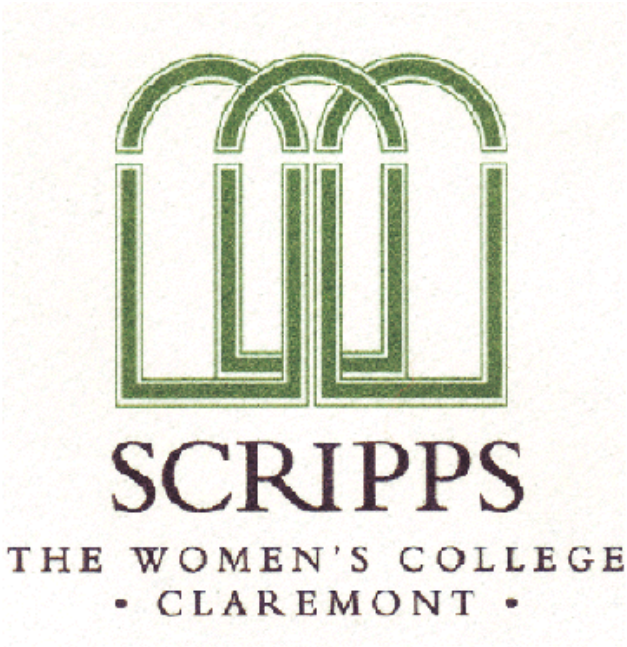

Determining Overrepresentation of Gene Ontology Terms using the Hypergeometric Distribution

Christine Ju

Professor Jo Hardin, Advisor
Professor Anie Chaderjian, Reader

Submitted to Scripps College in Partial Fulfillment
of the Degree of Bachelor of Arts

April 15, 2011

Department of Mathematics

Abstract

Gene ontology (GO) terms describe functions of genes and may occur in different amounts due to differing conditions. The hypergeometric distribution can be used to determine whether or not a GO term is overrepresented. Two approaches that apply the hypergeometric distribution are the *term-for-term* and *parent-child* methods. The *term-for-term* approach detects overrepresentation of GO terms individually; however, some terms are falsely highlighted as being overrepresented due to the complex relationships between terms which leads to inaccurate biological interpretations. The *parent-child* method addresses these issues by taking into account the relationships of GO terms with one another, allowing for more accurate analyses.

Contents

Acknowledgments

I would like to thank Professor Jo Hardin for her wonderful support, guidance, and patience. This thesis could not have been accomplished without her.

Chapter 1

Introduction

Differential expression of genes is part of a process in which cells from a species attempt to adapt to changed environments. Microarray experiments serve to measure discrepancies in gene expression for cells raised in differing conditions compared to cells raised in normal conditions[1].

Suppose researchers want to identify the difference in the level of gene expression between normal yeast cells and nutrient-deprived yeast cells. The genes from the two types of cells will form two different samples, and the difference in gene expression in the two samples can be compared by using microarrays. The genes that are differentially expressed by t-tests will form the study set while the population set will consist of the genes that are of interest or all the genes in the yeast cell.

Gene ontology (GO) terms provide a universal language for researchers to use that categorizes each indiviual gene and its related function. When many or all of the GO terms are combined, a directed acyclic graph forms to show the relationships between the GO terms and their connections to surrounding terms, illuminating parent and child relationships. The graph demonstrates how GO terms are not independent of each other, and in fact, each term is dependent on many other terms[2]. The *true-path-rule* states that if a gene annotates to a certain GO term, then that gene will also annotate to all of the parent terms, or less specific terms, of that GO term[3]. For example, *regulation of immune response* is the child of *immune response*.

All genes that annotate to *regulation of immune response* will also annotate to *immune response*. The existence of the *true-path-rule* can create issues when statistical tests are used to determine whether or not a gene ontology term is overrepresented, or occurs more often than expected by chance.

The hypergeometric distribution is used to determine if a GO term is overrepresented in a study set[3]. The hypergeometric distribution assumes that each element that is taken into consideration is independent of all other elements[4]. However, complications arise when determining overrepresentation of more than one GO term. It is reasonable to use a simple hypergeometric distribution for just one GO term; however, determining overrepresentation of the next GO term is not independent from the first GO term because it may be the parent or child of the previous GO term. If one GO term is overrepresented, its children terms may also have a higher chance of being overrepresented.

Two methods that use the hypergeometric distribution to find overrepresentation of GO terms, analyzed by Grossmann et al. [2007] are the *term-for-term* and *parent-child* approach. While the *term-for-term* approach does not account for the dependency issues between parent and child GO terms, the *parent-child* approach does. The *term-for-term* approach analyzes each GO term separately from all other terms and applies the basic form of the hypergeometric distribution. However, since the *parent-child* method takes the relationship between parent and child terms into account, the hypergeometric distribution that is used is slightly altered by conditioning the term of interest on its parents. A comparison of these two methods shows that the *parent-child* method is superior to the *term-for-term* approach since it accounts for the interrelatedness of GO terms[3].

The remainder of this paper is organized as follows: In Chapter 2, background information will be given on gene ontology terms and microarrays. In Chapter 3, the hypergeometric distribution will be explained. Overrepresentation of GO terms and the simulation done by Grossmann et al. [2007] will be covered in Chapter 4 followed by the issues of multiple testing in Chapter 5. Finally, the biological significance of choosing between the *parent-child* and *term-for-term* approach will be addressed in the conclu-

sion.

Chapter 2

Background

2.1 Gene Ontology

Gene ontology is an attempt to provide a uniform record of all known genes and their functions by annotating the genes to certain gene ontology categories (or terms)[2]. Having such a catalog allows for examination of the function of genes throughout different experimental conditions, by providing researchers a specific language to use when examining genes for the particular goals of the study. Examples of gene ontology terms are *DNA replication*, *DNA binding*, *immune response* and so forth.

