ECON5152 — Network Analysis

Week 2: Network Types and Statistics

Santiago Montoya-Blandón

January 22, 2026

Outline

- Size and density
- Local connectivity: degree and weighted degree
- Paths, walks, and cycles
- Connectivity, components, and distance
- Canonical network types
  - Complete
  - Trees
  - Stars
  - Lattices
  - Bipartite
- Applications
  - Input–Output (IO) tables
  - Bipartite country-product network
  - Political division

Deterministic Network Structure

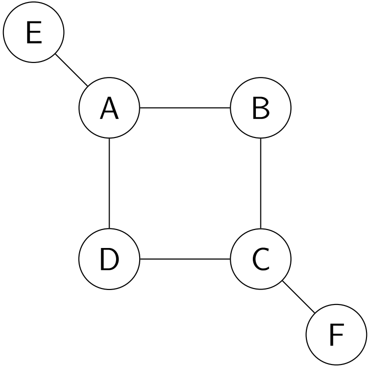
- We begin by studying networks $\mathcal{G}(\mathcal{N}, \mathcal{E})$ as fixed objects
- Postpone study of network formation
- No randomness or strategic behaviour
- Focus on network structure and its implications
- Concepts provide foundation for later statistical and dynamic models

Simplest Network Taxonomy

Major categories to divide networks

- Is the network directed?
  - Yes $\rightarrow$ asymmetric connections and adjacency matrix
  - No $\rightarrow$ symmetric connections and adjacency matrix
- Is the network weighted?
  - Yes $\rightarrow$ account for weights
  - No $\rightarrow$ use binary (0 and 1) representation
- Does the network allow self-loops?
  - Yes $\rightarrow$ node can connect with itself (e.g., sector trade)
  - No $\rightarrow$ edges only between distinct nodes (zero-diagonal adjacency matrix)

Running Example: A Small Network



This network will be used to illustrate concepts throughout the lecture.

Running Example: Graph and Adjacency

To practice previous concepts: for running example graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, find

- Nodes:
- Number of nodes:
- Edges:
- Number of edges:
- Adjacency:

Running Example: Graph and Adjacency

To practice previous concepts: for running example graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, find

- Nodes: $\mathcal{N} = \{A, B, C, D, E, F\}$
- Number of nodes: 6
- Edges:
- Number of edges:
- Adjacency:

Running Example: Graph and Adjacency

To practice previous concepts: for running example graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, find

- Nodes: $\mathcal{N} = \{A, B, C, D, E, F\}$
- Number of nodes: 6
- Edges: $\mathcal{E} = \{\{A, E\}, \{A, B\}, \{A, D\}, \{B, C\}, \{C, D\}, \{C, F\}\}$
- Number of edges: 6
- Adjacency:

Running Example: Graph and Adjacency

To practice previous concepts: for running example graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, find

- Nodes: $\mathcal{N} = \{A, B, C, D, E, F\}$
- Number of nodes: 6
- Edges: $\mathcal{E} = \{\{A, E\}, \{A, B\}, \{A, D\}, \{B, C\}, \{C, D\}, \{C, F\}\}$
- Number of edges: 6
- Adjacency:

$$\mathbf{A}(\mathcal{G}) = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

UofG

Sizes and Density

Size

The simplest measure of *size* are the number of nodes and number of edges.

Density

The *density* is the ratio of the number of edges to the total potential links.

- Number of total potential links depends on structure (assume n nodes):
  - Directed & self-loops $\rightarrow n^2$
  - Directed & no self-loops $\rightarrow n(n-1)$
  - Undirected & self-loops $\rightarrow n(n+1)/2$
  - Undirected & no self-loops $\rightarrow n(n-1)/2$
- Density is within 0 (no edges) and 1 (all possible edges)
- Density allows comparison of networks of different sizes

Degree: Definition

Degree

The *degree* d_i of each node $i \in \mathcal{N}$ is the number of edges connected to it.

- Degree distribution (collection of degrees): $\mathbf{d} = \{d_i\}_{i \in \mathcal{N}}$
- Measures local connectivity
- In directed networks
  - In-degree: number of incoming edges to each node
  - Out-degree: number of outgoing edges from each node

Weighted Degree

Weighted Degree

The *weighted degree* w_i of each node $i \in \mathcal{N}$ is the sum of weights from edges connected to a node.

