Deposited research article

A new estimate of the proportion unchanged genes in a microarray experiment

Per Broberg

Address: Biological Sciences, AstraZeneca R&D Lund, S-221 87 Lund, Sweden. E-mail: per.broberg@astrazeneca.com

Posted: 1 April 2004

Genome ***Biology*** 2004, **5:**P10

The electronic version of this article is the complete one and can be found online at http://genomebiology.com/2004/5/5/P10

Received: 30 March 2004

This is the first version of this article to be made available publicly.

A new estimate of the proportion unchanged genes in a microarray experiment

Per Broberg
Biological Sciences, AstraZeneca R&D Lund, S-221 87 Lund, Sweden
Correspondence: per.broberg@astrazeneca.com
Telephone: + 46 46 33 78 22
Fax: +46 46 33 71 64

Running heading : A new estimate of the proportion unchanged genes

Abstract

Background

In the analysis of microarray data one generally produces a vector of *p*-values that for each gene give the likelihood of obtaining equally strong evidence of change by pure chance. The distribution of these *p*-values is a mixture of two components corresponding to the changed genes and the unchanged ones. The basic question 'What proportion of genes is changed' is a non-trivial one, with implications for the way that such experiments are analysed. An estimate not requiring any assumptions on the distributions is proposed and evaluated. The approach relies on the concept of a moment generating function.

Results

A simulation model of real microarray data was used to assess the proposed method. The method fared very well, and gave evidence of low bias and very low variance.

Conclusions

The approach opens up a new possibility of sharpening the inference concerning microarray experiments, including more stable estimates of the false discovery rate.

Background

The microarray technology permits the simultaneous measurement of the transcription of thousands of genes. The analysis of such data has however turned out to be quite a challenge. In drug discovery one would like to know what genes are involved in certain pathological processes, or what genes are affected by the intervention of a particular compound. A more basic question is 'How many genes are affected or changed?' It turns out that the answer to this basic question has a bearing on the other ones.

In the two-component model for the distribution of the test statistic the mixing parameter p_0, which represents the proportion unchanged genes, is not estimable without strong distributional assumptions, see Efron *et al.* [1]. In this model the probability density function (pdf) f^t of a test statistic t may be written as the weighted sum of the null distribution pdf f_0^t and the alternative distribution pdf f_1^t

$$f^t(x) = p_0 \times f_0^t(x) + (1 - p_0) f_1^t(x).$$

If, on the other hand, we know the value of p_0 we can estimate f_0^t through a bootstrap procedure Efron *et al.* [1], and thus obtain also f_1^t.

This mixing parameter has attracted a lot of interest lately. Indeed it is interesting for a number of applications.

1) Knowing the proportion changed genes in a microarray experiment is of interest in its own right. It gives an important summary measure of the amount of changes studied.

2) The use of the False Discovery Rate (FDR) in the inference has increased, and that quantity may be estimated as

$$\hat{F}DR(\alpha) = \hat{p}_0 \times P_{(L)} / p(\alpha)$$

, where '^' above a quantity means it is a parameter estimate, $P_{(L)}$ is the largest *p*-value not exceeding α and $p(\alpha)$ is the proportion significant (the proportion of *p*-values less than α), see also Storey (2001) [2].

A very similar concept is that of the qvalue, which according to Storey and Tibshirani (2003) [3] represents the expected proportion of false positives.

3) Knowing p_0 we may calculate the posterior probability of a gene being changed

$$p_1(x) = 1 - p_0 \frac{f_0^t(x)}{f^t(x)}$$

see Efron et al. [1].

4) In the samroc methodology Broberg (2003) [4] one calculates estimates of the false positive and false negative rates as

$$\hat{F}P = \hat{p}_0 \alpha$$

and

$$\hat{F}N = 1 - \hat{p}_0 (1 - \alpha) - p(\alpha)$$

where α is the significance level and $p(\alpha)$ is the proportion of genes judged significant.

