AoN Brainhack
Warsaw 2017

Aspects of Neuroscience Conference Satellite Event

18-19th November 2017 @ University of Warsaw

Functional connectivity: can we find a common ground?

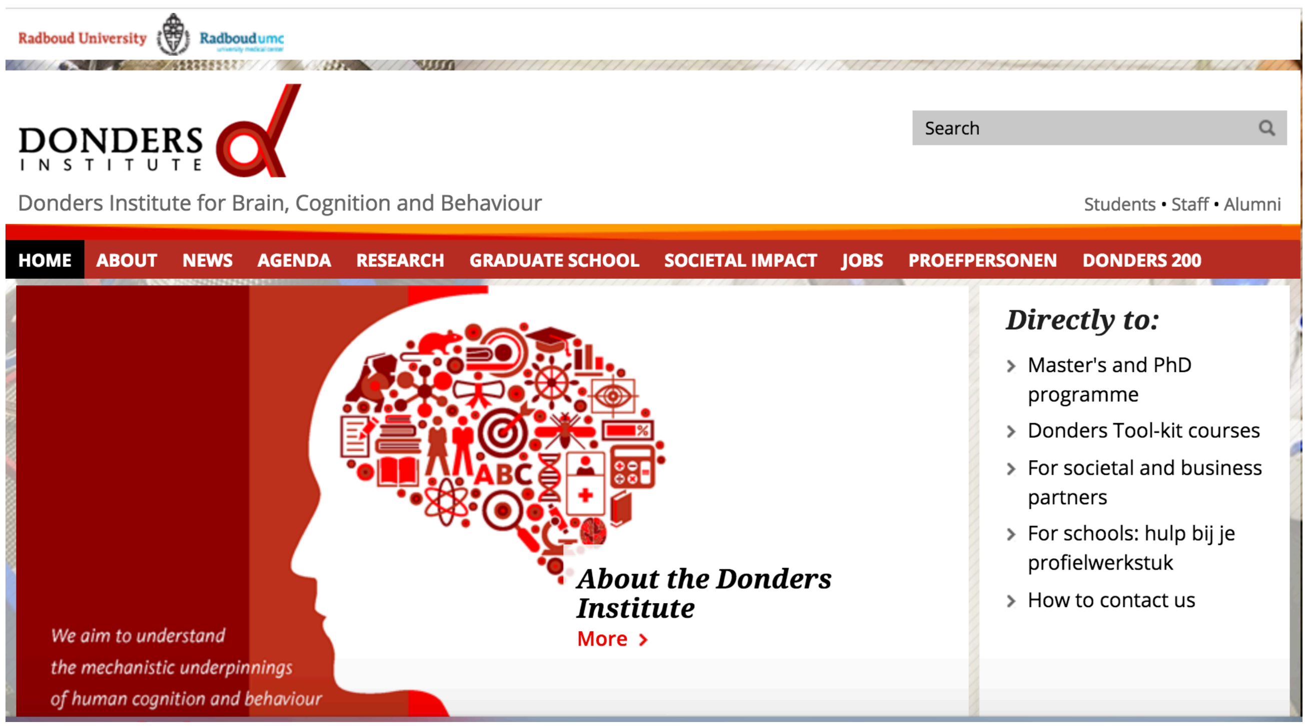
Natalia Bielczyk

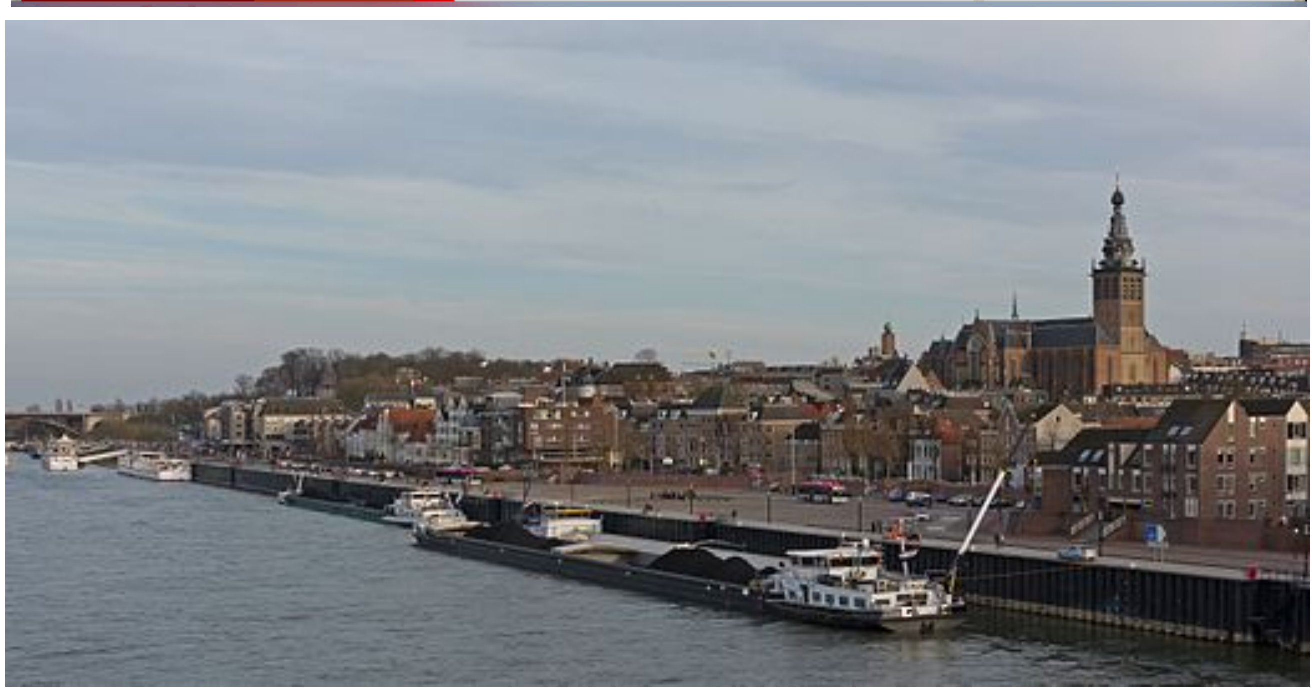
Warsaw, November 18th 2017

Who am I?

working on causal inference and signal detection in neuroimaging (especially fMRI)

will give a talk at the Aspects of Neuroscience Conference, Saturday 25th November, 12:45pm



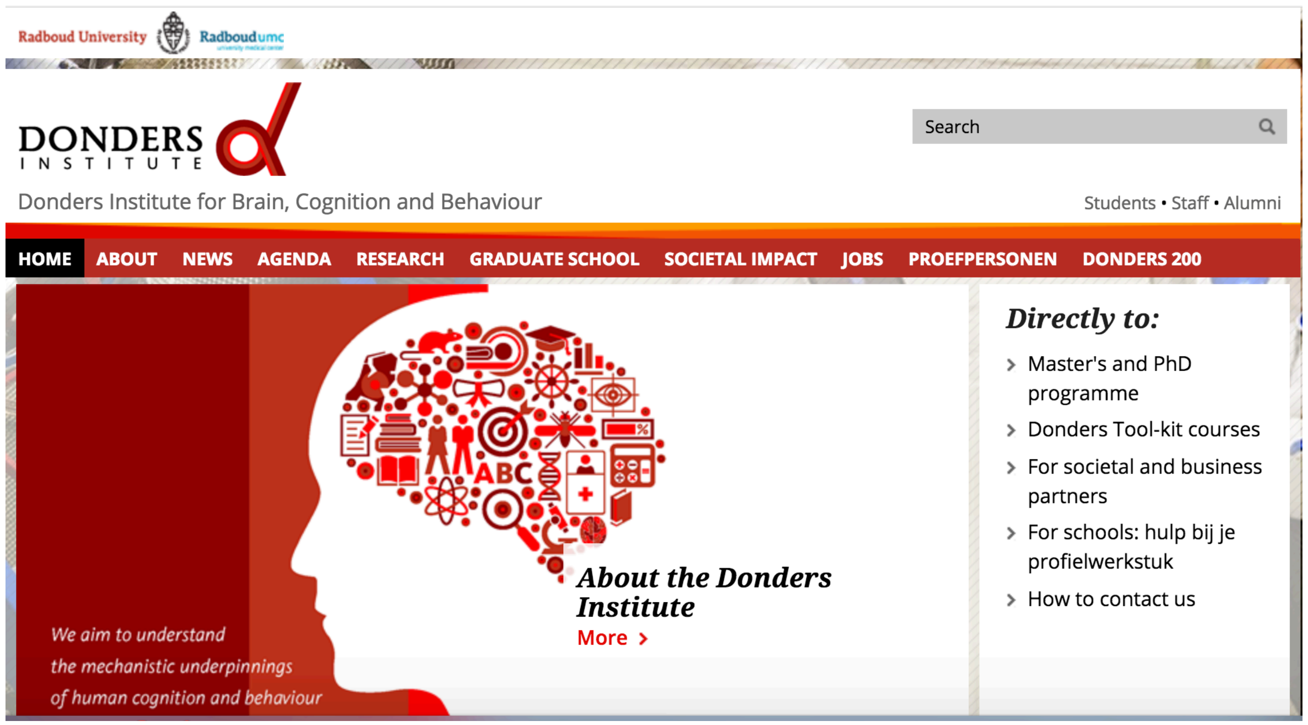


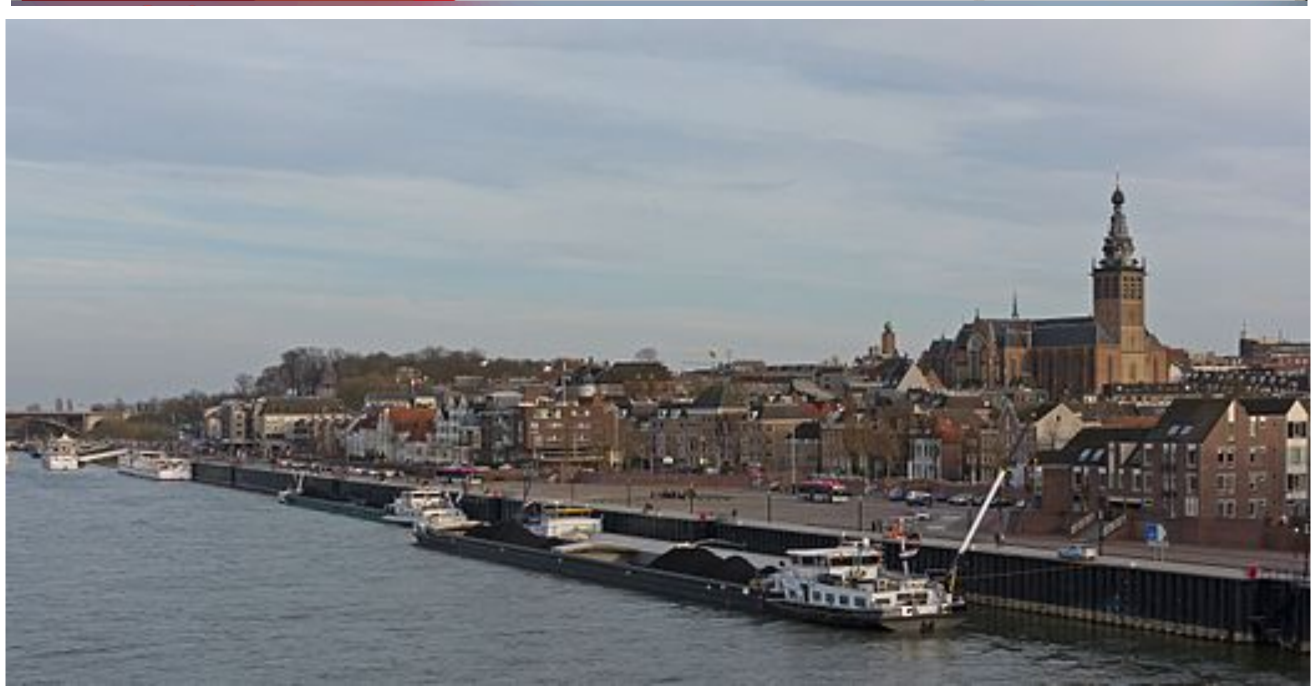
Nijmegen, the Netherlands

Who am I?

working on causal inference and signal detection in neuroimaging (especially fMRI)

will give a talk at the Aspects of Neuroscience Conference, Saturday 25th November, 12:45pm





Nijmegen, the Netherlands

And if you would like
to hang out together again one day…

@nbielczyk_neuro

www.linkedin.com/in/nataliabielczyk/

Must-knows before we start

Acknowledgments

The team who prepared the pipeline

Making-off

ONLINE BRAIN
INTENSIVE

Natalia Bielczyk
Donders Institute
Nijmegen, the Netherlands

Małgorzata Wierzba
Nencki Institute of Experimental Biology
Warsaw, Poland

Florencia Garro
OTTAA project
Cordoba, Argentina

Testing

Daniel Borek
Ghent University
Ghent, Belgium

Tools available during the project

- the project leader / your partner / our slack channel
- codes / datasets at the GitHub repository: **https://github.com/cryptofan/FunctionalConnectivity_AoNBrainhackWarsaw**
- an Overleaf project with the detailed description of methods and datasets: **https://www.overleaf.com/11096911dptdwdbwrzdd**

Tools available during the project

- the project leader / your partner / our slack channel
- codes / datasets at the GitHub repository: **https://github.com/cryptofan/FunctionalConnectivity_AoNBrainhackWarsaw**
- an Overleaf project with the detailed description of methods and datasets: **https://www.overleaf.com/11096911dptdwdbwrzdd**

If you are willing to contribute to the proceedings also *after* the event, please add your name and affiliation to the Overleaf document

Plan of the presentation

[0] what is functional connectivity?
[1] correlation vs causation
[2] Pearson’s correlation vs partial correlation
[3] infomation-theoretical measures of functional connectivity
[4] time domain vs frequency domain
[5] static vs dynamic functional connectivity
[6] hierarchical models of functional connectivity

[7]* higher order associations in the networks (multiple body interactions, graph theory)

Plan of the presentation

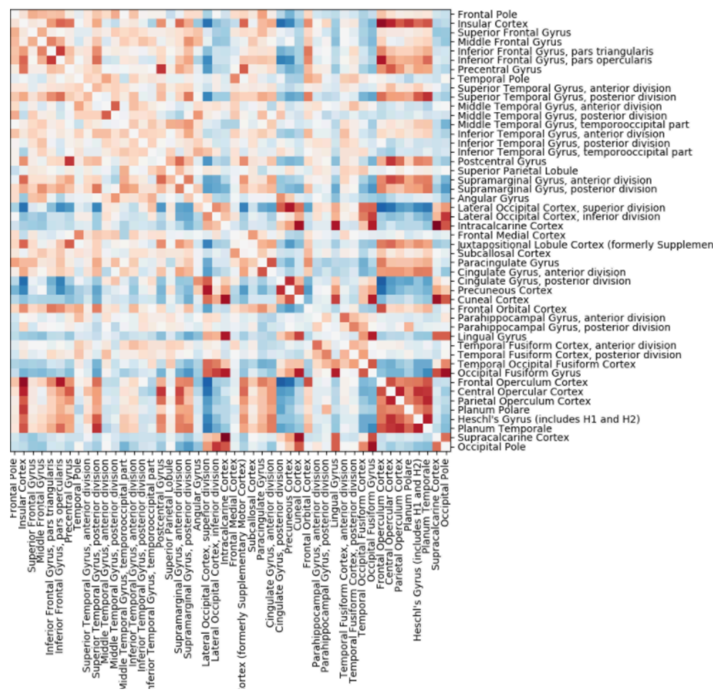
[0] what is functional connectivity?
[1] correlation vs causation
[2] Pearson’s correlation vs partial correlation
[3] infomation-theoretical measures of functional connectivity
[4] time domain vs frequency domain
[5] static vs dynamic functional connectivity
[6] hierarchical models of functional connectivity

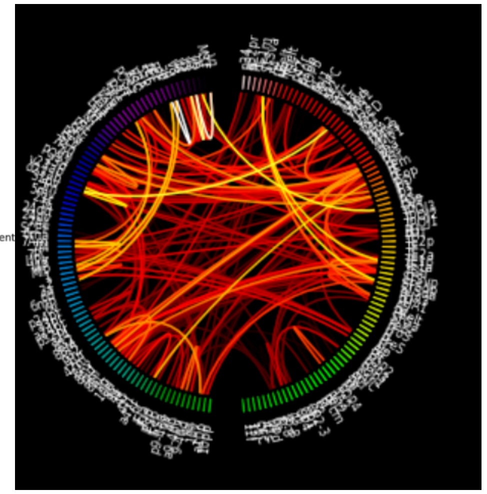
[7]* higher order associations in the networks (multiple body interactions, graph theory)

for most of these points, we will be using in-house written codes rather than Python packages

Special functions at this project

[0] who is good in data visualization?
…and could implement some alternative visualizations of functional connectomes?





[1] who is good at scientific writing?
…and could write one-two paragraphs (with references) in the Overleaf project about:
[a] Pearson's correlation (subsection 2.2.1)
[b] partial correlation (subsection 2.2.2)
[c] mutual information (subsection 2.2.3)
[d] coherence (subsection 2.3.1)
[e] dynamic functional connectivity (subsection 2.4.1)
[f] hierarchical models of functional connectivity (subsection 2.4.2)

[2] who is good at multitasking?
…and could be the slack communication liaison?

Datasets we will be working on in this project

[1] Human Connectome Project fMRI datasets
[2] MyConnectome Project multimodal datasets
[3] open datasets on weather / stock exchange / social networks

…the list to be updated during the Brainhack

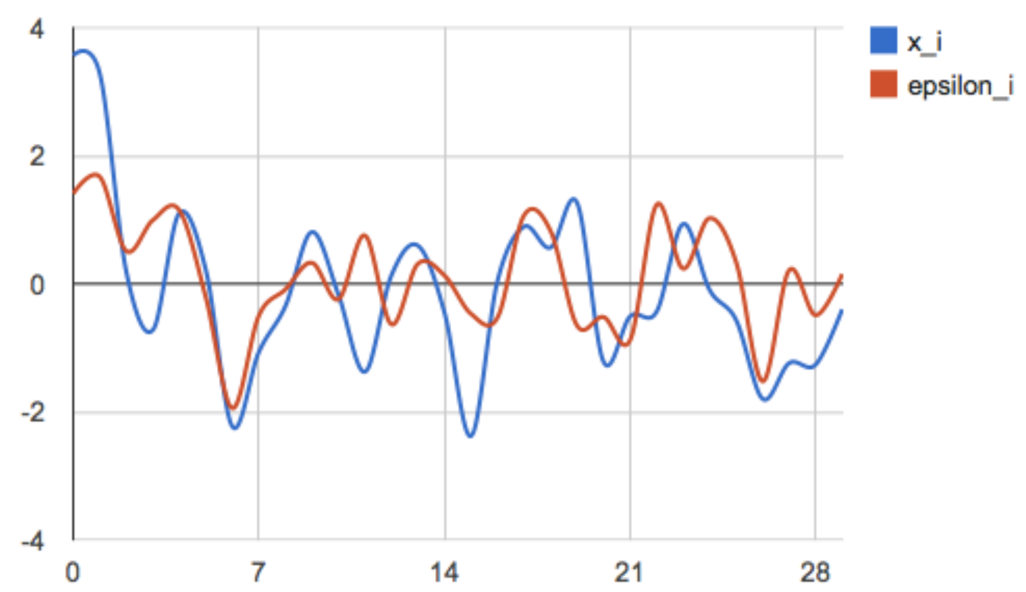
You can add your own datasets!!
(please do so both on GitHub and in the Overleaf project)


Let’s get started!

But what is the functional connectivity, on the conceptual level?

