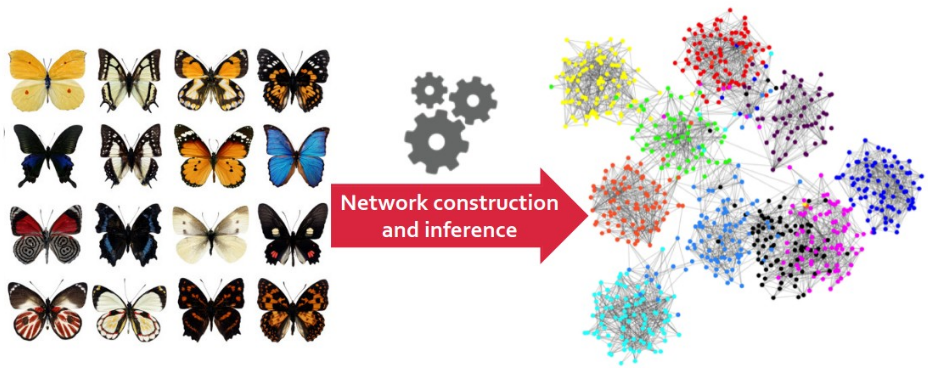
Introduction to the Analysis and Visualisation of Complex Networks

Pedro Ribeiro
(DCC/FCUP & CRACS/INESC-TEC)

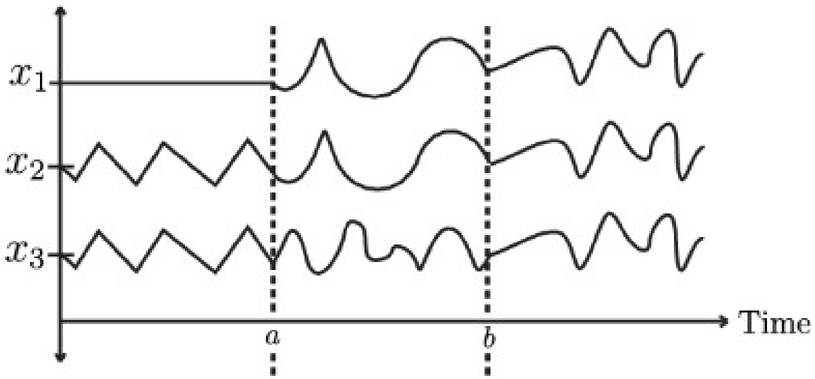


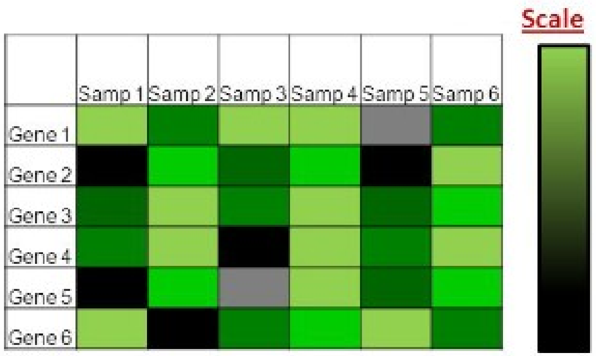
(this part includes some slides heavily based on material from Jure Leskovec @ Stanford University)

Associação Portuguesa de Classificação e Análise de Dados

Network Construction

Raw Data are often not networks





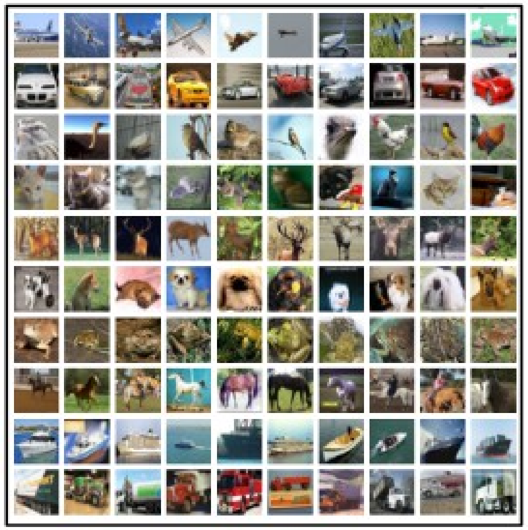

[–] Affable_Nitwit 559 points 15 hours ago
NOW That's What I Call Music. We're on 53, everybody. I checked. Online.
permalink save report give gold reply

[–] AggressiveToothbrush 165 points 15 hours ago
For the record, as of May 4th, we're on 54.
http://en.wikipedia.org/wiki/Now_That%27s_What_I_Call_Music!_54_%28U.S._series%2
permalink save parent report give gold reply

[–] summitorother 224 points 15 hours ago
Speak for yourselves, the UK is on 90.
permalink save parent report give gold reply

[–] anideaguy 206 points 10 hours ago
TIL 54 is 90 in metric.
permalink save parent report give gold reply

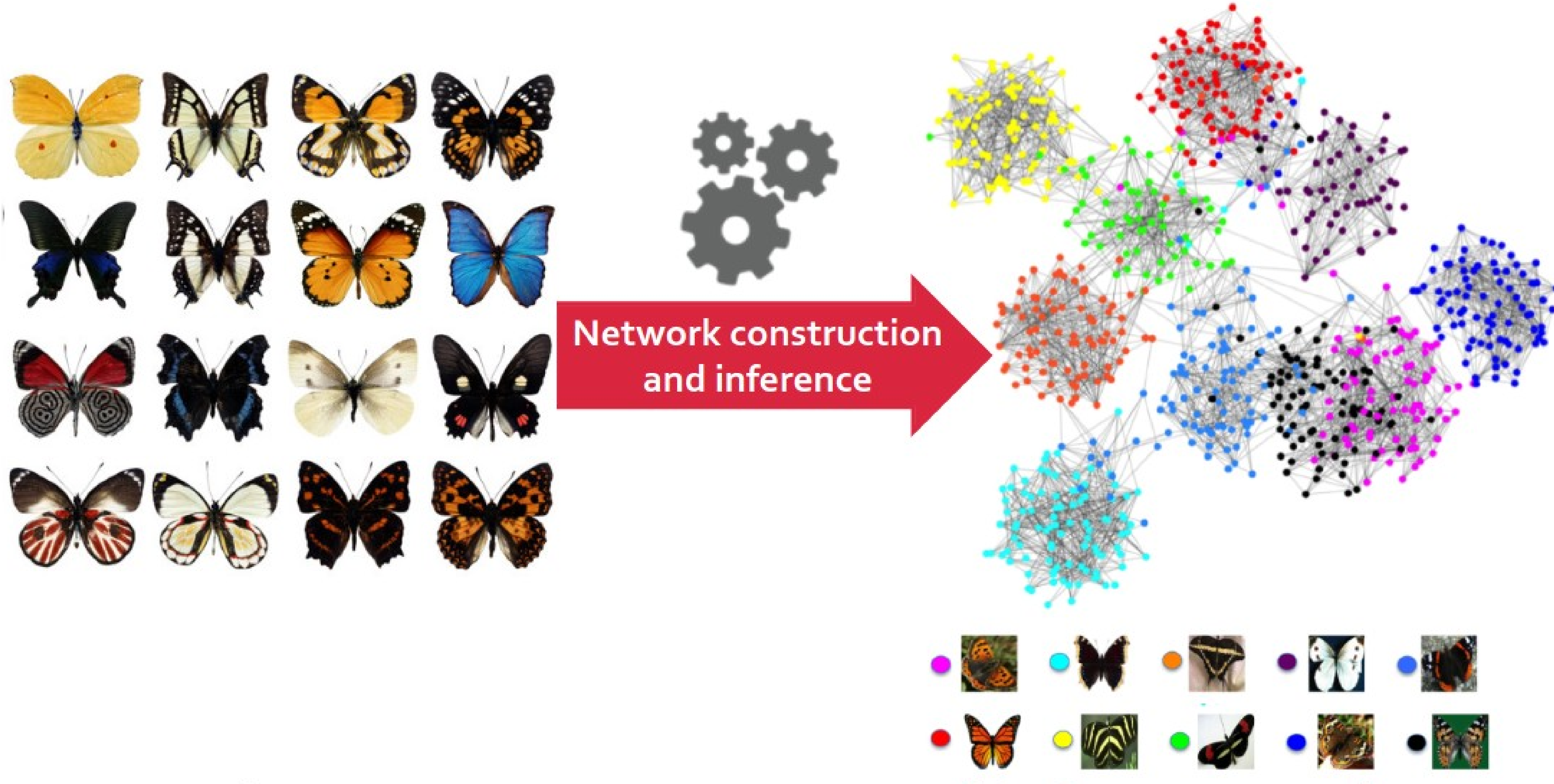
[–] LumberCockSucker 11 points 3 hours ago
The Uk uses imperial measurements
permalink save parent report give gold reply



	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1

Feature matrices, relationship tables, time series, document corpora, image datasets, etc.

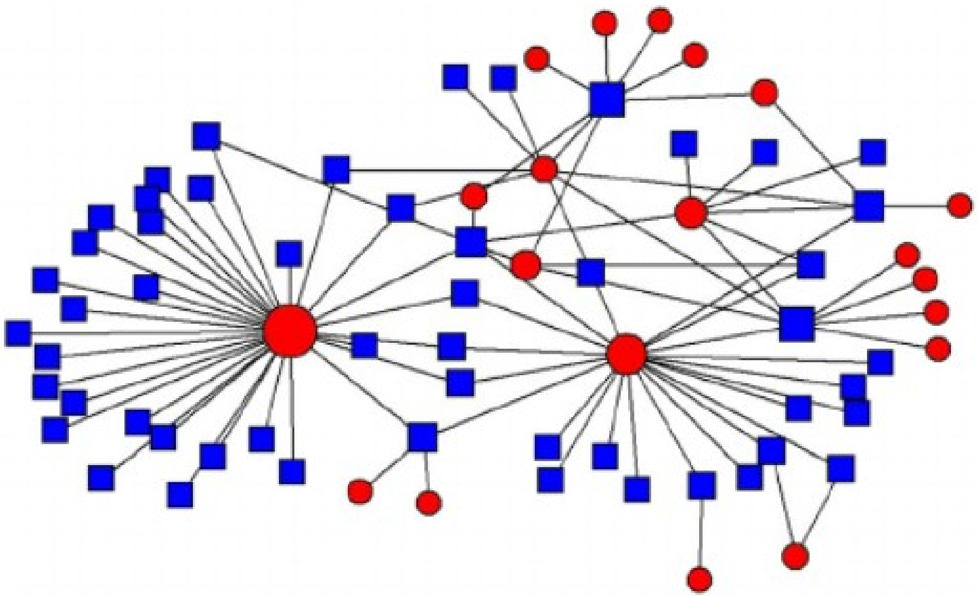
How to Construct Networks?



