

# Lecture 2: Network nonparametrics

Bryan S. Graham, UC - Berkeley & NBER

September 26, 2016

In Lecture 1 we discussed the structure of, and how to describe and summarize, network data. In this lecture we will review nonparametric summaries of network data. This will involve thinking about so called *graph limits*. Let  $G(\mathcal{V}, \mathcal{E})$  be a finite undirected random graph with agents/vertices  $\mathcal{V} = \{1, \dots, N\}$ , links/edges  $\mathcal{E} = \{\{i, j\}, \{k, l\}, \dots\}$ , and adjacency matrix  $\mathbf{D} = [D_{ij}]$  with

$$D_{ij} = \begin{cases} 1 & \text{if } \{i, j\} \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$

As the number of agents in  $G$  grows large does it approach some sort of limiting object? It turns out that in the case of *dense graphs*, where the probability of a link between any two agents is bounded away from zero, that such a limit exists. We will characterize this limit and show how it can be estimated using a finite network. Much of the work in this area is in the applied probability, machine learning and statistics literatures (e.g., Bickel and Chen, 2009; Olhede and Wolfe, 2014; Orbanz and Roy, 2015; Lovász, 2012). In addition to being technical, this work is generally motivated by questions quite far from those that arise in economics. Nevertheless I believe these methods have something to offer to economists and can be useful tools for empirical analysis and model building.

Let  $\pi$  be a permutation of the index set  $\{1, \dots, N\}$ . In many situations it is natural to assume that

$$D_{ij} \stackrel{d}{=} D_{\pi(i)\pi(j)} \tag{1}$$

for every permutation  $\pi$ . The notation  $\stackrel{d}{=}$  indicates that the random variables on each side of the equality have the same distribution.

Condition (1) implies that our beliefs about the probability of a link between two agents does not depend on their labels. Networks with this property are *exchangeable*. If agents' indices denote their order of entry into the network, or their position on a line, then condition (1) is unlikely to hold. However, even with vertex information available to the econometrician,

some components of the graph will likely still be exchangeable. For example if agents  $i$  and  $j$  have identical “locations” and likewise for  $k$  and  $l$ , then it is natural to assume that  $D_{ik} \stackrel{d}{=} D_{il} \stackrel{d}{=} D_{jk} \stackrel{d}{=} D_{jl}$ . In this lecture we will assume condition (1); this is most natural when agent attributes are unavailable. Nevertheless the ideas developed below remain useful, with modification, under weaker notions of exchangeability.

Bickel and Chen (2009) observed that an extension of Bruno de Finetti’s representation theorem for exchangeable sequences to jointly exchangeable arrays, due to Davis Aldous and Douglas Hoover, implies that any random adjacency matrix satisfying condition (1) has an “as if” representation of

$$D_{ij} = g(\alpha, U_i, U_j, \varepsilon_{ij}) \quad (2)$$

where (i)  $\alpha$ ,  $\{U_i\}_{i=1}^N$  and  $\{\varepsilon_{ij}\}_{i=1, i>j}^N$  are i.i.d.  $\mathcal{U}[0, 1]$  random variables and (ii) the function  $g(\alpha, U_i, U_j, \varepsilon_{ij})$  is symmetric in its second and third arguments. Here  $\alpha$  is mixing variable.<sup>1</sup> A slightly more useful, but equivalent, formulation is

$$D_{ij} = \mathbf{1}(\varepsilon_{ij} \leq h(\alpha, U_i, U_j)). \quad (3)$$

We will call the function  $h(\alpha, U_i, U_j)$  a *graphon*, which is short for **graph** function.

Representation (3) suggests the following network simulation algorithm (e.g., Orbanz and Roy, 2015):

1. “Draw”  $\alpha$  or choose a graphon;
2. Draw  $U_i \sim \mathcal{U}[0, 1]$  for agents  $i = 1, \dots, N$ ;
3. Construct  $\mathbf{D}$ , by sampling  $D_{ij} | h(\alpha, \bullet, \bullet), U_i, U_j \sim \text{Bernoulli}(h(\alpha, U_i, U_j))$  for every dyad  $\{i, j\}$  with  $i > j$ .

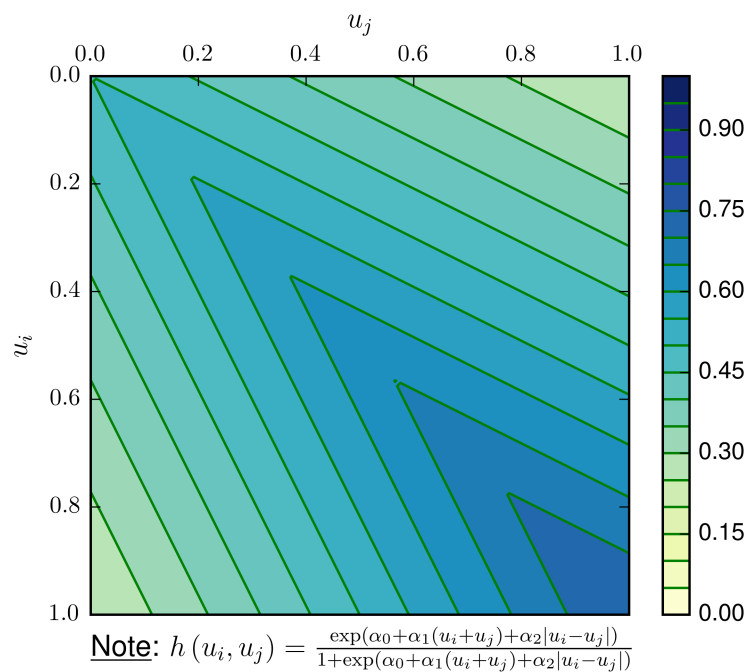
We shall see that this DGP is rather flexible (the graphon may be very complicated). A simple example of a graphon, which will serve as a test case below, is