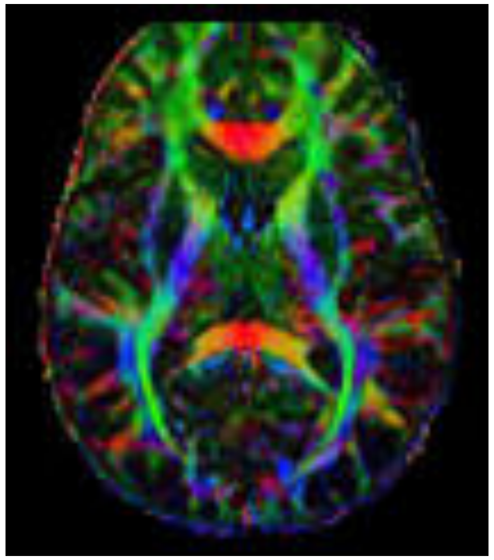
- functional connectivity = *association* between two variables

- assumption: if two variables correlate, they are a part of the same process

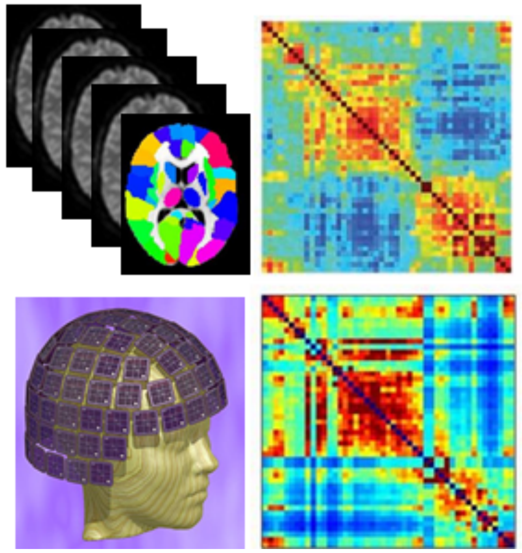


- there is no *official* definition of functional connectivity

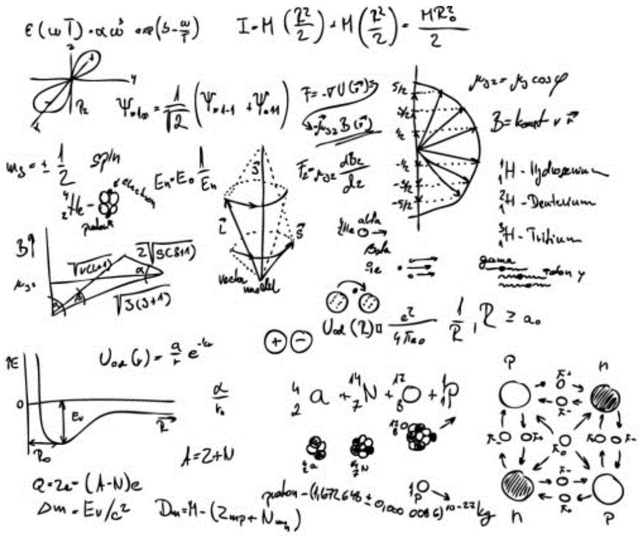
The concept of functional connectivity comes from brain research



anatomical connectivity



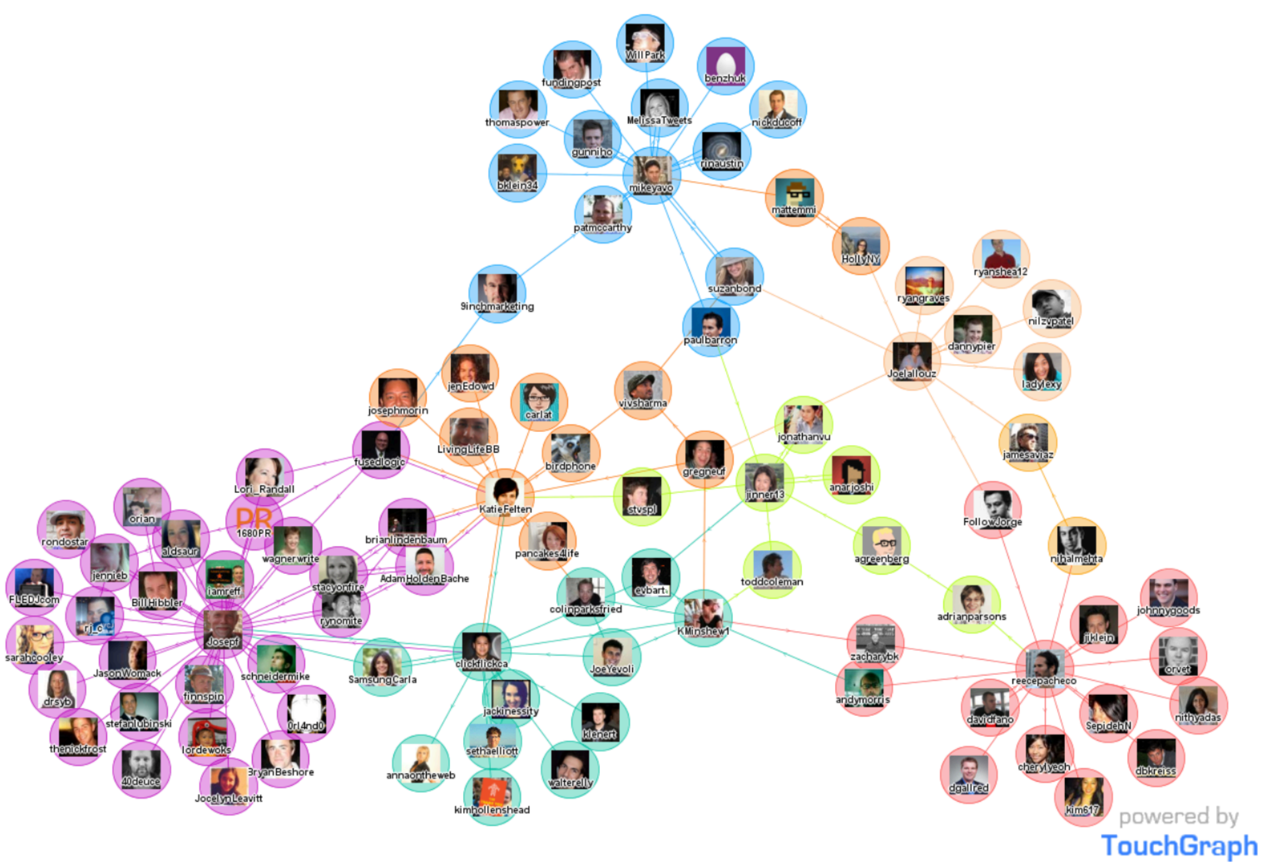
functional connectivity



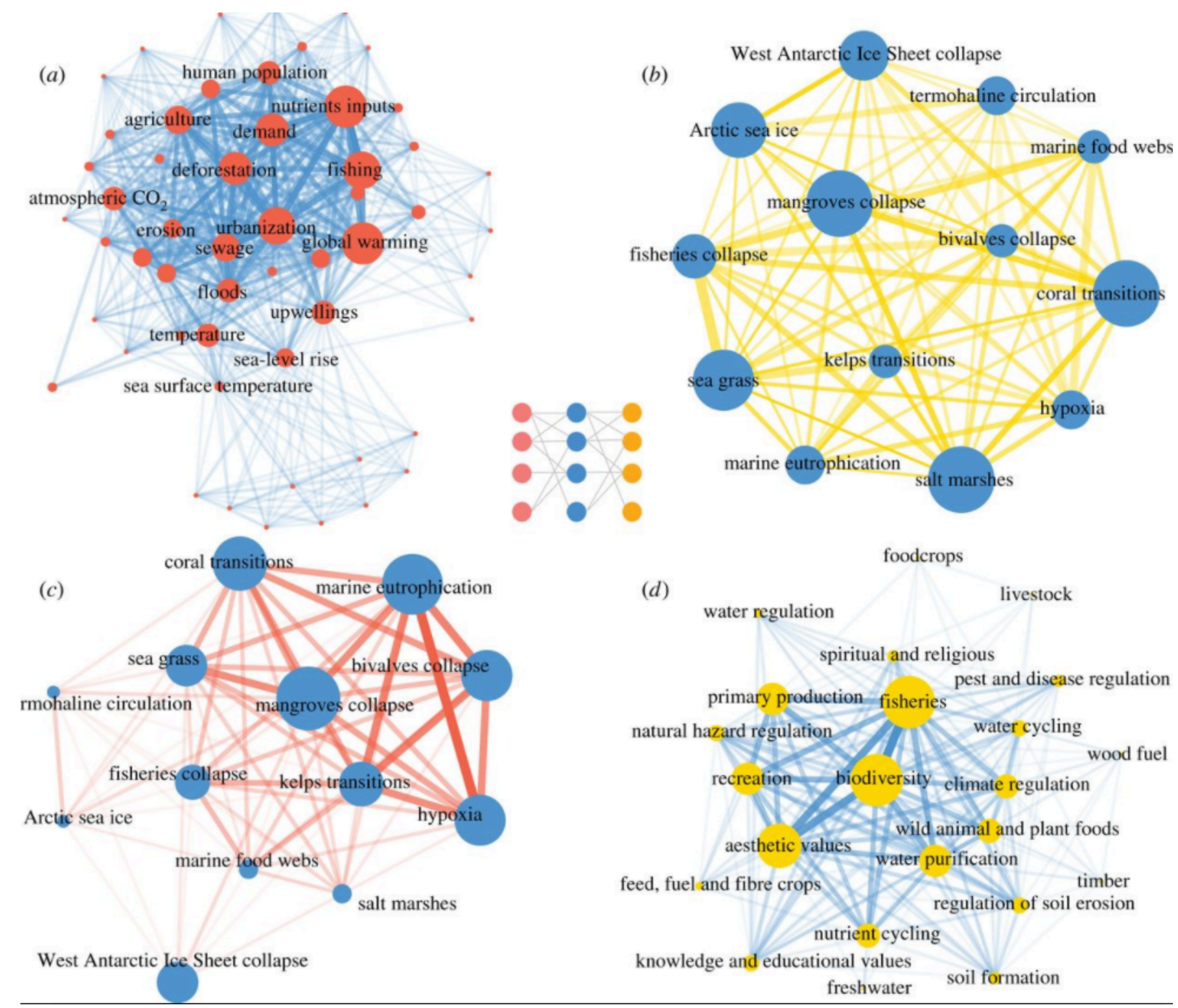
effective connectivity

In what other areas of natural sciences can we define functional connectivity?

- in any area where we have networks of interacting nodes

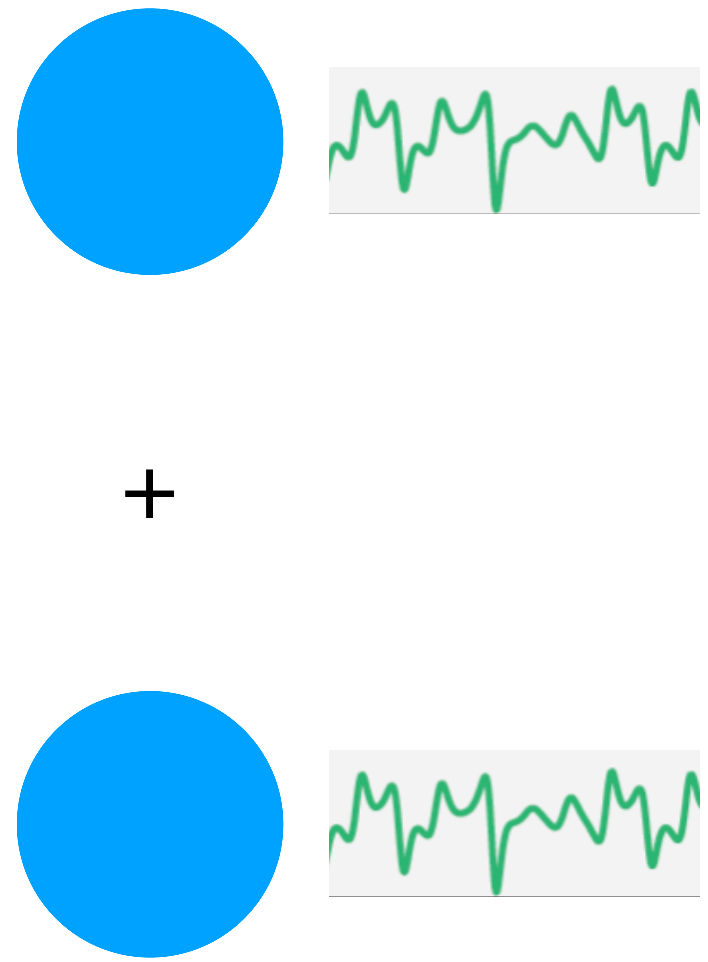


social networks

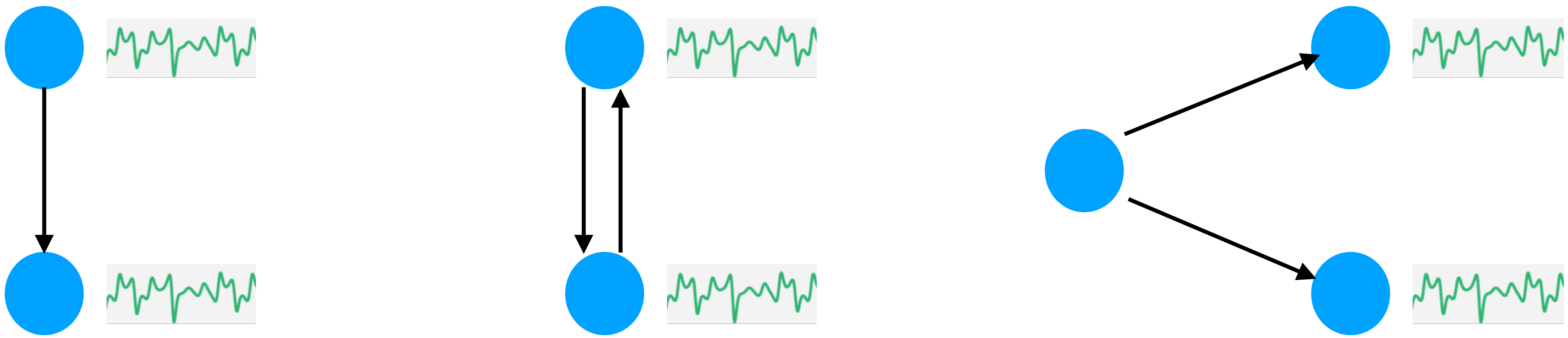


marine ecosystems
Rocha et al. (2014) Marine regime shifts: drivers and impacts on ecosystems services. *Philosophical Transactions of the Royal Society B*

Correlation vs causation



Correlation vs causation



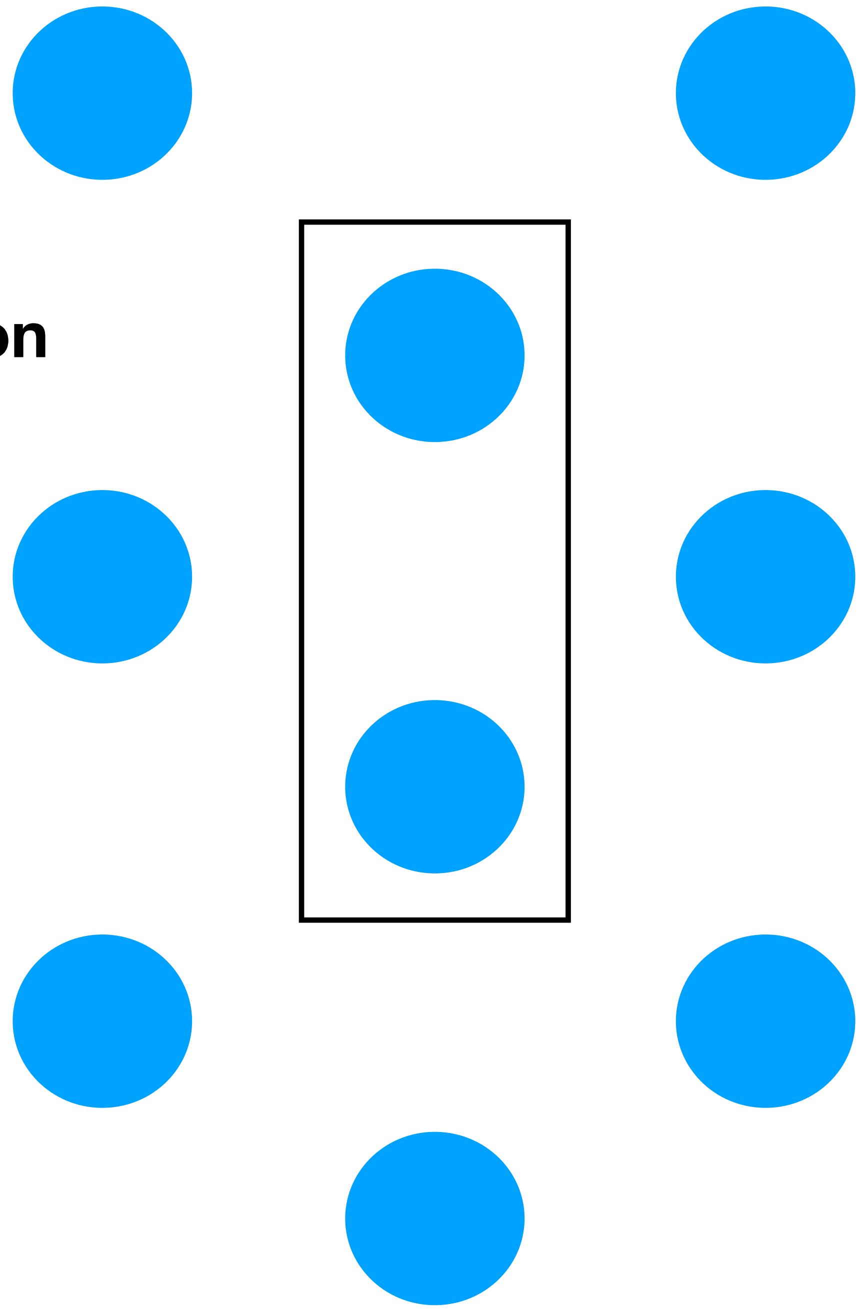
example: polio epidemy in 1950-1960

source: Freakonomics movie

How to *operationalize* the functional connectivity?

Pearson’s correlation vs partial correlation

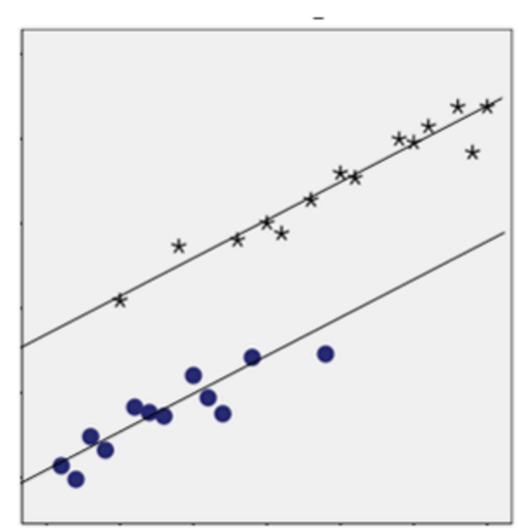
picture with linear interaction



Pearson’s correlation (Pearson’s r)

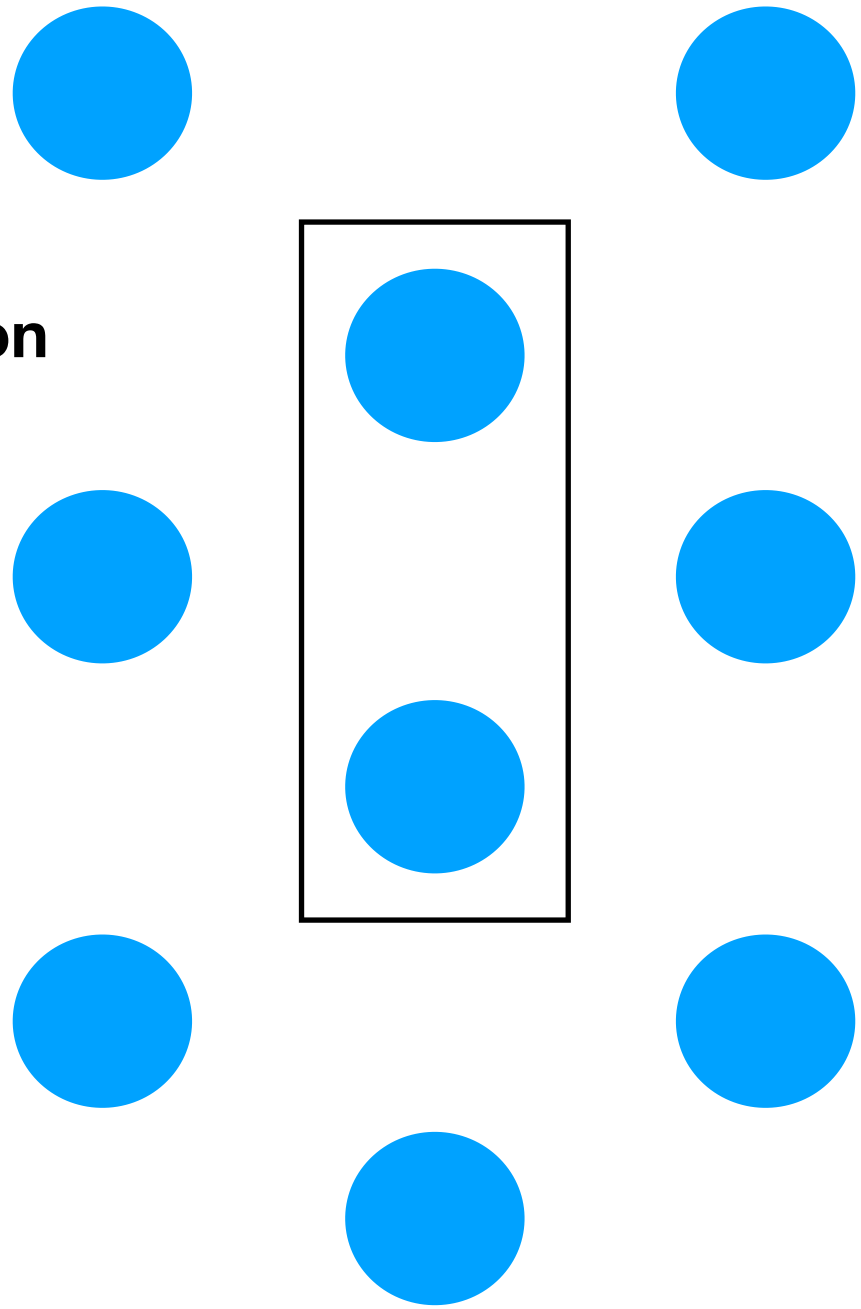
a pairwise measure of association which treats the pair of nodes as an isolated system

