Evaluating the scalability of clustering algorithms on protein-protein interaction networks

Jose Picado

Abstract

The structure of protein-protein interaction (PPI) networks reveals important properties of the functioning of living organisms. In this paper, we explore the automated detection of groups of clusters in PPI networks. We employ several clustering algorithms and evaluate their scalability as a function of the size of the input networks. We analyze PPI networks from six organisms, and report the running times of the algorithms on these networks. We find that some algorithms are better for scalability, while other algorithms are better for effectiveness. We find Fast Greedy to be reasonably scalable, while offering the best modularity scores. Therefore we analyze the significance of the discovered clusters by comparing its effectiveness in real networks against its effectiveness in random networks.

1 Introduction

An important focus in the field of bioinformatics is the study of physical interactions between proteins. The goal is to understand the protein complexes created by the interlocking of proteins complicated folded shapes [1]. These interactions are captured in protein-protein interaction (PPI) networks, where vertices are proteins and two vertices are connected by an undirected edge if the corresponding proteins interact.

One important task in the study of PPI networks is the automated detection of groups or clusters within the network. This task is known as clustering or community detection [1]. Clustering is particularly useful in PPI networks for finding sets of proteins that have many more interactions among themselves than with the rest of the network. It has been found that most clusters in a PPI network correspond to either protein complexes or functional modules [3]. This is particularly useful to discover the "building blocks" of molecular networks, and then leverage this information in other applications.

In this paper, we evaluate the scalability of several clustering algorithms by running these algorithms on real PPI networks. We employ the network analysis package igraph[1], which contains different clustering algorithms. We obtained the PPI networks from the BioGRID Database of Protein and Genetic Interactions[2] for the following organimsms: yeast, worm, fly, mouse, human, and plant. These are all networks of

[1] http://igraph.org/

[2] http://thebiogrid.org/

different sizes, with different number of interactions. Therefore we run the clustering algorithms on these networks and measure both the effectiveness and efficiency of the algorithms in terms of the size of the network. We evaluate their effectiveness by measuring the modularity of the resulting clustered networks. We employ the modularity function implemented in igraph, which corresponds to the one described by Newman and Girvan [2]. We then evaluate the algorithm that we find to offer the "best" trade-off between running time and effectiveness by running it on random networks.

2 Methods

We employed the following clustering algorithms, which are all implemented in igraph:

- Walktrap
- Leading Eigenvectors
- Fast Greedy
- Label Propagation
- InfoMap

We ran these algorithms on PPI networks for the following organisms:

- Yeast (Saccharomyces cerevisiae)
- Worm (Caenorhabditis elegans)
- Fly (Drosophila melanogaster)
- Mouse (Mus musculus)
- Human (Homo sapiens)
- Plant (Arabidopsis thaliana)

We wrote code to reformat the data, which included creating an edge list for each network by removing unnecessary information and filtering edges to only keep physical interactions. We then wrote code to load the networks as an undirected graph in igraph, run the clustering algorithms, and report the running times and modularity scores.

We found Fast Greedy to be the algorithm that offered the best trade-off between effectiveness and efficiency. Therefore we evaluate this algorithm on an ensemble of random networks to asses the significance of its findings. We generated random networks by taking the network edge-list and shuffling the associations between source and target vertices while preserving the in- and out- degree of each vertex. We took as base the PPI networks obtained from BioGRID. Therefore, we generated random networks for 6 different sizes. Each ensemble of random networks consists of 100 networks.

3 Results

In this section, we present the results of our experiments. As we are interested in the scalability of clustering algorithms, we show the results grouped on network sizes. Because we are interested in clustering algorithms, which care about interactions, we consider the number of edges in each network to be the network size, instead of the number of vertices. We present the number of vertices and number of edges of each network in Table 1.

Organism	**Number of vertices**	**Number of edges**
Worm	3288	6353
Plant	7200	21536
Mouse	8317	22820
Fly	8077	37611
Yeast	6341	138856
Human	18747	243671

Table 1: Network sizes.

In Figures 1 and 2 we present the running times and modularity scores [2], respectively. As can be seen, some algorithms such as InfoMap and Walktrap are heavily affected by the size of the network, as their running times increase significantly with the size of the network. We can also see that Label Propagation is impacted in terms of effectiveness, as its modularity score is very low in the biggest networks, which is not the case for the other algorithms. However, Label Propagation is very efficient, requiring less that 1 second to run, regardless of the network size. Leading Eigenvectors is not impacted at all by the sizes of the networks in terms of running time, which makes it really scalable. Fast Greedy is impacted by the size of the network, however its running time is not as long as other algorithms. Fast Greedy gets the best or second best modularity scores in all networks, which makes it very appealing.

We find Fast Greedy to offer the best trade-off between running time and effectiveness. For this reason, we evaluate the significance of its findings by running it on an ensemble of 100 random networks. These results are shown in Table 2. Fast Greedy proves to be effective, as the modularity in the real networks is bigger than the modularity in random networks, for all network sizes.

4 Conclusions

We evaluated the scalability of several clustering algorithms, by running them on real networks of different sizes. Results showed that some algorithms are affected by the size of the network, in either efficiency, effectiveness, or both. Some algorithms, such as Leading Eigenvectors, proved to be very scalable while obtaining a reasonable modularity score, which makes it appealing to use if analyzing big networks. Fast Greedy offered the best trade-off between efficiency and effectiveness, which is why we chose to evaluate it on random networks. It would be interesting to study more in depth these

Organism	**Real network**	**Random networks**
Worm	0.64	0.48
Plant	0.70	0.34
Mouse	0.63	0.36
Fly	0.48	0.25
Yeast	0.34	0.09
Human	0.32	0.13

Table 2: Modularity scores for the real and ensemble of random networks using Fast Greedy algorithm.

algorithms, to discover the reasons for the obtained results. It would be also interesting to analyze the scalability of algorithms based on other properties other than network size, such as average degree, average betweenness, etc.

