# Spectral Methods and Algorithms: applications in neuroscience

Jonathan J. Crofts

Department of Physics & Mathematics
Nottingham Trent University

August 5, 2011

jonathan.crofts@ntu.ac.uk

# Outline

# Background

**Joel E. Cohen**

**Mathematics is biology's next microscope, only better**

# Background

**Joel E. Cohen**

**Mathematics is biology's next microscope, only better**

Quantative/computational work in biology may be **data driven** or may arise through **modelling**

**Model**

**Quantative, simplified description of a natural system**

Useful for

- **testing/comparing hypotheses**
- **making predictions**

This talk will focus on **networks**: **extracting useful information** and **modelling**

# Typical Tasks

**Data Driven:**

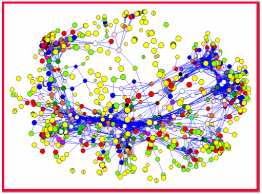
- find well-connected **clusters**
- find specific connectivity **substructures**
- find '**important**' nodes or links
- **compare** the properties of one network with another

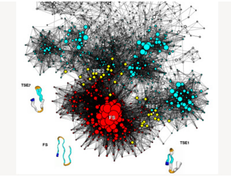
**Modelling Arguments:**

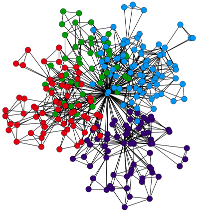
- summarize a network in terms of a few **parameters**
- **explain** how the connectivity has arisen
- discover **missing** or **spurious** links
- make predictions concerning **future growth** of the network

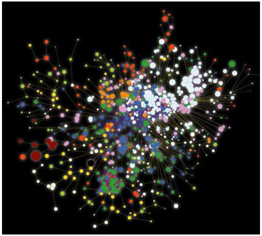
# Network Science: connections are important

**Complex networks are the structural skeletons of complex systems**



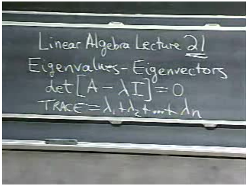






# Outline

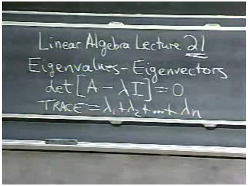
# Graph Spectra




**Spectral methods:**

- **matrix representation** of the network
- study the spectra of the resulting matrix, i.e., **eigenvalues** and **eigenvectors**

# Graph Spectra



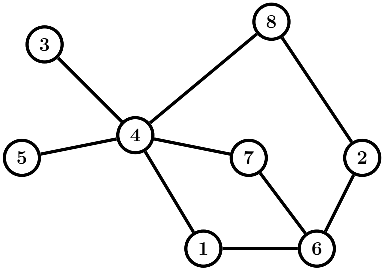

**Spectral methods:**

- **matrix representation** of the network
- study the spectra of the resulting matrix, i.e., **eigenvalues** and **eigenvectors**

**Importantly:**

- this allows us to compute **graph invariants** using basic linear algebra; and
- to implement **data-mining tools** to study networks, i.e., determine **patterns** and **features**

# Graph Spectra: adjacency matrix

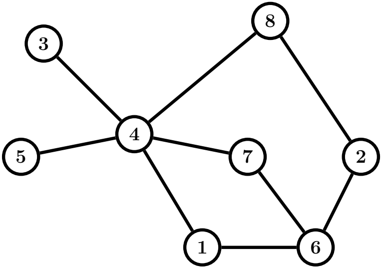



$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$A = \begin{cases} 1, & i \sim j \\ 0, & \text{otherwise} \end{cases}$$

here $\sim$ denotes that vertices $i$ and $j$ are **adjacent**

# Other Possibilities Include . . .




$$L = \begin{pmatrix} 2 & 0 & 0 & -1 & 0 & -1 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 5 & -1 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 3 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 2 & 0 \\ 0 & -1 & 0 & -1 & 0 & 0 & 0 & 2 \end{pmatrix}$$

**Graph Laplacian**

$$L = \begin{cases} d_i, & i = j \\ -a_{ij}, & \text{otherwise} \end{cases}$$

here $d_i$ denotes the **degree** of node $i$

# Other Possibilities Include . . .

**Normalised Laplacian:**

$$\mathcal{L} = \begin{cases} 1, & i = j \\ -\frac{a_{ij}}{\sqrt{d_i d_j}}, & \text{otherwise} \end{cases}$$

**Signless Laplacian:**

$$Q = \begin{cases} d_i, & i = j \\ a_{ij}, & \text{otherwise} \end{cases}$$

- less well studied
- can determine **bipartite structures** **(Kirkland and Paul 2011)**