- Captures intensity of interactions
- Central in economic and financial networks
- In directed networks
  - Weighted in-degree: sum of incoming edge-weights to each node
  - Weighted out-degree: sum of outgoing edge-weights from each node

Running Example: Degree

- Degree(A) $= 3$ (B, D, E)
- Degree(B) $= 2$ (A, C)
- Degree(C) $= 3$ (B, D, F)
- Degree(D) $= 2$ (A, C)
- Degree(E) $= 1$ (A)
- Degree(F) $= 1$ (C)

Nodes A and C are locally more connected than others.

Walks, Paths, and Cycles

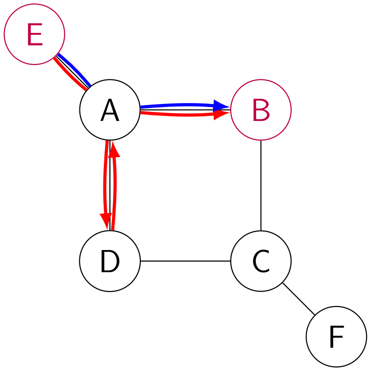
Definitions

- A *walk*: sequence of connected nodes (allows repetitions)
- A *path*: walk with no repeated nodes
- A *cycle*: path that starts and ends at the same node

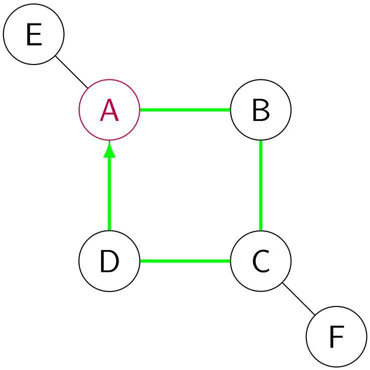
These concepts describe indirect connections.

Running Example: Walk, Paths and Cycles

A walk and a path between B and E



A cycle starting and ending in A



Distance, Diameter and Shortest Path

Shortest Path

The *shortest path* between two nodes $(i, j) \in \mathcal{N}$ contains the smallest possible number of edges.

Distance

The *distance* between two nodes $(i, j) \in \mathcal{N}$ is the length of the *shortest path* connecting them.

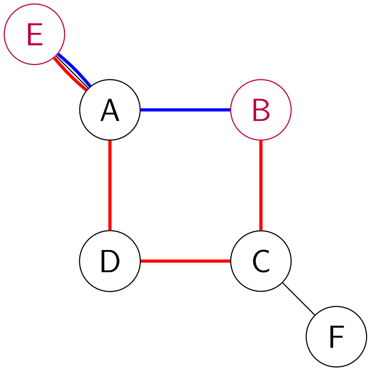
Diameter

The *diameter* is the maximum distance between any two nodes in $\mathcal{N}$.

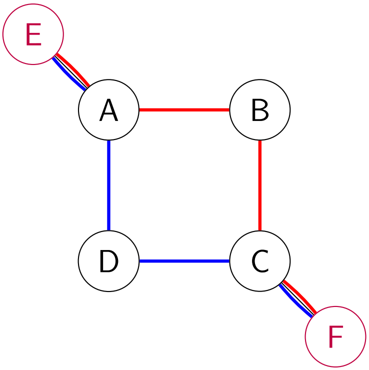
These summarize global network reach.

Running Example: Paths and Distances

Two paths between E and B



Two shortest paths between E and F



Running Example: Paths and Distances

- Distance(B, E) $=$ Length of Shortest path from B to E $=$ 2
- Shortest path: E $\longleftrightarrow$ A $\longleftrightarrow$ B
- Distance (E, F) $=$ Length of Shortest path from E to F $=$ 4
- One shortest path: E $\longleftrightarrow$ A $\longleftrightarrow$ B $\longleftrightarrow$ C $\longleftrightarrow$ F $=$ 4
- Diameter($\mathcal{G}$) $=$ 4

Connectivity and Components

Connectivity

A network is *connected* if a path exists between every pair of nodes.

