

CONTROLLING FALSE DISCOVERY WHEN PLANNING MICROARRAY EXPERIMENTS.

Eric R. Siegel, Trey Spencer, and Rudolph S. Parrish.

Department of Biostatistics, University of Arkansas for Medical Sciences, Little Rock, AR 72205-7199

KEY WORDS: Microarray, multiple comparison, false discovery, planning, Sidak, Šidák.

Abstract:

Sophisticated resampling methods exist to control False Discovery (FD) in microarray experiments once the data are in hand. Less attention has been paid to control of FD during the design stage. In this context, Simon proposes controlling expected FD (EFD) for tests on N genes as follows: simply choose the level p of test such that $Np \leq u$, where u is the desired EFD. We propose a method almost as simple as Simon's for controlling the actual FD. Under independence and the complete null hypothesis of no differential expression for any gene, FD is binomially distributed, and one can calculate the exact probability $P(\text{FD} \leq u | \text{Bin}(N, p))$ of having u or fewer false discoveries with a given level p for testing. One can then adjust the level p needed to raise this probability to a desired "1- α " confidence level, thus controlling FD. For $u=0$, our method reduces to the Šidák multiple-comparison adjustment procedure for controlling familywise error. For $u > 0$, the needed level p can be approximated to high precision using Newton's method in an Excel spreadsheet; convergence typically takes about six iterations.

Introduction:

Traditional multiple comparison methods may be too stringent to apply to gene-expression microarray data, in which the number of comparisons can number in the tens of thousands. In response, many have abandoned the use of familywise error in favor of the broader concept of false discovery, and seek to control the rate (FDR) or number (FD) of tests that result in Type I error. When the data are in hand, sophisticated resampling methods exist to do this. Two examples are:

- SAM, a permutation-based testing algorithm developed by Tusher *et al.* (2001), which implements the FDR procedure of Benjamini and Hochberg (1995).
- An extension of the step-down resampling procedure of Westfall and Young (1993), developed by Korn *et al.* (2001). The Westfall and Young procedure was developed to control familywise error (i.e., hold FD to zero) when variables are correlated; Korn *et al.*'s extension generalizes that procedure to analyses where desired FD is non-zero.

Resampling methods are computationally intensive, but their performance does not depend on

distributional assumptions such as normality. However, resampling methods require data, usually from a completed experiment.

Little attention has been paid to controlling FD during the design of future experiments. A common practice during the design stage is to set the level p of test at a moderately low level such as 0.001, then multiply this value by the number N of genes to obtain an estimate of Expected FD (EFD). Simon (2002) takes this practice one step further, and proposes to control EFD as follows: Decide on a desired EFD, call it u , then calculate the needed level of test as $p = u/N$. Two questions naturally flow from Simon's proposal:

- When $\text{EFD} = u$ given N and p , what is the probability that the actual FD will be u or fewer? In other words, what is $P(\text{FD} \leq u | N, p)$?
- If one can calculate $P(\text{FD} \leq u | N, p)$, can one alter p to change it? Specifically, can p be adjusted to control FD by raising $P(\text{FD} \leq u | N, p)$ to a desired "1- α " confidence level?

The answers to these questions are Yes.

Theory:

Under the complete null hypothesis, no gene shows true differential expression with treatment. Therefore, every discovery (i.e., significant test result) will be a false discovery. We assume that, under the complete null hypothesis, the N tests of gene expression have independent outcomes. Then FD will be binomially distributed,

$$\text{FD} \sim \text{Bin}(N, p)$$

with N being the number of tests for differential expression, and p being the level of test. We therefore have $\text{EFD} = Np$, in accordance with common practice and Simon's proposal. We also have the binomial Cumulative Distribution Function (CDF):

$$P(\text{FD} \leq u | N, p) = \sum_{i=0}^u \binom{N}{i} (p)^i (1-p)^{N-i}$$

Given N tests, a desired FD no greater than u , and a desired 1- α confidence level that $\text{FD} \leq u$, the problem, then, is to find a level of test \tilde{p} such that

$$1 - \alpha \leq \sum_{i=0}^u \binom{N}{i} (\tilde{p})^i (1 - \tilde{p})^{N-i} \quad (1)$$

