Building a network with assortative mixing starting from preference functions, with application to the spread of epidemics

Appendix

Razvan G. Romanescu

July 23, 2024

1 Derivation of a closed form solution to the degree preference $f_{i,j}$ for the income network

Start by writing variables Y and D in terms of correlated standard normal variates. Namely, $Y = F_Y^{-1}(\Phi(z_Y))$, and $D = F_D^{-1}(\Phi(z_D))$, with z_Y and z_D connected via $z_Y = \rho z_D + \sqrt{1-\rho^2}z_\epsilon$, where z_ϵ is another standard variate independent of z_D. Here F_Y and F_D are cdf's of variables Y, D and Φ is the standard normal cdf. For tractability, we treat the degree as continuous and approximate the final answer $f_{i,j}$ for discrete u and v with its continuous counterpart. Now we can write the expectation in (6) as

$$f_{i,j} = E[f(D_k, Y_k; D_l, Y_l)|D_k = i, D_l = j] \propto E\left[Y_k^{\delta_k} Y_l^{\delta_l} \middle| D_k = i, D_l = j\right] \quad (A1)$$

$$= E\left[\left\{F_Y^{-1}\left(\Phi\left(\rho z_{D_k} + \sqrt{1-\rho^2}z_{\epsilon_k}\right)\right)\right\}^{\delta_k}\left\{F_Y^{-1}\left(\Phi\left(\rho z_{D_l} + \sqrt{1-\rho^2}z_{\epsilon_l}\right)\right)\right\}^{\delta_l} \middle| \begin{matrix} z_{D_k} = \Phi^{-1}(F_D(i)), \\ z_{D_l} = \Phi^{-1}(F_D(j)) \end{matrix}\right]$$

The expectation is over random variables z_{ϵ_k} and z_{ϵ_l}. We have derived closed form solutions for F_Y and F_D in the next subsection, making the above expectation easy to compute via Monte Carlo simulation. One can then proceed to build the network using $f_{i,j}, i = 1,..,M, j = 1,..,M$ as a preference matrix.

2 Derivation of F_D and F_Y.

We approximate the degree cdf via the continuous version of a power law on the interval from $[0.5, M + 0.5]$. Thus $F_D(x) = P(D \le x) = c\int_{0.5}^{x} u^{-\lambda}du = c\frac{0.5^{1-\lambda} - x^{1-\lambda}}{\lambda - 1}$, where c is a normalizing constant, determined from condition $c\int_{0.5}^{M+0.5} u^{-\lambda}du = 1$. This implies $c = \frac{\lambda-1}{0.5^{1-\lambda} - (M+0.5)^{1-\lambda}}$. By inverting the cdf we get $F_D^{-1}(y) = [0.5^{1-\lambda} - \frac{\lambda-1}{c}y]^{\frac{1}{1-\lambda}}$.

For the income distribution, the Pareto has a cdf $F_Y(x) = 1 - \left(\frac{x_m}{x}\right)^{\tau} = 1 - x^{-\tau}$, where $\tau > 0$ and we fixed $x_m = 1$. We find the inverse by setting $y = F_Y(x)$ and solving for y, to get $F_Y^{-1}(y) = (1-y)^{-1/\tau}$.

3 Visualization of networks

We visualize both the preference, and the actual edge matrix as a ratio relative to the edge density of the CM network. Under CM we have $e_{ij} \propto ijq_iq_j$, and this density is included for reference in Figure S1 of the Supplementary Material. In that figure it is interesting to notice that although the CM network is neutral in terms of assortativity, its edge matrix is not flat; in fact, it displays features of both assortative networks, such as a very high fraction of small degree pairs (notably $(1,1)$), as well as other features of disassortative networks, such as concavity and raised wings around $(1, M)$ and $(M, 1)$. Due to some of these features – especially a high peak at $(1,1)$ – featuring prominently in most constructed graphs, displaying the ratio is sound because it allows for better comparison and contrast between graphs. One other issue with visualizing the edge matrix is a spiky appearance in matrix $\boldsymbol{e}$, especially for high degree pairs. This is due to the constructed network having very few nodes with high degrees, leading to zero estimates of e_{ij} for many entries with high i, j. To avoid this problem we smooth the matrix over square patches. Thus, e_{ij} will be averaged over $(i \pm \Delta, j \pm \Delta)$, where $\Delta = 1 + \text{round}(\frac{ij}{25^2})$, meaning that we smooth over a larger area for higher degree pairs.

We note that in Figure 1 A in the main text, the step-like appearance of the preference function (normalized by the CM edge density) is due to the empirical degree distribution in the network being a step function. For instance, there is one vertex in each degree class from 76 – 117, and zero vertices of higher degree. Because we are limited to integer nodes, the fractions of total vertices computed via the power law distribution are rounded to the nearest integer.

4 Implementation details

A couple of notes on algorithm implementation:

(i) In the CM network construction, vertices are also categorized as unexposed (no copies paired), partially exposed (some but not all copies paired), and fully exposed (all copies paired). That algorithm proceeds by giving priority to partially exposed IDs, and matching all their copies before starting to match an unexposed vertex [11]. We do not do this here, as it is not a necessary step, but rather a matter of preference.

(ii) If at the end of the matching process, any stubs cannot be connected, we ignore them and consider the network without the leftover edges. In practice, this might happen for some stubs of the last one or two vertices. This will not impact processes on the network for large N, as it does not involve a significant fraction of nodes/edges.

All simulations were coded in R version 4.3.2. The code is written in base R without the significant use of packages, except for copula generation. Network plots were generated via package 'plotly'.

In terms of algorithm complexity, generating the network takes $O(NM^2)$ time. Given that M is dependent on N in our implementation via the condition $q_M N \approx 0.5$, for a power law we can solve for $M \propto N^{1/\lambda}$. This makes the complexity $O(N^{1+2/\lambda})$. The SIR simulations take $O(NT)$ operations, where T is the number of time steps an epidemic lasts.

In the SIR simulations, we consider that an epidemic enters the deterministic phase, or "takes off", when $I_t > 5n_0$, where n_0 is the number of initial infections. There is no agreed-upon optimal threshold for when the epidemic has taken off, so one should take this definition to be good only in a comparative sense, i.e., as a way to compare networks, and not as an absolute characteristic of a network. We set $n_0 = 15$ in all simulations, except those varying α, where it is set to 5.