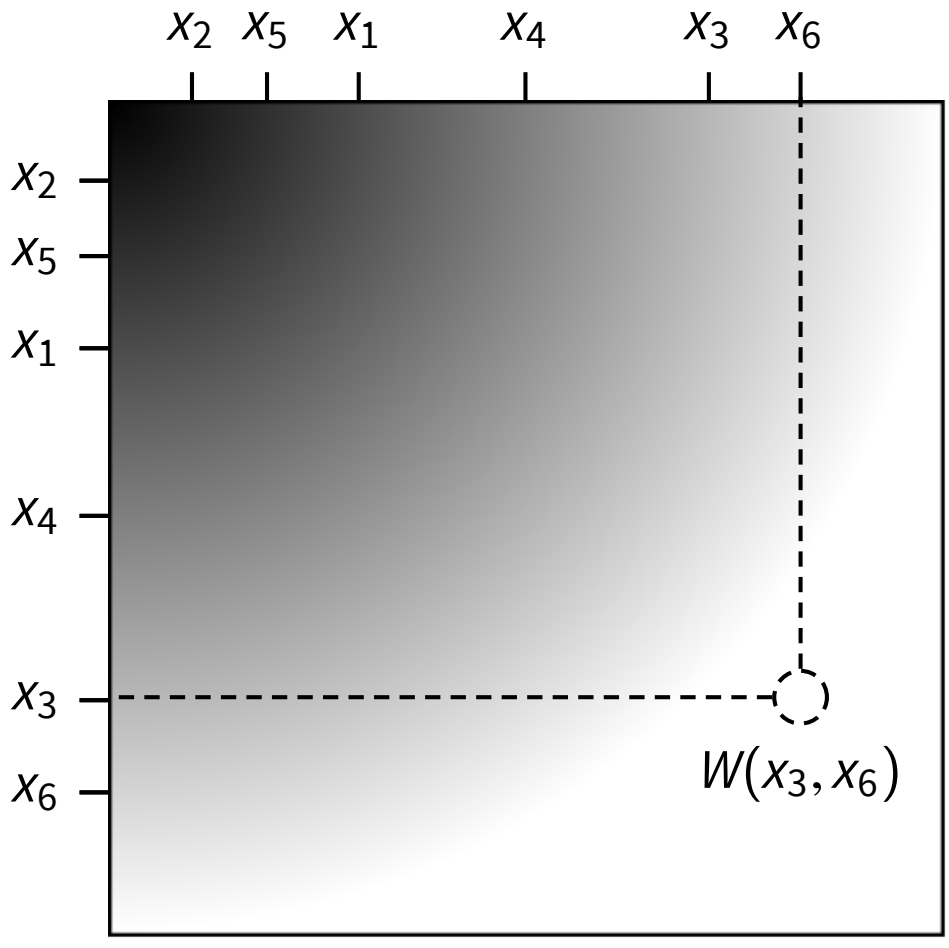
By chance, edge (x_1, x_5) is included.

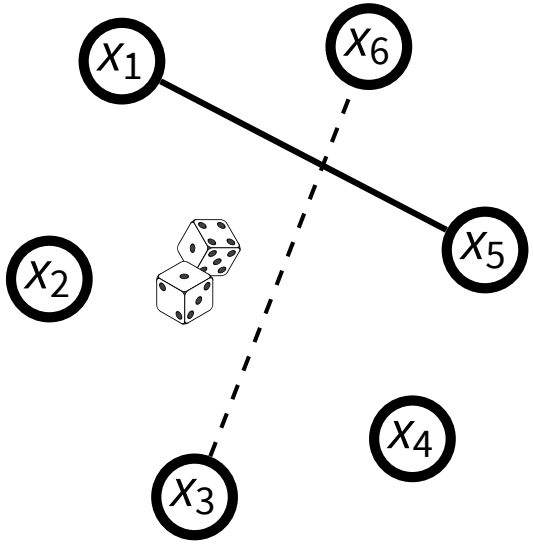




Sampling a graph from W.

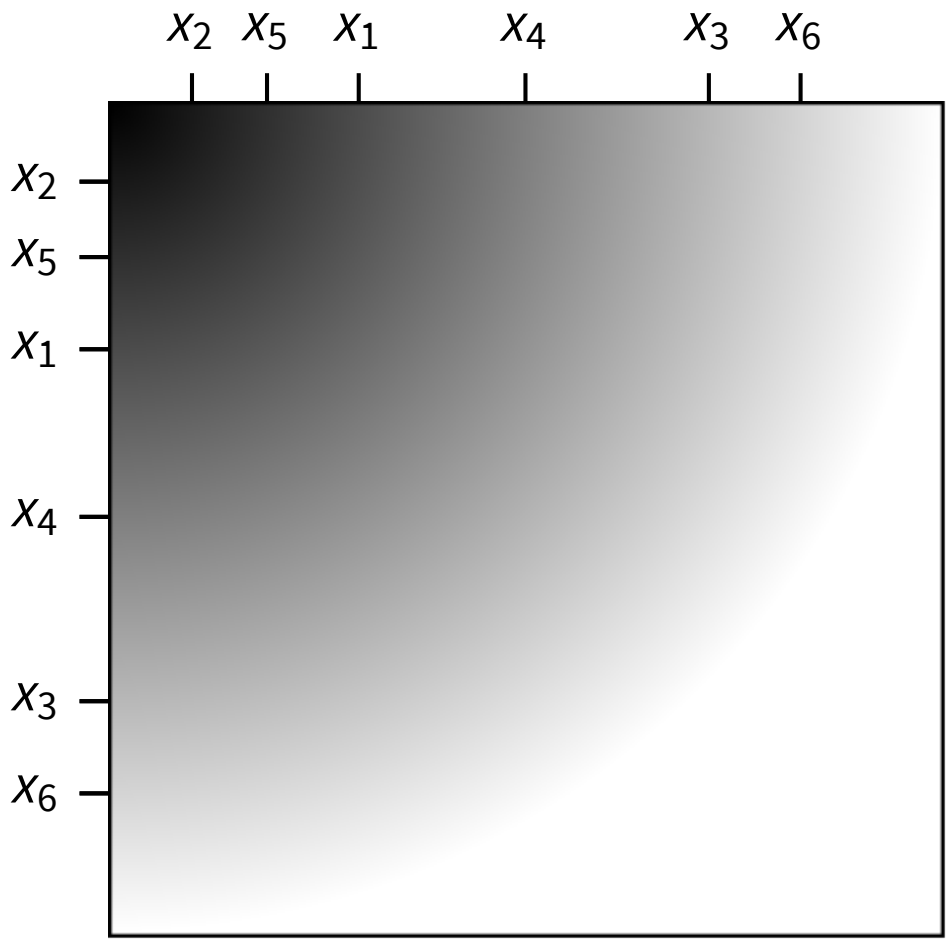
Include edge (x_3, x_6) with probability $W(x_3, x_6)$.

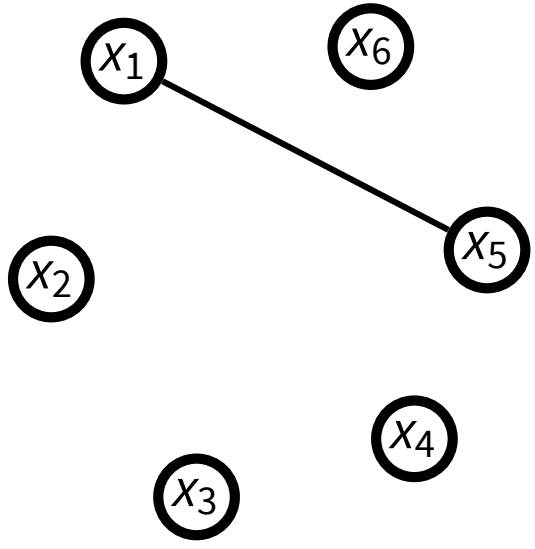




Sampling a graph from W.

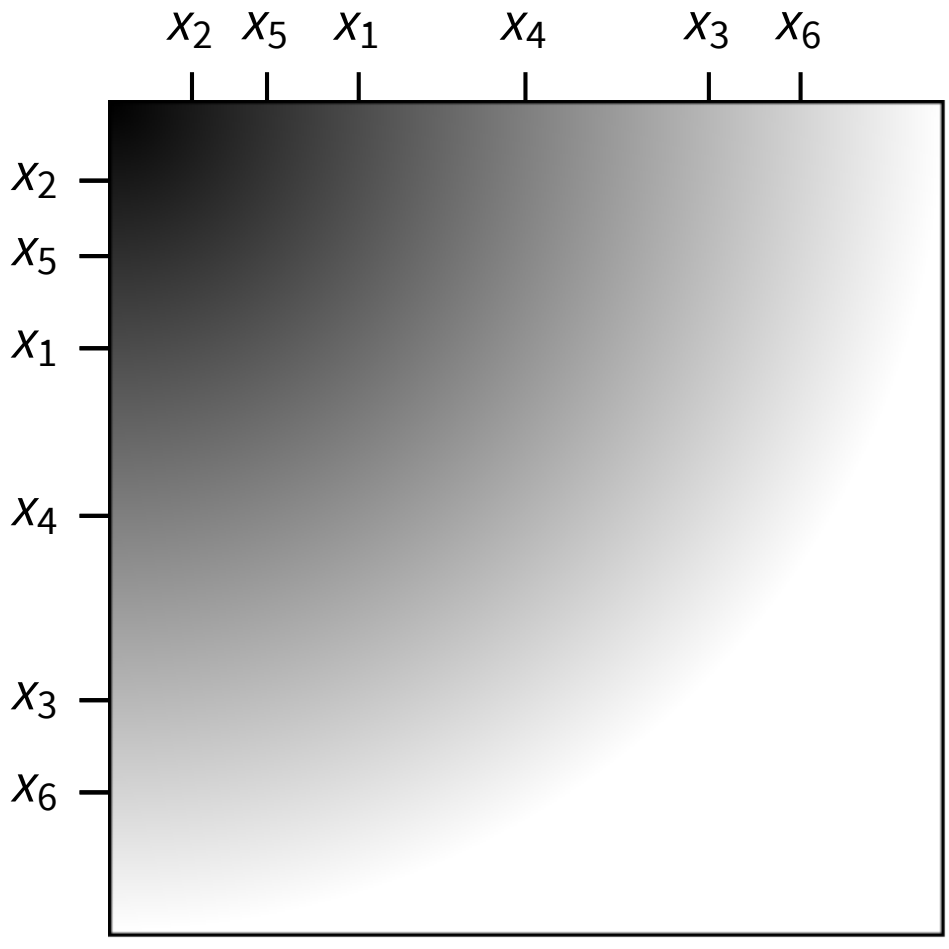
By chance, edge (x_3, x_6) is omitted.

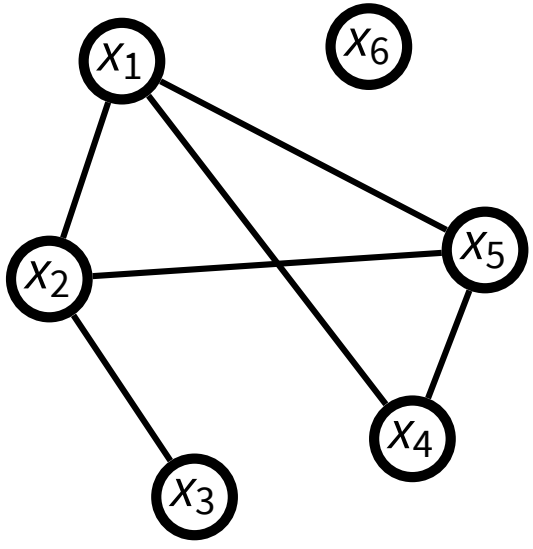




Sampling a graph from W.

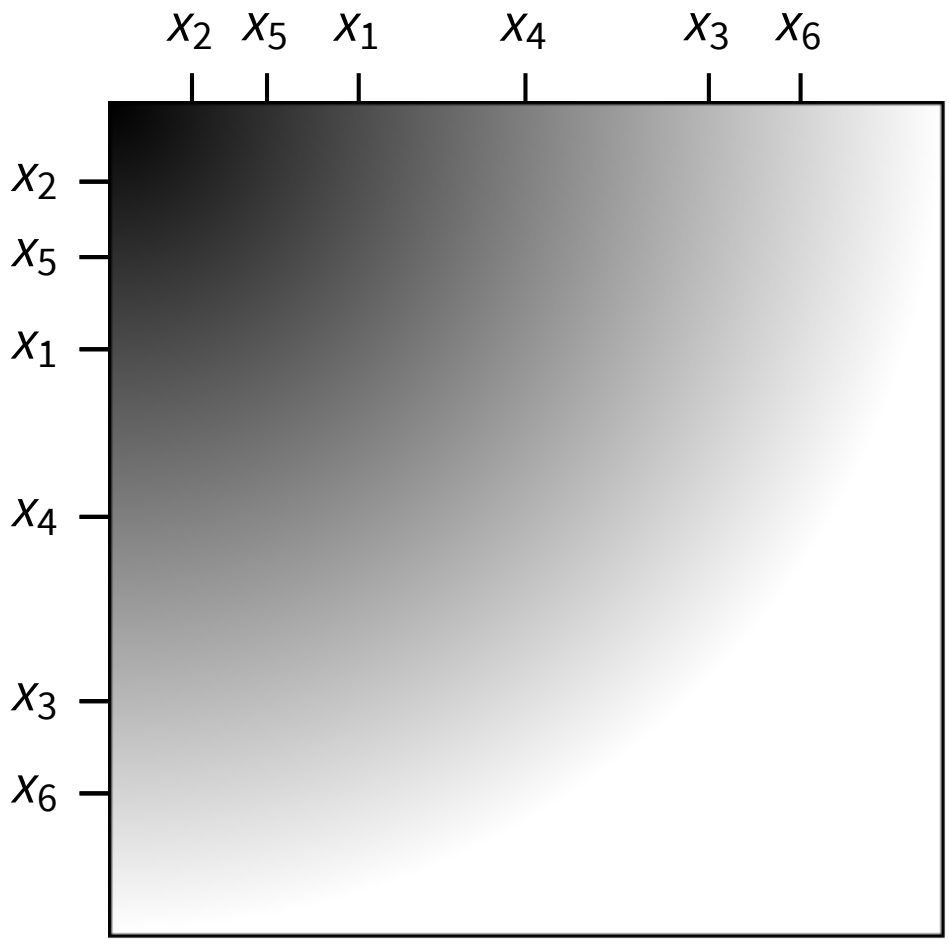
Repeat for all possible edges.

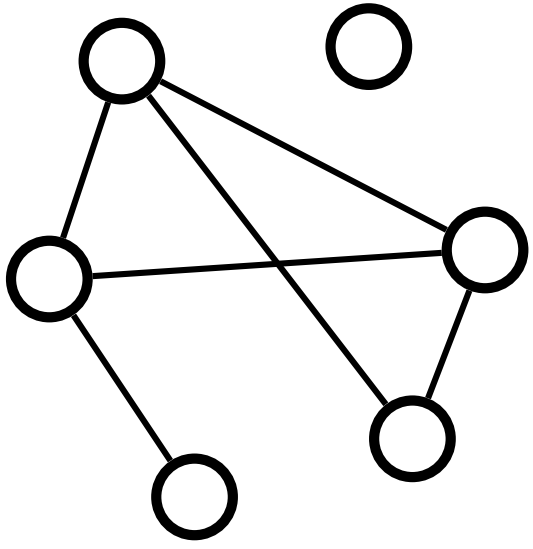




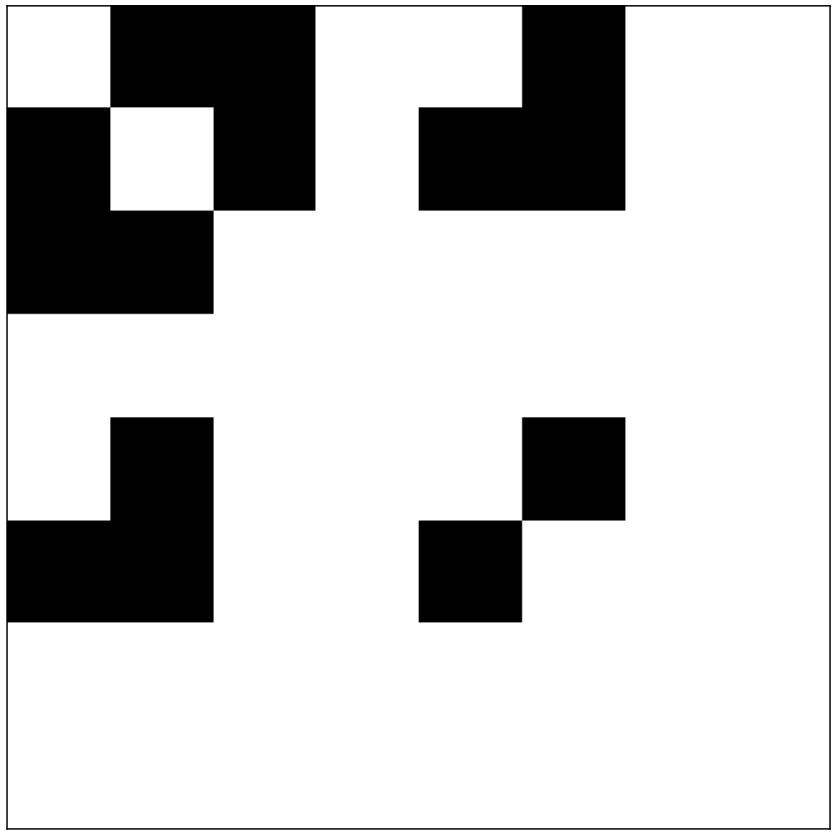
Sampling a graph from W.

Forget node labels, obtaining undirected & unweighted graph.

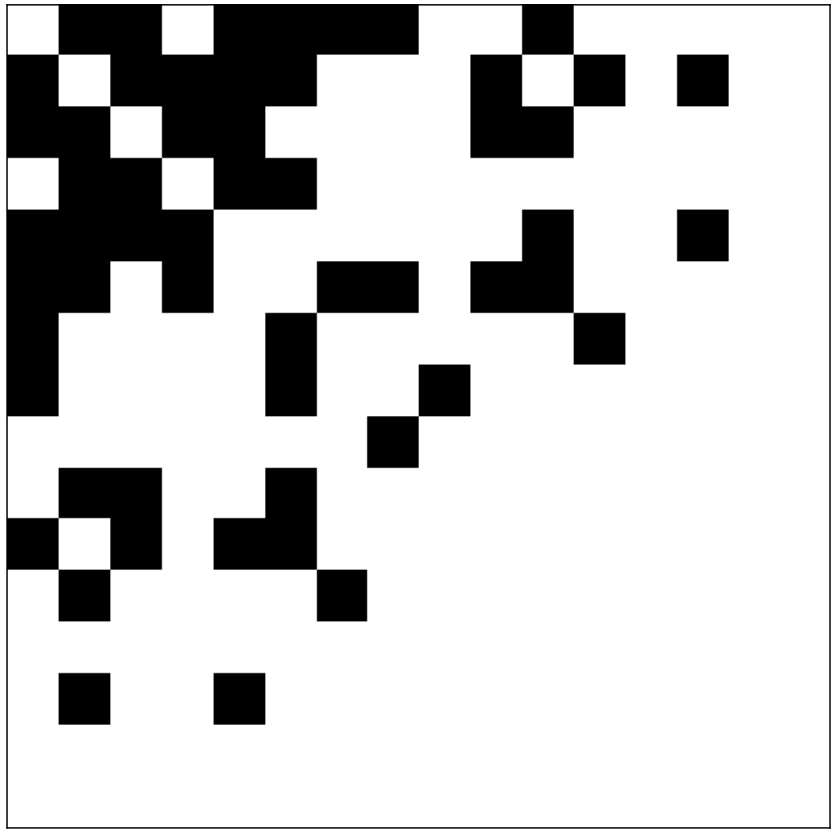




Sampled graphs converge to the graphon they were sampled from.

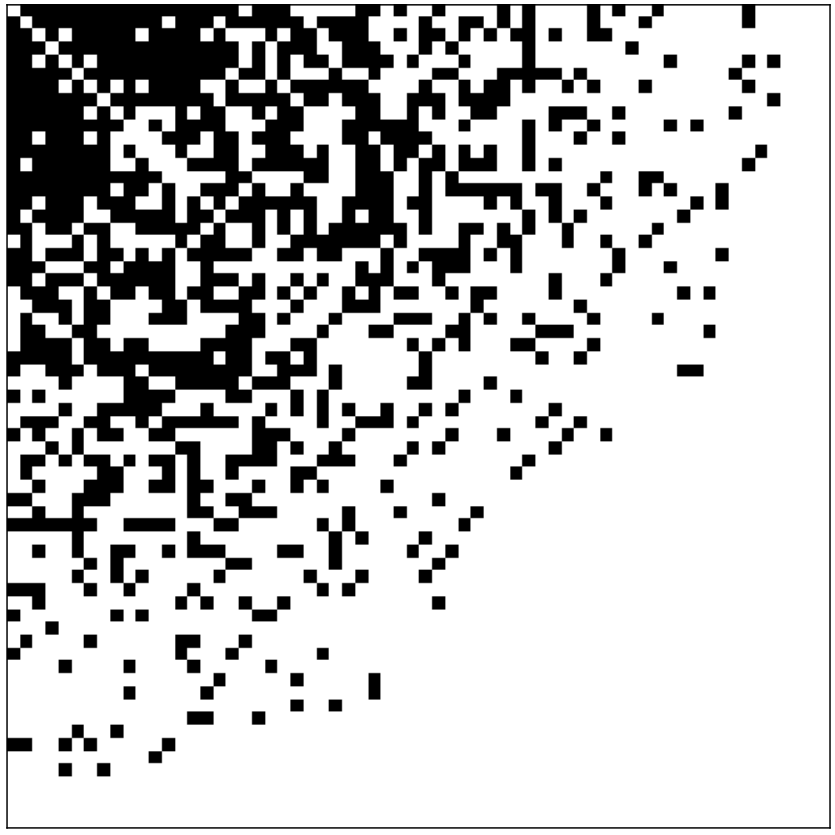


Sampled graphs converge to the graphon they were sampled from.

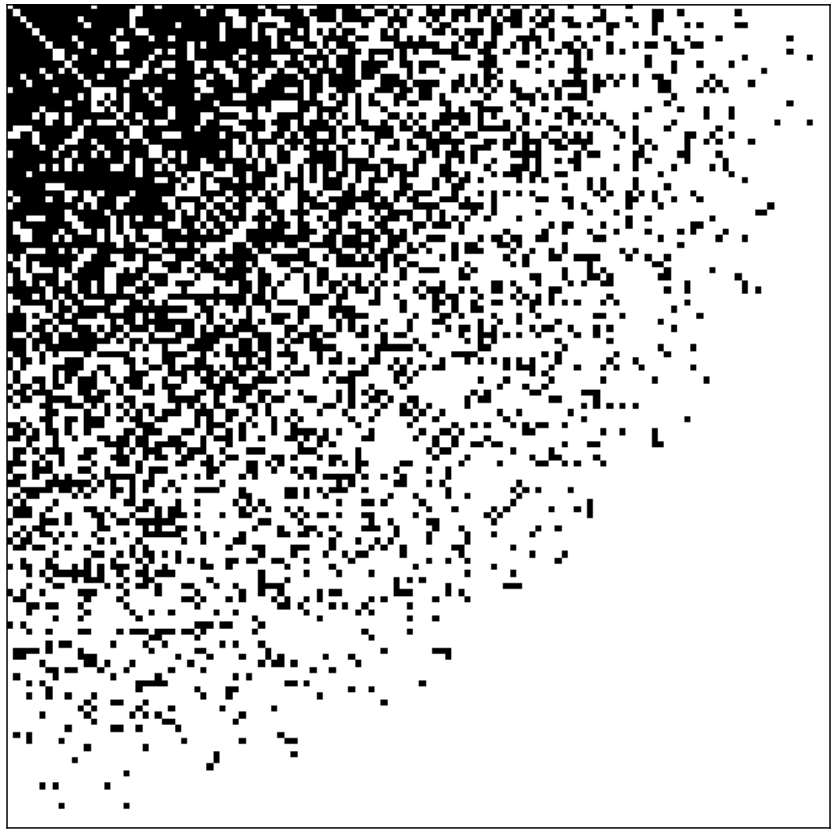


Sampled graphs converge to the graphon they were sampled from.