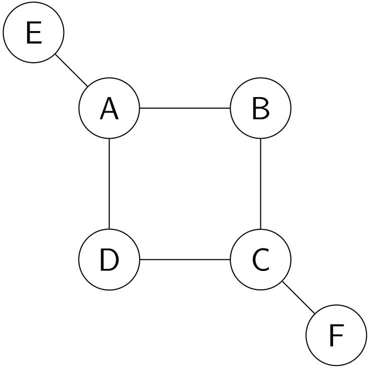
Subgraph

A *subgraph* is composed of a subset of edges and their corresponding nodes from the original graph.

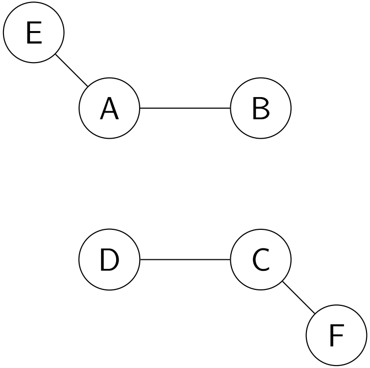
- If not connected, network divides into *components* (connected subgraphs)
- *Largest connected component*: connected subgraph with largest diameter
- Presence of components limit diffusion and interaction

Running Example: Connectivity

Connected: Single component with diameter 4



Disconnected: 2 components with diameter 2 each



Digression: How to find shortest paths?

- For simple networks, shortest paths can be identified by hand
- How to systematically identify distances in any network?

Digression: How to find shortest paths?

- For simple networks, shortest paths can be identified by hand
- How to systematically identify distances in any network?
- Introduce *breadth-search first*: simplest algorithm for finding shortest paths
- Intuition: trace out distances one by one (*broad* search at each step)

Digression: How to find shortest paths?

- For simple networks, shortest paths can be identified by hand
- How to systematically identify distances in any network?
- Introduce *breadth-search first*: simplest algorithm for finding shortest paths
- Intuition: trace out distances one by one (*broad* search at each step)

Step 1 Select a node $i \in \mathcal{N}$ at random

Step 2 Identify all nodes a distance of 1 away from i, denoted
$\mathcal{D}_1 := \{j \in \mathcal{N} : \text{Distance}(i, j) = 1\}$

Step 3 For each nodes in $\mathcal{D}_1$, identify other nodes a further distance of 1 away (aside from i and those you have already visited)

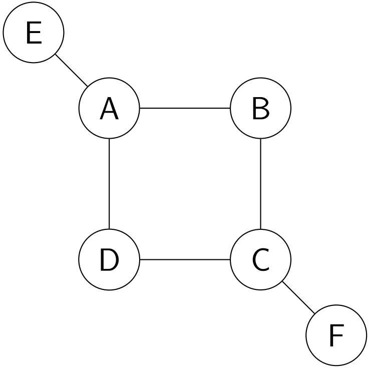
Step 4 Collect these nodes in $\mathcal{D}_2$. These are a distance of 2 away from i

Step 5 Repeat process until you visit all possible nodes, increasing distance by 1

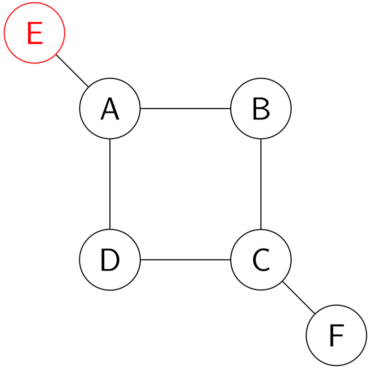
- $\text{Distance}(i, j) =$ Number of steps until reaching node $j \in \mathcal{N}$