Furthermore, the criterion

$$C = \sqrt{FP^2 + FN^2}$$

is minimised by choosing an optimal pair of values of the tuning parameter S_0 in the SAM statistic Tusher et al. (2001) [5] and the significance level α. The statistic is defined by

$$d = \frac{diff}{S_0 + S}$$

where *diff* is an effect estimate, e.g. a group mean difference, and *S* is a standard error.

Earlier research providing estimates of p_0 include Efron et al (2001) [1], Tusher et al (2001) [5], Storey (2001) [2], Allison et al (2002) [6], Storey and Tibshirani (2003) [3] and Pounds and Morris (2003) [7].

Methods

Denote the pdf of *p*-values by *f*, the proportion unchanged by p_0 and the distribution of the *p*-values corresponding the changed genes by f_1. Then the distribution of *p*-values may be written as

$$f(x) = p_0 \times 1 + (1 - p_0) f_1(x)$$

using the fact that *p*-values for the unchanged genes follow a uniform distribution.

The present approach is based on the moment generating function (mgf), which is a transform of a random distribution, which yields a function *R* characteristic of the distribution, cf. Fourier or Laplace transforms, e.g. Feller (1971) [8]. In fact the mgf is a Laplace transform. Knowing the transform means knowing the distribution. It is defined as the expectation (or the true mean) of the antilog transform of *s* times a random variable *X*, i.e. the expectation of e^{sX} or in mathematical notation:

$$R(s) = \int e^{sx} f(x) dx .$$

Transforming the above theoretical distribution yields the weighted sum of two transformed distributions:

$$R(s) = p_0 \frac{e^s - 1}{s} + (1 - p_0) \int e^{sx} f_1(x) dx$$

Denoting the first transform by *g(s)* and the second by $R_1(s)$ we finally have

$$R(s) = p_0 g(s) + (1 - p_0) R_1(s).$$

Now, the idea is to estimate these mgf's and to solve for p_0. In the above equation *R(s)* and *g(s)* can be estimated based on an observed vector of *p*-values and calculated exactly, respectively, while p_0 and $R_1(s)$ cannot be estimated independently. The estimable transform is, given the observed *p*-values $p = p_1, \ldots, p_n$, estimated by

$$\hat{R}_p(s) = \sum_{i=1}^{n} \frac{e^{sp_i}}{n} .$$

(From now on drop the index *p*.)

Instead of a straightforward mean as above, a smoothed estimate of the density will be tried elsewhere.

However, one can solve the above relation for p_0 for any value of *s*.

$$p_0 = \frac{R(s) - R_1(s)}{g(s) - R_1(s)} \qquad (1)$$

Let us do so for $s_n > s_{n-1}$, equate the two ratios defined by the right hand side in (1) and solve for $R_1(s_n)$. This gives the recursion

$$R_1(s_n) = \frac{R(s_n) g(s_{n-1}) - R(s_{n-1}) g(s_n) + R_1(s_{n-1}) (g(s_n) - R(s_n))}{g(s_{n-1}) - R(s_{n-1})} \qquad (2)$$

If we can find a suitable start for this recursion we should be in a position to approximate the increasing function $R_1(s)$ for $s = s_1 < s_2 < \dots < s_m$ in (0, 1]. Now, note that $1 \leq R(s)$, for any mgf, with close to equality for small values of s. Thus it makes sense to start the recursion with $R_1(s_1) = (1 + R(s_1))/2$. (In general, it will hold true that $1 < R_1(s_n) < R(s_n) < g(s_n)$, since f_1 puts weight to the lower range of the p-values at the expense of the higher range, the uniform puts equal weight, and f being a mixture lies somewhere in between.) We calculate g, R and R_1 for a series of values s in (0,1], e.g. for s in $(0.01, 0.0101, 0.0102, \dots, 1)$. The output from one data set appears in Figure 1. From (1) we obtain a series of estimates of p_0, and may take the mean as the final estimate.

