Sample Size and Power Analysis for Microarray Studies

Maarten van Iterson and Renée de Menezes
Center for Human and Clinical Genetics,
Leiden University Medical Center, The Netherlands
Package `SSPA`, version 0.1.3

April 22, 2010

Contents

1 Introduction

This document shows the functionality of the R-package `SSPA`. The package performs the power and sample size analysis method as described by [2, 3]. Our implementation allows for fast and realistic estimates of power and sample size for microarray experiments, given pilot data. By means of two simple commands (`pilotData(), sampleSize()`), a researcher can read their data in and compute the desired estimates. Other functions are provided to facilitate interpretation of results.

Given a set of test statistics from the pilot data, the knowledge of their distribution under the null hypothesis and the sample size used to compute them, the method estimates the power for a given false discovery rate. The multiple testing problem is controlled through the adaptive version of the Benjamini and Hochberg method [1]. For more details about the implementation and method we refer to [9, 2, 3]. In [8] we describe two biological case studies using the package `SSPA`.

For comparison we implement the power and sample size estimation method proposed by Ruppert [6] which uses a different estimation approach. Two ad-

ditional packages need to be installed for using this method namely `quadprog` and `splines`.

2 Basic Example

We demonstrate the functionality of this package by using the preprocessed gene expression data from the leukemia ALL/AML study of [4] from the `multtest` package. To load the leukemia dataset, use `data(golub)`, and to view a description of the experiments and data, type `?golub`. The number of samples per group are shown by `table(golub.cl)` where ALL is class 0 and AML class 1.

```
> library(multtest)
> data(golub)
> table(golub.cl)

golub.cl
 0  1
27 11
```

The required input for the sample size and power analysis is a vector of test-statistics and the sample sizes used to compute them. The test-statistics are obtained by a differentially gene expression analysis using one of the available packages like `limma, maanova, multtest` (it is also possible to import the vector of test-statistics in `R` if they are calculated using any other software than `R`). Here we will use the function `mt.teststat` from the `multtest` package to obtain a vector of test-statistics from the leukemia data.

```
> tst <- mt.teststat(golub, golub.cl)
```

The first step in doing the sample size and power analysis is creating a object of class-`PilotData` which will contain all the necessary information needed for the following power and sample size analysis; a vector of test-statistcs and the sample sizes used to compute them. A user-friendly interface for creating an object of `PilotData` is available as `pilotData()`.

```
> library(SSPA)
> pd <- pilotData(name = "ALL/AML", testStatistics = tst, sampleSizeA = 11,
     sampleSizeB = 27)
```

Several ways of viewing the content of the `PilotData`-object are possible either graphically or using a `show`-method by just typing the name of the created object of `PilotData`:

```
> pd
```

```
> layout(matrix(c(1, 2), nrow = 2))
> hist(pd, cex.main = 1)
> plot(pd, cex.main = 1)
```

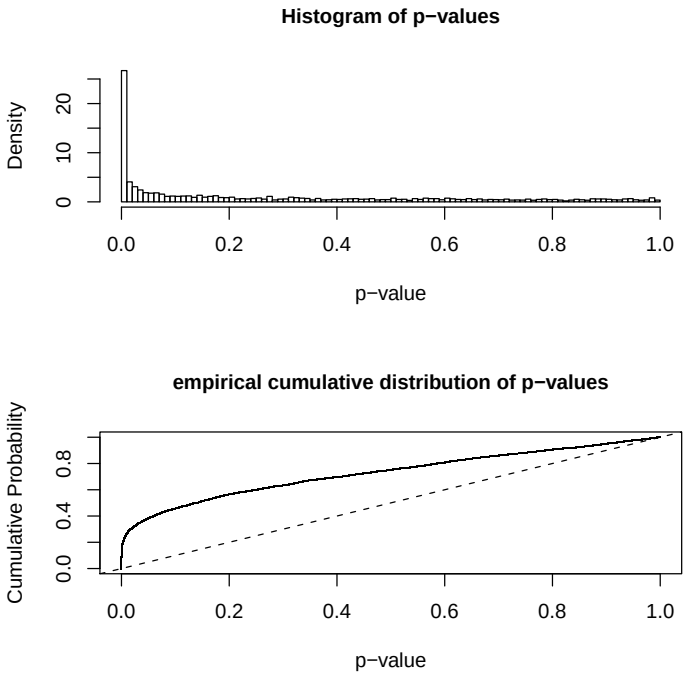

Figure 1: Histogram and ECDF plot of p-values.

```
An object of class "PilotData"
Experiment name:            ALL/AML
Number of test-statistics:  3051
Effective sample size:      7.82
Null distribution:      normal
```

A histogram of p-values is obtained by just calling `hist(pd)` and an empirical cumulative distribution of p-values by `plot(pd)`.