UofG

Running Example: Bread-Search First

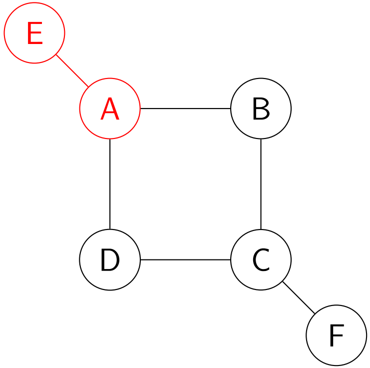


Running Example: Bread-Search First



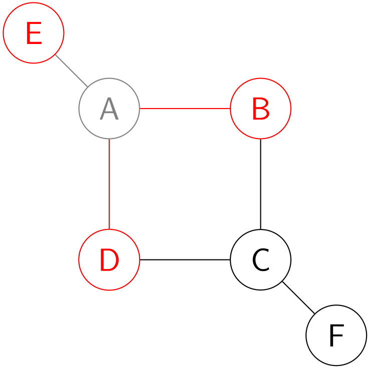
1. Begin search at node E

Running Example: Bread-Search First



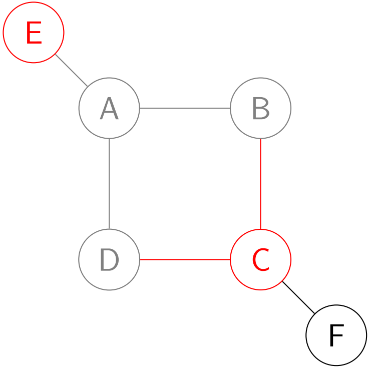
1. Begin search at node E
2. Nodes distance of 1 away from E:
   $\mathcal{D}_1 = \{A\}$

Running Example: Bread-Search First



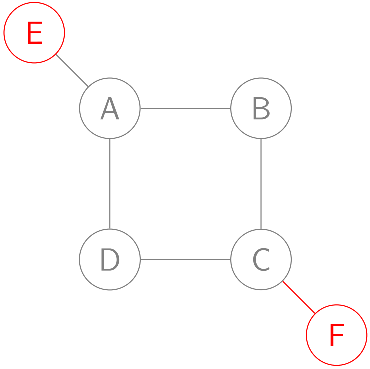
1. Begin search at node E
2. Nodes distance of 1 away from E:
   $\mathcal{D}_1 = \{A\}$
3. Nodes distance of 2 away from E:
   $\mathcal{D}_2 = \{B, D\}$

Running Example: Bread-Search First



1. Begin search at node E
2. Nodes distance of 1 away from E: $\mathcal{D}_1 = \{A\}$
3. Nodes distance of 2 away from E: $\mathcal{D}_2 = \{B, D\}$
4. Nodes distance of 3 away from E: $\mathcal{D}_3 = \{C\}$

Running Example: Bread-Search First



1. Begin search at node E
2. Nodes distance of 1 away from E: $\mathcal{D}_1 = \{A\}$
3. Nodes distance of 2 away from E: $\mathcal{D}_2 = \{B, D\}$
4. Nodes distance of 3 away from E: $\mathcal{D}_3 = \{C\}$
5. Nodes distance of 4 away from E: $\mathcal{D}_4 = \{F\}$
6. Algorithm provides all distances within connected components

Network Types

We identify a few important special classes of networks:

- Complete
- Trees
- Stars
- Lattices
- Bipartite

Provide general definition and small examples.

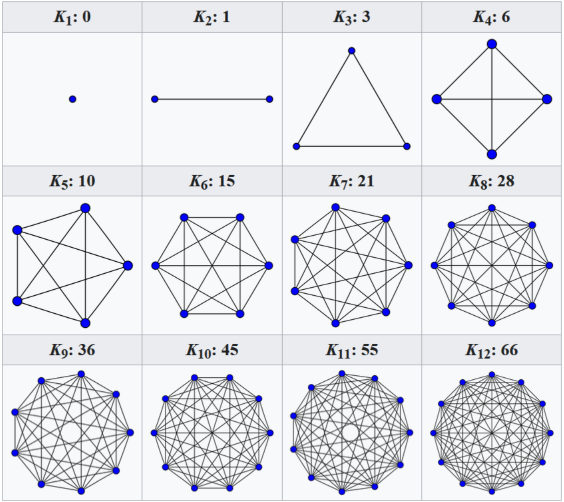
Complete: Definition

Complete

A *complete* graph is one where all nodes are connected to each other.

- All paths have distance 1
- If n nodes, diameter equals n
- Each node has maximum degree $n - 1$
- Maximal connectivity structure
- Simple approximation to unknown networks
- However, most networks are not complete

Complete: Example



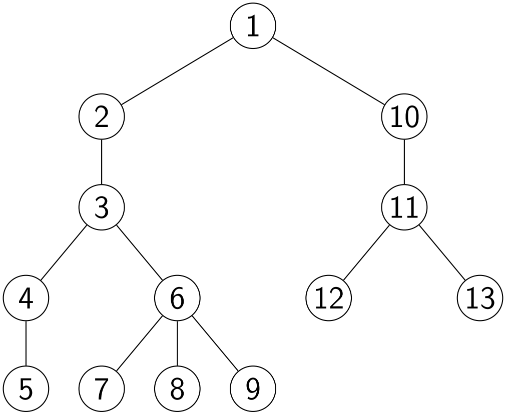
Trees: Definition

Tree

A *tree* is a connected network with no cycles.