Results

A simulation of data for 3000 genes was repeated 200 times for true p_0 values ranging from 0.6 to 0.95 using the R script from Broberg (2003) [4]. The current method *p0.mgf* was compared to the estimate presented in Storey and Tibshirani (2003), denoted qva, and to the bootstrap method from Storey (2002), implemented in the R package SAG [9, 10, 11]. These methods are both based on a comparison of the empirical p-value distribution to that of the uniform. There will likely be fewer p-values close to 1 in the empirical than in the null distribution, which is a uniform. The observed proportion of p-values exceeding some threshold value η over the expected proportion under the null hypothesis, $1 - \eta$, will estimate p_0. In fact, the ratio $\{1-F_e(\eta)\}/\{1-\eta\}$, F_e denoting the empirical distribution, will often be a good estimate of p_0 for an astutely chosen threshold η.

With the simulated data all methods perform rather well, see Table 1 and Figure 2.

Choosing a statistical method generally involves a trade-off between bias and variation. The proposed method misses its target by on an average 1.6% (underestimates p_0) , which is not as good as Storey's bootstrap method but better than qvalue, but it provides estimates with close to half the mean squared error of the alternatives. So if robustness is an issue then *p0.mgf* seems like a good choice. Minor perturbations of the data will not affect the result.

Discussion

In Broberg (2002) [12] an attempt was made to use the mgf for finding differentially expressed genes, with varying results. The main problem there lay in the few replicates. In the current application there is ample data to accurately capture the mgf, providing the p-values were obtained in a reliable fashion, e.g. by a warranted normal approximation, a bootstrap or a permutation method. Pounds and Morris [7] mention a case when a two-way ANOVA F-distribution was used and the distributional assumptions were not met. The estimate of p_0 gave an unrealistic answer. When permutation p-values were used instead their method gave a more realistic result. Similar caveats apply to any method based on p-values.

The current method may be used to provide a good starting point for a method like the EM algorithm. That algorithm is crucially dependent on a good start of the iteration. Such a combined algorithm remains to be explored. Another twist would be to take the estimate of R_1, fit a spline curve, predict the value of $R_1(0)$, which ought to be unity. Then, based on the difference $R_1(0) - 1$, adjust the value of $R_1(s_1)$ and reiterate (2). This will be tested elsewhere.

A further development would be to use the current approach directly on the test statistic, e.g. a *t*-test statistic, and to obtain *p*-values by modelling the null distribution instead of the common bootstrap approach. This has been tried in another context [13] and seems very encouraging.

The method is implemented in R and will appear in the package SAG v 1.2 [11].

References

1. Efron B, Tibshirani R, Storey JD, Tusher VG: **Empirical Bayes analysis of a microarray experiment.** *Journal of the American Statistical Association 2001,* **96:** 1151-1160

2. Storey JD: (2001) **A Direct Approach to False Discovery Rates** J Roy Stat Soc B, **64**, 479-498

3. Storey JD and Tibshirani R : Statistical significance for genomewide studies: *Proc. Natl. Acad. Sci.USA 2003*, **100:** 9440-9445

4. Broberg. P: **Statistical methods for ranking differentially expressed genes**. *Genome Biology* 2003, **4:**R41
[http://genomebiology.com/2003/4/6/R41]

5. Tusher V.G., Tibshirani R., Chu G: **Significance analysis of microarrays applied to the ionizing radiation response.** *Proc. Natl. Acad. Sci.USA 2001*, **98:** 5116-5121

6. Allison DB, Gadbury GL, Moonseong H, Fernandez JR, Cheol-Koo L, Prolla TA and Weindruch RA: **A mixture model approach for the analysis of microarray gene expression data.** *Computational Statistics and Data Analysis 2002*, **39**, 1-20

7. Pounds S and Morris SW: **Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of *p*-values.** *Bioinformatics 2003*, Vol **19**, 10, 1236-1242

8. Feller W: *An Introduction to Probability Theory and Its Applications, Volume 2.* Second Edition. New York: Wiley, 1971

9. **The R project**
[www.cran.r-project.org]

10. Ihaka R, Gentleman R: (1996) **R: A language for data analysis and graphics.** *Journal of Computational and Graphical Statistics 1996,* **5:** 299-314

11. The SAG homepage
[http://home.swipnet.se/pibroberg/expression_hemsida1.html]

12. Broberg P: (2002) **Ranking genes with respect to differential expression.** *Genome Biology*, **3:**preprint0007.1-0007.23