$$h(\alpha, U_i, U_j) = \frac{\exp(\alpha_0 + \alpha_1(U_i + U_j) + \alpha_2|U_i - U_j|)}{1 + \exp(\alpha_0 + \alpha_1(U_i + U_j) + \alpha_2|U_i - U_j|)}. \quad (4)$$

This graphon is depicted in Figure 1. We place the origin of the figure in its upper-left-hand corner so that the ordering of  $u_i$  and  $u_j$  coincide with that of the row and column indices,  $i$  and  $j$ , in the adjacency matrix  $\mathbf{D}$ . The form of  $h(\alpha, U_i, U_j)$  induces positive *assortative matching* on the latent agent type  $U_i$ . It is also the case that links are more likely among high type agents. This induces *degree heterogeneity*.

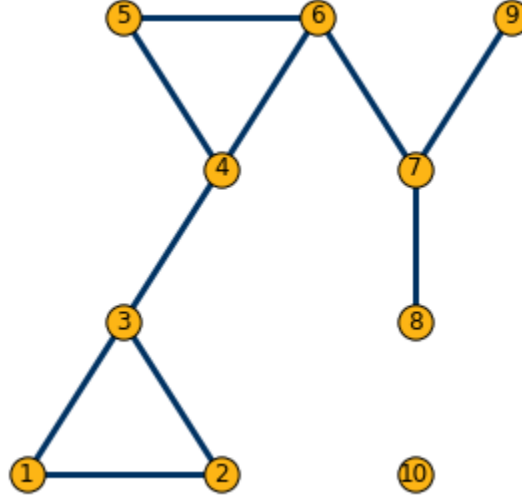
<sup>1</sup>Recall that de Finetti’s original result represents exchangeable sequences as mixtures of i.i.d. sequences.

Figure 1: Illustration of a homophily-inducing graphon  
Graphon contour plot



Notes: Contour plot of equation (4) in the main text. Axes begin in the upper-left-hand corner to coincide with the ordering of matrix rows and columns. The elements of  $\alpha = (\alpha_0, \alpha_1, \alpha_2)'$  are chosen such that  $h(\alpha, 0, 0) = 0.25$ ,  $h(\alpha, 0, 1) = 0.5$  and  $h(\alpha, 1, 1) = 0.75$ .

Figure 2: Understanding homomorphism densities



Notes: Let  $S = \triangle$  and  $G$  be a graph of interest to the econometrician, and as depicted in the figure. Since the subgraph  $\triangle$  appears twice in  $G$  we have  $\text{hom}(\triangle, G) = 2 \times 6 = 12$  (recall that a triad can be ordered in  $3!$  different ways). The density of  $S = \triangle$  in  $G$  is therefore  $t(S, G) = \frac{\text{hom}(S, G)}{N^K} = \frac{12}{10^3} = 0.012$  (where  $N = 10$  and  $K = 3$ ).



Representation (3) raises the question of whether  $h(\alpha, u, v)$  is nonparametrically estimable from the realized adjacency matrix,  $\mathbf{D}$ . A little introspection reveals that there is not a one-to-one mapping between the probability distribution for  $\mathbf{D}$  and the graphon  $h(\alpha, u, v)$ . To see why consider the transformation  $\varphi(u) = 1 - u$ , which maps high values of  $U_i$  to low values and vice-versa. This transformation is measure preserving since  $\varphi(U_i) \sim \mathcal{U}[0, 1]$  and it is clear that  $h(\alpha, \varphi(u), \varphi(v))$  implies the same probability distribution for  $\mathbf{D}$  as  $h(\alpha, u, v)$ . The best we can hope for is identification of the graphon up to a measuring preserving transformation. Latent agent types,  $U_i$ , are not identifiable without imposing additional restrictions.

More optimistically, we will show that although  $h(\alpha, u, v)$  is non-identifiable, its values at all pairs  $\{U_i, U_j\}_{i=1, i > j}^N$ , corresponding to the *edge probabilities*,  $\{P_{ij}\}_{i=1, i > j}^N$ , or entries of the *expected adjacency matrix*

$$\mathbf{P} = \mathbb{E}[\mathbf{D} | \alpha, \mathbf{U}] \quad (5)$$

are non-parametrically identified under weak conditions (here  $\mathbf{U} = (U_1, \dots, U_N)'$ ).