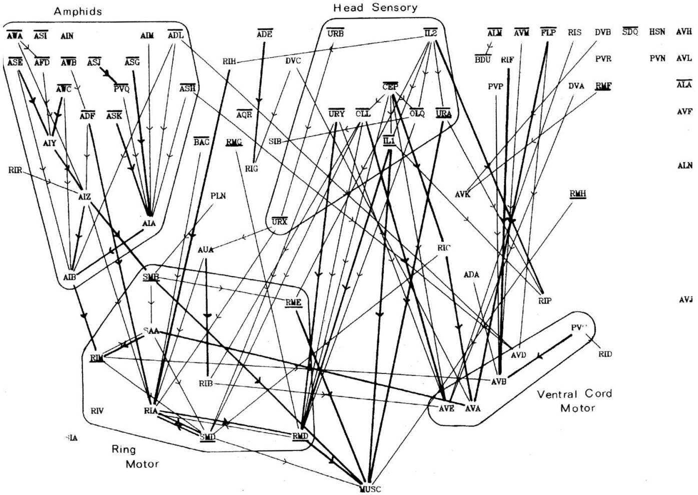
**Line Graph:**

- used to detect **community structure** **(Evans & Lambiotte 2010)**

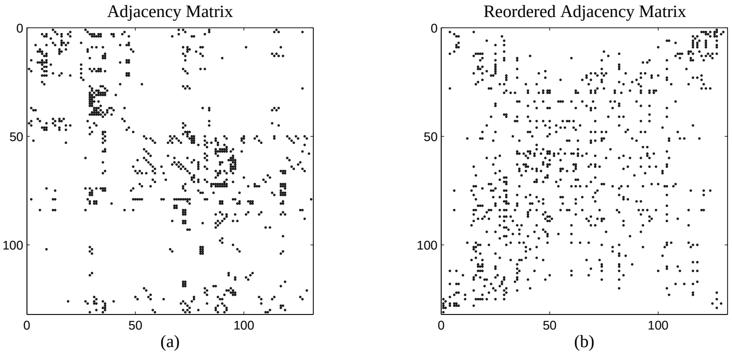
# An Example: neuronal network of *C. elegans*

- **C. elegans** are tiny (1mm long), transparent, round worms
  - **Model organism in biology**
  - **Connectome consists of some 302 neurons linked by over 7000 synaptic connections**

# Motivation

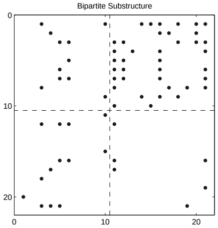

Amphids
Head Sensory
AWA ASI AIN AIM ADL
ADE URB IL2
ALM AVM FLP RIS DVB SDQ HSN AVH
ASE AFD AWB ASJ ASG
RIH DVC
BDU RIF PVR PVN AVL
AWC PVQ ASH CEP
PVP DVA RMF ALA
ADF ASK AQR URY CLL OLQ URA
AVF
AIY BAC RMG SIB IL1
RIR RIG ALN
AIZ PLN AVK RMH
AIA URX
AUA RIC
AIB SMB ADA
RME RIP AVJ
SAA
RIM AVD RID
RIB AVB
RIV RIA AVE AVA
Ventral Cord Motor
SMD RMD
SIA
Ring Motor
MUSC

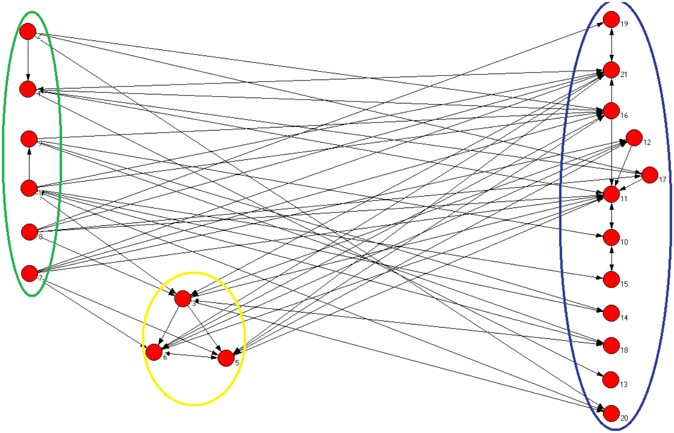
# Local C. elegans Neuronal Network




- Reordering the frontal neurons of C. elegans using eigenvectors of the **signless Laplacian Q** reveals **bipartite substructures**

# C. elegans Example Ctd.



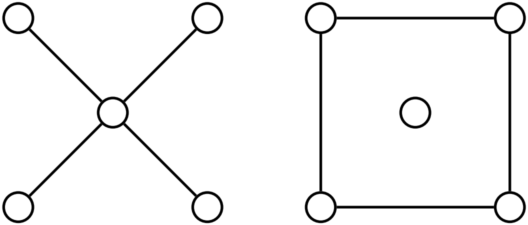
# Properties of the Different Graph Spectra