There are three main subontologies of GO terms: *biological process*, *molecular function*, and *cellular component*. All genes annotate to at least one term in each subontology[2].

The first subontolgy, *biological process*, describes the biological outcome that usually involves a chemical or physical transformation[2]. Some examples of terms that fall under *biological process* include *cell growth and maintenance* and *translation*. The terms that fall under biological process describe the biological outcome.

The next subontology, *molecular function*, describes the chemical component of a biological process such as the specific bindings to ligands[2]. Examples of terms that fall under *molecular function* include *enzyme* and *adenylate cyclase*.

The last subontology, *cellular component*, describes where the biological process takes place[2]. Examples include the *ribosome* and the *nuclear membrane*.

All GO terms are connected in a network, such as a directed acyclic graph[2]. The graphs identify the connections between the parent and child terms. The parent terms are broader, whereas the child terms describe specific aspects of the parent term. For example, *regulation of immune response* is a child of *immune response*. A child term can be a parent for a different term and a parent can be a child of a different term. For example, *response to stress* is the child of *response to stimuli* and the parent of *response to wounding*. GO terms may also have more than one parent[2]. *DNA ligation* is the child of *DNA replication*, *DNA repair*, and *DNA recombination*.

The complex relationships between the GO terms can make analyses of overrepresentation of particular GO categories much more difficult.

2.2 Microarrays

Genes are segments of DNA that code for proteins. The formation of protein occurs through two processes: transcription and translation. Transcription is where DNA is transcribed, and mRNA, the complementary strand of DNA, is formed. During translation, mRNA is used to produce an amino acid chain that eventually becomes a protein[5].

The purpose of using DNA microarrays is to look at the expression levels of many genes at the same time under particular experimental conditions[1]. Different gene expressions occur when specific genes are turned on or off at different times. One purpose of using microarrays is to identify a subset of genes that are differentially expressed under two conditions by comparing every gene of interest[1].

When using DNA microarray, the genes of interest which contain many individual DNA sequences, are printed on a chip. In order to compare the genes of the two different samples, first mRNA is isolated and cDNA is produced by reverse transcription. The cDNA is labeled with different fluorescent dyes for the two samples and are merged and matched onto their

respective locations to the DNA on the chip. The intensity of the fluorescent dyes between the two samples identifies which genes are differentially expressed since the amount of dye varies for each gene in the two samples. The intensity of the dye can convey how much that DNA sequence is replicated (and therefore producing mRNA and later amina acids). If the intensity of the dye is strong, then it can tell whether any of the terms occur more than expected or if the terms are being overrepresented[1].

Chapter 3

Hypergeometric Distribution

The hypergeometric distribution describes the probabilities associated with sampling randomly without replacement from a finite population where all elements have an equal chance of being drawn[4]. Because elements are not being replaced, each selection influences the number of individuals of a certain type that are left in a population, making each selection dependent on the previous selections, unlike the binomial distribution which is based on independent trials.

3.1 Distinct Elements

In order to understand the hypergeometric distribution, first consider a population with a certain number of unique elements. Suppose elements are being drawn one by one randomly without replacement. Let N be the population size and let n be the sample size where $n<N$. Next, let $(N)_n$ be the number of different possible orderings of n out of N elements[4].

$$(N)_n = N(N-1)(N-2)...(N-n+1) \tag{3.1}$$

Choosing is the process of selecting elements from a group of elements where order does not matter. Ordering, however, does take into account which element is chosen first, second, etc. From Equation (3.1), we have N

different ways to select the first element. After the first unique element has been drawn, we only have $N-1$ left and after the second element has been drawn we only have $N-2$ unique elements left and so forth. For the last draw, we will have $N-n+1$ elements left. For example, suppose we have 20 distinct elements and we want to order all 20 of them. The last draw will only have 20-20+1=1 element left; therefore, there is only one way to select the last element if all elements are being drawn.

An example: Suppose you have 20 different elements. Let the elements be different colored balls (red, blue, green, purple...). How many ways are there to order 3 out of the 20 elements without repeating any colors? Here *N*=20 and *n*=3.

$$\begin{aligned}(20)_3 &= 20(20-1)(20-2) \\ &= 20(19)(18) \\ &= 6840\end{aligned}$$

There are 6840 ways to select 3 different colored balls out of 20, where order matters.

3.2 Elements of the Same Type

The number of ways of choosing n out of N where order doesn't matter is denoted as $\binom{N}{n}$ called *N choose n*[4].

$$\binom{N}{n} = \frac{N!}{n!(N-n)!} \tag{3.2}$$

In Equation (3.2), *N!* gives all the ways to order N things, as shown in Equation (3.1). Consider the *N* elements broken into the first *n* and the next $(N-n)$. Because order does not matter, we can choose the same n items first in $n!$ different ways. So, we divide by *n!* because within *N!* there are *n!* repeats of the items of interest (the chosen ones) and $(N-n)!$ repeats of the items that are not of interest (the unchosen ones). A repeat occurs when

the same items are drawn but in a different order.

