ECON5152 — Network Analysis

Week 4: Linear Network Models

Santiago Montoya-Blandón

February 5, 2026

Outline

1. Motivation: first-order approximations
2. Networks as linear operators
3. Paths, walks, and propagation
4. Iteration and equilibrium outcomes
5. Example: Katz-Bonacich centrality
6. Application: Production networks

Motivation: Why Linear Network Models?

- Linear network models are ubiquitous in sciences
- Linear structure arises as a *first-order approximation* to more complex social, economic, or technological relationships
- By deconstructing propagation in steps, achieve complexity
- Linear structure maintains tractability
- Network structure transforms simple *local* effects into *global* through paths and feedback
- Connection between micro and macro worlds

VIA VERITAS VITA
UofG

First-Order Approximations

- Let y be an outcome of interest: GPA, output, prices, portfolio shares, ...
- Environment with n agents $\mathcal{N} := \{1, \dots, n\}$
- Suppose my outcome y_i depends on outcome of others:

$$y_i = f_i(y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_n)$$

First-Order Approximations

- Let y be an outcome of interest: GPA, output, prices, portfolio shares, ...
- Environment with n agents $\mathcal{N} := \{1, \ldots, n\}$
- Suppose my outcome y_i depends on outcome of others:

$$y_i = f_i(y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)$$

- Effect on my outcome y_i from exogenous shock changing y_j to $(1+\beta) \cdot y_j$:

$$y_i \approx f_i(y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_j, \ldots, y_n) + \beta \cdot \frac{\partial f_i}{\partial y_j} \cdot y_j$$

- Assign a name to marginal effects $A_{i,j} = \partial f_i / \partial y_j$ and constant $\alpha_i := f_i$
- Collect all effects into network $\mathbf{A} \implies$ *Linear network models*!

Linear Network Models

- Collect outcomes, marginal effects and constant as

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,n} \\ A_{2,1} & A_{2,2} & \cdots & a_{2,n} \\ \vdots & & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,n} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}$$

- Using matrix notation, we can compactly write

$$\mathbf{y} \approx \boldsymbol{\alpha} + \beta \cdot \mathbf{A}\mathbf{y}$$

Linear Network Models

- Collect outcomes, marginal effects and constant as

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,n} \\ A_{2,1} & A_{2,2} & \cdots & a_{2,n} \\ \vdots & & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,n} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}$$

- Using matrix notation, we can compactly write

$$\mathbf{y} \approx \boldsymbol{\alpha} + \beta \cdot \mathbf{A}\mathbf{y}$$

- Interdependent behaviour approximated as linear
- Useful approximation when size of shocks is not too large
- Common to assume no own shock effects: $A_{i,i} = 0$ for all $i \in \mathcal{N}$

UofG

Example: Peer Effects in Academic Achievement

- Nodes: n students in a class
- Edges: study-group or collaboration links $A_{i,j}$
- Outcome y_i: GPA or exam score of student $i \in \mathcal{N}$
- Assumption: my achievement (GPA) depends on my study group's achievements

$$y_i = \alpha + \beta \sum_{j=1}^{n} A_{i,j} y_j + \varepsilon_i$$

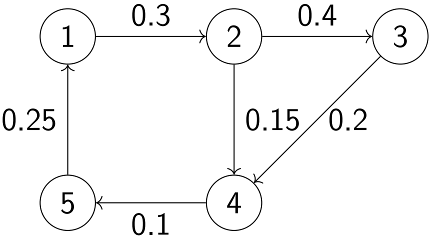
Peer Effects as a Linear Network Model

- Write concisely as

$$\mathbf{y} = \alpha \mathbf{1}_n + \beta \mathbf{A}\mathbf{y} + \boldsymbol{\varepsilon}$$

- α measures average baseline GPA
- β measures effect of others' GPA on my own (*peer effects*)
- $A_{i,j}$ encodes who studies with whom and/or how intensely
- ε_i represents all other factors that affect GPA for i
- Same structure appears in centrality, diffusion, production networks, and many others

An example network



Matrix Representation

$$
\begin{array}{cc}
 & \text{Columns} = \text{sources} \\
\text{Rows} = \text{receivers} & \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix}
\end{array}
$$

A matrix encodes a *linear map of structured interactions*

Matrix Representation

Columns = sources

Rows = receivers

$$
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\
a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\
a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\
a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\
a_{51} & a_{52} & a_{53} & a_{54} & a_{55}
\end{bmatrix}
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}
\longrightarrow
\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix}
$$

Input source vector and output a linear combination of source values

Networks as Linear Operators

Think of $\mathbf{A}$ as a machine:

$$x \mapsto \mathbf{A}x$$

- Each node aggregates weighted inputs from neighbours
- Direction and weights matter
- Repeated application compounds effects

$$x \mapsto \mathbf{A}x \mapsto \mathbf{A}^2x \mapsto \mathbf{A}^3x \mapsto \cdots$$