When $u=0$, a closed-form solution for \tilde{p} exists:

$$1 - \alpha \leq \binom{N}{0} (\tilde{p})^0 (1 - \tilde{p})^N$$

$$\downarrow$$

$$1 - \alpha \leq (1 - \tilde{p})^N$$

which, on rearrangement, yields

$$\tilde{p} \leq 1 - (1 - \alpha)^{1/N},$$

the Šidák adjusted \tilde{p} for control of familywise error in multiple comparisons. When $u > 0$, closed-form solutions for \tilde{p} generally do not exist; however, \tilde{p} can be found through a simple iterative algorithm described below.

Our procedure for control of FD can be viewed as a generalization of the Šidák multiple-comparison adjustment procedure to analyses in which the desired maximum FD is greater than zero. The Šidák procedure is known to be conservative; it owes its conservative property, in fact, to the assumption that the N tests have independent outcomes. Because we make the same assumption, we conjecture that the conservative property of the Šidák procedure is inherited by our generalization of it.

Solution for \tilde{p} via Newton's method:

Letting $F(\tilde{p})$ represent the right-hand side of (1), form the function $F(\tilde{p}) - (1 - \alpha)$. The zeroes of this function may be found using Newton's method (Arfken 1985). Let $\tilde{p}_{(j)}$ represent the value of \tilde{p} at the j th iteration. Then Newton's method provides for successive approximations as follows:

$$\tilde{p}_{(j+1)} = \tilde{p}_{(j)} - \frac{F(\tilde{p}_{(j)}) - (1 - \alpha)}{(dF/dp)|_{\tilde{p}_{(j)}}}$$

where

$$\left. \frac{dF}{dp} \right|_{\tilde{p}_{(j)}} = -N \binom{N-1}{u} (\tilde{p}_{(j)})^u (1 - \tilde{p}_{(j)})^{N-1-u} \quad (2)$$

Equation (2), of course, is the first derivative of $F - (1 - \alpha)$ evaluated at $\tilde{p}_{(j)}$. Convergence is attained when $F - (1 - \alpha) \leq \epsilon$, say 10^{-9} . Because $1 - F$ is a Beta CDF with respect to \tilde{p} (see below), there is only one zero to find, i.e. $F - (1 - \alpha) = 0$ has only one solution. Although dF/dp never attains zero in the open interval $(0, 1)$, it can come very close given the high N found in microarray experiments, yielding iterations that do not converge. To avoid this, it is important to choose an initial estimate $\tilde{p}_{(1)}$ that's not too distant from the solution. We find that Simon's $p = u/N$ for EFD control works well as an initial estimate.

Implementation in Excel Spreadsheet:

Figure 1 shows an example of Newton's method implemented in an Excel spreadsheet. The microarrays have 21,035 genes, and the goal is to find the level \tilde{p} of test for each gene needed to hold FD to 21 or fewer with 95% confidence. In the row labeled "Initial", 1-alpha is the confidence level, N is the number of genes, maxFD is the desired maximum FD, p_(j) is the ratio maxFD/N (Simon's $p = u/N$), and F(maxFD|N,p) is the value of the binomial CDF calculated by the Excel "Binomdist" function from maxFD, N, and p_(j). F-(1-alpha) is $F - (1 - \alpha)$, F-prime is dF/dp (calculated using the the Excel "Binomdist" function), and p_(j+1) is $\tilde{p}_{(j+1)}$, the Newton's-method result to be fed into the next iteration. In the row labeled "iter. 02", all the cells are the same as before, except for p_(j), which is set equal to p_(j+1) from the previous row. To create "iter. 03" and subsequent iterations, the formulas in the cells of the "iter. 02" row are copied to cells in subsequent rows. The curved arrows in Figure 1 show how p_(j+1) in each row becomes p_(j) in the