## Graphons as graph limits

Graphons can also be motivated in terms of *graph limits*. To develop this idea we require some additional notation. Let  $S$  and  $G$  be two graphs. It is helpful to think of  $S$  as a small subgraph or graph “motif” such as a triangle  or two star . We can think of  $G$  as a larger graph of interest; for concreteness let  $G$  be the 10 agent network depicted in Figure 2. Imagine inserting  $S = \text{triangle}$  into  $G$ , as depicted in Figure 2. There are two triangles in  $G$ , but since we can order the three vertices in a triangle in  $3! = 6$  different ways there are a total of  $2 \times 6 = 12$  different ways we can insert  $S = \text{triangle}$  into  $G$  while preserving edge adjacency (i.e, the “structure” of  $G$ ).

More formally a *graph homomorphism* is a map from  $\mathcal{V}(S)$ , the nodes of  $S$ , to  $\mathcal{V}(G)$ , the nodes of  $G$ , that preserves edge adjacency. For example if we label the vertices in  $S = \text{triangle}$   $a$ ,  $b$  and  $c$  going clockwise from the top, one *graph homomorphism* is the map  $a \rightarrow 1$ ,  $b \rightarrow 2$ , and  $c \rightarrow 3$ . The map  $a \rightarrow 2$ ,  $b \rightarrow 10$ , and  $c \rightarrow 8$ , is not a graph homomorphism. Such an insertion would require a re-wiring of  $G$  (specifically the addition of the edges  $\{2, 8\}$ ,  $\{2, 10\}$  and  $\{8, 10\}$ ).

Let  $\text{hom}(S, G)$  denote the number of homomorphisms from  $S$  to  $G$ . For  $G$  as depicted in Figure 2, we have  $\text{hom}(\text{triangle}, G) = 12$ .

Let  $|\mathcal{A}|$  denote the number of elements in the set  $\mathcal{A}$ , so that  $|\mathcal{V}(G)| = N$  equals the number of agents in  $G$ , and  $|\mathcal{V}(S)| = K$  the number of nodes in the subgraph  $S$ . Observe that there are a total of  $N^K$  maps from  $S$  to  $G$  (if we allow for maps like  $a \rightarrow 1, b \rightarrow 1, c \rightarrow 1$ ).<sup>2</sup>

With these preliminaries in hand we can define the *homomorphism density* of  $S$  in  $G$  as

$$t(S, G) = \frac{\text{hom}(S, G)}{N^K} \quad (6)$$

Returning to  $G$ , as depicted in Figure 2, we have  $t(\text{triangle}, G) = \frac{12}{10^3} = 0.012$ . This is very close to the fraction of all triads in  $G$  which are closed triangles, which equals  $2\binom{N}{3}^{-1} = \frac{2 \times 6}{10 \times 9 \times 8} \approx 0.017$ . Hence as  $N = |\mathcal{V}(G)|$  grows large,  $t(S, G)$  equals the density of the finite subgraph  $S$  in  $G$ .

Homomorphism densities are useful because they allow us to consider graph limits. Let  $\{G_N\}_{N=1}^{\infty}$  be a sequence of networks. If

$$\lim_{N \rightarrow \infty} t(S, G_N) = t(S, h)$$

<sup>2</sup>If we wish to rule out maps which assign multiple nodes in  $S$  to the same node in  $G$ , then the number of mappings is  $N \times (N-1) \times \dots \times (N-K+1)$ . This distinction will not be important when  $N$  is large relative to  $K$ .

for some graphon  $h(\alpha, \bullet, \bullet)$  and all fixed subgraphs  $S$  we say that  $G_N$  converges to  $h(\alpha, \bullet, \bullet)$ . Here we define, for example,

$$t(\triangle, h) = \int_0^1 \int_0^1 \int_0^1 h(\alpha, u, v) h(\alpha, v, x) h(\alpha, x, u) du dv dx$$

with  $t(S, h)$  for an arbitrary subgraph  $S$  defined analogously. Imagine we have a large graph and we execute a triangle census, computing  $t(\triangle, G)$ ,  $t(\triangle, G)$ ,  $t(\triangle, G)$  and  $t(\triangle, G)$ . Say we have a specific parametric graphon of interest. Using this graphon we compute, perhaps numerically,  $t(\triangle, h)$ ,  $t(\triangle, h)$ ,  $t(\triangle, h)$  and  $t(\triangle, h)$ . If our graphon model is among the equivalence class of models which “generated” the network, then the two sets of triad densities should be very close. If our sample densities are far from those computed using the null model, we can take that as evidence against the hypothesized model. For some results which could be used to make claims of this type more precise see the monograph by Lovász (2012). Glasscock (2015) provides a very short introduction to graphons that I found useful for getting started.