**Important:** **cospectral graphs are not necessarily isomorphic**

# Properties of the Different Graph Spectra

**Important:** **cospectral graphs are not necessarily isomorphic**

**Example:**



$$\chi_G = \lambda^5 - 4\lambda^3$$

# Properties of the Different Graph Spectra

Properties the spectrum (eigenvalues) **can** and **cannot** distinguish:

| Matrix | # edges | bipartite | # components | # bipartite components |
|---|---|---|---|---|
| $A$ | Yes | Yes | No | No |
| $L$ | Yes | No | Yes | No |
| $\mathcal{L}$ | No | Yes | Yes | Yes |
| $Q$ | Yes | No | No | Yes |

# Properties of the Different Graph Spectra

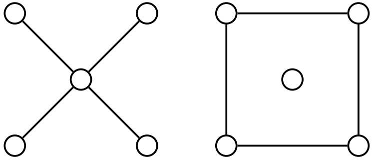
- Which graphs are determined by their spectrum? (**Van Dam & Haemers (2003)**)
  - **For almost all graphs this is an open question**
  - **Numerical simulations suggest that 'almost all graphs are'**

# Properties of the Different Graph Spectra

- Which graphs are determined by their spectrum? (**Van Dam & Haemers (2003)**)
  - **For almost all graphs this is an open question**
  - **Numerical simulations suggest that 'almost all graphs are'**

- A combination of eigenvalues and eigenvectors does the trick, e.g., **subgraph centrality**

$$C_{\mathcal{S}}(i) = \sum_{k=1}^{n} e^{\lambda_k} \mathbf{x}_i^{[k]^2}$$

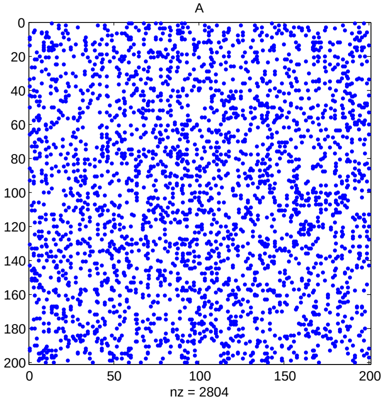


$$C_{\mathcal{S}} = \begin{pmatrix} 1.6905 \\ 1.6905 \\ 3.7622 \\ 1.6905 \\ 1.6905 \end{pmatrix} \quad \& \quad \begin{pmatrix} 2.3811 \\ 2.3811 \\ 1.0000 \\ 2.3811 \\ 2.3811 \end{pmatrix}$$

# Outline

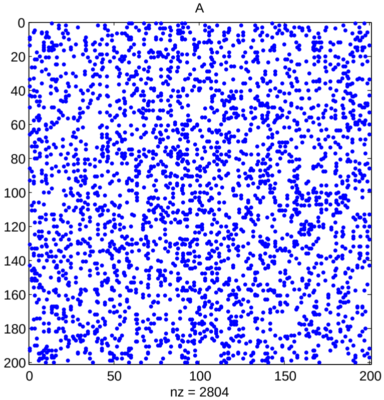
# An Example: short-range structure

- **The network reordering problem**:



# An Example: short-range structure

- **The network reordering problem**:



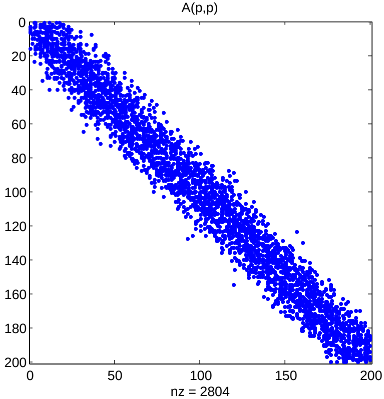

- Solve

$$\min \sum_{\{p \in \mathcal{P}\}} (p_i - p_j)^2 a_{ij}$$

$\mathcal{P}$ denotes the set of permutations of the integers $\{1, \ldots, n\}$

# An Example: short-range structure

**The network reordering problem**:




- Solve

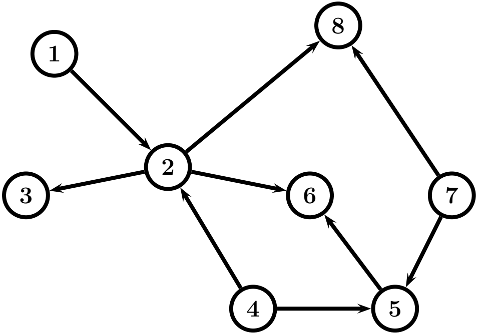
$$\min \sum_{\{p \in \mathcal{P}\}} (p_i - p_j)^2 a_{ij}$$

$\mathcal{P}$ denotes the set of permutations of the integers $\{1, \ldots, n\}$

- An **approximate solution** is given by the first, non-zero eigen- -vector of $L = D - A$

