Function Guided Clustering of Protein-Protein Interaction Networks

Rob Gevers - Computer Science (rgevers@purdue.edu) , Olga Vitek - Statistics and Computer Science (ovitek@stat.purdue.edu)

3/31/2008

INTRODUCTION

Since proteins perform functions via interactions with other proteins, understanding the functional role of proteins requires an understanding of which proteins interact with each other. These interactions are what we model as edges in Protein-Protein Interaction networks.

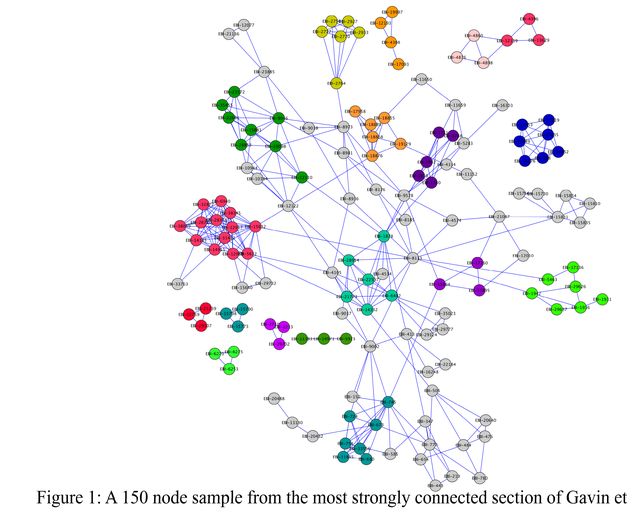

Figure 1: A 150 node sample from the most strongly connected section of Gavin et. Al. 2002 with colored clusters identified by our algorithm.

Once we can identify pairs of interacting proteins, the next step is to identify clusters of proteins which interact with each other.

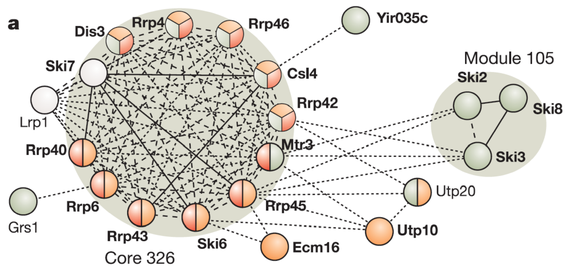

Figure 2: Exosome Protein Cluster from Gavin 2006 dataset

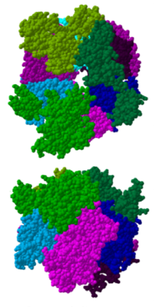

Figure 3: Exosome Crystal Structure

Unfortunately high-throughput data generally has more noise than manually generated data sets. For example, in many techniques cells are destroyed in the process of extracting the proteins contained within them. This allows proteins which aren't normally found in physical proximity to potentially interact. The resulting edge in the graph is a false positive since it represents an interaction which cannot actually occur despite the compatible protein domains.

Goal: To define statistically motivated criteria for determining the point at which a structural clustering no longer identifies biologically relevant clusters, and therefore should be stopped.

BIOLOGICAL DATA

Database

The IntAct database contains a wide array of experimental datasets in a common format with a common set of protein IDs. This makes it ideal for exploring new techniques.

We run our initial analysis on the Gavin 2002 dataset which has 1470 nodes and 3756 edges. Since our clustering is based on network topology, we can isolate sections to cluster independently. We isolate 150 nodes within the largest connected component of the dataset with 656 edges. This allows us to make our initial evaluations more quickly. This sample is not intended to represent the entire dataset.

Gene Ontology

Our functional information comes from the Gene Ontology. GO is a hierarchical approach to organizing functional information. The GO tree has three main branches. As we move down the GO tree the terms become more specific. We use the Lin algorithm, as implemented in the SemSim package in Bioconductor, for calculating semantic similarity based on this tree.

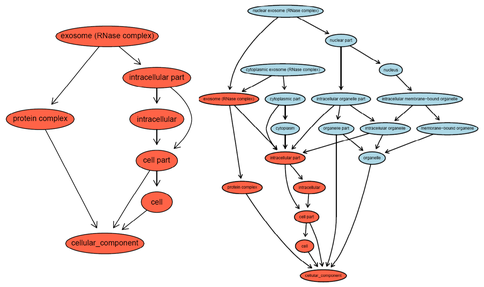

Figure 4 & 5: Exosome GO Single and Multi-Term Sub-Trees

For each GO term *c*, define the probability that the term occurs in this dataset as a probability *p(c)* = (total number proteins mapped to this term)/(total number of proteins.)
For two GO terms, define the probability of minimal subsumer p_{ms} as (total number of proteins mapped to the most specific parent of the terms)/(total number of proteins) or:

$$p_{ms}(c1,c2) = \min_{c \in S(c1,c2)} \{p(c)\} \quad (1)$$

Similarity as defined by the Lin algorithm is then:

$$\mathrm{sim}(c1,c2) = \frac{2 \times [\ln p_{ms}(c1,c2)]}{\ln p(c1) + \ln p(c2)} \quad (2)$$

The goal of using semantic similarity with respect to the cellular component ontology in GO is to have a numerical notion of how close two proteins are within a cell. If all of the proteins in a cluster are very similar, it is more likely that this cluster actually represents a cluster of interacting proteins.