To summarize: the data generating process for an exchangeable network can be represented in terms of a graphon. Graphons also have an interpretation as the limit of a sequence of graphs. Of particular interest is the fact that we can calculate the limiting frequency of different types of subgraphs  $S$  in  $G$  using the graphon. This can simplify probabilistic calculations for finite graphs (by using the asymptotic graphon approximation). However it also suggest as approach to learning about the graphon from the frequencies of different types of subgraphs observed in the network in hand (cf., Bickel et al., 2011). See also the counting and inverse counting results in Chapter 10 of Lovász (2012).

For certain types of statistical analyses it can be convenient to formulate the graphon somewhat differently (e.g., Bickel and Chen, 2009). Consider the network DGP

$$\Pr(D_{ij} = 1 | \alpha, U_i, U_j) = h(\alpha, U_i, U_j)$$

and define

$$\rho(\alpha) = \int_0^1 \int_0^1 h(\alpha, u, v) du dv, \quad w(\alpha, u, v) = \Pr(U_i = u, U_j = v | D_{ij} = 1, \alpha).$$

Using this notation we have, using the fact that  $f_{U_i, U_j | \alpha}(u, v | \alpha) = 1$  on  $[0, 1]^2$ , the formula-

$$h(\alpha, u, v) = \rho(\alpha) w(\alpha, u, v).$$

This can be useful for considering sequences of network DGPs where  $\rho(\alpha)$ , the network density, is indexed by  $N$ . For example we might have  $\rho_N(\alpha) \rightarrow 0$  and  $N \rightarrow \infty$  at some rate (we assume  $\rho_N(\alpha) w(\alpha, u, v) \leq 1$  for all  $N$ ). Observe that the expected degree of the randomly sampled agent  $i$  is  $\mathbb{E} \left[ \sum_{j \neq i} \mathbb{E} [D_{ij}] \right] = (N-1) \rho_N$ . By setting  $\rho_N(\alpha) = \frac{c\alpha}{N}$ , for example, we can construct a sequence of graphs that are sparse in the limit (i.e., where each agent has only a finite number of links). Setting  $\rho_N(\alpha) = c\alpha \ln N/N$  yields an intermediate case, where average degree increases with  $N$ , but nevertheless becomes small relative to the overall size of the network.

Most of the results discussed in this note apply only to the dense case where  $\rho_N(\alpha)$  is a positive constant, or approaches a positive constant in the limit. There are more theoretical results available for dense graphs (in both the probability and statistics literature), although the sparse case is more empirically relevant for most applications in economics.

In what follows I will generally suppress the dependence of  $h(\alpha, u, v)$ ,  $\rho_N(\alpha)$  and  $w(\alpha, u, v)$  on  $\alpha$  and/or  $N$ .

## Consistent estimation of edge probabilities

Let  $U_i = u$ , the realized value of  $U_i$  for agent  $i$ , denote an agent's *type*. If agents  $i$  and  $j$  are of the same type, then they will have identical linking behavior in expectation; any differences in the configuration of their links will be purely random. However it is also possible that two agents of different types may nevertheless link in identical ways (in expectation).

To be more precise about this observation, as well as its implications, define the inner product

$$\langle f, g \rangle = \int f(u) g(u) du$$

with the associated norm

$$\|f\| = \langle f, f \rangle^{1/2} = \left[ \int f(u)^2 du \right]^{1/2}.$$

The linking behavior of an agent of type  $u$  is summarized by the *graphon slice*  $\rho w(u, \bullet)$ . For example the probability a type  $u$  agent links with a type  $t$  agent is given by  $\rho w(u, t)$ ; by varying  $t$  across the unit interval we map out the linking behavior of type  $u$  agents with all other possible types in the network. This observation suggests measuring the “distance” of agent  $i$ , with  $U_i = u$ , from agent  $j$ , with  $U_j = v$ , by the  $l_2$  distance between their respective

graphon slices (cf., Zhang et al., 2015):

$$d(u, v) = \|\rho w(u, \cdot) - \rho w(v, \cdot)\|_2 = \rho \left[ \int [w(u, t) - w(v, t)]^2 dt \right]^{1/2} \quad (7)$$

It is also helpful to consider a finite analog of  $d(u, v)$ . Recall that  $\mathbb{E}[\mathbf{D}|\mathbf{U}] = \mathbf{P}$  denotes the expected adjacency matrix. Let  $\mathbf{P}_{i\bullet}$  denote the  $i^{\text{th}}$  row of this matrix. Call this object an *expected adjacency matrix slice* for agent  $i$ . If, *after removing the  $i^{\text{th}}$  and  $j^{\text{th}}$  elements*,  $\mathbf{P}_{i\bullet} = \mathbf{P}_{j\bullet}$ , then agents  $i$  and  $j$  are “behaviorally identical” (even if  $U_i$  differs from  $U_j$ ). This gives the finite network analog of (7):

$$d_N(i, j) = \|\mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}\|_2 = \left[ \frac{1}{N-2} \sum_{k \neq i, j} (P_{ik} - P_{jk})^2 \right]^{1/2}. \quad (8)$$

Here, and in what follows, it is implied that the  $i^{\text{th}}$  and  $j^{\text{th}}$  elements have been removed from both  $\mathbf{P}_{i\bullet}$  and  $\mathbf{P}_{j\bullet}$  when calculating the  $l_2$  distance between them.

