DREAM Challenge Aleph final submission

Search Register Sign in Help

Synapse ID: syn7352969

Annotations

Storage Location: Synapse Storage

Wiki

Files

Tables

Discussion

Docker beta

Disease Module Identification by Adjusting Resolution in Community Detection Algorithms

Sergio Gómez[1,2], Manlio De Domenico[1], Alex Arenas[1]

[1] Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona, Spain.
[2] Contact: S.G. (sergio.gomez@urv.cat)

Abstract

Community detection methods for complex networks usually ignore the finding of the correct scale of resolution. This is crucial in this Disease Module Identification challenge, since communities must contain no more than 100 genes. Therefore, we have used the Multiresolution technique in (Arenas, Fernández and Gómez, 2008) to choose an appropriate scale for two different methods: modularity optimization in subchallenge 1, and a multiscale Multimap approach (the extension of Infomap to multilayer networks, De Domenico et al., 2015) in subchallenge 2.

Contents

Introduction

In the absence of information on the cluster sizes of the graph, a method should be able to explore all possible topological scales at which clusters may satisfy the definition of module, to make sure that it will eventually identify the right communities (Arenas, Fernández and Gómez, 2008). *Multiresolution* methods are based on this principle, proposing the screening of the topology at different resolution levels. Moreover, many real graphs display hierarchical cluster structures, with clusters inside other clusters. In these cases, there are more levels of organization of vertices in clusters, and more relevant scales (Granell, Gómez and Arenas, 2012). In principle, algorithms devoted to community detection should be able to identify them. Multiresolution methods can perform this task, as they scan continuously the range of possible cluster scales. The determination of the correct scale of description is reinterpreted as the scale that is more persistent at changes in the resolution parameter. While single-scale methods find a unique partition satisfying the criteria imposed by a quality function or other information maximization techniques, multiresolution based algorithms are able to scrutinize the space of partitions in a deeper way. Eventually, the heterogeneity of the partitions that better classify the data in groups is captured by tuning the topological resolution.

The multiresolution method in (Arenas, Fernández and Gómez, 2008) works by just adding a parameter r, known as *resistance*, to the community detection algorithms. This resistance controls the aversion of nodes to form communities; the larger the resistance, the smaller the size of the modules. For community detection algorithms based on the optimization of the well-known *modularity* function (Newman and Girvan, 2004), this resistance takes the form of a self-loop (with a weight equal to r) which is added to all nodes of the network. In this way, all nodes contribute to the internal strength of their modules with a constant amount r, which is independent of the rest of the connectivity of the network. In mathematical form, if W is the matrix formed by the weights w_{ij} from node i to node j, the new weights matrix W_r after the addition of the resistance r is $W_r = W + rI$, where I is the identity matrix. Of course, when the resistance is zero, the standard (and implicit) scale of resolution is recovered ($W_0 = W$).

Although this multiresolution methodology was conceived for community detection algorithms based on the optimization of modularity, it can also be applied to other completely different approaches, for example to those based on flow dynamics or random walks. In these cases, the resistance offers the flow (or random walkers) the possibility of remaining in the node, thus enabling the access to new scales of resolution. One of the algorithms which can take advantage of our multiresolution approach is *Infomap* (Rossvall and Bergstrom, 2008), which finds communities by looking for minimum description lengths of the paths of random walkers through the network.

Another important aspect for this challenge is the alignment of the networks in subchallenge 2, which can be modeled as layers of a multilayer system (De Domenico et al., 2013). Here we have used the multiscale *Multimap* approach (the extension of Infomap to multilayer networks, De Domenico et al., 2015) to unravel the mesoscale structure of the overall network. The method encodes the dynamics of random walkers exploring the multilayer network (De Domenico et al., 2014; De Domenico et al., 2016) while minimizing its description length. This algorithm has also been adapted to take advantage of our multiresolution methodology, with the addition of a resistance to all nodes of the multilayer network.