The number of ways to choose n elements from N elements is $\binom{N}{n}$ and the number of ways of ordering the elements is $n!$[4]. Multiplying the two together will give the number of different possible orderings of n out of N elements:

$$(N)_n = \binom{N}{n} n! \tag{3.3}$$

Another way of writing the equation for $\binom{N}{n}$ is by rearranging the equation above:

$$\binom{N}{n} = \frac{(N)_n}{n!} \tag{3.4}$$

Now instead of having all different elements, let the population consist of only green, G, and blue, B, balls, where $G+B=N$ is the total number of balls in the population. One way of getting g green and b blue balls such that $g+b=n$ is by drawing g green balls in the first g trials and then drawing b blue balls in the next b trials. Consider selecting n balls from N:

$$\begin{aligned} P(g\ green\ followed\ by\ b\ blue\ balls) &= (\frac{G}{N})(\frac{G-1}{N-1})...(\frac{G-g+1}{N-g+1})(\frac{B}{N-g})(\frac{B-1}{N-g-1})..\\ &.. \quad (\frac{B-b+1}{N-g-b+1}) \\ &= \frac{(G)_g(B)_b}{(N)_n} \end{aligned} \tag{3.5}$$

Note that Equation (3.5) gives the probability for some explicit order or only one of the ways to draw g green and b blue balls. This means that the $P(1\ green,\ b\ blue,\ (g-1)\ green)$ will also be the same as Equation (3.5). However, there are many different possible ways of drawing the balls. In order to find the probability of g green and b blue in any order, the previous equation needs to be multiplied by $\binom{n}{g}$, the number of different combinations[4]. We can think about the number of different combinations the following way: we arrange the n items by picking the g spots for green

and filling in the remaining $(n - g)$ or b spots for blue. This will give the equation:

$$\begin{aligned} P(g \text{ green and } b \text{ blue balls in any order}) &= \binom{n}{g} \frac{(G)_g (B)_b}{(N)_n} \\ &= \frac{n!}{g!b!} \frac{\binom{G}{g} g! \binom{B}{b} b!}{\binom{N}{n} n!} \\ &= \frac{\binom{G}{g}\binom{B}{b}}{\binom{N}{n}} \end{aligned} \tag{3.6}$$

Equation (3.6) gives the total number of possible unordered samples with g green and b blue balls out of all possible unordered samples of size n. Multiplying $\binom{G}{g}$ by $\binom{B}{b}$ will give the number of possible unordered samples with g green and b blue balls.

In a more general form: If N denotes the total population size, consisting of M marked elements and $N - M$ unmarked elements, the the probability of drawing k marked elements in a sample size of n will be [4]:

$$\begin{aligned} X &= \textit{number of marked elements when selecting n items from a population of size N} \\ X &\sim \text{Hypergeometric}(M, N, n) \\ P &(X = K) = \frac{\binom{M}{k}\binom{N-M}{n-k}}{\binom{N}{n}} \end{aligned} \tag{3.7}$$

Example: Suppose an urn consists of 50 balls. Out of the 50 balls, 20 are green and 30 are blue. If 10 balls were drawn randomly, what is the probability of having exactly 5 green balls?

$$\begin{aligned} P(X = 5) &= \frac{\binom{20}{5}\binom{50-20}{10-5}}{\binom{50}{10}} \\ &= 0.215 \end{aligned}$$

The probability of drawing 10 balls from 50 balls with exactly 5 green

balls is 0.215.

Chapter 4

Overrepresentation of Gene Ontology Terms

Suppose that the goal of an experiment is to find the genetic factors that differentiate yeast cells in a normal environment from yeast cells in a changed environment. In order to answer the question of the experiment, gene expressions from yeast cells in a normal environment can be compared to gene expressions from yeast cells in the changed environment using microarrays. Our population set will consist of the genes on the microarray chip (i.e., those measured in our experiment), which may contain all the genes in the yeast cell. To determine which genes will be contained in our study set, we need to find the genes that are differentially expressed between the control group (genes from normal cells) and the experimental group (genes from changed cells). A t-test is a test of means; for each gene, the average gene expression from the control group will be compared to the average gene expression from the experimental group to see if they are significantly different (i.e., whether the difference in averages are bigger than would be expected given the inherent variability in the samples). There are multiple replicates for both the control and experimental group. From the replicates, the average of the gene expression level for each gene in the control and experimental group can be calculated and compared. A t-test will be used for each gene to assess the differential expression and produce a

p-value. The study set will be defined as all the genes that were considered to be significant by the t-test (e.g., p-value ≤ 0.05).

Let our control group be all the genes in a yeast cell raised in a normal environment and our experimental group be all the genes in a yeast cell raised in a nutrient-deprived environment. Suppose that the population set contains all the genes in a yeast cell, G genes in total. The microarray chip will also consist of G genes. In order to compare the nutrient deprived condition with the normal condition, all the samples will be used to do one t-test per gene. We will get G p-values, and the n genes that are differentially expressed (i.e., have significant p-values) will become part of our study set.

After the study set is determined, it is important to see if there are any GO terms in the study set that are significantly overrepresented. Overrepresentation occurs when the number of genes that annotate to a specific GO term is higher than expected by chance when considering the number of genes annotated to the GO term in the population. For example, suppose 1% of all genes in the population annotate to the GO term *vitamin synthesis*, yet we find that 5% of the genes in our study set annotate to *vitamin synthesis*. If this increase in percentage for vitamin synthesis in our study set cannot be attributed to chance, then the GO term, *vitamin synthesis*, is significantly overrepresented. We use the hypergeometric distribution to model chance behavior of selecting genes from the population to be in the study set because we can model chance selection of genes into the study set using the same ideas of selecting balls from an urn without replacement. However, testing for overrepresentation of GO terms is not always straightforward.

4.1 Term-for-Term

One method of analyzing overrepresentation is the *term-for-term* approach. The *term-for-term* approach is the simplest method for calculating overrepresentation of a single GO term. When using this approach, each GO term is analyzed individually to see if it is significantly overrepresented by using

the hypergeometric distribution. The hypergeometric distribution is used to compare the number of genes that annotate to a specific GO term in the study set versus the number of genes we would expect to annotate to that specific GO term in a randomly drawn subset of the population, which is the same size as the study set[3].

