Social Network Analysis
Graph Representations

Robin Burke

DePaul University

Chicago, IL

Outline

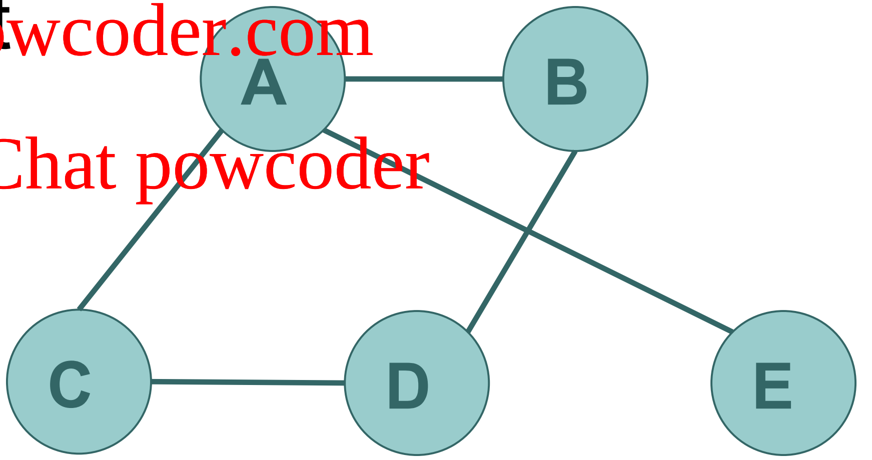
- Internal representations
  - edge list
  - adjacency matrix
  - adjacency list
  - database table
- Bipartite networks
- Example

Network representation

- Pictures are nice!
  - easy for humans to examine at a glance
- A picture is just one representation of a network
  - not necessarily the best one for all purposes
- Computers are not so good at looking at pictures

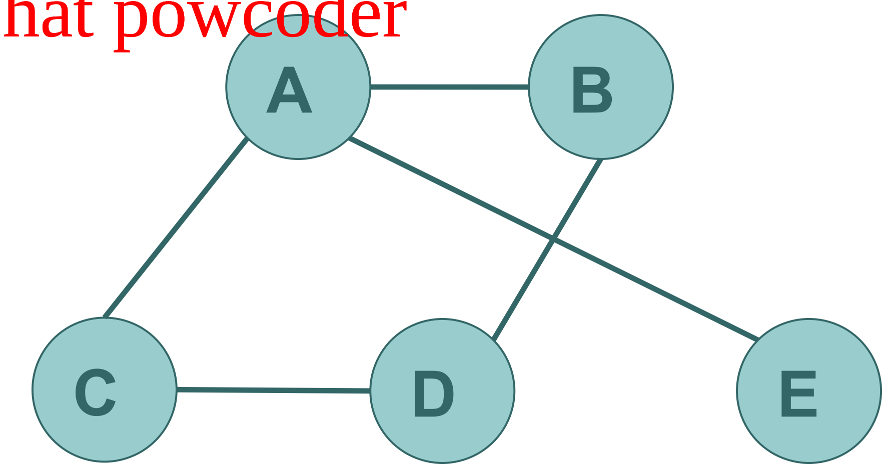
Internal representations

- Edge list
- Adjacency matrix
- Adjacency list



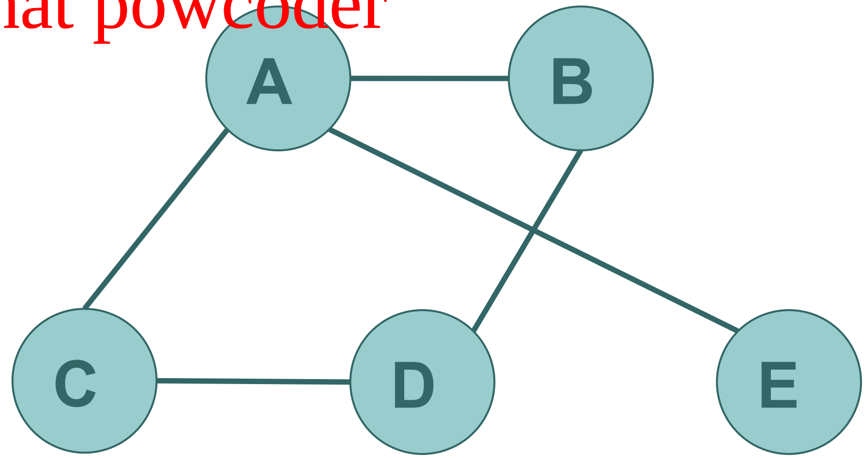
Edge list

- A graph is a list of edges
  - usually we include a list of nodes, too
  - but not strictly necessary
- Edge list…