Bi-partite and K-partite Networks

- Most of the time, when we create a network, all nodes represent **objects of the same type:**
  - People in social nets, bus stops in route nets, genes in gene nets

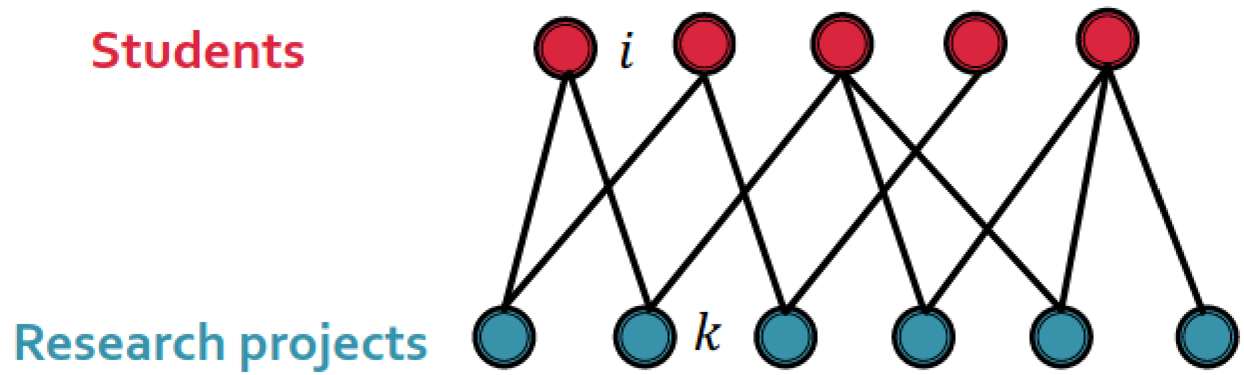
- **Multi-partite networks** have **multiple types of nodes**, where edges exclusively go from one type to the other:
  - **2-partite student net:** Students <-> Research projects
  - **3-partite movie net:** Actors <-> Movies <-> Movie Companies



Network on the left is a social bipartite network. **Blue squares** stand for people and **red circles** represent organizations

One-Mode Projections: Example

- **Example:** Bipartite student-project network:
  - Edge: Student i works on research project k

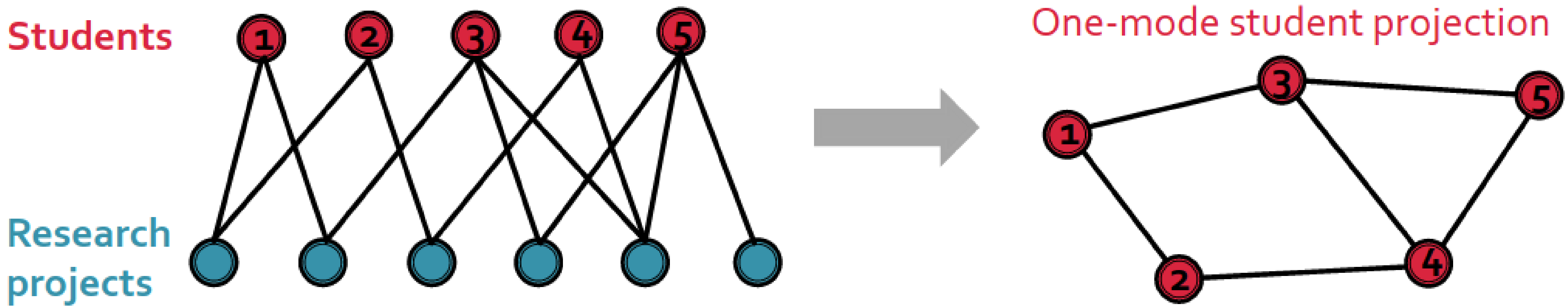


- Two network projections of student-project network:
  - **Student network:** Students are linked if they work together in one or more projects
  - **Project network:** Research projects are linked if one or more students work on both projects

- **In general:** K-partite network has K one-mode network projections

One-Mode Projections: Example

- **Example:** Projection of **bipartite student-project network** onto the **student mode**:



- **Consider students 3, 4, and 5 connected in a triangle:**
  - Triangle can be a result of:
    - **Scenario #1:** Each pair of students work on a different project
    - **Scenario #2:** Three students work on the same project
  - One-mode network projections **discard some information:**
    - Cannot distinguish between #1 and #2 just by looking at the projection

Constructing One Mode Projections

- Construct network projections by applying **node similarity** measures

- Two example similarity measures:
  - **Common neighbors:** #(shared neighbors of nodes)
    - Student network: *i* and *j* and are linked if they work together in *k* or more projects
    - Project network: *i* and *j* are linked if *k* or more students work on both projects
  - **Jaccard index:**
    - Ratio of shared neighbors in the complete set of neighbors for 2 nodes

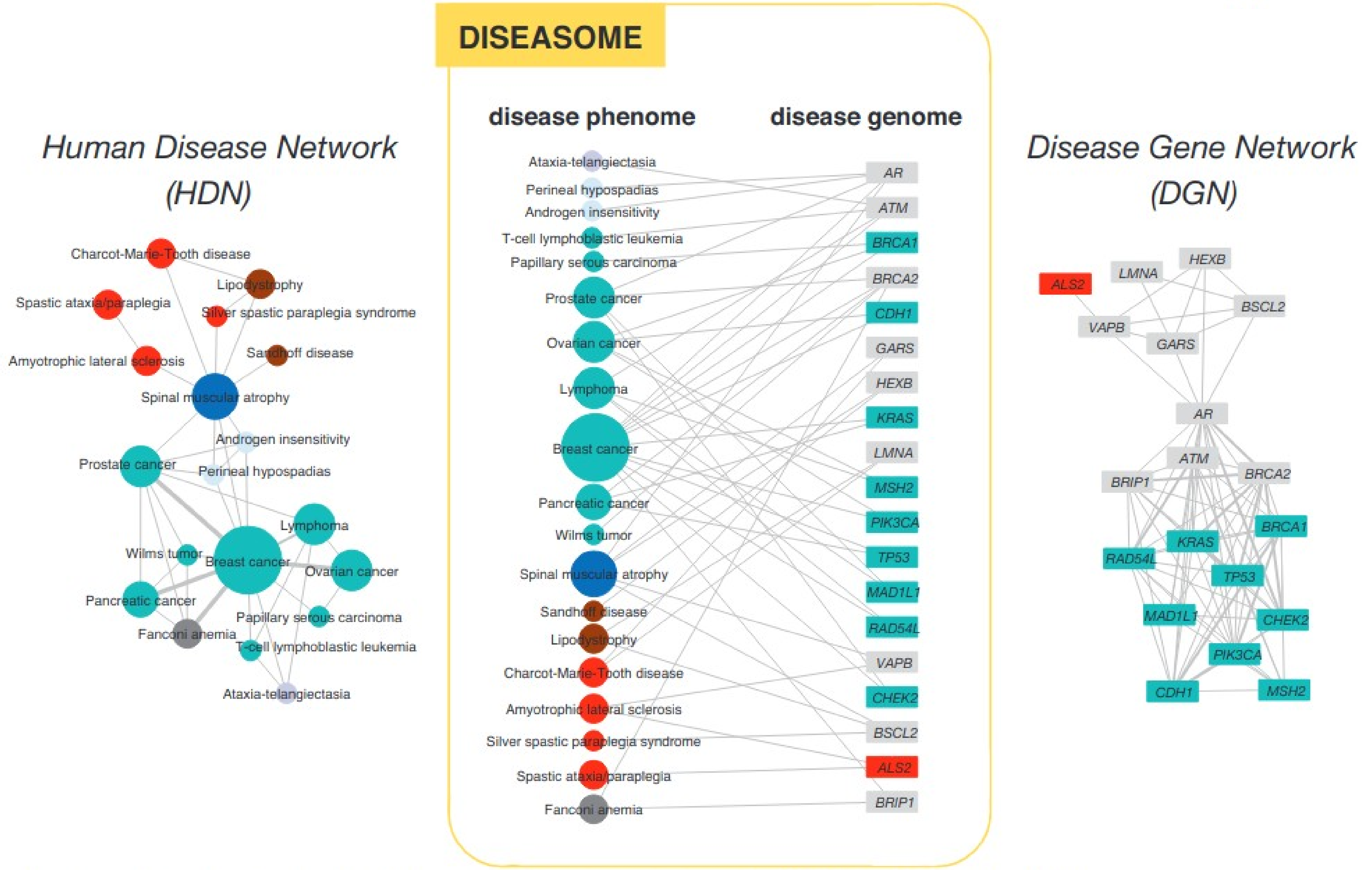
Constructing One Mode Projections

- Construct network projections by applying **node similarity** measures

- Two <u>example similarity measures</u>:
  - **Common neighbors:** #(shared neighbors of nodes)
    - Student network: *i* and *j* and are linked if they work together in *k* or more projects
    - Project network: *i* and *j* are linked if *k* or more students work on both projects
  - **Jaccard index:**
    - Ratio of shared neighbors in the complete set of neighbors for 2 nodes

- Many possible variations, such as:
  - **Weight** representing strength of similarity

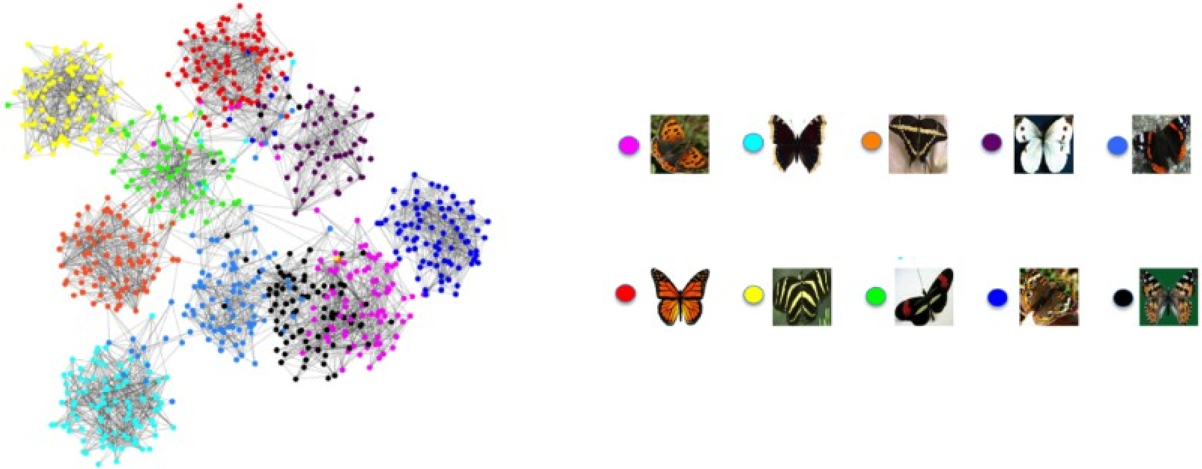
Example: Human Disease Network



Kwang-Il Goh *et al.*, The human disease network. *PNAS*, 104:21, 2007.