Let P be the population of size m that consists of all genes on the microarray that each annotate to at least one GO term, S be the study set of size n of genes that are differentially expressed, as defined previously, and t be the GO term of interest. Further notation gives P_t as the set of genes of size m_t in the population that annotate to the GO term of interest, t, and S_t as the set of genes of size n_t in the study set that annotate to t. Let Σ be the subset of size n that is drawn randomly from the population. Σ_t are the genes in Σ that annotate to t in the random subset and let the number of genes in Σ_t be denoted by σ_t. The probability of getting exactly k annotations in the random subset from the population can be calculated by the hypergeometric distribution defined in Chapter 3 [3]:

$$P(\sigma_t = k) = \frac{\binom{m_t}{k}\binom{m-m_t}{n-k}}{\binom{m}{n}} \tag{4.1}$$

Equation (4.1) gives the likelihood of obtaining k genes which annotate to t in a randomly chosen subset of size n from the population P. However, we are actually interested in getting n_t annotations or more becase we are looking for our data, n_t, or more extreme, the p-value. The probability of getting n_t or more genes that annotate to t if we sample n genes randomly from our population can be calculated by the cumulative hypergeometric distribution [3]:

$$P(\sigma_t \geq n_t) = \sum_{k=n_t}^{min(m_t,n)} \frac{\binom{m_t}{k}\binom{m-m_t}{n-k}}{\binom{m}{n}} \tag{4.2}$$

In Equation (4.2), we are calculating the probability of drawing n_t genes

or more that annotate to t by finding the number of ways of choosing k genes that annotate to t from the total number of genes in the population that annotate to t (i.e., m_t) and the number of ways of drawing the remaining genes in the study set ($n - k$) from the genes that don't annotate to t in the population ($m - m_t$). Then, the relevant probability is calculated by dividing by the number of ways of choosing the number of genes in the study set (n) from the number of genes in the population (m). We only sum up k ranging from n_t to the minimum of m_t, the number of genes in the population that annotate to t, and n, the number of genes in the study set because it is not certain which one is smaller. Both m_t and n are bounds on σ_t, the number of genes that annotate to t in the random subset from the population. σ_t will never be greater than m_t because m_t is all of the genes in the population that annotate to t, and the random subset is being drawn from all of the genes in the population. σ_t will never be greater than n because n is also the size of the random subset.

Since the *term-for-term* approach analyzes each GO term of interest individually, it does not take into account inherent relationships between the GO terms. The *true-path-rule* states that when a gene annotates to a certain GO term, it also annotates to the parents of that specific term[3]. For example, *physiological response to wounding* is the child of *response to wounding*. This also means that *response to wounding* is the parent term of *physiological response to wounding*. By the *true-path-rule*, all the genes that annotate to *physiological response to wounding* also classify under the GO term *response to wounding*. Because the *term-for-term* approach does not take into account parent and child GO terms, this causes an *inheritance problem*, whereby the descendant terms of a specific GO term have a higher chance of being significantly overrepresented if the parents of that term are overrepresented[3]. If the children terms are significantly overrepresented, then there is also a higher chance that the parents are significantly overrepresented. This is problematic because terms that are not overrepresented show up inaccurately as such.

4.2 Parent-Child

4.2.1 One Unique Parent

A method that measures for overrepresentation of gene terms is the *parent-child* method. The *parent-child* method takes into account the parents of the GO terms, which addresses the *inheritance problem*. This is done by altering the *term-for-term* approach slightly. First consider the situation where the GO term of interest, *t*, has only one parent, *pa(t)*. Instead of randomly drawing a general subset of size *n* from the population like the *term-for-term* approach, the *parent-child* method has the same number of genes that annotate to the parent of *t* in the random subset as in the study set ($\sigma_{pa(t)} = n_{pa(t)}$). The random subset is created by drawing a subset of size $n_{pa(t)}$ from the genes that annotate to the parent of *t* in the population, and the genes that don't annotate to the parent of *t* are disregarded. So unlike the *term-for-term* method, the number of genes that annotate to *pa(t)* are the same in both the study set and the subset of the population for the *parent-child* method. Since conditioning on the same number of genes that annotate to the parent of *t* in the study set and the random subset accounts for the *inheritance problem* by looking for overrepresentation within the constraints of related terms, there will be less of an influence of the parent term on the terms being analyzed for significance, thus lowering the problem of the *inheritance effect*[6]. The probability of drawing exactly n_t genes that annotate to *t*, conditioned on the fact that the number of genes that annotate to *pa(t)* are the same in both the subset and study set can be calculated by the hypergeometric distribution[3]:

$$P(\sigma_t = n_t | \sigma_{pa(t)} = n_{pa(t)}) = \frac{\binom{m_t}{n_t}\binom{m_{pa(t)}-m_t}{n_{pa(t)}-n_t}}{\binom{m_{pa(t)}}{n_{pa(t)}}} \tag{4.3}$$

Equation (4.3) gives the probability of drawing the same number of genes in our random subset that annotate to *t* (σ_t) as in our study set (n_t) conditioned on the fact that the number of genes that annotate to the parent of *t* are the same in both the subset and the study set ($\sigma_{pa(t)} = n_{pa(t)}$).

This can be achieved by finding the number of ways of selecting the n_t genes from m_t, the number of genes in the population set that annotate to *t*, and selecting the number of remaining genes in our study set ($n_{pa(t)} - n_t$) from the remaining number of genes in the population set ($m_{pa(t)} - m_t$). The relevant probability is divided by selecting the number of genes in the study set that annotate to *pa(t)* (i.e., $n_{pa(t)}$) from the number of genes in the population set that annotate to *pa(t)* (i.e., $m_{pa(t)}$).