References

[1] M. Newman. *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA, 2010.

[2] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review*, E 69(026113), 2004.

[3] V. Spirin and L. A. Mirny. Protein complexes and functional modules in molecular networks. *Proceedings of the National Academy of Sciences of the United States of America*, 100, 2003.

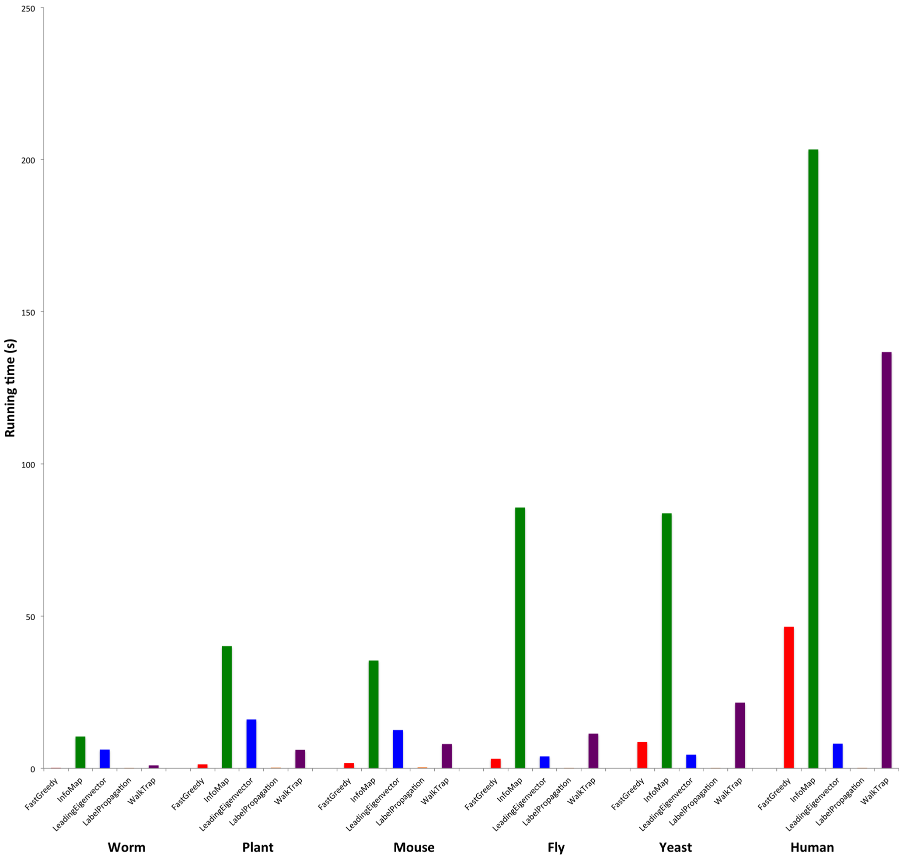

Figure 1: Running times.

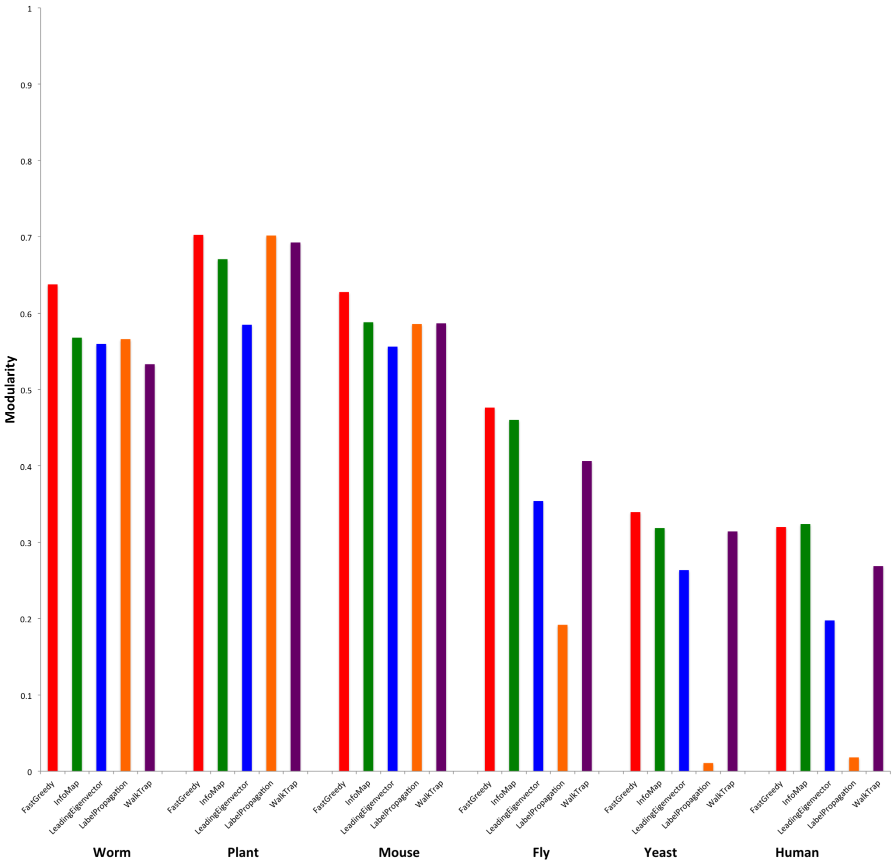

Figure 2: Modularity scores.