Sets of Objects as a Network

- Even in a dataset with no explicit graph structure, we can use similarity measures to produce a network:
  - **K-nearest neighbor graph (K-NNG)** for a set of objects V is a directed graph with vertex set V:
    - Edges from each $v \in V$ to its K most similar objects in V under a given similarity measure:
      - *e.g.*, Cosine similarity for text
      - *e.g.*, l_2 distance of CNN-derived features for images



Times Series as Networks

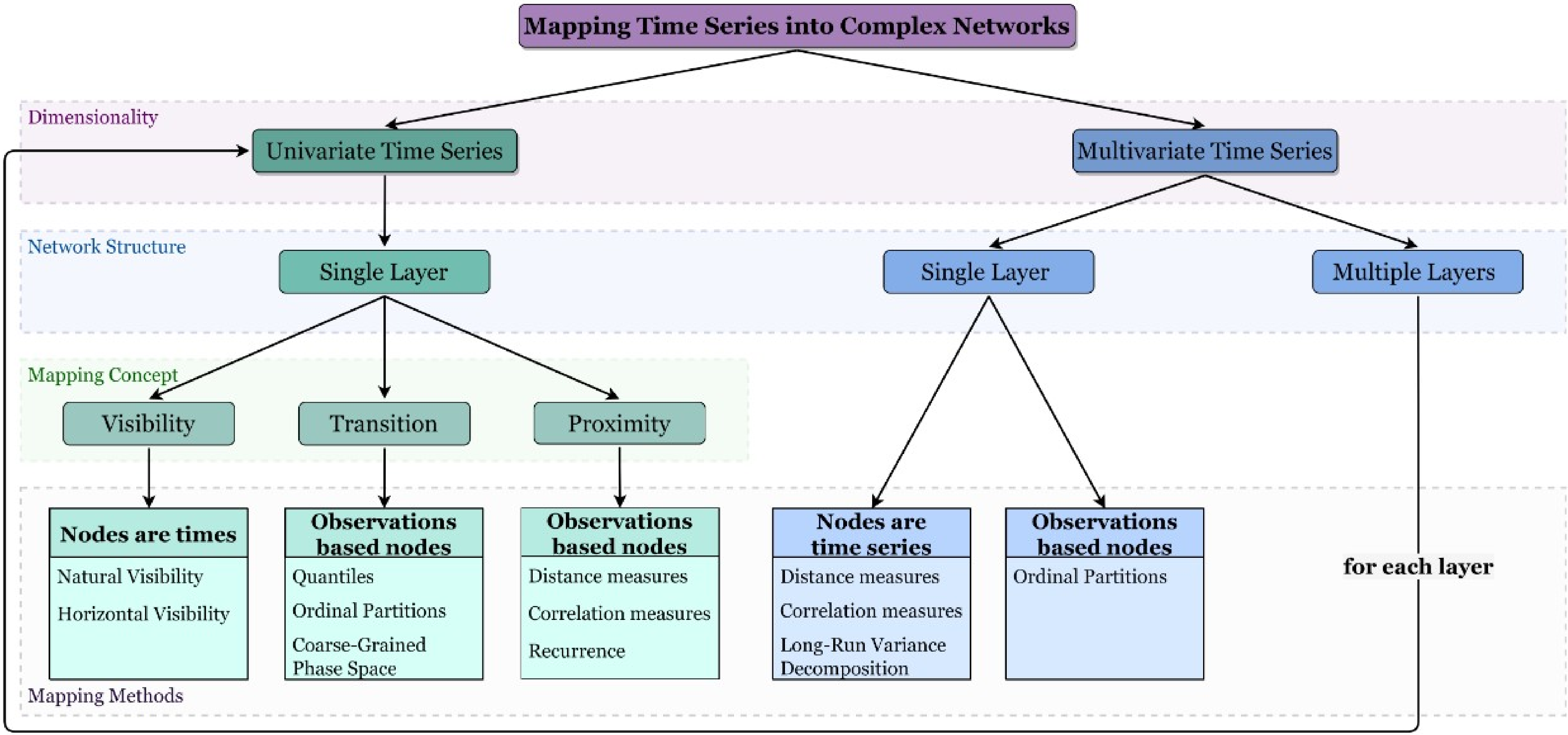
Time series analysis via network science: Concepts and algorithms

Vanessa Freitas Silva, Maria Eduarda Silva, Pedro Ribeiro, Fernando Silva

First published: 01 March 2021 | https://doi.org/10.1002/widm.1404 | Citations: 1

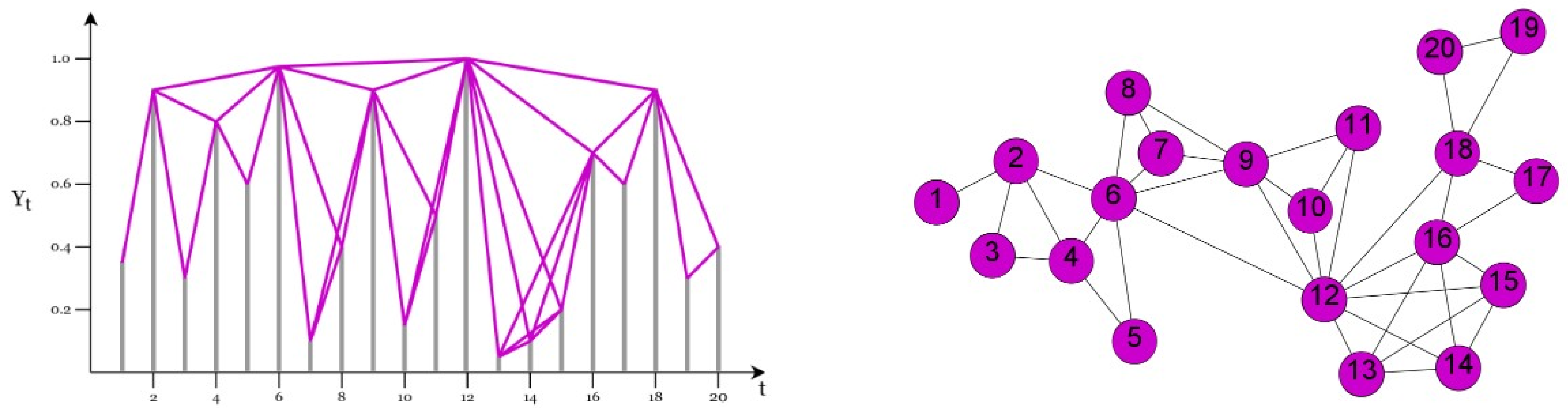
WIREs DATA MINING AND KNOWLEDGE DISCOVERY

Advanced Review



Times Series as Networks

- Ex: Visibility Graphs



(a) Toy time series

(b) NVG

Figure 4: On the left side, we present the plot of a toy time series and, on the right side, the network generated by the natural visibility algorithm. The purple lines in the time series plot represent the lines of visibility (and hence the edges of the graph) between data points.

Times Series as Networks

- Ex: Quantile Graphs

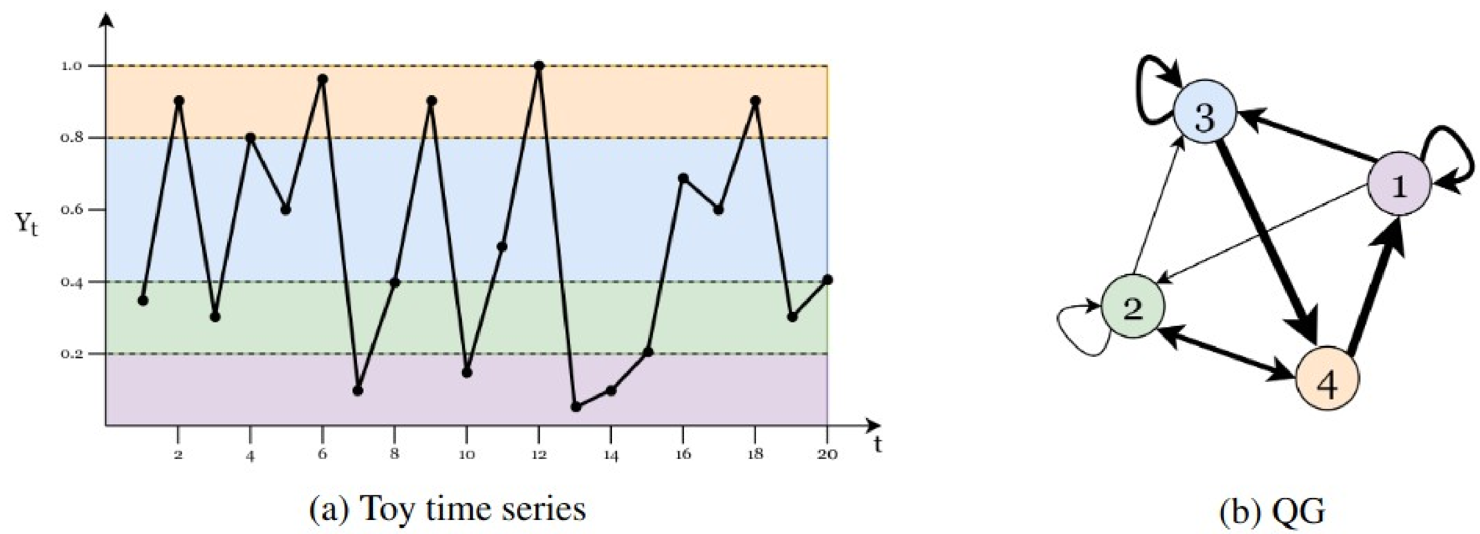


Figure 8: Illustrative example of the quantile graph algorithm for $Q = 4$. On the left panel we present the plot of a toy time series and on the right panel the network generated by the quantile graph algorithm. The different colors in the time series plot represent the regions corresponding to the different quantiles. In the network, edges with larger weights represented by thicker lines correspond to the repeated transitions between quantiles.

