Supplementary Methods for:

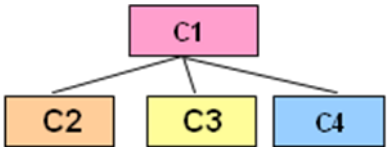
A Probabilistic Generative Model for GO Enrichment Analysis

Examples for the likelihood function used by our model

To illustrate the likelihood function we are using (Equation 1) consider as an example the set of genes identified in an experiment arresting cells at the S phase of the cell cycle. Both the 'S phase' and the 'Cell cycle' categories would be enriched with such genes. However, since 'S phase' is more specific and fully captures the condition, we would like our GO analysis tool to return 'S phase' rather than the more general 'Cell cycle'. This is exactly the idea captured in the likelihood term of Equation 1. If we select 'Cell cycle' then (assuming low noise) most of the selected gene nodes (A_g) would be accounted for. However, many cell cycle genes participate in other cell cycle phases and so S_g would also be large which will reduce the likelihood (when $p \geq 0.5$). In contrast, selecting 'S phase' would still lead to large value for A_g but a much smaller value for S_g resulting in a higher likelihood. Note that selecting a category that does not reflect the condition (for example, 'Metabolism') would lead to small values for A_g and large values for S_g and A_n, again reducing the likelihood function (since $p >> q$).

Below we present another example which uses concrete numerical values to illustrate the differences between the results of our method and the results of the classic hypergeometric p-value calculation.

Assume the total number of genes is 10,000. We consider four categories, C1 and its three direct decendents: C2, C3, and C4 (see figure below). The number of genes annotated to each category is as follows |C1| = 100, |C2| = 10, |C3| = 10, |C4| = 80



Assume that under certain experimental condition, 50 genes are determined to be active. Among these genes, 20 are in category C1, 9 in C2, 9 in C3, and 2 in C4. Intuitively, in this experiment the active biological processes are the functions encoded by C2 and C3 because most genes in both categories are active. On the other hand, the category C1 is probably not active since otherwise we would expect all its descendents, including C4, to contain active genes.

Analysis of this example using the classic method and GenGO leads to different conclusions. If we use the classic method and compute p-values based on the hypergeometric distribution, three categories, C1 (p-value=5*10^-28), C2 (p-value=10^-20), and C3 (p-value=10^-20), will be determined to be significantly enriched. C4 is not significantly enriched (p-value=0.06). C1 is the most significant category with the smallest p-value.

The answer is different if we use GenGO. Specifically, we look at the log-likelihood for the following three cases:

1) C1, C2, and C3 are active, C4 is inactive. In this case, |Ag| = 20, |An| = 30, |Sg| = 82, |Sn| = 78. The log-likelihood achieves its maximum at -123.3 when p=0.20 and q=0.28.

2) C2 and C3 are active, C1 and C4 are inactive. In this case, |Ag| = 18, |An| = 32, |Sg| = 2, |Sn| = 158. The log-likelihood achieves its maximum at -98.6 when p=0.90 and q=0.17.

3) Only C1 is active. In this case, |Ag| = 20, |An| = 30, |Sg| = 80, |Sn| = 80. The log-likelihood achieves its maximum at -117.5 when p=0.20 and q=0.27.

The likelihood for case 2 is the highest. As a result, the GenGO algorithm will correctly determine that only C2 and C3 are active.

Finding the optimal set of GO terms is NP hard

Here we show that the task of finding the optimal set of active GO nodes (that is, a set that will maximize the likelihood of our target function specified in Equation 1) is a NP hard problem. To prove this we reduce the Minimum Set Covering (MSC) problem to our problem. Given an MSC instance $\{U,F\}$, where U is a set and F is a family of subsets of U, we construct an activation graph as follows: For each element $u \in U$, we add a gene node g_u to the graph; for each subset $f \in F$, we add a GO node g_f and connect it with gene nodes g_u if and only if $u \in f$. In addition, we include all gene nodes in the active set, and let p=1, q=0, and $\alpha = 1$. In other words each gene node has to be explained by at least one GO node (q does not matter in this case since all genes are 'active'). Maximizing the log likelihood for this graph is equivalent to finding a minimal set C of GO nodes such that every gene node in the graph is connected to at least one node in C. The requirement for a minimal set is achieved via the penalty term $\alpha |C|$. If we can find the solution C, we can recover the solution to the original MSC instance by taking the subsets in F that correspond to a node in C.

Finding the best GO set by learning parameters *p* and *q*

These two parameters can also be optimized by maximizing the log likelihood defined in Equation 1. The algorithm is as follows:

Algorithm 2 (Find the best GO set by learning parameters *p* and *q*)

(1) Initialization. Set p_0=0.5, q_0=$|G|/|R|$, where G is the set of active genes, and R is the reference set .

(2) Carry out steps in Algorithm 1, using p_i and q_i.

(3) Based the solution found in the previous step, we compute the maximum likelihood estimation of p and q: $p_{i+1} = \frac{|A_g|}{|A_g| + |S_g|}$, $q_{i+1} = \frac{|A_n|}{|A_n| + |S_n|}$.

(4) if $\max(|p_{i+1} - p_i|, |q_{i+1} - q_i|) \geq \varepsilon$, go to step 2, otherwise stop. (ε is a small positive number to control convergence.)

Because both steps in Algorithm 1 and 2 only increase the likelihood, the algorithm above is guaranteed to converge to a local maximum.