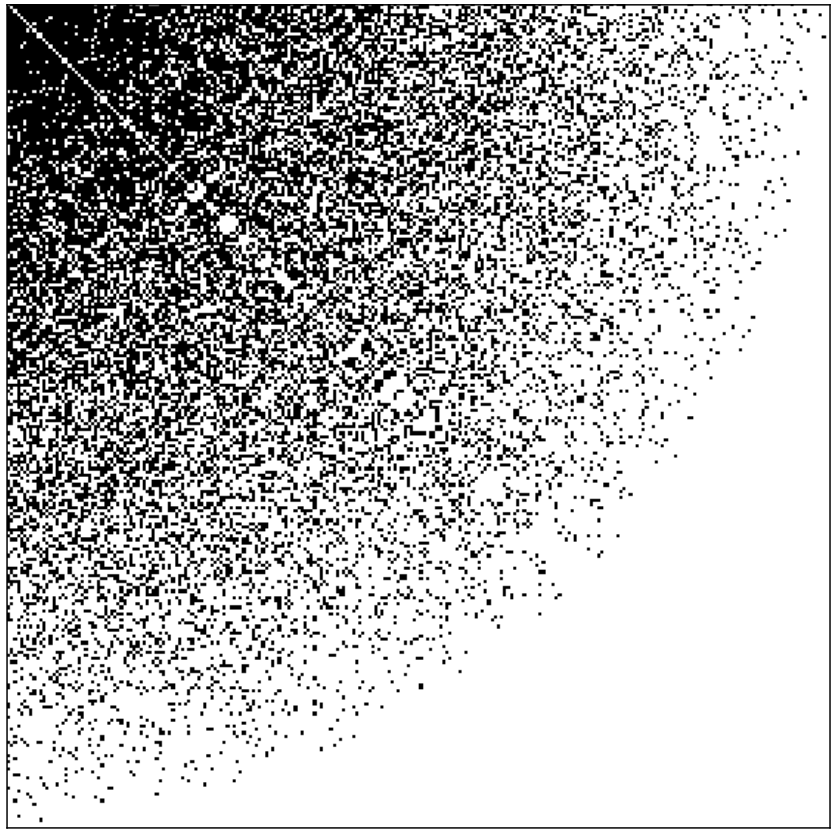
Sampled graphs converge to the graphon they were sampled from.



Sampled graphs converge to the graphon they were sampled from.



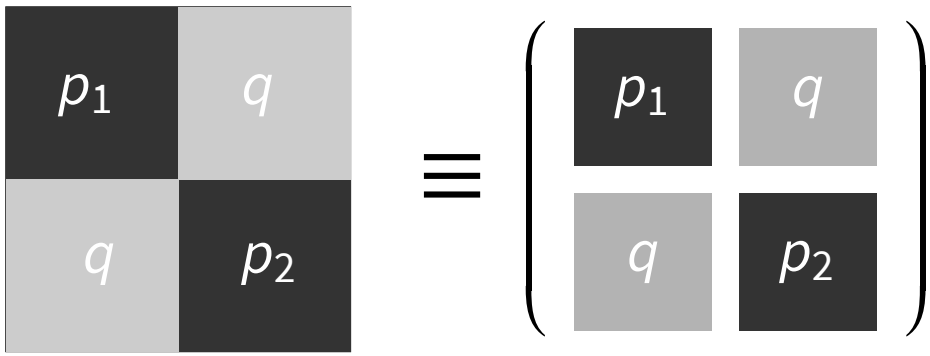
Sampled graphs converge to the graphon they were sampled from.



Sampled graphs converge to the graphon they were sampled from.

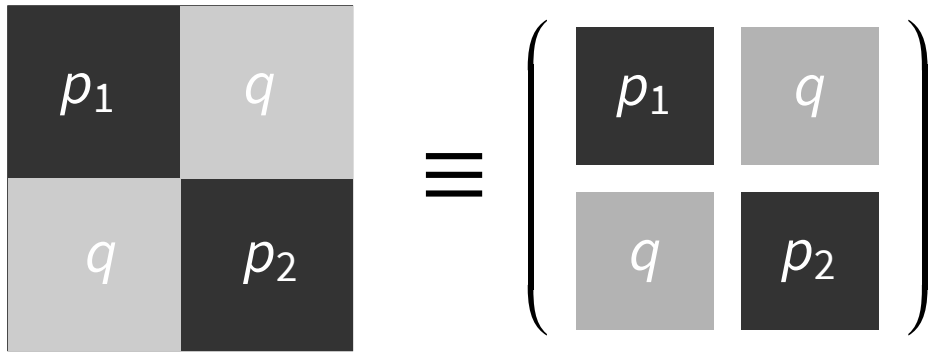
A graphon W defines a very rich distribution on graphs.

- Better models real-world data (Hoff, 2002).
- Subsumes many models, e.g., blockmodel:



A graphon W defines a very rich distribution on graphs.

- Better models real-world data (Hoff, 2002).
- Subsumes many models, e.g., blockmodel:



Warning! Graphons can be much more complex than blockmodels.

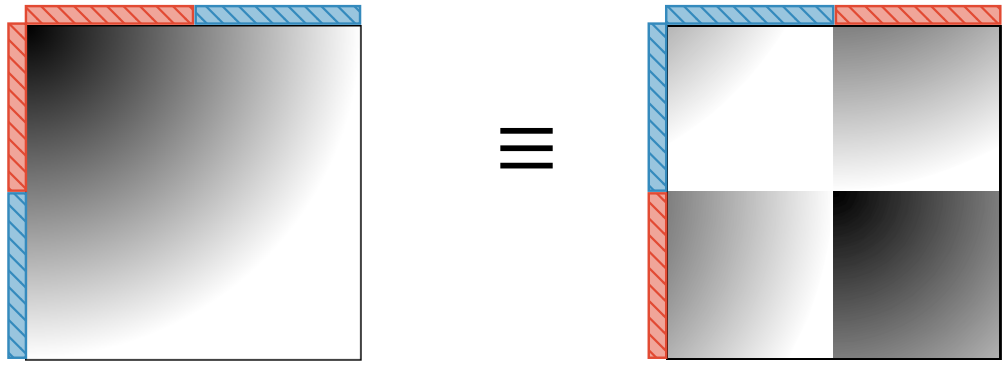
- Present several unique and subtle technical issues.

Issue 1: A graphon is truly a measure-theoretic object.

```
$ grep -i 'measur' nips-2016_paper/**.tex
200
$ grep -i 'measur' this_talk/**.tex
3
```

We will ignore this in the interest of the exposition.

As with finite graphs, we have a notion of graphon isomorphism.



(Lovász, 2012): W_1 and W_2 define the same random graph model

$\Longleftrightarrow$

they are equivalent up to relabeling.

Issue 2: In general, there is no canonical way to label the model.
Definitions must not strongly rely on particular labeling.

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

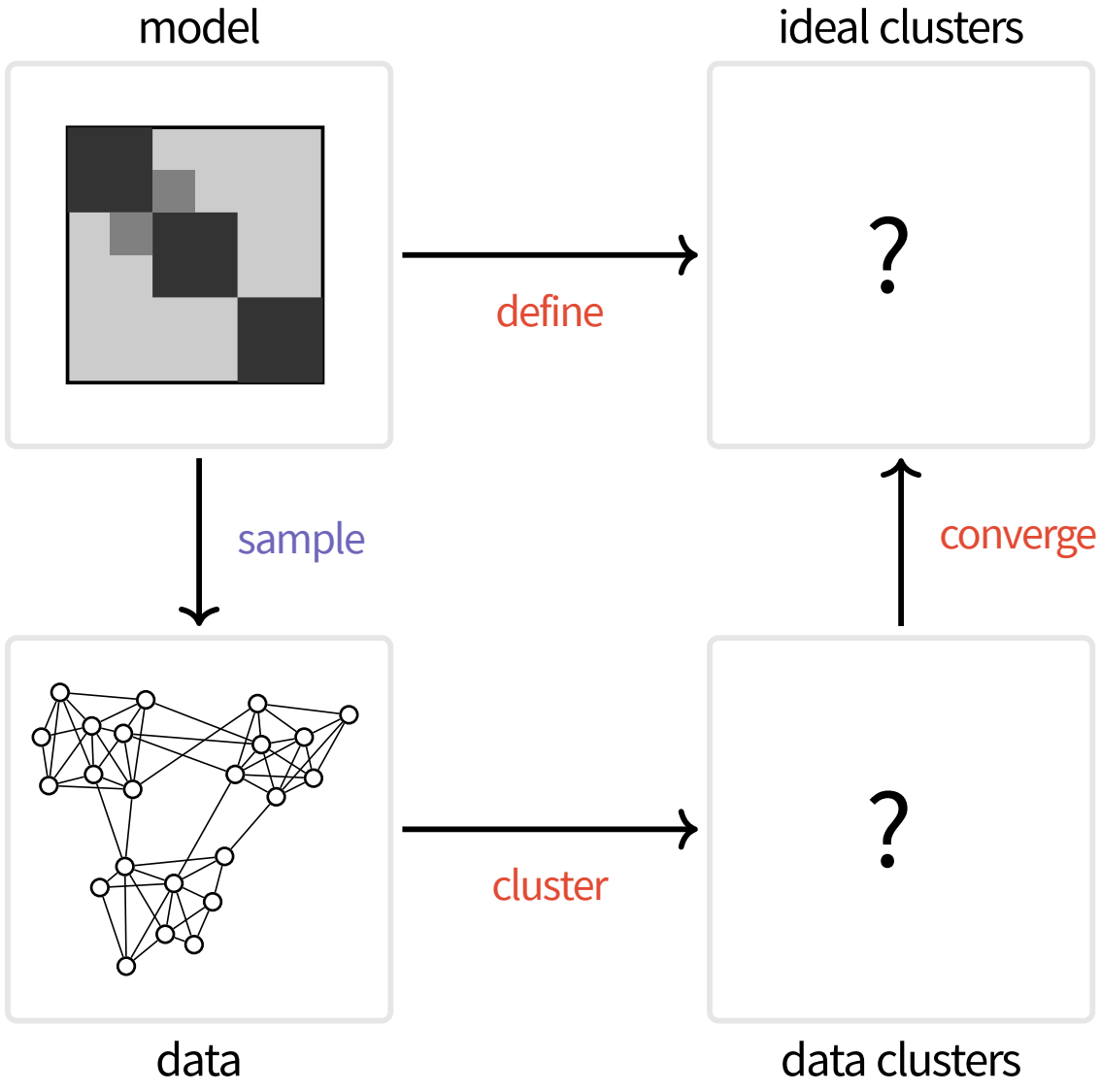
Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

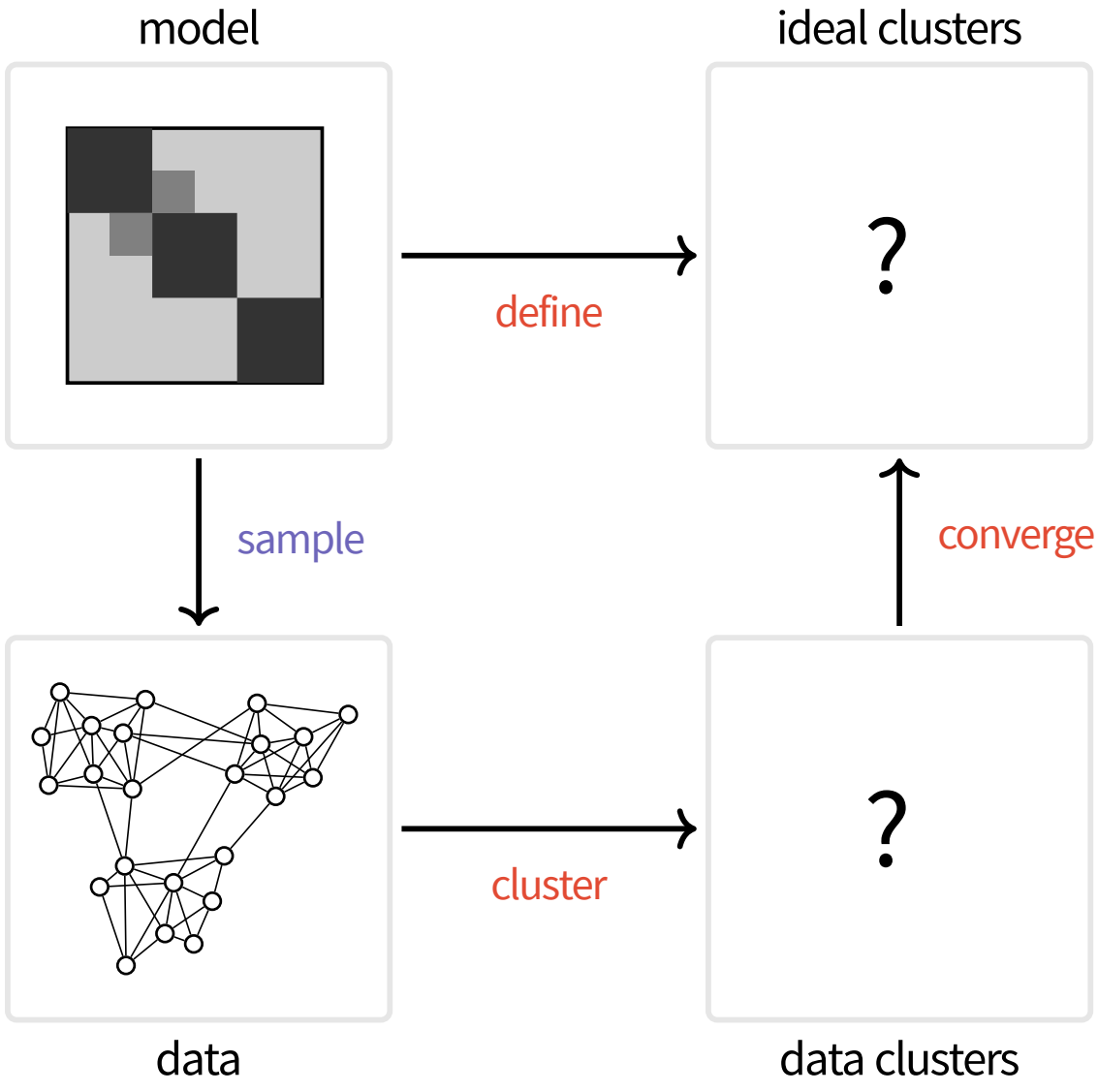
- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

Applications & Directions

Clustering in the graphon model was virtually unstudied.

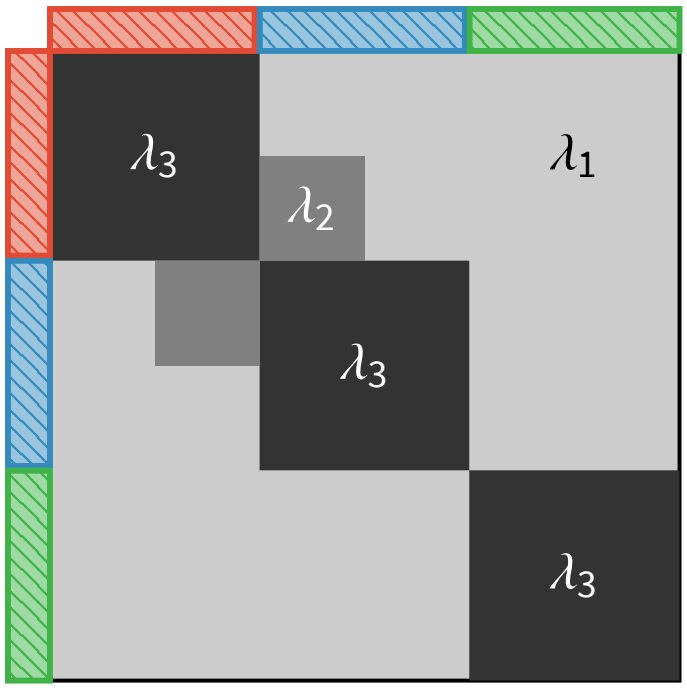


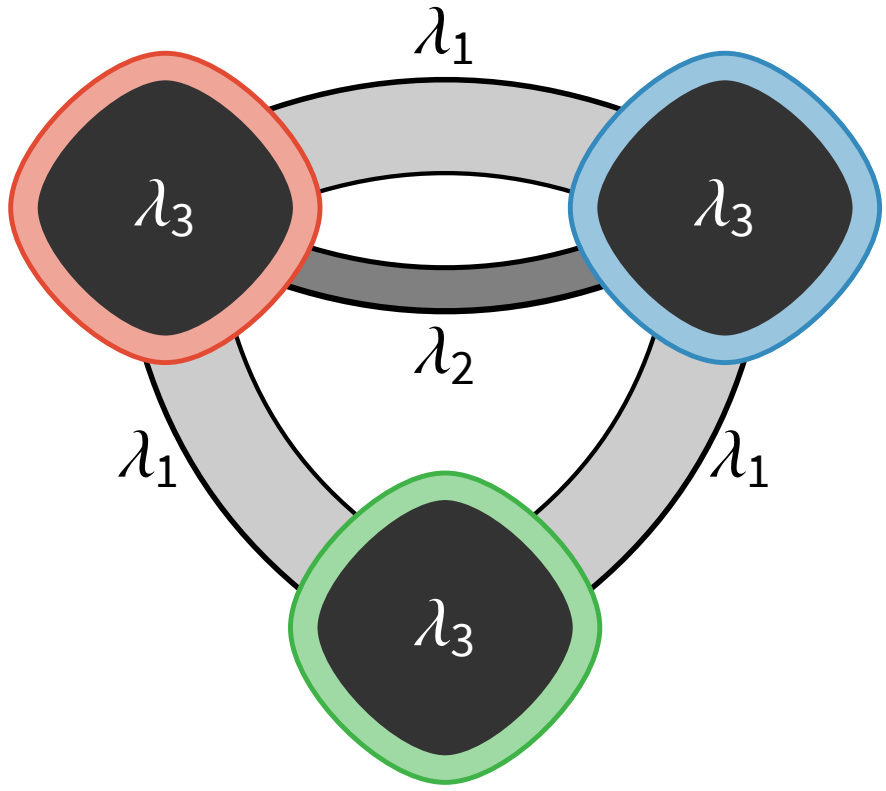
Step 1: Define the clusters of a graphon.



What are the clusters of a graphon?

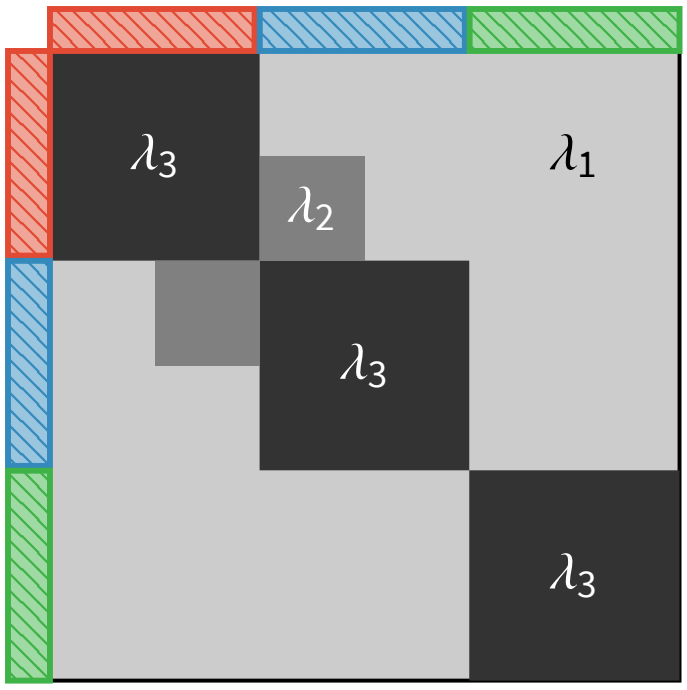
We interpret the graphon as the adjacency of an infinite weighted graph.

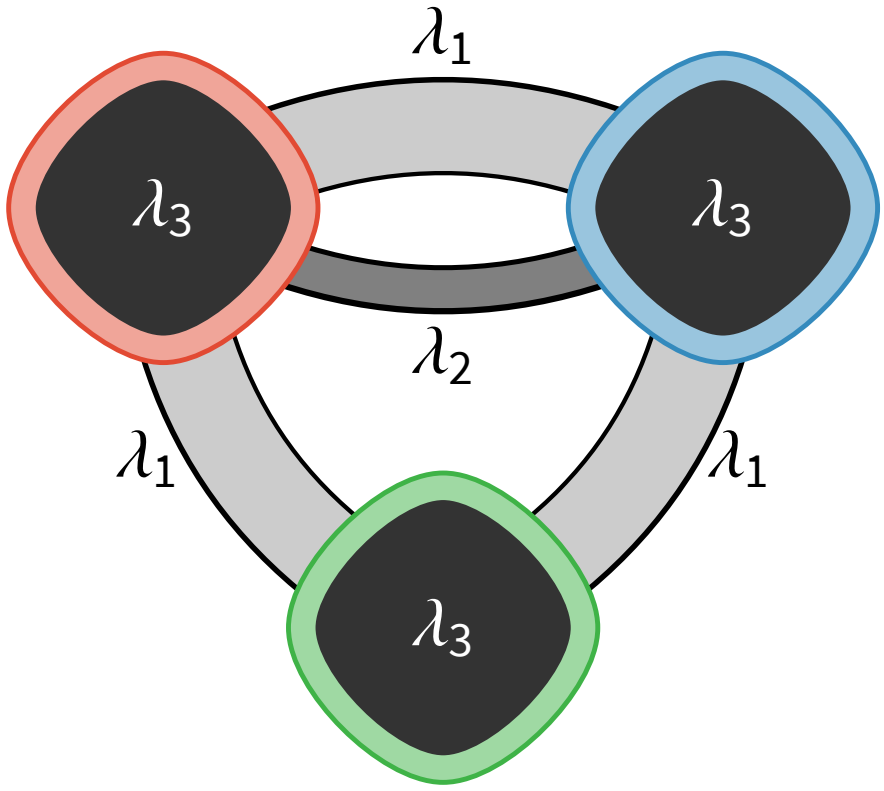




What are the clusters of a graphon?

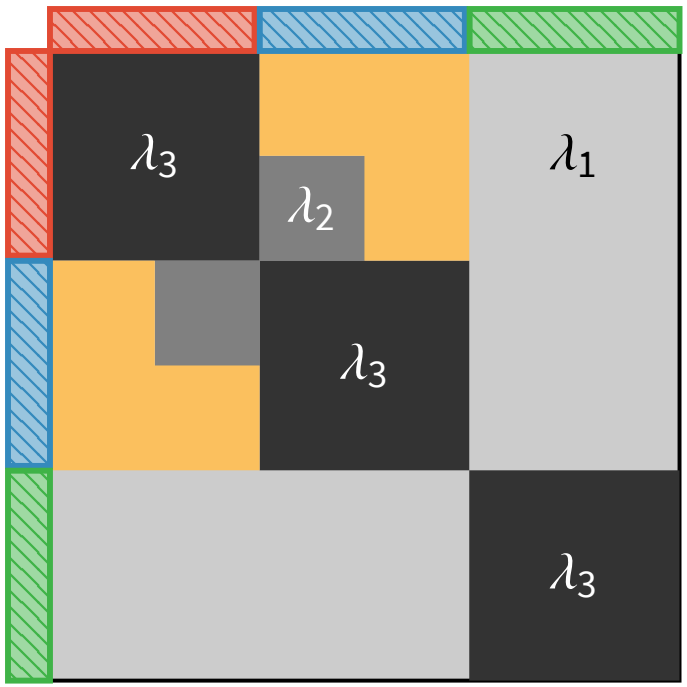
Each link in this depiction corresponds to a region of the graphon.

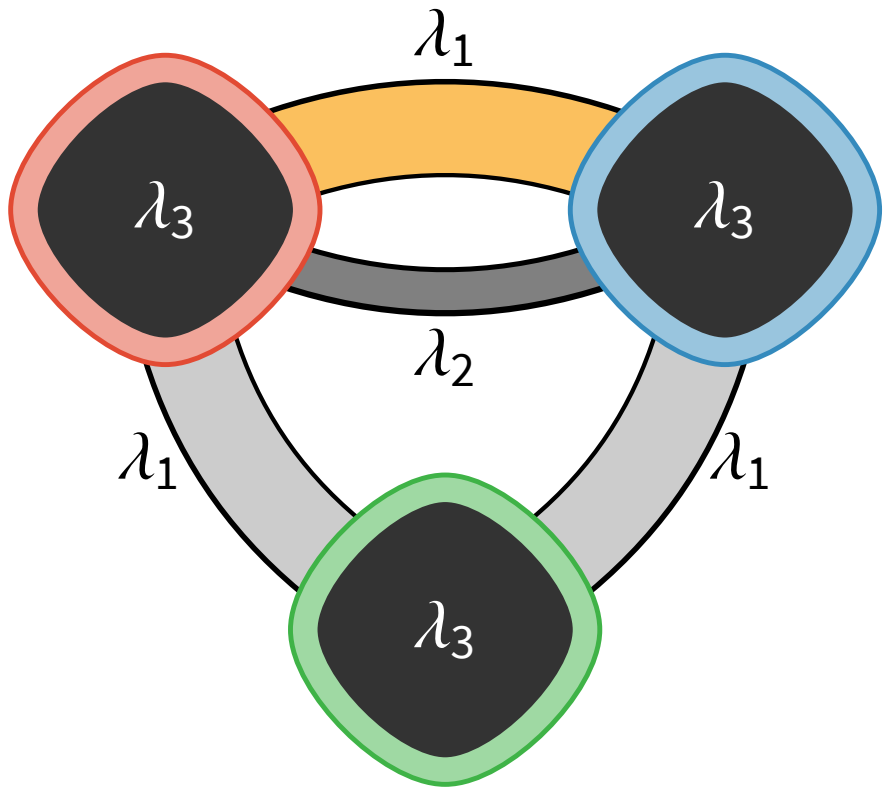




What are the clusters of a graphon?