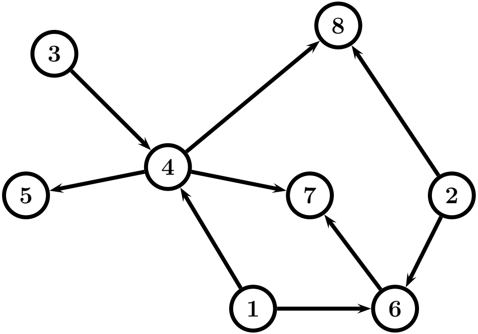
# Outline

# Directed Hierarchies




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

# Directed Hierarchies




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

- In general it is possible to find such an ordering iff we have a **DAG**

# Out Minus In Degree

## One-Sum Optimisation Problem

$$\min_{p \in \mathcal{P}} \sum_{i,j} (p_i - p_j) a_{ij}$$

## Proof

$$\begin{aligned}\sum_{i,j} (p_j - p_i) a_{ij} &= \sum_i p_i \cdot \mathrm{deg}_i^{\mathrm{out}} - \sum_j p_j \cdot \mathrm{deg}_j^{\mathrm{in}} \\ &= \sum_i p_i \cdot (\mathrm{deg}_i^{\mathrm{out}} - \mathrm{deg}_i^{\mathrm{in}})\end{aligned}$$

# Out Minus In Degree

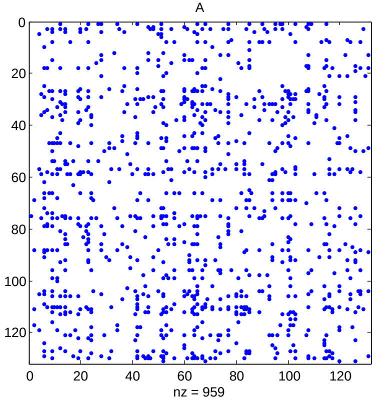
## One-Sum Optimisation Problem

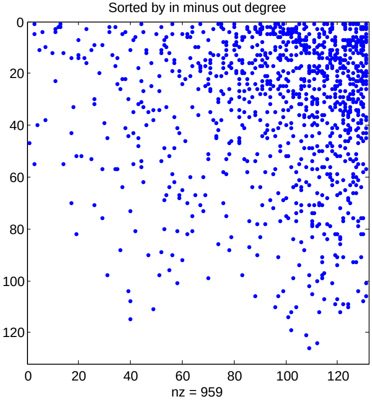
$$\min_{p \in \mathcal{P}} \sum_{i,j} (p_i - p_j) a_{ij}$$

## Proof

$$\begin{aligned} \sum_{i,j} (p_j - p_i) a_{ij} &= \sum_i p_i \cdot \mathrm{deg}_i^{\mathrm{out}} - \sum_j p_j \cdot \mathrm{deg}_j^{\mathrm{in}} \\ &= \sum_i p_i \cdot (\mathrm{deg}_i^{\mathrm{out}} - \mathrm{deg}_i^{\mathrm{in}}) \end{aligned}$$

# Synthetic Network Example

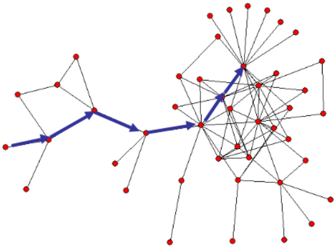


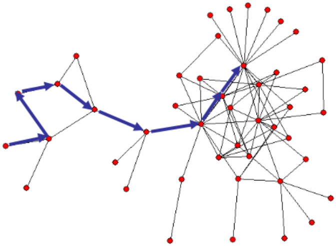

- Hierarchical structure is uncovered in RHS using out-in degree

# Outline

# Paths Vs Walks



**Path**



**Walk**

**Shortest Path**

# Counting Walks

From the following identity

$$(A^k)_{ij} = \sum_{i_1=1}^{n} \sum_{i_2=1}^{n} \cdots \sum_{i_{k-1}=1}^{n} a_{i,i_1} a_{i_1,i_2} \cdots a_{i_{k-1},j},$$

we see that $(A^k)_{ij}$ **counts the number of different walks** of length $k$ between nodes $i$ and $j$; moreover, the quantity

$$F_{ij} = (c_0 I + c_1 A + c_2 A^2 + c_3 A^3 \cdots)_{ij}$$

with the $c_k$ constant, gives a **measure of the total number of walks** between nodes $i$ and $j$

# Counting Walks

From the following identity

$$(A^k)_{ij} = \sum_{i_1=1}^{n} \sum_{i_2=1}^{n} \cdots \sum_{i_{k-1}=1}^{n} a_{i,i_1} a_{i_1,i_2} \cdots a_{i_{k-1},j},$$

we see that $(A^k)_{ij}$ **counts the number of different walks** of length $k$ between nodes $i$ and $j$; moreover, the quantity

$$F_{ij} = (c_0 I + c_1 A + c_2 A^2 + c_3 A^3 \cdots)_{ij}$$

with the $c_k$ constant, gives a **measure of the total number of walks** between nodes $i$ and $j$

