Data

We find some slight discrepancies (*in italics*) between the number of interactions and the p-values according to the cumulative binomial distribution. This Table corresponds to Table 1 in Ge et al. (2003).

Table 1: Statistical analysis

Expression-profiling	Protein-interaction				
experiment	data set	Total in map	Expected	Observed	P value
Cell cycle	YPD/MIPS pairs	315	12	42	*1.57×10^{-12}*
	Y2H pairs	305	11	16	*0.11*
	Combined pairs	*600*	22	55	*1.86×10^{-9}*

The P-value calculations used a cumulative binomial distribution

$$P(i \geq i_0) = \sum_{i=i_0}^{I} p^i (1-p)^{I-i} [\frac{I!}{i!(I-i)!}],$$

with

$$p = \frac{\sum_{k=1}^{K} [n_k(n_k-1)/2]}{T(T-1)/2}.$$

In the original cell cycle data clustering analysis, 2945 genes were divided into 30 clusters. We found that several genes were in fact multiply represented. For each multiply represented gene, we determined the cluster to which the gene was closest using Euclidean distance from the cluster mean, and then eliminated the repeated genes. This reduced the total cluster membership as follows in Table 2.

Table 2: Cluster Membership

Cluster	Original	Reduced	Cluster	Original	Reduced	Cluster	Original	Reduced
1	164	157	11	94	94	21	70	68
2	186	185	12	80	79	22	85	83
3	104	103	13	99	96	23	69	63
4	170	169	14	74	73	24	85	83
5	152	151	15	115	113	25	76	74
6	104	100	16	99	98	26	50	49
7	101	101	17	83	81	27	64	63
8	148	148	18	101	96	28	68	67
9	147	146	19	73	73	29	51	51
10	89	78	20	84	84	30	60	59
							2945	2885

Hypergeometric Distribution

In the literature protein-protein interaction list, there were 315 total interactions, 42 of which were between intracluster pairs. Suppose all of the possible pairwise interactions are represented by balls in an urn. Since we have 2885 genes, there are $\binom{2885}{2} = 4160170$ balls in the urn. If all the balls that represent intracluster interactions are red, and the intercluster interaction balls are white, then for the reduced cell cycle data set, there are 156205 red balls and 4003965 white balls. Suppose we select 315 balls at random from this urn. The probability of drawing 42 or more red balls, assuming all balls are drawn independently of each other, can be calculated using the hypergeometric distribution. Specifically,

$$P(\#\text{red balls} \geq 42) = \sum_{i=42}^{315} \frac{\binom{156205}{i}\binom{4003965}{315-i}}{\binom{4160170}{315}} = 1.797187 \times 10^{-12}.$$

We would conclude that it is highly unlikely to observe 42 or more red balls in a random draw of 315 balls.

The hypergeometric distribution is important when we look at the same problem of intracluster pairs in terms of graphs.

Transcriptome/Interactome as Graphs

The problem posed in Ge et al. (2001, 2003) can be phrased in terms of graphs. Figure 1 is a graph representation of the interacting protein pairs in the literature list. Each gene is represented by a node, and if two proteins are known to interact, then an edge is drawn between the two representative nodes. In this picture the names are left off for simpler visualization. In Figure 1, there are 298 nodes and 315 edges corresponding to the number of interacting protein pairs that were observed. There are an additional 2587 nodes of degree zero that are not pictured. These nodes represent genes that were used in the clustering analysis, but were not among the list of interacting protein pairs.

The cluster information can also be represented as a graph. For each cluster, draw an edge connecting a node to any other node with a corresponding gene in the same cluster. For the cell cycle data, the graph would consist of 30 completely connected subgraphs with the number of nodes in each subgraph corresponding to the number of genes in each cluster. Figure 2 is an example of a cluster graph for the first nine members of cluster 1 in the cell cycle data set. The graphs quickly become quite cumbersome to visualize as more nodes are added.

The number of intracluster edges can then be represented by counting the number of edges in the intersection graph. The intersection graph keeps all of the same nodes as in the PPI and cluster graphs, but only retains the edges that exist in both graphs. Figure 3 demonstrates the 42 observed edges among 65 nodes that are reported in Ge et al. (2003).

Statistical Inference for Graphs

Reference Distribution

Using the hypergeometric distribution to evaluate the statistical significance of observing 42 intracluster pairs, or 42 edges in the intersection graph, is equivalent to taking the 315 edges from the observed literature PPI graph, randomly reassigning the edges to different node pairs, and then counting the number of edges in the intersection graph of the randomized PPI graph and the cluster graph. Doing this many times will result in a reference distribution for the number of edges in the intersection graph. Figure 4 shows one example of the random reallocation of the edges in the PPI graph. There were 8 edges in the intersection of this graph with the cluster graph, the arrangement of which are shown in Figure 5. We will call this algorithm of permuting the edges the PE method.