Hard to compute with

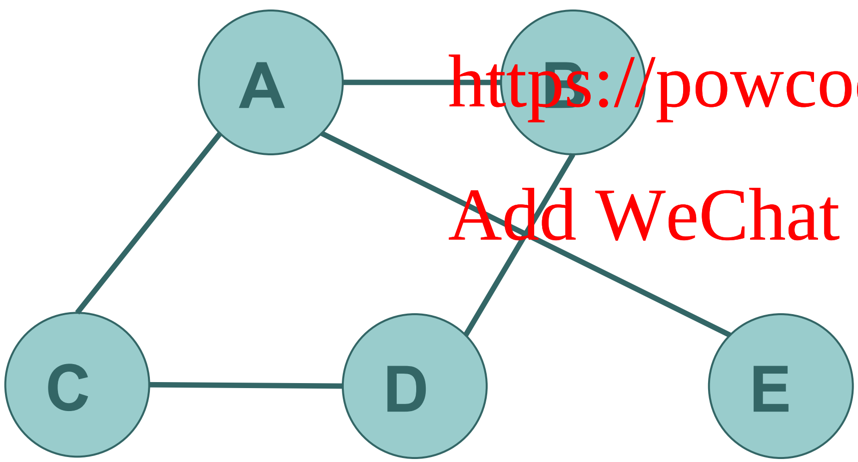
- Suppose we want to see if there's a path from node D to node E.
- Repeated searches of the list



Adjacency Matrix

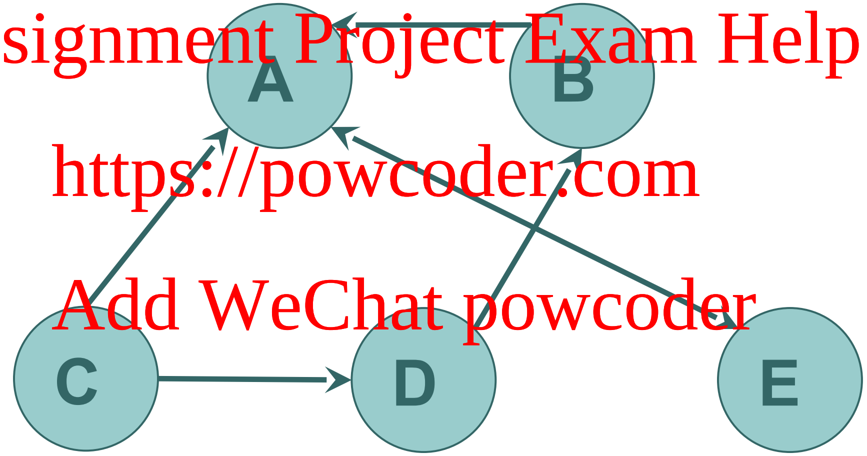
- Most important graph representation is the adjacency matrix
  - also “sociomatrix”
- Square marix with an entry for each pair of nodes
  - non-zero entry if the nodes are connected
- We will use this representation a lot
  - fundamental to network, sna, and other R packages

Matrix



$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Directed graph?



