A (not so) gentle introduction to networks in ecology

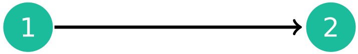
Timothée Poisot

Université du Québec à Rimouski

August 29, 2013

Why should I care about networks?

- A good way to harness complexity (especially of emergent niche patterns)
- A solid mathematical foundation
- Elegant algorithms
- Visualisations look *awesome*
- I'm going to talk about them for, like, two hours...


1
2

What is a network?

A mathematical approach

A *graph* is a **representation** of a **set of objects** where some pairs of objects are **connected** by **links**.

Or more formally, $G = (V, E)$, a *graph* G is an ordered pair of *vertices* V linked together by *edges* E.

Each element of E is a two-element subset of V.

The *order* of a graph is $|V|$, and its *size* is $|E|$.

What is a network?

An example

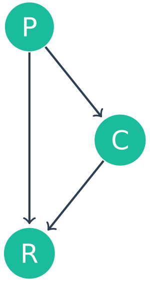
In an omnivory scenario, one top predator P consumes both an intermediate consumer C and a primary producer R. The intermediate consumer also consumes the producer.

This network is specified by

$$G = (\{P, C, R\}, \{\{P, C\}, \{P, R\}, \{C, R\}\})$$

Or for brevity

$$G = (\{P, C, R\}, \{PC, PR, CR\})$$



What is a network?

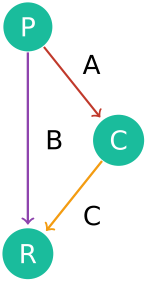
The adjacency matrix

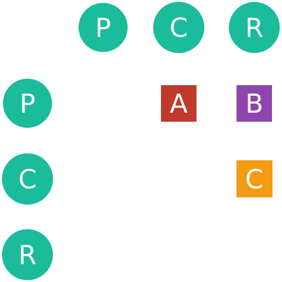
Networks are often represented by their **adjacency matrix**.

The adjacency matrix $\mathbf{A}$ of a graph $G = (V, E)$ has elements A_{ij} with value 1 if there is an edge between the node V_i and the node V_j, and 0 otherwise.

What is a network?

From the graph to the matrix





Exercise: Re-draw this graph (and the matrix) if C is carnivorous.

What is a network?

Edge direction

Edges can be *directed* (arcs, directed edges) or not. An edge between a vertex and itself (cannibalism) is a *self-loop*.

In an **undirected graph**, there are at most $|V|(|V|-1)/2$ edges if there are no *self-loops*.

In a **directed graph**, there are at most $|V|(|V|-1)$ edges if there are no *self-loops*.

Exercise: What is the maximal size of a graph of order n if there are self-loops? What is the *minimal* size of the same graph?

What is a network?

Edge weight

Edges in a network can have a **weight** (for example, the number of contacts between individuals).

The elements of the adjacency matrix $\mathbf{A}$ can be given *continuous* values.

It's possible to work both on the *weighted* and *unweighted* properties of a graph. However, there are many methods that (as of now) can only be applied to **undirected, unweighted** networks.

Number of partners

The number of vertices *receiving* a link from a focal vertex are called its **successors**

The number of vertices *establishing* a link towards a focal vertex are called its **predecessors**

The *total* number of edges connected to a focal vertex is this vertex **degree**

Some remarks on degrees

In a graph with n nodes, the degree k of a node i is $k_i = \sum_{j=1}^{n} A_{ij}$ (*iff* the graph is undirected).

If there are m edges, there are $2m$ *ends* of edges, with $2m = \sum_{i=1}^{n} k_i$. In an undirected graph, this is equal to the sum of degrees.

Thus, $m = \frac{1}{2} \sum_{i=1}^{n} k_i = \frac{1}{2} \sum_i \sum_j A_{ij}$

As the mean degree of nodes is simply $c = \frac{1}{n} \sum k_i$, we can write $c = \frac{2m}{n}$.

Some exercises on degrees

Exercise(s):

Using the same approach, find **the expression of the mean number of predecessors and successors** in a directed graph.

Discuss the relationship between these two values.

Is the mean degree different in a directed/undirected network?

Unipartite and bipartite graphs

In a **bipartite** graph, vertices are divided into two distinct sets, with edges established between two vertices from *different* sets.

This do not happen in **unipartite** graphs.

Exercise: Discuss the application of bipartite and unipartite networks to ecological systems.

In a bipartite graph, the adjacency matrix is (formally) called the **incidence matrix**.

Trees and cycles

Paths

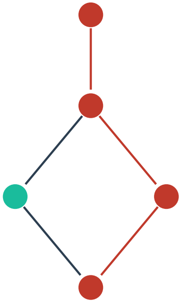
A **path** is a sequence of nodes, so that every consecutive pair of nodes are connected through an edge. You can “walk” along edges from the first to the last node along a path.

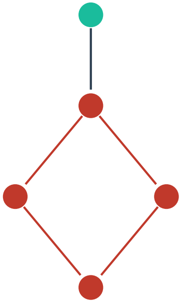
The **length** of a path is the number of *edges* in that path.

If nodes i and j are connected, then *there is a path of length 1 between them*. If they are connected through k, there is *a path of length 2*.

Cycles

A **cycle** is a path from node i to itself, visiting any number of nodes in between.