Notice that the graph in Figure 4 is composed of several components consisting of a very small number of nodes. Given the number of nodes and the number of edges that we are dealing with, this is consistent with the Erdos-Renyi theory of random graphs. The degree distribution for the random edge graph is exponential; however, the degree distribution for the PPI graph follows a power law (see Figure 6). As suggested in several other studies of the structure of PPI networks, our PPI graph appears to be scale-free. There is possibly cause for concern in using a random edge model as a basis for statistical inference since such models are likely not representative of actual PPI networks.

An alternative procedure for generating a reference distribution would be to retain the edge structure of the graph, and randomly permute the node labels (which we will call PN). This would guarantee that the structure of the PPI graph is from the sample space of possible PPI graphs. The question of interest is whether or not interacting protein pairs tend to come from the same cluster. In a random permutation, the node labels are not preferentially assigned to connected nodes, and so this leads to a natural reference distribution.

Test Statistics

Somewhat surprisingly, the PE and PN models result in similar distributions of the number of edges in the intersection graph. (See Figure 8.) It is possible, however that the number of edges in the intersection graph may not be the most descriptive test statistic. Figure 7 shows a graph with 42 edges and 84 vertices that would give the same p-value result as the observed intersection graph in Figure 3. These two graphs are quite different in terms of the arrangement of the edges. Figure 3 demonstrates a tendency toward connected components with greater numbers of nodes. Test statistics involving node degree and edge structure may give more insight into the structure of the intersection graph, and possibly the relevant biology.

For test statistics on the intersection graph other than the number of edges, Figure 8 shows that the PE and PN methods do give strikingly different distributions. Specifically, node degree ≥ 2, node degree ≥ 3, and the number of 3-cycles all have much different distributions depending on the algorithm. The PN method tends to give nodes with higher degree, and picks up more 3-cycles. Both of these features are evident in the observed intersection graph in Figure 3.

We also evaluated the number of connected components as a test statistic using the PE and PN algorithms. Although the distributions in Figure 8 are not strikingly different, there is some evidence that the PN algorithm tends to give fewer connected components in the intersection graph. This corresponds to finding larger groups of genes, possibly functional modules, that are connected in both the PPI and the cluster graph.

Statistical Theory

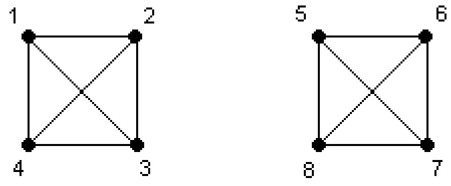
Fisher's Exact Test

Fisher's exact test is a classic statistical method for assessing independence between outcomes. The representation of the usual Fisher's exact test as graphs lends credence to conditioning on the structure of the graph when generating a reference distribution for statistical inference. Suppose we are interested in clustering eight genes into two distinct groups. We apply two different clustering methods to the data for these genes, and find the following two groups:

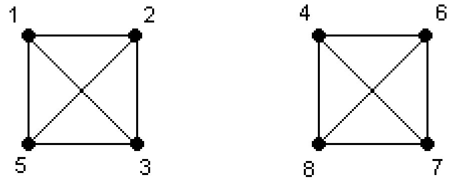
Clustering method 1: {1,2,3,4},{5,6,7,8}
Clustering method 2: {1,2,3,5},{4,6,7,8}

We can represent these clusters as graphs. Each gene is represented by a node, and edges connect genes that are in the same cluster.

Cluster graph 1:

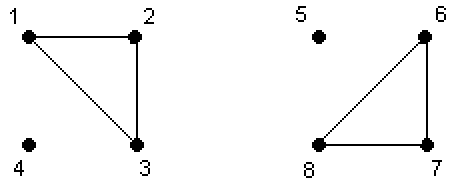


Cluster graph 2:



A question of interest may be whether the edges in the graph from clustering method 1 are overrepresented in the graph from clustering method 2. That is, are the two categorizations independent? For this simple example, the intersection between the two graphs is as follows.

Intersection:



In order to assess whether the number of edges in the intersection graph is more than would be expected by random chance, we might consider using the hypergeometric distribution with (28,12,12) as parameters. Suppose there are 28 balls in an urn, 12 of which are red and 16 of which are white. We then draw 12 balls from this urn. The number of red balls in that draw would represent the number of edges in the intersection graph. Using the hypergeometric distribution for inference is equivalent to generating a reference distribution for the number of edges in the intersection graph by using a random edge model.

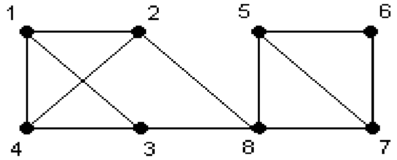
Consider a random edge model for generating a reference distribution for the number of intersecting edges in $G_f(V, E_f)$ $(|E_f| = n_f)$ and $G_r(V, E_r)$ $(|E_r| = n_r)$. In this model, one of the graphs is fixed, say G_f, and the edges of the second graph, say G_r are randomly reassigned. The number of intersecting edges, X, then follows a hypergeometric distribution with parameters $N = \frac{|V|(|V|-1)}{2}$, n_f, and n_r. Specifically,

$$P(X = i) = \frac{\begin{pmatrix} n_f \\ i \end{pmatrix}\begin{pmatrix} N - n_f \\ n_r - i \end{pmatrix}}{\begin{pmatrix} N \\ n_r \end{pmatrix}} \qquad i = \max(0, n_r - (N - n_f)), ..., \min(n_r, n_f).$$

In the case of the two cluster graphs, the number of edges in the intersection graph under the random edge model follows a hypergeometric distribution with parameters $N = 28$, $n_r = 12$, and $n_f = 12$.