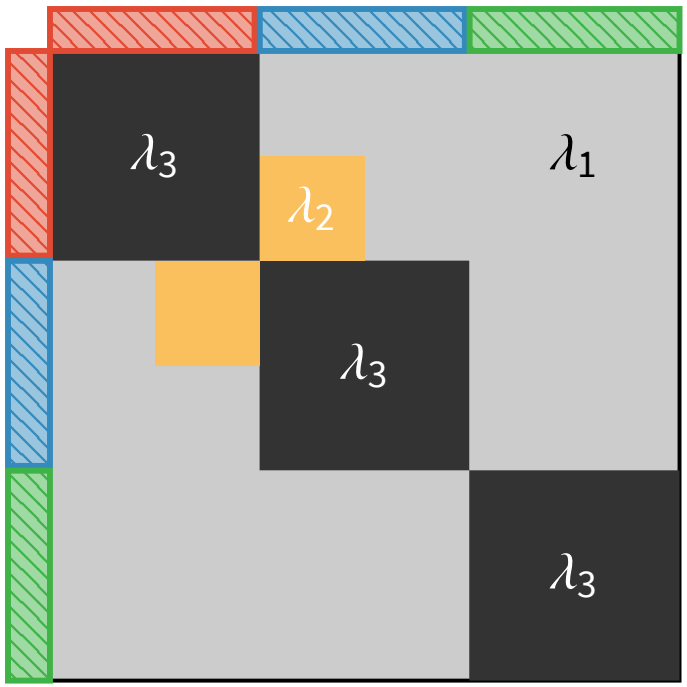
Each link in this depiction corresponds to a region of the graphon.

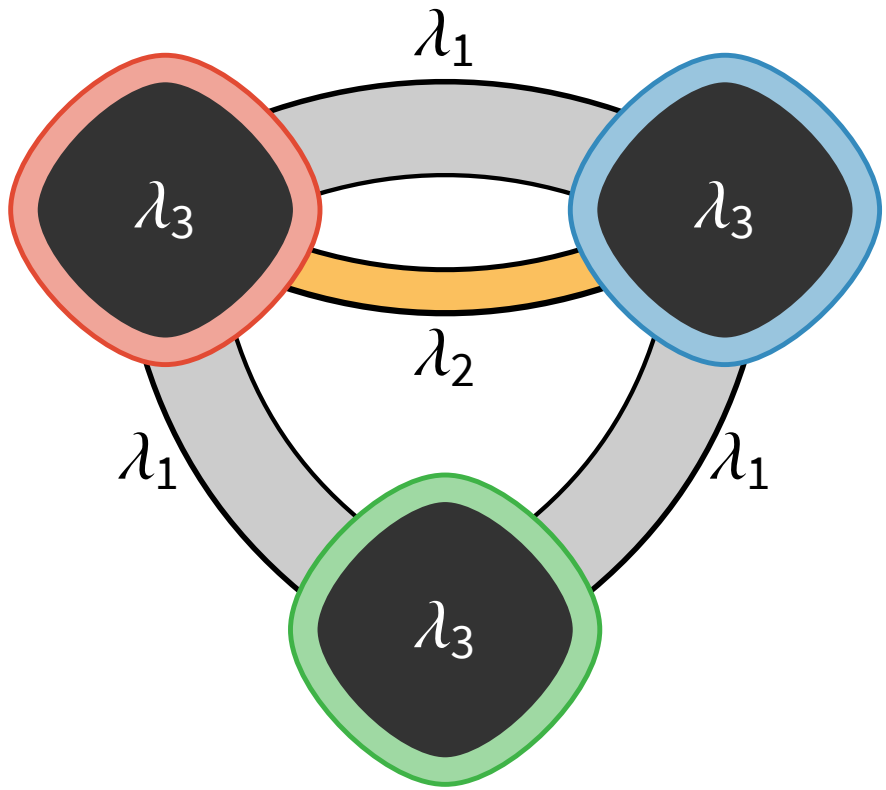




What are the clusters of a graphon?

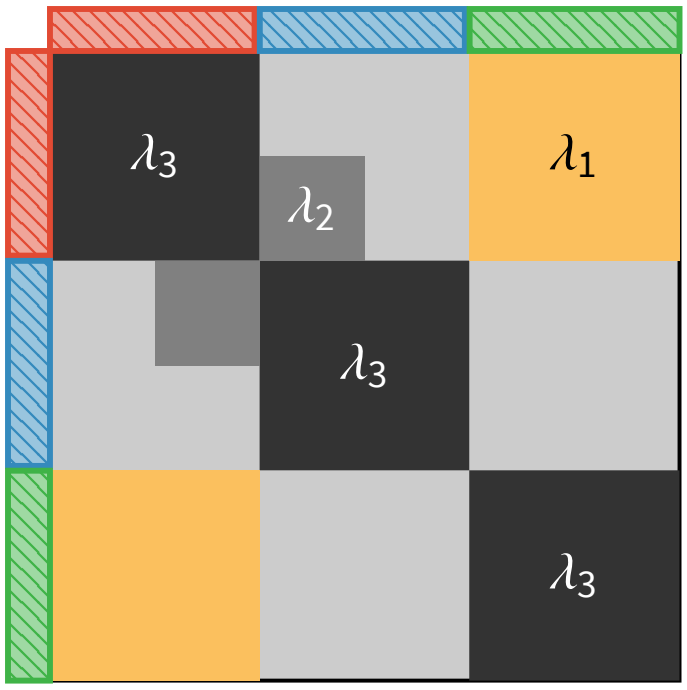
Each link in this depiction corresponds to a region of the graphon.

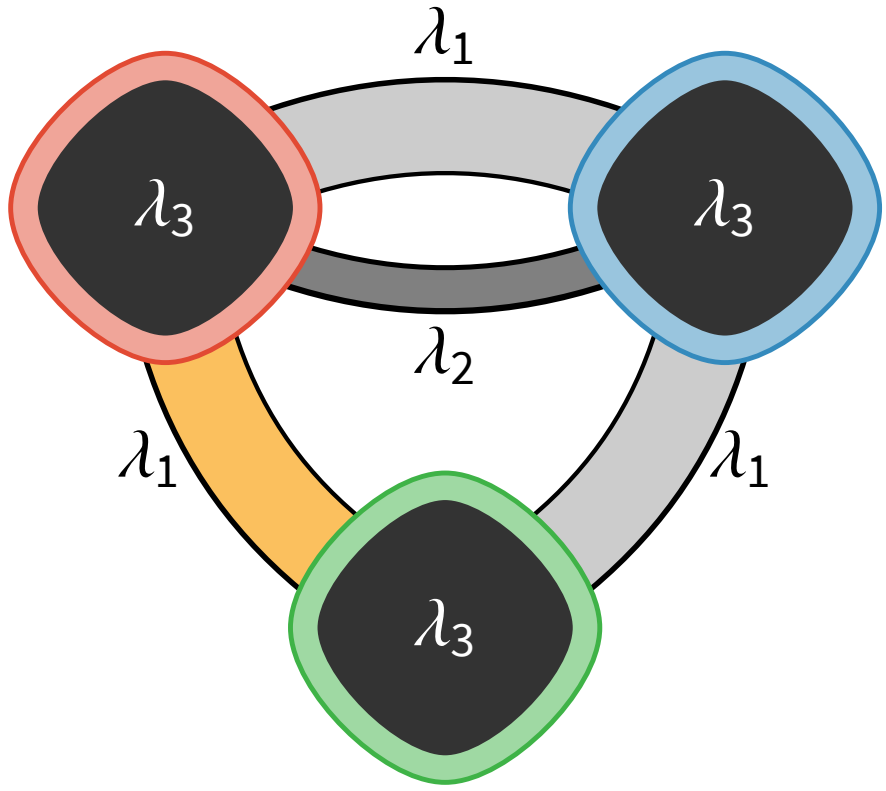




What are the clusters of a graphon?

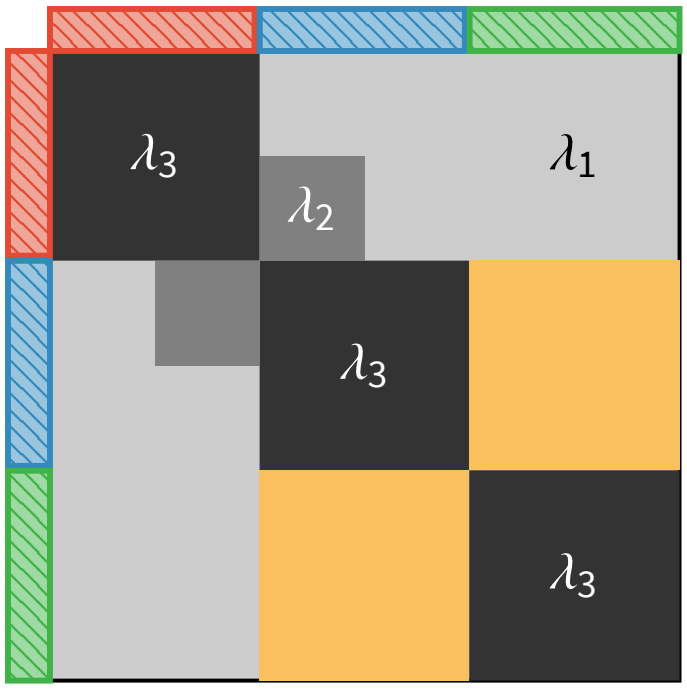
Each link in this depiction corresponds to a region of the graphon.

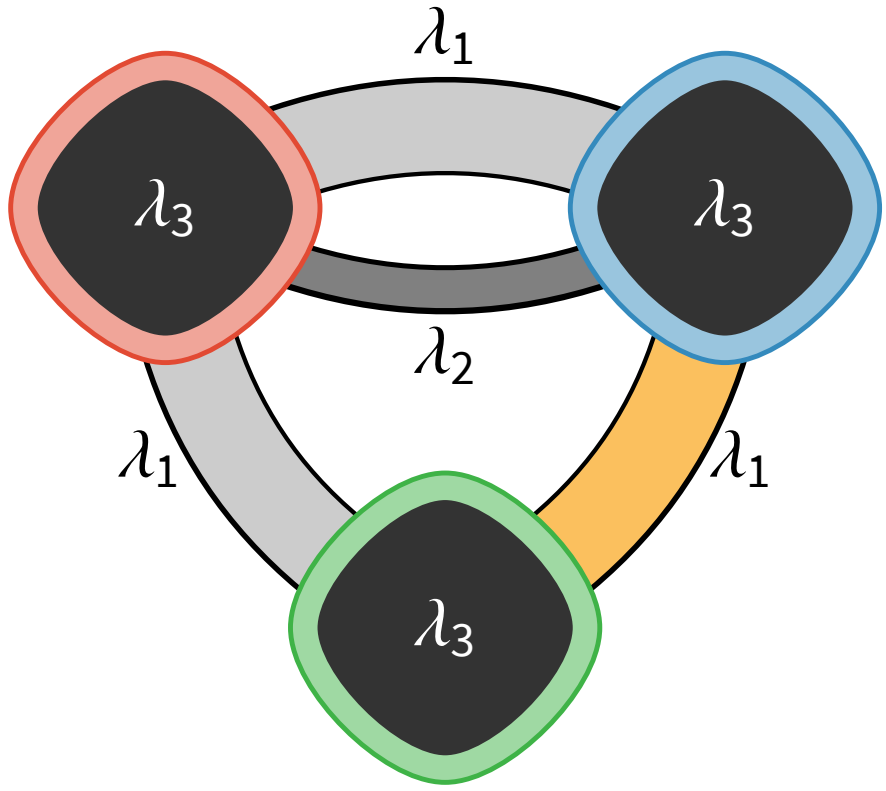




What are the clusters of a graphon?

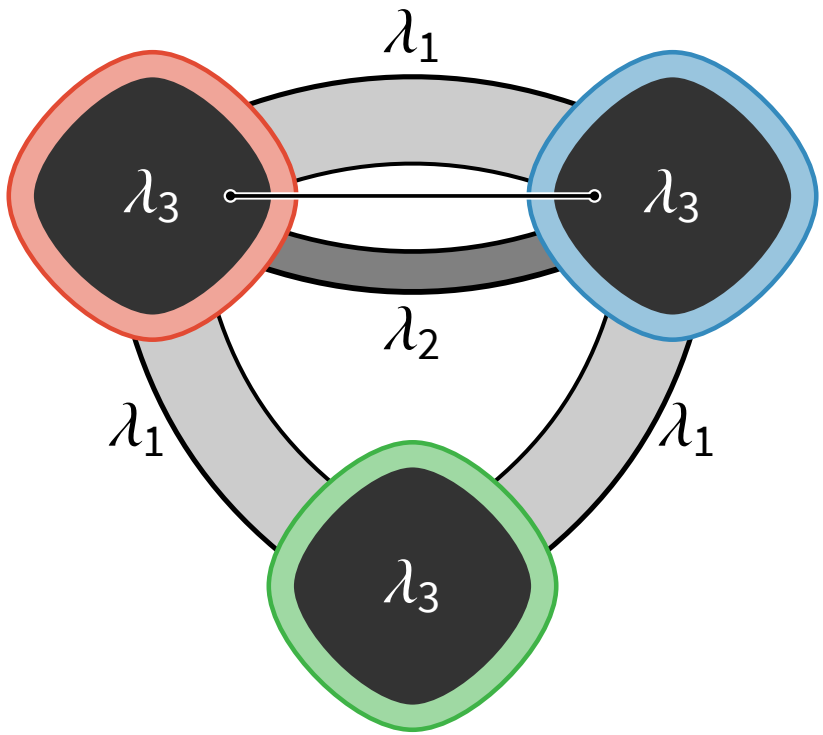
Each link in this depiction corresponds to a region of the graphon.





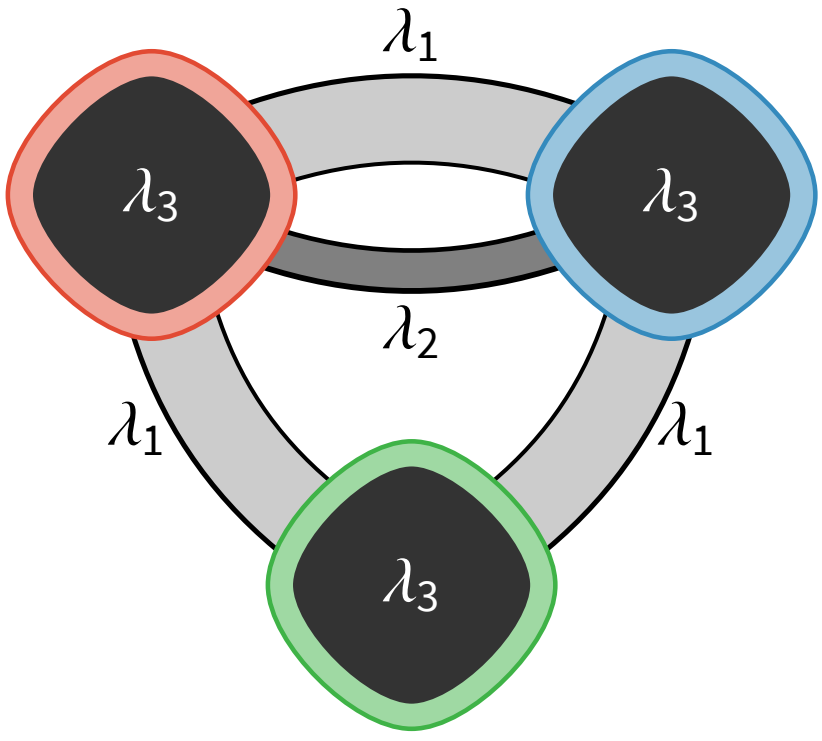
What are the clusters of a graphon?

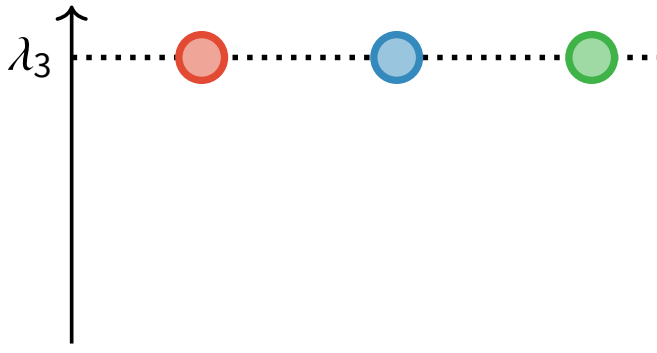
- We define clusters to be connected components.
- Generalize graph connectivity, extends (Janson, 2008). Details
- In fact, we can speak of the clusters at various levels.



What are the clusters of a graphon?

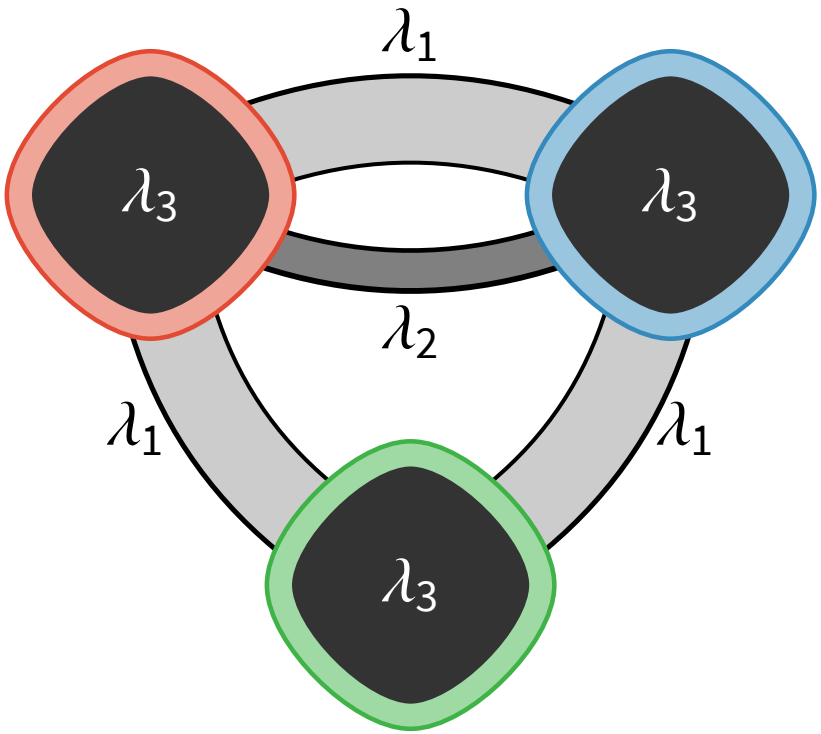
- Intuition: Two graphon nodes are connected at level λ if there is a path between them along which each edge has weight $\geq \lambda$.
- Three clusters (connected components) at level λ_3.
- Any pair (●, ●) are in same cluster at λ_3. Same for (●, ●) & (●, ●).

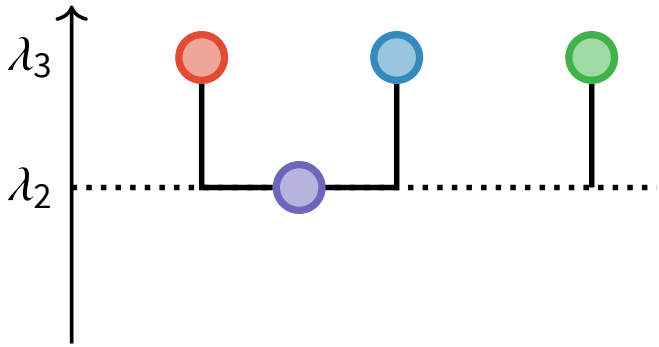




What are the clusters of a graphon?

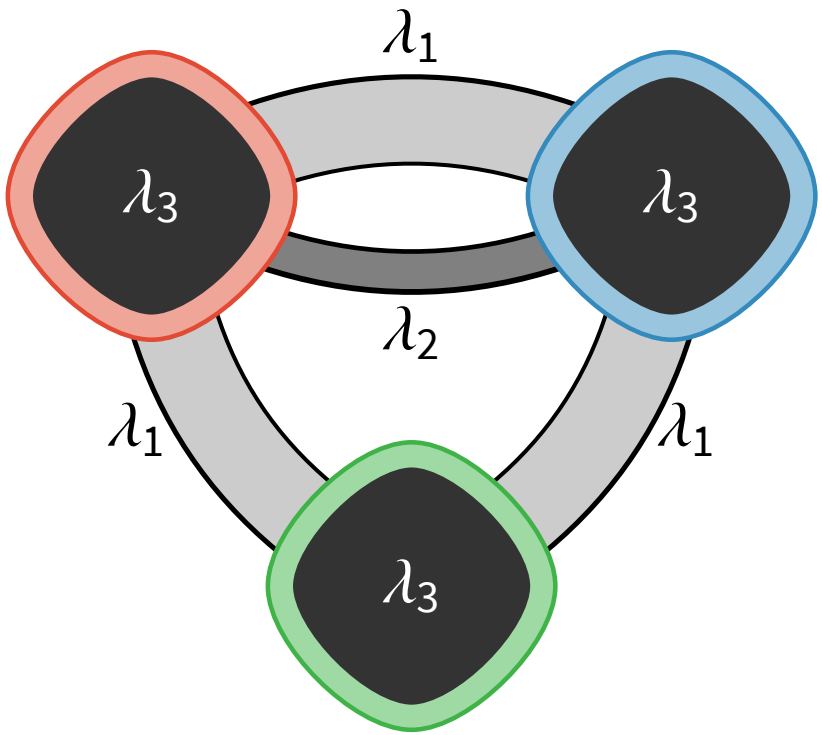
- Intuition: Two graphon nodes are connected at level λ if there is a path between them along which each edge has weight $\geq \lambda$.
- Intuitively: red and blue clusters merge at level λ_2.
- Any pair (●, ●) are in same cluster at λ_2.

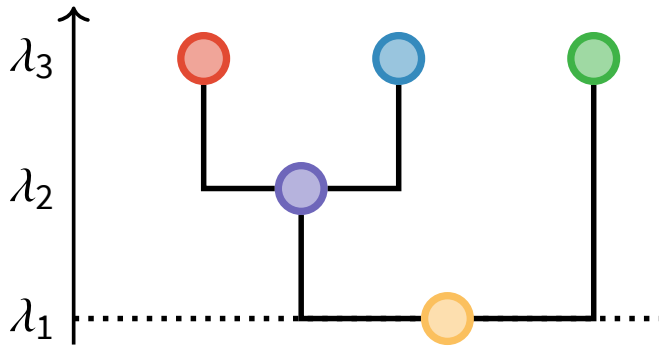


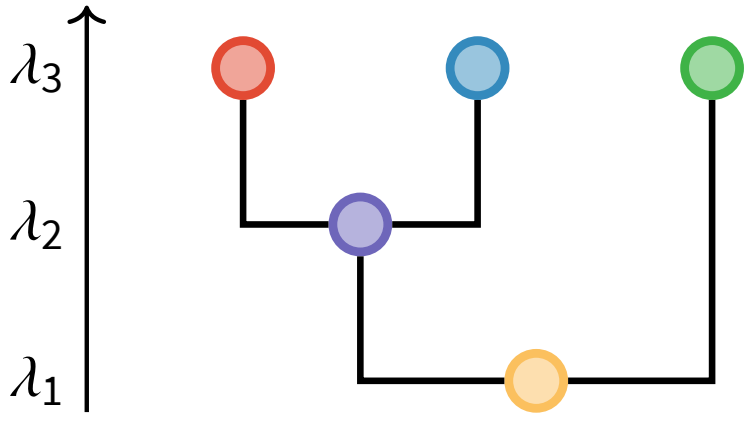


What are the clusters of a graphon?

- Intuition: Two graphon nodes are connected at level λ if there is a path between them along which each edge has weight $\geq \lambda$.
- All clusters merge at level λ_1.



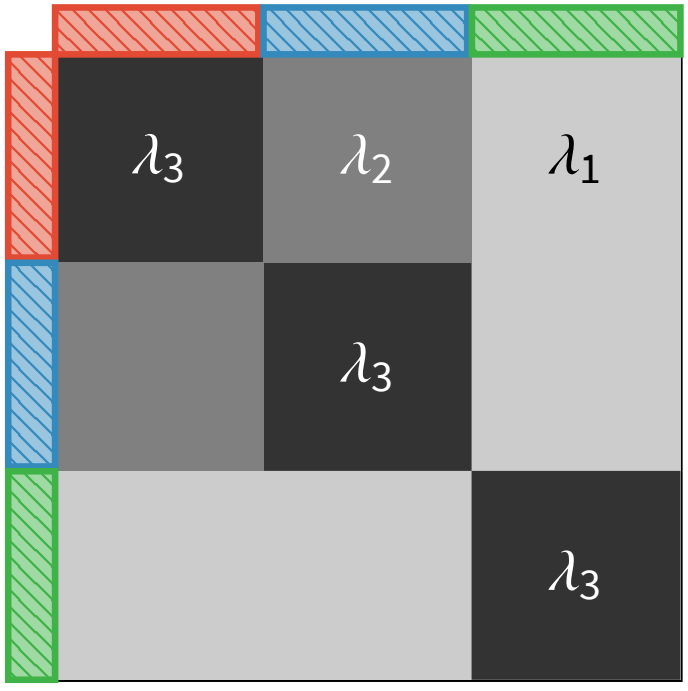


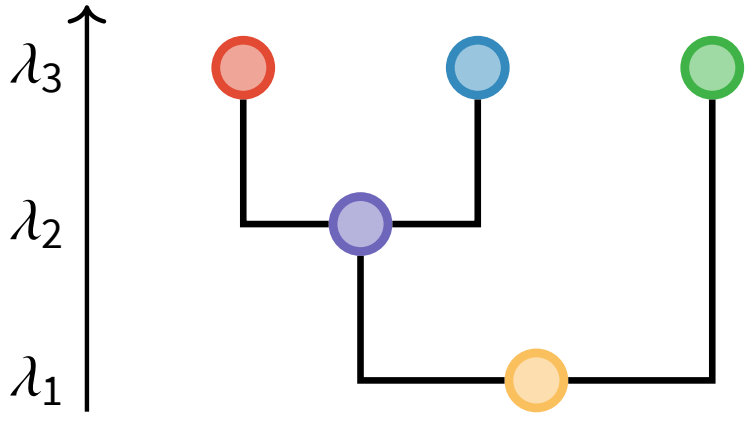


We call this structure the graphon cluster tree.