Pearson’s r is parametric and assumes linear relationship between variables



Pearson’s correlation vs partial correlation

picture with linear interaction

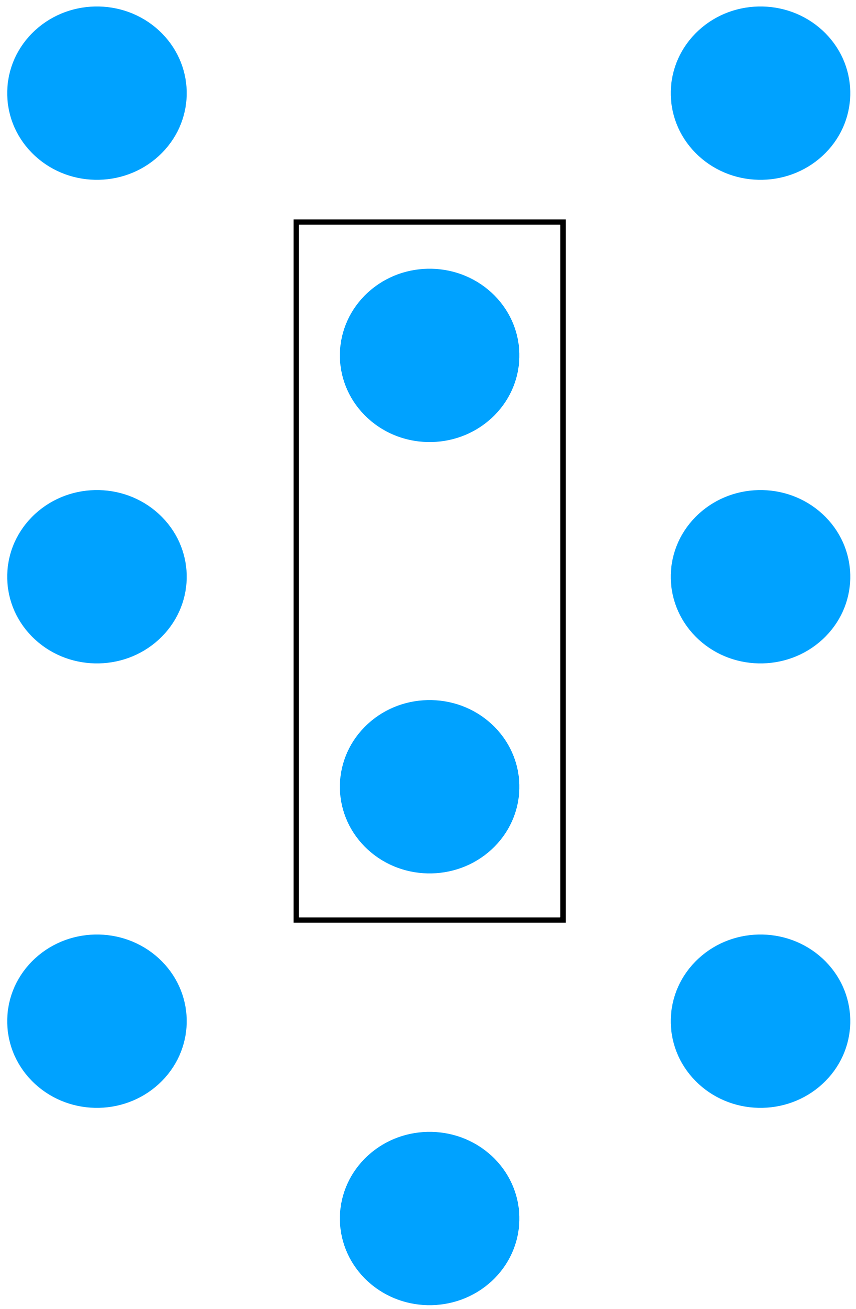


[1] operationalization:
one standard way:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

where
cov - covariance
σ - standard deviation

Pearson’s correlation vs partial correlation

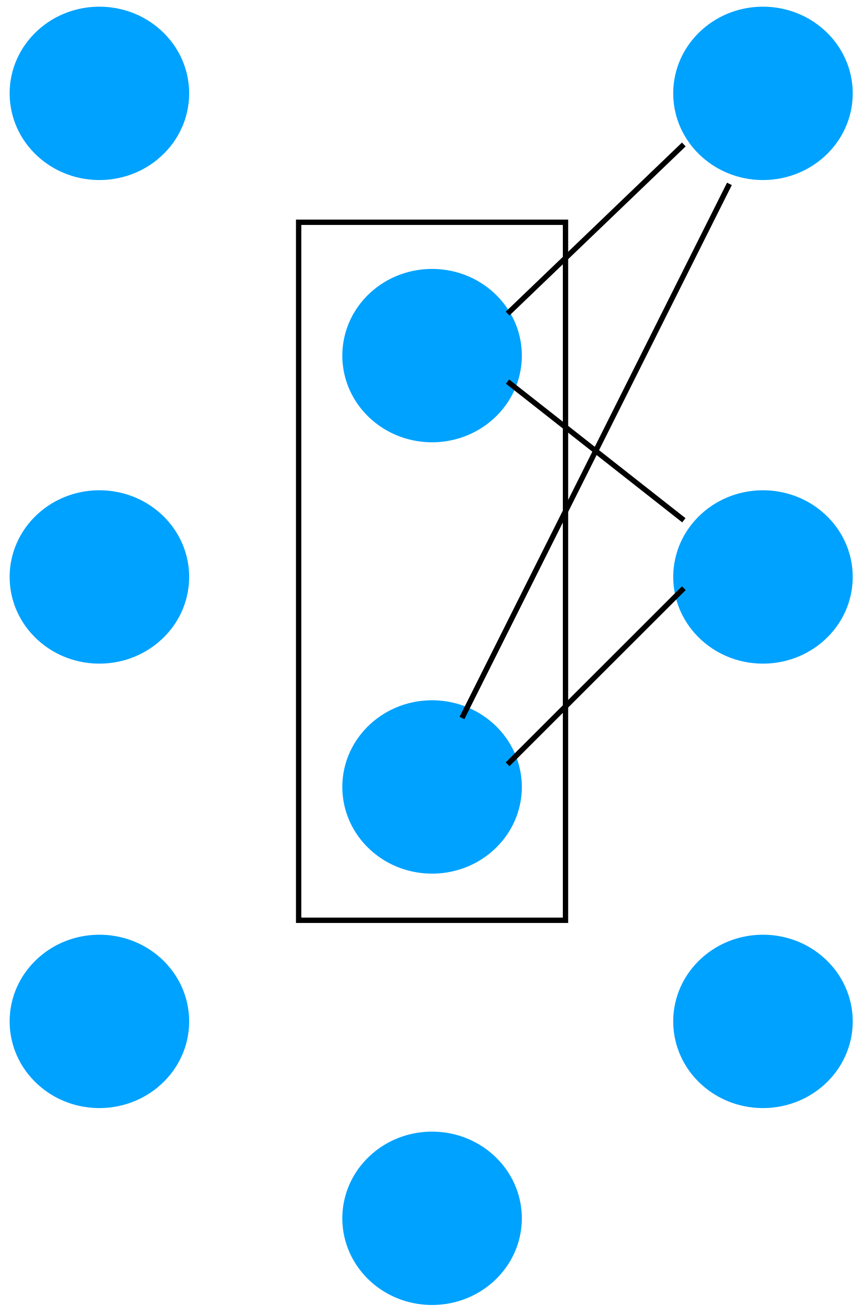


[2] building confidence intervals:
multiple ways:

[a] with use of t-statistic (MATLAB):
The p-value is computed by transforming the correlation to create a *t* statistic having N-2 degrees of freedom, where N is the number of rows of X.
The confidence bounds are based on an asymptotic normal distribution of 0.5*log((1+R)/(1-R)), with an approximate variance equal to 1/(N-3)

[b] with use of Z-statistic:
Since the sampling distribution of Pearson's r is not normally distributed, Pearson's r is converted to Fisher's z' and the confidence interval is computed using Fisher's z'. The values of Fisher's z' in the confidence interval are then converted back to Pearson's r’s

Pearson’s correlation vs partial correlation

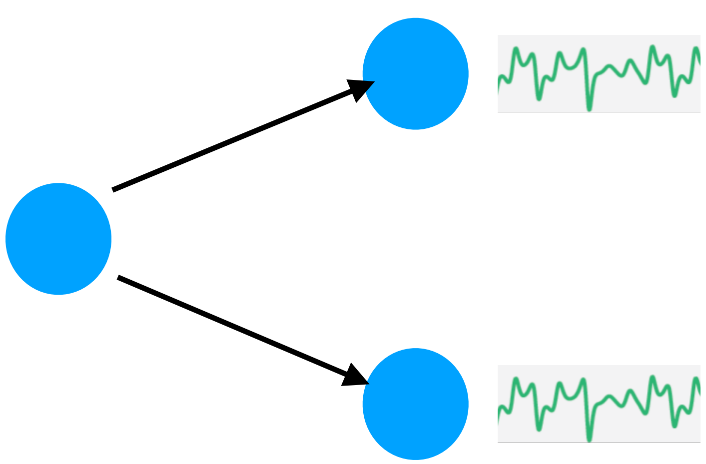


Partial correlation:

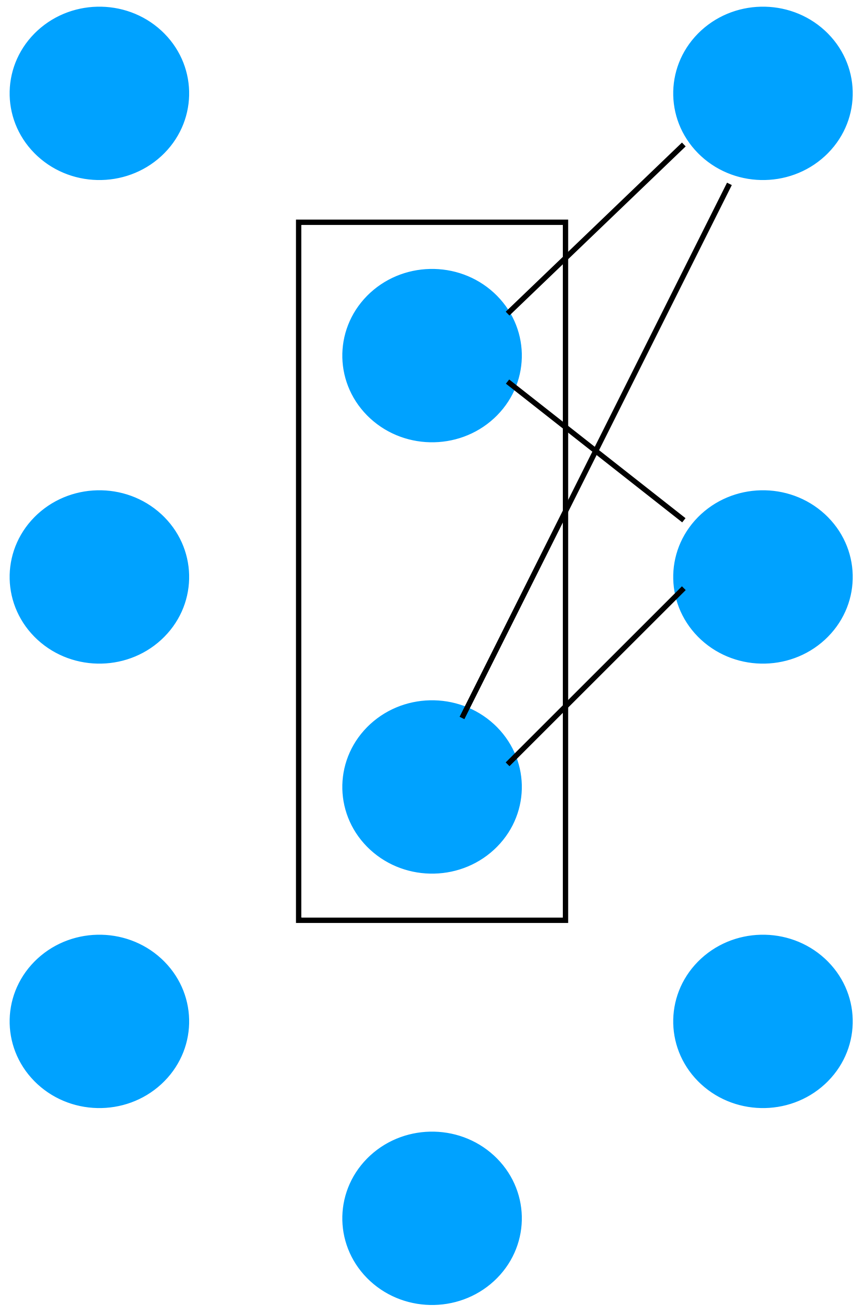
takes into account other nodes
in the network

‘first, regress the influence
of other nodes in the network
and isolate the two nodes, and then
compute the Pearson’s correlation
between the two nodes’

many ways to compute partial correlation!



Pearson's correlation vs partial correlation



[1] Operationalization:
Partial correlation can be computed in <u>multiple</u> ways.
Two most popular algorithms:

[a] via Ordinary Least-Square Regression:

$$\mathbf{w}_X^* = \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^{N} (x_i - \langle \mathbf{w}, \mathbf{z}_i \rangle)^2 \right\}$$
$$\mathbf{w}_Y^* = \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^{N} (y_i - \langle \mathbf{w}, \mathbf{z}_i \rangle)^2 \right\}$$

The residuals are then:

$$e_{X,i} = x_i - \langle \mathbf{w}_X^*, \mathbf{z}_i \rangle$$
$$e_{Y,i} = y_i - \langle \mathbf{w}_Y^*, \mathbf{z}_i \rangle$$

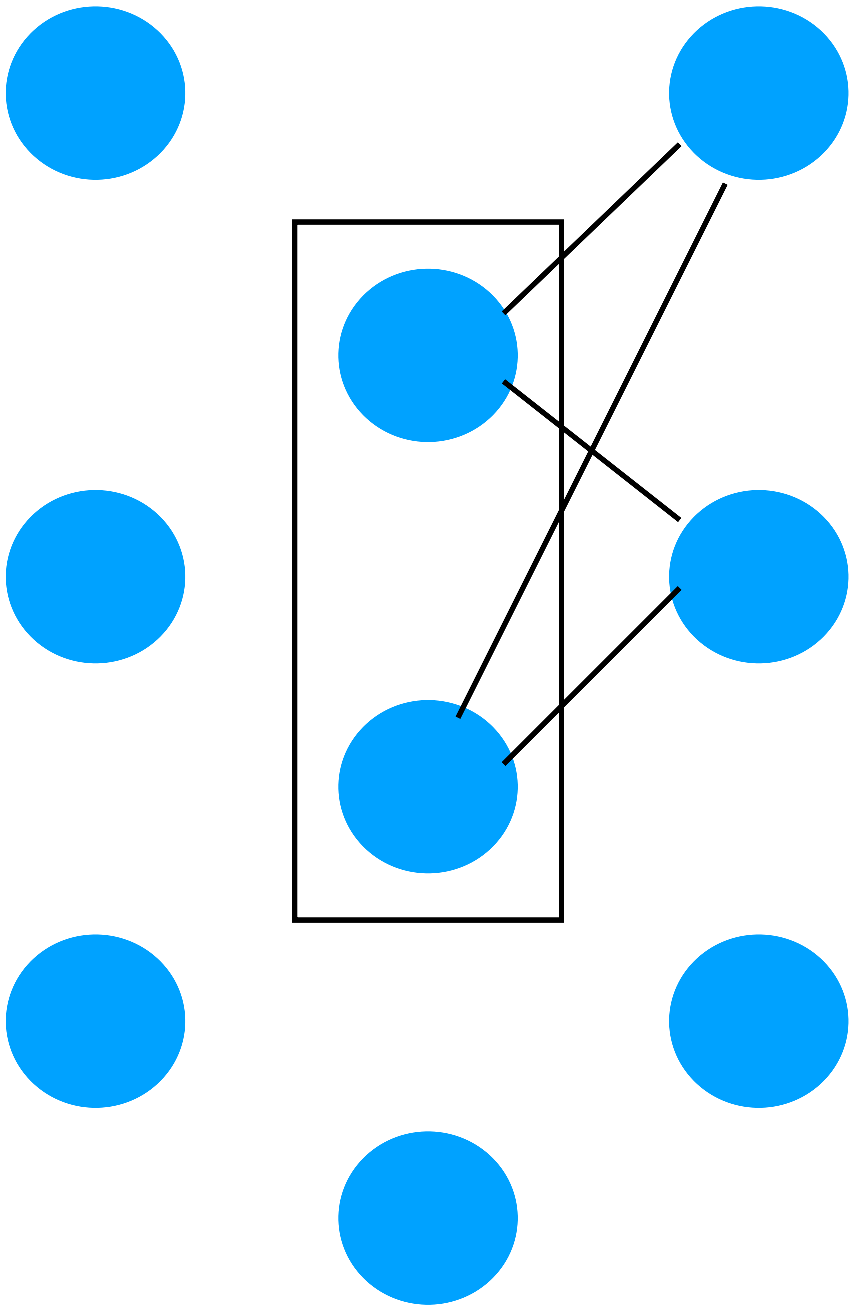
+ Pearson's correlation on the residuals

[b] *<u>an approximation</u>* through inverse covariance:

$$\rho_{ij\cdot} = \frac{\mathbb{C}_{ij}^{-1}}{\sqrt{\mathbb{C}_{ii}^{-1}\mathbb{C}_{jj}^{-1}}}$$

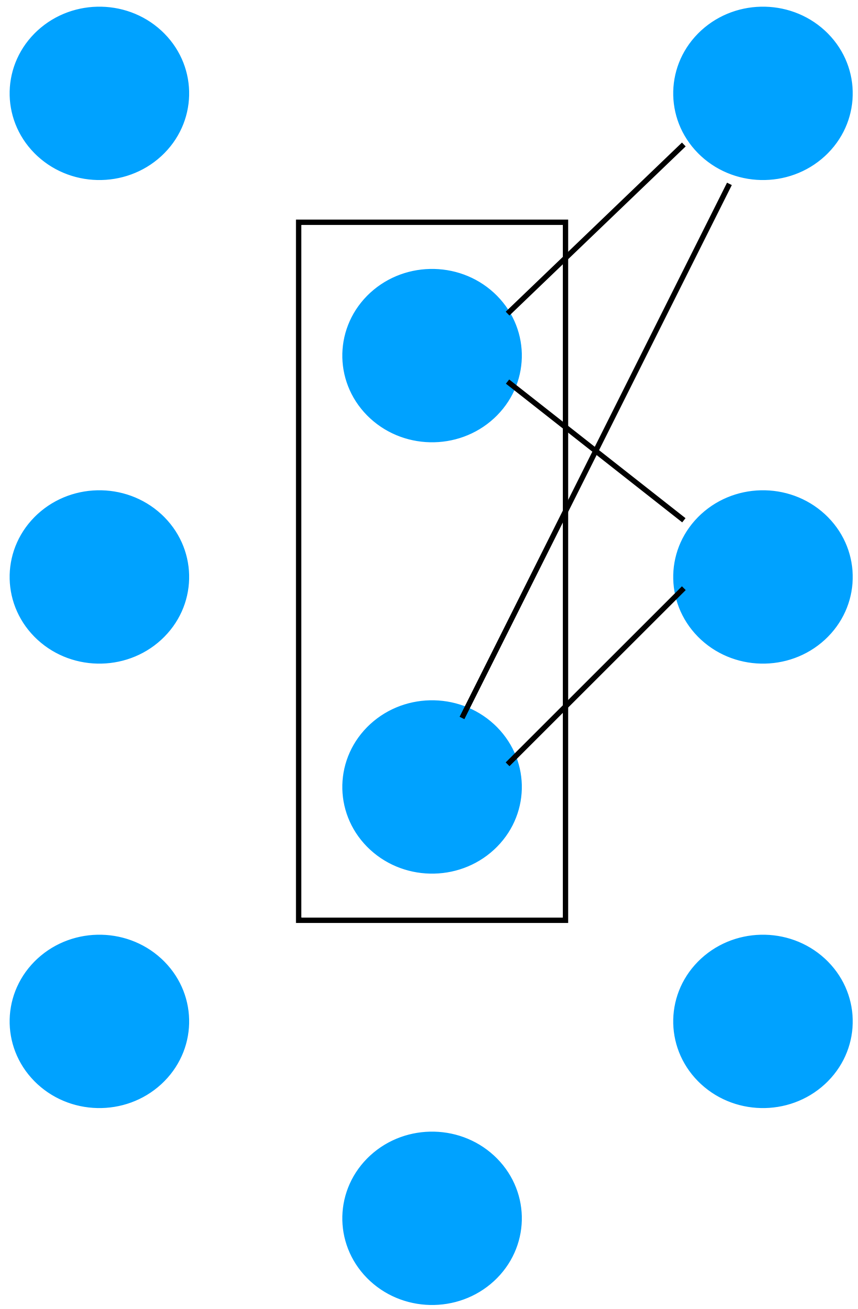
(estimation recommended for large networks)

