The Bipartite Map Equation

CHRISTOPHER BLÖCKER, ICELAB, UMEÅ UNIVERSITY, SWEDEN

WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

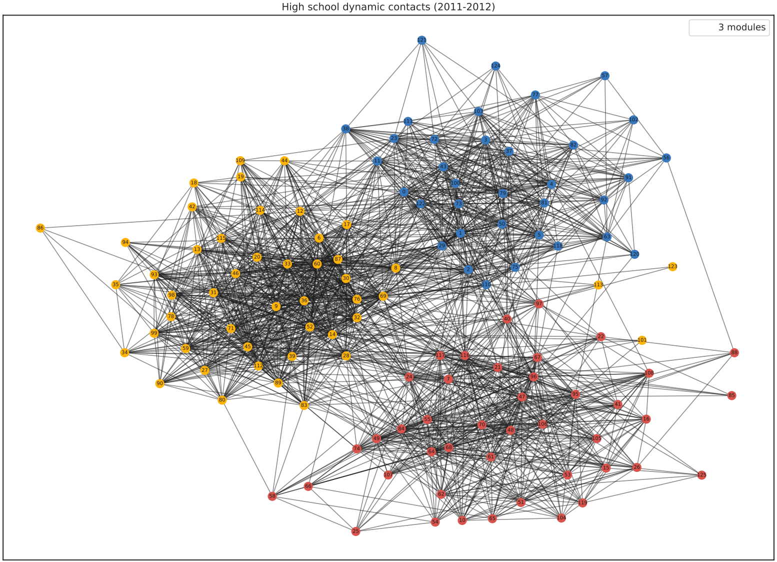
CHALMERS UNIVERSITY OF TECHNOLOGY

KTH VETENSKAP OCH KONST

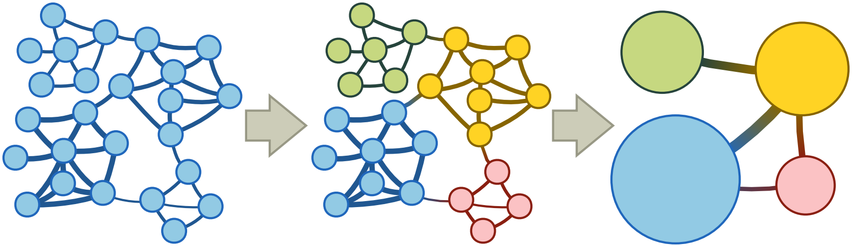
LINKÖPINGS UNIVERSITET

LUND UNIVERSITY

UMEÅ UNIVERSITET


High school dynamic contacts (2011-2012)
3 modules

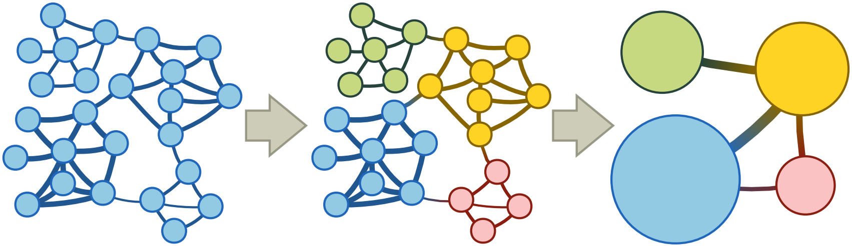
The Map Equation [Rosvall09]



$$L(M) = \overbrace{\mathcal{H}\left[\text{flows}(M)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}(M) + \sum_{m \in \text{sub}(M)} L(m)$$

We can use the cluster structure for link predictions. How?

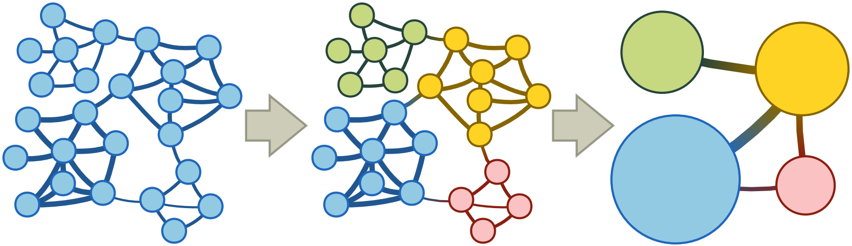
The Map Equation [Rosvall09]



$$L(M) = \overbrace{\mathcal{H}\left[\text{flows}(M)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}(M) + \sum_{m \in \text{sub}(M)} L(m)$$

We can use the cluster structure for link predictions. How?