Figure 1: Implementation of Newton's Method in an Excel Spreadsheet

	1-alpha	N	maxFD	p_(j)	F(maxFD N,p)	F-(1-alpha)	F-prime	p_(j+1)
Initial	0.95	21035	21	0.0009983	0.55768596268	-0.392314	-1824.89	0.0007833564913
iter. 02	0.95	21035	21	0.0007834	0.88896984142	-0.0610302	-1030.84	0.0007241519254
iter. 03	0.95	21035	21	0.0007242	0.93969030416	-0.0103097	687.33	0.0007091522074
iter. 04	0.95	21035	21	0.0007092	0.94939321001	-0.0006068	-607.09	0.0007081526967
iter. 05	0.95	21035	21	0.0007082	0.94999739989	-2.6E-06	-601.89	0.0007081483768
iter. 06	0.95	21035	21	0.0007081	0.94999999995	-4.77E-11	-601.87	0.0007081483767
iter. 07	0.95	21035	21	0.0007081	0.95000000000	2.365E-13	-601.87	0.0007081483767
iter. 08	0.95	21035	21	0.0007081	0.95000000000	-1.018E-12	-601.87	0.0007081483767
iter. 09	0.95	21035	21	0.0007081	0.95000000000	1.699E-12	-601.87	0.0007081483767
iter. 10	0.95	21035	21	0.0007081	0.95000000000	-1.285E-12	-601.87	0.0007081483767

following row. In “iter. 07” and subsequent rows, it can be seen that $p_{(j)}$ and $p_{(j+1)}$ have converged to the solution $\tilde{p} = 0.0007081483767$, with fluctuations of $\tilde{p}_{(j)}$ about \tilde{p} of order 10^{-14} , and fluctuations of $F-(1-\alpha)$ about 0 of order 10^{-12} .

Alternative Implementation in Excel:

In Excel 2000, the “Binomdist” function will return the result “#NUM!” when the binomial N -choose- k coefficient is larger than about 10^{307} . For $N=21,035$ genes, this happens when $\max FD \geq 115$. However, one can still implement Newton’s method in Excel for large desired FD using the following identity between the binomial CDF and Beta function:

$$\sum_{i=0}^k \binom{N}{i} (\tilde{p})^i (1-\tilde{p})^{N-i} = \frac{\Gamma(N+1)}{\Gamma(k+1)\Gamma(N-k)} \int_{\tilde{p}}^1 z^k (1-z)^{N-k-1} dz$$

Note that, since the limits of integration are from \tilde{p} to 1, the Beta function in the above identity is equal to 1 minus the CDF for the Beta distribution with parameters $(k+1, N-k)$, and the Beta function’s first derivative at \tilde{p} is therefore the negative of the corresponding Beta PDF. The Beta CDF can be computed directly in Excel 2000 by using the Excel

“Betadist” function. The Beta PDF can be computed indirectly in Excel 2000, using the Excel “Gammaln” function to compute the natural logarithms of the Gamma terms in the Beta-function coefficient.

References:

- Arfken G, 1985. *pp* 963-964 in: *Mathematical Methods for Physicists*, Third Edition. Academic Press, Inc., Orlando, Florida, USA.
- Benjamini Y and Hochberg Y, 1995. Controlling the False Discovery Rate: A practical and powerful approach to multiple testing. *J Roy Stat Soc* 57:289-300.
- Korn EL, Troendle JF, McShane LM, and Simon R, 2001. Controlling the number of false discoveries: Application to high-dimensional genomic data. Technical Report 003, National Cancer Institute, <http://linus.nci.nih.gov/~brb/TechReport.htm>.
- Simon R, 2002. Design issues in DNA microarray based studies. Talk 008, National Cancer Institute, <http://linus.nci.nih.gov/~brb/TechReport.htm>.
- Tusher VG, Tibshirani R, and Chu G, 2001. Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci* 98:5116-5121.
- Westfall, PH, and Young SY, 1993. *Resampling-Based Multiple Testing: Examples and Methods for P-value adjustment*. John Wiley & Sons, Inc., New York. 340 pp.