The novelty of the approach is twofold: 1) it relies on the introduction of self-loops that rescale all the topological dimensions involved in the process of capturing the best modular structure of networked data; 2) it makes use of multilayer community detection based on how information flows through the network.

Methods

The methodology applied involves the following main steps:

1. Preprocess and filter the input data, and build the networks
2. Find communities at different resolutions, and choose the appropriate one
3. Extract large communities as new networks, and find subcommunities at an appropriate resolution
4. Process and check the final partitions

1. Preprocess

The goal here is to build weighted networks for the posterior community detection phase, removing the non-significant edges, and normalizing them when required. Network "3_signal_anonym_directed_v3" in subchallenge 1 and the multilayer network in subchallenge 2 are considered and analyzed as directed networks.

Subchallenge 1

The analysis of the input data showed that most of the networks were very dense (average degrees above 100), so a first step was to discard the less significant edges. Thus, we only retained between 10% and 20% of the edges (those with largest weights), with the exception of network "3_signal_anonym_directed_v3" (average degree 4.15), for which all edges were maintained.

Subchallenge 2

Since the ranges of variation of the weights in each layer were different, we first ensured each layer was between 0 and 1 (normalizing those which did not fulfill this requirement), and then the overall 15% of the largest weights were used to form the input multilayer network.

2. Multiresolution community detection

When you directly apply community detection algorithms to our current networks, without caring about the resolution scale, in most of the cases you obtain between 20 and 40 communities of sizes larger than 100, which include several ones with more than 300 nodes, and even some above 500 nodes. This means that almost half of the nodes belong to modules which would be discarded by the challenge size requirements: "Only modules that contain at least 3 genes and at most 100 genes will be used for the scoring (modules outside this range will simply be ignored)". As explained above, we have applied our Multiresolution method (Arenas, Fernández and Gómez, 2008) to look for partitions at more appropriate scales of resolution, capable of delivering most communities below the 100 nodes threshold. However, it is not convenient to use a too large resistance to attain this goal. The problem lays in the other extreme of the modules' sizes: the larger the resistance, the larger the number of nodes in very small communities. As a general rule, small modules are non-important ones, but we want to avoid their proliferation. Therefore, the best solution is to maintain a trade-off between large and small communities, and this can be achieved by maximizing the proportion of nodes inside communities of the desired sizes, i.e. between 3 and 100. Only a few values of the resistance parameter have been checked for each network (between 5 and 10 different values) due to the time cost of each community detection step, but that has been enough to find much better resolutions than the default one ($r = 0$).

Subchallenge 1

We have found the partitions of communities by optimizing modularity (Newman and Girvan, 2004). In fact, specific versions of modularity for weighted (Newman, 2004a) and directed (Arenas et al., 2007) networks are needed to avoid losing this valuable information. Our optimization of modularity has involved the use of a "cocktail" of community detection algorithms. The idea is that a combination of several algorithms has less chances to get stacked in a bad partition, and the quality of the final partition is much improved. Specifically, they are:

- Extremal optimization (Duch and Arenas, 2005)
- Spectral optimization (Newman, 2006)
- Fast algorithm (Newman, 2004b)
- Fine-tuning by iterative reposition of individual nodes

Subchallenge 2

The selected community detection algorithm is Multimap (De Domenico et al., 2015), the extension of Infomap (Rossvall and Bergstrom, 2008) to multilayer networks.

3. Analysis of large communities

The partitions obtained with the previous multiresolution analysis may contain several communities with more than 100 nodes. To avoid them being ignored by the scoring process of the challenge, we extracted those communities from the full graph, creating new reduced networks, and applied them a second level of multiresolution analysis. This procedure is similar to the fully hierarchical method in (Granell, Gómez and Arenas, 2012).

Subchallenge 1

In this case, we replaced the Extremal and Spectral methods of the "cocktail" of modularity optimization algorithms with the Tabu search (Arenas, Fernández and Gómez, 2008). The advantage of Tabu search is its capacity to obtain partitions with a much better modularity than the Extremal and Spectral ones. Unfortunately, Tabu is much slower than Extremal and Spectral, thus it was not possible to apply it to the full networks in the previous phase.