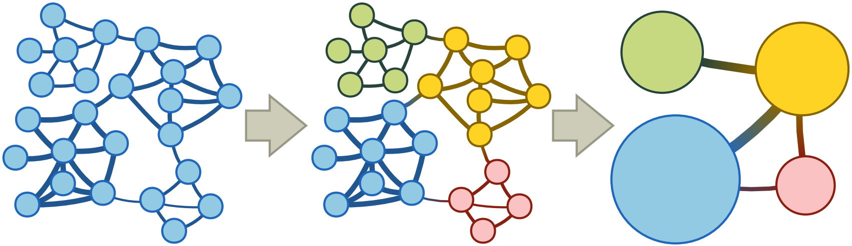
The Map Equation [Rosvall09]



$$L(M) = \overbrace{\mathcal{H}\left[\text{flows}(M)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}(M) + \sum_{m \in \text{sub}(M)} L(m)$$

We can use the cluster structure for link predictions. How?

The Map Equation [Rosvall09]



$$L\left(M\right) = \overbrace{\mathcal{H}\left[\text{flows}\left(M\right)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}\left(M\right) + \sum_{m \in \text{sub}(M)} L\left(m\right)$$

We can use the cluster structure for link predictions. How?

To predict links in a network

1. Find a partition M and calculate $L(M)$
2. Assign scores to every non-edge e, as the difference in $L(M)$ that would result from e's existence
3. Sort the non-edges by their values from (2) in ascending order

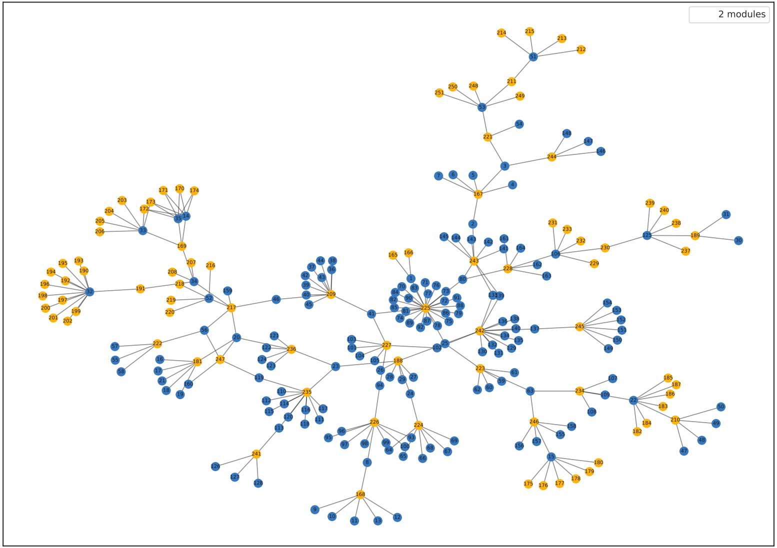
Evaluating Overfit and Underfit in Models of Network Community Structure

Amir Ghasemian, Homa Hosseinmardi and Aaron Clauset

Abstract—A common graph mining task is community detection, which seeks an unsupervised decomposition of a network into groups based on statistical regularities in network connectivity. Although many such algorithms exist, community detection's No Free Lunch theorem implies that no algorithm can be optimal across all inputs. However, little is known in practice about how different algorithms over or underfit to real networks, or how to reliably assess such behavior across algorithms. Here, we present a broad investigation of over and underfitting across 16 state-of-the-art community detection algorithms applied to a novel benchmark corpus of 572 structurally diverse real-world networks. We find that (i) algorithms vary widely in the number and composition of communities they find, given the same input; (ii) algorithms can be clustered into distinct high-level groups based on similarities of their outputs on real-world networks; (iii) algorithmic differences induce wide variation in accuracy on link-based learning tasks; and, (iv) no algorithm is always the best at such tasks across all inputs. Finally, we quantify each algorithm's overall tendency to over or underfit to network data using a theoretically principled diagnostic, and discuss the implications for future advances in community detection.

arXiv:1802.10582v3 [stat.ML] 16 Apr 2019

Norwegian Boards of Directors (2002-2011, bipartite)