The random edge model for the graph highlights an important problem in using the hypergeometric distribution for inference. Specifically, the structure of the graph is not preserved, and the random edge-generated reference distibution includes observations that are outside of our sample space. For example, the random edge model might result in the following randomized cluster graph 2.

Random edge allocation on cluster graph 2:

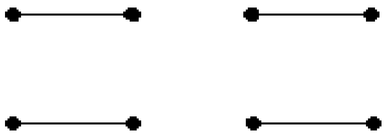


This particular random edge allocation is not possible in the framework under which we are working, and so we would not want to use the random edge model as a basis for inference.

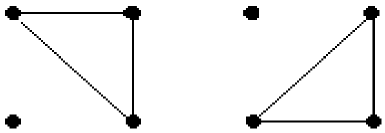
Consider instead a random node model for the cluster graph. Again, fix cluster graph 1, and randomly permute the nodes on cluster graph 2. The new intersection graph is an observation from the reference distribution which we will use for inference. Note that the structure of the graphs is always preserved under the random node model.

The random node model allows for 3 possible intersection graphs with 4, 6, and 12 edges respectively.

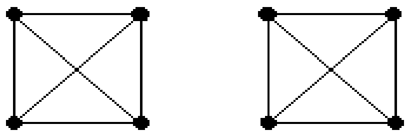
Intersection graph with 4 edges:



Intersection graph with 6 edges:



Intersection graph with 12 edges:



Simulations demonstrate that the number of the edges in these graphs corresponds to Fisher's exact test. This can also be proven analytically. This result makes sense if we enter the genes into a 2x2 table using rows and columns as the categorizations using each clustering method, and perform Fisher's exact test in the usual way.

	X	Y	
A	3	1	4
B	1	3	4
	4	4	8

Other Statistical Theory Aspects to Consider

The simple demonstration of the PN model and Fisher's exact test motivates conditioning on the structure of the graphs in other settings for doing inference on the relatedness of multiple graphs. In the 2×2 table formulation of Fisher's exact test, the marginal totals are ancillary, and so inference is made conditional on these statistics. Since we can generate the same distribution by conditioning on the structure of the two cluster graphs and permuting the nodes, this suggests that the observed structure of the graphs may in some sense also be ancillary.

There are many other investigations into latent-variable type models for random graphs that may help frame hypotheses of interest in terms of reasonable parameters, suggest sufficient and ancillary statistics, and lay further groundwork for doing statistical inference on multiple graphs.

References

Ge, H., Liu, Z., Church, G. & Vidal, M. (2001). Correlation between transcriptome and interactome mapping data from *Saccharomyces cerevisiae*. *Nature Genetics* **29**, 482–486.

Ge, H., Liu, Z., Church, G. & Vidal, M. (2003). Does mapping reveal correlation between gene expression and protein-protein interaction? -In reply. *Nature Genetics* **33**, 16–17.