# Counting Walks: matrix functions

- For **suitable choices** of $\{c_k\}_{k\geq 0}$ the series overpage converges
  - for example, $c_k = \{\delta^k\}_{k\geq 0}$, gives the **matrix resolvent** $(\mathbf{I} - \delta\mathbf{A})^{-1}$
  - different choices of $c_k$ allow for different scalings

# Counting Walks: matrix functions

- For **suitable choices** of $\{c_k\}_{k\geq 0}$ the series overpage converges
  - for example, $c_k = \{\delta^k\}_{k\geq 0}$, gives the **matrix resolvent** $(\mathbf{I} - \delta\mathbf{A})^{-1}$
  - different choices of $c_k$ allow for different scalings
- In particular, if $f$ is defined on the spectrum of $A$, we can define

$$F = f(A) = Pf(D)P^{-1}$$

**(here $\mathbf{A} = \mathbf{PDP^{-1}}$ is the eigendecomposition)**

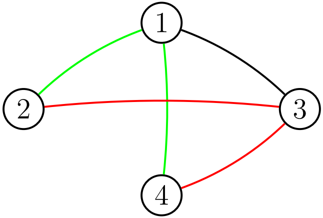
**Proof** Let

$$f(A) = c_0 I + \sum_{k=1}^{\infty} c_k A^k$$

substituting $A = PDP^{-1}$ into the above

$$f(A) = f(PDP^{-1}) = c_0 I + \sum_{k=1}^{\infty} c_k (PDP^{-1})^k = P\left(c_0 I + \sum_{k=1}^{\infty} c_k D^k\right) P^{-1}$$

# Example: communicability




$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \; A^2 = \begin{pmatrix} 3 & 1 & 2 & 1 \\ 1 & 2 & 1 & 2 \\ 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix}, \cdots$$

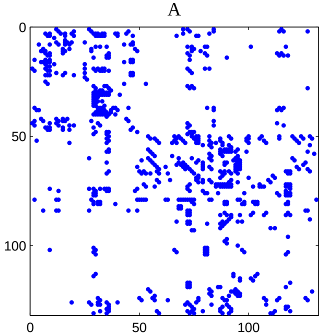
**Communicability** between distinct nodes $i$ and $j$

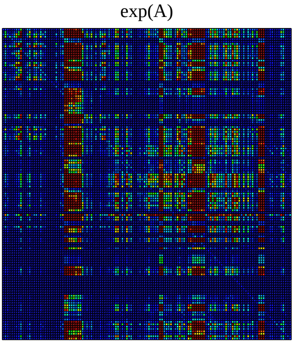
$$\left( I + A + A^2/2! + A^3/3! + \cdots \right)_{ij}$$

that is $\left(e^A\right)_{ij}$

**Spectral form:** $\left(\sum_{k=1}^{n} e^{\lambda_k} \mathbf{x}^{[k]}(i)\mathbf{x}^{[k]}(j)\right)_{ij}$

# Example: communicability




**Communicability applied to the frontal network of C. elegans**

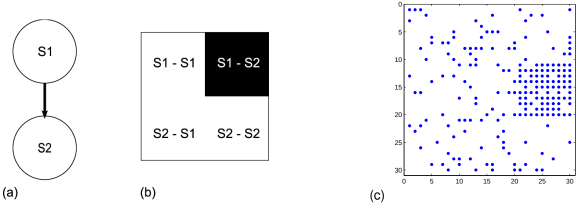
# Outline

# Approximate Directed Bipartite Community

Crofts, Estrada, Higham, Taylor **Elec. Trans. Numer. Anal (2010)**

Distinct subsets of nodes $S_1$ and $S_2$ such that

- $S_1$ has few internal links
- $S_2$ has few internal links
- there are many $S_1 \to S_2$ links
- few other links involve $S_1$ or $S_2$



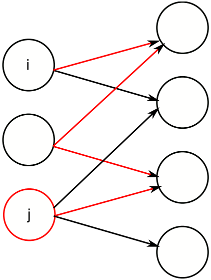
**An alternating walk of length $k$ from node $i_1$ to node $i_{k+1}$**

is a list of nodes

$$i_1, i_2, i_3, \ldots, i_{k+1}$$

such that $a_{i_s, i_{s+1}} \neq 0$ for $s$ odd, and $a_{i_{s+1}, i_s} \neq 0$ for $s$ even

Loosely, an alternating walk is a traversal that successively follows links in the forward and reverse directions



# This Motivates . . .

$$f(A) = I - A + \frac{AA^T}{2!} - \frac{AA^TA}{3!} + \frac{AA^TAA^T}{4!} - \cdots$$