Recovering the graphon cluster tree is a natural goal of clustering in the graphon setting.

We introduce a special function M which we call the mergeon.

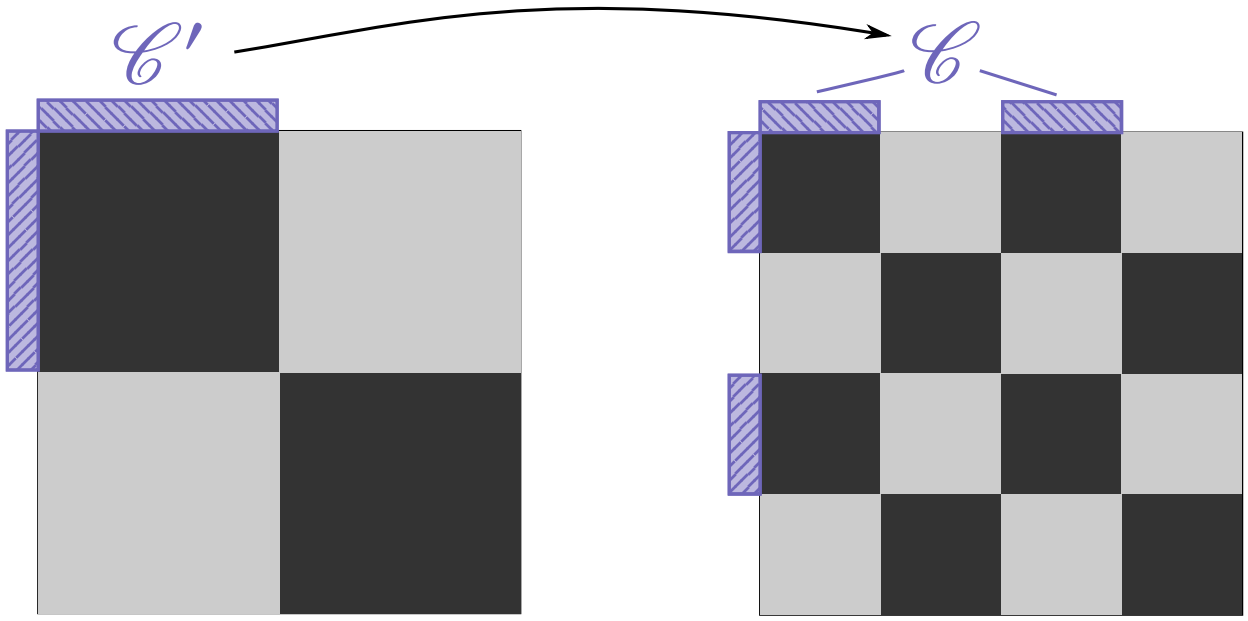




$M(x, y)$ encodes the height at which points x and y merge in the graphon cluster tree.

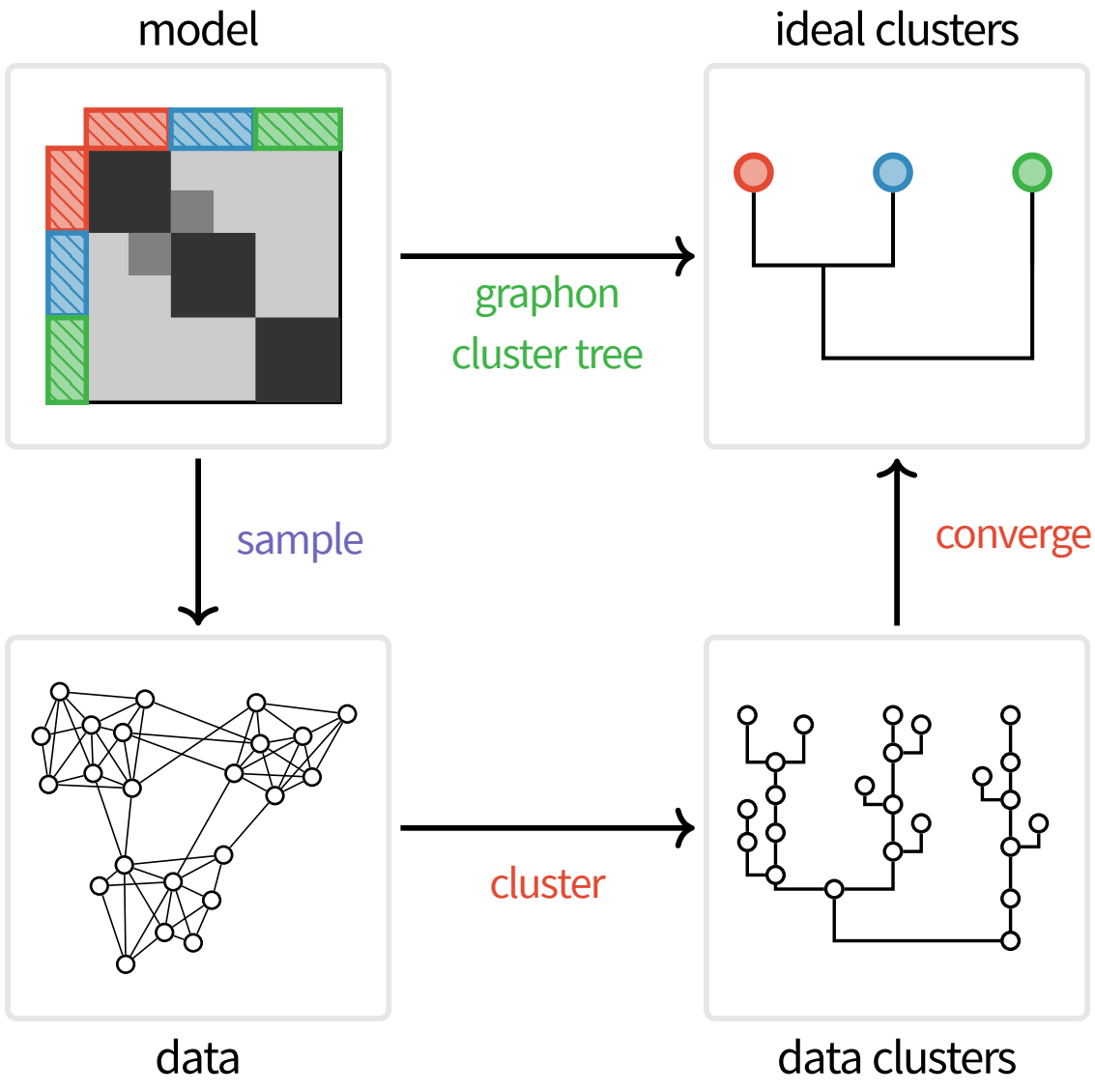
Theorem (Eldridge, 2016):

If graphons W_1 and W_2 are the same up to relabeling, then their mergeons and cluster trees are the same up to relabeling.

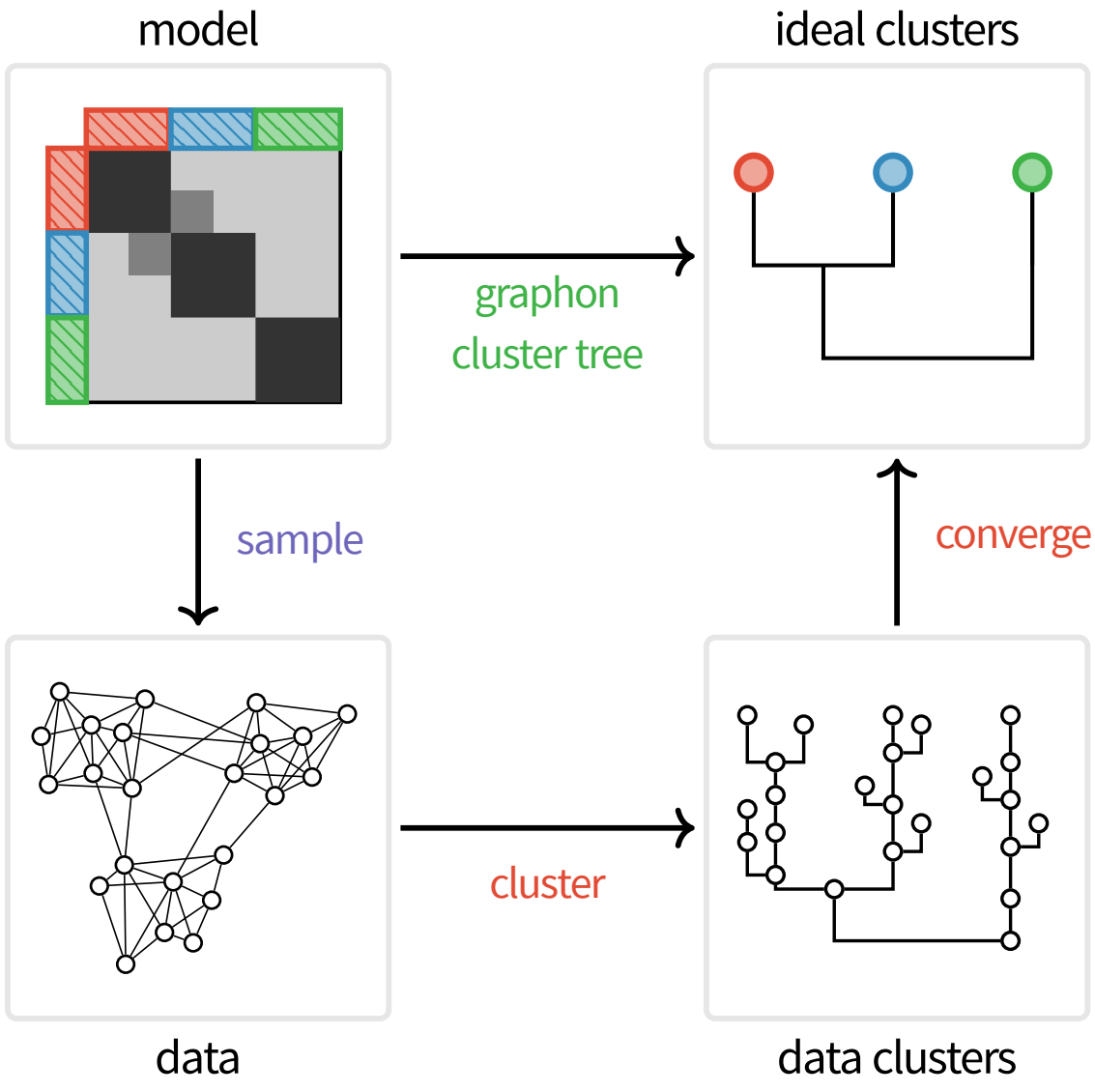


Surprisingly non-trivial to show.

The ideal clusters form the graphon cluster tree.



Step 2: Define convergence to the graphon cluster tree.



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

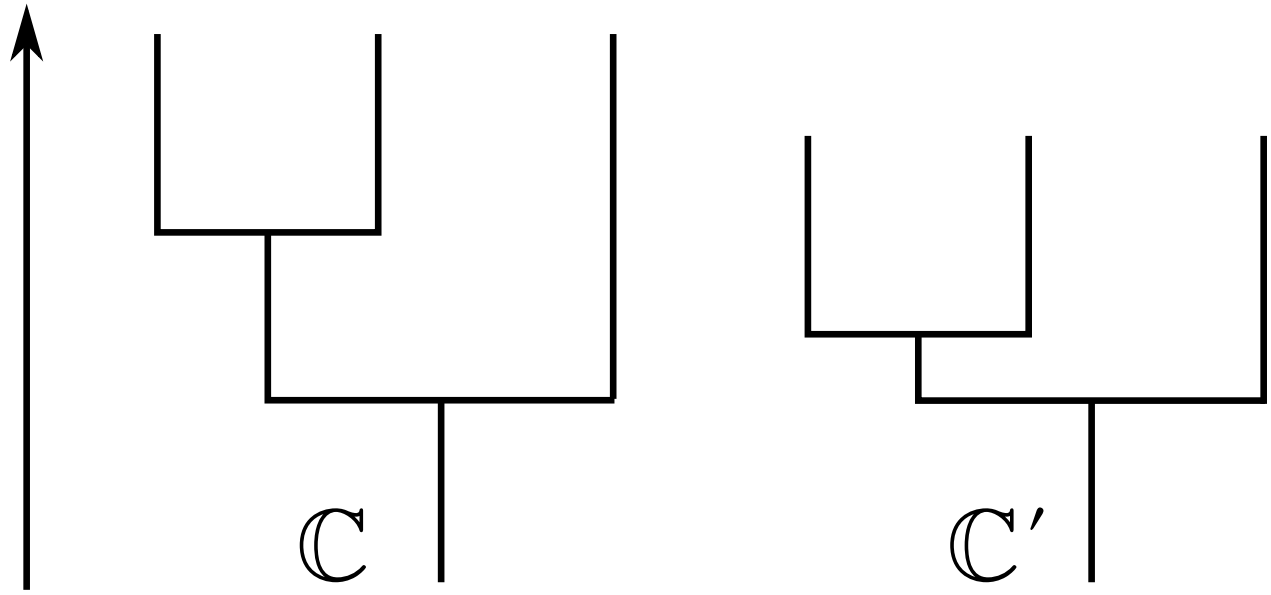
Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

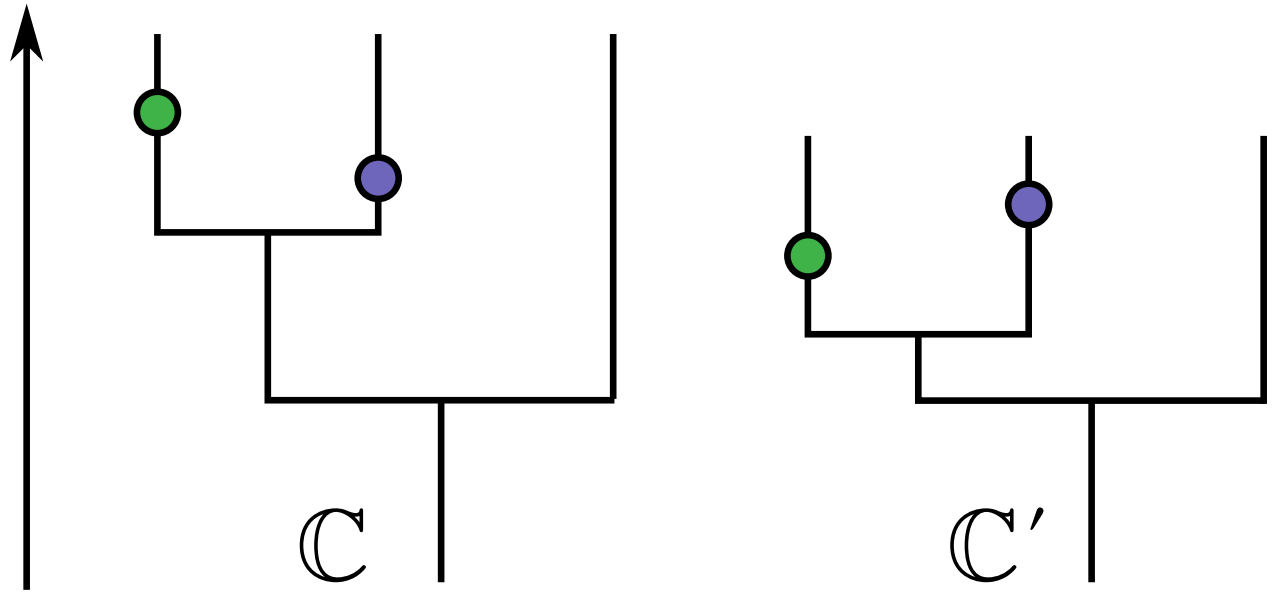
Applications & Directions

The merge distortion



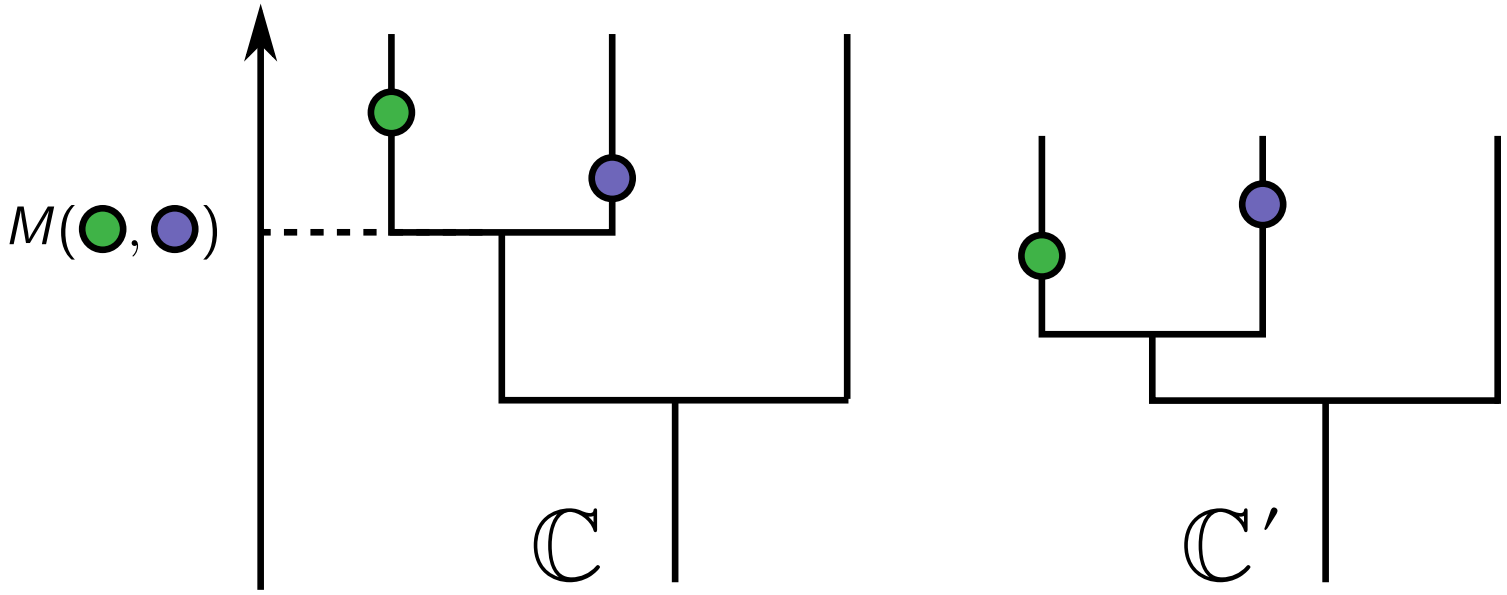
How "close" are $\mathbb{C}$ and $\mathbb{C}'$?

The merge distortion



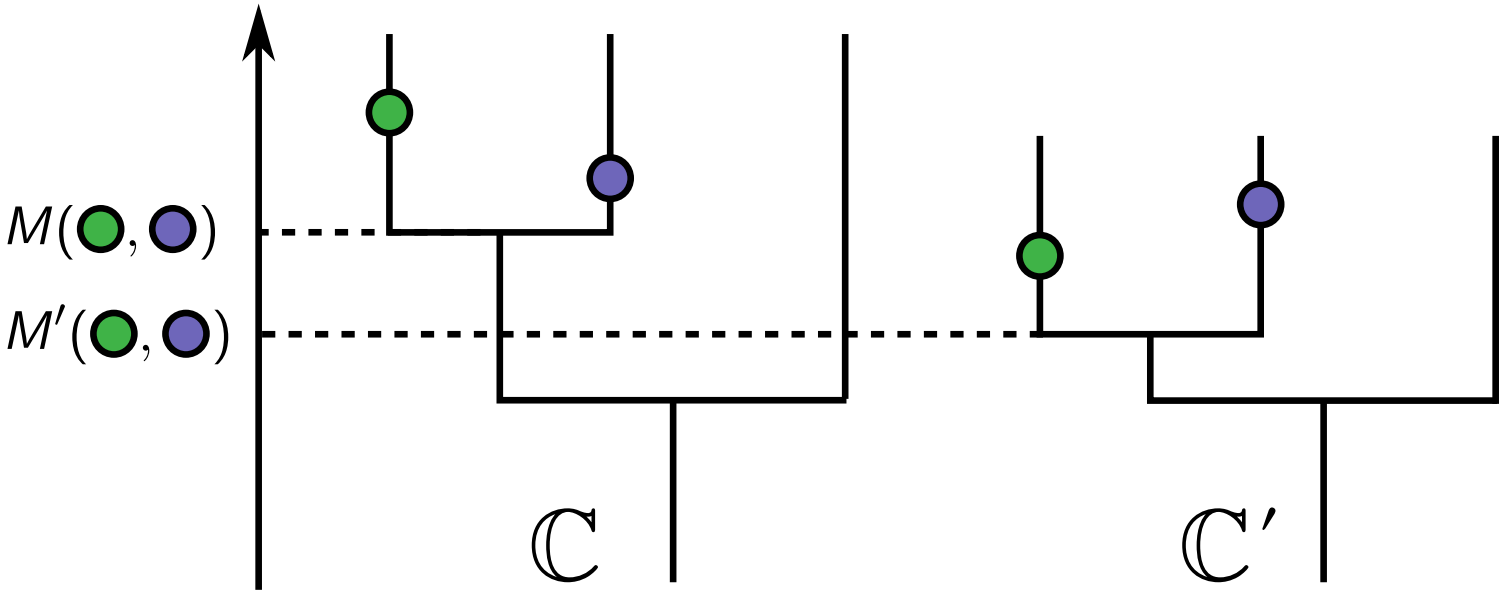
Intuitively, corresponding pairs of nodes should merge at around the same height in each tree.

The merge distortion



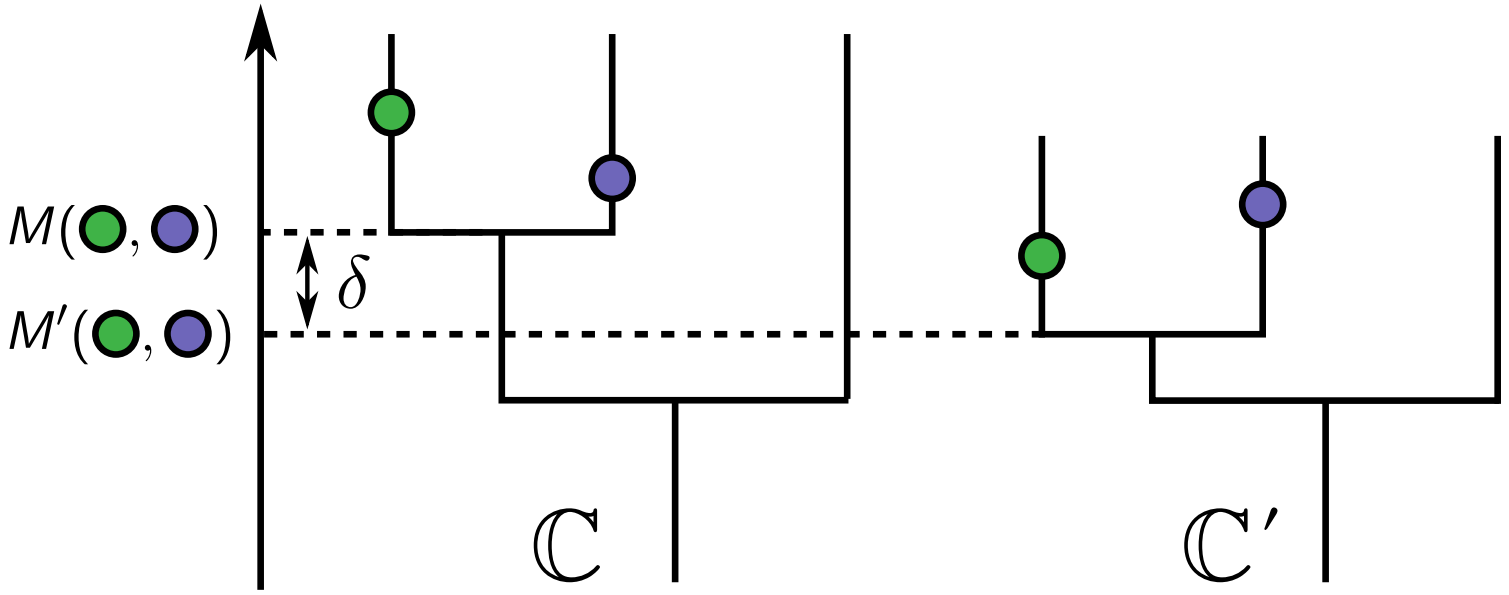
Merge heights are encoded in the mergeon.