- Exactly one path between any two nodes
- Number of edges equals number of nodes minus one
- Minimal connectivity structure
- Trees are *fragile*: breaking any one link disconnects the network
- Common in organizational networks, computer science, biology

Trees: Example



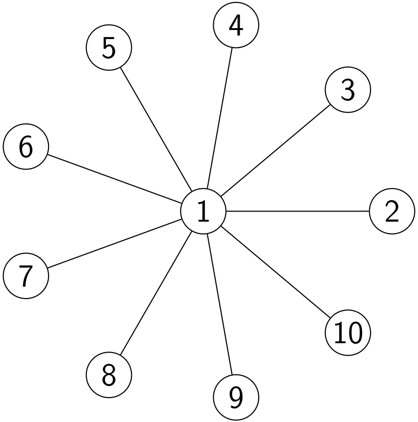
Stars: Definition

Star

A *star* network is one where every node connects only to a central (or star) node.

- Divides nodes into two categories: star vs. non-stars
- Simplex example of core-periphery network
- If n nodes, $n-1$ edges present
- Heterogeneous connectivity: low for non-stars (degree 1), maximum for star (degree $n-1$)
- Heterogeneous fragility:
  - Can remove non-stars without altering network structure
  - Full disconnection (no paths between nodes) if star is removed
- Common in computer science, economics, geography, organizations

Stars: Example



Lattices: Definition

Lattice

A *lattice* is a network where nodes are arranged in a regular, repeating pattern.

- Each node has a similar local neighborhood
- Used to model local interactions and geographic constraints
  - Often embedded in space (grid coordinates)
  - Abstraction from complex geographical features
  - Crystal structures
- Lattices are strongly connected and resilient to fragility

Lattices: Example

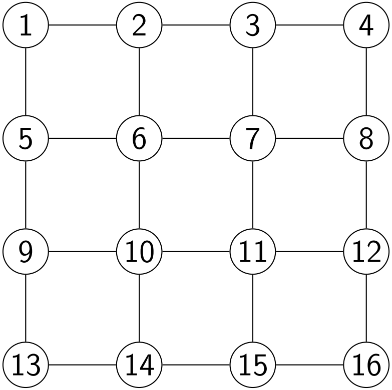


Figure: Lattice network

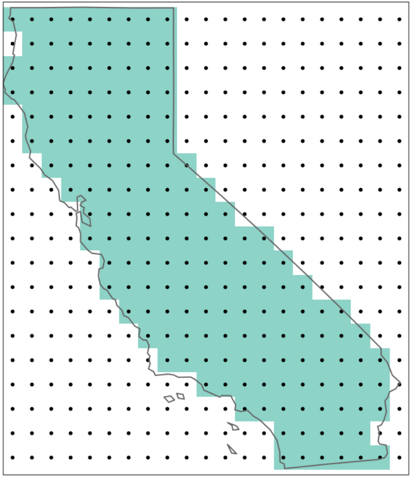


Figure: Lattice over California, USA

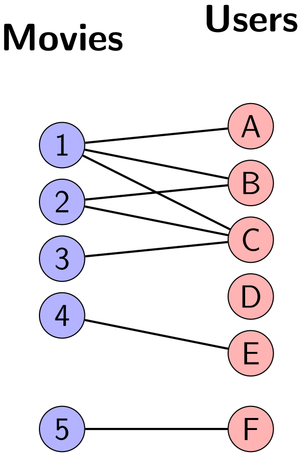
Bipartite Networks: Definition

Bipartite Network

A *bipartite network* consists of two disjoint sets of nodes, with edges running only across the sets.