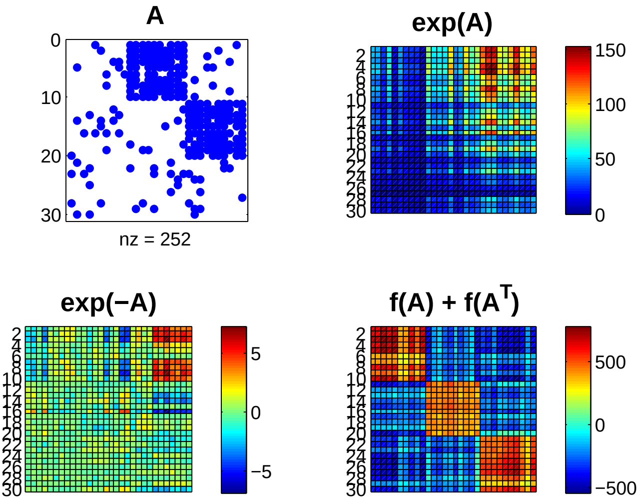
**Overall idea:** $f(A) + f(A^T)$ has

- positive values representing inter-community $S_1 \leftrightarrow S_1$ and $S_2 \leftrightarrow S_2$ relationships, and
- negative values representing extra-community $S_1 \leftrightarrow S_2$ relationships

Also, $f(A) + f(A^T)$ is a symmetric matrix, so amenable to standard clustering techniques

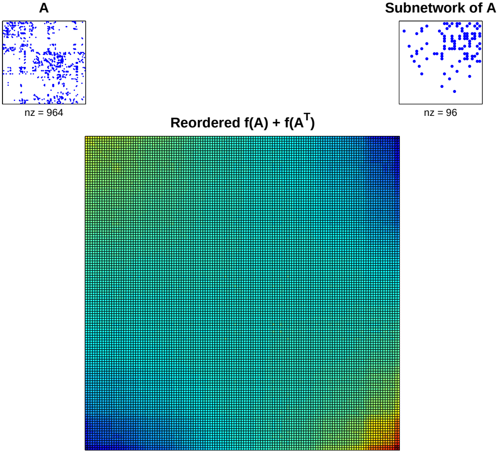
**Note:** $f(A)$ defined above is **not** a matrix function

# Synthetic Example



# C. elegans Neural Data

Automates the computations of Durbin, (PhD thesis, Cambridge, 1987)



# Outline

# Communicability for a Weighted Network

## W is symmetric with non-negative real weights

**Let**

$$d_i = \sum_{k=1}^{n} w_{ik} \quad \& \quad D := \mathrm{diag}\left(d_i\right)$$

**Normalisation:**

$$W \mapsto D^{-1/2} W D^{-1/2}$$

**for at least 2 reasons**

- avoid overflow
- to stifle promiscuous nodes

# Communicability for a Weighted Network

**W is symmetric with non-negative real weights**

**Let**

$$d_i = \sum_{k=1}^{n} w_{ik} \quad \& \quad D := \mathrm{diag}\,(d_i)$$

**Normalisation:**

$$W \mapsto D^{-1/2} W D^{-1/2}$$

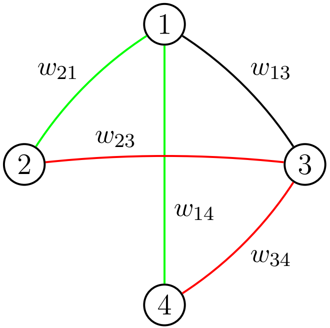
**for at least 2 reasons**

- **avoid overflow**
- **to stifle promiscuous nodes**

**Communicability measure:**

$$\exp\left(D^{-1/2} W D^{-1/2}\right)$$

# But What Does it Actually Mean?




$$W = \begin{pmatrix} 0 & w_{12} & w_{13} & w_{14} \\ w_{21} & 0 & w_{23} & w_{24} \\ w_{31} & w_{32} & 0 & w_{34} \\ w_{41} & w_{42} & w_{43} & 0 \end{pmatrix}$$

The $k$th powers of $W$ provide a **measure of the total strength** contained within walks of length $k$ between nodes $i$ and $j$:

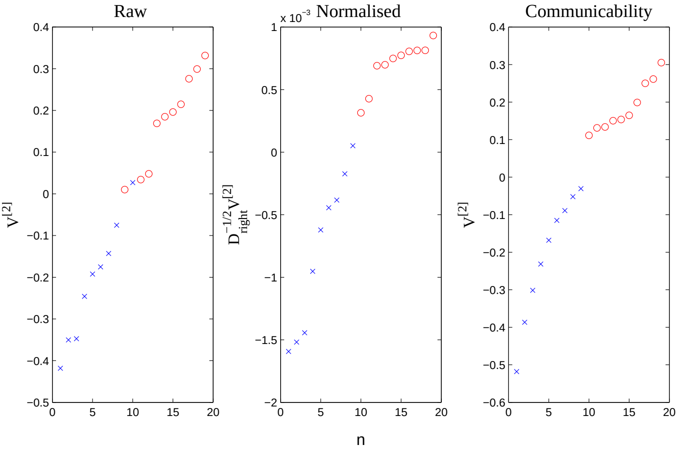
$$(W^2)_{ij} = \sum_k w_{ik} w_{kj}, \, (W^3)_{ij} = \cdots$$

# Example: anatomical connectivity data

- **9** subjects - at least 6 months following first, left-hemisphere, subcortical stroke; and **10 (18)** age matched controls
- **Diffusion Tensor Imaging** computes all connections between all voxels
- Connectivity network based on the **Harvard-Oxford cortical and subcortical structural atlas**: **48** cortical regions and **8** subcortical regions

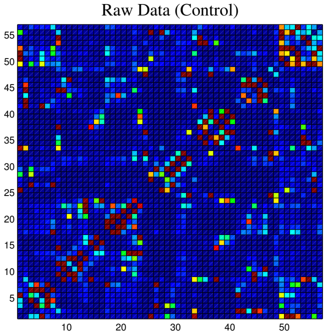
http://wwww.fmrib.ox.ac.uk/fsl/fslview/atlas-descriptions.html

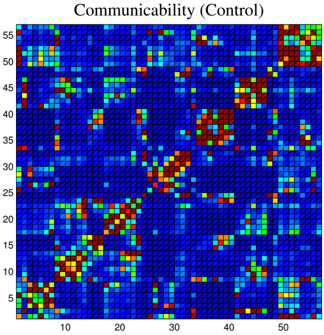
# Unsupervised Clustering of Patients




**Crofts & Higham, Roy. Soc. Interface (2009)**

# Communicability Adds Value to the Raw Data



# What Are we Actually Detecting?

**Question**

Exactly what enables us to differentiate between strokes & controls?

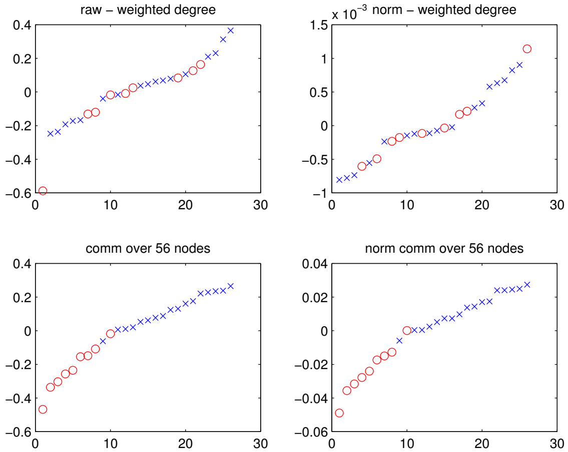
# What Are we Actually Detecting?

**Question**

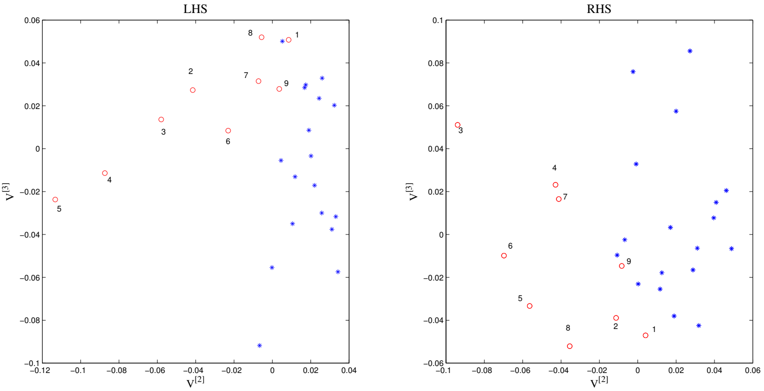
Exactly what enables us to differentiate between strokes & controls?

- If it is merely the fact that many connections have been destroyed close to the infarcted region this is not very interesting – **an MRI scan can tell us this with no further analysis needed!**
  - Why not rerun the tractographies including only those connections within the **non-stroke hemisphere**?
  - If we still distinguish between the two classes, this will be more relevant from a bio point of view