Matrix

- Great for graph computing
- Matrix multiplication = path traversal
  - more on this later

Sparsity

- Most networks are sparse
  - only a small percentage of possible edges exist

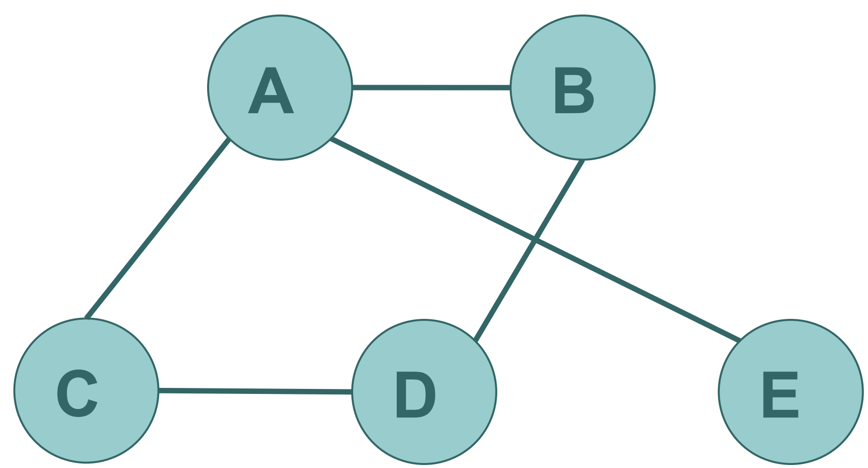
- Make sense especially in social networks

  - 9.5 million people in metro Chicago
  - If 1% of the edges existed

    - everyone would have to know 95,000 others
  - 0.001% more likely
- Adjacency matrix is inefficient for sparse networks
  - matrix is 99.999% zeros

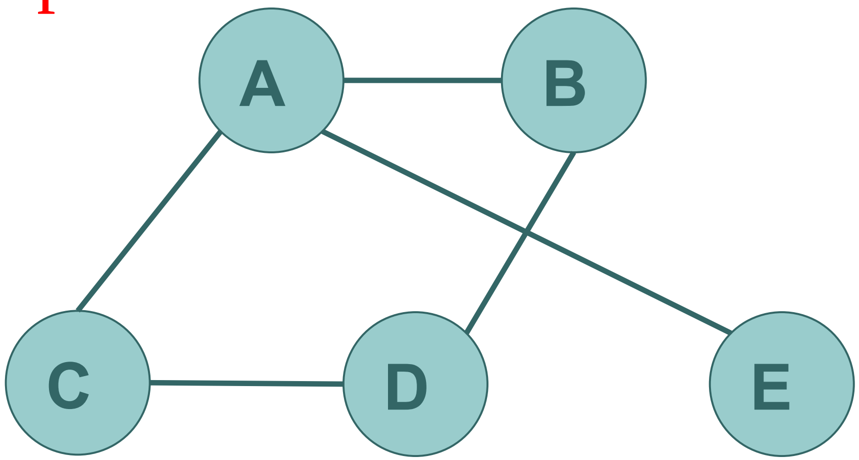
Adjacency list

- Adjacency list is a compromise
  - For each node, a list of nodes you can get to
- Adjacency list...
- There is redundant information
  - but computation is easier
- Typically this is the internal representation for sparse matrices



Quick question

- What is the adjacency list representation for node A
- A: A -> { A, B, C, D, E }
- B: A -> { B, E, D }
- C: A -> { B, C, E }
- D: A -> { A, B, C, E }



...t is the adjacency list representation for nod...

A -> { A, B, C, D, E }

A -> { B, E, D }

A -> { B, C, E }

A -> { A, B, C, E }

Start the presentation to activate live content

If you see this message in presentation mode, install the add-in or get help at PollEv.com/app

Graph representations

- Edge list
  - compact, good for loading the network
  - bad for computation
- Adjacency matrix
  - space-inefficient, mathematically elegant
- Adjacency list
  - reasonable compromise

Database storage

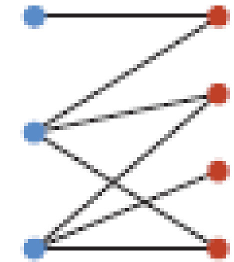
- Edge list
  - table of <i,j> pairs for each edge

- Adjacency matrix

  - table of <i,j,value> for all i,j
    - where value = 0 if no edge, 1 if edge

  - still inefficient
- Adjacency list
  - not so easy to store directly
  - index against edge list for the same effect

Bipartite Networks



- Data where the vertices are divided into classes
  - edges can only go between classes