Reminder: multiplying matrices

- Only *square* matrices can be raised to an exponent (Exercise: What of bipartite rectangular networks?)
- $(\mathbf{AB})_{ij} = \sum_{p=1}^{m} A_{ik} B_{jk}$
- $\mathbf{A}^0 = \mathbf{I}$
- $\mathbf{A}^2 = \mathbf{AA}$
- $(r\mathbf{A})^k = r^k \mathbf{A}^k$
- $\det(\mathbf{A}^k) = \det(\mathbf{A})^k$ because $\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B})$

Number of paths

The number of paths of length 2 between i and j is given by

$$N_{ij}^{(2)} = \sum k = 1^n A_{ik} A_{kj}$$

.

This is generalized for paths of length n to

$$N_{ij}^{(n)} = (\mathbf{A}^n)_{ij}$$

.

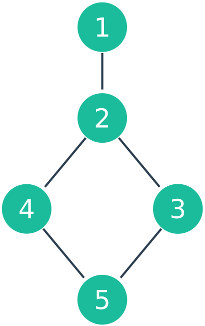
Number of paths

A little bit of programming

Exercise

Write this undirected network as an adjacency matrix, and calculate the number of paths of length 1, 2, . . . , between species 1 and 3, and between species 1 and 5.

Reminder: $\mathbf{A} \times \mathbf{B}$ = `A %*% B`



Number of paths

A little bit of programming

```
matPow = function(A,n)
{
    for(i in c(1:(n-1))) A = A%*%A
    return(A)
}

A = matrix(0, ncol=5, nrow=5)
A[1,2] = A[2,4] = A[2,3] = A[4,5] = A[3,5] = 1
A[2,1] = A[4,2] = A[3,2] = A[5,4] = A[5,3] = 1
for(pathLength in c(0:5)) print(matPow(A, pathLength))
```

Exercise: What happens if the edges are directed (*e.g.* from lower to higher numbers)?

Number of paths

Find the number of cycles

Based on the previous result, we can identify the number of paths of size n from one node to itself, $(\mathbf{A}^n)_{ii}$.

Because for square matrices, $\mathrm{Tr}(\mathbf{A}) = \sum \mathrm{diag}(\mathbf{A})$, the *total number of cycles of length* n is given by $L_n = \mathrm{Tr}(\mathbf{A}^n)$.

Identify the shortest path

A **very crude** way to find the shortest path between i and j is to find the *smallest value of* n for which $(\mathbf{A}^n)_{ij} > 0$.

The shortest path is called the **geodesic** path.

This method in inefficient, and modern software implements much better search strategies.

Other remarkable paths

An **Eulerian path** visits each *edge* once. The graph is Eulerian (unicursal) if this path is a cycle, and semi-Eulerian (traversable) if not.

A **Hamiltonian path** visits each *node* once. A graph with a Hamiltonian path is traceable.

Exercise: Prove that for a complete undirected graph with n nodes, there are $(n-1)!/2$ Hamiltonian paths.

Where is the ecology in all that?

graph The whole community, *i.e.* the populations and their interactions

vertices The composition of the community (species present)

edges The interactions between the populations

Where is the ecology in all that?

Example of “networkable” systems

- Trophic systems
- Plant–pollinators
- Hosts–parasites
- Mutualism
- Social interactions

Any system in which the **same ecological interaction** happens several time in a community can (should) be studied using network theory

Connectance

Connectance is the *proportion of possible interactions realized* (also called the density ρ in graph theory)

The order of an ecological network is usually called S, and its size L

Ecologists often define connectance as $Co = L/S^2$.

Exercise: What do you think of this definition based on results about the maximal order of a graph or size S?

Number of partners

The number of (*e.g.*) preys of a predator is its **generality** (number of successors)

The number of (*e.g.*) predators of a prey is its **vulnerability** (number of predecessors)

Number of partners

```
web = read.table('web.dat')
generality = rowSums(web)
vulnerability = colSums(web)
degree = generality + vulnerability
```

Null models

The ecological use

Even just *by chance*, networks will have statistical properties. The truth is in the residuals.

Null models allow to disentangle the random and non-random properties of networks.

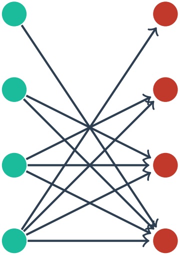
Example of a question: is a given network more or less nested than expected by chance?

Null models

WTF is nestedness?

In a *nested* network, a species with a degree d_i interacts with a subset of the species with a degree d_j so that $d_j > d_i$.

The more this is true, the more a network is nested.



```
library(vegan)
web = read.table('matrix.txt')
nestednodf(web)
```

Null models

The basic approach

We want to simulate *pseudo-random* networks that reproduce *some* features of the empirical (observed) network.

For any empirically measured property X, we will have a distribution of $\mathcal{X}$.

We might want to test that $\sum(X - \mathcal{X})/n = 0$, *i.e.* the observed property is not significantly different from the random expectation.

Null models

Controlling for mean degree

This is formally called a *Erdös–Renyi graph*. Each pair of nodes has a probability ρ of being connected.

Exercise: write a function in R filling a matrix with 0 and 1 at random.

Null models

Controlling for node degree

We want each node to receive and establish interactions (almost) as in the original network.

$$P(i \rightarrow j) = \frac{1}{2}\left(\frac{g_i}{d_i} + \frac{v_j}{d_j}\right)$$

Exercise: Using operations on matrices, write a function to generate a random network in R, conserving either the connectance, or the degrees.

The niche model of food webs