Times Series as Networks

- Ex: Correlation Networks

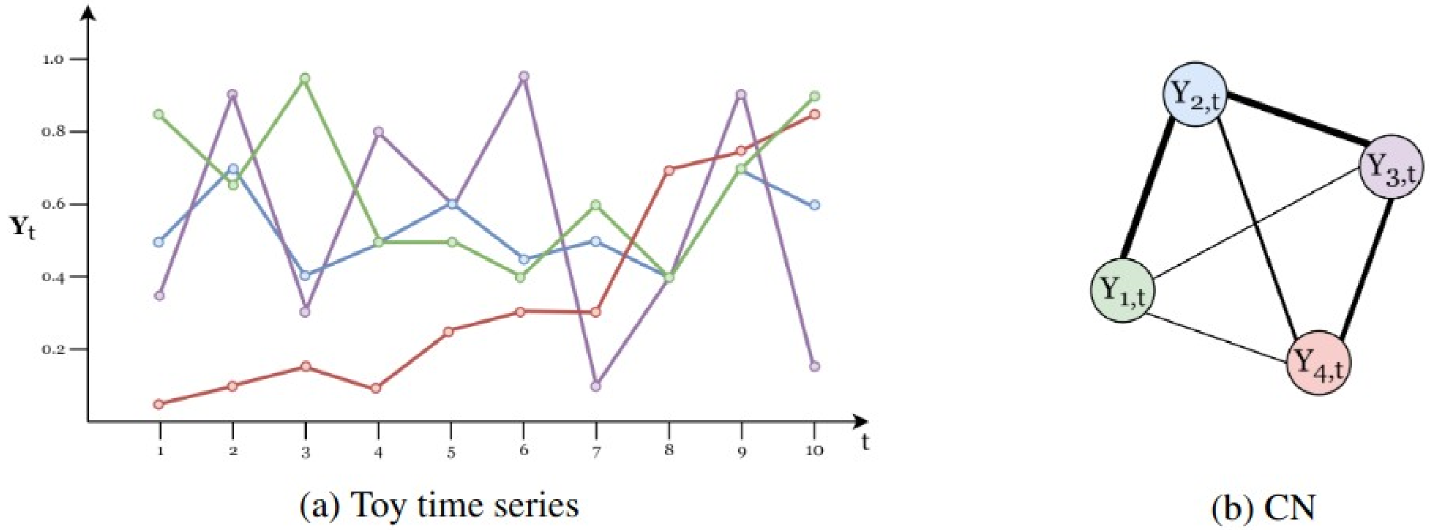


Figure 10: Illustrative example of the correlation network algorithm. On the left side we present the plot of a toy multivariate time series and on the right side the network generated by the correlation algorithm (using contemporaneous cross-correlation). The different colors represent the time series. Higher correlation values result in edges in the network with larger weights represented by thicker lines.

Example Application

Data Mining and Knowledge Discovery (2022) 36:1062–1101
https://doi.org/10.1007/s10618-022-00826-3

Novel features for time series analysis: a complex networks approach

Vanessa Freitas Silva[1] · Maria Eduarda Silva[2] · Pedro Ribeiro[1] · Fernando Silva[1]

Example Application

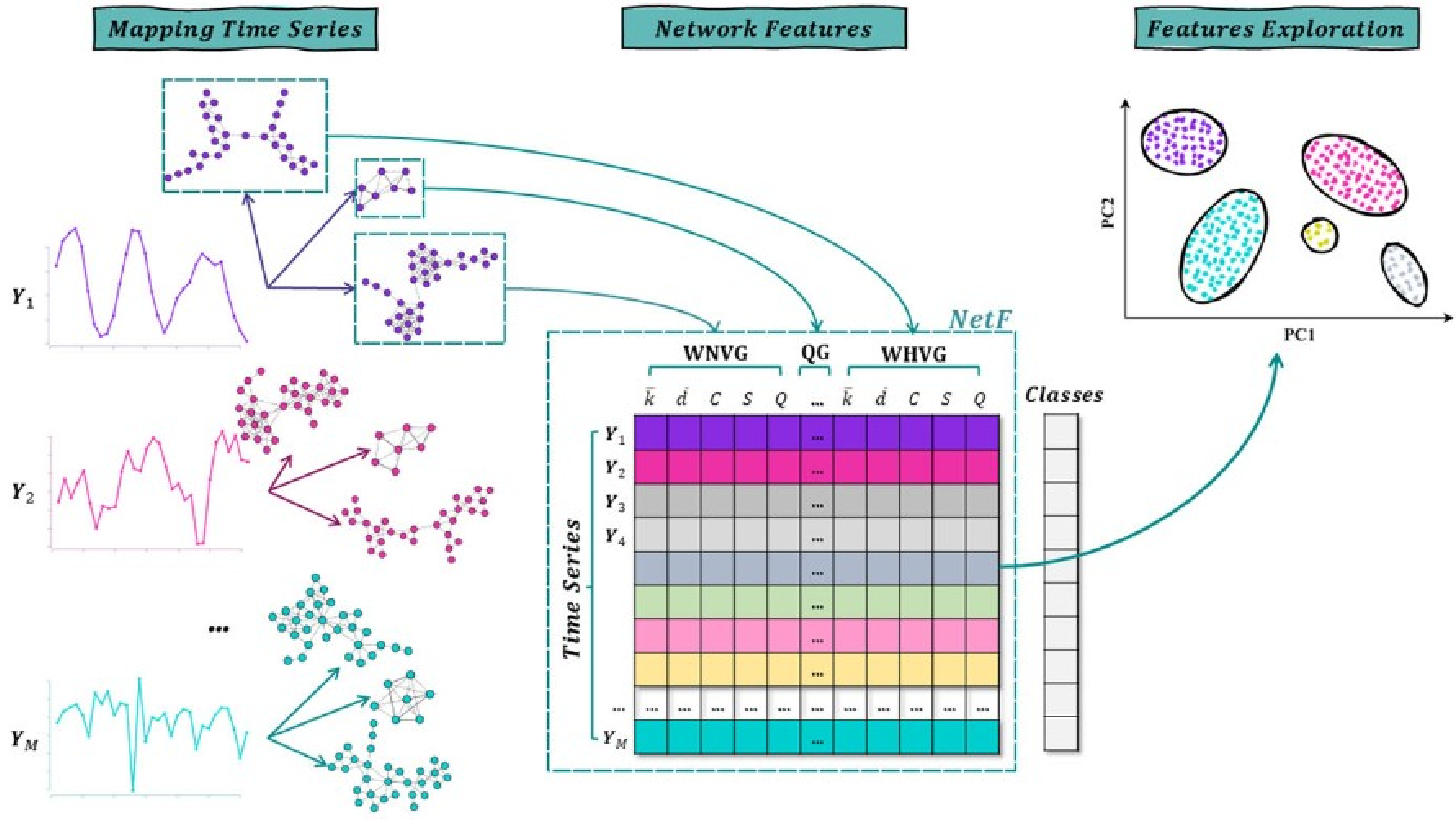
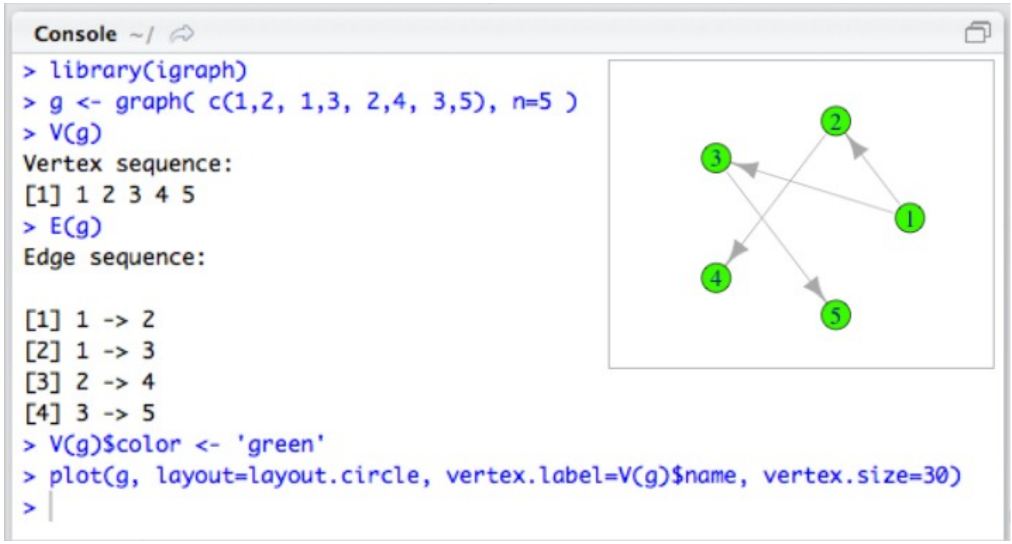


Fig. 1 Schematic diagram of the network based features approach to time series mining tasks

Network Analysis (programming) Toolkits

igraph

- Igraph - https://igraph.org/ igraph
  - igraph is a collection of network analysis tools with the emphasis on efficiency, portability and ease of use
  - Open source and free, can be programmed in R, Python and C/C++
  - Runs on Windows, Linux and MacOS
  - Relies on other libraries for visualization



igraph Documentation

- Official documentation:
  - R: https://igraph.org/r/doc/
  - Python: https://igraph.org/python/doc
  - C: https://igraph.org/c/doc/

- Tutorials:
  - R: https://kateto.net/netscix2016
  - Python: https://igraph.org/python/versions/latest/tutorial.html
    https://igraph.org/python/versions/latest/gallery.html
  - C: https://igraph.org/c/doc/igraph-Tutorial.html

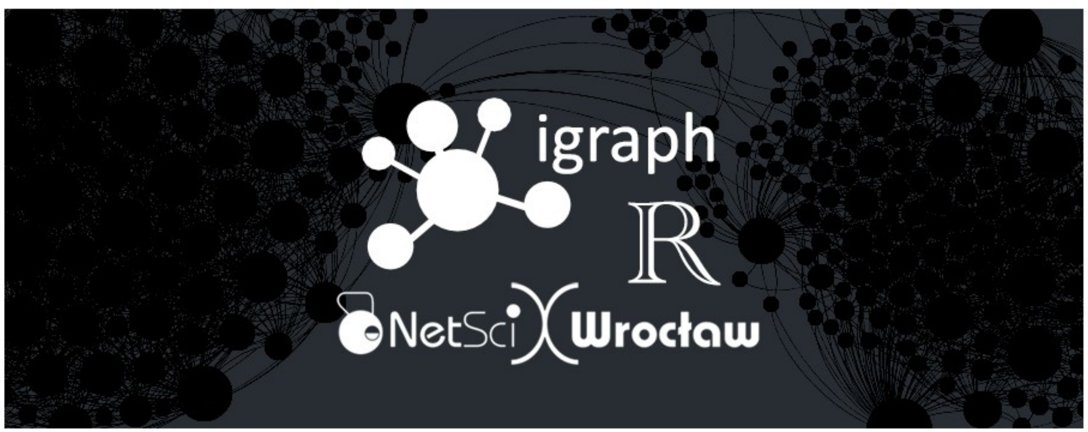
igraph demo

- Tutorial from NetSciX 2016
  (Katherine Ognyanova)
  https://kateto.net/netscix2016

Network Analysis and Visualization with R and igraph

Katherine Ognyanova, www.kateto.net

NetSciX 2016 School of Code Workshop, Wroclaw, Poland



Note: You can download all workshop materials here, or visit kateto.net/netscix2016.
This tutorial covers basics of network analysis and visualization with the R package igraph (maintained by Gabor Csardi and Tamas Nepusz). The igraph library provides versatile options for descriptive network analysis and visualization in R, Python, and C/C++. This workshop will focus on the R implementation. You will need an R installation, and RStudio. You should also install the latest version of `igraph` for R:

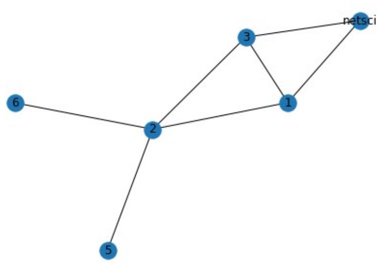
```
install.packages("igraph")
```

DEMO!