Pearson’s correlation vs partial correlation



[2] building confidence intervals: <u>multiple ways</u> (the same as in case of Pearson’s r)

Pearson’s correlation vs partial correlation

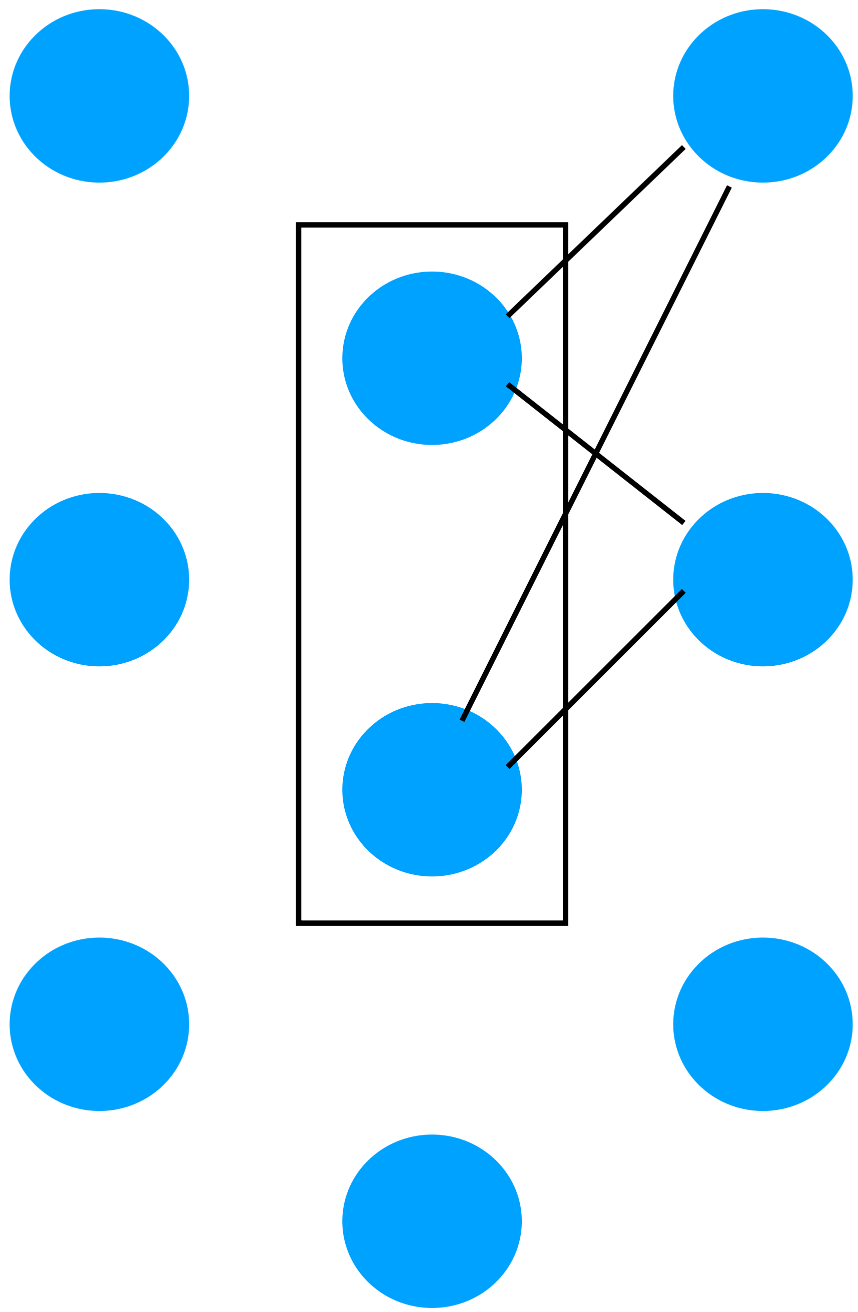


[2] building confidence intervals: multiple ways (the same as in case of Pearson’s r)

a control question: how do these two types of correlation - Pearson and partial - relate to each other?

Which one of them requires more assumptions with respect to the data?

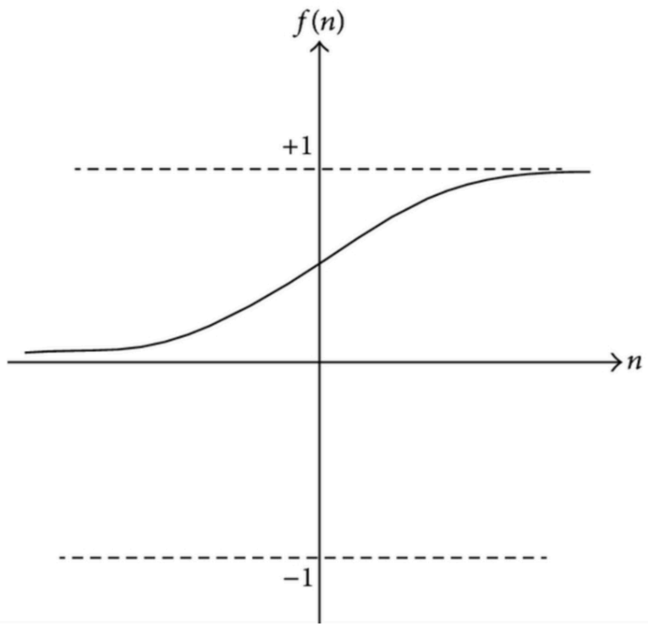
Pearson's correlation vs partial correlation



[2] building confidence intervals: multiple ways (the same as in case of Pearson's r)

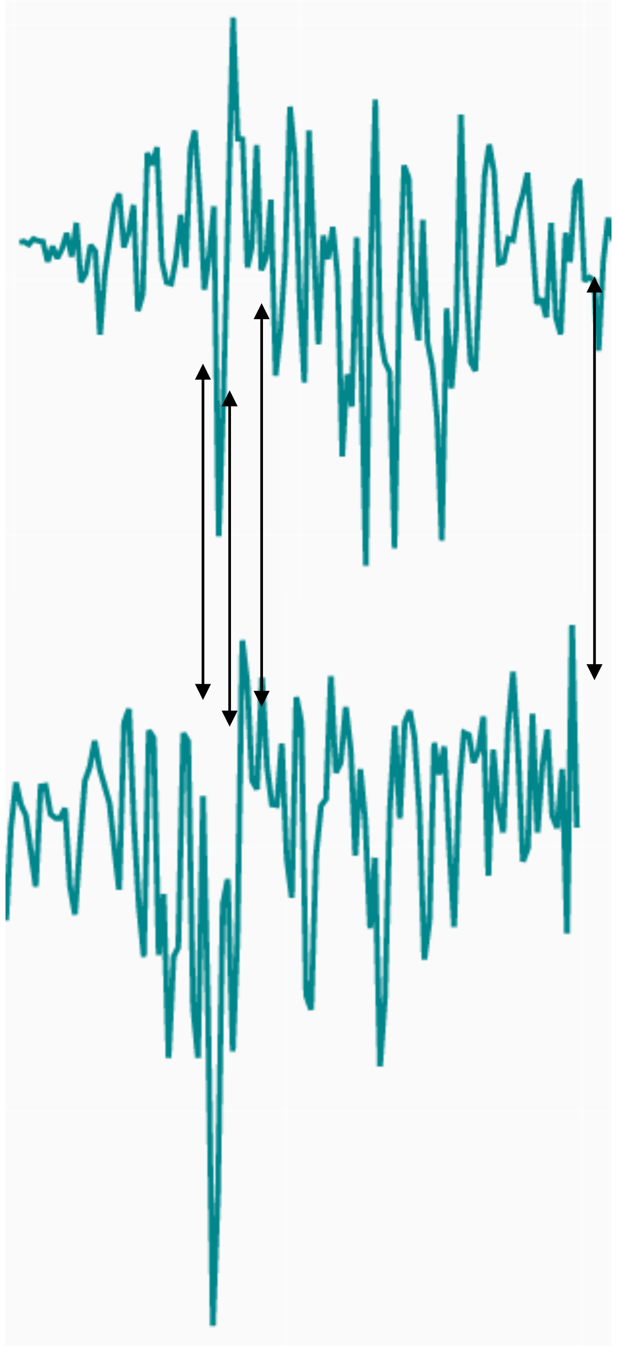
a control question: how do these two types of correlation - Pearson and partial - relate to each other?

Which one of them requires more assumptions with respect to the data?



Other measures of association between two time series

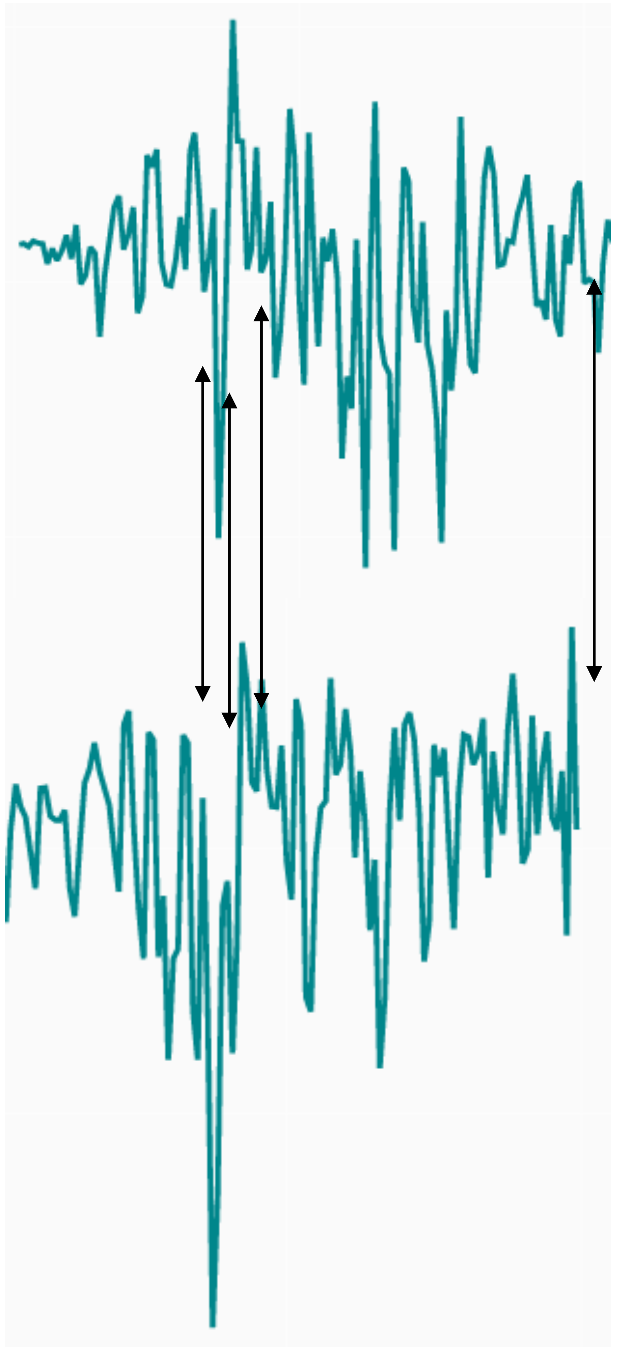
measures of the **distance** between two time series:

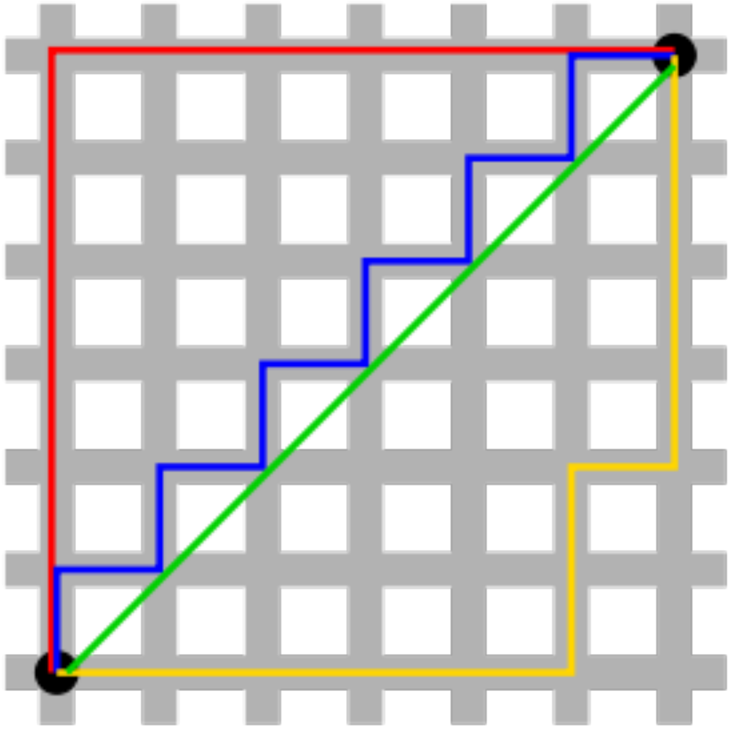


rarely encountered in brain research
but popular in economy

Other measures of association between two time series

measures of the **distance** between two time series:





Euclidean vs Manhattan distance:

Manhattan distance
is the distance between two points
as the sum of the absolute differences
of their Cartesian coordinates

Information-theoretical measures of functional connectivity

Shannon information

Shannon information (or Shannon entropy) H(x) quantifies the information contained in a signal of unknown spectral properties as the amount of uncertainty, or unpredictability. For example, a binary signal that only gets values of 0 with a probability p, and values of 1 with a probability 1 - p , is most unpredictable when p = 0.5. This is because there is always exactly a 50% chance of correctly predicting the next sample. Therefore, being informed about the next sample in a binary signal of p = 0.5 reduces the amount of uncertainty to a higher extent than being informed about the next sample in a binary signal of, say, p = 0.75. This can be interpreted as a larger amount of information contained in the first signal as compared to the latter.

The formula which quantifies the information content according to this rule reads as follows:

$$H(X) = -\sum_i P(x_i)\log_2 P(x_i)$$

where xi are the possible values in the signal (for the binarized signal, there are only two possible values: 0 and 1).

Information-theoretical measures of functional connectivity

Mutual information quantifies the reduction in uncertainty obtained about one random variable, through the other random variable.

$$
\begin{aligned}
I(X;Y) &\equiv \mathrm{H}(X) - \mathrm{H}(X|Y) \\
&\equiv \mathrm{H}(Y) - \mathrm{H}(Y|X) \\
&\equiv \mathrm{H}(X) + \mathrm{H}(Y) - \mathrm{H}(X,Y) \\
&\equiv \mathrm{H}(X,Y) - \mathrm{H}(X|Y) - \mathrm{H}(Y|X)
\end{aligned}
$$

$$
I(X;Y) = \sum_{x,y} P_{XY}(x,y) \log \frac{P_{XY}(x,y)}{P_X(x)P_Y(y)}
$$

where
Pxy - joint probability for X and Y
Px, Py - marginals

Facts about mutual information

The difference between correlation and mutual information

Pearson's correlation measures the *linear* relationship (Pearson's correlation) between two variables, X and Y.

Mutual information is more general and measures the reduction of uncertainty in Y after observing X. Therefore, MI can measure non-monotonic relationships and other more complicated relationships.

practical implementation of mutual information contains a free parameter (bin size)

not much done in the literature in terms of efficient ways of computing confidence intervals for mutual information

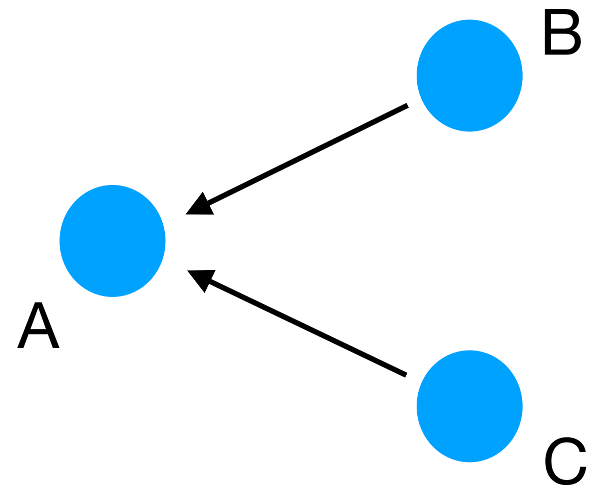
Most common traps in functional connectivity research

[1]
- Pearson's correlation can result in networks denser than the true underlying connectivity
- partial correlation can result in networks sparser than the true underlying connectivity

<u>Conclusion:</u> it is a good idea in general to compare compute both of them as the truth is somewhere in between

[2] the multiple comparisons problem —> true degree of freedom is unknown

[3] Berkson's paradox



happens when calculating partial correlation with use of Ordinary Least Squares regression for two unrelated sources with a common sink

regressing time series A from B and C induces a *spurious negative correlation* between B and C

Assignment 1: in iPython notebook

1_measures_of_functional_connectivity_basic.ipynb

you have got an overview of the aforementioned methods for functuonal connectivity in the time domain.
Go through the codes, read the implementation of the four methods:

- Pearson's correlation
- partial correlation
- measures of similarity
- mutual information

(a) How do the outcome connectomes differ from each other?
Are the functional connectomes coming from Pearson and partial correlation similar to each other?
Discuss this topic with your partner.

(b) Load a dataset from another subject and compare the results between two subjects. Are methods differ in terms of cross-subject reprodicibility?

(b) Create a new notebook 'Assignment1', choose datasets other than fMRI from the 'datasets' datebase, and create other functional connectomes in datasets coming from other disciplines: stock exchange datasets, climate data etc. Which type of significance testing would you use for these datasets: permutation testing or t-tests? How do these connectomes look like? Do you see any patterns?

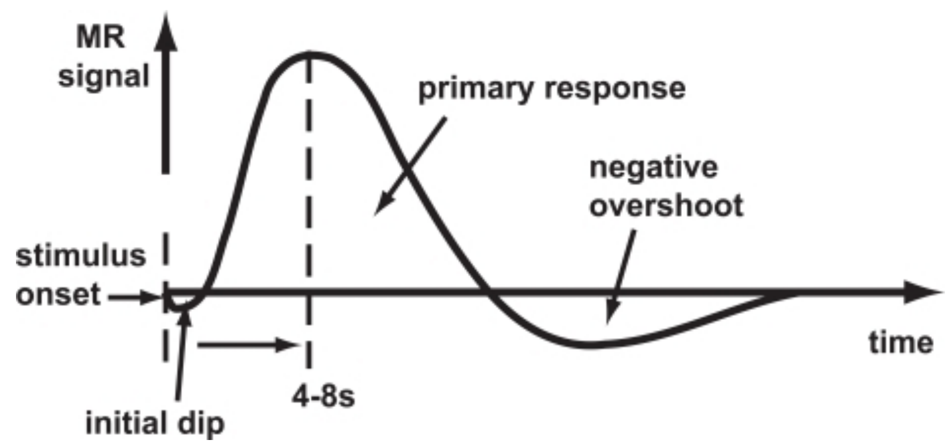
(d) Can you think of any other measures of functional connectivity in the time domain? (space for creativity) Uou can propose any function of the data you can think of. You can define it in a separate function, give it a name and

- test on a pair of random time serie in order to check whether your measure is unbiased
- apply to fMRI datasets and compare to other, established measures of functional connectivity

General remark about building confidence intervals for functional connectivity measures

- building confidence intervals for correlation variables and mutual information through **t-test** or **F-test** contains an assumption that the samples are *independent*

- in many datasets this assumption is not true. E.g. in the fMRI data, samples are dependent: slow haemodynamics induces crosscorrelations between subsequent samples



Possible solutions:
[1] permutation testing

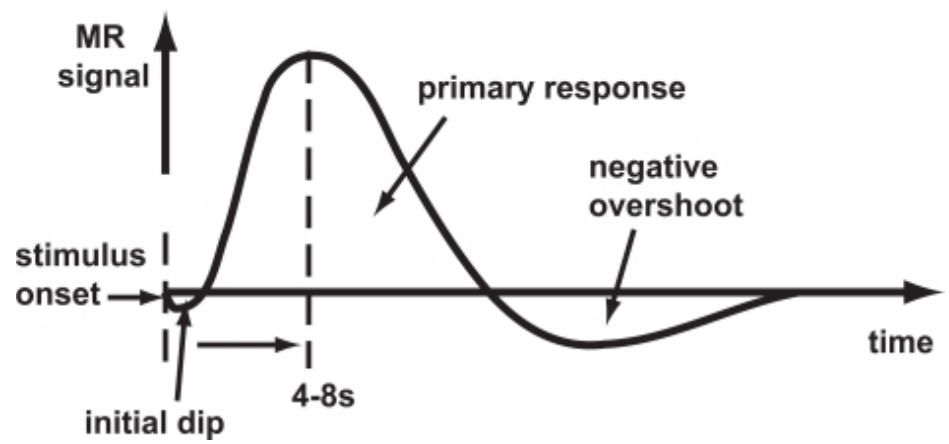
[2] through mixture modeling
(Bielczyk & Walocha et al, NeuroImage 2017)