The merge distortion



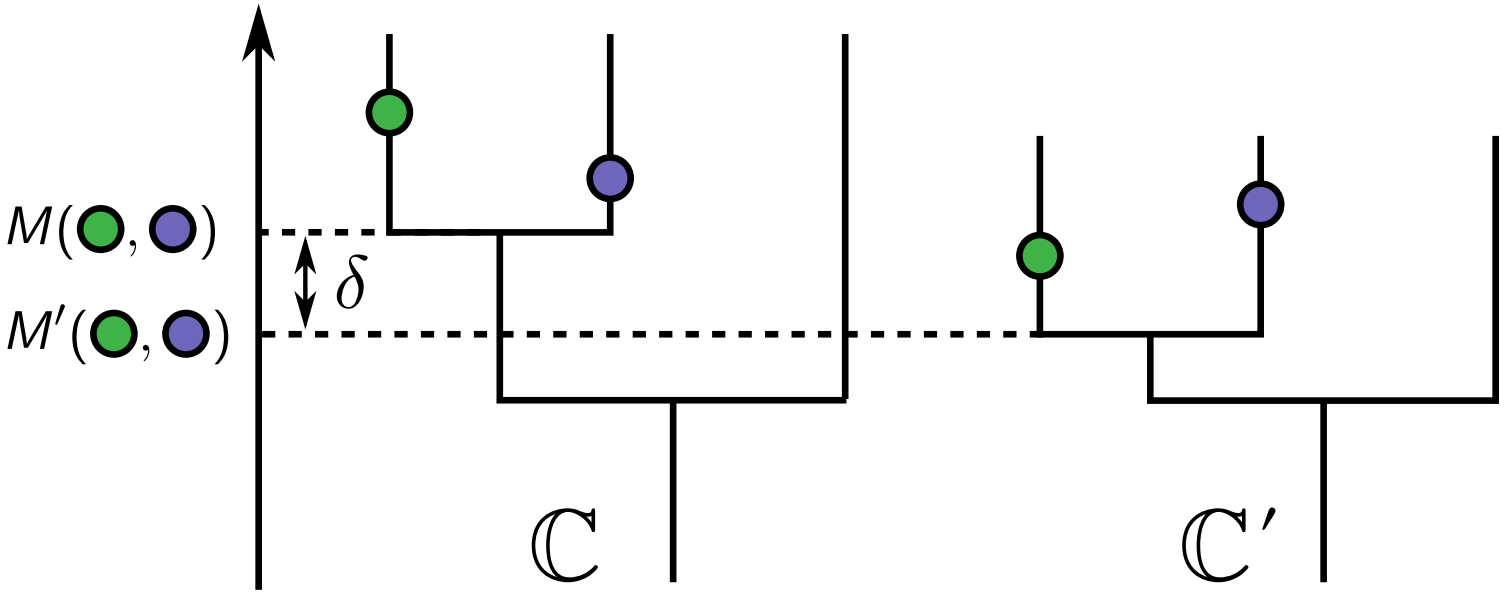
Merge heights are encoded in the mergeon.

The merge distortion



$|M(\bigcirc, \bigcirc) - M'(\bigcirc, \bigcirc)|$ is the difference in merge height of ●, ●.

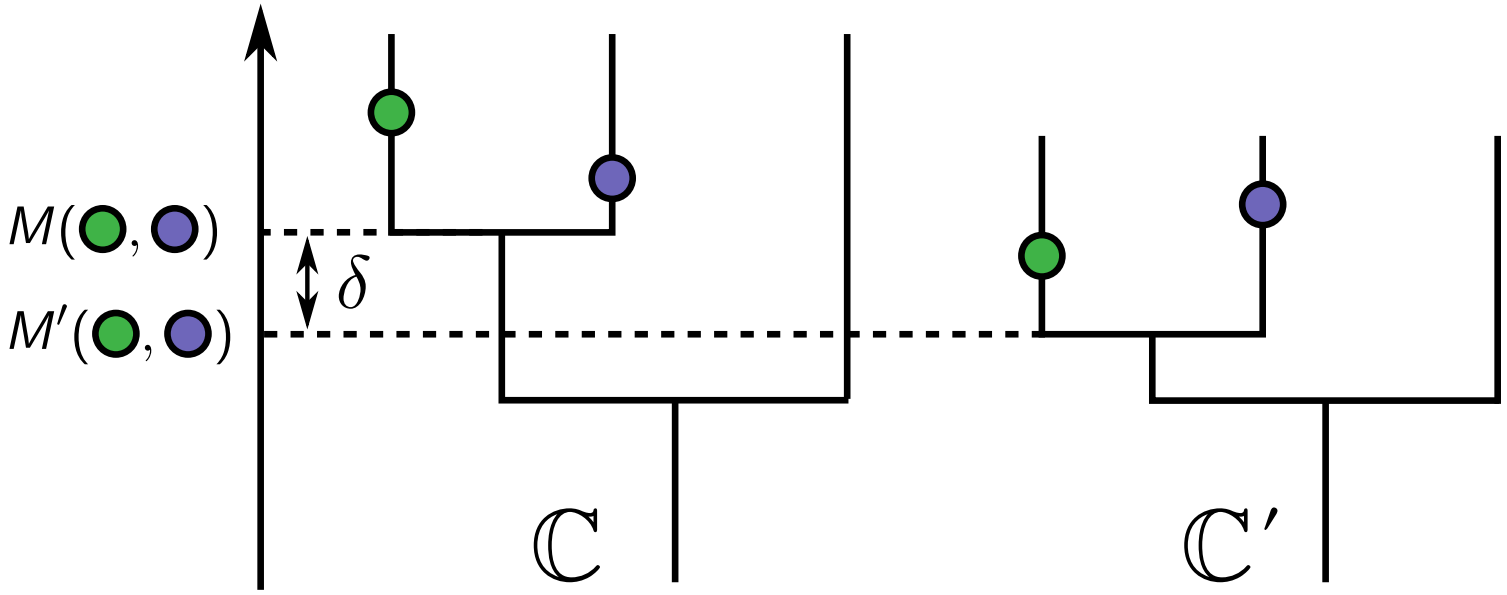
The merge distortion



The merge distortion $d(\mathbb{C}, \mathbb{C}')$:
the maximum difference in merge height over all pairs, i.e,

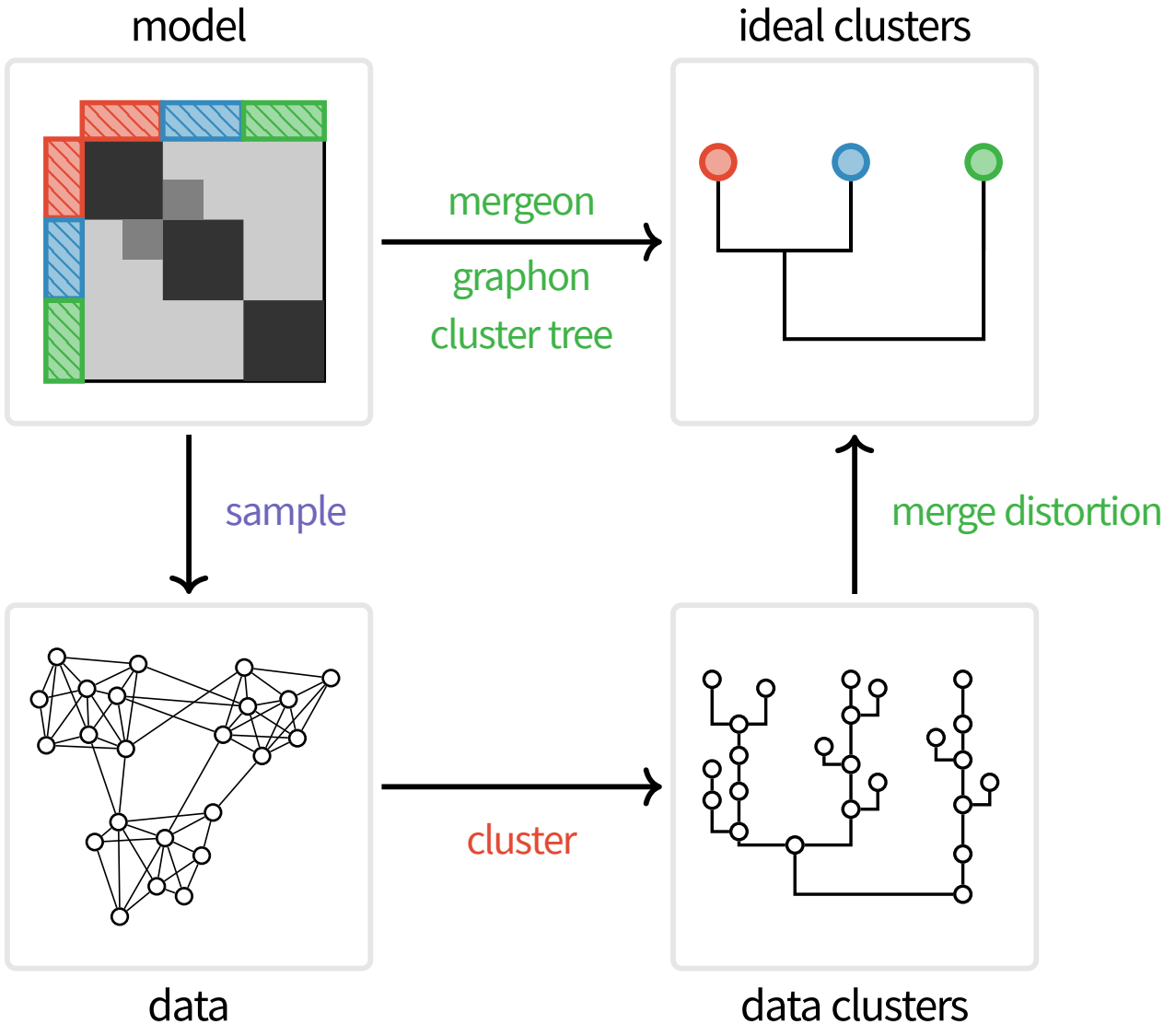
$$d(\mathbb{C}, \mathbb{C}') = \max_{\bullet,\bullet} \left| M(\bullet,\bullet) - M'(\bullet,\bullet) \right|.$$

The merge distortion

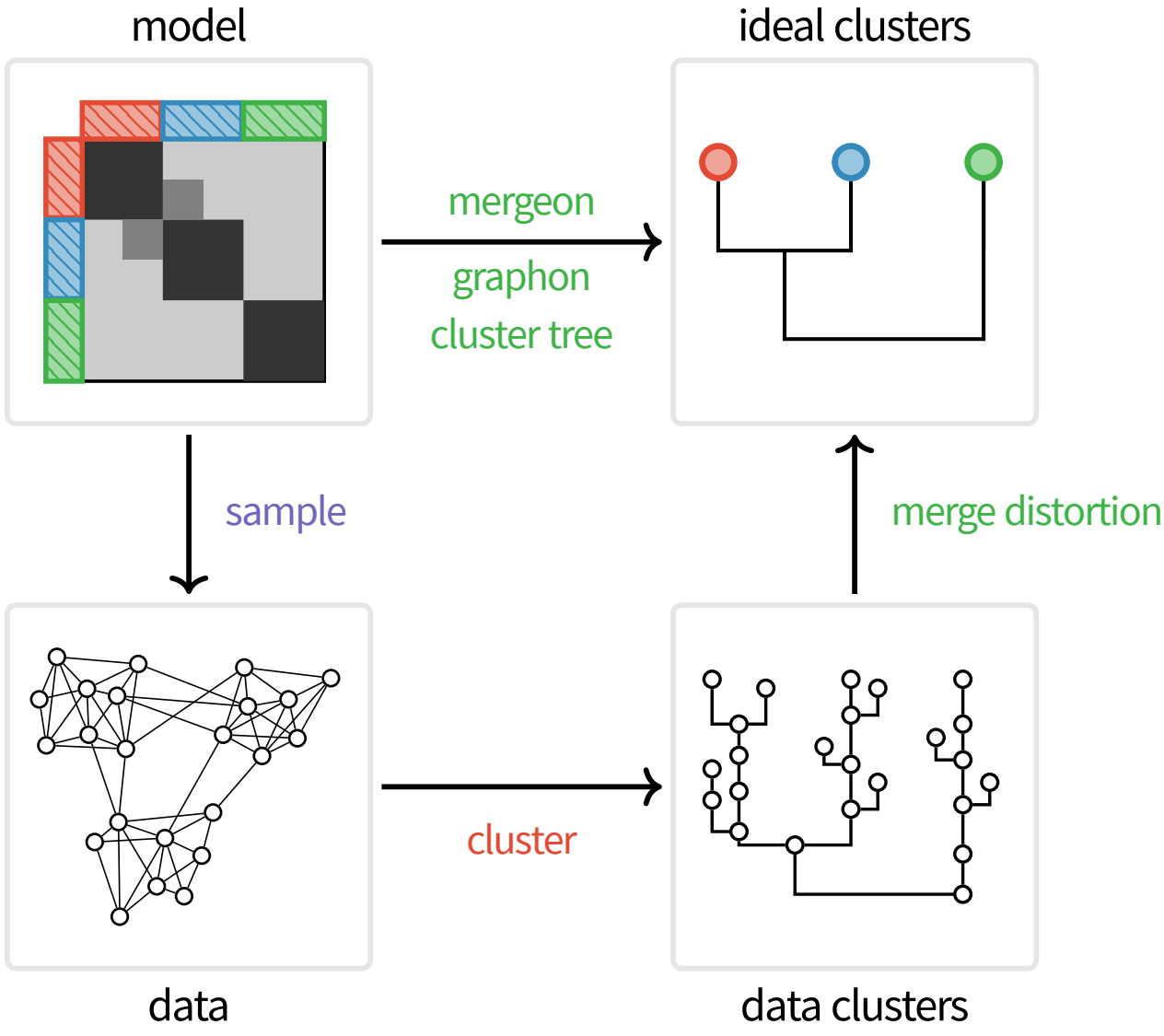


We define consistency as convergence to the graphon cluster tree in the merge distortion.

We have defined convergence in merge distortion for graphons.



Step 3: Does a clustering algorithm exist which converges?



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

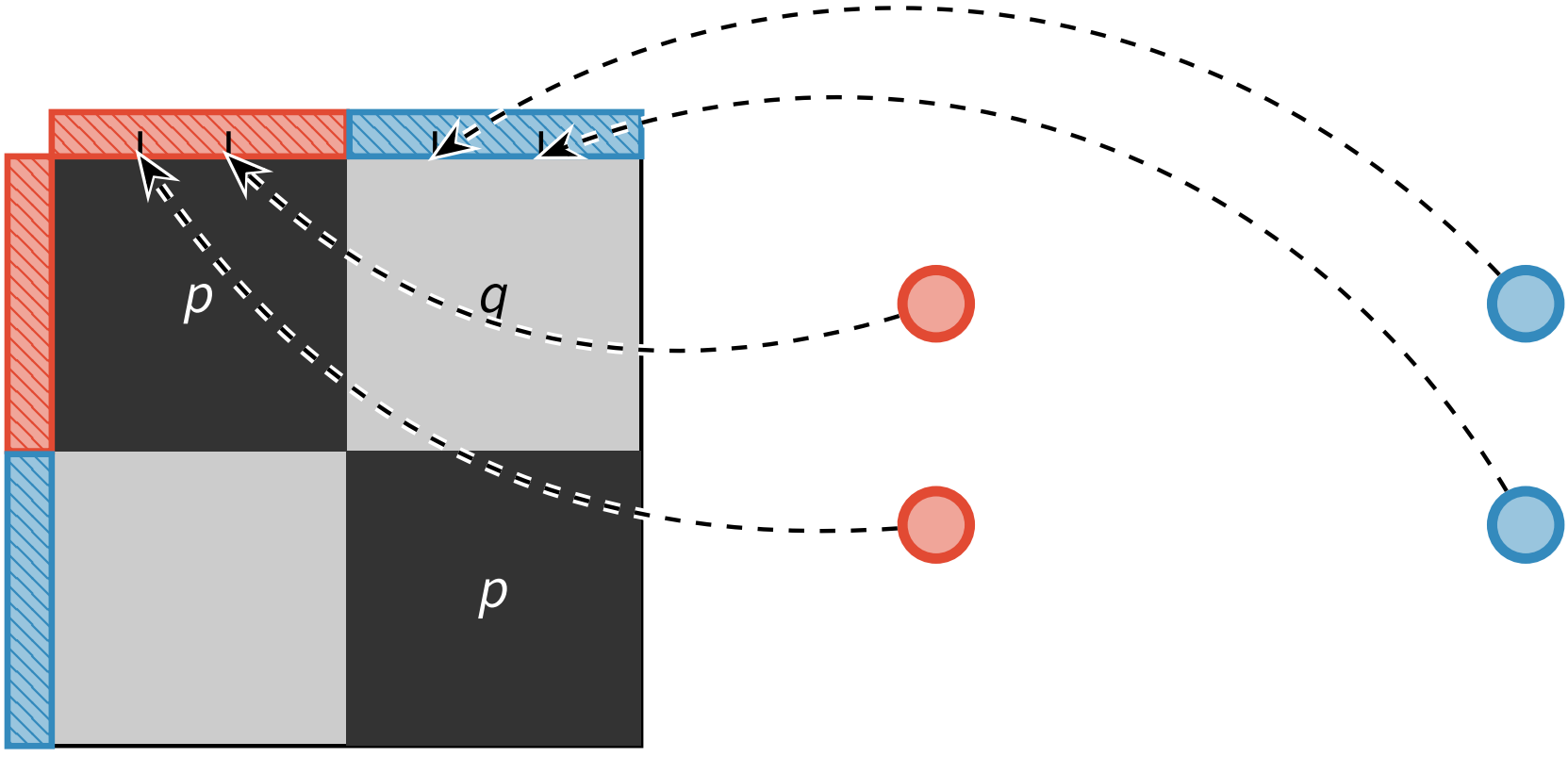
Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

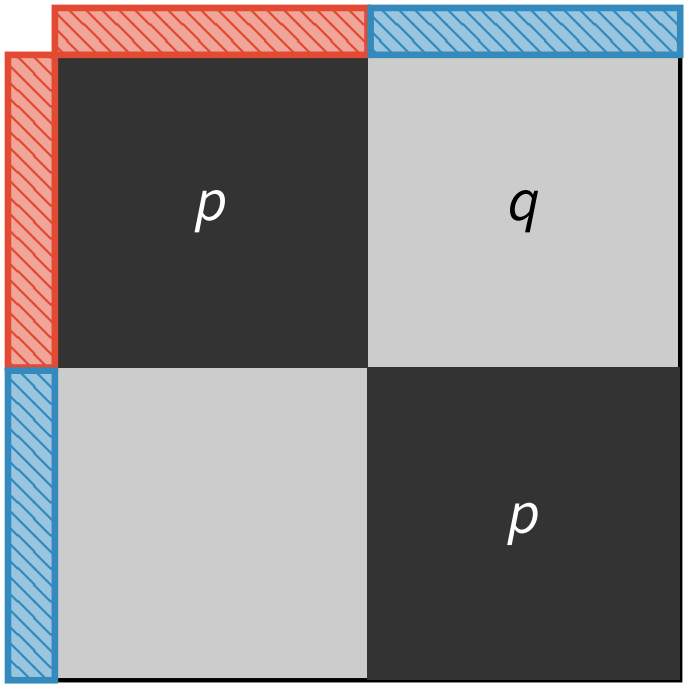
Applications & Directions

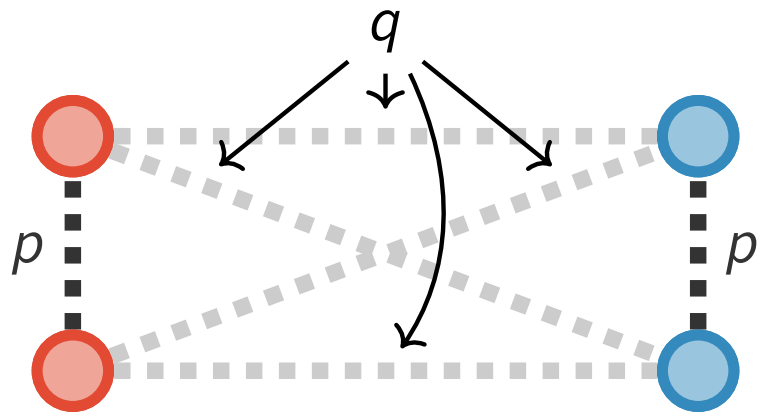
Estimating edge probabilities. Details



Suppose we sample a graph from this graphon.

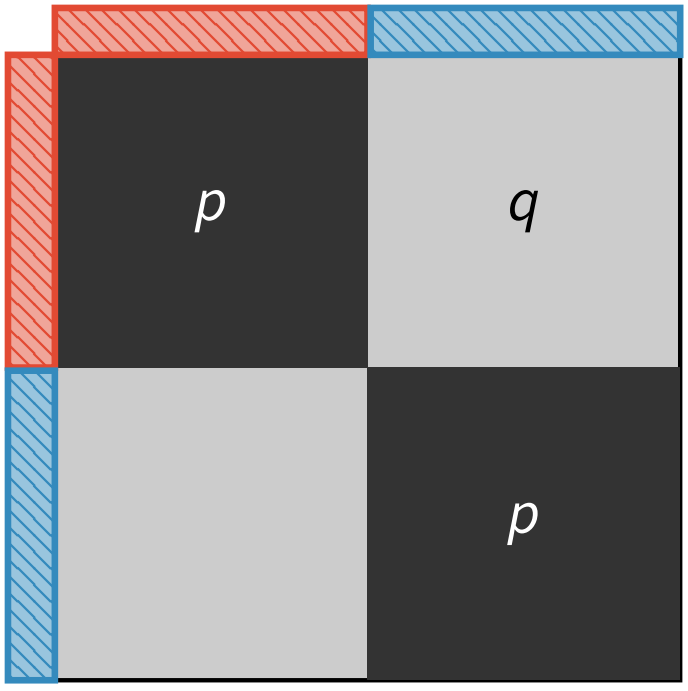
Estimating edge probabilities. Details

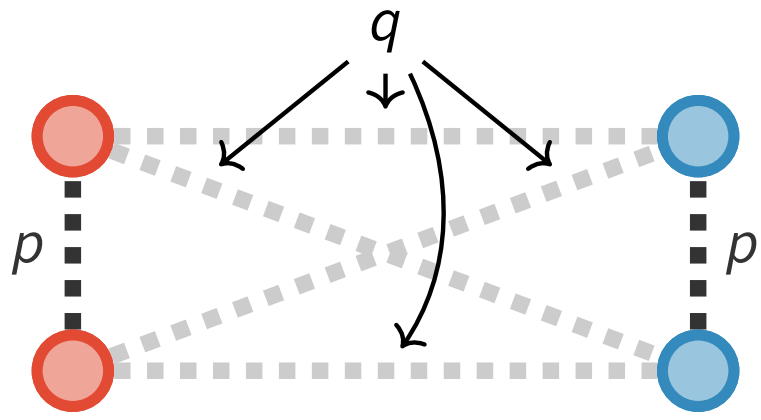




Edges within communities have probability p;
edges across communities have probability q.

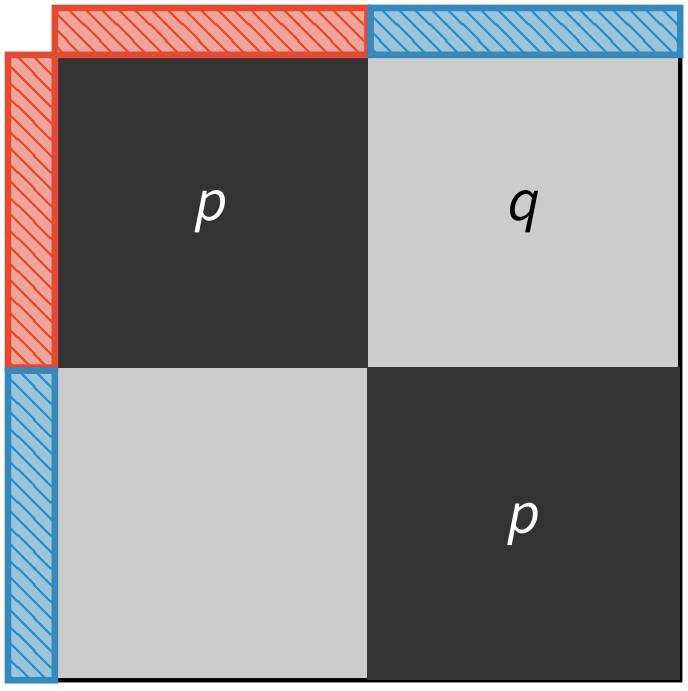
Estimating edge probabilities. Details

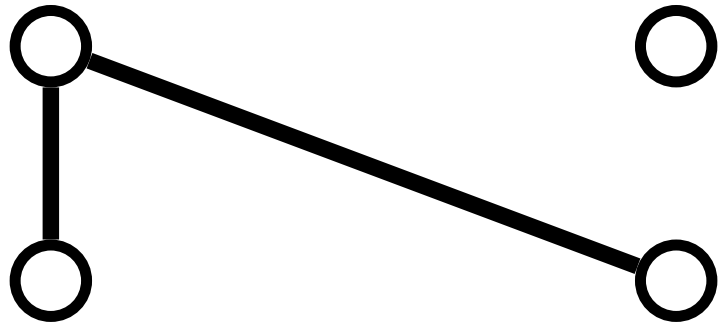




If we knew these edge probabilities we could recover the correct clusters.