- Typical example
  - affiliation network

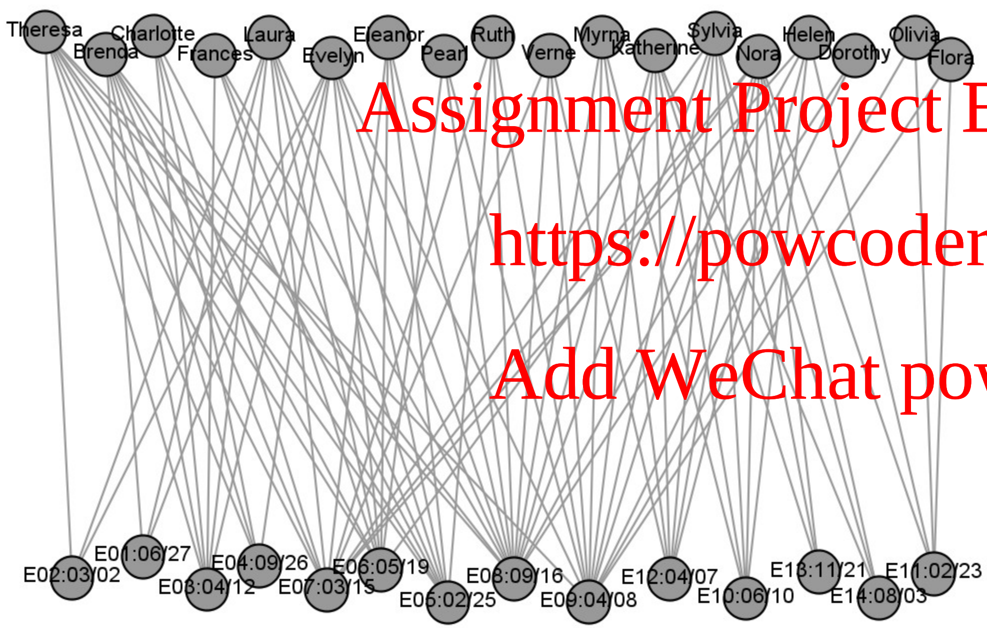
  - DePaul courses -> students enrolled in those courses
  - only edges between students and courses

    - no course-course or student-student edges
- Also two-mode network
- Can be extended to additional types
  - Professors -> courses -> students
  - three-mode network

Bipartite networks in social media

- Very common structure
  - users are connected
    - by commenting on the same item
    - by using the same hashtag in posts
    - by "liking" the same post
    - etc.
- A way to extract a community with common interests

Bipartite Example



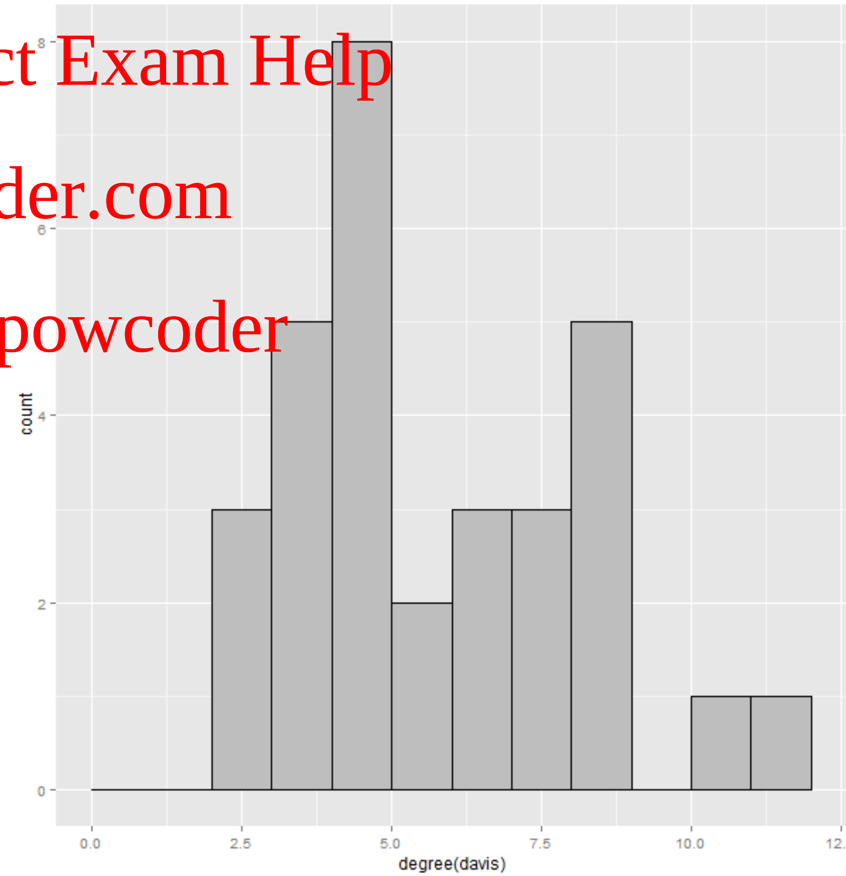
- Two-mode data
  - women and social events in Nachez, MS in the 1930s
  - “davis” data