Even though it is circular, it is instructive to consider how one might estimate  $\mathbf{P}$  if it was, in fact, already known! We say that  $j$  is an *exact neighbor* of  $i$  if  $d_N(i, j) = 0$  or they have identical (expected) adjacency (matrix) slices. Concretely, if  $d_N(i, j) = 0$ , then the  $i^{\text{th}}$  and  $j^{\text{th}}$  rows of the expected adjacency matrix are identical (the  $i^{\text{th}}$  and  $j^{\text{th}}$  elements of these two rows excepted).

In a finite network it may be that agent  $i$  has no exact neighbors, but we can still find a set of *nearest neighbors*:

$$\mathcal{N}_i = \{j : \|\mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}\|_2 \leq q_i(h_N)\} \quad (9)$$

where  $q_i(h_N)$  is the  $h_N^{\text{th}}$  sample quantile of  $\{\|\mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}\|_2\}_{j=1, j \neq i}^N$ . If  $N = 1,000$  and  $h_N = 0.05$ , then we would take the 50 nearest neighbors of  $i$  (defined in terms of the  $l_2$  distance between  $\mathbf{P}_{i\bullet}$  and other rows of  $\mathbf{P}$ ).

With a set of nearest neighbors in hand for all  $\binom{N}{2}$  agents in the network we could then estimate  $P_{ij}$  by the following local average of the adjacency matrix entries

$$\hat{P}_{ij}^{\text{oracle}} = \frac{1}{2} \left( \frac{\sum_{k \in \mathcal{N}_i} D_{kj}}{|\mathcal{N}_i|} + \frac{\sum_{l \in \mathcal{N}_j} D_{il}}{|\mathcal{N}_j|} \right). \quad (10)$$

The first term in (10) equals the frequency with which  $i$ 's nearest neighbors link with agent  $j$ . Since  $\mathbf{P}_{i\bullet} \approx \mathbf{P}_{k\bullet}$  for  $k \in \mathcal{N}_i$  this average should be close to  $P_{ij}$  for  $|\mathcal{N}_i|$  large enough and  $h_N$  small enough. The second term in (10) equals the frequency with which  $j$ 's nearest neighbors



link with agent  $i$ , which also estimates  $P_{ij}$ .<sup>3</sup> For (10) to provide a consistent estimate of  $P_{ij}$  we require that  $h_N \rightarrow 0$  (to eliminate bias), and  $|\mathcal{N}_i| \rightarrow \infty$  (to reduce variance) as the number of agents in the network grow large. These conditions effectively require the network to be dense in the limit. Standard bias-variance trade-offs are involved in the choice of  $h_N$ .

Unfortunately  $\mathbf{P}$  is unobserved by the econometrician, and  $\mathbf{D}$  is not a good estimate of it (it is unbiased but not consistent). A feasible direct analog of (10) is not available.

Instead of  $\mathbf{P}$  consider its matrix square:

$$\mathbf{P}^2 = \begin{pmatrix} \sum_{i \neq 1} P_{1i}P_{i1} & \sum_{i \neq 1,2} P_{1i}P_{i2} & \cdots & \sum_{i \neq 1,N} P_{1i}P_{iN} \\ \sum_{i \neq 2,1} P_{2i}P_{i1} & \sum_{i \neq 2} P_{2i}P_{i2} & \cdots & \sum_{i \neq 2,N} P_{2i}P_{iN} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i \neq N,1} P_{Ni}P_{i1} & \sum_{i \neq N,2} P_{Ni}P_{i2} & \cdots & \sum_{i \neq N} P_{Ni}P_{iN} \end{pmatrix}.$$

If agents  $i$  and  $j$  are such that  $P_{ik} = P_{jk}$  for  $k \neq i, j$  (i.e., they have identical adjacency slices), then  $\sum_{k \neq i,l} P_{ik}P_{kl} = \sum_{k \neq j,l} P_{jk}P_{kl}$  for all  $l \neq i, j$ .<sup>4</sup> Therefore, if  $\mathbf{P}_{i\bullet} = \mathbf{P}_{j\bullet}$ , we have that  $[\mathbf{P}^2]_{i\bullet} = [\mathbf{P}^2]_{j\bullet}$  (again we assume that the  $i^{\text{th}}$  and  $j^{\text{th}}$  elements of these two rows are removed prior to checking for equality). A second key observation is that the  $(i, j)^{\text{th}}$  off-diagonal entries of  $\mathbf{P}^2$  can be estimated by  $\sum_{k \neq i,j} D_{ik}D_{kj}$  (estimation of the diagonals is not as easy). This suggests defining nearest neighbors in terms of

$$\hat{d}_N^*(i, j) = \left[ \frac{1}{N-2} \sum_{l \neq i,j} \left( \sum_{k \neq i,l} D_{ik}D_{kl} - \sum_{k \neq j,l} D_{jk}D_{kl} \right)^2 \right]^{1/2} \quad (11)$$

and estimating link probabilities by

$$\hat{P}_{ij} = \frac{1}{2} \left( \frac{\sum_{k \in \mathcal{N}_i} D_{kj}}{|\hat{\mathcal{N}}_i|} + \frac{\sum_{l \in \mathcal{N}_j} D_{il}}{|\hat{\mathcal{N}}_j|} \right). \quad (12)$$

Here  $\hat{\mathcal{N}}_i$  is constructed using  $\hat{d}_N^*(i, j)$ .