- Large complexity arises from simple linear rules

VIA VERITAS VITA
UofG

From Micro to Macro

- Each edge $A_{i,j}$ of $\mathbf{A}$ represents a small interaction
- Networks create:
  - chains of interactions
  - multiple paths between nodes
  - cycles and feedback loops
- Repeated linear propagation of small shocks generates large systemic effects
- Network structure determines:
  - strength of amplification
  - reach of shocks
  - systemic importance of nodes

Paths, Walks and Propagation

- $\mathbf{A}_{i,j}$: direct effect from j to i
- $(\mathbf{A}^2)_{i,j}$: all 2-step walks from i to j
- $(\mathbf{A}^k)_{i,j}$: all walks of length k from i to j

Each multiplication aggregates over *all possible intermediaries*

$$(\mathbf{A}^2)_{i,j} = \sum_{k=1}^{n} A_{i,k} A_{k,j}$$

- Products $\implies$ path strength
- Sums $\implies$ multiple routes
- Cycles re-enter naturally

Propagation as Iteration

- Start with an initial distribution of outcomes (for example, equal):

$$\mathbf{y}^{(0)} = \mathbf{1}_n$$

- Set constants equal to initial levels: $\boldsymbol{\alpha} = \mathbf{y}^{(0)}$
- Iterate under the linear network model given by $\mathbf{A}$:

$$\mathbf{y}^{(t+1)} = \boldsymbol{\alpha} + \beta \mathbf{A} y^{(t)}$$

- Each step adds longer paths to the calculation

Iteration and Equilibrium Outcomes

- Expanding iteratively:

$$\begin{aligned}
\mathbf{y}^{(1)} &= \mathbf{1}_n + \beta \mathbf{A}\mathbf{1}_n \\
\mathbf{y}^{(2)} &= \mathbf{1}_n + \beta \mathbf{A}\mathbf{1}_n + \beta^2 \mathbf{A}^2\mathbf{1}_n \\
\mathbf{y}^{(3)} &= \mathbf{1}_n + \beta \mathbf{A}\mathbf{1}_n + \beta^2 \mathbf{A}^2\mathbf{1}_n + \beta^3 \mathbf{A}^3\mathbf{1}_n \\
&\vdots \\
\mathbf{y}^{(t)} &= \mathbf{1}_n + \beta \mathbf{A}\mathbf{1}_n + \beta^2 \mathbf{A}^2\mathbf{1}_n + \beta^3 \mathbf{A}^3\mathbf{1}_n + \cdots + \beta^t \mathbf{A}^t\mathbf{1}_n
\end{aligned}$$

- If elements of $\mathbf{A}$ are "small enough" then

$$\mathbf{y}^* := \lim_{t\to\infty} \mathbf{y}^{(t)} = (\mathbf{I}_n - \beta \cdot \mathbf{A})^{-1}\mathbf{1}_n = \sum_{k=0}^{\infty} \beta^k \mathbf{A}^k \mathbf{1}_n$$

- Convergence is a property of the *network structure*

Centrality as a Linear Network Outcome

Bonacich / Katz centrality:

$$\mathbf{c} = \mathbf{1}_n + \beta \mathbf{A}\mathbf{c}$$

- Important nodes are connected to important nodes
- Centrality aggregates all paths, discounted by length

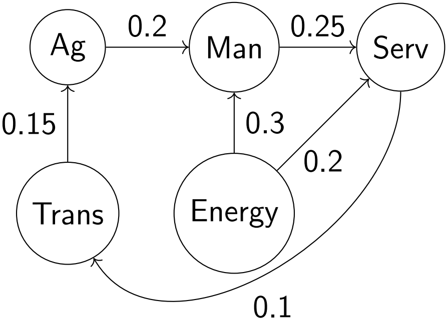
Compare centrality intuition

- Degree: only $\mathbf{A}$
- Eigenvector centrality: fixed point of $\mathbf{A}$
- Katz centrality: Neumann (power or iterative) sum of $\mathbf{A}$

Transition to Production Networks

- Replace “influence” with “input requirement”
- Replace “importance” with “output”
- Same linear propagation logic

Input–Output Network Structure



IO Matrix

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0.15 \\ 0.2 & 0 & 0 & 0.3 & 0 \\ 0 & 0.25 & 0 & 0 & 0.2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 \end{pmatrix}$$

A_{ij} = input from j required to produce unit output of i

Leontief Model

$$y = d + \mathbf{A}y \quad \Rightarrow \quad y = (I - \mathbf{A})^{-1} d$$

- d: final demand
- $(I - \mathbf{A})^{-1}$: total requirements matrix

Economic interpretation

- Shocks propagate through supply chains
- Cycles amplify disturbances
- Network position determines systemic importance

Big Picture

- Linear models + networks = complexity
- Paths, not just links, drive outcomes
- Production networks are one powerful application explored in lab and assignment