General remark:
in datasets where samples are autocorrelated, building confidence intervals on the second level of analysis is advised

General remark about building confidence intervals for functional connectivity measures

- building confidence intervals for correlation variables and mutual information through **t-test** or **F-test** contains an assumption that the samples are *independent*

- in many datasets this assumption is not true. E.g. in the fMRI data, samples are dependent: slow haemodynamics induces crosscorrelations between subsequent samples



Possible solutions:

[1] permutation testing

[2] through mixture modeling
(Bielczyk & Walocha et al, NeuroImage 2017)

General remark:
in datasets where samples are autocorrelated, building confidence intervals on the second level of analysis is advised

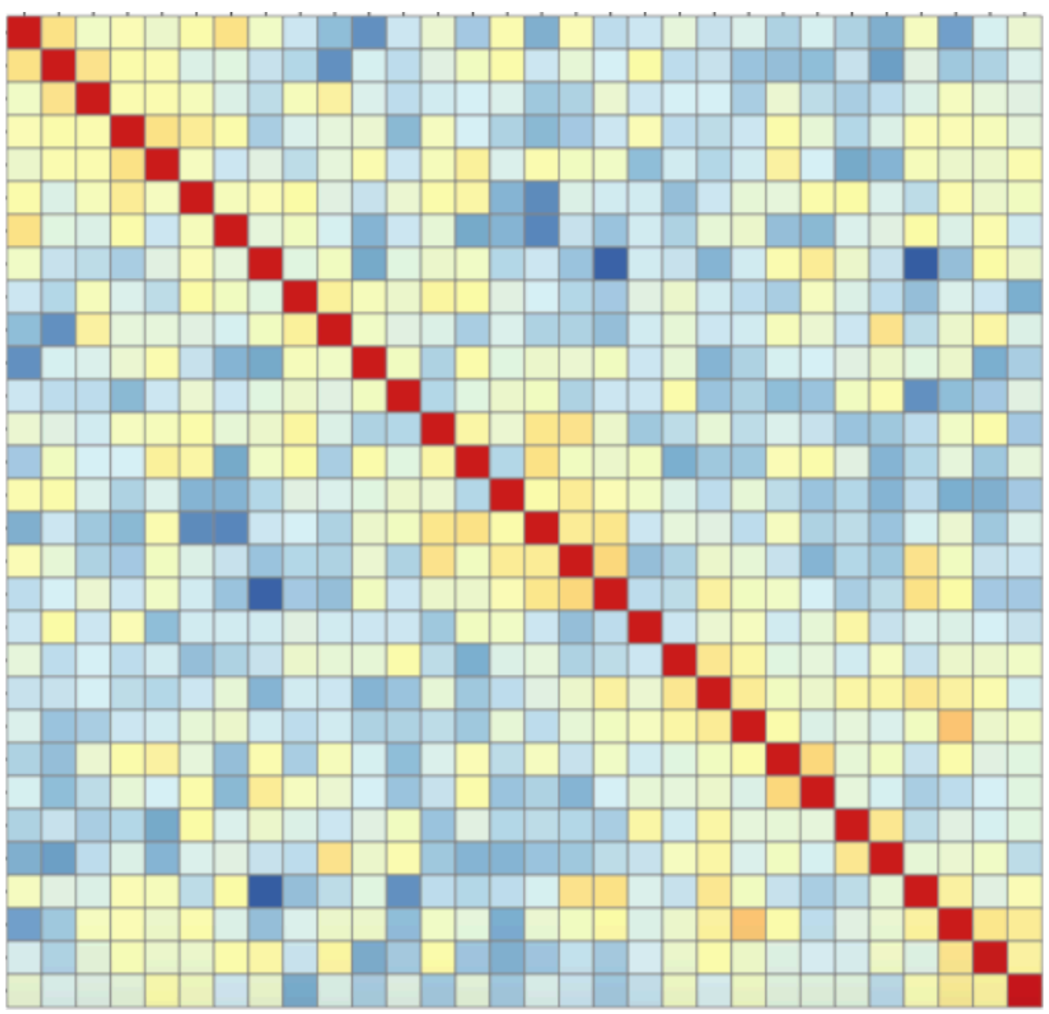
General remark about building confidence intervals for functional connectivity measures

- how about the difference between two conditions?
- a number of subjects, task versus rest

- paired t-test can be a solution, as subjects are independent from each other, but the measurement is repeated per subject

General remarks about building confidence intervals for functional connectivity measures

FC matrices can be of any arbitrary size
—> separate p value for every connection
—> testing for multiple comparisons necessary



[1] Bonferroni correction

Assumption: comparisons are independent from each other
Method: divide the confidence level by the number of comparisons

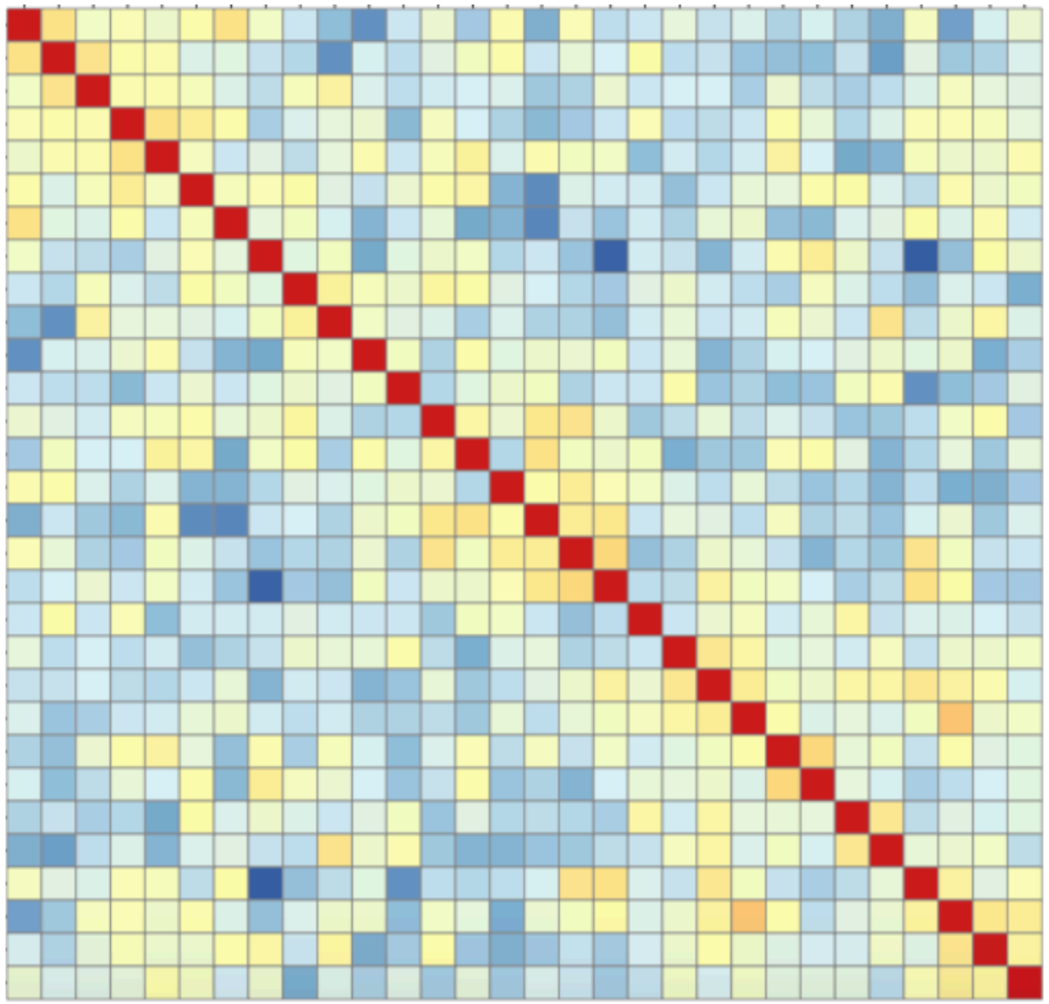
[2] Benjamini-Hochberg correction

Assumption: comparisons independent from each other
Method:

1. Put the individual p-values in ascending order.
2. Assign ranks to the p-values. For example, the smallest has a rank of 1, the second smallest has a rank of 2.
3. Calculate each individual p-value's Benjamini-Hochberg critical value, using the formula (i/m)Q, where:
   - i = the individual p-value's rank,
   - m = total number of tests,
   - Q = the false discovery rate (a percentage, chosen by you).
4. Compare your original p-values to the critical B-H from Step 3; find the largest p value that is smaller than the critical value.

General remarks about building confidence intervals for functional connectivity measures

FC matrices can be of any arbitrary size
—> separate p value for every connection
—> testing for multiple comparisons necessary



[1] Bonferroni correction

Assumption: comparisons are independent from each other
Method: divide the confidence level by the number of comparisons

[2] Benjamini-Hochberg correction

Assumption: comparisons independent from each other
Method:

1. Put the individual p-values in ascending order.
2. Assign ranks to the p-values. For example, the smallest has a rank of 1, the second smallest has a rank of 2.
3. Calculate each individual p-value's Benjamini-Hochberg critical value, using the formula (i/m)Q, where:
   - i = the individual p-value's rank,
   - m = total number of tests,
   - Q = the false discovery rate (a percentage, chosen by you).
4. Compare your original p-values to the critical B-H from Step 3; find the largest p value that is smaller than the critical value.

General remark about building confidence intervals for functional connectivity measures

- how about the difference between two conditions?
- a number of subjects, task versus rest

- paired t-test can be a solution, as subjects are independent from each other, but the measurement is repeated per subject

Assignment 2: in iPython notebook
3_measures_of_functional_connectivity_second_level_computation_significance_testing.ipynb

you have examples of significance testing in fMRI data with use of permutation testing (effect versus no effect, e.g. answering the question: which correlations in the resting state are significant?)

The task now is to answer another question: which differences between two conditions (e.g., between *task* and *rest* or, *between resting state day 1* versus *resting state day 2)* are significant? Try to answer this question with use of the available fMRI second-level datasets, which you can find in the folder 'output_second_level'

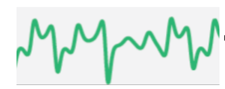
Note: when you are comparing task versus rest, choose the version derived from the resting state datasets of the same sample length as the task dataset (e.g. 558 sampels for the MOTOR task and 800 sampels for the WM task)

Time domain vs frequency domain

poor time resolution
no oscillatory patterns

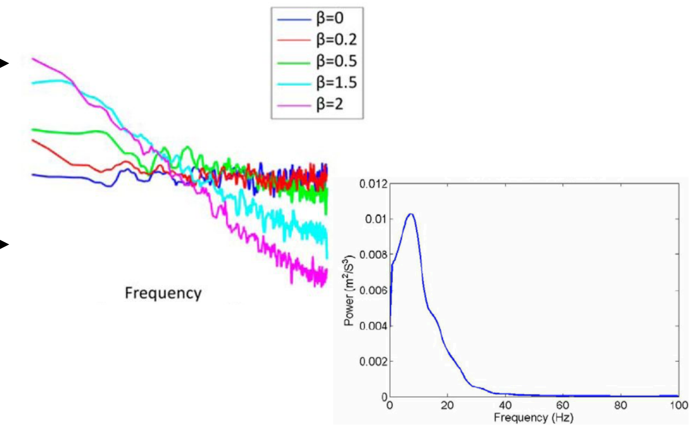
X(t)

Y(t)

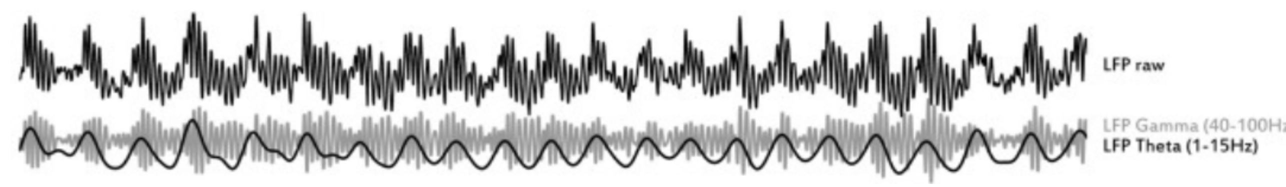


X(t)

Y(t)



power spectrum



Correlation

> Pearson's correlation

> partial correlation

Coherence

> phase-phase coupling

> phase-amplitude coupling

> amplitude-amplitude coupling

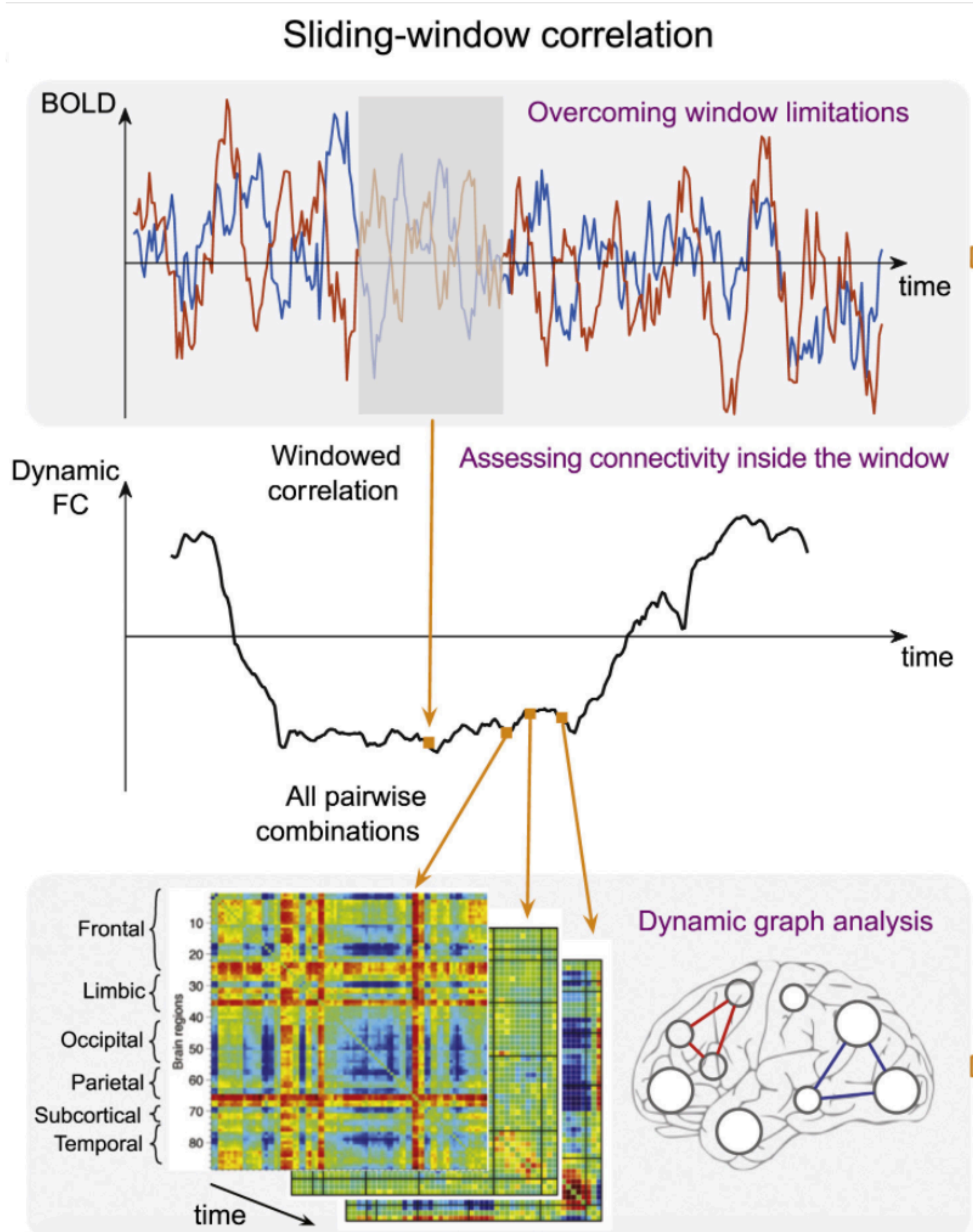
time domain

frequency domain

Static vs dynamic functional connectivity

analysis of the temporal patterns in functional connectivity with use of a *sliding window* (Sakoğlu et al, 2010)

"chronnectome"

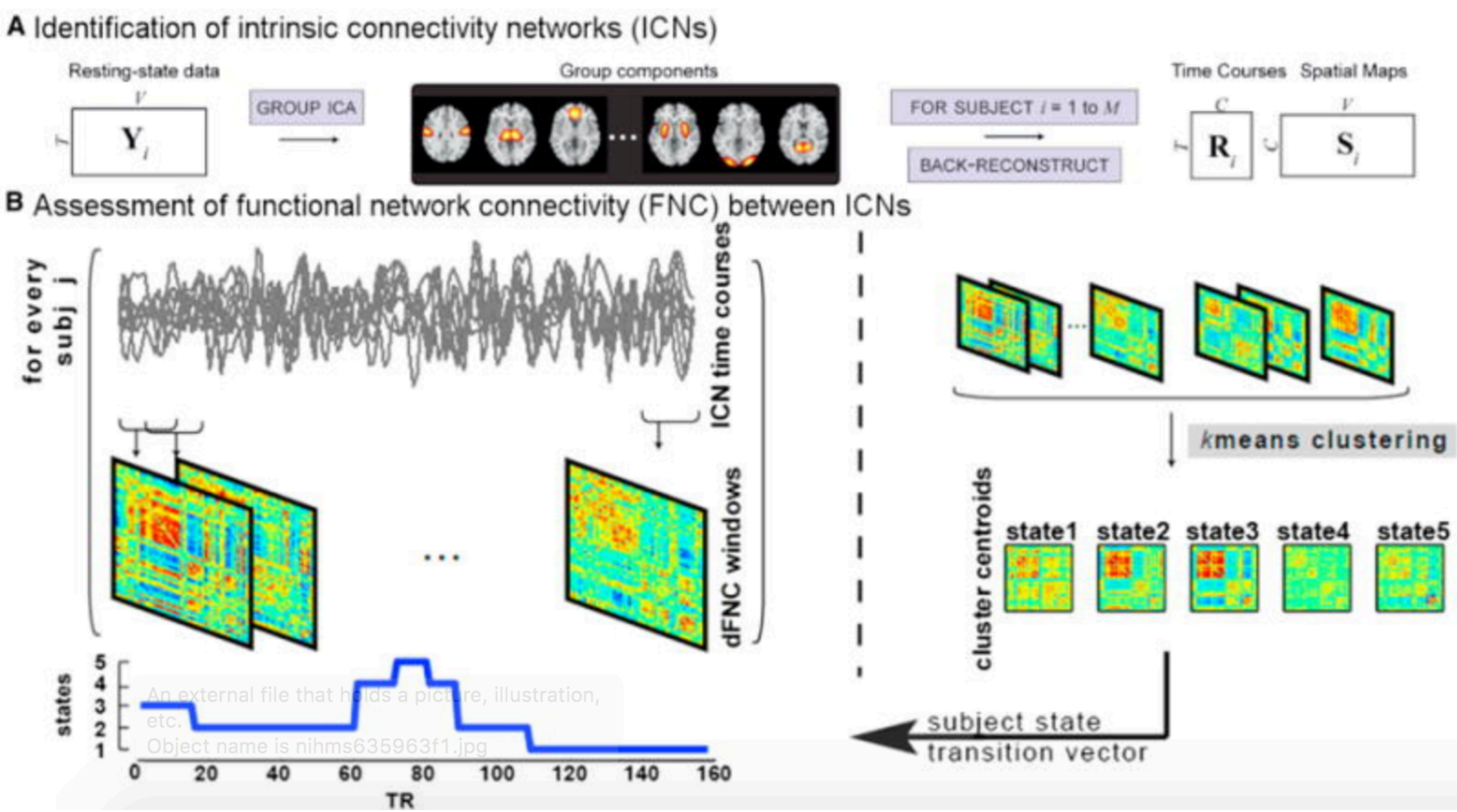


used primarily in fMRI research, see a review paper:

http://www.sciencedirect.com/science/article/pii/S1053811916307881

Static vs dynamic functional connectivity

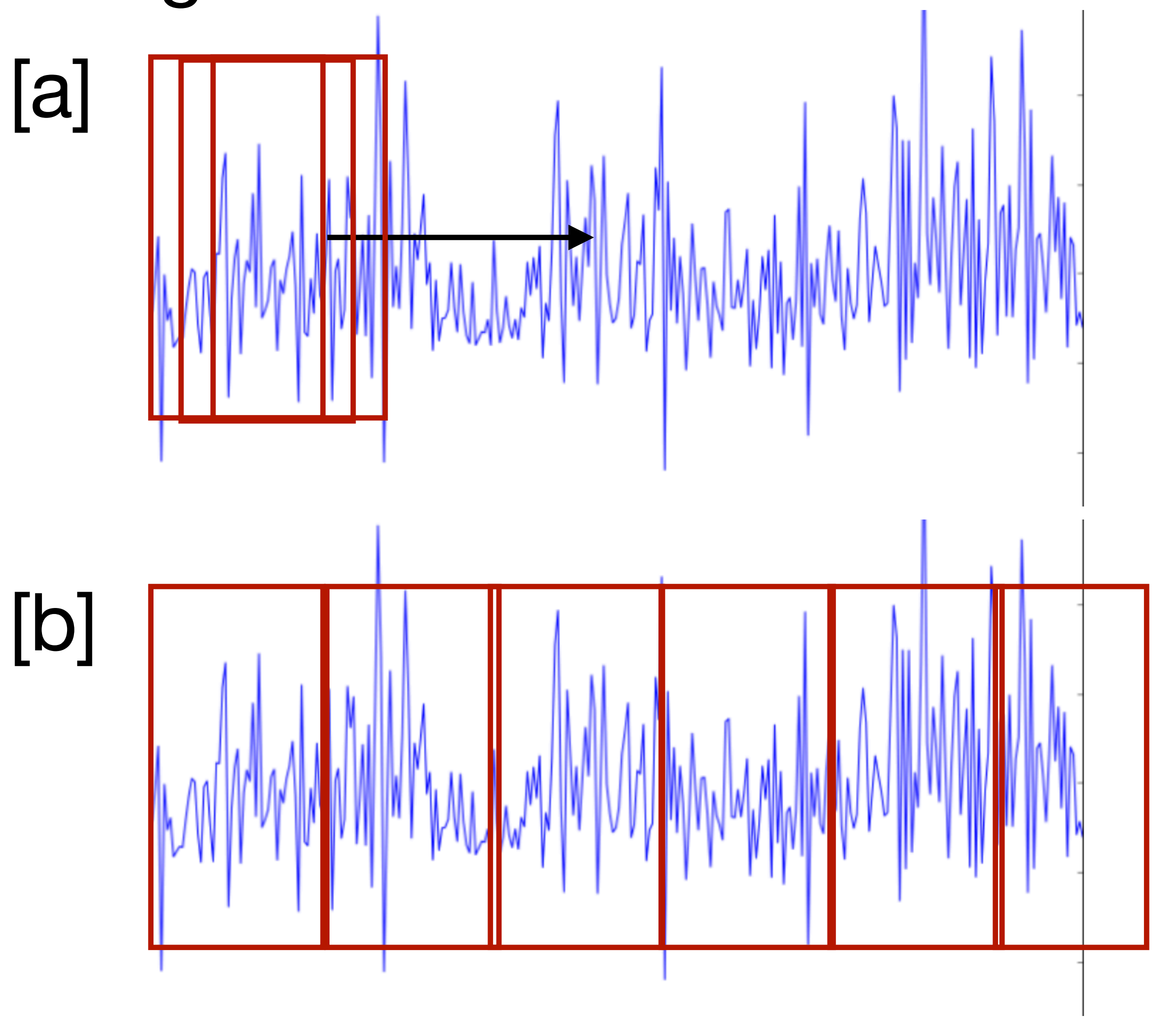
A ‘Chronnectome’ idea:



Calhoun et al. (2014)

Two main ways of computing dynamic functional connectivity (dFC)

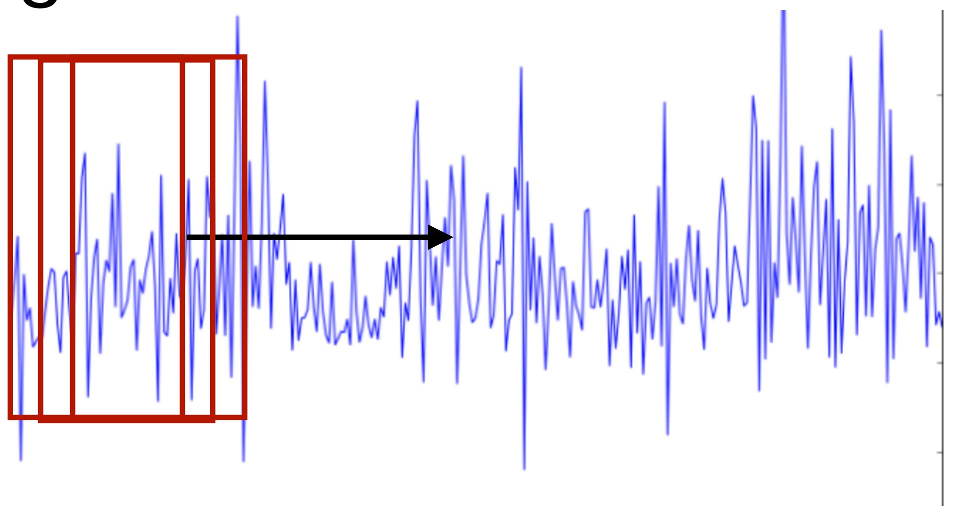
[a] through sliding window (either rectangular or more fancy)
[b] through series of windows



Two main ways of computing dynamic functional connectivity (dFC)

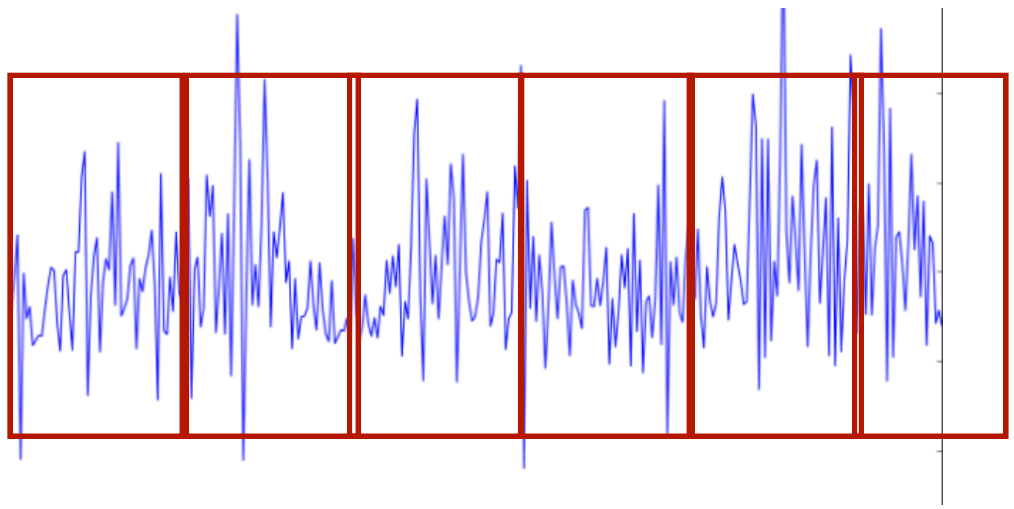
[a] through sliding window (either rectangular or more fancy)
[b] through series of windows

[a]



<u>Pros:</u>
dynamics is easier to catch
<u>Cons:</u>
windows are highly correlated

[b]

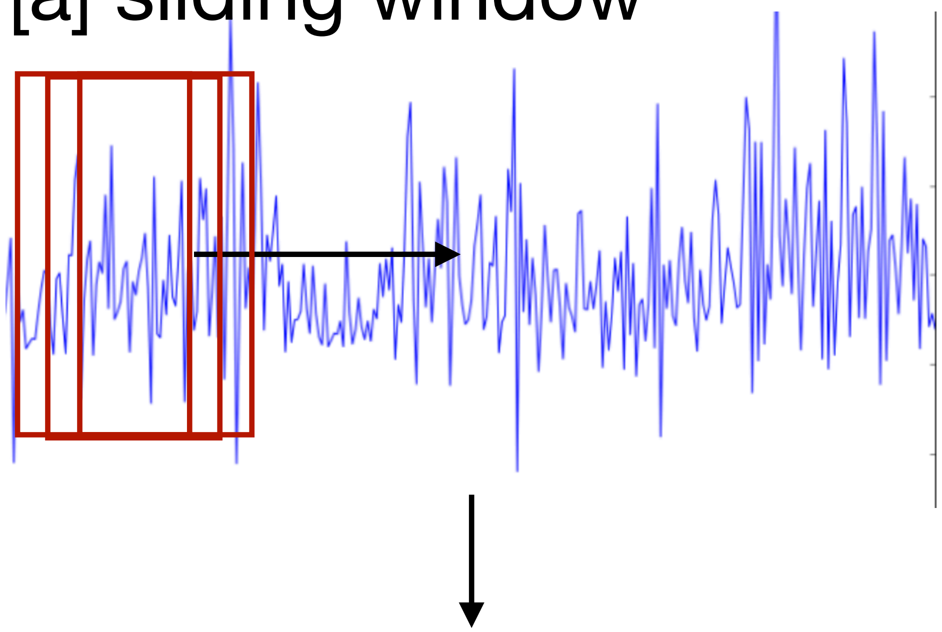


<u>Pros:</u>
FC within each window can be treated as an independent sample
<u>Cons:</u>
dynamics is very poor

Which method is better? —> Hindriks et al, NeuroImage 2016

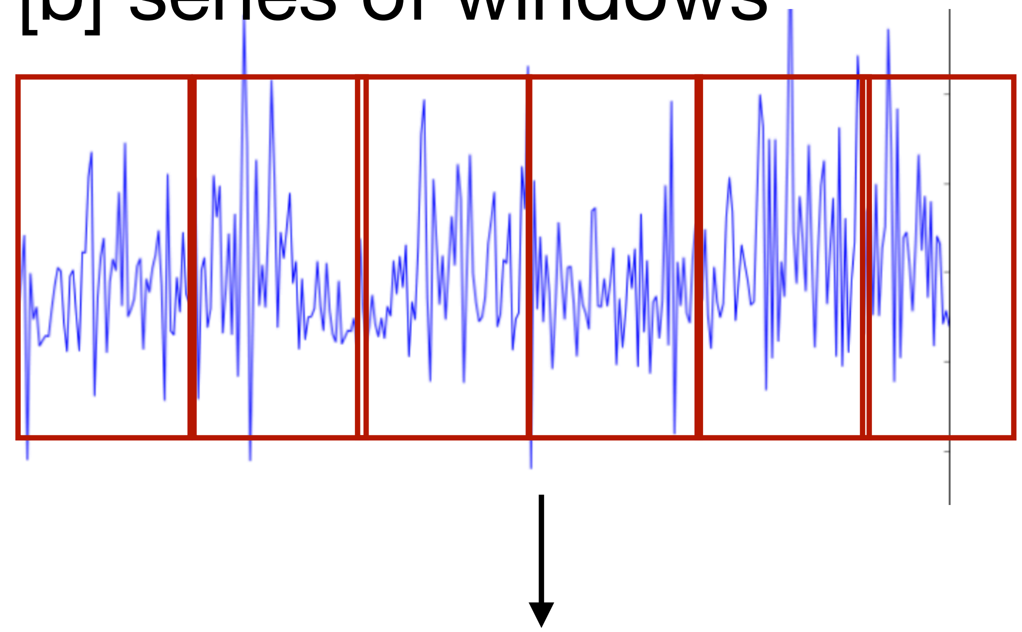
How to test significance in the dFC?

[a] sliding window



samples dependent
from each other

[b] series of windows

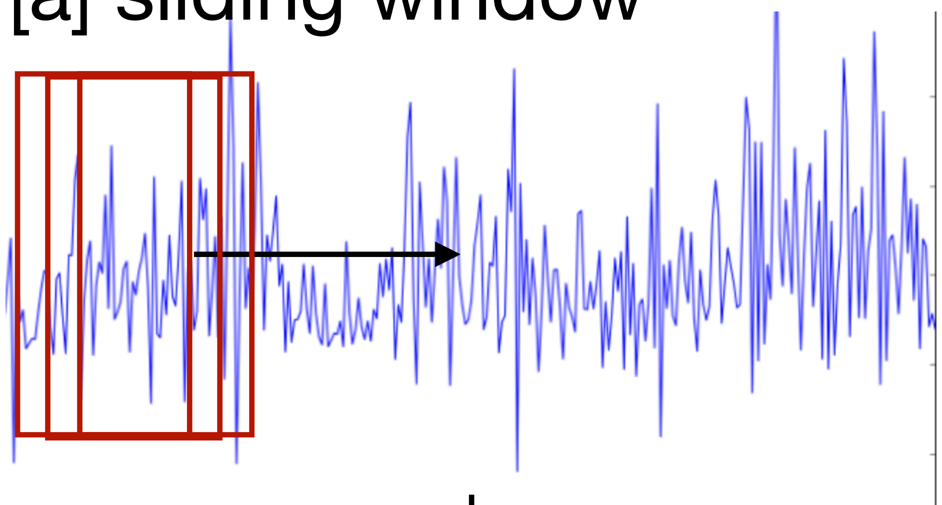


samples independent
from each other

Which method is better? —> Hindriks et al, NeuroImage 2016

How to test significance in the dFC?

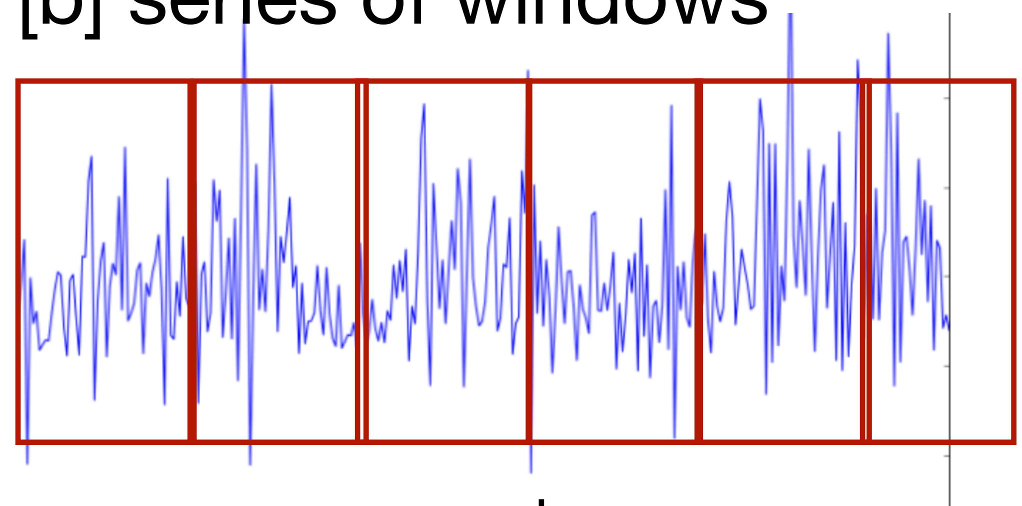
[a] sliding window



samples dependent
from each other

permutation testing

[b] series of windows

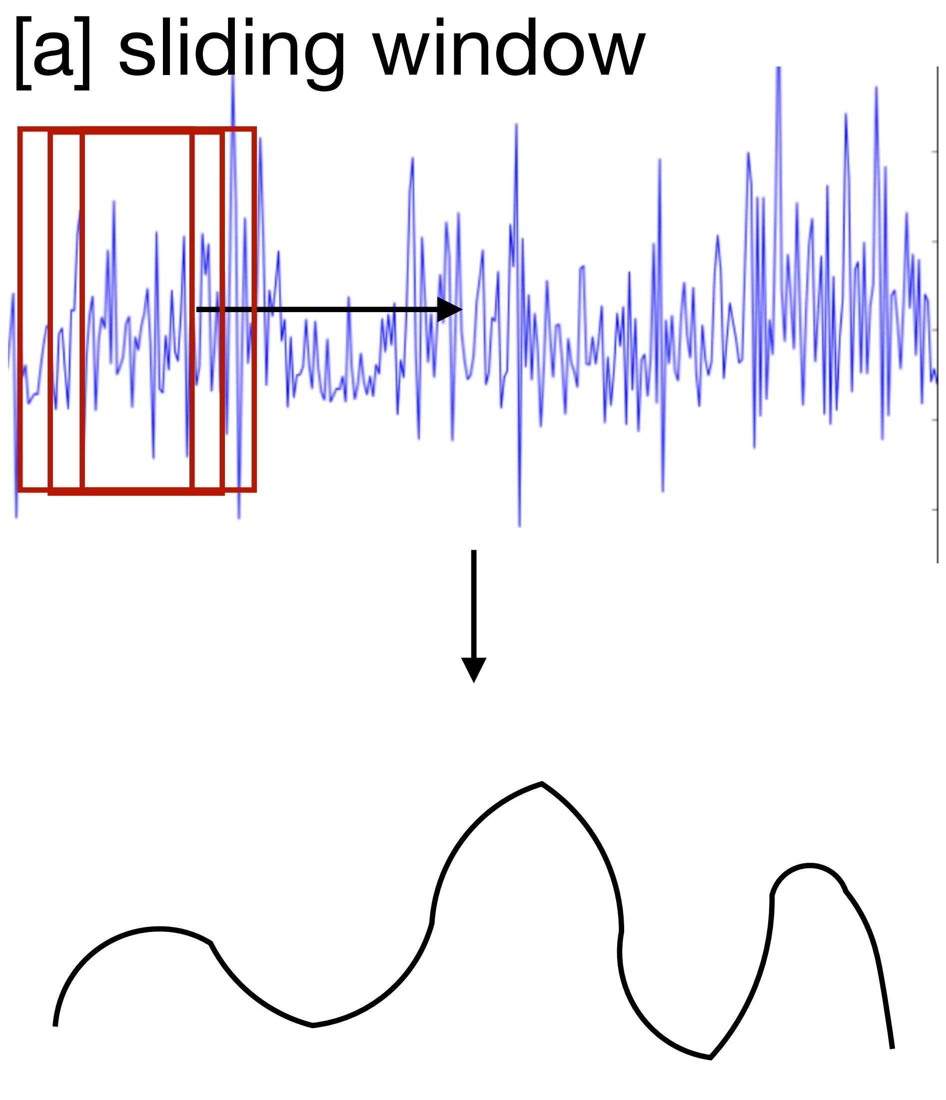


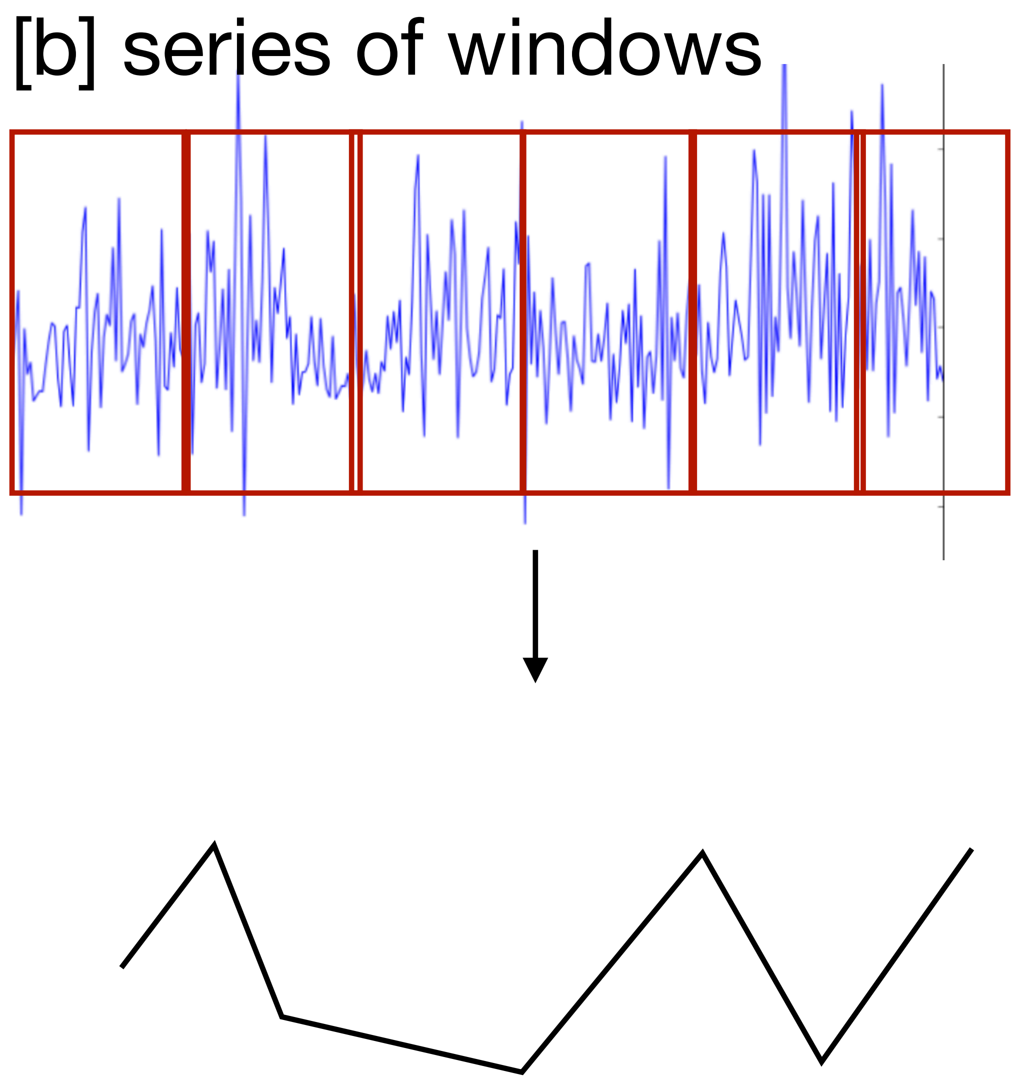
samples independent
from each other

t-test

Which method is better? —> Hindriks et al, NeuroImage 2016

What do we get out of dFC?