NetworkX

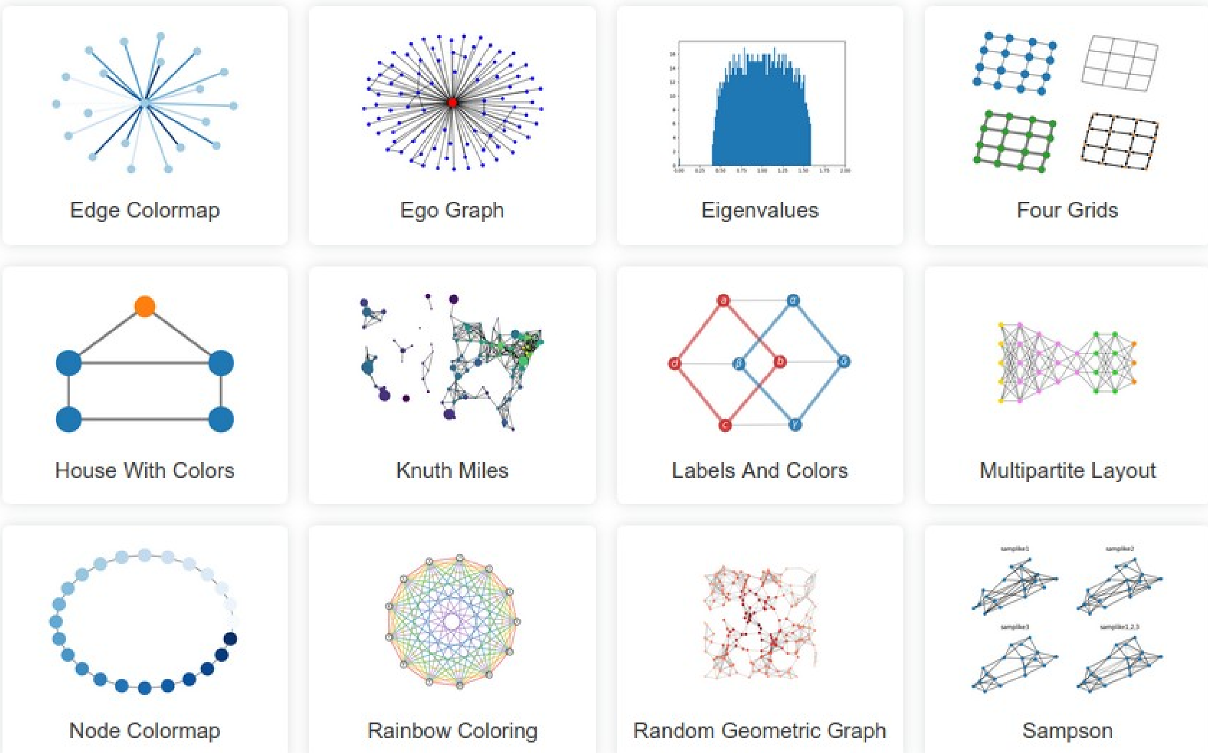
- NetworkX - https://networkx.org/
  - Python library, relies on other libraries for visualization
  - Runs on Windows, Linux and MacOS
  - Can handle “large” networks
  - Supports the main file formats for networks
  - Supports all kinds of networks and complex systems, dynamic and hierarchical graphs

```
# Create an empty undirected graph
G = nx.Graph()

# Create some nodes
G.add_node(1)              # create a single node
G.add_nodes_from([2,3])    # create nodes from a list
G.add_node("netsci")       # node labels can be of different types

# Create some edges
G.add_edge(1,2)
G.add_edges_from([(2,3),(3,"netsci"), ("netsci", 1), (1,3)])
G.add_edges_from([(2,5),(2,6)]) # if a node does not exist, it is created

# Draw the graph
nx.draw(G, with_labels = True)
```



NetworkX Documentation

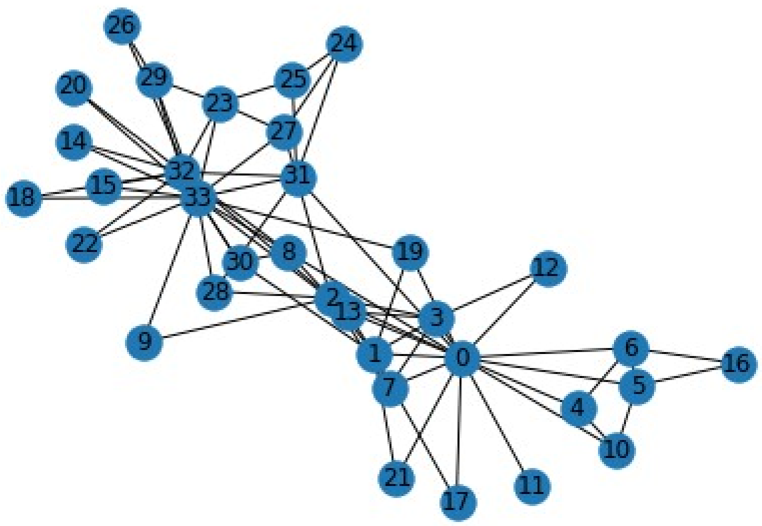
- NetworkX — NetworkX Network Analysis in Python
  - https://networkx.org/documentation/stable/
    - Includes tutorial, reference and gallery



NetworkX

- Tutorial from NetSci class
  (Pedro Ribeiro)
  https://www.dcc.fc.up.pt/~pribeiro/clad2022/networkx.html

```
G = nx.karate_club_graph()
nx.draw(G, with_labels=True)
```

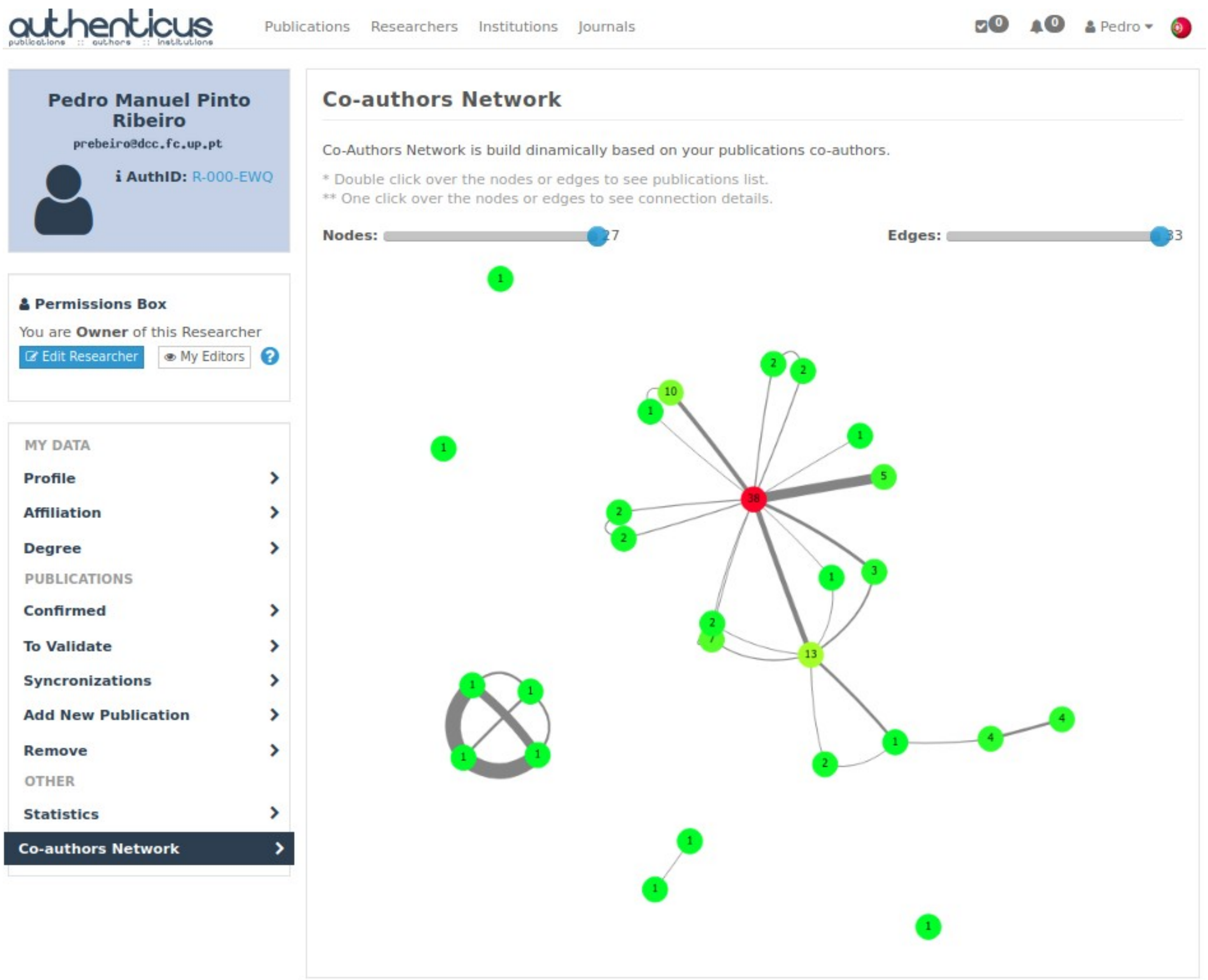


DEMO!