## Finding agent neighborhoods in practice

To develop our argument rigorously we need to consider the structure of the  $l_2$  distance between adjacency slices, introduced in equation (8) above, more carefully. Taking the

<sup>3</sup>Careful inspection of (10) indicates that it is defined correctly for a directed network as well, but in the text I only consider the undirected case.

<sup>4</sup>If  $\mathbf{P}$  is symmetric, as occurs when the graph is undirected, then these equalities hold for all  $l = 1, \dots, N$ .

square of  $d_N(i, j)$  yields

$$\begin{aligned} d_N^2(i, j) &= \langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet} \rangle \\ &= (\langle \mathbf{P}_{i\bullet}, \mathbf{P}_{i\bullet} \rangle + \langle \mathbf{P}_{j\bullet}, \mathbf{P}_{j\bullet} \rangle \\ &\quad - 2 \langle \mathbf{P}_{i\bullet}, \mathbf{P}_{j\bullet} \rangle). \end{aligned} \tag{13}$$

As suggested above, direct analog estimator is available for the last term in (13), namely,

$$\frac{2}{N-2} \sum_{k \neq i, j} D_{ik} D_{jk}.$$

To understand this estimator observe that  $\mathbb{E}[D_{ik} D_{jk} | U_i, U_j] = \rho^2 \int w(u_i, t) w(u_j, t) dt$ . Furthermore, since we are able to sum over  $N-2$  agents, there is some hope that this estimate is a good one (i.e., that  $(N-2)^{-1} \sum_k D_{ik} D_{jk} \xrightarrow{p} \rho^2 \int w(u_i, t) w(u_j, t) dt$ ).

Unfortunately, and also as noted above, the obvious analog estimate of  $\langle \mathbf{P}_{i\bullet}, \mathbf{P}_{i\bullet} \rangle = \frac{1}{N-2} \sum_{k \neq i, j} P_{ik}^2$ , namely  $\sum_k D_{ik} D_{ik} = \sum_k D_{ik}$ , is not a good one since  $\mathbb{E}[D_{ik} | U_i] = \rho \int w(u_i, t) dt$  which is not the target of interest.

It turns out, however, that we can derive an upper bound for  $d_N^2(i, j)$  that *is estimable* when  $N$  is large. If we can find agents that are close in terms of this upper bound, we can conclude that they are close in terms of (8), and construct neighborhoods accordingly. This basic idea is due to Zhang et al. (2015) and Auerbach (2016), whose arguments we will follow.

Rewriting (13) yields

$$\begin{aligned} d_N^2(i, j) &= \langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} \rangle - \langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{j\bullet} \rangle \\ &\leq |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} \rangle| + |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{j\bullet} \rangle| \end{aligned}$$

with the second line following from the triangle inequality (TI).

To make further progress we need to assume that  $w(u, v)$  is Lipschitz continuous<sup>5</sup> with a Lipschitz constant of  $C$  :

$$\rho \|w(u, \cdot) - w(v, \cdot)\|_2 \leq C \|u - v\|_2.$$

If the size of the network in hand is large enough then, with very high probability, we should be able to find a third agent  $k \neq i, j$  such that

$$\|U_i - U_k\|_2 = |U_i - U_k| \leq \epsilon_N$$

---

<sup>5</sup>In fact piecewise Lipschitz continuity is sufficient.

for some  $\epsilon_N = o_N(1)$ . This gives the series of inequalities

$$\begin{aligned}
|\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} \rangle| &= |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{k\bullet} \rangle| \\
&\quad + |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} - \mathbf{P}_{k\bullet} \rangle| \\
&\leq |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{k\bullet} \rangle| \\
&\quad + |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{i\bullet} - \mathbf{P}_{k\bullet} \rangle| \\
&\leq |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{k\bullet} \rangle| \\
&\quad + \|\mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}\|_2 \|\mathbf{P}_{i\bullet} - \mathbf{P}_{k\bullet}\|_2 \\
&\leq |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{k\bullet} \rangle| + C_{i,j} \epsilon_N
\end{aligned}$$

where the first inequality follows from the TI, the second from the Cauchy-Schwarz Inequality (CSI) (i.e.,  $|\langle f, g \rangle|^2 \leq \langle f, f \rangle \cdot \langle g, g \rangle$ ), and  $C_{i,j} = C \|\mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}\|_2$  is proportional to the actual  $l_2$  distance between the expected adjacency matrix slices associated with agents  $i$  and  $j$ .