Figure 1: Observed Literature Protein-Protein Interaction Graph

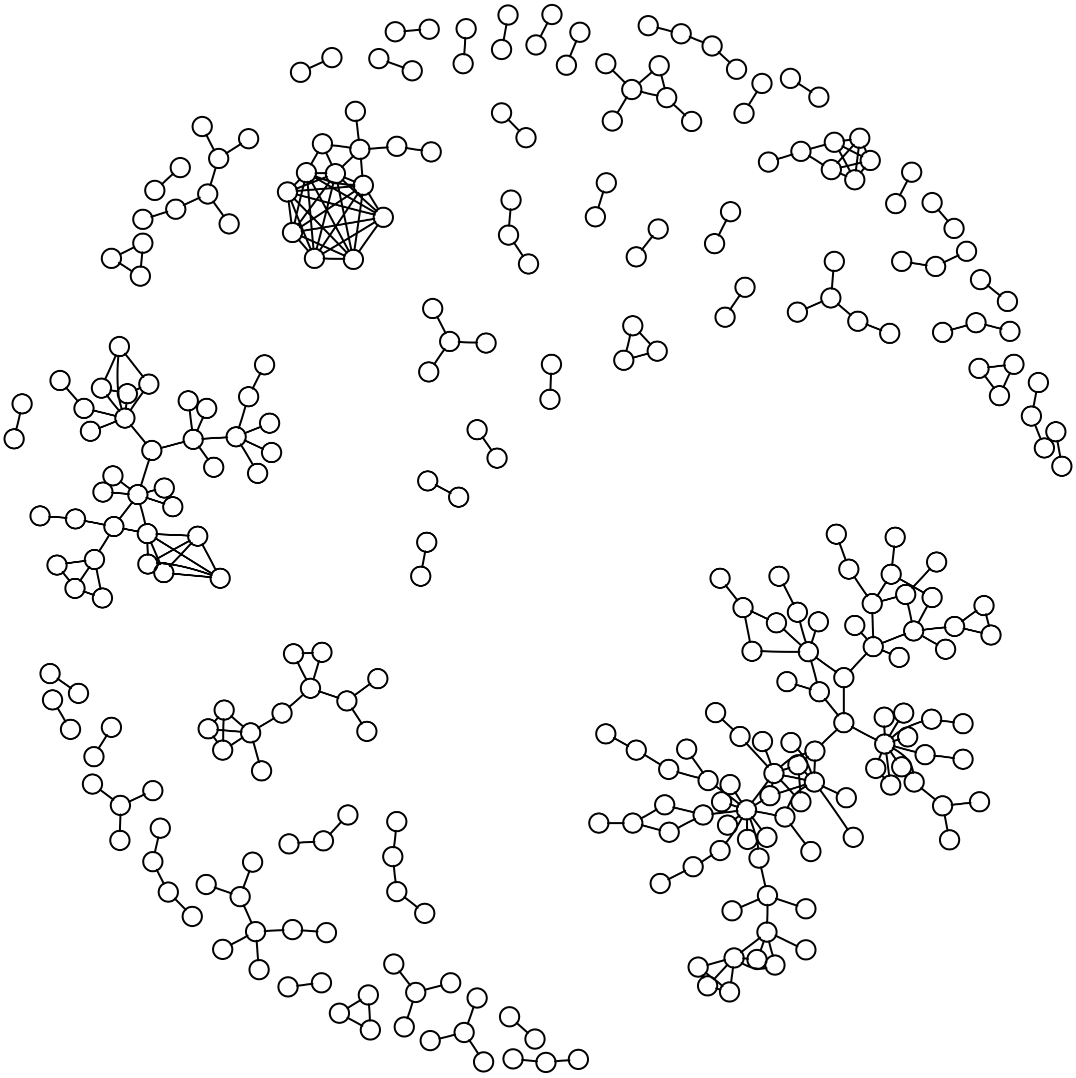


Figure 2: Part of the Subgraph for the Genes in Cell Cycle Cluster 1

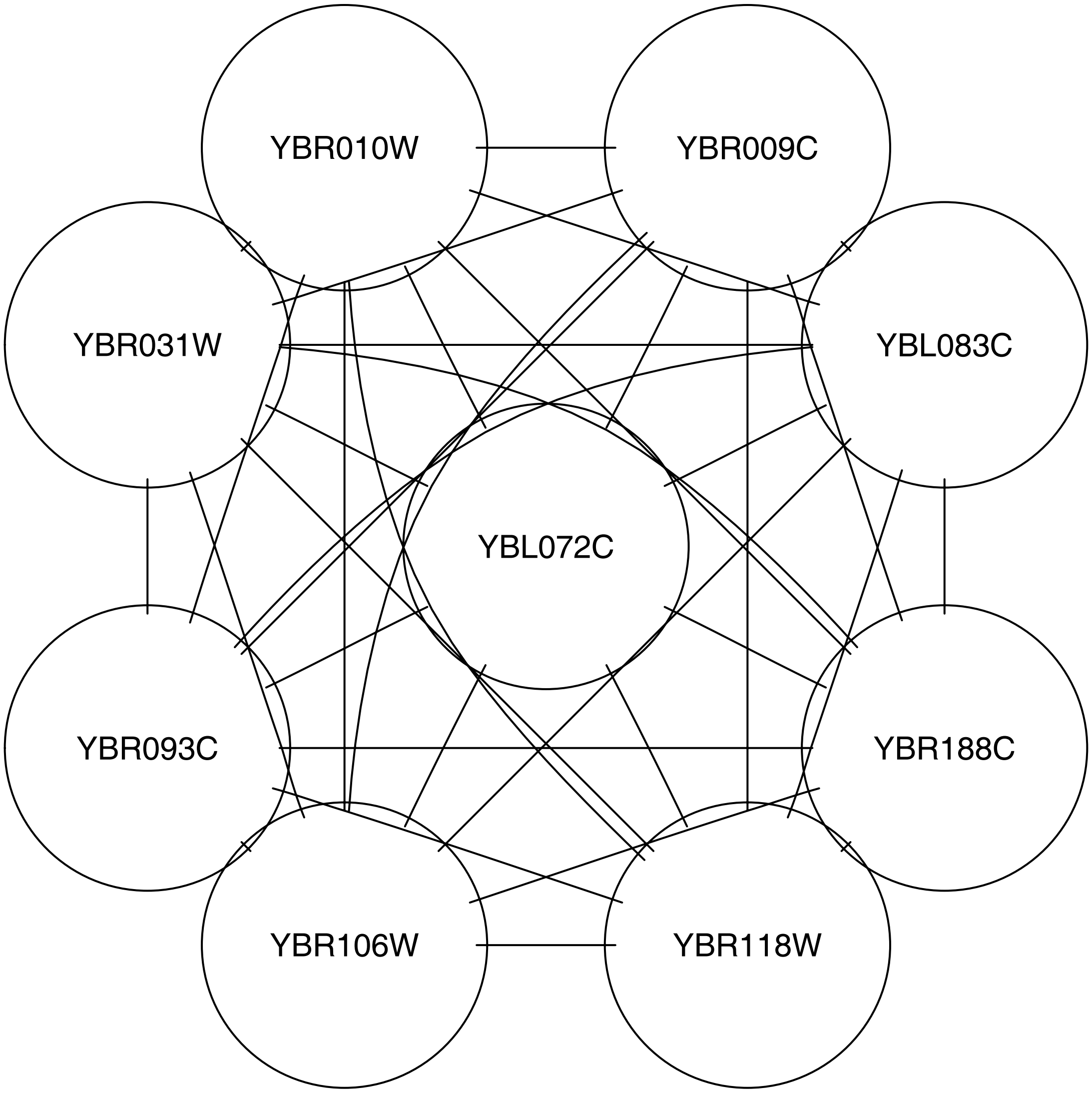


Figure 3: Observed Intracluster Edges in Ge, et al. (2003)