Once again, we are interested in finding the probability of our data or more extreme (the p-value) in order to determine whether or not the over-represesetation of *t* is due to chance. This means that we want to find the probability of randomly drawing n_t or more genes that annotate to the GO term of interest, conditioned on the fact that the number of genes that annotate to *pa(t)*, the parent of *t*, are the same in both the random subset and the study set. The p-value will be given by:

$$P(\sigma_t \geq n_t | \sigma_{pa(t)} = n_{pa(t)}) = \sum_{k=n_t}^{min(n_{pa(t)}, m_t)} \frac{\binom{m_t}{k}\binom{m_{pa(t)} - m_t}{n_{pa(t)} - k}}{\binom{m_{pa(t)}}{n_{pa(t)}}} \tag{4.4}$$

In Equation (4.4), we are finding the probability of selecting n_t or more genes that annotate to *t*. The number of ways of choosing *k* genes from the number of genes that annotate to *t* in the population (m_t) and the number of ways of selecting the remaining genes that annotate the parent of *t* in the study set after drawing *k* genes ($n_{pa(t)} - k$) from the remaining genes that annotate to the parent of *t* in the population ($m_{pa(t)} - m_t$) is divided by the number of ways of selecting the number of genes in the study set that annotate to the parent of *t* ($n_{pa(t)}$) from the number of genes in the population that annotate to the parent of *t* ($m_{pa(t)}$). The sum only goes from n_t to the minimum of $n_{pa(t)}$ and m_t because like Equation (4.2), we do not know which one will be greater. In general, the number of genes that annotate to the parent of *t* will be greater, but if we have a smaller study set, then the number of genes that annotate to the parent in the study set will be smaller than the number of genes that annotate to *t* in the populaton (m_t). The number of genes that annotate to *t* in the random subset (σ_t) is

bounded by the number of genes that annotate to t in the population (m_t) because it can never be greater since Σ, the random subset, is drawn from the population. σ_t will also never be greater than $n_{pa(t)}$, because we are drawing $n_{pa(t)}$ genes in the random subset.

4.2.2 Multiple Parents

Previously, in Equation (4.4), we assumed that the GO term of interest has only one unique parent; however, typically each GO term has more than one parent[6]. For example, *physiological defense response* and *physiological response to wounding* are both parents of *inflammatory response*. Consider a term of interest, *t*, and let one of the parents be t' and *pa(t)* be all the parents of t. In order to take into account all the parents of t, the genes belonging to the set of parents of the term of interest in our study set is redefined as genes that can annotate to any of the parents of t. This is called the *parent-child-union*[6]. The number of genes that annotate to *pa(t)* now becomes:

$$n_{pa(t)} = |\bigcup_{t' \in pa(t)} S_{t'}| \tag{4.5}$$

In Equation (4.5), $S_{t'}$ is the set of genes that annotate to a particular parent of t and the parents of t is defined as the set of genes in the union of all $S_{t'}$. $n_{pa(t)}$ is redefined as the number of genes in the study set that annotate to any of the parents of t. Like $n_{pa(t)}$, $m_{pa(t)}$ is now the number of genes in the population set that annotate to any of the parents of t, and $\sigma_{pa(t)}$ is the number of genes in our random subset of the population that annotate to any of the parents of t.

The *parent-child-intersection* defines the set of parents of the term t as the intersection of the sets of genes that annotate to a particular parent of t[3]. Mathematically, this becomes:

$$n_{pa(t)} = |\bigcap_{t' \in pa(t)} S_{t'}| \tag{4.6}$$

In Equation (4.6), $S_{t'}$ is still the set of genes that annotate to a particular parent of *t*; however, unlike Equation (4.5), the parents of *t* are now defined as the intersection of all $S_{t'}$. $n_{pa(t)}$ is redefined as the number of genes in the study set that annotate to all the parents of *t*. Likewise, $m_{pa(t)}$ is now the number of genes in the population set that annotate to all of the parents of *t* and $\sigma_{pa(t)}$ is the number of genes in our random subset of the population that annotate to all of the parents of *t*.

To help understand the *parent-child-union* and *parent-child-intersection* method, consider the following example. Let P_1, P_2, and P_3 be gene ontology terms and the parents of the GO term of interest, t. Suppose gene 1 annotates to P_1, P_2, and P_3, and gene 2 only annotates to P_1 and P_2. Both gene 1 and gene 2 would be included in the *parent-child-union* method since this method includes all the genes annotated to any of P_1, P_2, and P_3. However, only gene 1 would be included in the parent-child-intersection method since this method only includes genes that annotate to all P_1, P_2, and P_3.

The probability of getting n_t or more genes for both the *parent-child-union* and *parent-child-intersection* method is the same equation as the *parent-child* method with one unique parent:

$$P(\sigma_t \geq n_t | \sigma_{pa(t)} = n_{pa(t)}) = \sum_{k=n_t}^{min(n_{pa(t)}, m_t)} \frac{\binom{m_t}{k}\binom{m_{pa(t)}-m_t}{n_{pa(t)}-k}}{\binom{m_{pa(t)}}{n_{pa(t)}}} \tag{4.7}$$

However, the number of genes that annotate to the parents of *t* in the study set, population, and random subset ($n_{pa(t)}$, $m_{pa(t)}$,$\sigma_{pa(t)}$, respectively) are now defined according to the *parent-child-union* or the *parent-child-intersection* in Equations (4.5) and (4.6).

Example: Let the study set contain 100 genes where 3 of the genes annotate to the term of interest, *t*. That is, of the 100 genes which were significant in the experiment, 3 of them annotated to *t*. Suppose 35 of the 100 genes in the study set annotate to the parents of *t*. Let the population consist of 1000 genes where 400 of the genes annotate to the parents of *t* and 12 of the genes annotate to the term *t*. This means that out of all the genes in the

population, we would only find 1.2% that annotated to our term of interest; however we found 3% in our study set.

	study set	population
annotate to t:	$n_t = 3$	$m_t = 12$
annotate to $pa(t)$:	$n_{pa(t)} = 35$	$m_{pa(t)} = 400$
total:	$n = 100$	$m = 1000$

We can use both the *parent-child* and *term-for-term* approach to calculate the p-value and to see if the 3% was just due to chance. That is, we can find the probability of getting at least 3 genes that annotate to *t* in a subset from the population.

