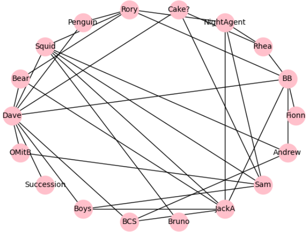
CS4423: Networks

Lecture 7: Permutations and Bipartite Networks

Dr Niall Madden

School of Mathematical and Statistical Sciences, University of Galway

Week 4, Lecture 1 (Wed, 5 Feb 2025)



These slides are by Niall Madden. Elements are based on "A First Course in Network Theory" by Estrada and Knight. Also AC's notes...

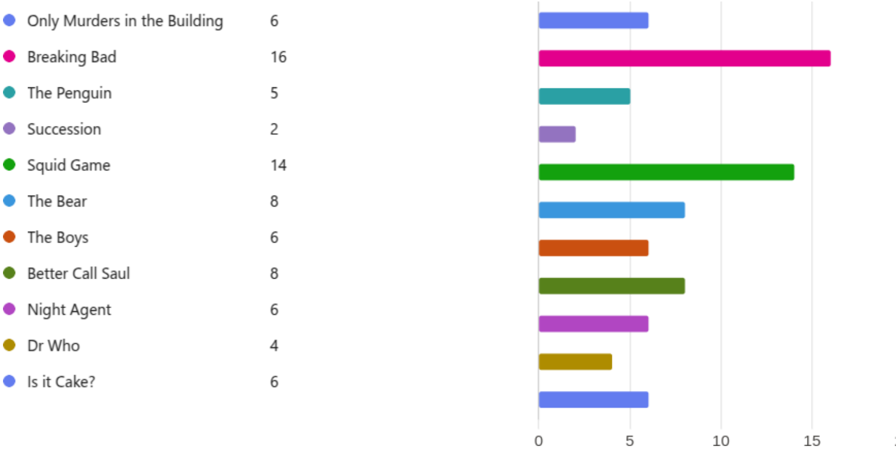
Outline

For further reading, see Section 2.4 of A First Course in Network Theory (Knight).

Slides are at:
https://www.niallmadden.ie/2425-CS4423

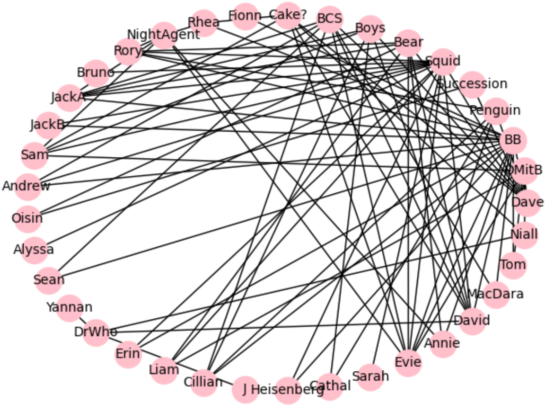
Thanks for completing the survey!

Here is some of the data we collected:



Thanks for completing the survey!

Here is what it looks like as a graph:



Its order is 37, and size is 81; we'll return to this later...

Notation

-

Graph Connectivity

- A graph/network is **connected** if there is a path between every pair of nodes.
- If the graph is *not* connected, we say it is **disconnected**.
- We now know how to check if a graph is connected by looking at powers of its adjacency matrix. However, that is not very practical for large networks.
- However, we can determine if a graph is connected, but just looking at the adjacency matrix, providing we have ordered the nodes properly.

Permutation matrices

We know that the structure of a network is not changes by relabelling its nodes. Sometimes, it is is useful to relabel them in order to expose certain properties, such as connectivity.

Example:

Since we think of the nodes as all being numbered from 1 to n, this is the same as **permuting** the numbers of some subset of the nodes.

Permutation matrices

When working with the adjacency matrix of a graph, such a permutation is expressed in terms of a **permutation matrix**, P: this is a $0-1$ matrix (a.k.a. a "Boolean" or "binary" matrix), where there is a single 1 om every row and column.

If the nodes of a graph G (with adjacency matrix A) are listed as entries in a vector, q, then

- Pq is a permutation of the nodes, and
- $\boxed{PAP^T}$ is the adjacency matrix of the graph with that node permutation applied.

Permutation matrices

Permutation matrices are important when studying graph connectivity because...

FACT!

A graph with adjacency matrix A is **disconnected** if and only if there is a permutation matrix P such that

$$A = P \begin{pmatrix} X & O \\ O^T & Y \end{pmatrix},$$

where O represents the zero matrix with the same number of rows as X and the same number of columns as Y.

Example:

Connected Components

If a network is not connected, then we can divide it into **components** which *are* connected.

The number of connected components is the number of blocks in the permuted adjacency matrix:

Bipartite Graphs (again)

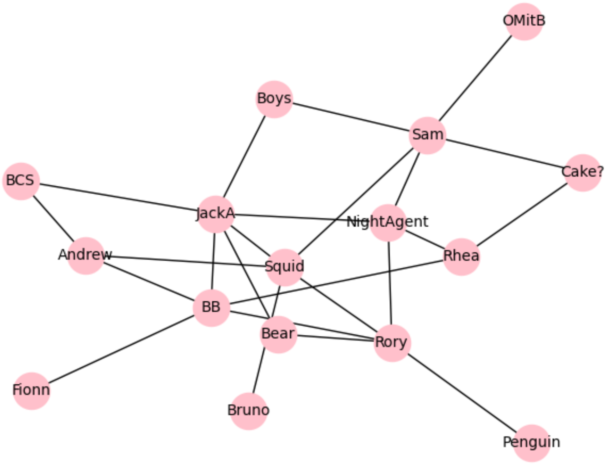
One reason we did the survey is that the resulting data set is a good example of a **bipartite** graph: nodes represent either people or programmes that they watch, with an edge between a person and a programme that they watch.