13. Efron B: (2003) **Large-Scale Simultaneous Hypothesis Testing: the choice of a null hypothesis**. Report Stanford [http://www-stat.stanford.edu/~brad/papers/Large-Scale_2003.pdf]

Figures

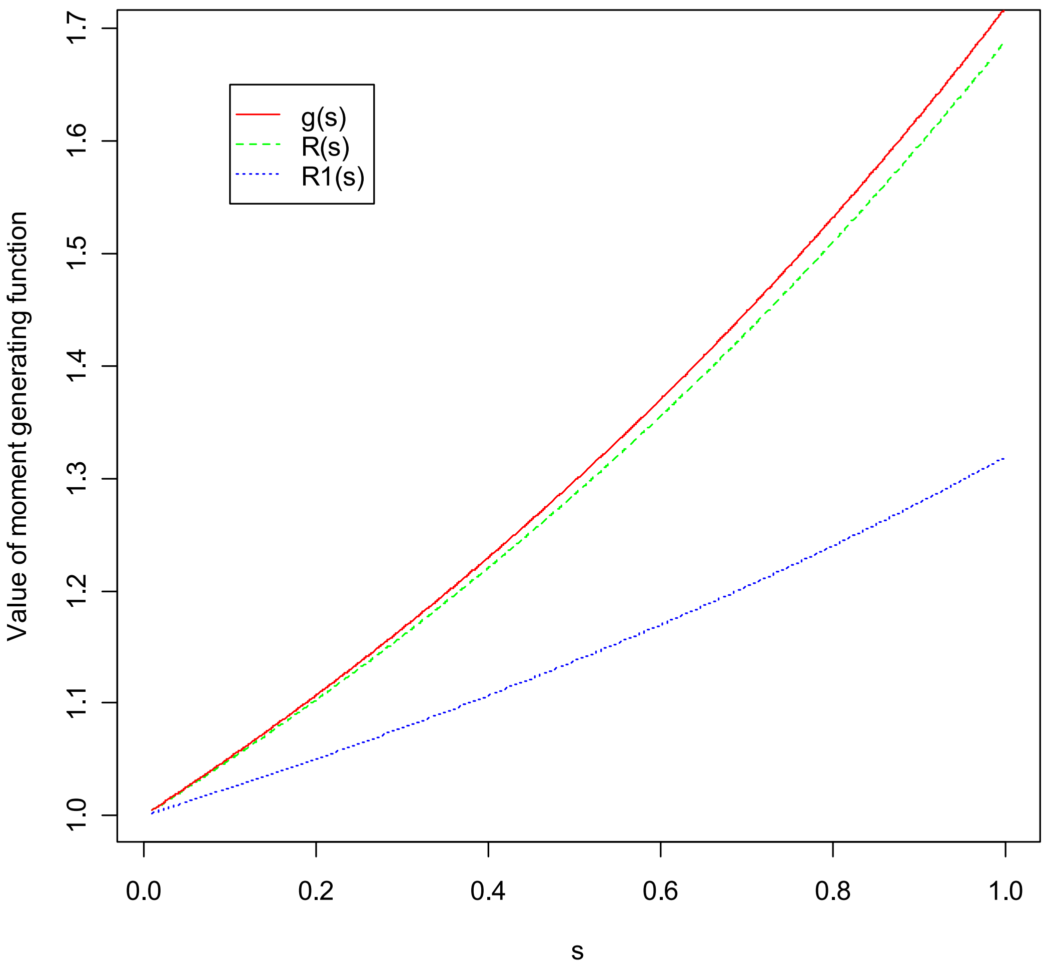

Figure 1. Estimated moment generating functions (mgf's). Given an observed vector of *p*-values it is possible to calculate mgf's for the observed distribution *f* (*R*) and the unobserved distribution f_1 (R_1), and without any observations we can calculate the mgf for the uniform (*g*).