Other Topics

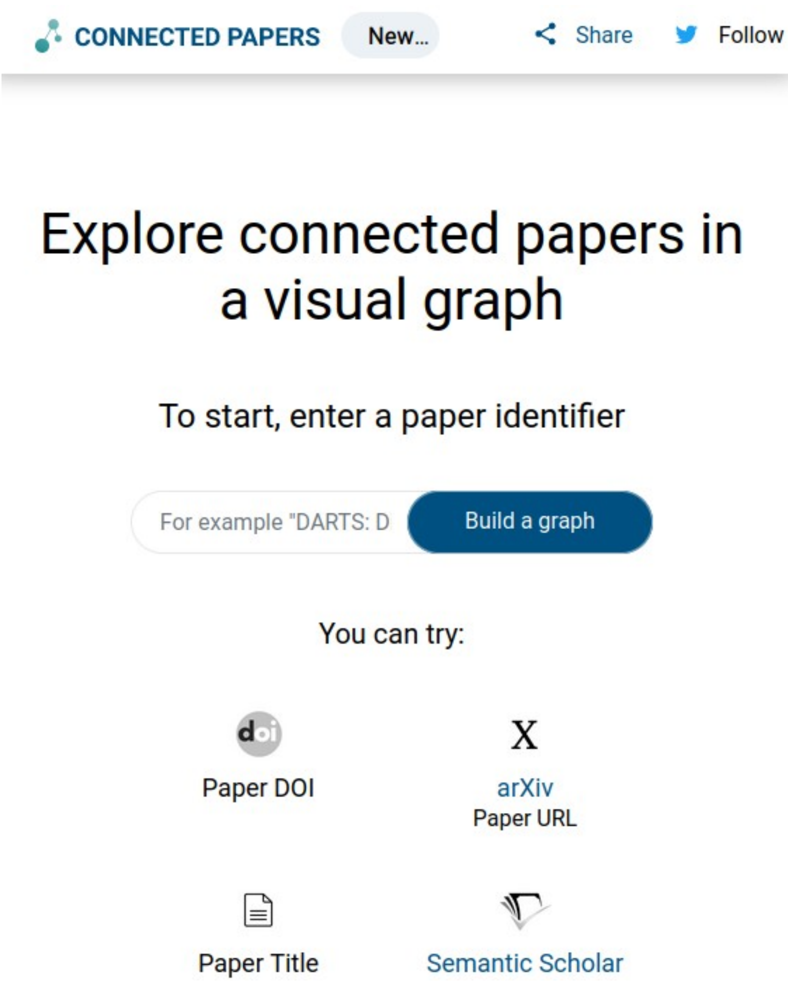
Scientific Websites

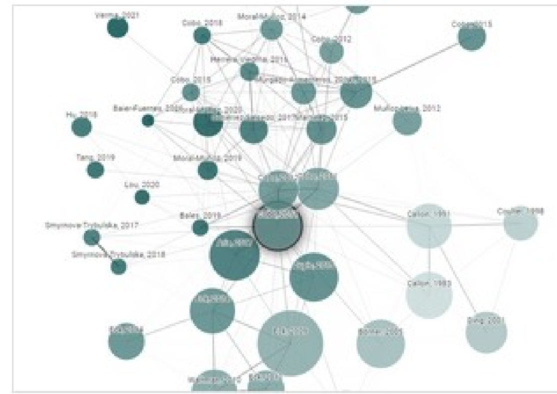
- Authenticus: https://www.authenticus.pt/



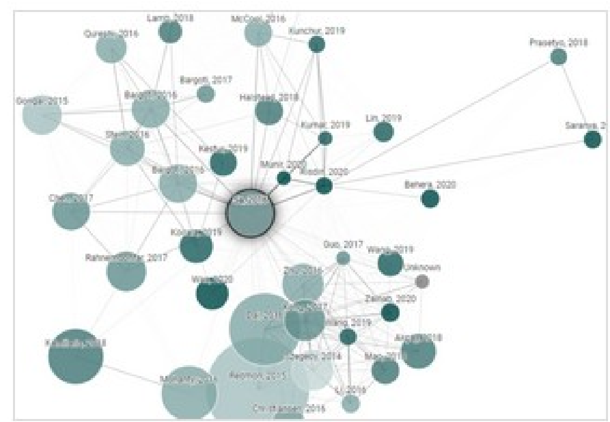
Scientific Websites

- Connected Papers: https://www.connectedpapers.com/





Science mapping software tools: Review, analysis, and cooperative study among tools (Cobo, 2011)



DeepFruits: A Fruit Detection System Using Deep Neural Networks (Sa, 2016)

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

PLOS | ONE

RESEARCH ARTICLE

Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison

David Aparício*, Pedro Ribeiro, Fernando Silva

CRACS and INESC-TEC, Faculdade de Ciências, Universidade do Porto, R. Campo Alegre, 1021, 4169-007 Porto, Portugal

* daparicio@dcc.fc.up.pt

Check for updates

OPEN ACCESS

Citation: Aparício D, Ribeiro P, Silva F (2018) Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison. PLoS ONE 13(10): e0205497. https://doi.org/10.1371/journal.pone.0205497

Editor: Gareth J. Baxter, University of Aveiro, NEW ZEALAND

Received: May 24, 2018

Accepted: September 26, 2018

Abstract

Given a set of temporal networks, from different domains and with different sizes, how can we compare them? Can we identify evolutionary patterns that are both (i) characteristic and (ii) meaningful? We address these challenges by introducing a novel temporal and topological network fingerprint named Graphlet-orbit Transitions (GoT). We demonstrate that GoT provides very rich and interpretable network characterizations. Our work puts forward an extension of graphlets and uses the notion of orbits to encapsulate the roles of nodes in each subgraph. We build a transition matrix that keeps track of the temporal trajectory of nodes in terms of their orbits, therefore describing their evolution. We also introduce a metric (OTA) to compare two networks when considering these matrices. Our experiments show that networks representing similar systems have characteristic orbit transitions. GoT correctly groups synthetic networks pertaining to well-known graph models more accurately than competing static and dynamic state-of-the-art approaches by over 30%. Furthermore, our tests on real-world networks show that GoT produces highly interpretable results, which we use to provide insight into characteristic orbit transitions.

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

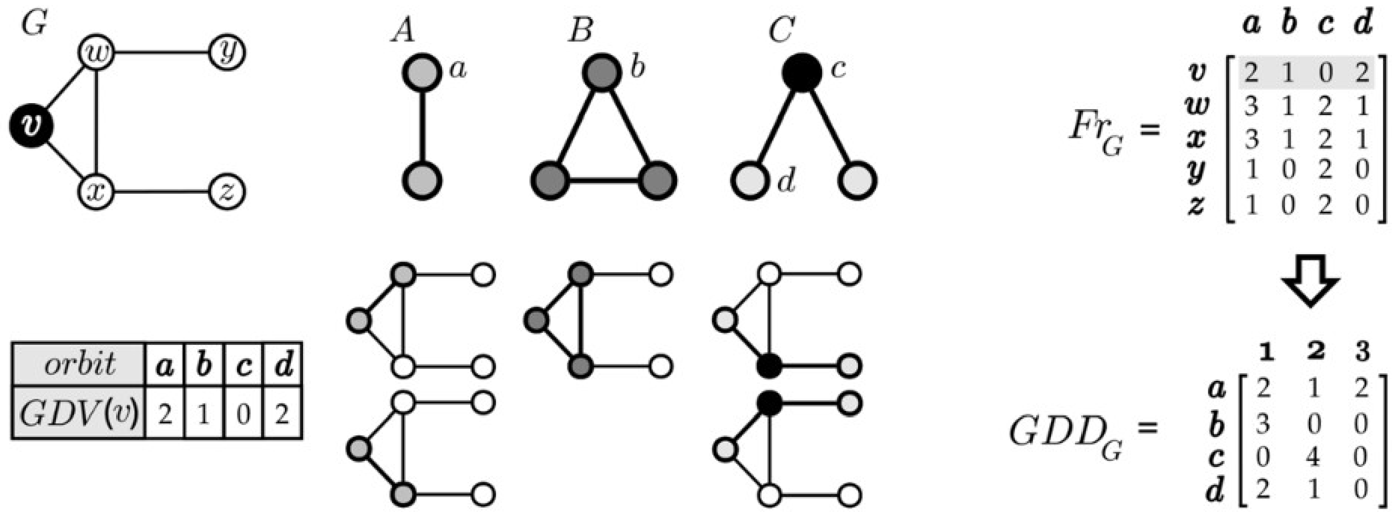


Fig 2. ***GDV*(*v*) obtained by enumerating all undirected graphlet-orbits of sizes 2 and 3 (*A*, *B* and *C*) touching *v*, and resulting Fr_G and GDD_G matrices for the complete subgraph census (*GDV*(*v*) is highlighted in gray in Fr_G).**

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

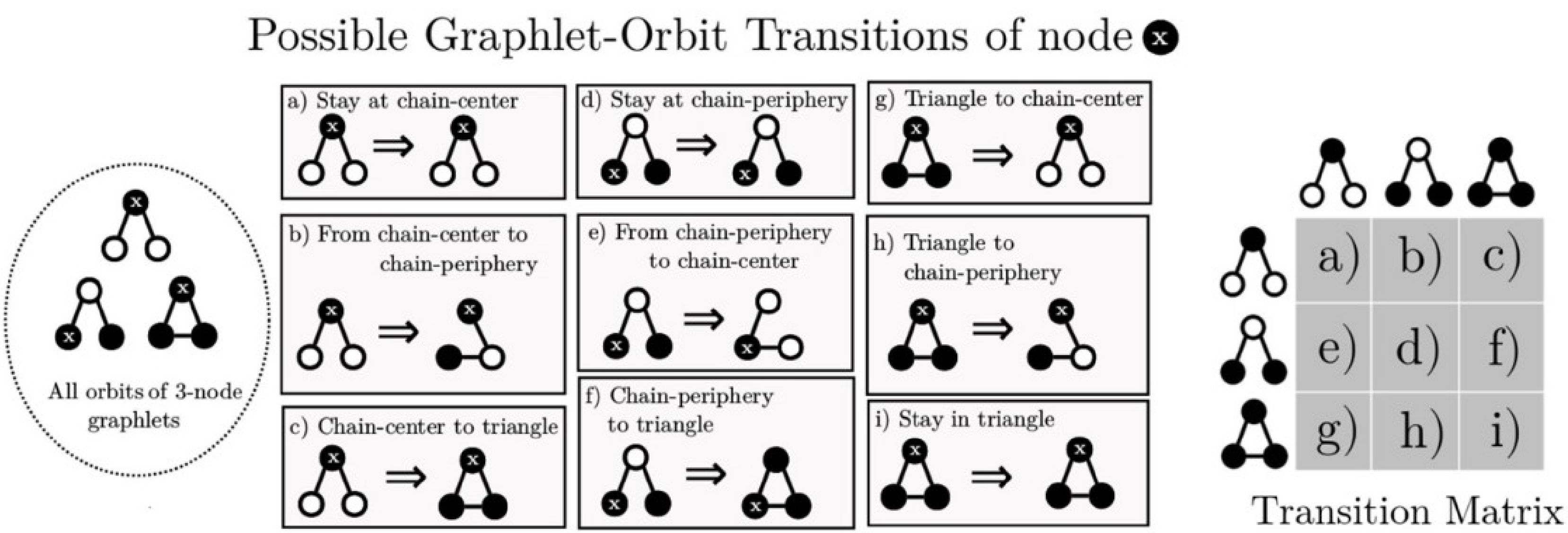


Fig 3. All possible orbit transitions of 3-node undirected graphlets and corresponding orbit-transition matrix. Node x is the node being currently considered and black nodes are nodes in the same orbit as x.