Now we can create an object of class `SampleSize` which will perform the estimation of the proportion of non-differentially expressed genes and the density of effect sizes. Several options are possible `?sampleSize`. The default method for estimation of the proportion of non-differentially expressed genes is the method proposed by [5] as implemented by `convest` from the package `limma`. Additionally the method by [7] and [3] are available, also a user-defined proportion can be given. Again a generic `show`-method is implemented.

```
> plotEffectSize(ss, type = "l")
```

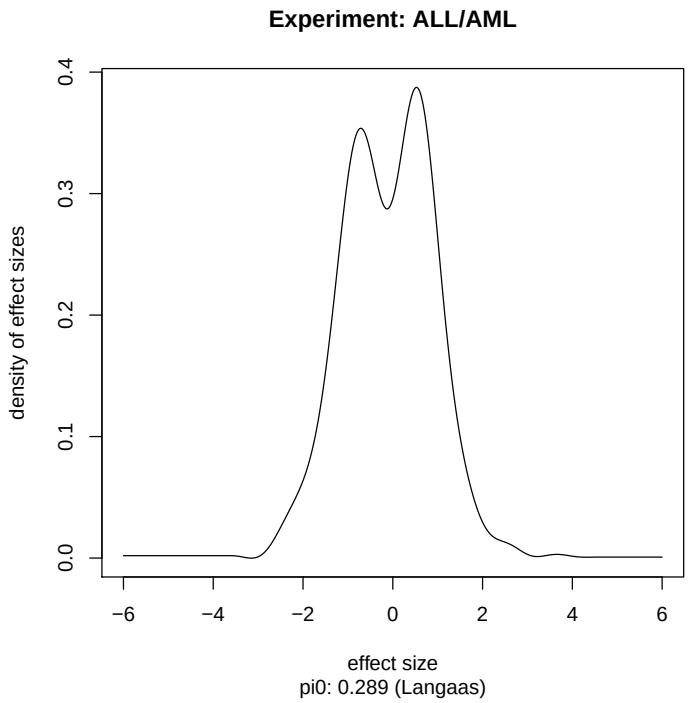

Figure 2: Density of effect-sizes and estimated average power.

```
> ss <- sampleSize(pd)
> ss

An object of class "SampleSize"
Distribution of effect sizes is estimated from -6 to 6 using 1024 points.
Method for estimation of the proportion of non-differntially expressed: "Langaas
Fraction of non-differently expressed genes: 0.4702 (adjusted=0.2887).
Kernel used in the deconvolution is "fan" with bandwidth 0.353.
```

The density of effect size can be shown by a call to `plotEffectSize()`

Estimating the average power for other sample sizes then that of the pilot-data can be performed with the `Power()`-function. The user can also give the desired false discovery rate level or possible multilpe false discovery rate levels.

```
> layout(matrix(c(1:2), nrow = 2))
> pwr <- Power(ss, plot = FALSE, samplesizes = c(5, 10, 15, 20),
    fdr = 0.01)
> plot(c(5, 10, 15, 20), pwr, ylim = c(0, 1), type = "b", ylab = "Power",
    xlab = "Sample size per group")
> legend("bottomright", colnames(pwr), col = c(1:ncol(pwr)), pch = 1,
    lty = 1)
> pwr <- Power(ss, plot = FALSE, samplesizes = c(5, 10, 15, 20),
    fdr = c(0.01, 0.05))
> matplot(c(5, 10, 15, 20), pwr, ylim = c(0, 1), type = "b", pch = 1,
    ylab = "Power", xlab = "Sample size per group")
> legend("bottomright", colnames(pwr), col = c(1:ncol(pwr)), pch = 1,
    lty = 1)
```

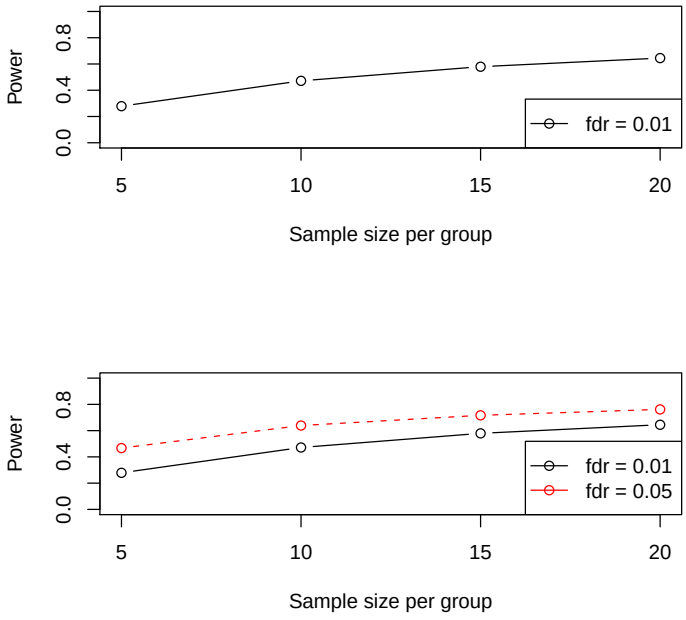

Figure 3: Density of effect-sizes and estimated average power.

```
> layout(matrix(c(1, 2), nrow = 1))
> ss <- sampleSize(pd, method = "Ferreira", pi0 = seq(0.05, 0.5,
     0.05), doplot = TRUE)
> plotEffectSize(ss)
```