METHODS

Algorithm

We employ a minimum cut algorithm for clustering. A cut is a partitioning of a graph into two sets of nodes. A minimum cut is a cut which removes the fewest edges to create the two sets.

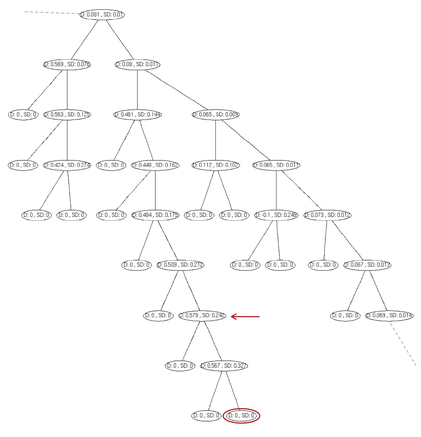

Figure 6: Cut tree showing similarities at each step.

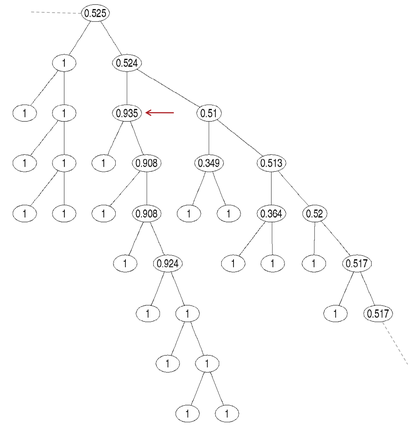

Figure 7: Cut tree showing gap statistic and standard deviation

In this example, the arrow in figure 6 corresponds to the node with the highest gap statistic on a path from root to circled leaf. The highest node with a gap statistic above {max – sd(random)} is identified by the arrow in figure 7.

1. Create a minimum cut partition of the protein interaction graph, starting from the entire graph, and stopping when each protein represents its own cluster.
2. For each node in the cut tree, calculate W = average pairwise functional similarity of proteins in that node.
3. For 1:100
   1. Randomly permute protein identities in the protein interaction graph
   2. For each node in the cut tree, calculate W^i = average pairwise functional similarity of proteins with permuted labels in that node.
4. Calculate the Gap statistic = $(W - W_R)$ where W_R = mean(W^i) and the standard deviation sd(Gap) = sd(W^i) for all i.
5. For each path from root to leaf of the cut tree, prune the branch at the highest node with Gap > max(Gap) - sd(Gap)For selected nodes on a common path resulting from different leaves, select the lowest.

RESULTS

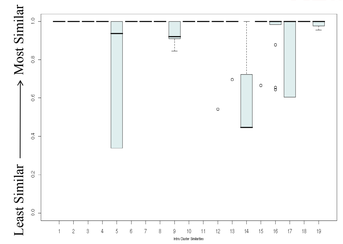

Figure 8: Similarity Within Clusters

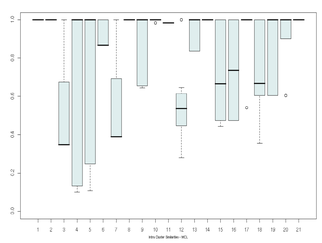

Figure9: Similarity Within Clusters - MCL

Our algorithm produced 19 disjoint clusters. The average similarity within each of these clusters is plotted in figure 8. Figure 9 shows the similarity for the 21 clusters identified by the MCL clustering algorithm without the use of a gap statistic. Our algorithm produced more similar clusters than topology only MCL.

CONCLUSIONS

Our results demonstrate the utility of using a gap statistic with biological similarity measures as stopping criteria for topology based clustering algorithms. For future analyses, additional biological references and datasets will be used to offset the bias introduced by using a single measure of similarity for clustering and validation. Further refinement of the algorithm can also be explored to further fine tune the resulting clusters. The end goal of having the clusters will be to further study the datasets themselves and work toward a statistical notion of biological confidence for the observed interactions.

ACKNOWLEDGEMENTS

We thank Susanne Hambrusch, Sagar Mittal and John Valko for generating the MCL algorithm clusters as well as helpful discussion.

REFERENCES

The Gene Ontology Consortium. Gene ontology: tool for the unification of biology. Nature Genet., 25:25–29, 2000.

Gavin. et. Al. Functional organization of the yeast proteome by systematic analysis of protein complexes. Nature, 415:141–147, 2002.

X. Guo. Gene Ontology-based Semantic Similarity Measures. Bioconductor Library, Oct 2007.

S. Kerrien et. Al. Intact - open source resource for molecular interaction data. Nucleic Acids Research, 2006.

R. Tibshirani, G.Walther, and T. Hastie. A constraint-based framework for diagrammatic reasoning. Applied Artificial Intelligence, 14:327–344, 2000.