The Bipartite Map Equation

$$\begin{aligned} L_B(M) &= \mathcal{H}\left[\text{flows}_{\text{LR}}(M)\right] \cdot \sum \text{flows}_{\text{LR}}(M) \\ &+ \mathcal{H}\left[\text{flows}_{\text{RL}}(M)\right] \cdot \sum \text{flows}_{\text{RL}}(M) \\ &+ \sum_{m \in sub(M)} L_B(m) \end{aligned}$$

Unipartite:

$$L(M) = \overbrace{\mathcal{H}\left[\text{flows}(M)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}(M) + \sum_{m \in \text{sub}(M)} L(m)$$

The Bipartite Map Equation

$$\begin{aligned} L_B(M) &= \mathcal{H}\left[\text{flows}_{\text{LR}}(M)\right] \cdot \sum \text{flows}_{\text{LR}}(M) \\ &+ \mathcal{H}\left[\text{flows}_{\text{RL}}(M)\right] \cdot \sum \text{flows}_{\text{RL}}(M) \\ &+ \sum_{m \in sub(M)} L_B(m) \end{aligned}$$

Unipartite:

$$L(M) = \overbrace{\mathcal{H}\left[\text{flows}(M)\right]}^{\text{Shannon entropy}} \cdot \sum \text{flows}(M) + \sum_{m \in \text{sub}(M)} L(m)$$

Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities

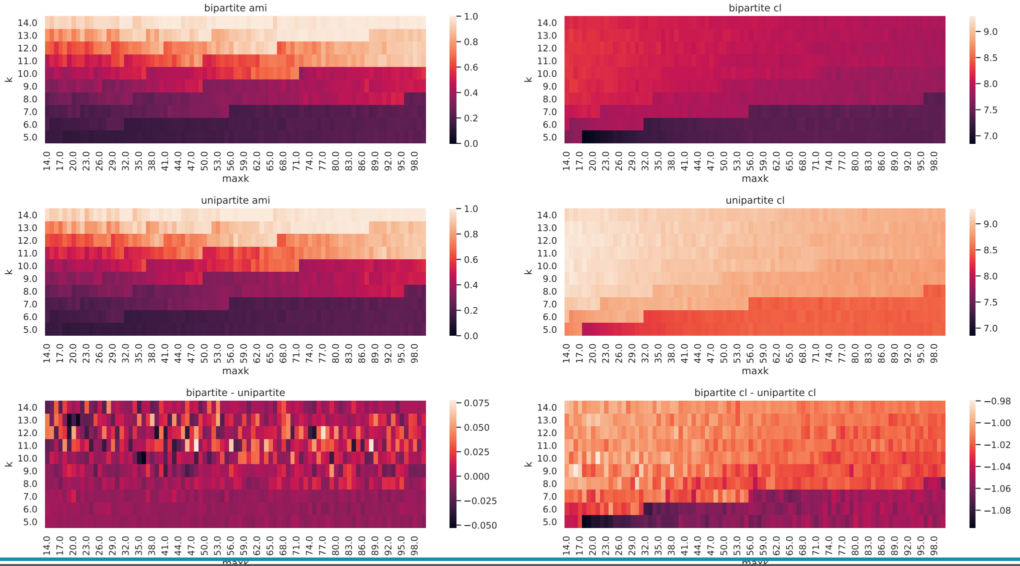
Andrea Lancichinetti[1] and Santo Fortunato[1]

[1] *Complex Networks Lagrange Laboratory (CNLL),*
Institute for Scientific Interchange (ISI), Viale S. Severo 65, 10133, Torino, Italy

