Extraction of overlapping modules in networks via spectral methods and information theory

Rion Brattig Correia[1,2,*], Paulo Navarro Costa[1,3], and Luis M. Rocha[1,4,*]

[1] Instituto Gulbenkian de Ciência, Oeiras, Portugal
[2] CAPES Foundation, Ministry of Education of Brazil, Brasília DF, Brazil
[3] ISAMB - Instituto de Saúde Ambiental, Faculdade de Medicina, Universidade de Lisboa, Lisbon, Portugal
[4] Luddy School of Informatics, Computing & Engineering, Indiana University, Bloomington IN, USA
*rionbr@gmail.com,rocha@indiana.edu

Networks are a common method to model multivariate interactions in a variety of complex systems found in nature and society. The interactions captured by networks can include a multitude of complex phenomena occurring at various levels of observation and intensity, which are difficult to disentangle automatically. For instance, functionally-relevant gene modules in a network of gene interactions, or disease-related terminology in networks derived from drug and symptom mentions in social media [1] typically have overlapping clusters of widely varying size and strength. To address these and other similar questions, a variety of community structure algorithms have have been proposed in the literature [2,3,4]. However, most modularity algorithms seek an optimal partition of the network where each node must belong to a module. This hard-boundary assumption does not match the fluid phenomena often found in biomedical complexity. Indeed, many genes are involved in different biochemical pathways depending on their expression levels and which other biochemical species are present. Hence, the same gene can participate in distinct modules. In these biomedical problems it is more reasonable to assume that network variables can map to multiple, partially overlapping functional communities. Thus, for biomedical applications of network science there has been much recent interest in spectral, overlapping clustering methods [5].

Here we propose a new spectral method to automatically extract overlapping clusters from networks. It is based on two steps: 1) the *Singular Value Decomposition* (SVD) of weighted graph adjacency matrices, in a process akin to *Principal Component Analysis* (PCA) of gene expression data [6], and 2) automatic extraction of overlapping modules using information theory and a polar coordinate projection of data onto singular vector (or component) subspaces. In the first step, when the original network is a bipartite graph relating two distinct sets of variables (e.g. genes vs assays in time, or disease codes vs social media user timelines), we compute the SVD of the bipartite adjacency matrix. If the network is a weighted graph of a single set of variables (e.g. genes), we perform the PCA of the (covariance-normalized) adjacency matrix (see [6] for the difference between SVD and PCA). Fig. 1A depicts the eigenvector variance spectrum of a *Drosophila* gene interaction network obtained via PCA, where the first eigenvector (or component) explains 20% of the variance in the gene co-expression data, and is associated with a large module involving most genes and their regular expression patterns (e.g. cell division, housekeeping and cell cycle).

In functional analysis we are most interested in biochemical processes involving smaller modules which have specific regulatory functions beyond the regular cell operations captured by the first component [7]. Thus, in the second step of the method, we target subsets of lower components. Fig. 1B depicts a biplot of all genes in the network projected as points onto components 2 and 3 of the spectrum. The majority of points is (randomly) projected at the origin of the biplot, showing that they are not correlated with the phenomena captured by either component. Therefore, we want to identify those points (genes) that most protrude and cluster away from the origin, as those are most correlated with the target components. To do so, we transform the Cartesian coordinates of every point to polar coordinates (see Fig. 1D), apply a moving window over the range of radiuses, and compute the Shannon entropy of the distribution of points over angle bins in each radius window. We use overlapping bins (or fuzzy intervals [8]) for both radiuses and angles, meaning that a node at a particular polar coordinate can contribute to more than one angle and radius bin simultaneously (see Fig. 1C & E, respectively).