Term-for-term approach:

$$\begin{aligned} P(\sigma_t \geq 3) &= \sum_{k=3}^{min(n=100, m_t=12)} \frac{\binom{12}{3}\binom{988}{97}}{\binom{1000}{100}} \\ &= 0.9151 \end{aligned} \tag{4.8}$$

Parent-child-union approach:

$$\begin{aligned} P(\sigma_t \geq 3 | \sigma_{pa(t)} = n_{pa(t)} = 35) &= \sum_{k=3}^{min(n_{pa(t)}=400, m_t=12)} \frac{\binom{12}{3}\binom{388}{32}}{\binom{400}{35}} \\ &= 0.93667 \end{aligned} \tag{4.9}$$

Note: Both the results of the *term-for-term* and *parent-child* approaches are not significant. That is, an overrepresentation of 3% as compared to 1.2% is not significant.

Unlike Equation (4.8), Equation (4.9) is asking for the probability of getting at least 3 genes that annotate to *t* if we know that 35 annotate to the parents of *t* after removing the non-parents from the population. The *term-for-term* approach will tend to have a smaller probability than the *parent-child* approach since the denominator will tend to be greater in Equation (4.8) than Equation (4.9). Even though the numerator in Equation (4.8) is greater than the numerator in Equation (4.9), the denominator is more

dominant and will affect the outcome more. It makes sense that Equation (4.8) produces a smaller probability than Equation (4.9) because they are both calculating the probability of seeing our data or more extreme, or the p-value. If the p-value is small, there is a higher chance that the terms are significantly overrepresented; however, we only want the terms that are truly overrepresented to show up as such. Equation (4.9) should be chosen over Equation (4.8) because the *parent-child* approach conditions on the parents due to the *inheritance problem* and is more conservative. This means that the *parent-child* approach is measuring the impact of term *t* within the constraints of related terms.

In order to show that the *parent-child* approach is a more accurate method than *term-for-term* approach for determining overrepresentation, Grossmann et al. [2007] generated study sets with one specific GO term purposefully overrepresented and compared the number of false positives and false negatives using both the *term-for-term* approach and the *parent-child* approach.

4.3 False Positives

4.3.1 Simulating Data

Let the one specific GO term be known as $t_{over}(S)$. The population set consists of genes that can be annotated to $t_{over}(S)$, P_t, and genes that cannot be annotated to $t_{over}(S)$. To create artificial study sets where a GO term is overrepresented, a certain percentage, also called *term proportion* was randomly drawn from P_t, and a certain percentage, also called *population proportion* was drawn from genes in the population that cannot be annotated to $t_{over}(S)$[6]. P-values can then be calculated for each term in the study set by both the *term-for-term* approach and *parent-child* approach to see if the terms were significantly overrepresented as was shown previously using the hypergeometric distribution.

For example: Let the specific GO term of interest be *immune response*, the *term proportion* be 0.75, and the *population proportion* be 0.10. This means 75% of all genes that can be be annotated to *immune response* from the pop-

ulation are randomly drawn. The subset may contain genes that can be directly annotated to *immune response* or annotated to a descendant of *immune response* such as *regulation of immune response*. Next, 10% of the population of genes that could not be annotated to *immune response* are randomly drawn. As a result, the study set contains genes that either can or cannot be annotated to *immune response* with each group represented by set proportions from their larger populations. A p-value, which calculates the probability of getting our data or more extreme, for every GO term can then be obtained by both the *term-for-term* and *parent-child* method to see if the term *immune response* and its children are overrepresented.

Grossmann et al. [2007] generated 1115 study sets using genes from *S. cerevisiae* where one GO term was purposefully overrepresented. The *term-for-term* and *parent-child* methods were compared using different combinations of *term proportions* of 0.75, 0.50, and 0.25 and *population proportions* of 0.10 and 0.20. In all the different combinations of *term* and *population proportions* the *parent-child* method was superior to the *term-for-term* method in that the *term-for-term* method highlighted more terms that were actually not overrepresented as being overrepresented. The comparison between the two methods can be shown by using receiver operating characteristic (ROC) curves which is explained later in Chapter 4.3.2.

To further highlight the problems of the *term-for-term* approach, Grossmann et al. [2007] once again used genes from *S. cerevisiae* and purposefully overrepresented the GO term *DNA repair* (GO:0006281). The study set was created by using a *term proportion* of 0.5 and a *population proportion* of 0.1. The *term-for-term* approach correctly showed that *DNA repair* was significantly overrepresented; however, it also showed that three of the children of *DNA repair* were significantly overrepresented since the *term-for-term* approach does not take into account dependencies between the parent and children terms. This is problematic because although the children terms were not suppose to be overrepresented, they still showed up as such. This may lead researchers that are examing genes from *S. cerevisiae* to only examine *DNA repair* and the three children terms and to disregard the other children terms that did not show up as significantly overrepresented; how-

ever the other children terms of *DNA repair* may be just as important[3].

4.3.2 Computing False Positive Rates and False Negative Rates

A false positive occurs when a gene term that is not purposefully overrepresented shows up as such, and a false negative is when a gene term that is purposefully overrepresented does not show as being overrepresented. The artificial study sets created by Grossmann et al. [2007] can be used to calculate the false positives and false negatives for each study set. ROC curves, which will be explained at the end of this section, can be used to compare the number of false positives between the *parent-child* approach and the *term-for-term* approach.