How to visualize dFC in Python?

HoloViews ABOUT GALLERY DOCS GITHUB

GETTING STARTED USER GUIDE GALLERY REFERENCE GALLERY API FAQ Search

HoloViews ¶

Stop plotting your data - annotate your data and let it visualize itself.

HoloViews is an open-source Python library designed to make data analysis and visualization seamless and simple. With HoloViews, you can usually express what you want to do in very few lines of code, letting you focus on what you are trying to explore and convey, not on the process of plotting.

For examples, check out the thumbnails below and the other items in the Gallery of demos and apps and the Reference Gallery that shows every HoloViews component. Be sure to look at the code, not just the pictures, to appreciate how easy it is to create such plots yourself!

The Getting-Started guide explains the basic concepts and how to start using HoloViews, and is the recommended way to understand how everything works.

The User Guide goes more deeply into key concepts from HoloViews, when you are ready for further study.

The API is the definitive guide to each HoloViews object, but the same information is available more conveniently via the *hv.help()* command and tab completion in the

Tweets by @HoloViews

HoloViews
@HoloViews

New in GeoViews 1.4: Native GeoPandas support for easy choropleths and interactive colormapping thanks to @BokehPlots

Embed View on Twitter

Rafał Skolasiński

Assignment 3: in iPython notebook

4_dynamic_functional_connectivity.ipynb

you have a simple implementation of dFC. First, reimplement the dFC so that next to [a] the sliding window, you will also have version [b]. You can also experiment with the window types (rectangular versus smooth)

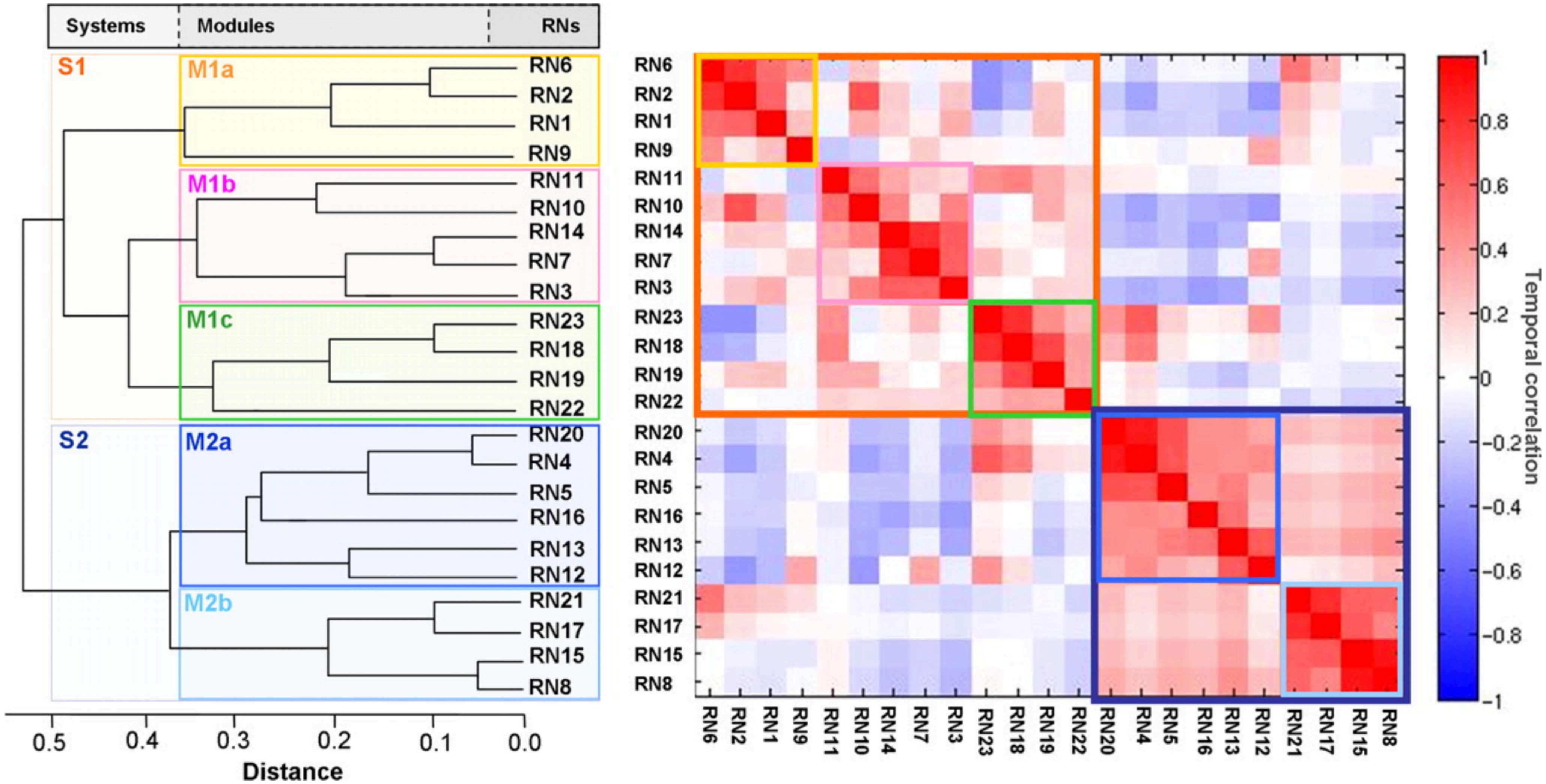
Then, answer the following questions:
[a] how does dFC depend on the window length?
[b] which of the two types of dFC is more informative to you? why?
[c] load the dataset from two different subjects, and check if the dFC dynamics looks similar
[d] load the datasets other than fMRI and compute dFC on this data. Does the dynamics differ between datasets?

[e] in iPython notebook

5_coherence.ipynb

you can find codes for computing a coherence between two time series (from EEG/MEG datasets etc.). Get familiar with these codes, and play around with the synthetic data in the notebook in order to figure out how the coherence works.
[f]* how to operationalize dFC *dynamics*? would you rather go for the variance of dFC or for FFT analysis?
[g]* reimplement dFC so that coherence is computed within each window (coherence instead of correlation). Compute dFC on exemplary EEG and fMRI datasets, and discuss the results with your partner.

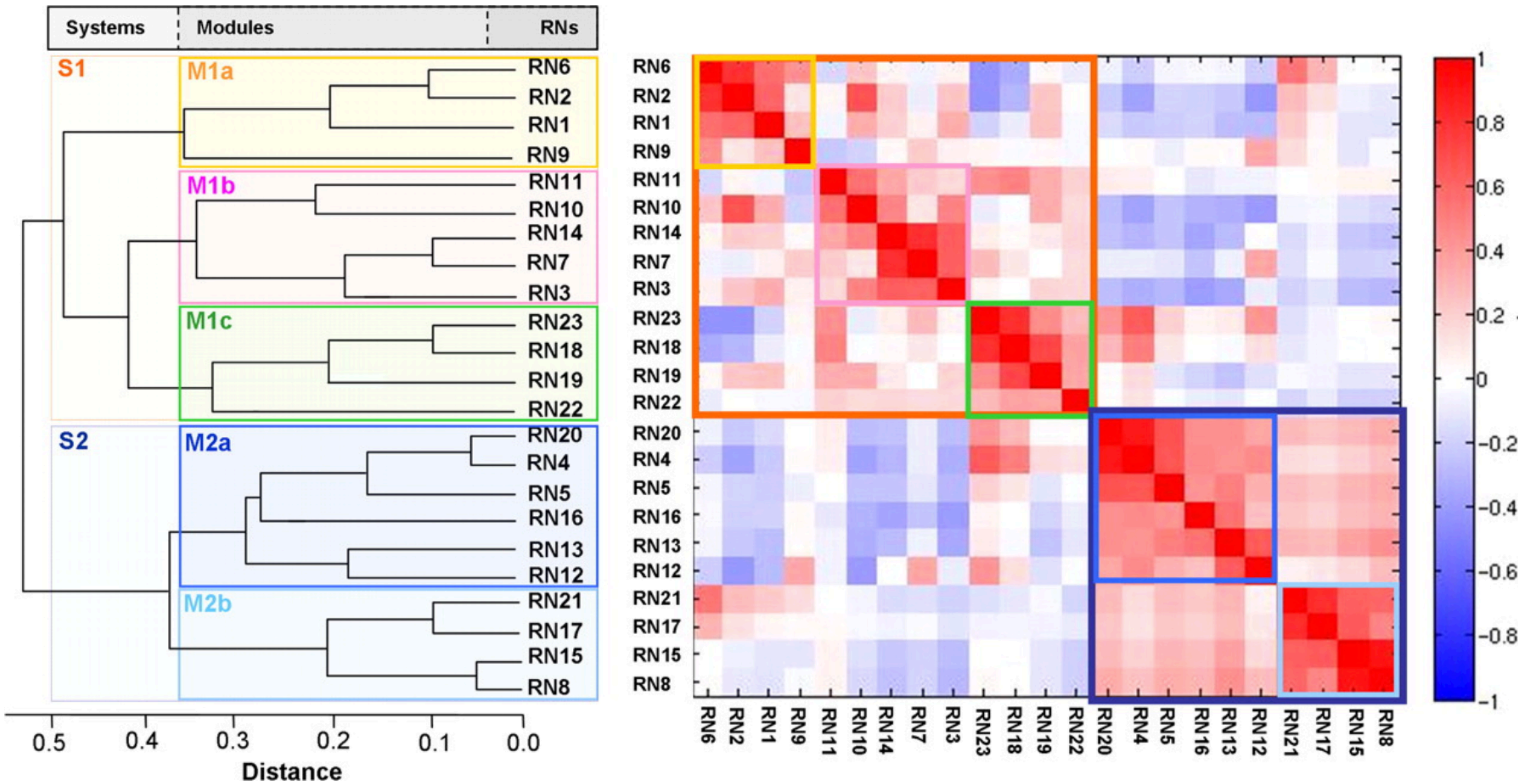
Hierarchical models of functional connectivity



'start from grouping leaves into branches,
the branches into limbs
and eventually into the trunk'

Doucet et al., Journal of Neurophysiology 2011

Hierarchical models of functional connectivity



Ward method

Doucet et al., Journal of Neurophysiology 2011

Hierarchical models of functional connectivity

The Ward method for hierarchical clustering (for N variables):

[1] compute a distance matrix for the dataset* *D*
[2] compress matrix *D* to a vector of all the pairwise interactions $\binom{n}{2}$
[3] launches iterative algorithm: starts out with *N* clusters of size 1 and continues until all the observations are included into one cluster

Ward algorithm looks at cluster analysis as an analysis of variance problem, instead of using distance metrics or measures of association.

In the first step of the algorithm, n - 1 clusters are formed, one of size two and the remaining of size 1. The error sum of squares and r^2 values are then computed. The pair of sample units that yield the smallest error sum of squares, or equivalently, the largest r^2 value will form the first cluster.
Then, in the second step of the algorithm, n - 2 clusters are formed from that n - 1 clusters defined in step 2. These may include two clusters of size 2, or a single cluster of size 3 including the two items clustered in step 1. Again, the value of r^2 is maximized.
Thus, at each step of the algorithm clusters or observations are combined in such a way as to minimize the results of error from the squares or alternatively maximize the r^2 value. The algorithm stops when all sample units are combined into a single large cluster of size n.

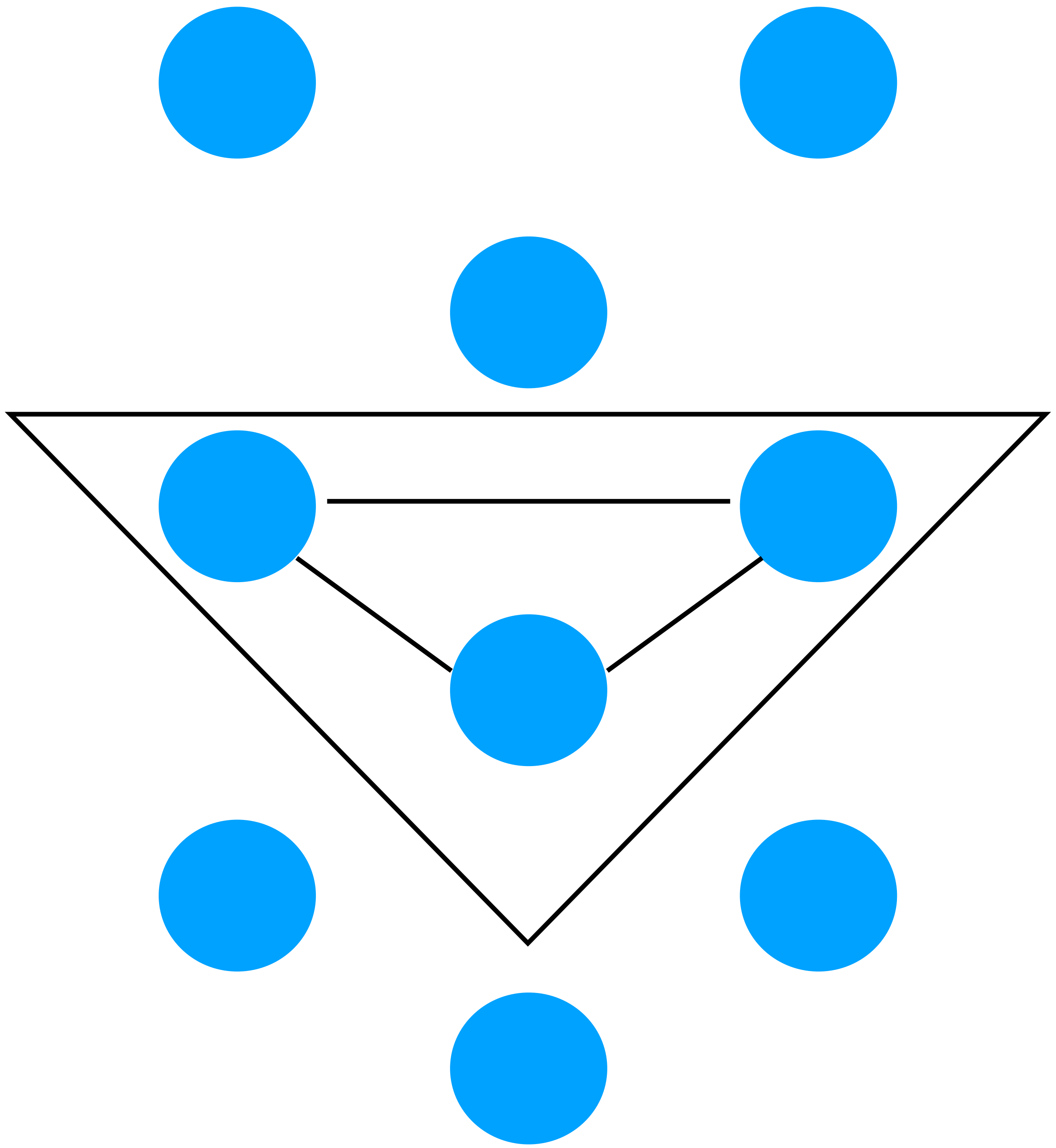
Assignment 4: in iPython notebook

6_hierarchical_models_of_functional_connectivity.ipynb

you will find codes for finding hierarchy in functional connectomes. Get familiar with this notebook, and answer the questions:
[a] is the hierarchy reproducible between subjects?
[b] how do the hierarchical models differ between different datasets?

*Other intriguing topics within the subject-matter of functional connectivity

Higher order statistics part 1: Multiple body interactions



Pearson's r for two nodes:

$$\frac{\mathrm{E}\{(X-\mu_X)(Y-\mu_Y)\}}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}}$$

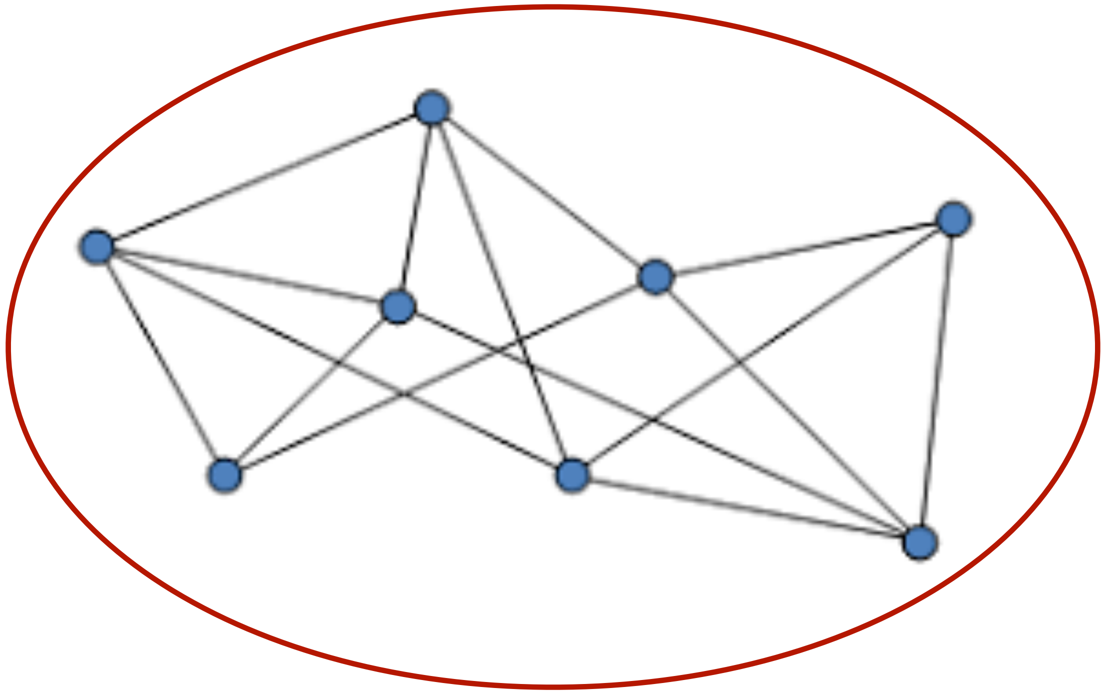
Pearson's r for three nodes:

$$\frac{\mathrm{E}\{(X-\mu_X)(Y-\mu_Y)(Z-\mu_Z)\}}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)\mathrm{Var}(Z)}}$$

Assignment 5*:
Implement this statistic and launch the code on datasets of choice. What do you see?

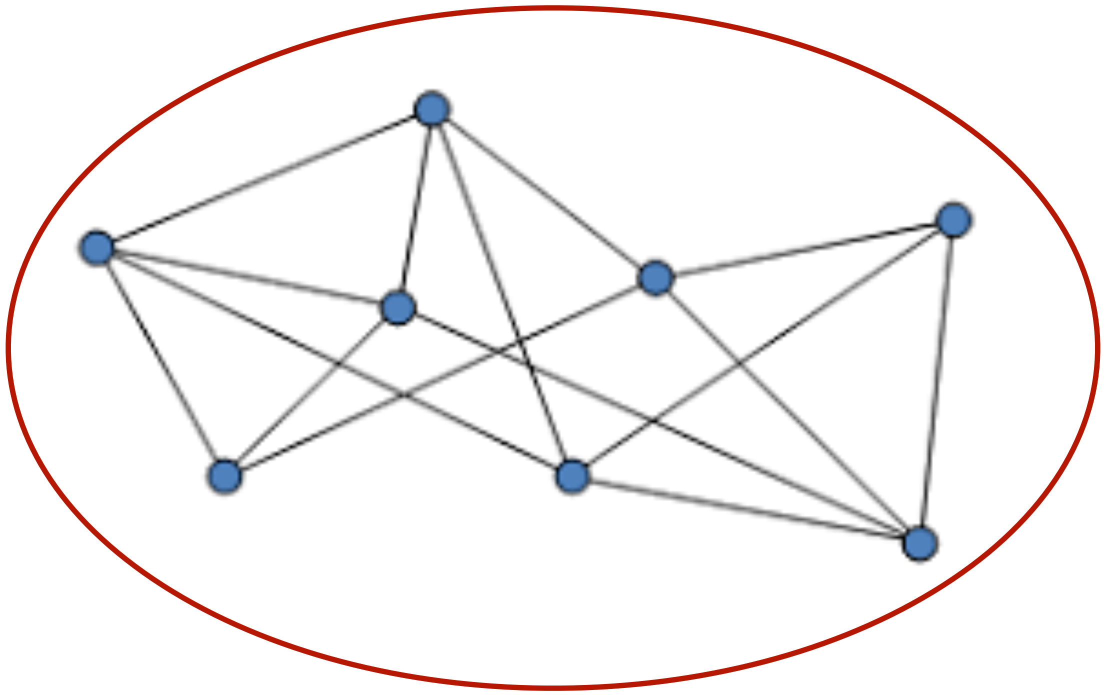
Higher order statistics part 2: Graph theory

Why do we do graph theory?