Computing the Shannon entropy (see red line in Fig. 1D) allows us to track when the distribution of points in polar angle bins transitions from a random to a more structured arrangement. Because points near the origin (radius close to zero) are uncorrelated with the components of interest, the distribution of polar angles tends to be uniformly random, as seen in Fig. 1B,D. As the radius increases, points tend to cluster near specific angles, leading to lower Shannon entropy of the angle distribution. Thus, the goal of the second step of the algorithm is to identify the radius where important transitions in Shannon entropy occur, especially where the distribution of polar angles moves away from a uniform distribution (see blue lines in Fig. 1B,D). Naturally, several entropy transitions may occur, as some clusters are more correlated with components of interest than others—and thus have a higher radius. In other words, identifying the best clusters becomes a multi-objective optimization problem. Several measures can be used to optimize, but we exemplify the method with the rank-sum of radius and entropy to identify the radiuses that maximize the number of points selected while simultaneously minimizing the entropy value. Once a radius is selected, we retrieve only the points that lay beyond the circle it defines. The distinct clusters are then formed by the circle segments that contain similar polar angles; see red polygon in Fig. 1B with radius ≥ 4 selected by rank-sum. In our example, the red module corresponds to genes involved in protein regulation via the proteasome complex, as characterized via gene ontology enrichment analysis (GOEA) [9]. Finally, it important to stress that the clusters thus identified, contain genes that may overlap with clusters found in other component subspaces. In other words, the same network nodes can contribute to overlapping modules associated with distinct phenomena.

In the talk, we will discuss variations of the entropy and multi-objective optimization measures, and apply the method to dense and noisy network data from four examples: (i) synthetic networks; (ii) a gene interaction network from transcriptomic data (RNAseq) from Drosophila intestinal cells (Fig. 1); (iii) a knowledge network of drug and symptom terms extracted from social media user timelines [1]; and (iv) a workspace social interaction network collected using radio-frequency identification (RFID) [10]. Our intuitive method to extract overlapping network clusters will be useful to a wide range of network practitioners from the biomedical to the social sciences.