Key point

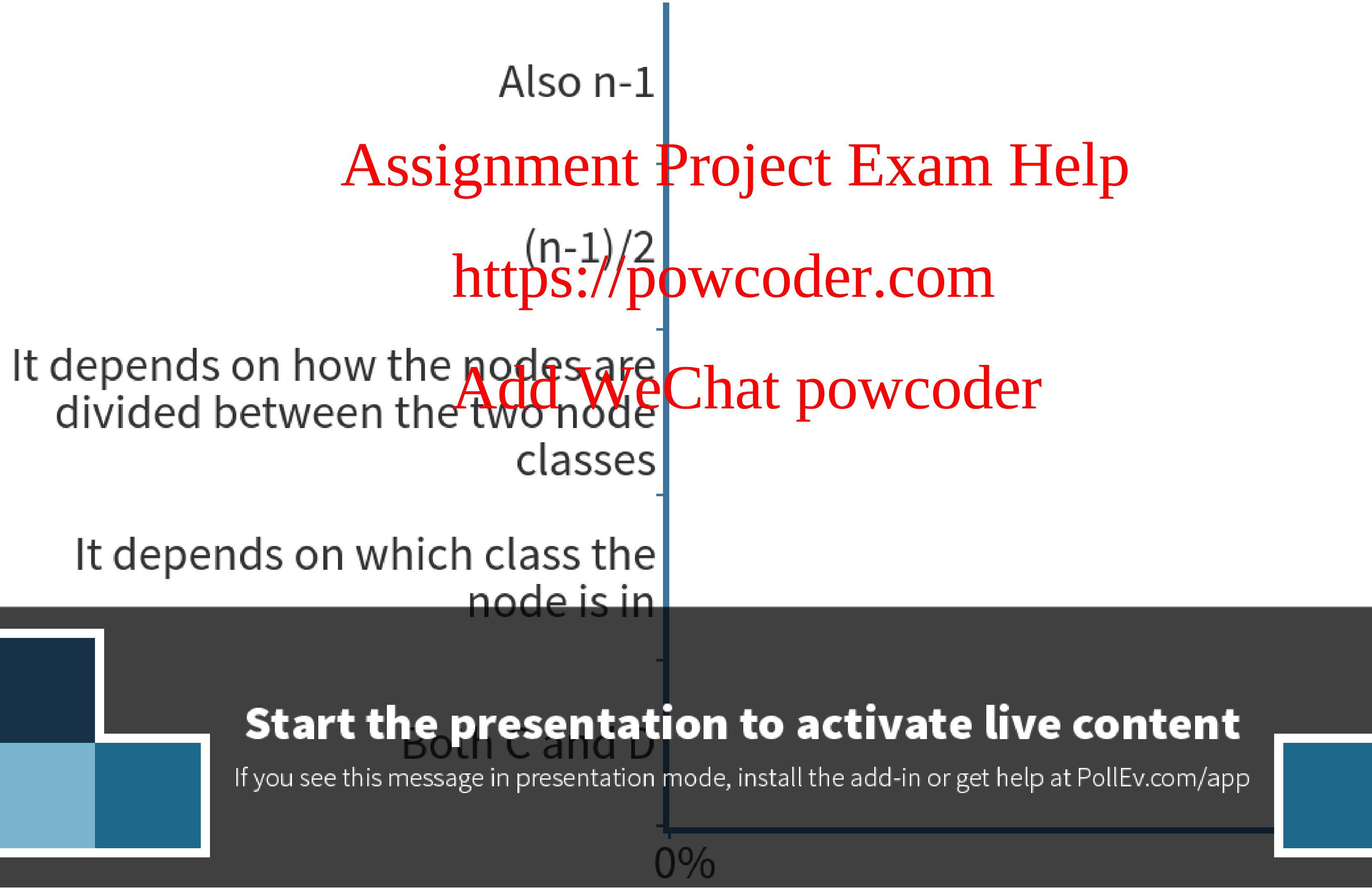
- Bipartite networks are not that useful in themselves
  - strange structure
  - computations have different meaning