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

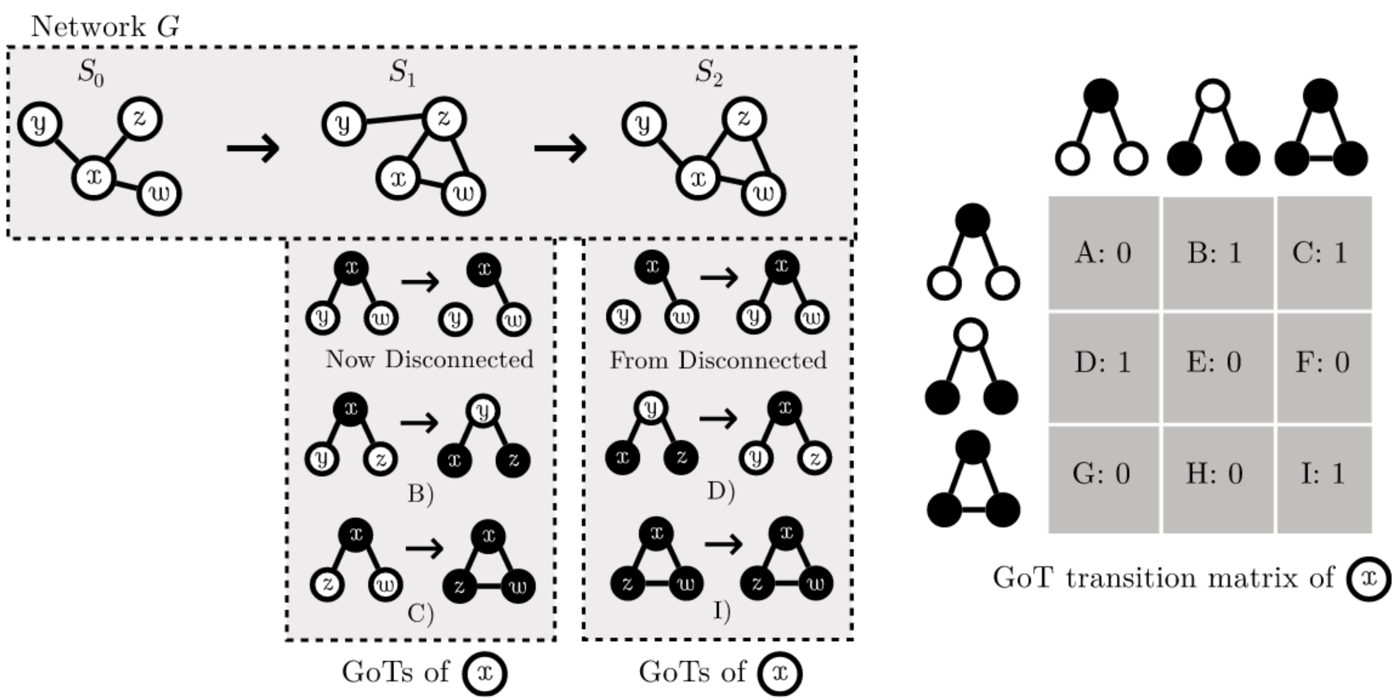


Fig 4. Graphlet-orbit transitions of node *x*. Note that transitions to (and from) disconnected graphlets are not considered.

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

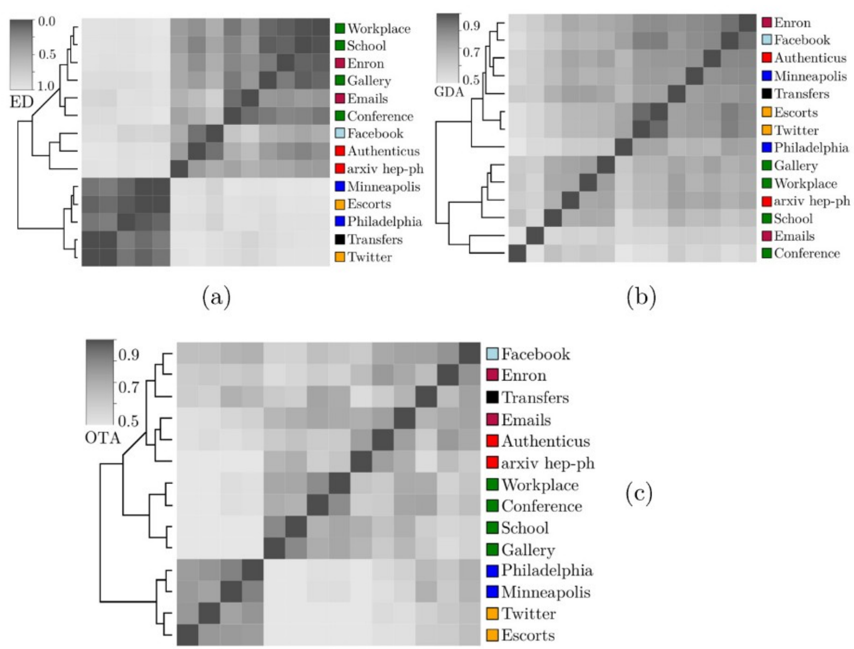


Fig 10. Similarity matrices according to (a) motif-fingerprints' Euclidean distance (*ED*), (b) graphlet-degree-agreement (*GDA*) and (c) orbit-transition-agreement (*OTA*). Clustering is performed using hierarchical clustering with complete linkage.

https://doi.org/10.1371/journal.pone.0205497.g010

Other Subjects

- Network Comparison and Network Dynamics

War Story - A paper on Network Dynamics and Patterns

David Aparicio, Pedro Ribeiro and Fernando Silva.
Graphlet-orbit Transitions (GoT): A fingerprint for temporal network comparison.
In *PloS One*, Vol. 13(10), e0205497, October, 2018.

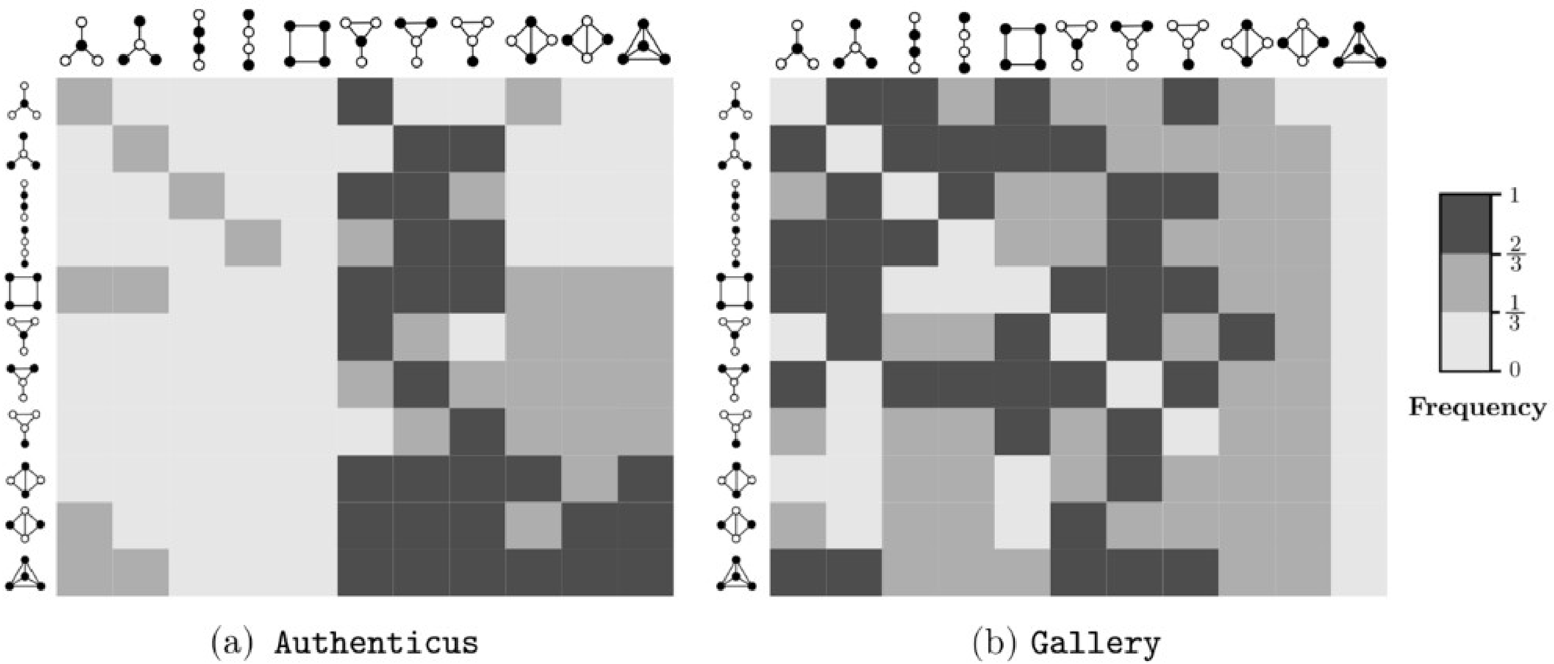


Fig 11. Orbit-transition matrices of (a) a collaboration network and a (b) physical interaction network for all 4-node orbits.

Other Subjects

- Link Prediction

War Story A paper on Link Prediction

Miguel Araújo, Pedro Ribeiro and Christos Faloutsos
TensorCast: Forecasting with Context using Coupled Tensors *(Best Paper Award)*
IEEE International Conference on Data Mining (ICDM), pp. 71-80, IEEE, New Orleans, USA, November, 2017.

TensorCast: Forecasting with Context using Coupled Tensors

Miguel Araujo
School of Computer Science
CMU and INESC-TEC
miguelaraujo.cs@gmail.com

Pedro Ribeiro
Computer Science Department
University of Porto and INESC-TEC
pribeiro@dcc.fc.up.pt

Christos Faloutsos
School of Computer Science
Carnegie Mellon University
christos@cs.cmu.edu

***Abstract*—Given an heterogeneous social network, can we forecast its future? Can we predict who will start using a given hashtag on twitter? Can we leverage side information, such as who retweets or follows whom, to improve our membership forecasts? We present TENSORCAST, a novel method that forecasts time-evolving networks more accurately than current state of the art methods by incorporating multiple data sources in coupled tensors. TENSORCAST is (a) *scalable*, being linearithmic on the number of connections; (b) *effective*, achieving over 20% improved precision on top-1000 forecasts of community members; (c) *general*, being applicable to data sources with different structure. We run our method on multiple real-world networks, including DBLP and a Twitter temporal network with over *310 million* non-zeros, where we predict the evolution of the activity of the use of political hashtags.**

I. Introduction

If a group has been discussing the #elections on Twitter, with interest steadily increasing as election day comes, can we predict who is going to join the discussion next week? Intuitively, our forecast should take into account other hashtags (#) that have been used, but also user-user interactions such as followers and retweets.

Similarly, can we predict who is going to publish on a given conference next year? We should be able to make use of, not only the data about where each author previously published, but also co-authorship data and keywords that might indicate

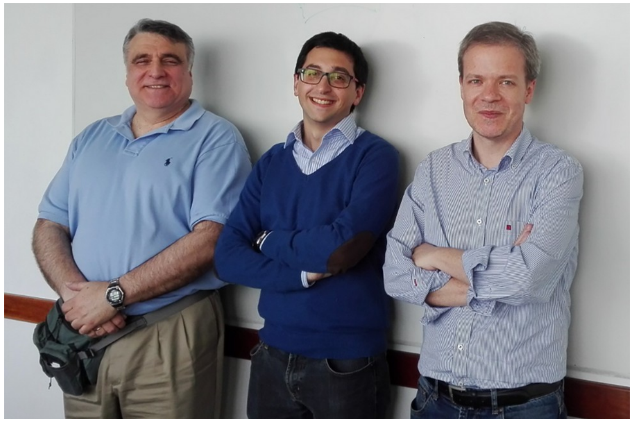
Find interactions likely to occur in the future efficiently.