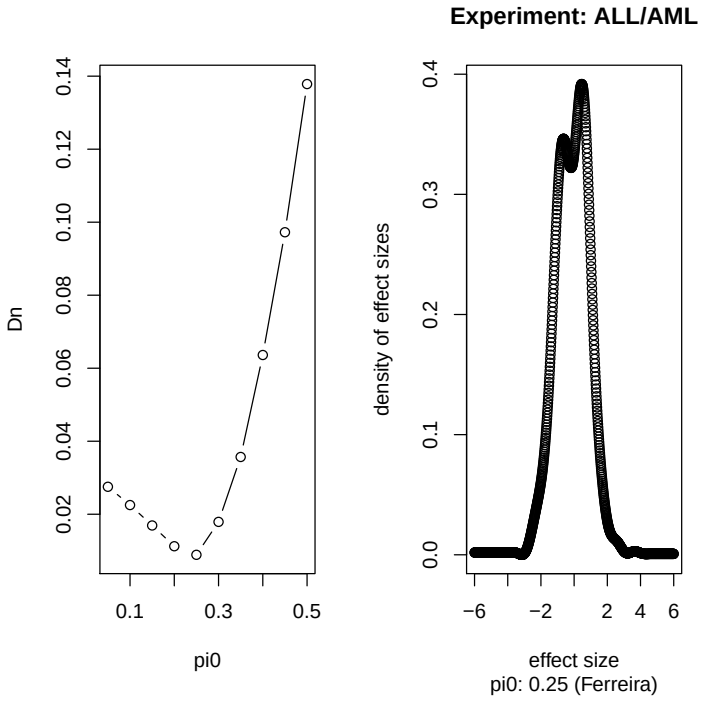

Figure 4: *Left panel:* The global minimum shows the π_0 estimate. *Right panel:* The estimation of the density of effect sizes using π_0 estimate obtained with Ferreira's method.

3 Additional functionality

3.1 Ferreira's π_0 estimate

Ferreira *et al.* propose a semi-parametric method to estimate the proportion of non-differentially expressed genes[2, 3]. Figure 4 output of the π_0 estimation method of Ferreira is shown.

3.2 Rupperts estimation method

For using the method proposed by Ruppert [6] two additional packages need to be installed for using this method namely `quadprog` and `splines`. Using `sampleSize(pd, method="Ruppert", nKnots = 11, bDegree = 3)` both π_0 and

the density of effect sizes is estimated by Ruppert's method.

References

[1] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. Roy. Stat. Soc. Ser. B*, 57:289–300, 1995.

[2] J.A. Ferreira and A. Zwinderman. Approximate Power and Sample Size Calculations with the Benjamini-Hochberg Method. *Int. J. Biostat.*, 2(1), 2006.

[3] J.A. Ferreira and A. Zwinderman. Approximate Sample Size Calculations with Microarray Data: An Illustration. *Statistical Applications in Genetics and Molecular Biology*, 5(1), 2006.

[4] T R Golub, D K Slonim, P Tamayo, C Huard, M Gaasenbeek, J P Mesirov, H Coller, M L Loh, J R Downing, M A Caligiuri, C D Bloomfield, and E S Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–7, 1999.

[5] M. Langaas, B.H. Lindqvist, and E. Ferkingstad. Estimating the proportion of true null hypotheses, with application to DNA microarray data. *J.R. Statist. Soc. B*, 67(4):555–572, 2005.

[6] D. Ruppert, D. Nettleton, and J.T.G. Hwang. Exploring the information in p-values for the analysis and planning of multiple-test experiments. *Biometrics*, 63(2):483–95, 2007.

[7] J.D. Storey. A direct approach to false discovery rates. *J.R. Statist. Soc. B*, 64:479–498, 2002.

[8] M. van Iterson, P.A.C. 't Hoen, P. Pedotti, G.J.E.J. Hooiveld, J.T. den Dunnen, G.J.B. van Ommen, J.M. Boer, and R.X. Menezes. Relative power and sample size analysis on gene expression profiling data. *BMC Genomic*, 2009. Submitted for publication.

[9] van Iterson M., J.A. Ferreira, and R.X. de Menezes. SSPA: Power and Sample Size Analysis package. *Bioinformatics*, 2009. Submitted for publication.