- Nodes are naturally separated into two categories
- No edges within nodes of the same category
- Natural representation of matching relationships
  - Markets: buyers mapped to sellers
  - Employment networks: employees matched to firms
  - Recommendation networks: movies or series watched by users
  - Dictionary search: keys correspond to a given value (words and definitions)

Bipartite Networks: Example



Application: Supply Chain or Input–Output (IO) Networks

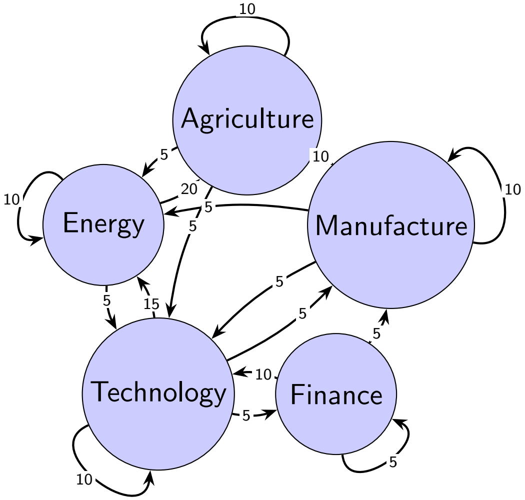
- Nodes represent firms, industries or sectors
- Edges represent input–output relationships: buying inputs to produce goods and services
- Paths capture indirect dependence amongst buyers and suppliers
- Long paths imply vulnerability to upstream shocks
- Economists use terminology IO while others use supply chains

Application: Small Input–Output Table

	Agriculture	Energy	Technology	Finance	Manufacture
Agriculture	10	5	5	0	0
Energy	20	10	5	0	0
Technology	0	15	10	5	5
Finance	0	0	10	5	5
Manufacture	10	5	5	0	10

- Rows: buyers (sectors purchasing from others)
- Columns: suppliers (sectors selling to others)
- Directed network
- Edges represent value traded (say in millions of £)
- Self-loops allowed (sectors buy from themselves)

Application: IO Table as a Weighted Graph



Application: Weighted Directed Degrees in IO Table

- Weighted in-degree: row sums of adjacency $w_i^{\text{in}} := \sum_{j=1}^{n} A_{i,j}$
- Captures upstream dependence in IO context
- $w_{\text{Agriculture}}^{\text{in}} = 20$, $w_{\text{Energy}}^{\text{in}} = 35$, $w_{\text{Technology}}^{\text{in}} = 35$, $w_{\text{Finance}}^{\text{in}} = 20$, $w_{\text{Manufacture}}^{\text{in}} = 30$

Application: Weighted Directed Degrees in IO Table

- Weighted in-degree: row sums of adjacency $w_i^{\text{in}} := \sum_{j=1}^{n} A_{i,j}$
- Captures upstream dependence in IO context
- $w^{\text{in}}_{\text{Agriculture}} = 20,\ w^{\text{in}}_{\text{Energy}} = 35,\ w^{\text{in}}_{\text{Technology}} = 35,$ $w^{\text{in}}_{\text{Finance}} = 20,\ w^{\text{in}}_{\text{Manufacture}} = 30$
- Weighted out-degree: column sums of adjacency $w_i^{\text{out}} := \sum_{j=1}^{n} A_{j,i}$
- Captures downstream importance in IO
- $w^{\text{out}}_{\text{Agriculture}} = 40,\ w^{\text{out}}_{\text{Energy}} = 35,\ w^{\text{out}}_{\text{Technology}} = 35,$ $w^{\text{out}}_{\text{Finance}} = 10,\ w^{\text{out}}_{\text{Manufacture}} = 20$

Application: Weighted Degrees in IO table

- High *in-degree* (row sum):
  - sector relies heavily on intermediate inputs
  - vulnerable to upstream shocks
- High *out-degree* (column sum):
  - sector supplies many others
  - important for shock propagation
- Heterogeneity increases with number and aggregation of sectors
- In practice, IO table transformed to adjacency matrix by dividing over total sector output and setting small entries to 0

Application: Country-Product Networks

- Caldarelli, G., Cristelli, M., Gabrielli, A., Pietronero, L., Scala, A., & Tacchella, A. (2012). "A Network Analysis of Countries' Export Flows: Firm Grounds for the Building Blocks of the Economy", *PLOS One* 7(10) (`doi:10.1371/journal.pone.0047278`).
- Consider a set of countries $\mathcal{C}$ and products $\mathcal{P}$
- Exports of product $p \in \mathcal{P}$ from country $c \in \mathcal{C}$ denoted as $X_{c,p}$
- Revealed Comparative Advantage (RCA) of product is measured as country export share / global export share:

$$\mathrm{RCA}_{c,p} := \frac{X_{c,p}}{\sum_{q \in \mathcal{P}} X_{c,q}} \Bigg/ \frac{\sum_{d \in \mathcal{C}} X_{d,p}}{\sum_{d \in \mathcal{C}} \sum_{q \in \mathcal{P}} X_{d,q}}$$