Example

- Degree distribution
  - Davis data
  - This is really hard to make sense of
- Two different kinds of nodes mixed together



est degree a node can have in an undirected is n-1. What is the highest degree a node can bipartite network?



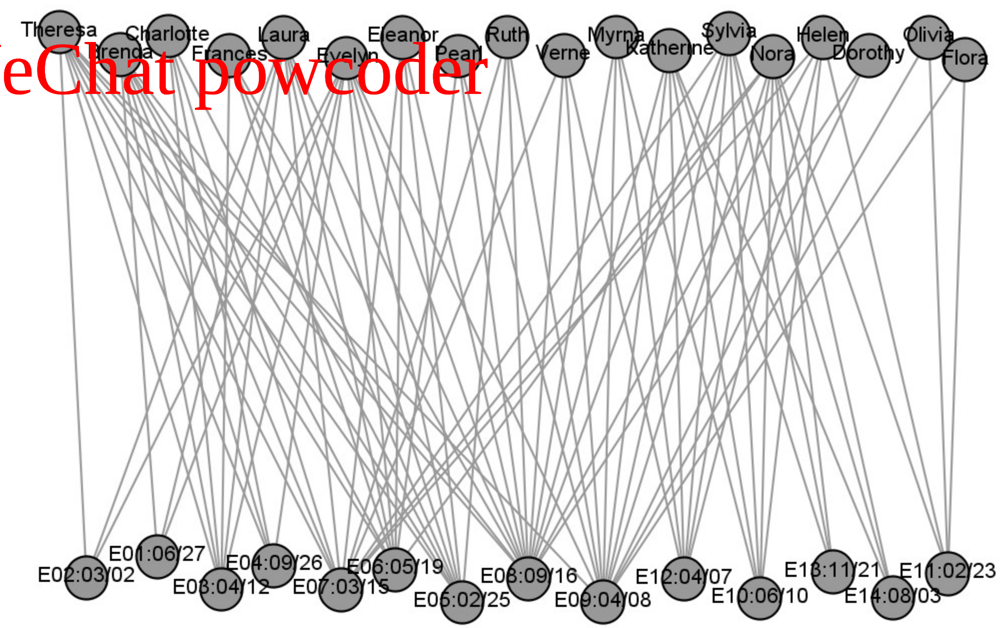
Projections

- We can "project" a two-mode network