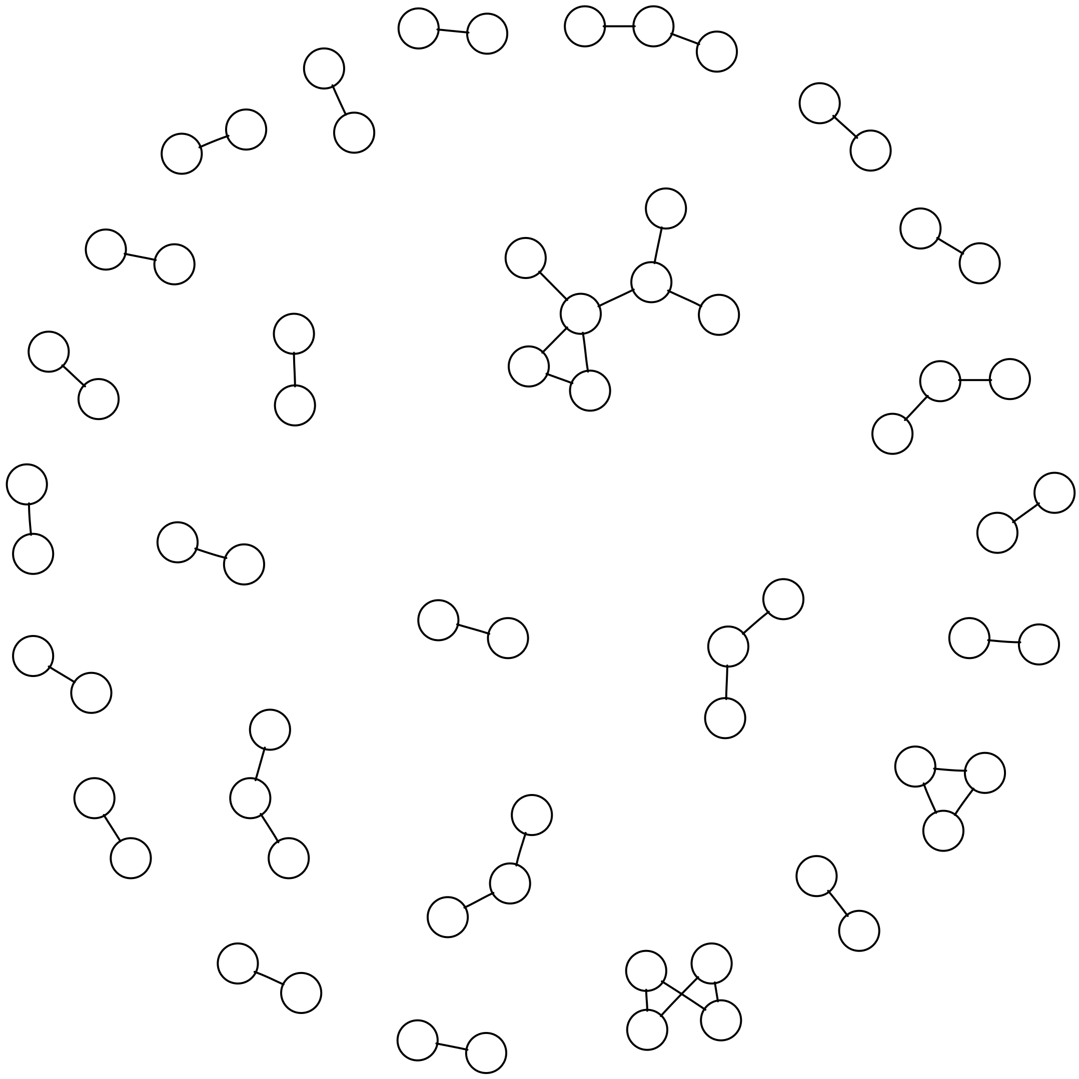


Figure 4: Randomly Reassigned Edges in PPI Graph

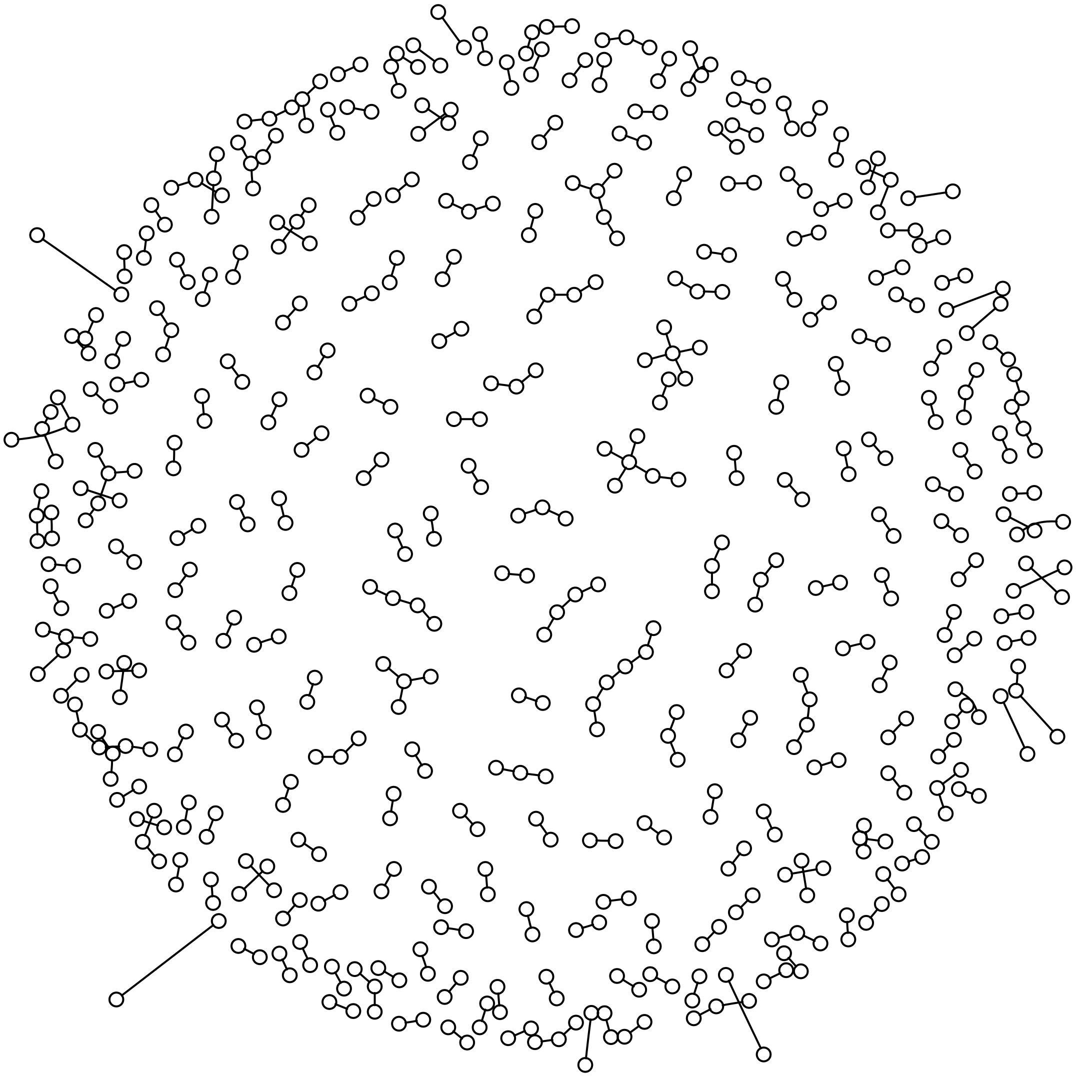


Figure 5: Intersection Graph after Random Edge Reassignment