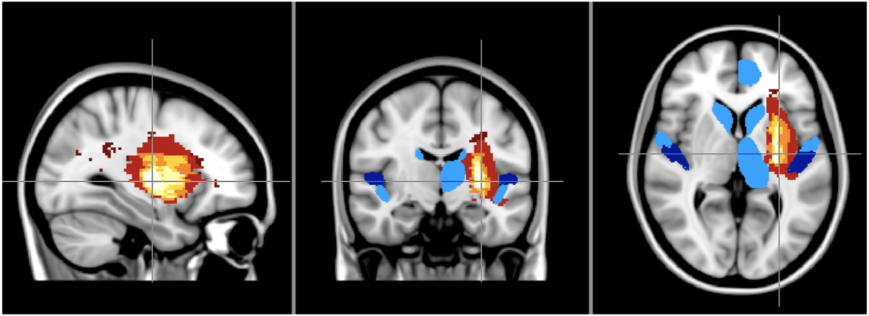
# RHS Hemisphere Sorted by Brain Region



# Left (Stroke Side) Vs Right

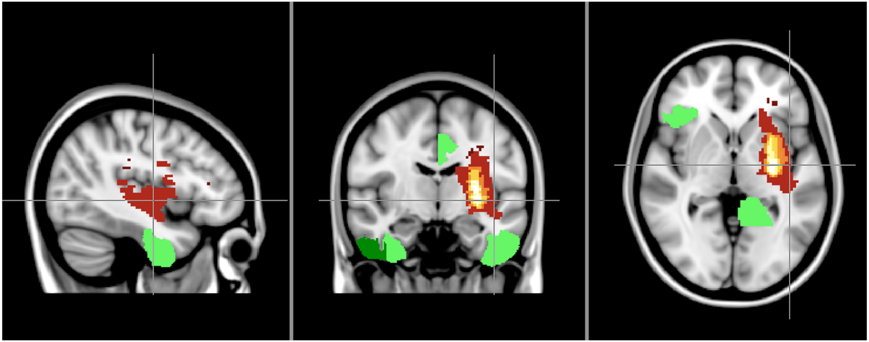


# Overlay of Stroke Overlap Volume



**Highlighted regions are those where patients showed reduced communicability relative to controls**

# Crofts *et al.* NeuroImage (2011)



**And increased communicability**

# GSVD

- **Generalised singular value decomposition**

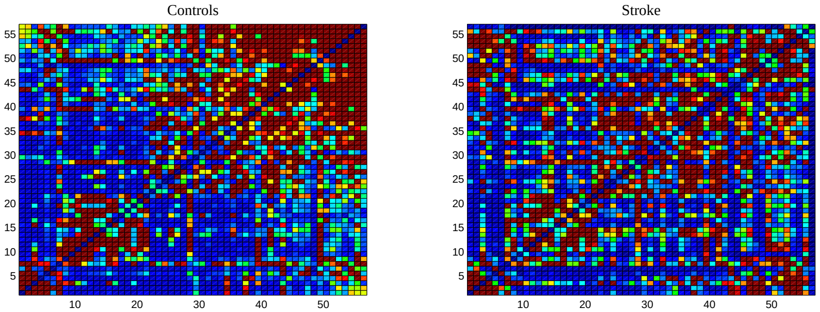
$$A = UCX^{-1} \quad \text{and} \quad B = VSX^{-1}$$

  of a pair of matrices can be used to determine clusters that are 'good' in one network and 'poor' in the other - and vice versa **[Xiao et al. (2011)]**

- Another way to understand this is to note that, in the case where **A and B are invertible** the GSVD is closely related to the SVD of $AB^{-1}$ and $BA^{-1}$, hence

$$AB^{-1} = UCS^{-1}V^T \quad \text{and} \quad BA^{-1} = VSC^{-1}U^T$$

# GSVD Example: anatomical connectivity data




**GSVD** finds a group of brain regions that are much better connected in controls than strokes

# Outline

# The Take Home Message

- **Spectral methods provide a useful tool for determining different types of network architecture**
  - **scalable**
  - **most work has focused on the adjacency matrix**

# The Take Home Message

- **Spectral methods provide a useful tool for determining different types of network architecture**
  - **scalable**
  - **most work has focused on the adjacency matrix**

- **Walk based measures are attractive since:**
  - **combinatorics are described in terms of basic linear algebra**
  - **information does not necessarily flow along geodesics**
  - **walks are more tolerent to errors**

# The Take Home Message

- **Spectral methods provide a useful tool for determining different types of network architecture**
  - **scalable**
  - **most work has focused on the adjacency matrix**
- **Walk based measures are attractive since:**
  - **combinatorics are described in terms of basic linear algebra**
  - **information does not necessarily flow along geodesics**
  - **walks are more tolerent to errors**
- GSVD and extensions:
  - **tensor decompositions (plus multiple tasks/modalities, time-dependent networks)**
  - **NNMF (Lee et al (2010)), ICA (Smith et al (2005))**

# Thank you!

**Colleagues at Strathclyde: Des Higham, Ernesto Estrada & Alan Taylor**

**Colleagues from Oxford: Heidi Johanson-Berg & Tim Behrens**
The stroke data was supplied by **Rose Bosnell**

This work was supported by the **Medical Research Council** under project no. MRC G0601353