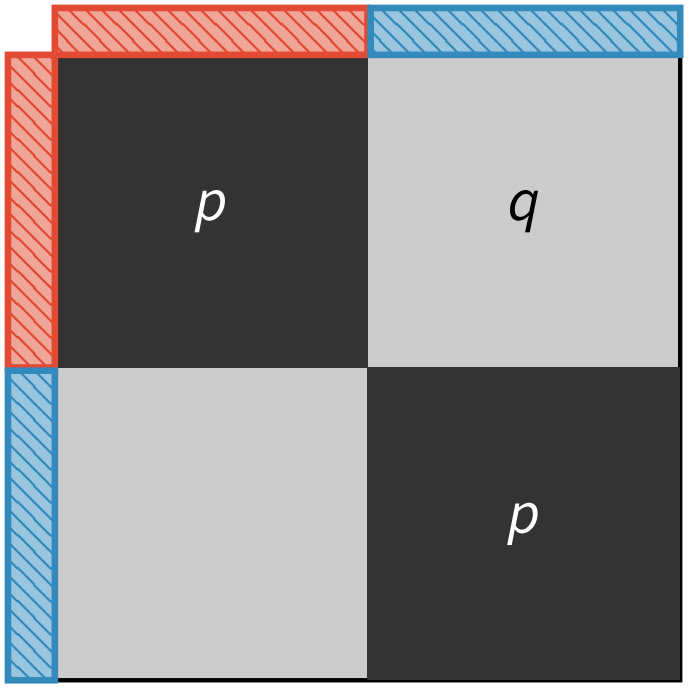
Estimating edge probabilities. Details

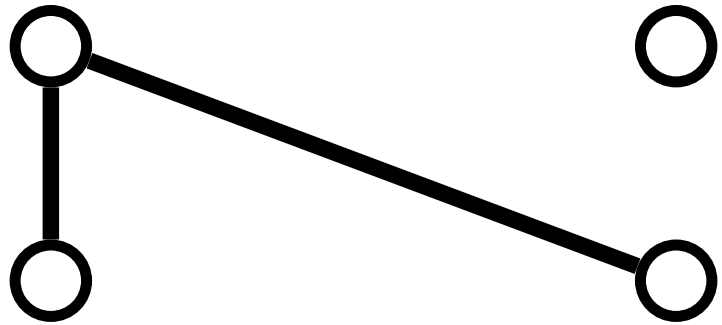




But the edge probabilities are unknown and the presence/absence of an edge (i, j) tells us little about its probability, P_{ij}.

Estimating edge probabilities. Details





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

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

Let $\hat{P}$ be a matrix of estimated edge probabilities,
and let P be the true edge probabilities.

Theorem (Eldridge, 2016):

If $\max_{ij} |\hat{P}_{ij} - P_{ij}| \to 0 \implies$
single-linkage on $\hat{P}$ converges to the graphon cluster tree.

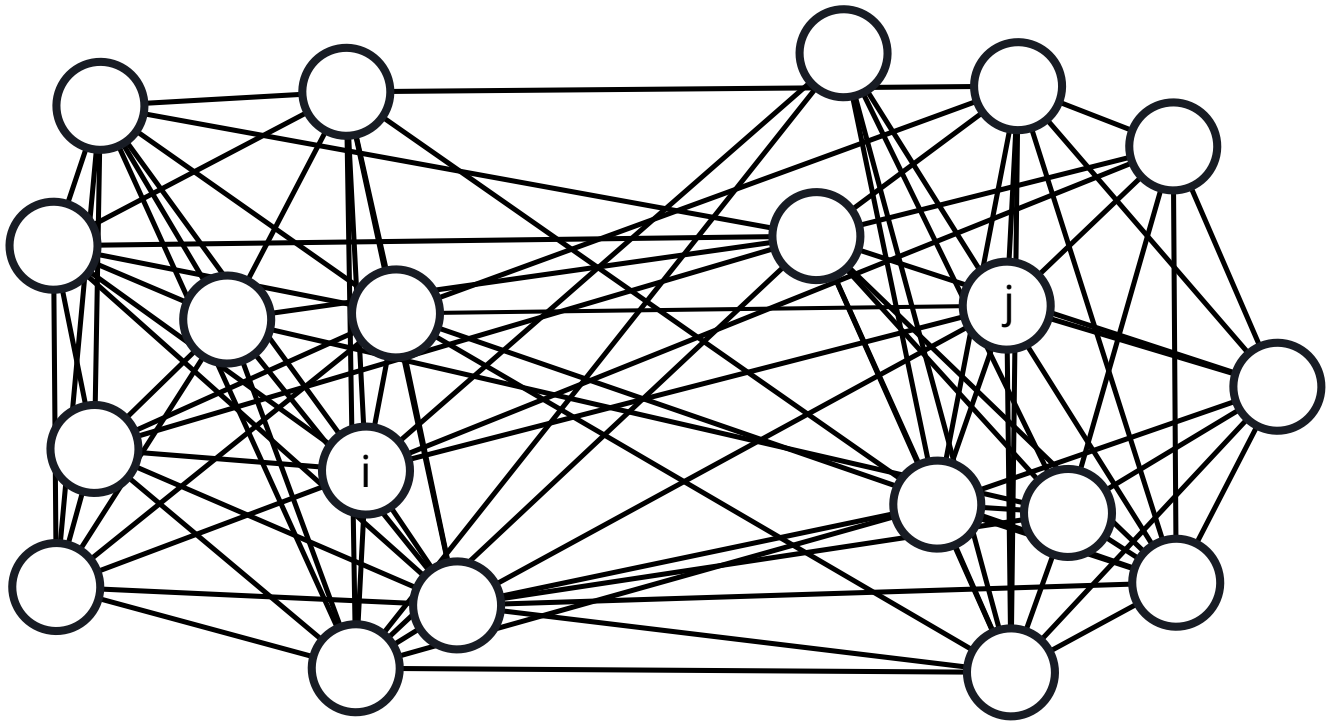
Let $\hat{P}$ be a matrix of estimated edge probabilities,
and let P be the true edge probabilities.

Theorem (Eldridge, 2016):

If $\max_{ij} |\hat{P}_{ij} - P_{ij}| \to 0 \implies$
single-linkage on $\hat{P}$ converges to the graphon cluster tree.

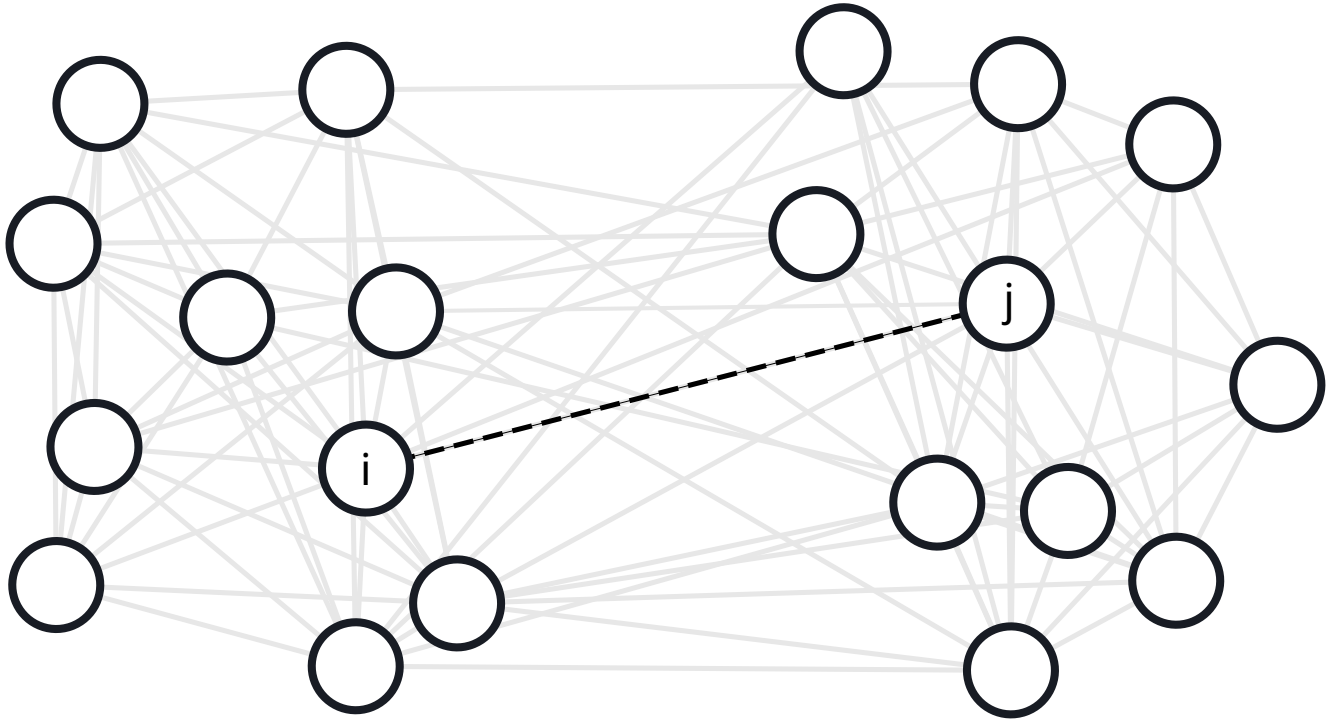
- There are many recent graphon edge probability estimators.
- But all analyses are in aggregate error:
  - i.e., they allow a small # of edge probability estimates to be bad.
- These results are too weak for our purposes.
- We modify and analyze the neighborhood smoothing method of (Zhang et al., 2015) to obtain consistency in max-norm.

Neighborhood smoothing



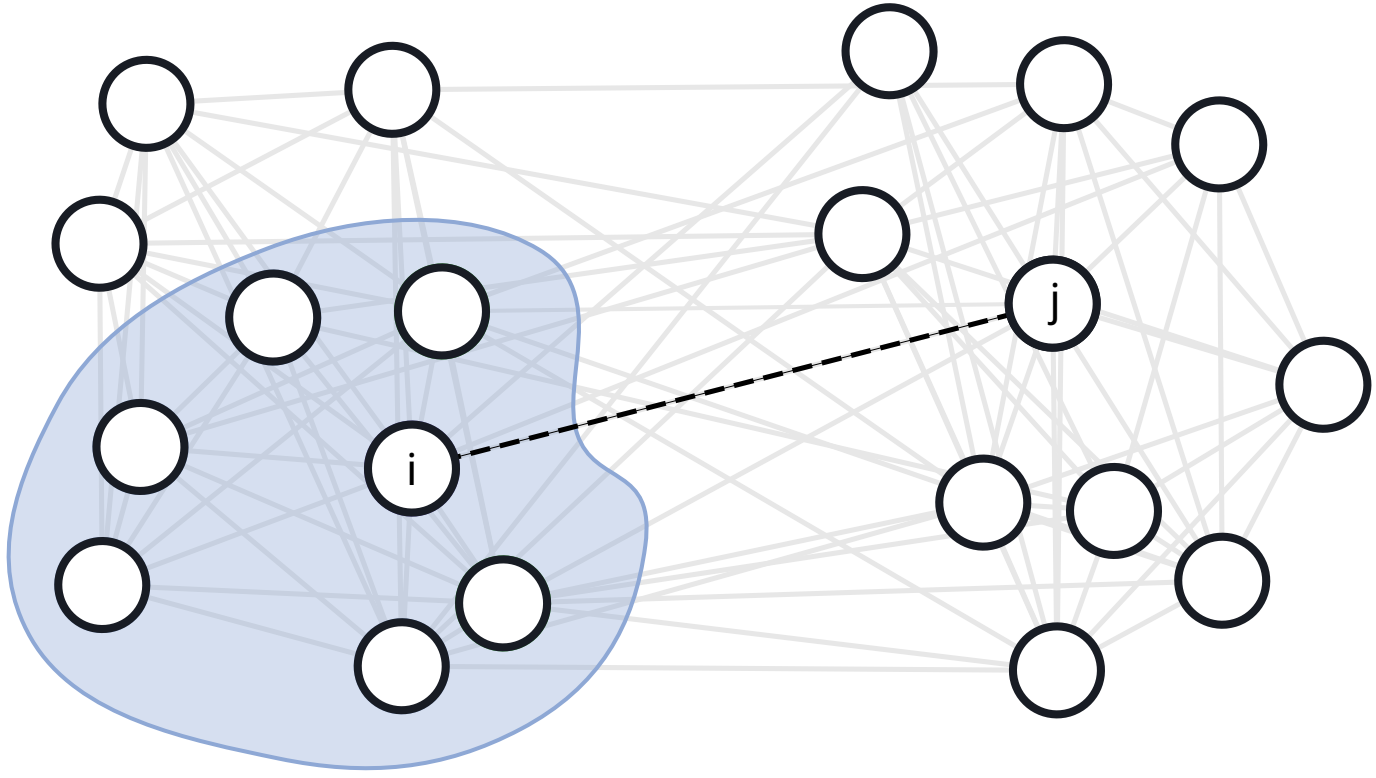
Given this graph...

Neighborhood smoothing



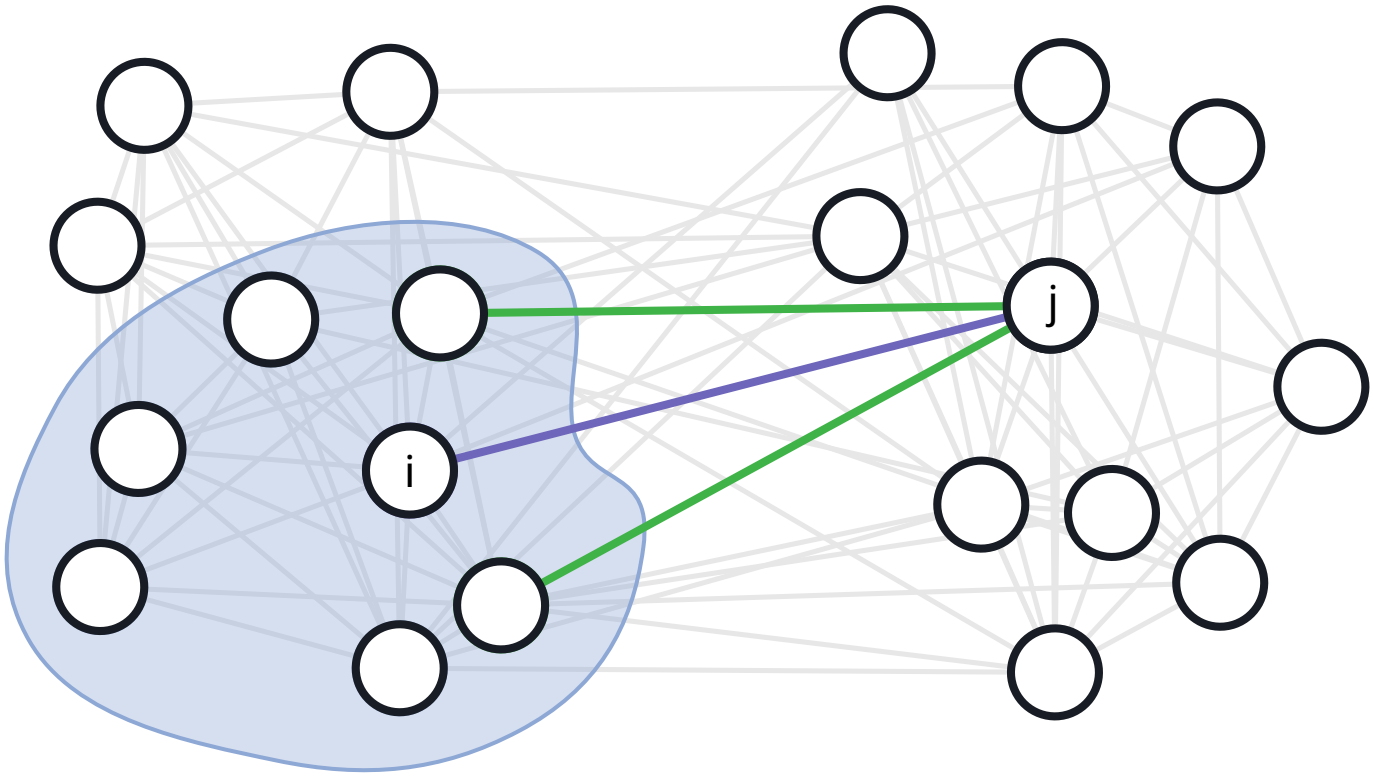
Given this graph... estimate P_{ij}.

Neighborhood smoothing



Build a neighborhood N_i of nodes with similar connectivity to that of i.

Neighborhood smoothing



- Average number edges from node in neighborhood N_i to j.
- Estimated edge probability: $\hat{P}_{ij} = 2/6 = 1/3$.

Theorem (Eldridge, 2016):

Our modified neighborhood smoothing edge probability estimator for P is consistent in max-norm.

Theorem (Eldridge, 2016):

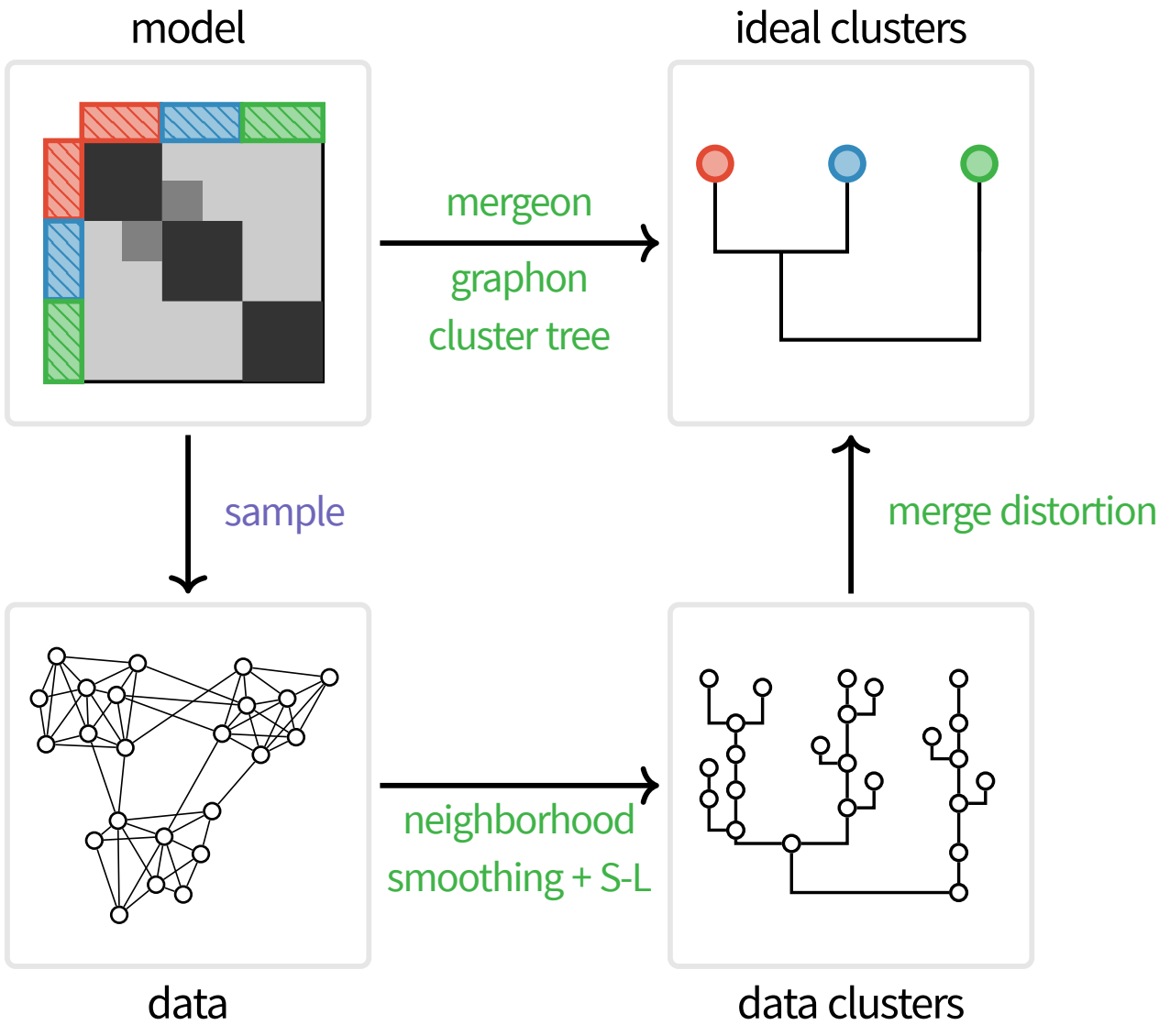
Our modified neighborhood smoothing edge probability estimator for P is consistent in max-norm.

Corollary (Eldridge, 2016):

Consistent graphon clustering method:

1. Estimate edge probabilities with our modified neighborhood smoother.
2. Apply single-linkage clustering to estimated edge probabilities.

Clustering in the stochastic blockmodel.



Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

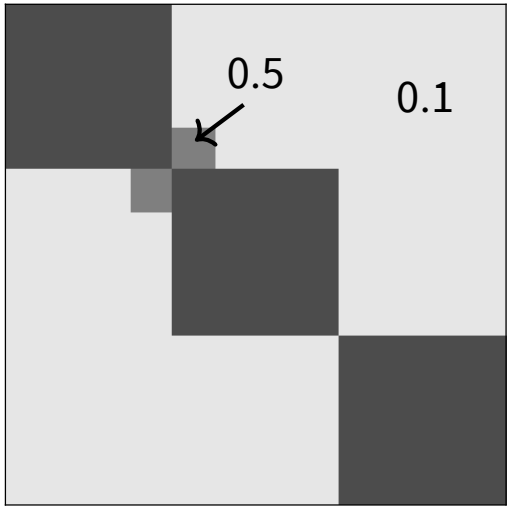
Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

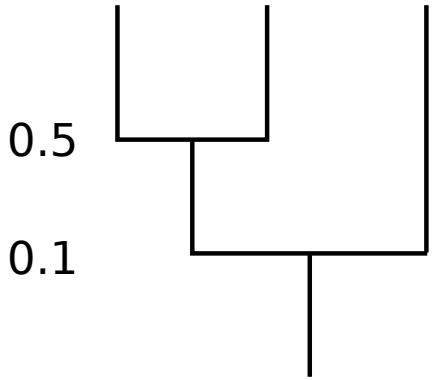
- Background: stochastic blockmodel
- The graphon model
- The clusters of a graphon
- Merge distortion for graphons
- Converging clustering methods
- Examples