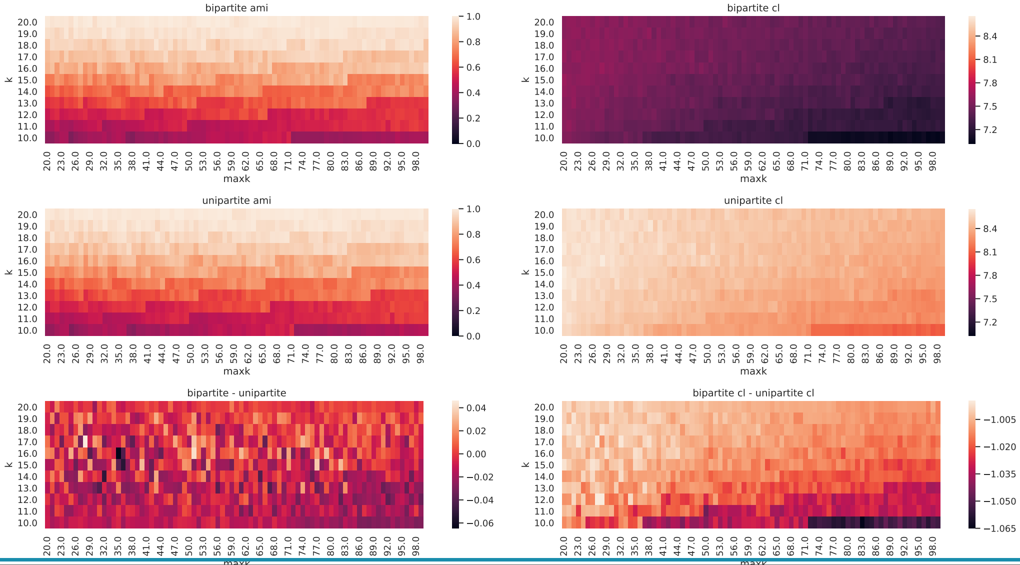
Many complex networks display a mesoscopic structure with groups of nodes sharing many links with the other nodes in their group and comparatively few with nodes of different groups. This feature is known as community structure and encodes precious information about the organization and the function of the nodes. Many algorithms have been proposed but it is not yet clear how they should be tested. Recently we have proposed a general class of undirected and unweighted benchmark graphs, with heterogenous distributions of node degree and community size. An increasing attention has been recently devoted to develop algorithms able to consider the direction and the weight of the links, which require suitable benchmark graphs for testing. In this paper we extend the basic ideas behind our previous benchmark to generate directed and weighted networks with built-in community structure. We also consider the possibility that nodes belong to more communities, a feature occurring in real systems, like social networks. As a practical application, we show how modularity optimization performs on our new benchmark.

arXiv:0904.3940v2 [physics.soc-ph] 31 Jul 2009

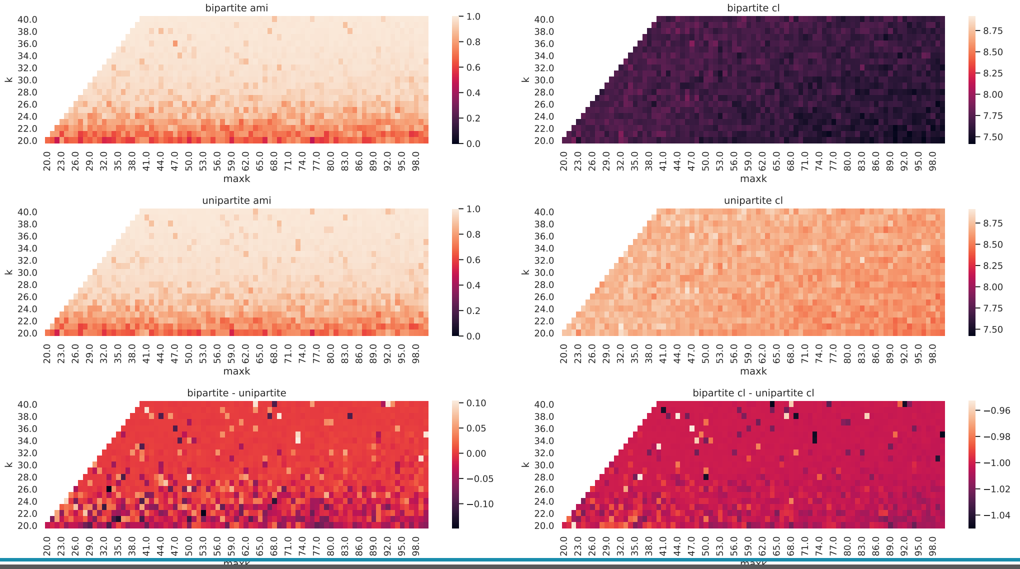
$\mu = 0.1, b = 0.5, c \in [100, 250], n = 100$



$$\mu = 0.1, b = 0.7, c \in [100, 250], n = 100$$



$\mu = 0.2, b = 0.8, c \in [100, 250], n = 10$



Questions?

Thank you for your attention!

Questions?

Thank you for your attention!