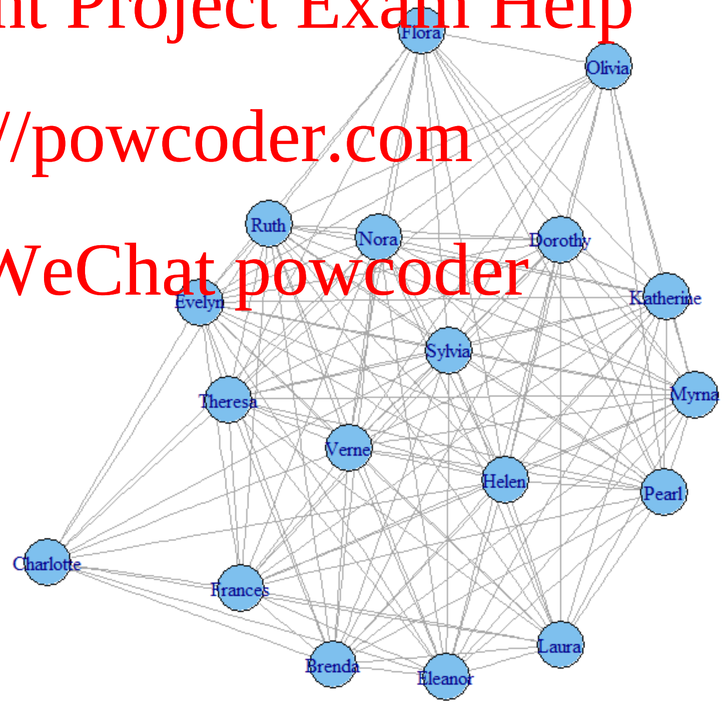
  - into two single mode networks

- Example

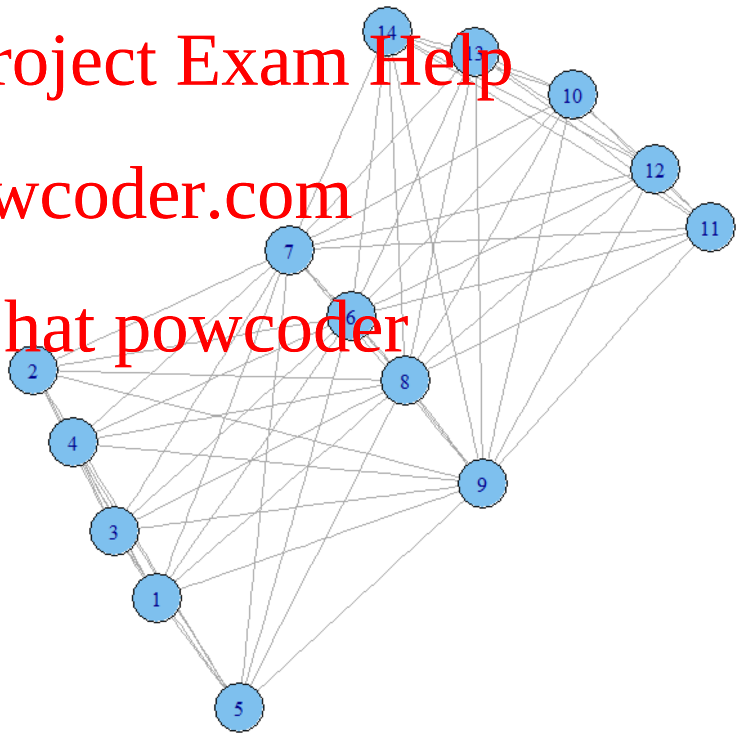
  - Davis data



Projection 1: Person-Person

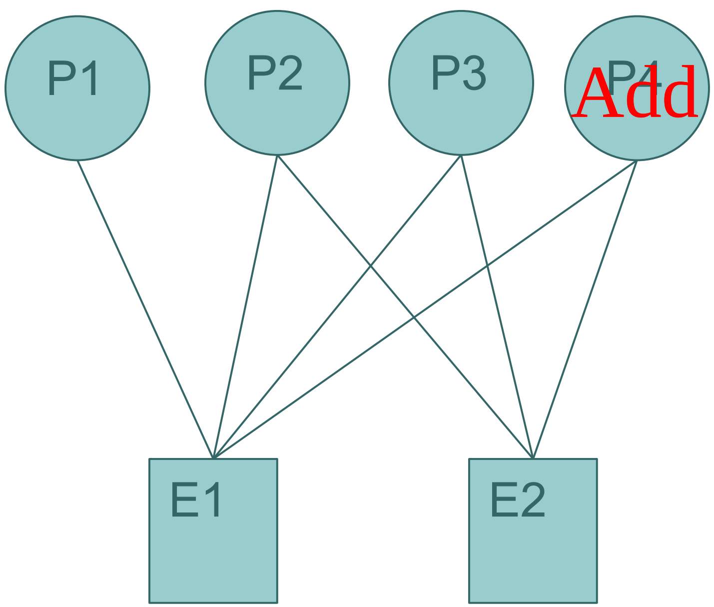


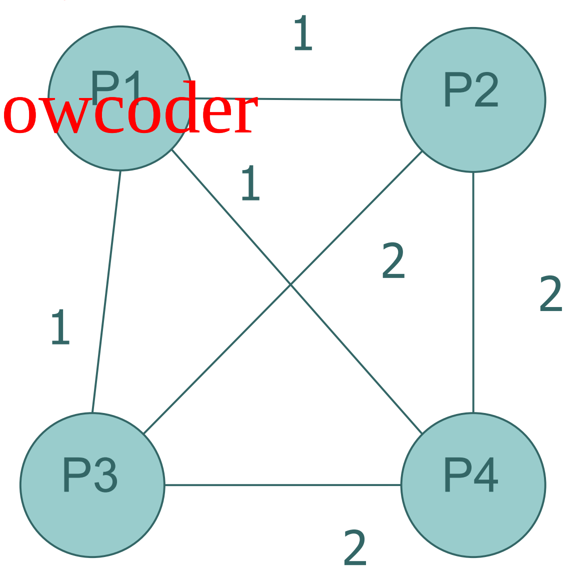
Projection 2: Event-Event



Projection results