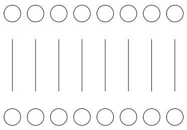


Figure 6: Node Degree Distributions of PPI Graphs

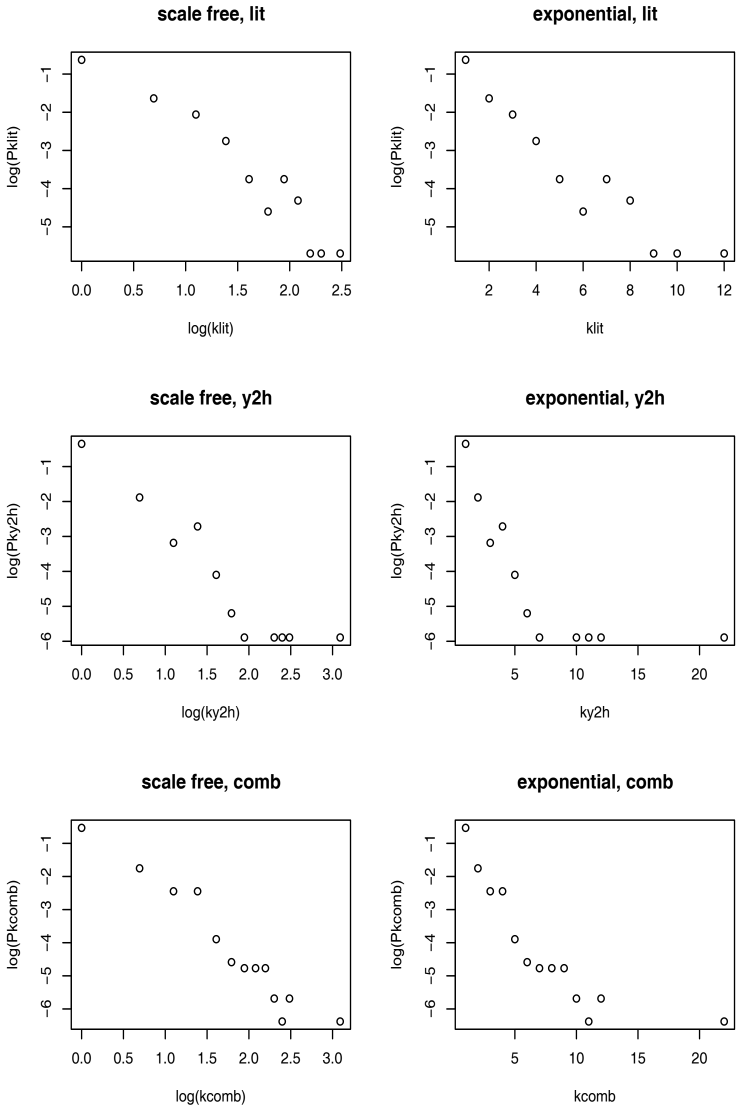


Figure 7: Graph with 42 Edges and 84 Nodes

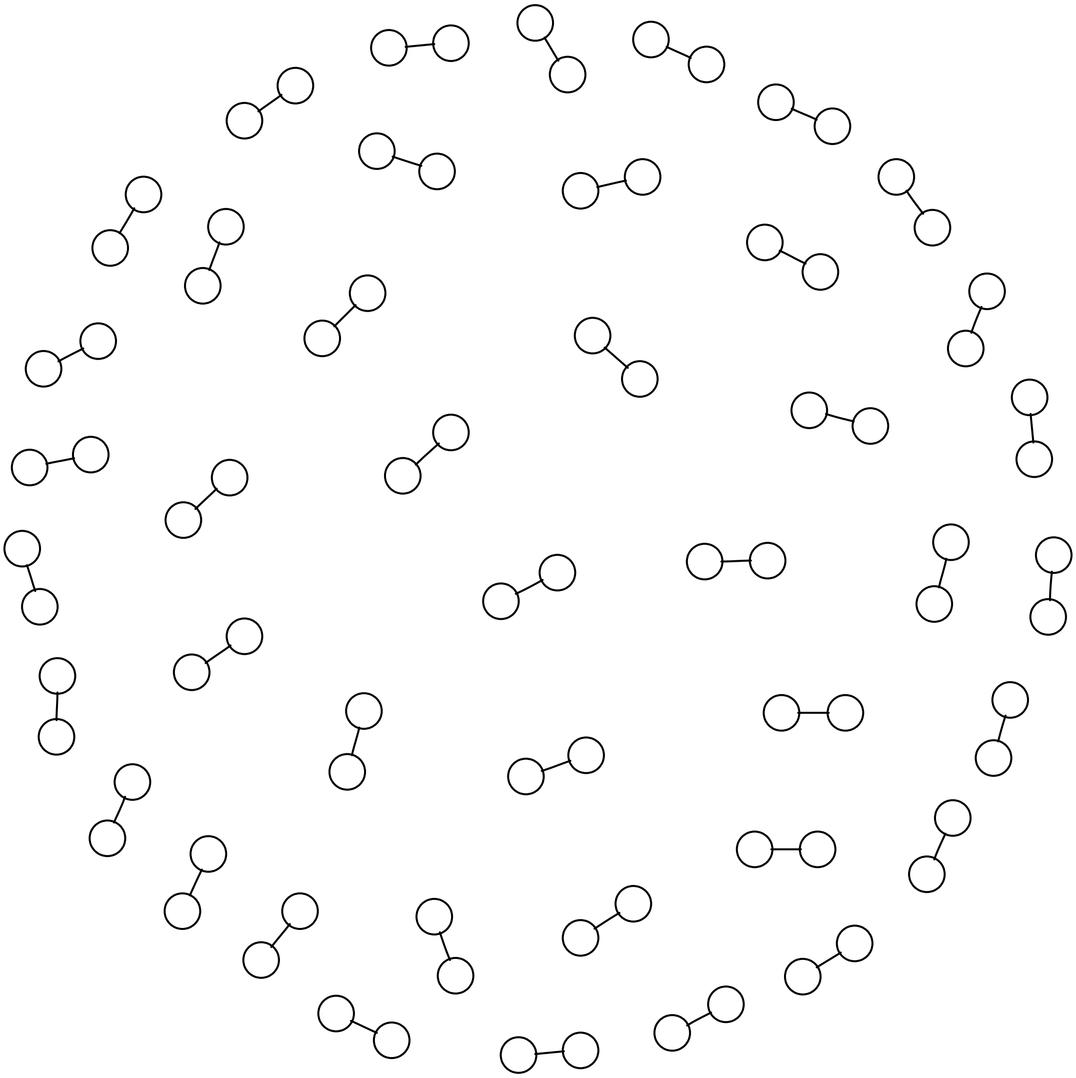