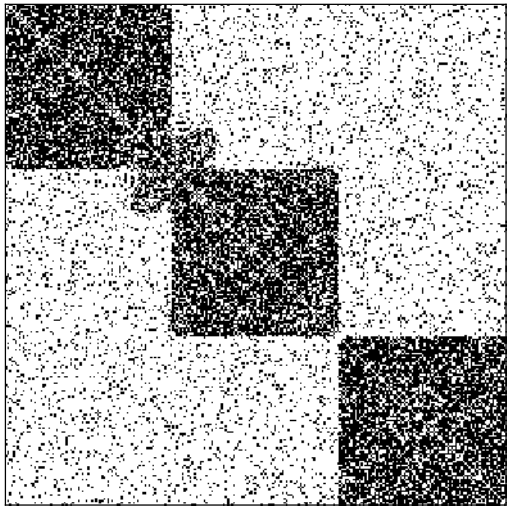
Applications & Directions

Example

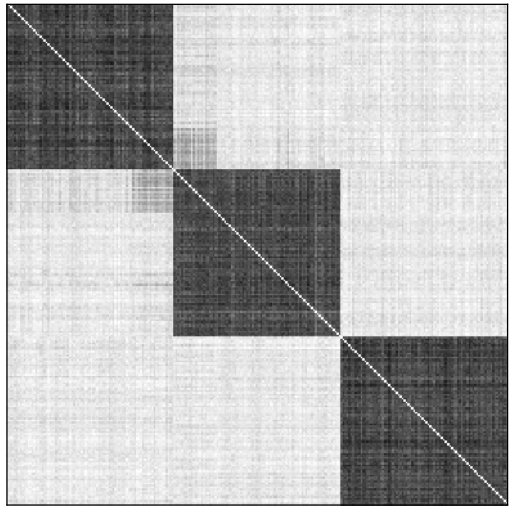
graphon W







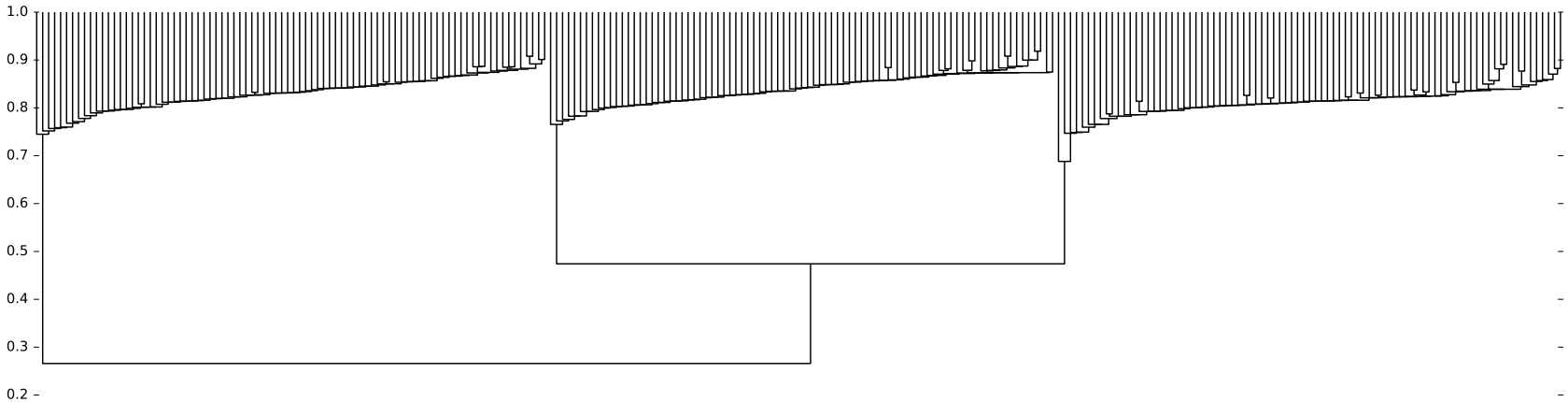
adjacency A



estimated edge probs. $\hat{P}$

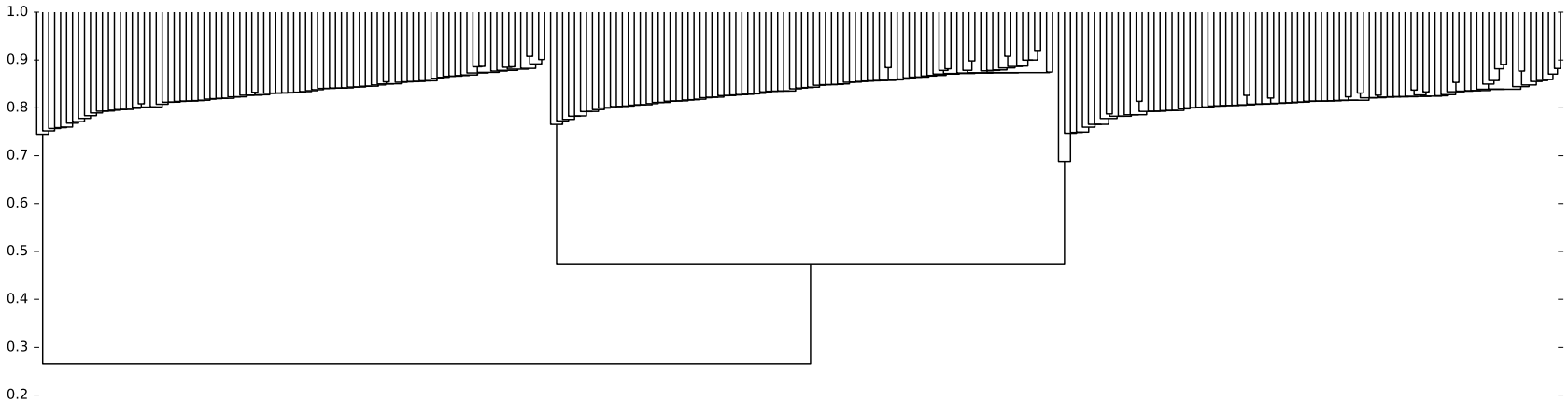
Example

Our method: Neighborhood smoothing + single-linkage

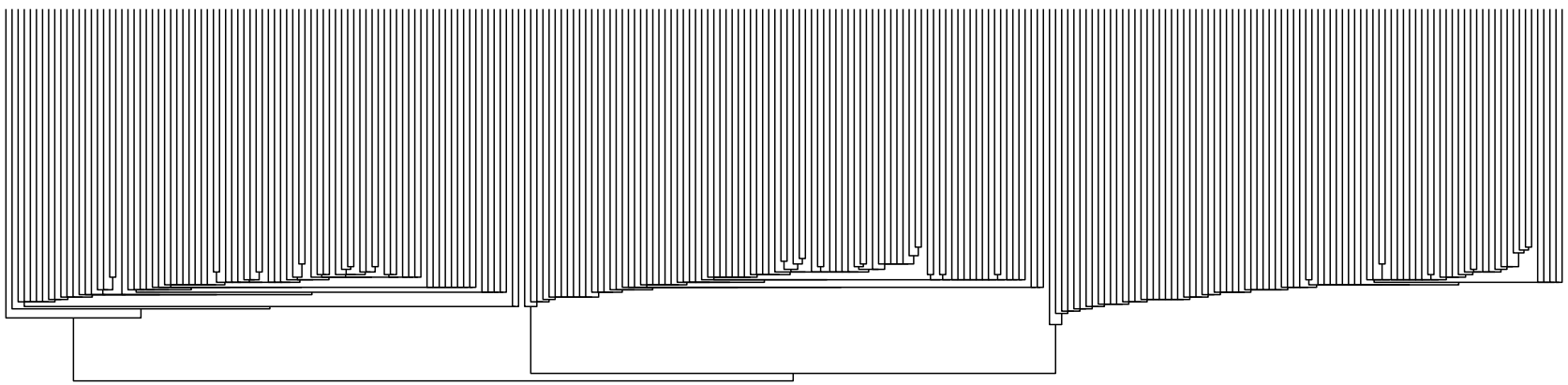


Example

Our method: Neighborhood smoothing + single-linkage

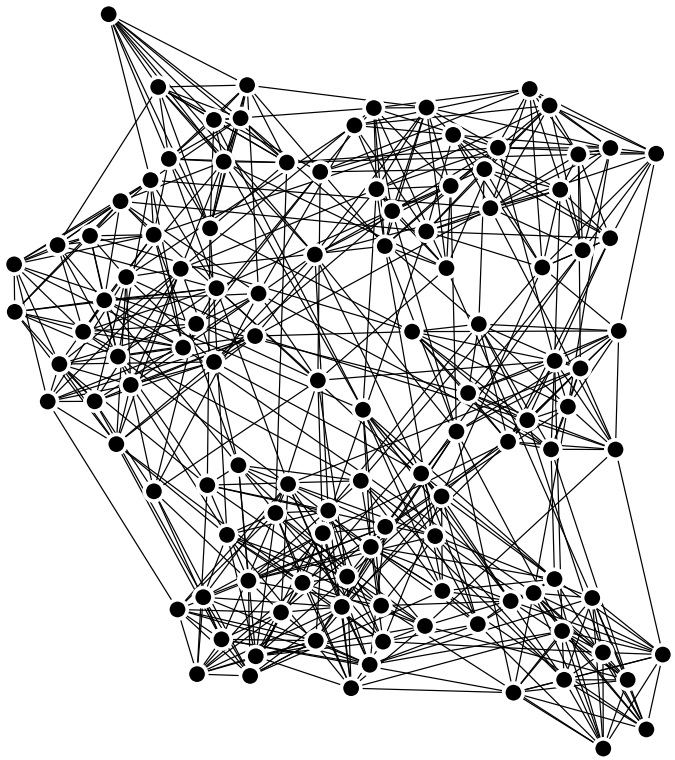


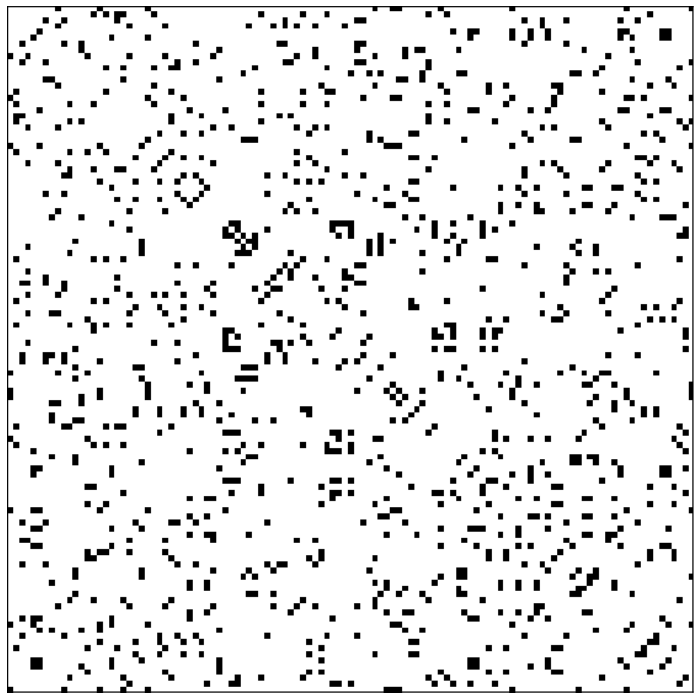
Single linkage



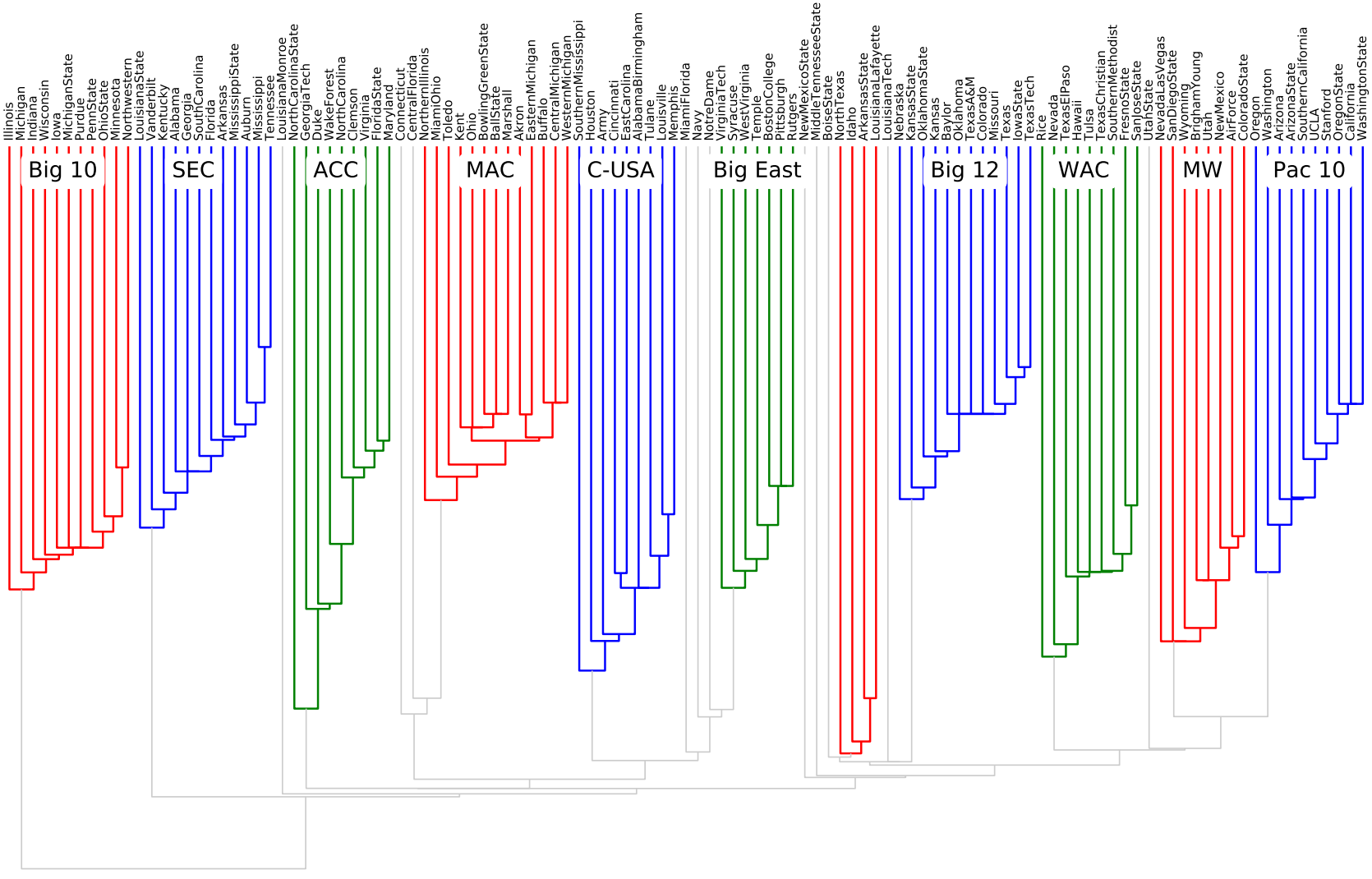
Example

Network of college football games in 2001.





Each node is a team, an edge exists if teams played.


Illinois
Michigan
Indiana
Wisconsin
Iowa
MichiganState
Purdue
PennState
OhioState
Minnesota
Northwestern
LouisianaState
Vanderbilt
Kentucky
Alabama
Georgia
SouthCarolina
Florida
Arkansas
MississippiState
Auburn
Mississippi
Tennessee
LouisianaMonroe
NorthCarolinaState
GeorgiaTech
Duke
WakeForest
NorthCarolina
Clemson
Virginia
FloridaState
Maryland
Connecticut
CentralFlorida
NorthernIllinois
MiamiOhio
Toledo
Kent
Ohio
BowlingGreenState
BallState
Marshall
Akron
EasternMichigan
Buffalo
CentralMichigan
WesternMichigan
SouthernMississippi
Houston
Army
Cincinnati
EastCarolina
AlabamaBirmingham
Tulane
Louisville
Memphis
MiamiFlorida
Navy
NotreDame
VirginiaTech
Syracuse
WestVirginia
Temple
BostonCollege
Pittsburgh
Rutgers
NewMexicoState
MiddleTennesseeState
BoiseState
NorthTexas
Idaho
ArkansasState
LouisianaLafayette
LouisianaTech
Nebraska
KansasState
OklahomaState
Kansas
Baylor
Oklahoma
TexasA&M
Colorado
Missouri
Texas
IowaState
TexasTech
Rice
Nevada
TexasElPaso
Hawaii
Tulsa
TexasChristian
SouthernMethodist
FresnoState
SanJoseState
UtahState
NevadaLasVegas
SanDiegoState
Wyoming
BrighamYoung
Utah
NewMexico
AirForce
ColoradoState
Oregon
Washington
ArizonaState
Arizona
SouthernCalifornia
UCLA
Stanford
OregonState
California
WashingtonState
Big 10
SEC
ACC
MAC
C-USA
Big East
Big 12
WAC
MW
Pac 10

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

Recap:

- The graphon is a rich model
- Virtually unstudied w.r.t. clustering
- Define clusters and convergence
- Prove nbhd. smoothing + S-L converges

Context

Approaches to “correct” clustering.

Part I: Density

Beyond Hartigan Consistency @ COLT 2015
Awarded best student paper.

Part II: Graphon

Graphons, mergeons, and so on! @ NIPS 2016
Awarded full oral presentation.

Applications & Directions

Practical application: Choosing a clustering algorithm

- Goal: recover regions of high density
  Example: Group customers according to previous purchases.
  - Consider algorithms which converge to density cluster tree.
  - Robust single-linkage.

Practical application: Choosing a clustering algorithm

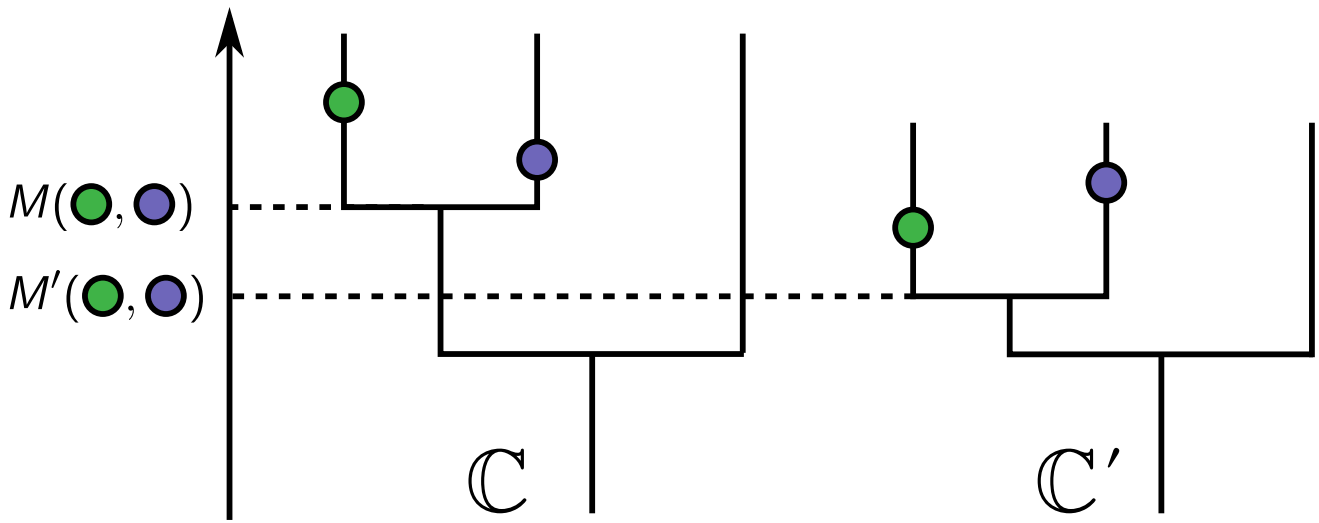
- Goal: recover regions of high density
  Example: Group customers according to previous purchases.
  - Consider algorithms which converge to density cluster tree.
  - Robust single-linkage.
- Goal: recover cluster structure of a graph
  Example: Find communities in social network.
  - Consider algorithms which converge to graphon cluster tree.
  - Neighborhood smoothing + single-linkage.

Practical application: Choosing a clustering algorithm

- Goal: recover regions of high density
  Example: Group customers according to previous purchases.
  - Consider algorithms which converge to density cluster tree.
  - Robust single-linkage.
- Goal: recover cluster structure of a graph
  Example: Find communities in social network.
  - Consider algorithms which converge to graphon cluster tree.
  - Neighborhood smoothing + single-linkage.
- Quite likely that density/graphon is a good fit to data.
- We guarantee strong convergence, no oversegmentation.
- Naturally discuss statistical significance of clusters.

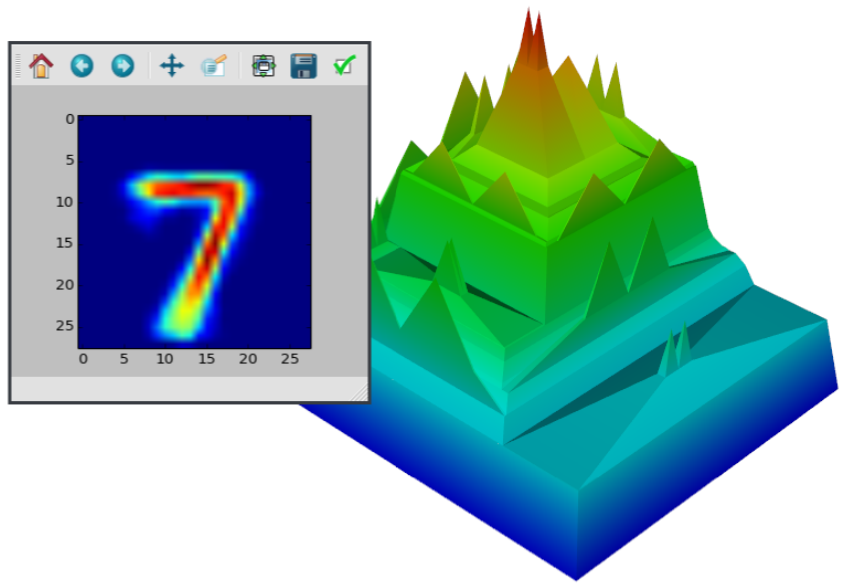
Practical application: Comparing clusterings quantitatively

- We'd often like to compare clusterings:
  - When do we stop our iterative algorithm?
  - How far are we from a partial clustering provided by a teacher?
- We provide the (efficiently computable) merge distortion:



Practical application: Visualizing high-dimensional densities

- Often have high-dimensional data which we can't simply plot.
- Important aspects of data's structure can be lost when projected.
- Approach: Cluster by estimating the density cluster tree & visualize.
- We have developed Denali[2] tree visualization software:



[2] http://denali.cse.ohio-state.edu

Direction: Formalizing correctness.

How do we combine approaches to formalizing clustering?

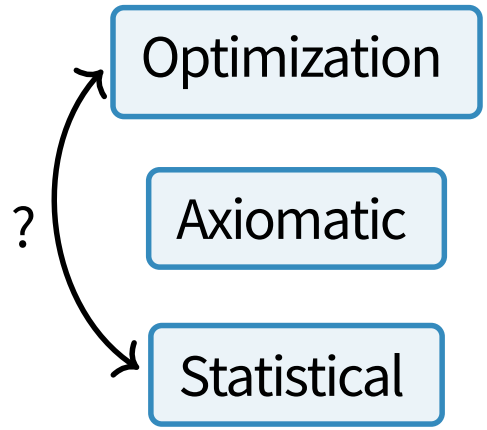
Optimization

Axiomatic

Statistical

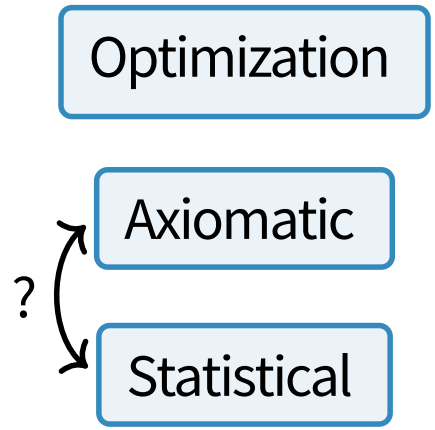
Direction: Formalizing correctness.

How do we combine approaches to formalizing clustering?



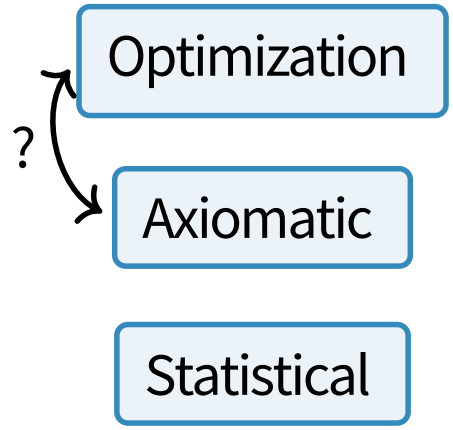
Direction: Formalizing correctness.

How do we combine approaches to formalizing clustering?



Direction: Formalizing correctness.

How do we combine approaches to formalizing clustering?



Direction: Formalizing correctness.

How do we combine approaches to formalizing clustering?

Optimization

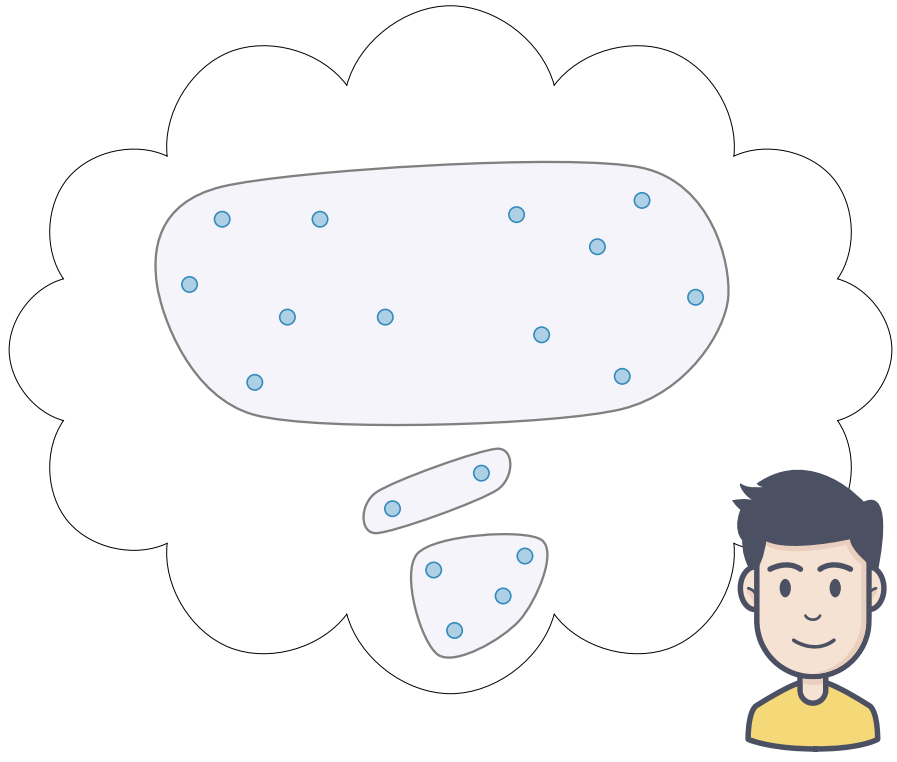
Axiomatic

Statistical

Practical importance:

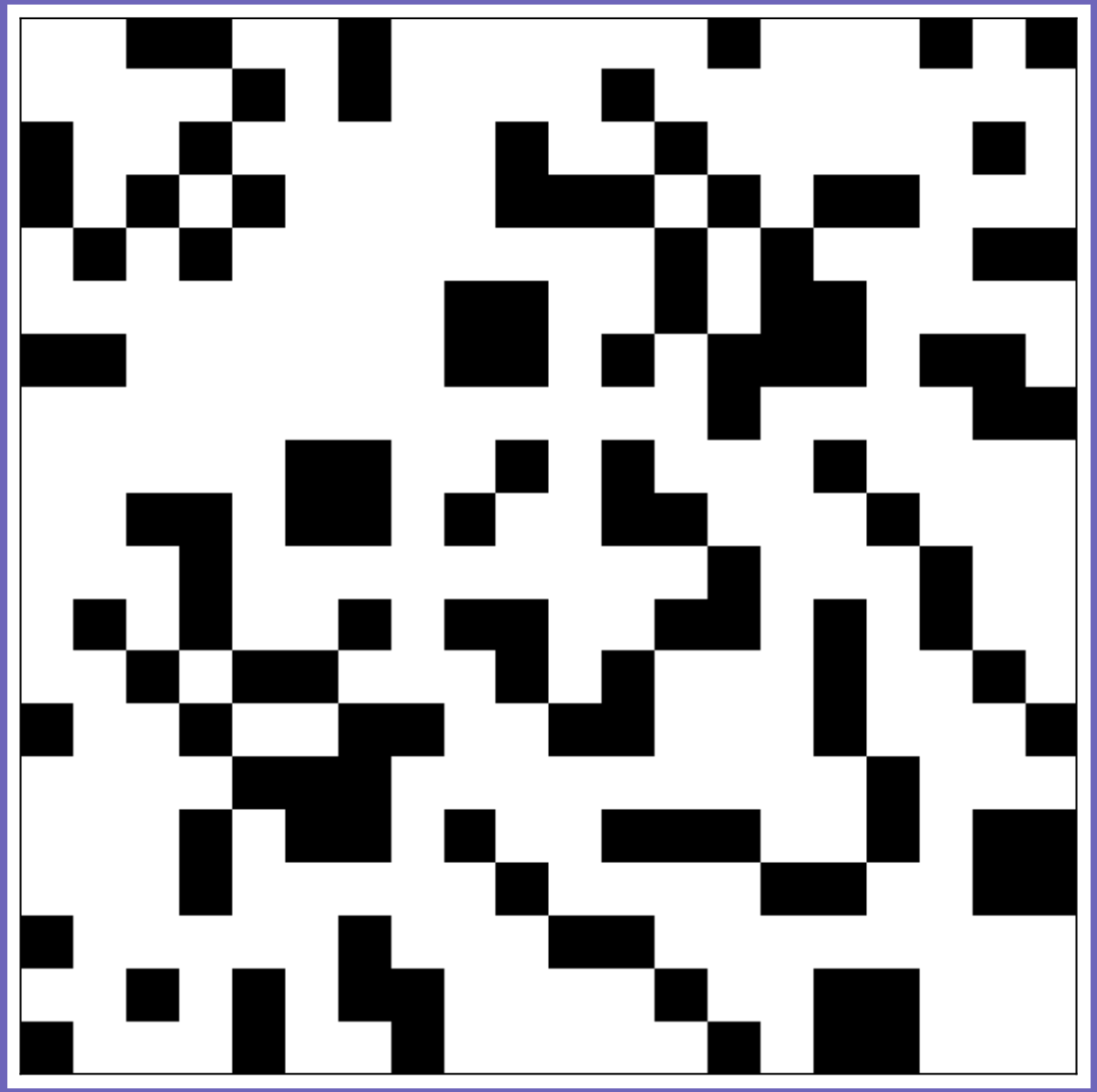
- Many popular clustering methods are not framed statistically.
- What do their clusters converge to?
- How do we interpret them?