- Define following network: $c \in \mathcal{C}$ and $p \in \mathcal{P}$ are connected if $\mathrm{RCA}_{c,p} > 1$
- Intuitively: bipartite network connecting countries to products they export

Application: Country-Product Network

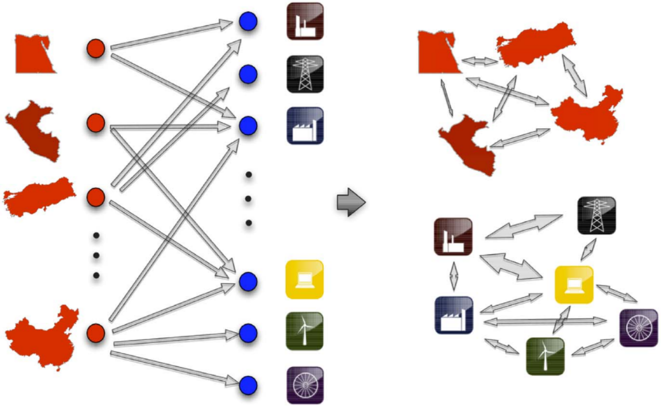


Figure: Bipartite structure of countries and their competitively exported products. Projections into networks on countries $\mathcal{C}$ and products $\mathcal{P}$ included.

Application: Political Preferences and Division

- Adamic, L. A., & Glance, N. (2005). "The political blogosphere and the 2004 US election: Divided they blog". *Proceedings of the 3rd international workshop on Link discovery*, pp. 36–43. dl.acm.org/doi/10.1145/1134271.1134277
- Paper maps network structure of political blogs
- Around election time, blogs actively link to each other in posts
- Citation network contains evidence of clear political division
- Two clear groups emerge, well connected within each group but less connected across groups

Application: Political Preference and Division

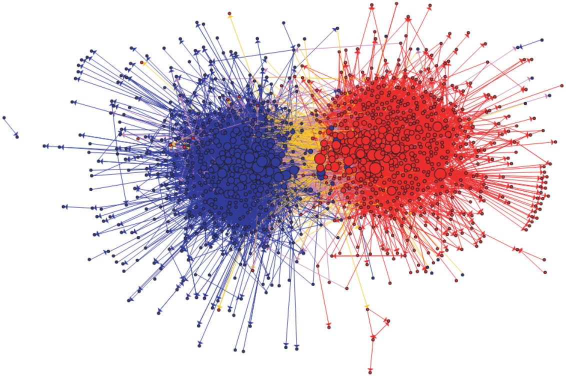
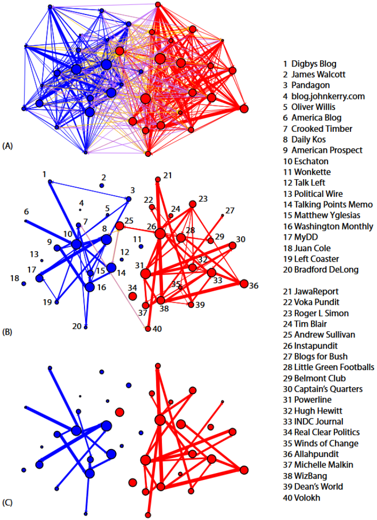


Figure: Community structure of political blogs in the United States around the 2004 election. Red for conservatives and blue for liberals. Orange links go from liberal to conservative, and purple ones from conservative to liberal. Size of each blog reflects number of other blogs that link to it.

Application: Political Preference and Division



- (A) Same network construction applied to top 40 blogs
- (B) and (C) limit connections to a given intensity (number of link citations)
- Network definition creates disconnected components

Summary: Network Measure Interpretation

Network Measure	Interpretation
Degree	Local connectivity / diversification
Weighted degree	Exposure or intensity
Paths	Indirect dependence
Distance	Speed of diffusion or communication
Diameter	Overall reach of the system
Components	Fragmentation or isolation