Using a *naive* approach, one would have to individually forecast every pair of users and entities - a prohibitively big number that quadratically explodes. How can one avoid quadratic explosion during forecasting? How can we obtain the K likely interactions without iterating through them all?

As a summary of our results, Figure 1a shows that TENSORCAST is able to achieve 20% more precision than competing methods on the task of predicting who is going to publish on which venue in 2015 using DBLP data. Figure 1b shows TENSORCAST scaling to hundreds of millions of non-zeros on TWITTER data.

We underline our main contributions:

1) **Effectiveness**: TensorCast achieves over 20% higher precision in top-1000 queries and double the precision when finding new relations than comparable alternatives.
2) **Scalability** : TENSORCAST scales well ($E + N \log N$) with the input size and is tested in datasets with over *300M* interactions.
3) **Context-awareness**: we show how different data sources can be included in a principled way.
4) **Tensor Top-K**: we show how to quickly find the K biggest elements of sums of three-way vector outer products under realistic assumptions.



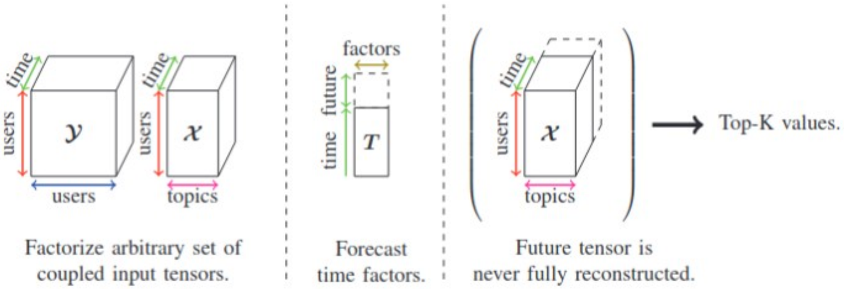
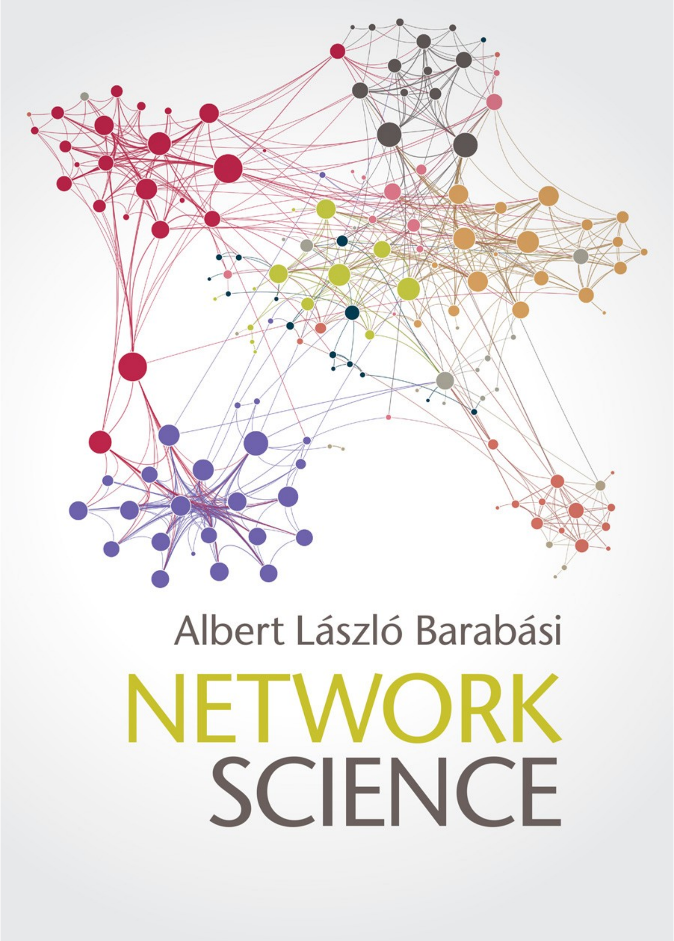


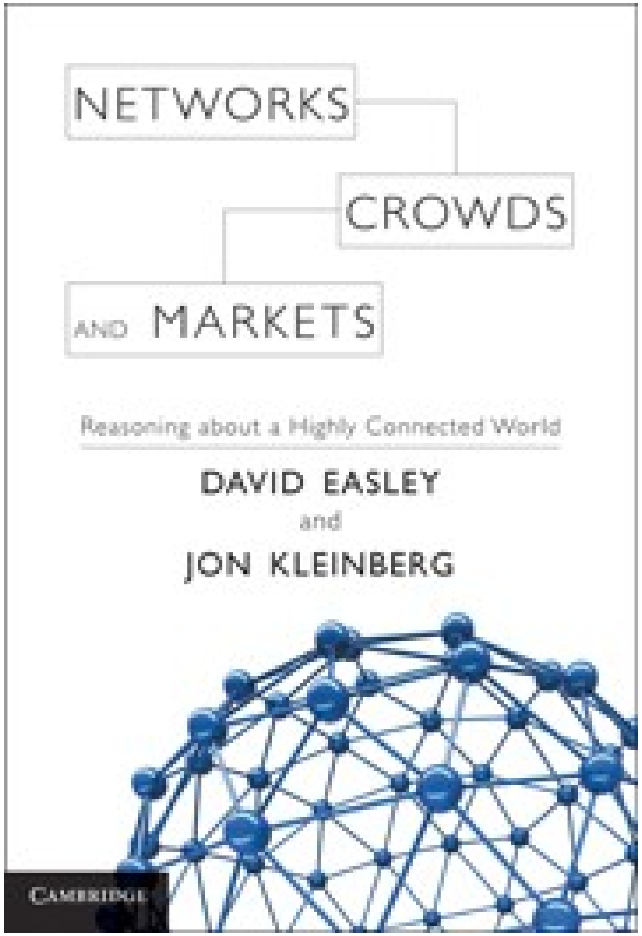
Fig. 4. **Overview** of TENSORCAST.

(Free) Book Sugestions

http://networksciencebook.com/

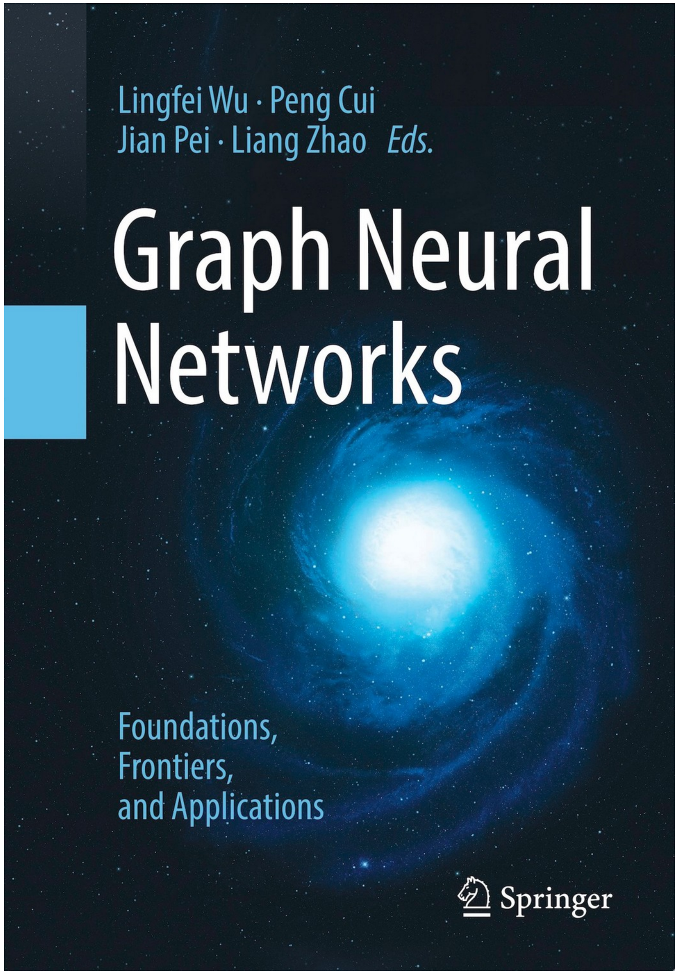


https://www.cs.cornell.edu/home/kleinber/networks-book/

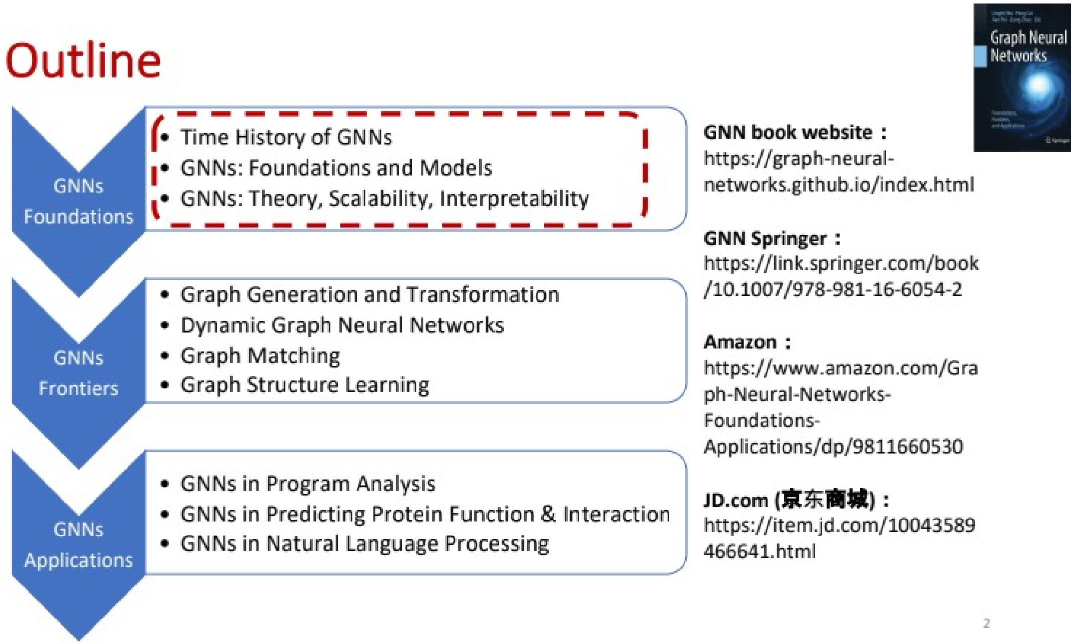


Intersections with many areas

- For instance, ***"deep learning meets network science"***



https://graph-neural-networks.github.io/



Networks are everywhere!

Any other data types

Networks!