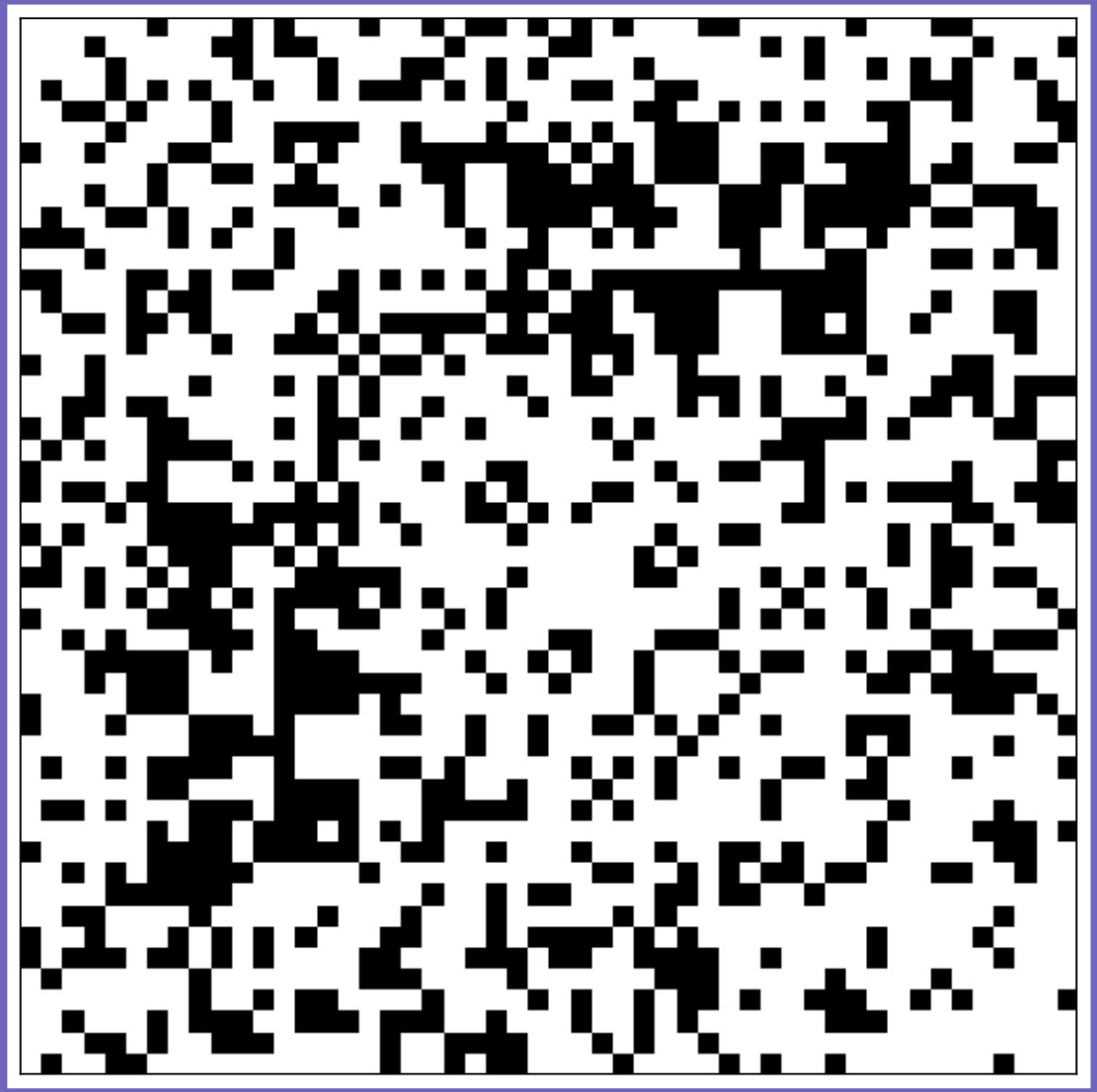
Direction: Interactive clustering

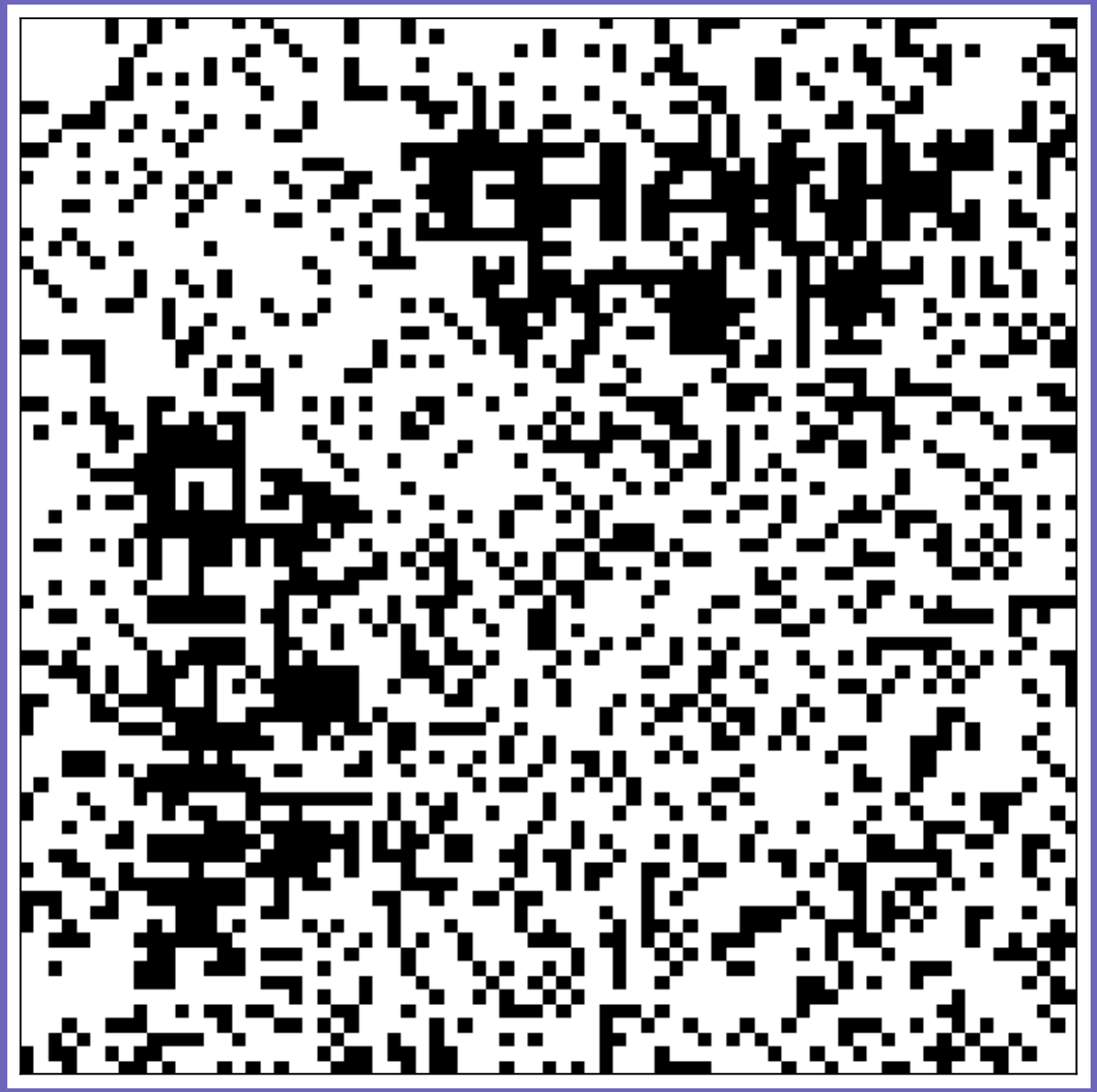


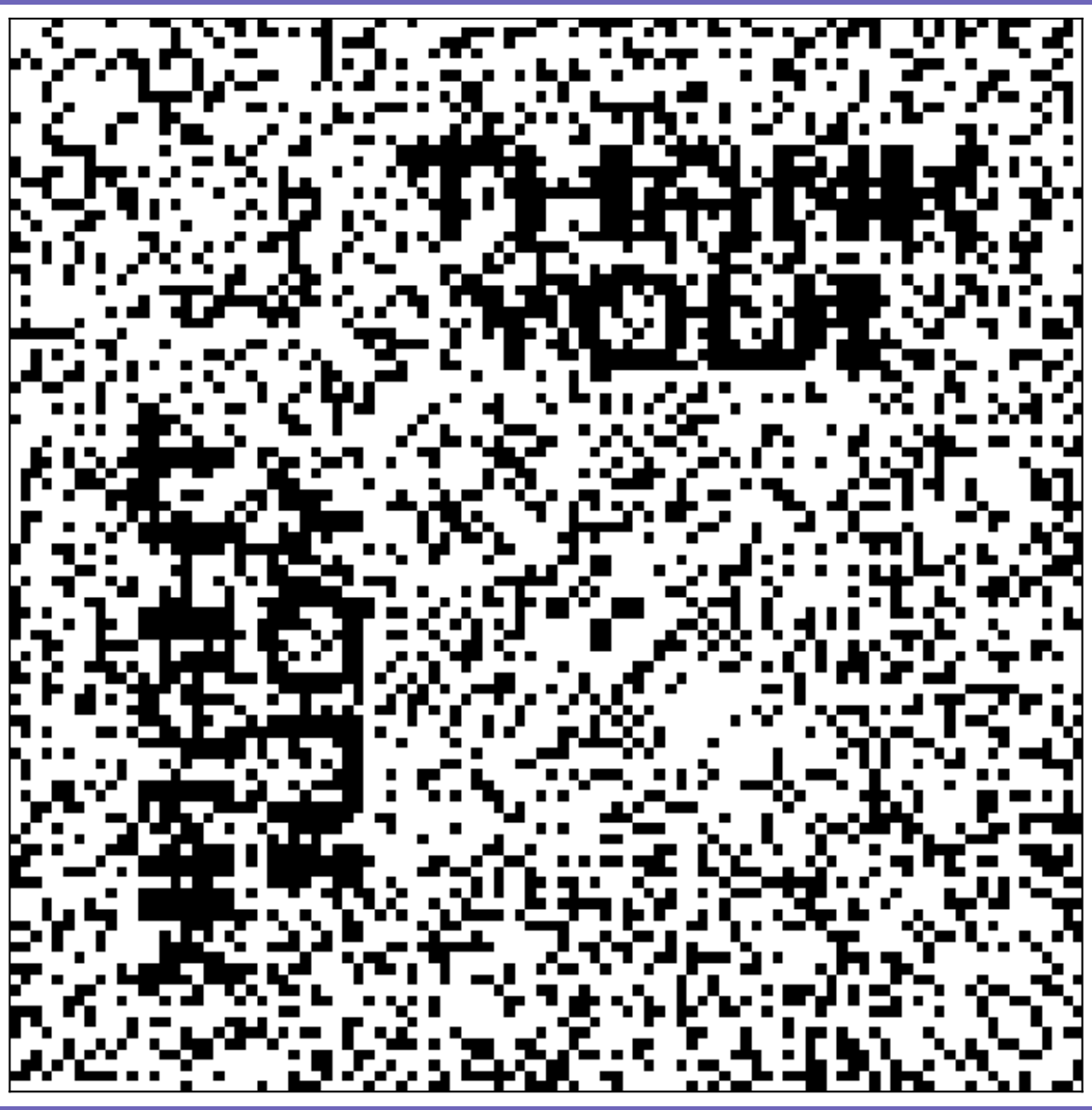
How do we formalize correctness?

“correct” clustering

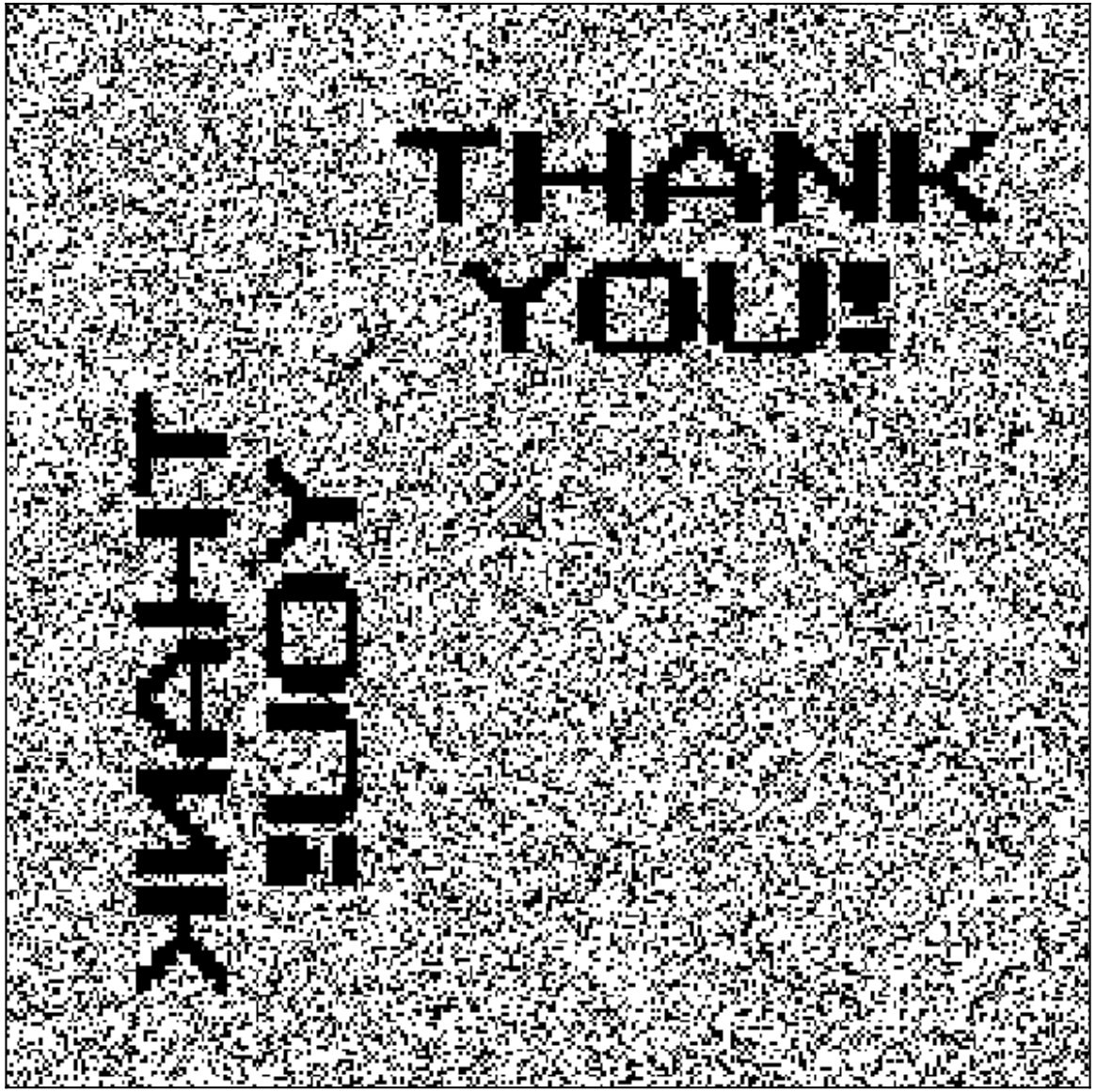


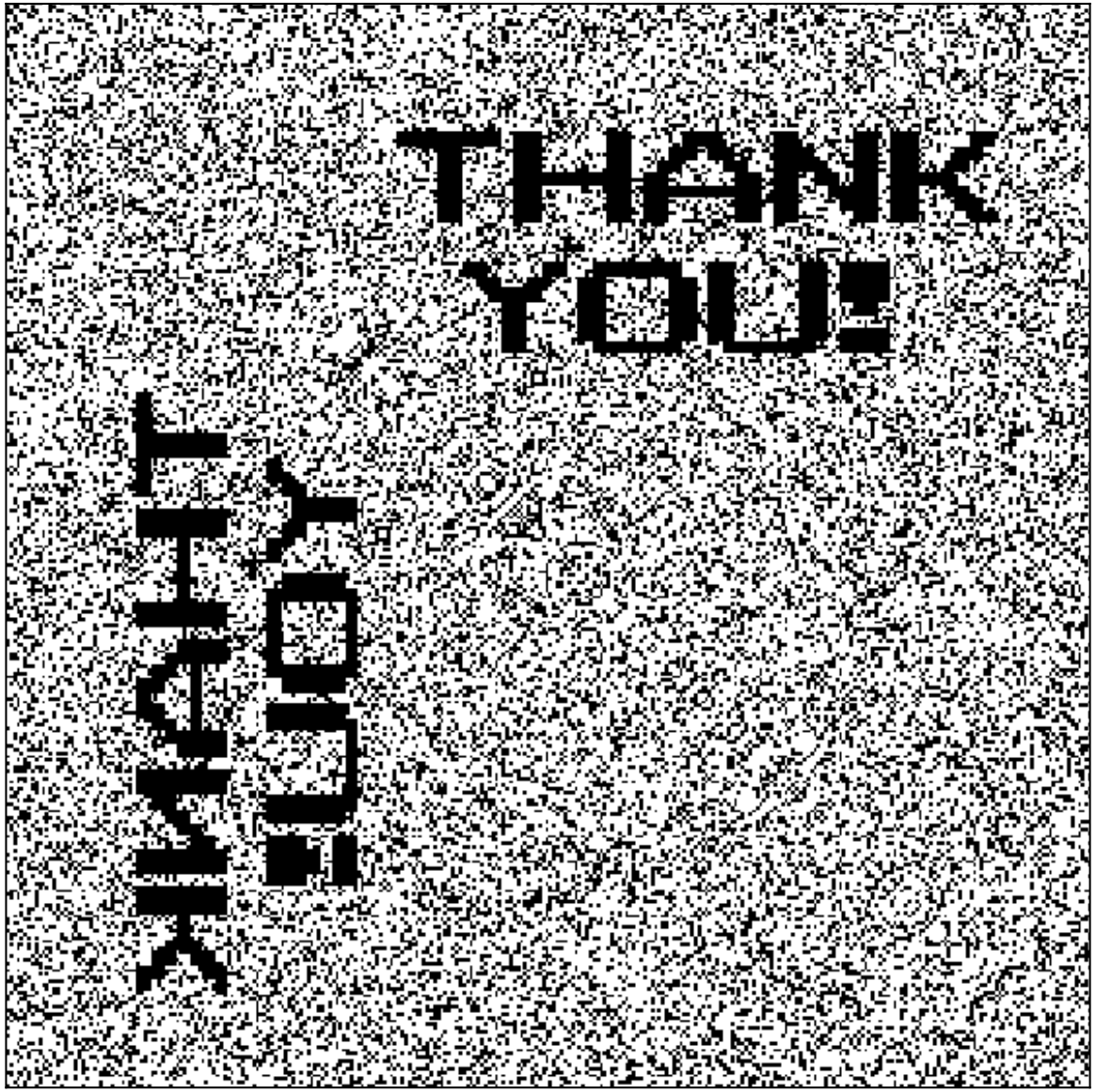


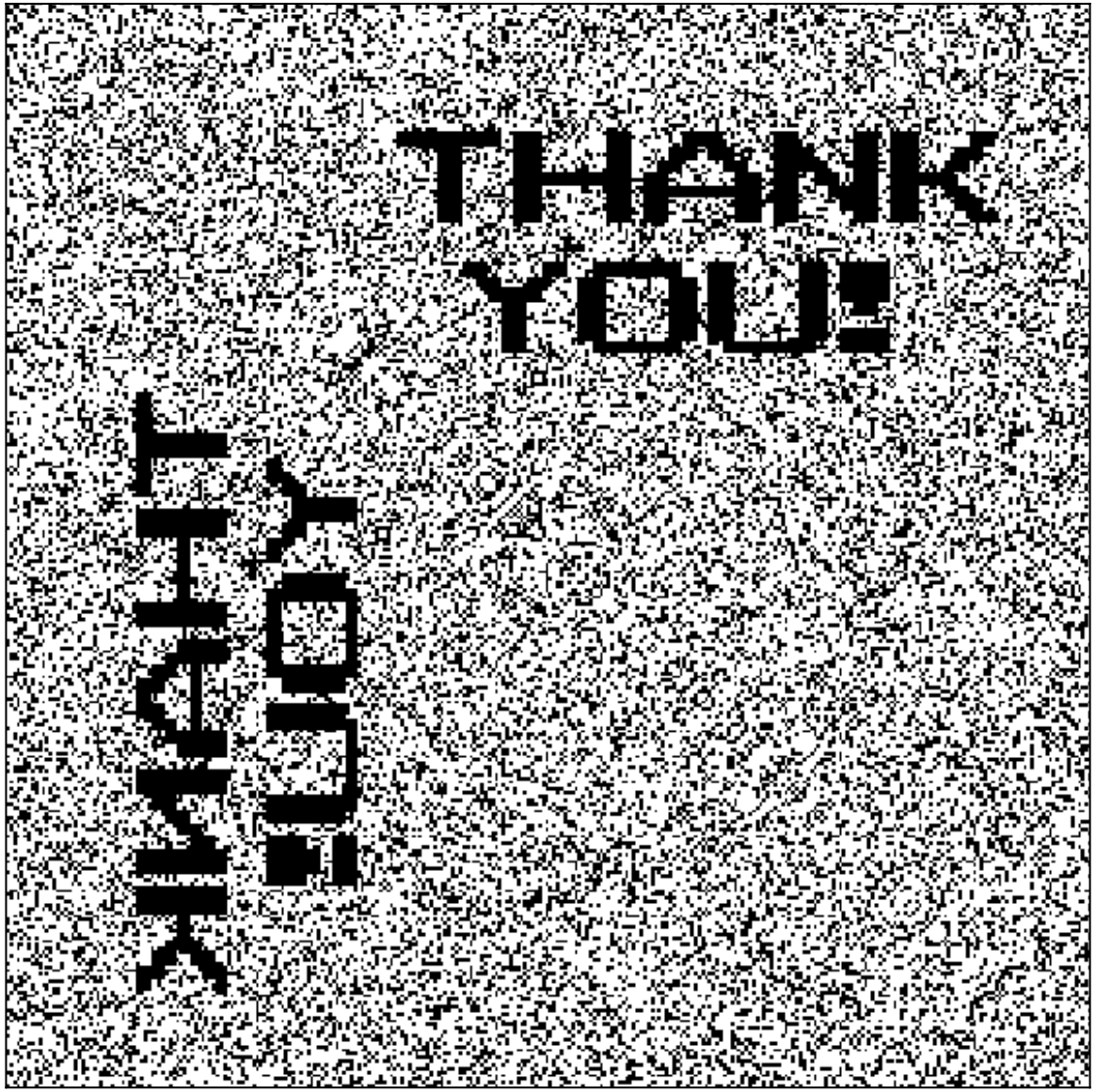




THANK YOU!


THANK YOU!
ありがとう!


THANK YOU!
ありがとう!


THANK YOU!
ありがとう!