- Dense, weighted networks



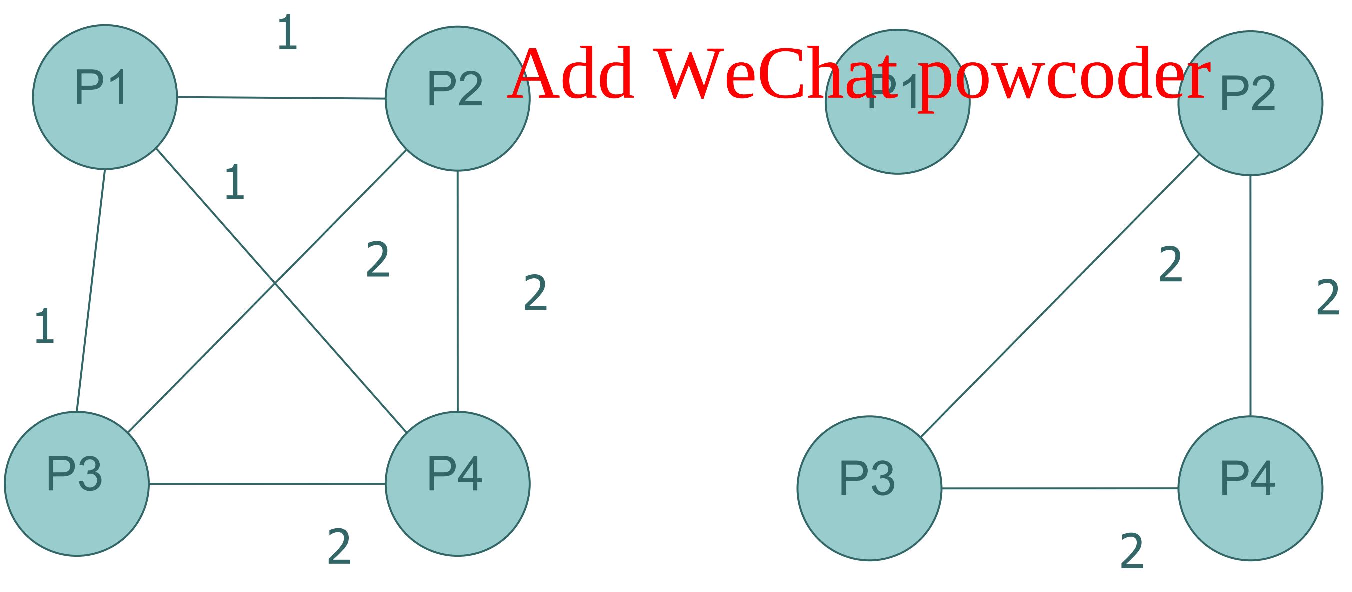


Projections

- Dense
  - because n incident edges = n(n-1)/2 edges in projection
- Weighted
  - so that we can keep track of the number of shared connections

Filtering

- Generally, filtering is necessary to clarify structure



Bipartite networks example