Only the terms that can be annotated to the specific GO term of interest, $t_{over}(S)$, are used in the calculation of false positive and false negative[6]. For example, the child term of *immune response*, *regulation of immune reponse*, is in the same subontology group and can also annotate to *immune response*. The terms that are used in the calculation of the false positive and false negative rates are labeled as $T_{test}(S)$. In the example mentioned above conducted by Grossmann et al. [2007], all children terms of *DNA repair* are in $T_{test}(S)$. Any term that shows up as being significantly overrepresented that is not intentionally overrepresented counts as a false positive. In the example where only *DNA repair* is purposefully overrepresented , any term other than *DNA repair*, such as the three children terms, counts as a false positive if it shows up as being significantly overrepresented. A false negative is defined as a term that is intentionally overrepresented and does not show up as such. *DNA repair* would be considered a false negative if it does not show as being significantly overrepresented. In more simplified terms:

$$\#False\ positives = \#\ of\ terms\ \epsilon\ T_{test}(S)\ which\ are\ overrepresented\ but\ are\ not\ t \tag{4.10}$$

$$\#False\ negatives = \{0, 1\} \tag{4.11}$$

Grossmann et al. [2007] defines false positive rates more specifically

to emphasize the effect of the *inheritance problem*. The subgroup, $T_{test}(S)$, is defined more conservatively to highlight the *inheritance problem*, in that more terms that are not significantly overrepresented will have a greater chance of showing up as such, when using the *term-for-term* approach than the *parent-child* approach. $T_{test}(S)$ now only consists of the strict descendants of $t_{over}(S)$. This new subgroup can be called $T_{desc}(S)$. The false positives in $T_{desc}(S)$ can then be counted at a p-value cutoff, π.

$$FPR_{desc}(\pi) = \frac{\sum_{S \epsilon \mathbf{S}} |t \epsilon T_{desc}(S) : p_t(S) < \pi|}{\sum S \epsilon \mathbf{S} |T_{desc}(S)|} \tag{4.12}$$

Above, we sum up all the terms that are significantly overrepresented with a p-value less than π in $T_{desc}(S)$ over all independent study sets. As in Equation (4.10), terms are considered to be false positives when they are not purposefully overrepresented. Note that the term that is purposefully overrepresented is not included in $T_{desc}(S)$.

Grossmann et al. used ROC curves to compare the *parent-child* method and the *term-for-term* method for each artifical study set that consisted of different *term proportions* and *population proportions* that was mentioned in Chapter 4.3.1. ROC curves are graphed with the false positive rate versus the true positive rate by using π cutoffs which range between 0 and 1. When π is 0, there are no false positives or false negatives since the probability of significance is 0. When π is at 1, all terms are significant, and the false positive rate is 1 since all p-values less than 1 are considered to be significant. The best method is the one that gives a higher true positive rate for any false positive rate. It can be seen from the ROC curves that the *term-for-term* approach has a much higher false positive rate and falsely highlights more descendants as being overrepresented than the *parent-child* method[6]. This is expected since the *parent-child* approach accounts for dependencies between the parent and child GO terms while the *term-for-term* approach assumes that the GO terms are independent from one another.

Chapter 5

Multiple Testing Issues

In the first part of this work, we have demonstrated how to deal with dependence issues involving parent GO terms. Independence across GO terms does not hold because the parent and child terms are dependent on one another due to the *inheritance problem* that was discussed previously. If the parent term is overrepresented, then the probability that its children terms are also overrepresented increases, and vice versa.

Additionally, we must consider multiple testing issues that as the number of tests increases, the number of false significances or false positives, will also increase[7]. For example, when $\alpha = 0.05$, this means that 5% of the null terms will show up as being differentially expressed just by chance, when in reality they are not. If we ran 20 tests, none of which were significant, we would expect 1 false positive due to chance. However, if we decide to run 200 null tests, then we would expect to see 10 false positives due to chance; therefore, as the number of tests increases, the number of false positives will also increase.

The combined false positive rates or Type I error rates from many different tests is the family wise error rate (FWER), which is used to adjust for multiple testing. The FWER is the probability that at least one of the null hypotheses will be rejected when all the null hypotheses are actually true

or the probability of making at least one Type I error[8].

$$\begin{aligned} FWER &= P(of\ making\ at\ least\ 1\ Type\ I\ error) \\ &= 1 - P(of\ making\ no\ Type\ I\ errors) \\ &= 1 - [P(i^{th}\ test\ is\ not\ a\ Type\ I\ error)]^N \\ &= 1 - (1-\alpha)^N \end{aligned} \tag{5.1}$$

In Equation (5.1), N is the number of null tests. Note that the above equation only holds true if the tests are independent; however, this is not a valid assumption in testing for overrepresentation of GO terms, since the second term may be the parent or child of the first term.

Two ways to fix multiple testing issues are the Bonferroni method and the Westfall-Young correction method.

5.1 Bonferroni Method

In the Bonferroni correction method, the p-value of each individual hypothesis test is adjusted in order to lower the FWER when all the tests are put together[8]. The adjusted p-value for just two null tests $N = 2$ is derived as the following [9]:

$$\begin{aligned} A_i &= rejecting\ i^{th}\ null\ (making\ a\ Type\ I\ error) \\ FWER &= P(making\ at\ least\ 1\ Type\ I\ error) \\ &= P(A_1 \bigcup A_2) \\ &= P(A_1) + P(A_2) - P(A_1 \bigcap A_2) \\ &\leq P(A_1) + P(A_2) \\ &\leq 2\alpha \end{aligned} \tag{5.2}$$

Which extends to *N* tests as:

$$\begin{aligned} FWER &\leq N\alpha \\ corrected\ \alpha\ for\ an\ individual\ test &= \alpha/N \end{aligned} \tag{5.3}$$

If we want the FWER to be less than $N\alpha$, we need to adjust the significance level for each individual test to $\frac{\alpha}{N}$. This means that if we are conducting 200 different tests with $\alpha = 0.05$, only individual tests that have a p-value less than 0.00025 will be considered significant. As N increases, the Bonferroni correction becomes more stringent, and we will be unable to reject any true relationships[10].