Subchallenge 2

No further second level analysis was performed with the multilayer network.

4. Postprocess

Despite the previous phase, some communities with more than 100 nodes still remained, due to either their appearance in the second level of optimization, or because their size was only a small amount above 100. In these cases, a random split in modules of 100 nodes (plus a remainder module) was performed to ensure their consideration in the scoring process. It must be emphasized that the restriction of 100 nodes is rather arbitrary, thus there appear communities which cannot fulfill this requirement in any natural way, tending to survive during the whole analysis. Finally, the first and second level communities were combined to form the definitive partitions, checking their consistency and being formatted in the required way.

Software

The software we have used is the following:

- Radalib (http://deim.urv.cat/~sergio.gomez/radalib.php): This is an "Ada library and tools for the analysis of Complex Networks and more". It contains, among many others, our programs to perform the Multiresolution community detection by the optimization of modularity. The software is freely available both in this main site and in GitHub (https://github.com/sergio-gomez/Radalib). Executables of the tools for the main platforms are also available in Radatools (http://deim.urv.cat/~sergio.gomez/radatools.php). The tools we have used are the following:
  - *Communities_Detection*: implements the modularity optimization algorithms (Tabu, Extremal, Spectral, Fast and Reposition), with resistance parameter, and weighted and directed modularity.
  - *List_To_Net*: converts a list of edges into a network in Pajek format.
  - *Network_Properties*: calculates many properties of the networks.
  - *Extract_Subgraphs* : extracts communities from a partition as individual networks.
  - *Reformat_Partitions* : to write partitions substituting indices of nodes by names.
- Infomap at MapEquation.org (http://www.mapequation.org/): implements Multimap, the extension of Infomap to multilayer networks, as an option within the Infomap program. It is freely available both in this main site and in GitHub (https://github.com/mapequation/infomap).
- A simple python script for the postprocess phase described above.

Implementation details

Here we describe some of the details needed to reproduce our results as submitted to this challenge. Of course, the stochastic nature of the optimization algorithms means that new executions could lead to slightly different results, but they serve as a guide.

Subchallenge 1

In the next table we summarize the main information corresponding to the analysis of the networks. During the preprocess phase, all edges with a weight below the pruning threshold are discarded. The multiresolution community detection makes use of an `ersrfr` optimization (e: extremal; s: spectral; r: reposition; f: fast), with the finally selected resistances shown in the Table. Then, the indicated number of largest communities are extracted as new networks for the 2nd level multiresolution community detection, a `trfr` optimization scheme is applied (t: tabu), and the corresponding selected resistances are shown in the last column (sorted by decreasing number of nodes). Note that some of the resistance values are negative, in order to increase the size of the new modules with respect to the ones at default ($r = 0$) resolution.

Network	Pruning threshold	1st level resistance	Number of 2nd level networks	2nd level resistances
1_ppi_anonym_v2	0.60	200.0	11	70, 25, 40, 10, 0, 10, 5, 5, -5, -5, -5
2_ppi_anonym_v2	0.60	400.0	3	0, -10, 0
3_signal_anonym_directed_v3	0.00	100.0	2	-10, -10
4_coexpr_anonym_v2	0.30	80.0	8	0, 0, -2, 0, 0, 0, 0, 0
5_cancer_anonym_v2	0.45	100.0	3	50, 10, 10
6_homology_anonym_v2	15.00	10000.0	3	500, 750, -5

Subchallenge 2

In this case, after the preprocessing step explained above which prunes 85% of the edges, the multilayer Infomap is run with the parameters in the next Table (the others being the default ones). Several values of the resistance parameter (self-loops) are tried until an appropriate scale is obtained, at $r = 2.0$.