Using the same argument we can find, again with high probability, an agent  $l \neq i, j$  in the network such that  $|U_j - U_l| \leq \epsilon_N$ . Putting these two results together then yields

$$\begin{aligned}
d^2(i, j) &\leq |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{k\bullet} \rangle| \\
&\quad + |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle| + 2C_{i,j} \epsilon_N \\
&\leq 2 \max_{l \neq i, j} |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle| + 2C_{i,j} \epsilon_N.
\end{aligned}$$

Define  $\bar{d}^2(i, j) = 2 \max_{l \neq i, j} |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle|$ . Now observe that if  $\bar{d}^2(i, j) = 0$ , then if the network is large enough, it will be the case that  $d^2(i, j) = 0$  so that agents  $i$  and  $j$  have similar expected adjacency matrix slices. Observe that the maximization in the last line of the above display absolves us of the need to find actual agents  $k$  and  $l$  with  $|U_i - U_k| \leq \epsilon_N$  and  $|U_j - U_l| \leq \epsilon_N$ .

Furthermore  $\bar{d}^2(i, j)$  is estimable by

$$\hat{\bar{d}}^2(i, j) = 2 \max_{l \neq i, j} \left| \frac{1}{N-2} \sum_{k \neq i, j} D_{ik} D_{lk} - \sum_{k \neq i, j} D_{jk} D_{lk} \right|. \quad (14)$$

In words: find an agent  $l$  such that the dyads  $(i, l)$  and  $(j, l)$  have the largest difference in the number of links in common as is possible. If this difference is zero, then, with high probability,  $i$  and  $j$  have expected adjacency matrix slices close to one another.

Using (14) we can define the *estimated neighborhood* of agent  $i$  as the index set

$$\hat{\mathcal{N}}_i = \left\{ j : \hat{d}^2(i, j) \leq q_i(h) \right\}$$

where  $q_i(h)$  is the  $h^{\text{th}}$  sample quantile of  $\left\{ \hat{d}^2(i, j) \right\}_{j=1, j \neq i}^N$ . This is the estimator introduced and studied by Zhang et al. (2015). Let  $K_N = \lfloor Nh_N \rfloor$ , then we take the  $K_N$  nearest neighbors of  $i$  as our estimate  $\hat{\mathcal{N}}_i$ . The consistency result given by Zhang et al. (2015) requires that  $h_N = C\sqrt{\frac{\ln N}{N}}$  for some  $C$ . Their Monte Carlo experiments suggest that  $C = 0.1$  works well in practice. This gives  $K_N = \lfloor 0.1(N \ln N)^{1/2} \rfloor$  or  $K_{1000} \approx 8$  and  $K_{2000} \approx 12$ . In my own (very limited) experience I have found that their method of finding nearest neighbors difficult to tune on test examples. An alternative approach, closer to that used by Auerbach (2016), is based on  $\hat{d}_N^*(i, j)$  as defined in (11) above. To justify such an approach observe that

$$\begin{aligned} \max_{l \neq i, j} |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle| &\leq \frac{1}{N-2} \sum_{l \neq i, j} |\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle| \cdot 1 \\ &\leq \frac{1}{N-2} \left[ \sum_{l \neq i, j} (\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle)^2 \right]^{1/2} \cdot \left[ \sum_{k \neq i, j} 1^2 \right]^{1/2} \\ &= \frac{1}{\sqrt{N-2}} \left[ \sum_{k \neq i, j} (\langle \mathbf{P}_{i\bullet} - \mathbf{P}_{j\bullet}, \mathbf{P}_{l\bullet} \rangle)^2 \right]^{1/2} \\ &= \left[ \frac{1}{N-2} \sum_{l \neq i, j} \left( \sum_{k \neq i, l} P_{ik} P_{kl} - \sum_{k \neq j, l} P_{jk} P_{kl} \right)^2 \right]^{1/2} \\ &= d_N^*(i, j) \end{aligned}$$

where the second inequality follows from Hölder's Inequality (i.e.,  $\|fg\|_1 \leq \|f\|_p \|g\|_q$  for  $p^{-1} + q^{-1} = 1$ ). This gives