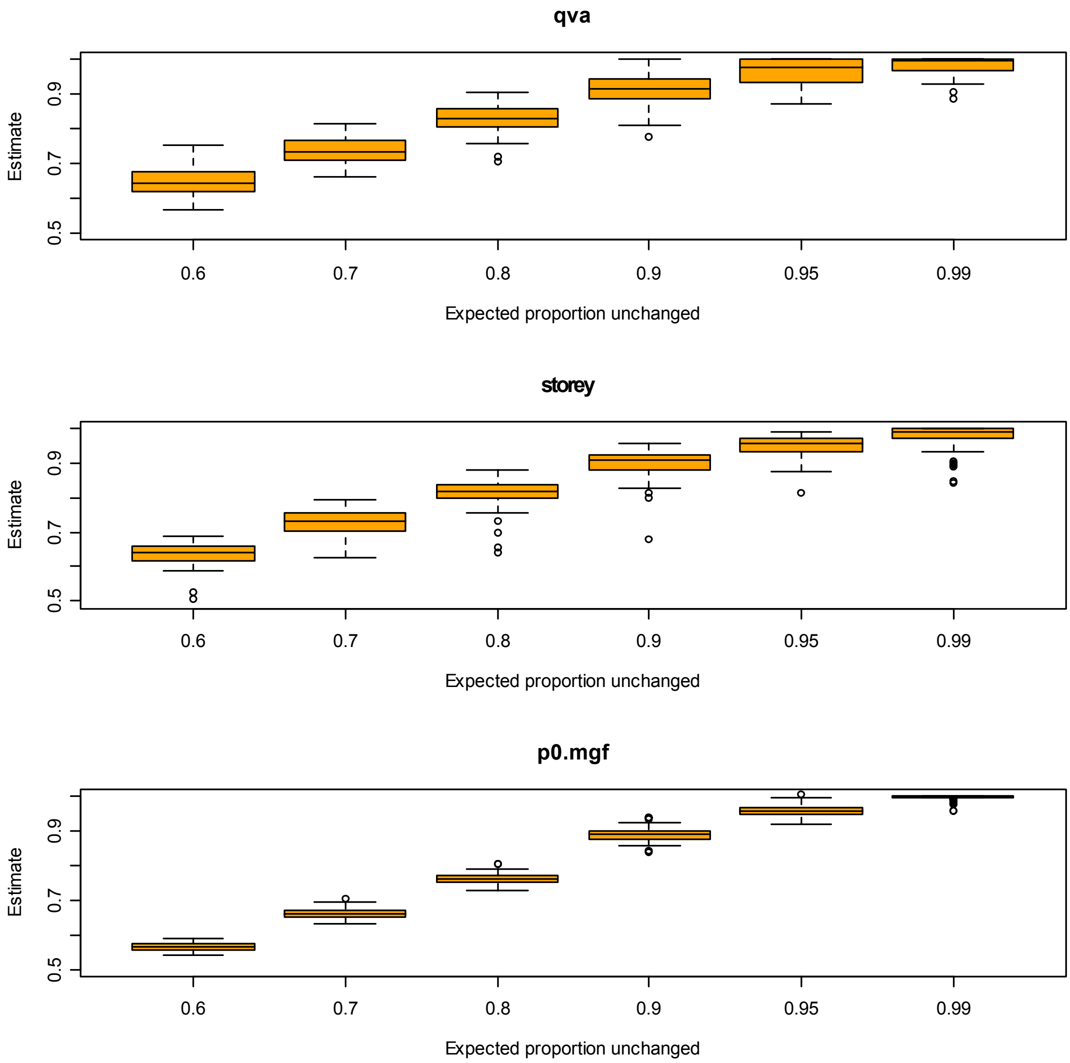

Figure 2. Boxplots of the simulation results. A simulation model of real-life microarray data was used to give data where the expected proportion of changed genes was set at 60, 70, 80, 90, 95 or 99%. The proposed method, denoted *p0.mgf* gave low bias and low variance over the whole range.

Tables

	qva	*storey*	*p0.mgf*
mean	-0.024	-0.0078	0.016
Sd	0.044	0.045	0.024

Table 1. Over-all results of simulations. The summary statistics of the difference between target value and its estimate show a rather good performance for all methods, with *p0.mgf* having the second smallest bias and the smallest variation.