Parameter	Value
Employ multiplex random walkers	`-i multiplex`
The network is directed	`--directed`
Number of independent tries	`-N 20`
Output the cluster file	`--clu`
Use specific value of the relax rate	`--multiplex-relax-rate 0.15`
Optimization method	`-2 --two-level`
Obtain communities of physical nodes (not of state nodes)	`--hard-partitions`

Conclusions

We have developed a strategy for disease module identification which, using different algorithms for community detection, shows the importance of adjusting the resolution of the study. Multiresolution analysis should always be a central part whenever you are interested in the mesoscale of a system, since many methods define (usually in an implicit form) a certain scale, which could be inappropriate for your needs. We have also shown how this multiresolution can be incorporated in two general classes of community detection algorithms: modularity optimization for single layer networks, and random walk flows in multilayer networks. Other multiresolution approaches exist in the literature, but the important message here is that the users must be aware of their central role, and use this knowledge to improve the outcome in all their practical applications.

Finally, it must be stated that we did not participated in the previous rounds of the challenge (we were invited to participate too late), which would have been an important source of information to improve our set-up and to automate the different phases of the whole process.

References

- Arenas, A., Duch, J., Fernández, A., and Gómez, S. (2007). Size reduction of complex networks preserving modularity. New J. Phys. 9, 176.
- Arenas, A., Fernández, A., and Gómez, S. (2008). Analysis of the structure of complex networks at different resolution levels. New J. Phys. 10, 053039.
- De Domenico, M. et al. (2013). Mathematical Formulation of Multi-Layer Networks. Phys. Rev. X 3, 041022.
- De Domenico, M., Solé-Ribalta, A., Gómez, S., and Arenas, A. (2014). Navigability of interconnected networks under random failures. Proc. Nat. Acad. Sci. USA 11, 8351.
- De Domenico, M., Lancichinetti, A., Arenas, A., and Rosvall, M. (2015). Identifying modular flows on multilayer networks reveals highly overlapping organization in social systems. Phys. Rev. X 5, 011027.
- De Domenico, M., Granell, C., Porter, M.A., and Arenas, A. (2016). The physics of spreading processes in multilayer networks. Nature Phys. 12, 901.
- Duch, J., and Arenas, A. (2005). Community detection in complex networks using extremal optimization. Phys. Rev. E 72, 027104.
- Granell, C., Gómez, S., and Arenas, A. (2012). Hierarchical multiresolution method to overcome the resolution limit in complex networks, Int. J. Bifurc. Chaos 22, 1250171.
- Newman, M.E.J., and Girvan, M. (2004). Finding and evaluating community structure in networks. Phys. Rev. E 69, 026113.
- Newman, M.E.J. (2004a). Analysis of weighted networks. Phys. Rev. E 70, 056131.
- Newman, M.E.J. (2004b). Fast algorithm for detecting community structure in networks. Phys. Rev. E 69, 066133.
- Newman, M.E.J. (2006). Modularity and community structure in networks. Proc. Nat. Acad. Sci. USA 103, 8577.
- Rosvall, M., and Bergstrom, C.T. (2008). Maps of random walks on complex networks reveal community structure. Proc. Nat. Acad. Sci. USA 105, 1118.

Author Contributions

S.G., M.D.D. and A.A. conceived the study. S.G. performed the study of subchallenge 1. M.D.D. performed the study of subchallenge 2. S.G refined and prepared the final submission. S.G., M.D.D. and A.A. wrote the manuscript.

Acknowledgements

A.A. and S.G. acknowledge MULTIPLEX (grant number 317532) of the European Commission, and the Spanish Ministerio de Economía y Competitividad (grant number FIS2015-71582-C2-1). A.A. also acknowledges ICREA Academia and the James S. McDonnell Foundation. M.D.D. acknowledges financial support from the Spanish program Juan de la Cierva (IJCI-2014-20225).

Created by S Sergio Gómez (mesoscales) on Sunday, October 9, 2016 8:52 PM

Modified by S Sergio Gómez (mesoscales) on Tuesday, October 11, 2016 11:09 PM

Wiki History