Sample Size Estimation for Microarray Experiments Using the `ssize` package.

Gregory R. Warnes
email:`gregory.r.warnes@pfizer.com`

October 28, 2005

Abstract

RNA Expression Microarray technology is widely applied in biomedical and pharmaceutical research. The huge number of RNA concentrations estimated for each sample make it difficult to apply traditional sample size calculation techniques and has left most practitioners to rely on rule-of-thumb techniques. In this paper, we briefly describe and then demonstrate a simple method for performing and visualizing sample size calculations for microarray experiments as implemented in the `ssize` R package, which is available from the Bioconductor project (`http://www.bioconductor.org`) web site.

1 Note

This document is a simplified version of the manuscript

> Warnes, G. R., Liu, P. (2005) Sample Size Estimation for Microarray Experiments, *submitted to Biometrics*.

Please refer to that document for a detailed discussion of the sample size estimation method.

2 Introduction

High-throughput microarray experiments allow the measurement of expression levels for tens of thousands of genes simultaneously. These experiments

have been used in many disciplines of biological research, including as neuroscience (Mandel *et al.*, 2003), pharmacogenomic research, genetic disease and cancer diagnosis (Heller, 2002). As a tool for estimating gene expression and single nucleotide polymorphism (SNP) genotyping, microarrays produce huge amounts of data which are providing important new insights.

Microarray experiments are rather costly in terms of materials (RNA sample, reagents, chip, etc), laboratory manpower, and data analysis effort. It is critical, therefore, to perform proper experimental design, including sample size estimation, before carrying out microarray experiments. Since tens of thousands of variables (gene expressions) may be measured on each individual chip, it is essential to take into account multiple testing and dependency among variables when calculating sample size.

3 Method

3.1 Overview

Warnes and Liu (2005) provides a simple method for computing sample size for micrarray experiments, and reports on a sereies of simulations demonstrating the performinace of the method. The key component of this method is the generation of cumulative plot of the proportion of genes achieving a desired power as a function of sample size, based on simple gene-by-gene calculations. While this mechanism can be used to select a sample size numerically based on pre-specified conditions, its real utility is as a visual tool for helping clients to understand the trade off between sample size and power. In our consulting work, this latter use as a visual tool has been exceptionally valuable in helping scientific clients to make the difficult trade offs between experiment cost and statistical power.

3.2 Assumptions

In the current implementation, we assume that a microarray experiment is set up to compare gene expressions between one treatment group and one control group. We further assume that microarray data has been normalized and transformed so that the data for each gene is sufficiently close to a normal distribution that a standard 2-sample pooled-variance t-test will reliably detect differentially expressed genes. The tested hypothesis for each gene is:

$$H_0 : \mu_T = \mu_C$$

versus

$$H_1 : \mu_T \neq \mu_C$$

where μ_T and μ_C are means of gene expressions for treatment and control group respectively.

3.3 Computations

The proposed procedure to estimate sample size is:

1. Estimate standard deviation (σ) for each gene based on *control samples* from existing studies performed on the same biological system.

2. Specify values for

   (a) minimum effect size, Δ, (log of fold-change for log-transformed data)

   (b) maximum family-wise type I error rate, α

   (c) desired power, $1 - \beta$.

3. Calculate the per-test Type I error rate necessary to control the family-wise error rate (FWER) using the Bonferroni correction:

$$\alpha_G = \frac{\alpha}{G} \tag{1}$$

   where G is the number of genes on the microarray chip.

4. Compute sample size separately for each gene according to the standard formula for the two-sample t-test with pooled variance:

$$\begin{aligned} & 1 - \beta \\ & = 1 - T_{n_1+n_2-2}\left(t_{\alpha_G/2,n_1+n_2-2}|\frac{\Delta}{\sigma\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right) \\ & \quad + \ T_{n_1+n_2-2}\left(-t_{\alpha_G/2,n_1+n_2-2}|\frac{\Delta}{\sigma\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right) \end{aligned} \tag{2}$$

   where $\mathrm{T}_d(\bullet|\theta)$ is the cumulative distribution function for non-central t-distribution with d degree of freedom and the non-centrality parameter θ.

5. Summarize the necessary sample size across all genes using a cumulative plot of required sample size verses power. An example of such a plot is given in Figure 2 for which we assume equal sample size for the two groups, $n = n_1 = n_2$.