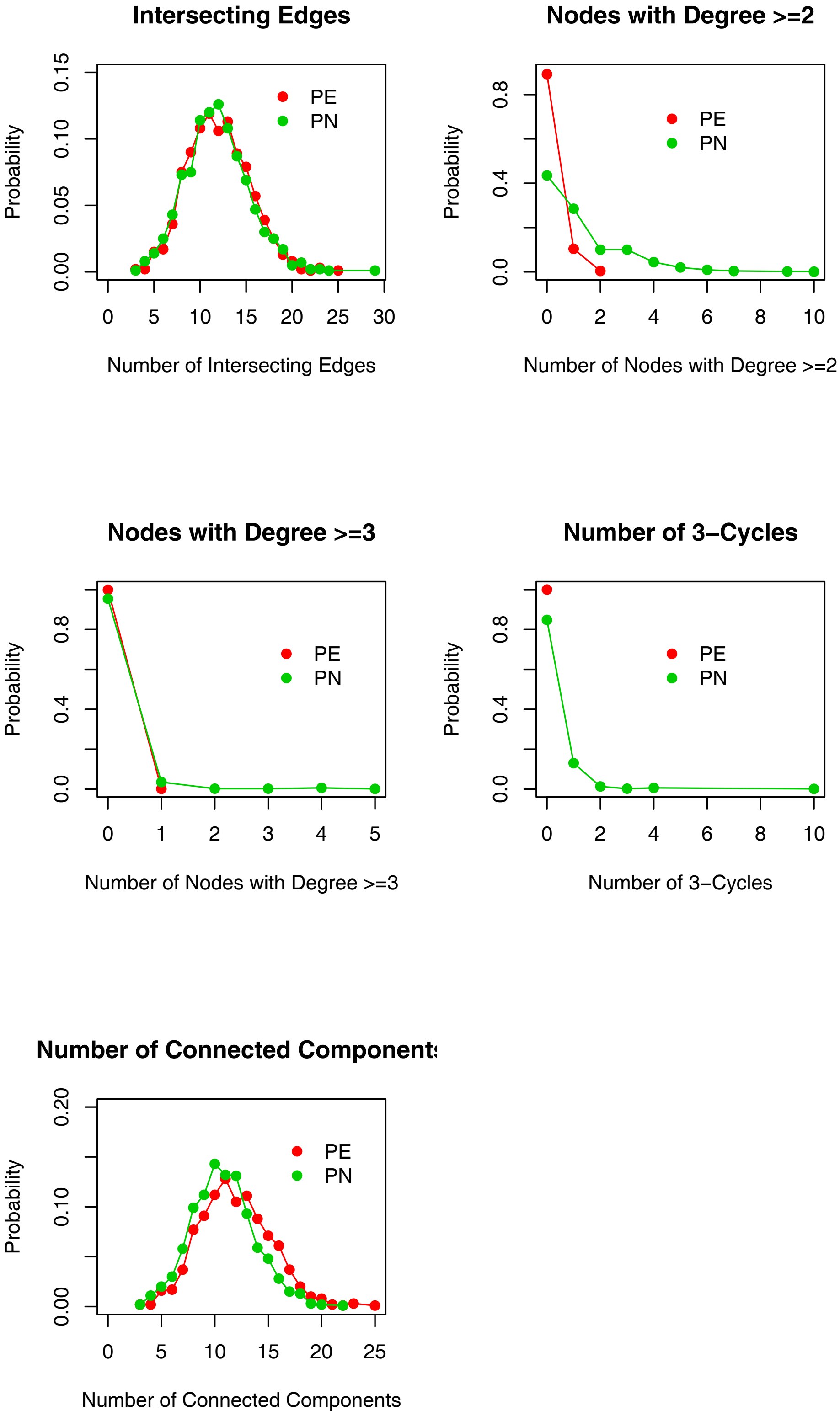


Figure 8: Test Statistics


Intersecting Edges
Probability
Number of Intersecting Edges
● PE
● PN
Nodes with Degree >=2
Probability
Number of Nodes with Degree >=2
● PE
● PN
Nodes with Degree >=3
Probability
Number of Nodes with Degree >=3
● PE
● PN
Number of 3−Cycles
Probability
Number of 3−Cycles
● PE
● PN
Number of Connected Components
Probability
Number of Connected Components
● PE
● PN