5.2 Westfall-Young Method

In the Westfall-Young correction method, one study set with *k* terms with *k* p-values (one for each term) is randomly resampled to remove the structures between genes. *k* "pseudo" p-values are calculated for each term by the *term-for-term* method or the *parent-child* method after randomly resampling. So if there were 200 resamplings, then each term will have 200 "pseudo" p-values. The adjusted p-value is calculated by seeing how many of the original p-values are smaller than the minimum of the "pseudo" p-values divided by the number of resamplings[8].

$$\begin{aligned} min(k \text{ "pseudo" } p-values) &= min(p)_i \; where \; i \; is \; the \; i^{th} \; resample \\ adjusted \; p-value &= \frac{\#of \; times \; the \; true \; p-value \; \leq min(p)_i}{\#of \; resamplings} \end{aligned} \quad (5.4)$$

In Equation (5.4), the minimum of the *k* p-values is used as the cutoff to not make a FWER. So if one p-value is greater than the minimum of the *k* p-values, then we will have at least one Type I error.

5.3 Comparison of the Two Methods

Grossmann et al. [2007] used both the Bonferroni correction and the Westfall-Young correction method to compare the *term-for-term* and *parent-child* approaches. 2000 randomly generated study sets of size 250 were created to remove the structure between genes and resampled 5000 times. The FWER plots for the *term-for-term*, *parent-child-union*, and *parent-child-intersection*

approaches showed that Westfall-Young method in combination with either the *parent-child-union* or *parent-child-intersection* approach was the best method since the Bonferroni correction tends to be too conservative[3].

Chapter 6

Conclusion

Analyses of GO terms with the hypergeometric distribution can easily be flawed if the *inheritance problem* is not accounted for. The *inheritance problem* describes how, if the parent GO term is overrepresented, there is a higher chance that its children terms will also show up as being overrepresented. This can also be examined conversely, meaning that if the children are overrepresented then the parents will also have a higher chance of being overrepresented.

Grossmann et al. [2007] showed that the *parent-child* method is superior to the *term-for-term* approach since the *parent-child* method takes dependencies between the parent and child terms into account when calculating for overrepresentation. From the ROC graphs, we can see that the *parent-child* method produces fewer false positives than the *term-for-term* approach in all varying simulated *term proportions* and *population proportions*. In the simulated studies, the *parent-child* method showed fewer children as being significantly overrepresented when compared to the *term-for-term* approach since the *parent-child* method takes into account the *inheritance problem*.

In actual studies, where the *inheritance problem* causes the parents to be overrepresented because the children terms are overrepresented, the reasearchers tend to examine the children terms, which will also allow them to examine the parent terms. However, if we consider the other scenario where some of the children terms show up by chance as being over-

represented because the parent term is overrepresented (also due to the *inheritance issue*), then the researcher may only examine the few children terms that showed up as overrepresented and ignore all other child terms. This direction is more problematic since the researcher may be missing crucial information by neglecting the children terms that were not shown as overrepresented, though they may be just as important as the other children terms.

The simulated experiments by Grossmann et al. [2007] showed that the *parent-child* method is a better method than the *term-for-term* approach since it highlights fewer children as significantly overrepresented of the purposefully overrepresented GO term. Although the children terms were not shown as being overrepresented, this does not mean that they are not important. All the *parent-child* approach is saying is that there is not enough information to show that the children terms are significant which is a superior outcome than producing misleading information that can cause incorrect biological interpretation resulting from the use of the *term-for-term* approach[3].

Bibliography

[1] S. Dudoit, Y. Yang, M. Callow, T. Speed (2000). *Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments* (No. 578). Berkeley: Dept. Statistics, University of California, Berkeley.

[2] The Gene Ontology Consortium. (May 2000). Gene Ontology: tool for the unification of biology. *Nature Genetics, 25*.

[3] Grossmann, S., Bauer, S., Robinson, P., Vingron, M. September 11, 2007. *Improved detection of overrepresentation of Gene-Ontology annotations with parent-child analysis.Bioinformatics, Vol. 23 no. 22 2007, pages 3024-3031*. doi:10.1093/bioinformatics/btm440.

[4] Pitman Jim, 2006. *P*robability. United States of America: Springer Science+Business Media, LLC.

[5] D. Sadava, H. Heller, G. Orians, P. William, D. Hillis. *Life: The Science of Biology*. W.H. Freeman and Company, New York, 2006.

[6] Grossmann, S., Bauer, S., Robinson, P., Vingron, M. Proceedings from RECOMB 2006: *Improved detection of overrepresentation of Gene-Ontology annotations with parent-child analysis* Venice-Lido, Italy.

[7] Westfall P.H., Young S.S. (1993). *Resampling-based Multiple Testing*. Wiley, New York.

[8] National Center for Education Evaluation and Regional Assistance (2008). *Technical Methods Report: Guidelines for Multiple Testing in*

Impact Evaluations. Retrieved from `http://ies.ed.gov/ncee/pubs/20084018/app_b.asp`

[9] Kutner M. H., Nachtsheim C. J., Neter J., Li W. (2005). *Applied Linear Statistical Models: Fifth Edition. New York, NY: McGraw-Hill/Irwin.*

[10] Silicon Genetics (2003). Multiple Testing Corrections*. Retrieved from `http://www.silicongenetics.com/Support/GeneSpring/GSnotes/analysis_guides/mtc.pdf`*