On the cumulative plot, for a point with x coordinate n, the y coordinate is the proportion of genes which require a sample size smaller than or equal to n, or equivalently the proportion of genes with power greater than or equal to the specified power $(1-\beta)$ at sample size n. This plot allows users to visualize the relationship between power for all genes and required sample size in a single display. A sample size can thus be selected for a proposed microarray experiment based on user-defined criterion. For the plot in Figure 2, for example, requiring 70% of genes to achieve the 80% power yields a sample size of 10.

Similar plots can be generated by fixing the sample size and varying one of the other parameters, namely, significance level (α), power $(1 - \beta)$, or minimum effect size (Δ). Two such plots are shown in Figures 3 and 4.

4 Example

First, we need to load the `ssize` library:

```
> library(ssize)
> library(xtable)
> library(gdata)
```

As part of the `ssize` library, I've provided an example data set containing gene expression values for smooth muscle cells from a control group of untreated healthy volunteers processed using Affymetrix U95 chips and normalized per the Robust Multi-array Average (RMA) method of Irizarry *et al.* (2003).

```
> data(exp.sd)
> str(exp.sd)
```

```
 Named num [1:12625] 0.1461 0.2107 0.1708 0.0771 0.1304 ...
 - attr(*, "names")= chr [1:12625] "1000_at" "1001_at" "1002_f_a" "1003_s_a" ...
```

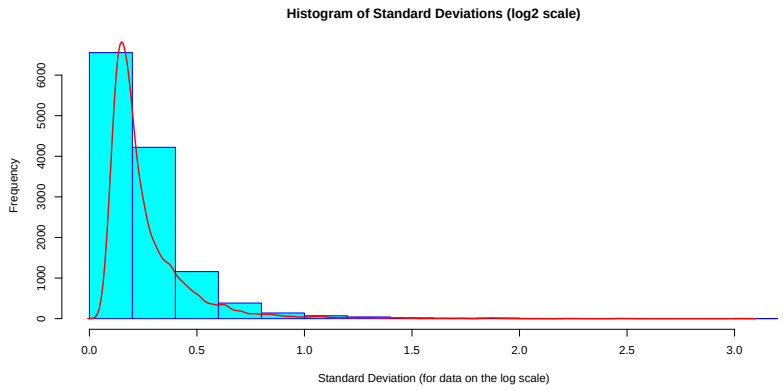
This data was calculated via something like

```
library(affy)
setwd("/data/rstat-data/Standard_Affymetrix_Analysis/GT_methods_050316/WORK")
load("probeset_data.Rda")
expression.values <- exprs(probeset.data)
covariate.data <- pData(probeset.data)
controls <- expression.values[,covariate.data$GROUP=="Control"] #$
exp.sd <- apply(controls, 1, sd)
```

Lets see what the distribution looks like:

Figure 1: Distribution of exp.sd

```
> hist(exp.sd, n = 20, col = "cyan", border = "blue", main = "",
+     xlab = "Standard Deviation (for data on the log scale)")
> dens <- density(exp.sd)
> lines(dens$x, dens$y * par("usr")[4]/max(dens$y), col = "red",
+     lwd = 2)
> title("Histogram of Standard Deviations (log2 scale)")
```



Note that this distribution is right skewed, even though it is on the $\log_2$ scale.

To make the computations run faster, we'll only use the standard deviations for the first 1000 genes on the chip. Everything will work if you skip this, but it will take longer.

```
> exp.sd <- exp.sd[1:1000]
```

There are 6 functions available in the `ssize` package.

```
> "?"(pow)
```

```
pow(sd, n, delta, sig.level, alpha.correct = "Bonferonni")
power.plot(x, xlab = "Power", ylab = "Proportion of Genes with Power >= x",
    marks = c(0.7, 0.8, 0.9), ...)

ssize(sd, delta, sig.level, power, alpha.correct = "Bonferonni")
ssize.plot(x, xlab = "Sample Size (per group)",
           ylab = "Proportion of Genes Needing Sample Size <= n",
           marks = c(2, 3, 4, 5, 6, 8, 10, 20), ...)

delta(sd, n, power, sig.level, alpha.correct = "Bonferonni")
delta.plot (x, xlab = "Fold Change",
            ylab = "Proportion of Genes with Power >= 80\% at Fold Change=del
            marks = c(1.5, 2, 2.5, 3, 4, 6, 10), ...)
```

You will note that there are three pairs.

pow, power.sd compute and display a cumulative plot of the fraction of genes achieving a specified power for a fixed sample size (`n`), effect size (`delta`), and significance level (`sig.level`).

ssize,ssize.plot compute and display a cumulative plot of the fraction of genes for which a specified sample size is sufficient to achieve a specified power (`power`), effect size (`delta`), and significance level (`sig.level`).

delta,delta.plot compute and display a cumulative plot of the fraction of genes which can achieve a specified power (`power`), for a specified sample size (`n`), and significance level (`sig.level`) for a range of effect sizes.