$$d_N^2(i, j) \leq d_N^*(i, j) + 2C_{i,j} \epsilon_N.$$

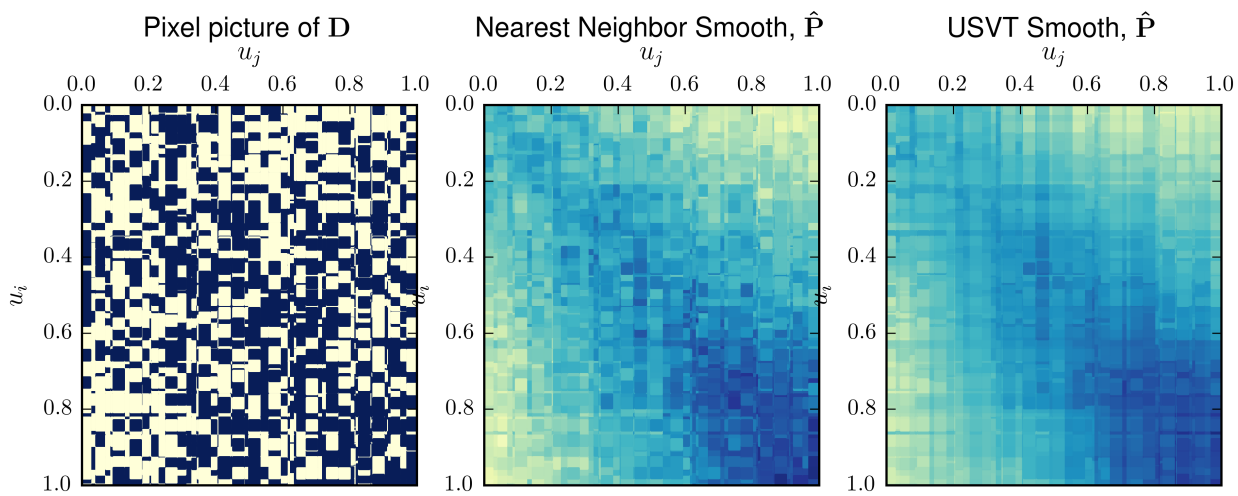
The second term to the right of the inequality will be small in large networks with high probability and the first term can be estimated using (11) above. I have found that selecting nearest neighbors based on (11) easier to tune.

Consider a network generated using the graphon depicted in Figure 1 above. The left-hand panel of Figure 3 plots what is called a *pixel picture* of a random adjacency matrix draw from graphon (4) with  $N = 2,000$  agents. The picture suggests greater density along the main diagonal, but its otherwise difficult to interpret. The middle-hand panel plots an

estimate of  $\hat{\mathbf{P}}$  based on 100 nearest neighbors defined using  $\hat{d}_N^*(i, j)$ . The estimate is a rather remarkable approximation of the true underlying graphon. The right-hand panel plots the USVT estimate of  $\hat{\mathbf{P}}$  due to Chatterjee (2015). This procedure is not describe here.

It is important to emphasize that in real empirical settings the researcher cannot construct a picture like Figure 1. This is because the latent agent types  $\{U_i\}_{i=1}^N$  are unobserved. Nevertheless the edge probabilities can be a useful input into further analysis (as is suggested by the work of Auerbach (2016)) and/or can be used – with some insight – to infer different types of underlying structure. In practice researchers might, for example, sort agents by their estimated degrees,  $(N-1) \sum_{j \neq i}^{-1} \hat{P}_{ij}$  for plotting purposes. If the graphon is monotone in agent type, this approach will “work” (Yang et al., 2014). Even in the absence of monotonicity it can be a reasonable default sorting of the data for the purposes of visualization and the discovery of community structure.

Figure 3: Nonparametric estimate of edge probabilities



**Notes:** Left panel: pixel picture of a random draw from (4) with  $N = 2,000$ . Middle panel: nearest neighbor smoothed graphon estimate. Right Panel: Universal Singular Value Thresholding graphon estimate (cf., Chatterjee, 2015).

## Exercises

1. Argue that the map  $\varphi(U) = 2U \bmod 1$  is measure preserving for  $U \sim \mathcal{U}[0, 1]$ .
2. Propose a restriction of  $h(u, v)$  such that agent types are identifiable.
3. Compute  $\text{hom}(\text{graphon}, G)$  for  $G$  as depicted in Figure 2.
4. Compute  $t(\text{graphon}, h)$  and  $t(\text{graphon}, h)$  for  $h(u, v) = \frac{1}{2}$ ,  $h(u, v) = \frac{1}{2}(u + v)$ ,  $h(u, v) = \frac{uv}{2}$ .  
Use your results to calculate (large) network transitivity for each graphon.

## References

- Auerbach, E. (2016). A matching estimator for models with endogenous network formation. Technical report, University of California - Berkeley.
- Bickel, P. J. and Chen, A. (2009). A nonparametric view of network models and newman-girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068 – 21073.
- Bickel, P. J., Chen, A., and Levina, E. (2011). The method of moments and degree distributions for network models. *Annals of Statistics*, 39(5):2280 – 2301.
- Chatterjee, S. (2015). Matrix estimation by universal singular value thresholding. *Annals of Statistics*, 43(1):177 – 214.
- Glasscock, D. (2015). What is a graphon? *Notices of the American Mathematical Society*, 62(1):46 – 48.
- Lovász, L. (2012). *Large Networks and Graph Limits*, volume 60 of *American Mathematical Society Colloquium Publications*. American Mathematical Society.
- Olhede, S. C. and Wolfe, P. J. (2014). Network histograms and universality of blockmodel approximation. *Proceedings of the National Academy of Sciences*, 11(41):14722 – 14727.
- Orbanz, P. and Roy, D. M. (2015). Bayesian models of graphs, arrays and other exchangeable random structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):437 – 461.
- Yang, J. J., Han, Q., and Airolidi, E. M. (2014). Nonparametric estimation and testing of exchangeable graph models. *JMLR Workshop and Conference Proceedings*, 33:1060 – 1067.

Zhang, Y., Levina, E., and Zhu, J. (2015). Estimating network edge probabilities by neighborhood smoothing. *arXiv:1509.08588v2 [stat.ML]*.