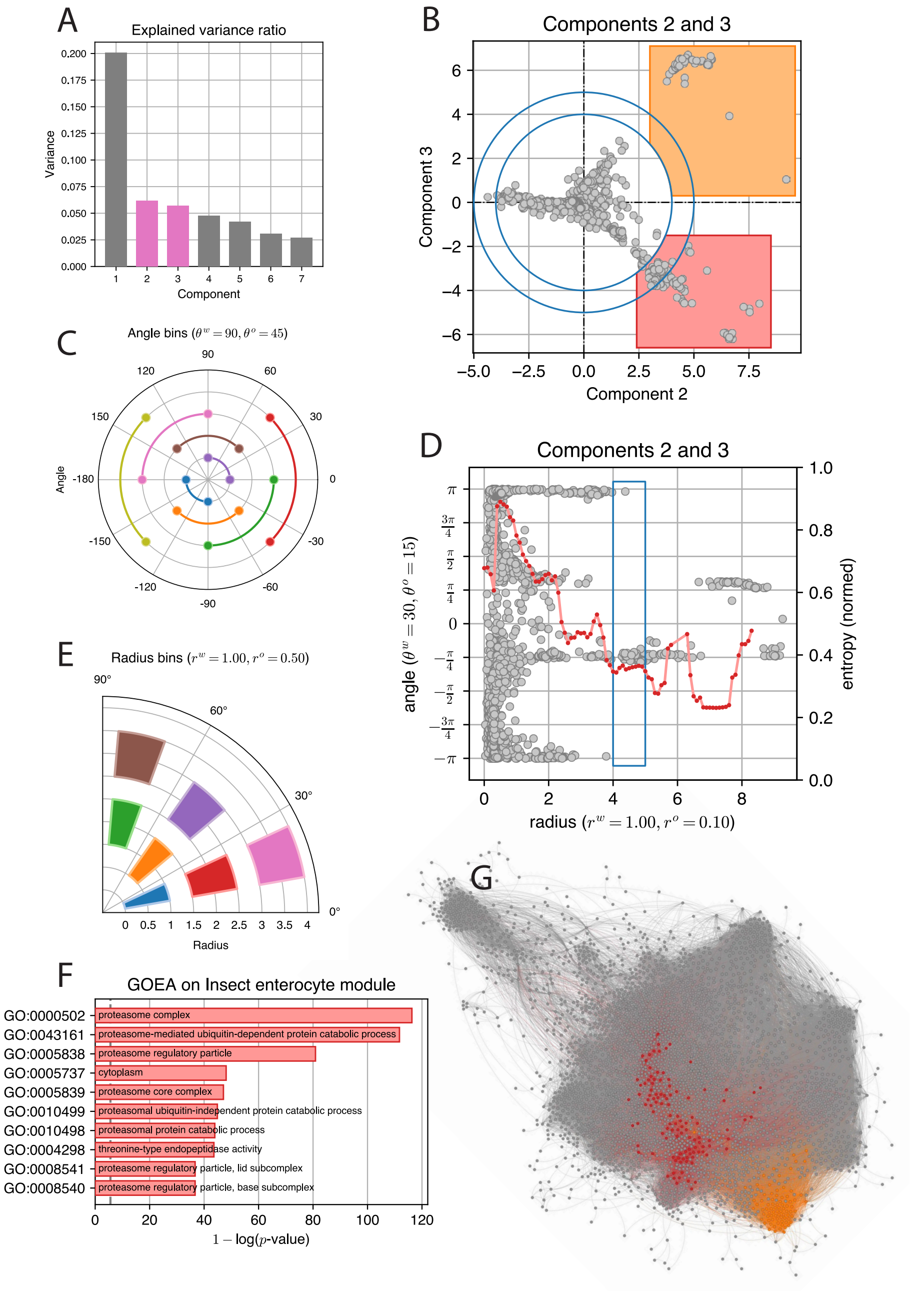

Fig. 1. Gene interaction network of insect (*Drosophila melanogaster*) intestinal cells. A. Spectrum of PCA components of the gene interaction network adjacency matrix, ordered by proportion of explained covariance. **B**. Projection of genes (network nodes) onto biplot of PCA components 2 and 3. Two network modules are highlighted in red and orange. Blue circles shows the minimum entropy window selected (also in D). **C**. Angle bins used in analysis, with width of $r^W = 90$ and overlap of $r^o = 45$. Bins positioned at varying radiuses for easier visualization. **D**. Radius (horizontal) and polar angle (vertical) of same points as in B (subspace of components 2 and 3). Red line and points show the normalized entropy values for each radius window computation ($\theta^W = 30$, $\theta^o = 15$; $r^W = 1.0$, $r^o = 0.1$). Blue rectangle shows the minimum entropy window selected (also in B). **E**. Radius bins used in analysis with width of $\theta^W = 1$ and overlap of $\theta^o = 0.5$. **F**. Gene ontology enrichment analysis (GOEA) of the identified red module (see B). Top 10 significant GO terms shown. **G**. Insect gene interaction network with red and orange modules identified (also in B).

References

1. Correia, R.B., Li, L., Rocha, L.M.: Monitoring potential drug interactions and reactions via network analysis of instagram user timelines. In: Pacific Symposium on Biocomputing. Volume 21. (2016) 492–503
2. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment **2008**(10) (10 2008) P10008
3. Newman, M.E.J.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74** (9 2006) 036104
4. Fortunato, S.: Community detection in graphs. Physics Reports **486**(3) (2010) 75–174
5. Van Lierde, H., Chow, T.W.S., Chen, G.: Scalable spectral clustering for overlapping community detection in large-scale networks. IEEE Transactions on Knowledge and Data Engineering **32**(4) (2020) 754–767
6. Wall, M.E., Rechtsteiner, A., Rocha, L.M.: Singular value decomposition and principal component analysis. In: A practical approach to microarray data analysis. Springer (2003) 91–109
7. Rechtsteiner, A.: Multivariate analysis of gene expression data and functional information: Automated methods for functional genomics. PhD thesis, Portland State University (2005)
8. Klir, G., Yuan, B.: Fuzzy sets and fuzzy logic. Volume 4. Prentice hall New Jersey (1995)
9. Klopfenstein, D.V., Zhang, L., Pedersen, B.S., Ramírez, F., Warwick Vesztrocy, A., Naldi, A., Mungall, C.J., Yunes, J.M., Botvinnik, O., Weigel, M., Dampier, W., Dessimoz, C., Flick, P., Tang, H.: Goatools: A python library for gene ontology analyses. Scientific Reports **8**(1) (2018) 10872
10. Génois, M., Vestergaard, C.L., Fournet, J., Panisson, A., Bonmarin, I., Barrat, A.: Data on face-to-face contacts in an office building suggest a low-cost vaccination strategy based on community linkers. Network Science **3**(3) (9 2015) 326–347