So, now lets see the functions in action.

First, lets define the values for which we will be investigating:

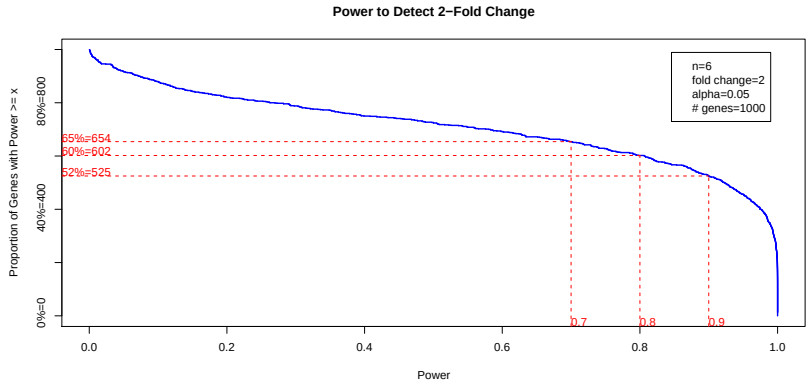
```
> n <- 6
> fold.change <- 2
> power <- 0.8
> sig.level <- 0.05
```

Figure 2: Effect of Sample Size on Power

```
> all.power <- pow(sd = exp.sd, n = n, delta = log2(fold.change),
+     sig.level = sig.level)

.....................................................................................

> power.plot(all.power, lwd = 2, col = "blue")
> xmax <- par("usr")[2] - 0.05
> ymax <- par("usr")[4] - 0.05
> legend(x = xmax, y = ymax, legend = strsplit(paste("n=", n, ",",
+     "fold change=", fold.change, ",", "alpha=", sig.level, ",",
+     "# genes=", nobs(exp.sd), sep = ""), ",")[[1]], xjust = 1,
+     yjust = 1, cex = 1)
> title("Power to Detect 2-Fold Change")
```



1. What is the power for 6 patients per group with $\delta = 1.0$, $\alpha = 0.05$?

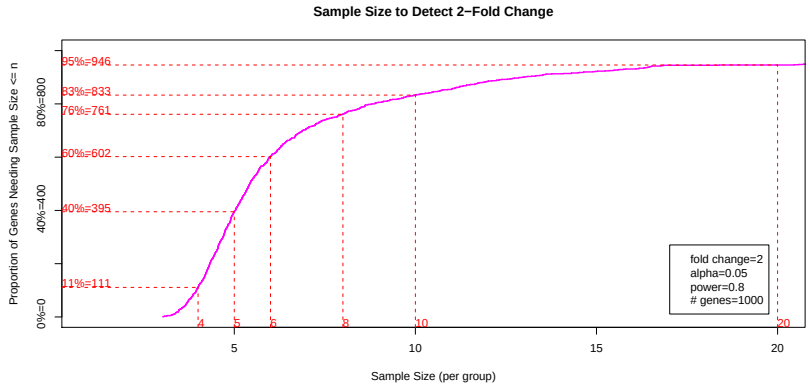
2. What is the necessary per-group sample size for 80% power when $\delta = 1.0$, and $\alpha = 0.05$?

Figure 3: Sample size required to detect a 2-fold treatment effect.

```
> all.size <- ssize(sd = exp.sd, delta = log2(fold.change), sig.level = sig.leve
+     power = power)

.....................................................................................

> ssize.plot(all.size, lwd = 2, col = "magenta", xlim = c(1, 20))
> xmax <- par("usr")[2] - 1
> ymin <- par("usr")[3] + 0.05
> legend(x = xmax, y = ymin, legend = strsplit(paste("fold change=",
+     fold.change, ",", "alpha=", sig.level, ",", "power=", power,
+     ",", "# genes=", nobs(exp.sd), sep = ""), ",")[[1]], xjust = 1,
+     yjust = 0, cex = 1)
> title("Sample Size to Detect 2-Fold Change")
```