Higher order statistics part 2: Graph theory

Why do we do graph theory?

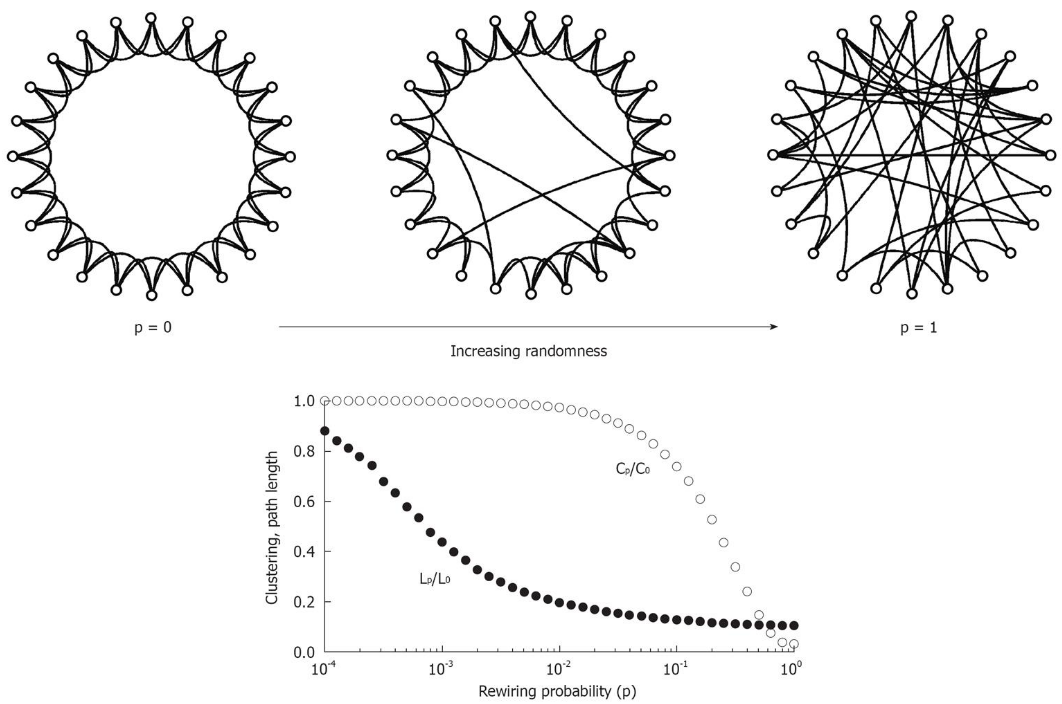


decentralized measures of network functional structure

<u>Pros:</u>
no need for the correction for multiple comparisons
<u>Cons:</u>
decentralized!

Higher order statistics part 2: Graph theory

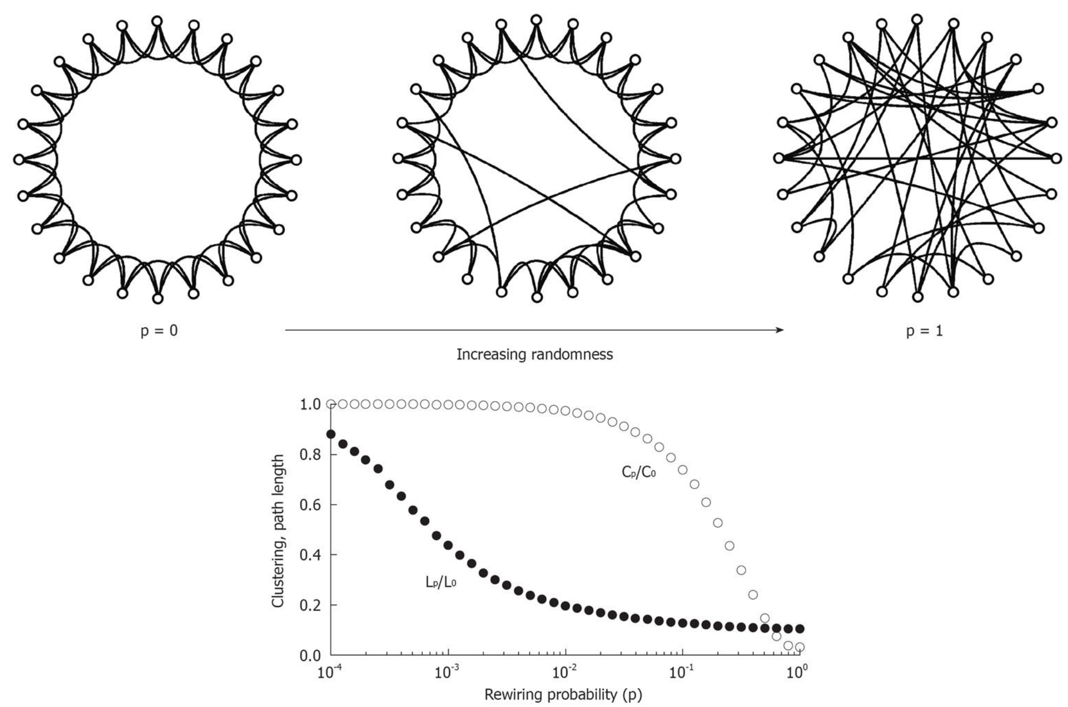
[1] **Small worldness** (a.k.a. global/mean clustering coefficient):
a measure of the degree to which nodes in a graph tend to cluster together



Watts and Strogatz,
Nature 1998

Higher order statistics part 2: Graph theory

[1] **Small worldness** (a.k.a. global/mean clustering coefficient):
a measure of the degree to which nodes in a graph tend to cluster together

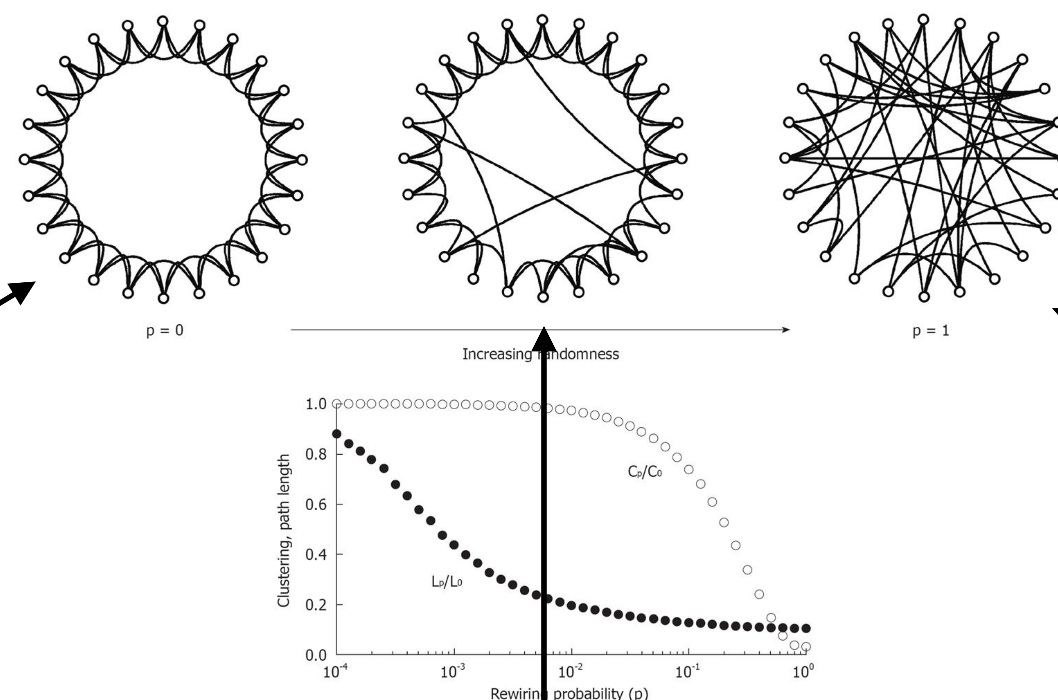


Watts and Strogatz,
Nature 1998

$$C = \frac{3 \times \text{number of triangles}}{\text{number of connected triplets of vertices}} = \frac{\text{number of closed triplets}}{\text{number of connected triplets of vertices}}$$

Higher order statistics part 2: Graph theory

[1] **Small worldness** (a.k.a. global/mean clustering coefficient):
a measure of the degree to which nodes in a graph tend to cluster together



lattice

Erdos-Renyi random network

Watts-Strogatz small-world network

Watts and Strogatz, Nature 1998

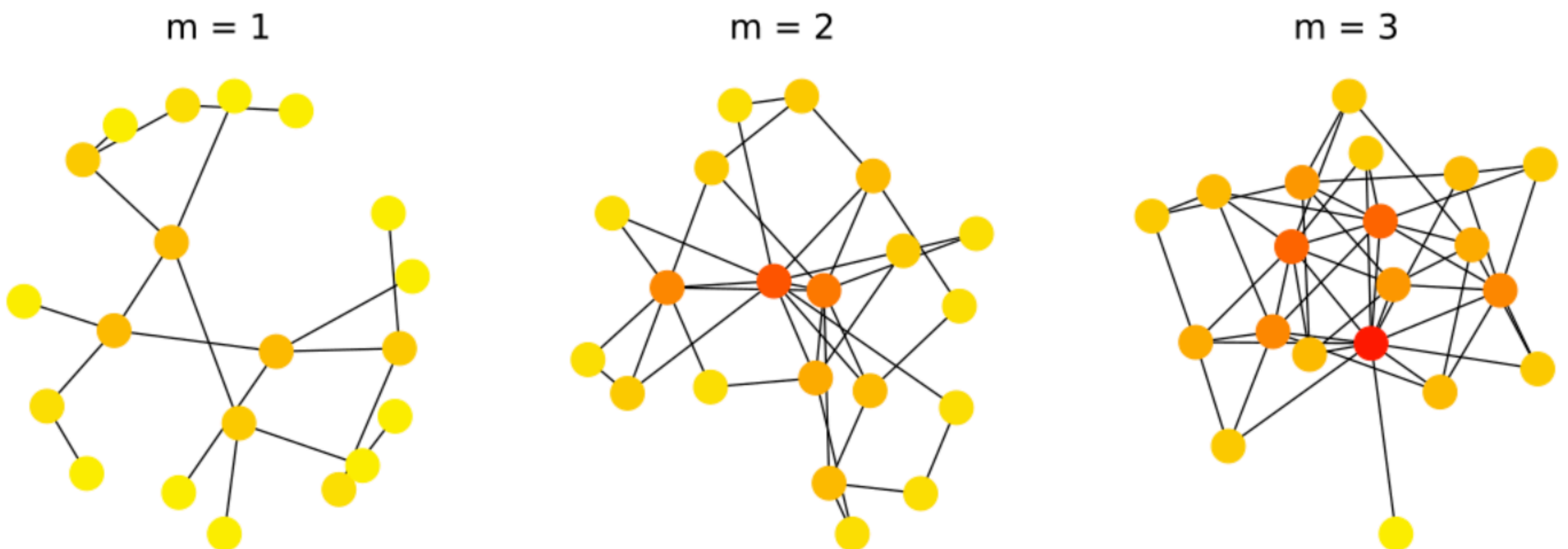
$$C = \frac{3 \times \text{number of triangles}}{\text{number of connected triplets of vertices}} = \frac{\text{number of closed triplets}}{\text{number of connected triplets of vertices}}$$

Higher order statistics part 2: Graph theory

[1] **Small worldness** (a.k.a. global/mean clustering coefficient):
a measure of the degree to which nodes in a graph tend to cluster together

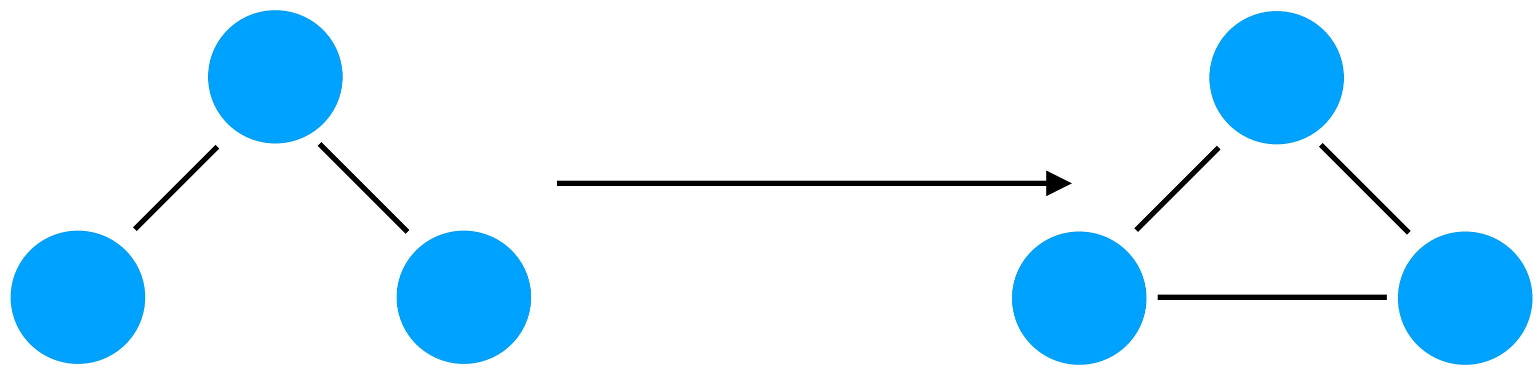
There are multiple possible algorithms for generating a small-world network.
A popular alternative to Watts-Strogatz algorithm is Barabasi-Albert algorithm:

a ‘preferential attachment’ mechanism:



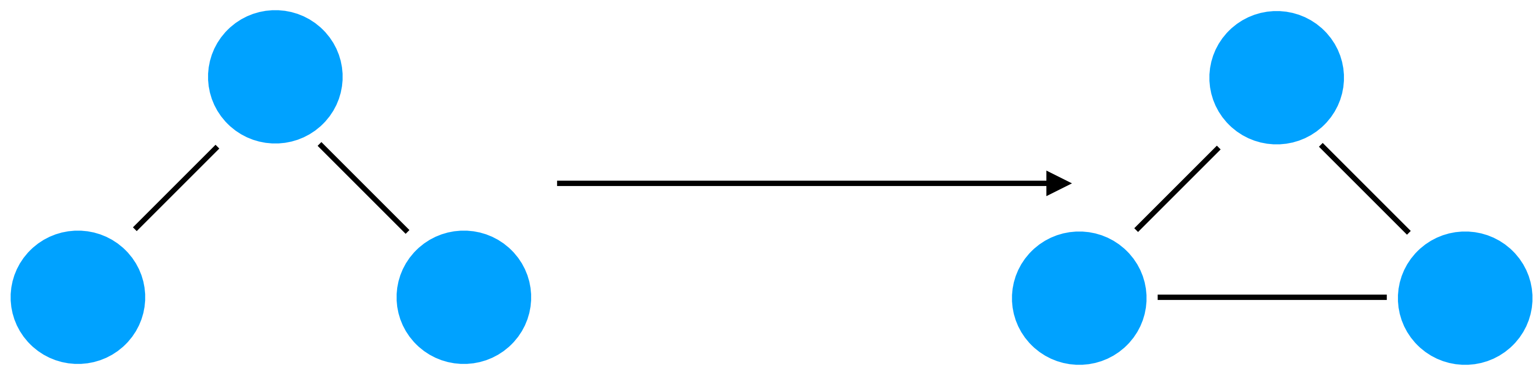
Higher order statistics part 2: Graph theory

[2] **Transitivity:**
the extent to which the relation that relates two nodes in a network that are connected by an edge is transitive



Higher order statistics part 2: Graph theory

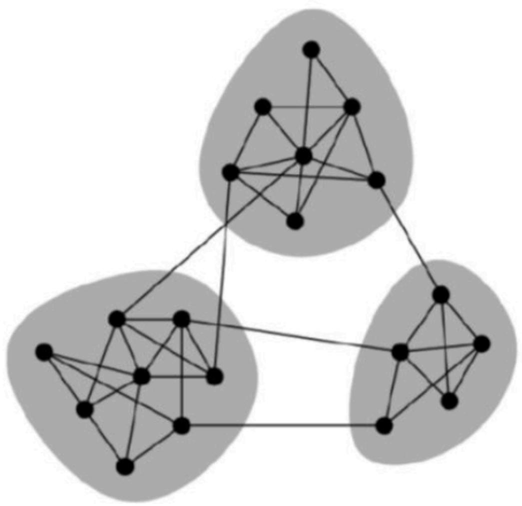
[2] **Transitivity:**
the extent to which the relation that relates two nodes in a network that are connected by an edge is transitive



It is very rare in real networks, since it implies that each component is a clique, meaning that each pair of reachable nodes in the graph would be connected by an edge

Higher order statistics part 2: Graph theory

[3] **Modularity**
(or community detection)



For a particular division of the network into two groups 1 and 2, modularity is defined as:

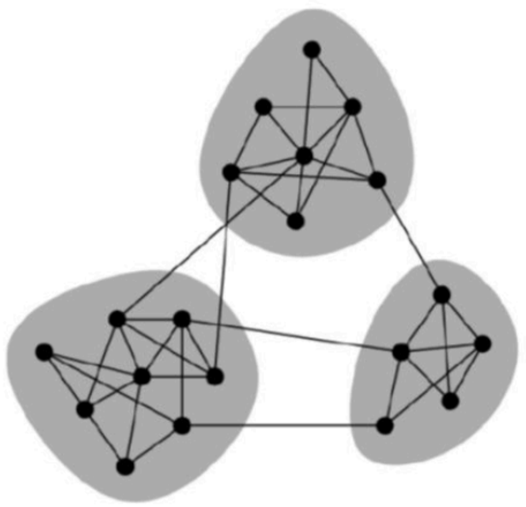
$$Q = \frac{1}{4m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) (s_i s_j + 1) = \frac{1}{4m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) s_i s_j,$$

where
A_{ij} are the elements of the connectivity (or adjacency) matrix
m - the total number of edges in the network
k_i - a degree of node *i*
s_i - spin of node *i* (1 if it belongs to group 1, and -1 if it belongs to group 2)

Newman, M. E. J. PNAS, 2006

Higher order statistics part 2: Graph theory

[3] **Modularity**
(or community detection)



For a particular division of the network into two groups 1 and 2, modularity is defined as:

$$Q = \frac{1}{4m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)(s_i s_j + 1) = \frac{1}{4m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)s_i s_j,$$

In order to compute modularity,
one needs to find the optimal partition
that maximizes Q

Multiple algorithms:
in this tutorial, we will use Louvain method

where

A_{ij} are the elements of the connectivity (or adjacency) matrix

m - the total number of edges in the network

k_i - a degree of node *i*

s_i - spin of node *i* (1 if it belongs to group 1, and -1 if it belongs to group 2)

Newman, M. E. J. PNAS, 2006

Higher order statistics part 2: Graph theory

Assignment 6*: in iPython notebook

7_graph_theory.ipynb

you will find the codes for computing few basic graph theoretical measures.

[1] Get familiar with this notebook, play around with the parameters for each of the quantities, and answer the questions:
[a] what can you learn about the network functional structure from each one of these quantities?
[b] can you give examples of a small-world network that has low modularity, and highly modular network that has low small worldness?
[c] what is the red lobster network? (no idea about this one…)

[2] Launch these codes on the datasets of choice (plug real datasets in place of random graph G) and answer the questions:
[a] which network features are data specific, and which of them are transferrable between datasets?
[b] is this true that small-world networks are highly prevalent in biological systems? Can you make such claims about modularity or transitivity?

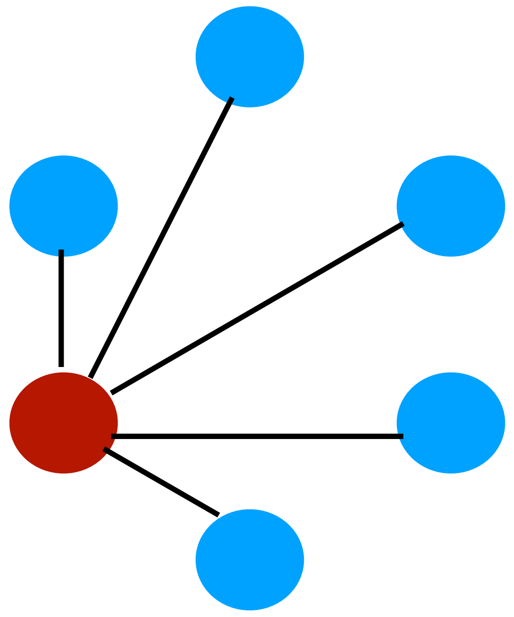
Food for thought:
Functional connectivity research vs prediction studies

<u>functional connectivity:</u>
do the two variables correlate?

<u>prediction study:</u>
you can predict a value of one variable from a value of another variable
(usually because these two variables correlate)

Linear Discriminant Analysis finds coefficients a_i such that:

$$v = \sum a_i v_i$$



<u>Conclusion</u>: these two research problems are mathematically close to each other

<u>A research question</u>:
How about applying Linear Discriminant Analysis to do a functional connectivity research?
Can the activity in one node of the network be predicted from activity in other nodes in the network?

Thank you for your attention!