So the graph must be bipartite.

Such a graph is called an **affiliation** network;

Bipartite Graphs (again)

Here is a **subgraph** of our survey, of order 16 and size 24, based on 7 randomly chosen people:



Bipartite Graphs (again)

This is the adjacency matrix:

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

Bipartite Graphs (again)

That version of the adjacency matrix is not very insightful. But if we order the nodes so all the people are listed first we get the matrix:

$$A = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

Bipartite Graphs (again)

Let's consider $B = A^2$:

$$B = \begin{pmatrix}
1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 3 & 2 & 0 & 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 5 & 1 & 4 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 4 & 1 & 6 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 2 & 2 & 1 & 3 & 5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 2 & 1 & 3 & 1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 3 & 1 & 1 & 3 & 2 & 1 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 4 & 2 & 1 & 3 & 2 & 2 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 3 & 1 & 1 & 5 & 2 & 2 & 2 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 1 & 2 & 2 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1 & 0 & 2 & 1 & 2 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 2 & 1 & 1 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1
\end{pmatrix}$$

Bipartite Graphs (again)

Since we know from Lecture 6 that $(A^k)_{ij}$ is the number of walks of length k between nodes i and j, we can see that in this context:

- For the first 7 rows and columns, b_{ij} for $i \neq j$ is the number programmes in common between person i and j.
- For the last 9 rows and columns, b_{ij} for $i \neq j$ is the number people who watch both programmes i and j.

It can be insightful to consider the submatrices of these blocks...

Bipartite Graphs (again)

Given a bipartite graph, G, whose node set, V, has parts V_1 and V_2, and **projection** of G onto (for example) V_1, is the graph with

- node set V_1
- an edge between a pair of nodes in V_1 if they share a common neighbour in G

In the context of our example, a projection onto V_1 (people) gives us the graph of people who share a common programme.

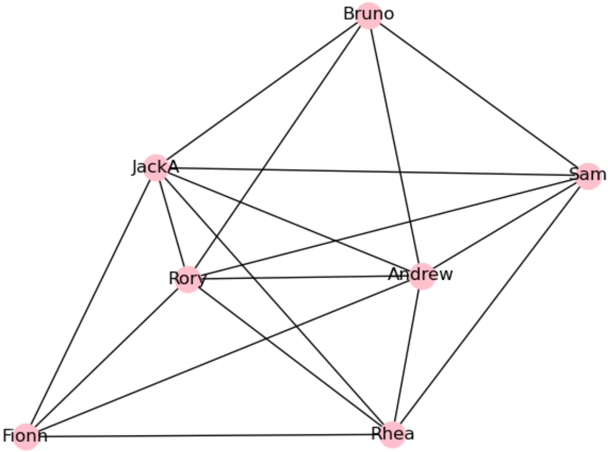
To make such a graph:

- Let A be the adjacency matrix of G.
- Let B be the submatrix of A^2 associated with the nodes in V_1.
- Let C be the (adjacency) matrix with the property

$$c_{ij} = \begin{cases} 1 & b_{ij} > 0 \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

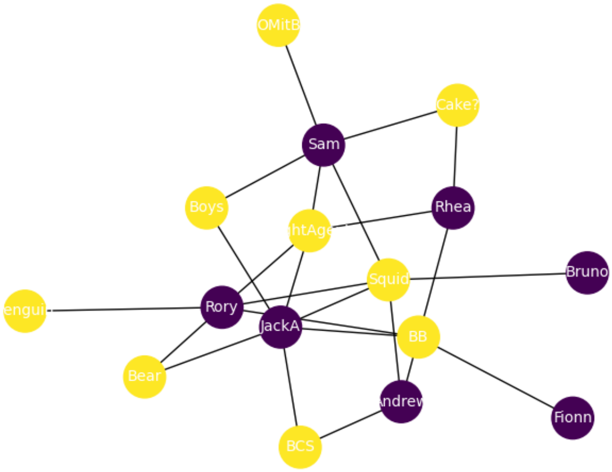
We'll see tomorrow how to do this in networkx.

But this is what it looks like:



Colouring

Our graph would look a bit better if we coloured the nodes, e.g.,



Colouring

For any bipartite graph, we can think of the nodes in the two sets as **coloured** with different colours. For instance, we can think of nodes in X_1 as white nodes and those in X_2 as black nodes.

Vertex colouring

- A **(vertex)-coloring** of a graph G is an assignment of (finitely many) colours to the nodes of G, so that any two nodes which are connected by an edge have **different** colours.
- A graph is called N-**colorable**, if it has a vertex coloring with (at most) N colors.
- The **chromatic number** of a graph G is *smallest* N for which a graph G is N-colourable.

FACT!

Let G be a graph. The following are equivalent:

- G is bipartite;
- G is 2-colorable;
- Each cycle in G has even length.

Later, we'll set how to get `networkx` to compute a colouring for us.

Exercise(s)

1. Let u be a vector with n entries. Let $D = \text{diag}(u)$. That is, $D = (d_{ij})$ is the diagonal matrix with entries

$$d_{ij} = \begin{cases} u_i & i = j \\ 0 & i \neq j. \end{cases}$$

   Verify that $PDP^T = \text{diag}(Pu)$.

2. In all the examples we looked at, we had a symmetric P. Is every permutation matrix symmetric? If so, explain why. If not, give an example.