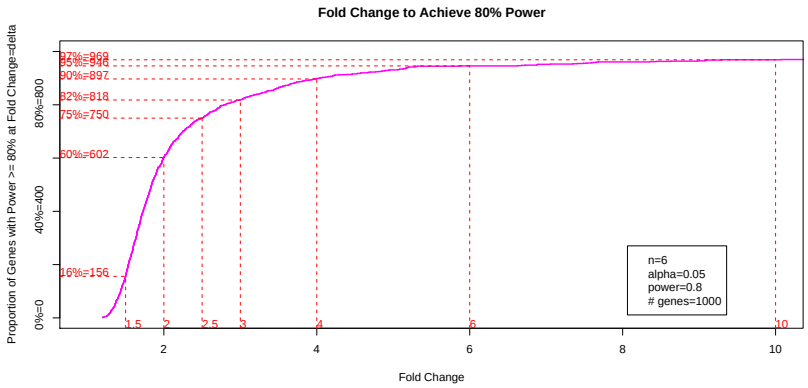
3. What is necessary fold change to achieve 80% with $n = 6$ patients per group, when $\delta = 1.0$ and $\alpha = 0.05$?

Figure 4: Given sample size, this plot allows visualization of the fraction of genes achieving the specified power for different fold changes.

```
> all.delta <- delta(sd = exp.sd, power = power, n = n, sig.level = sig.level)

.....................................................................................

> delta.plot(all.delta, lwd = 2, col = "magenta", xlim = c(1, 10))
> xmax <- par("usr")[2] - 1
> ymin <- par("usr")[3] + 0.05
> legend(x = xmax, y = ymin, legend = strsplit(paste("n=", n, ",",
+     "alpha=", sig.level, ",", "power=", power, ",", "# genes=",
+     nobs(exp.sd), sep = ""), ",")[[1]], xjust = 1, yjust = 0,
+     cex = 1)
> title("Fold Change to Achieve 80% Power")
```



5 Modifications

While the `ssize` package has been implemented using the simple 2-sample pooled t-test, you can easily modify the code for other circumstances. Simply replace the call to `power.t.test` in each of the functions `pow`,`ssize`,`delta` with the appropriate computation for the desired experimental design.

6 Future Work

Peng Liu is currently developing methods and code for substituting False Discovery Rate for the Bonferonni multiple comparison adjustment.

7 Contributions

Contributions and discussion are welcome.

Acknowledgment

This work was supported by Pfizer Global Research and Development.

References

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing, *Journal of Royal Statistical Society B*, **57:1**, 289-300.

Dow,G.S. (2003) Effect of sample size and p-value filtering techniques on the detection of transcriptional changes induced in rat neuroblastoma (NG108) cells by mefloquine, *Malaria Journal*, **2**, 4.

Heller, M. J. (2002) DNA microarray technology: devices, systems, and applications, *Annual Review in Biomedical Engineering*, **4**, 129-153.

Hwang,D., Schmitt, W. A., Stephanopoulos, G., Stephanopoulos, G. (2002) Determination of minimum sample size and discriminatory expression patterns in microarray data, *Bioinformatics*, **18:9**, 1184-1193.

Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U., Speed, T.P. (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data, *Biostatistics*, **4:2**, 249-264.

Mandel, S., Weinreb, O., Youdim, M. B. H. (2003) Using cDNA microarray to assess Parkinson's disease models and the effects of neuroprotective drugs, *TRENDS in Pharmacological Sciences*, **24:4**, 184-191.

Yang, Y. H., Speed, T. Design and analysis of comparative microarray experiments *Statistical analysis of gene expression microarray data*, Chapman and Hall, 51.

Storey, J., (2002) A direct approach to false discovery rates, *Journal of Royal Statistical Society B*, **64:3**, 479-498.

Warnes, G. R., Liu, P. (2005) Sample Size Estimation for Microarray Experiments, *submitted to Biometrics*.

Yang, M. C. K., Yang, J. J., McIndoe, R. A., She, J. X. (2003) Microarray experimental design: power and sample size considerations, *Physiological Genomics*, **16**, 24-28.

Zien, A., Fluck, J., Zimmer, R., Lengauer, T. (2003) Microarrays: how many do you need?, *Journal of Computational Biology*, **10:3-4**, 653-667.