Bioconductor's DEDS package

Yuanyuan Xiao[1] and Yee Hwa Yang[2]

March 31, 2005

Departments of [1]Biopharmaceutical Sciences and [2]Medicine
University of California, San Francisco
yxiao@itsa.ucsf.edu
jean@biostat.ucsf.edu

Contents

1 Overview **1**

2 Case study 1: Affymetrix Spike-in Experiment **2**
- 2.1 Data . . . 2
- 2.2 The `deds.stat.linkC` and `deds.stat` functions . . . 3
- 2.3 The plotting functions `pairs.DEDS` and `hist.deds` . . . 4

3 Case study 2: ApoA1 Experiment **4**
- 3.1 Data . . . 4
- 3.2 Application of the `deds.stat.linkC` function . . . 5

4 Statistical functions **5**
- 4.1 The `comp.t` and other related functions . . . 5
- 4.2 The `comp.stat` function . . . 7
- 4.3 Computing adjusted p values and q values for associated statistics . . . 8

1 Overview

This document provides a tutorial for the `DEDS` package for assessment of differential expression (DE) in microarray data.

Introduction to DEDS. There are numerous statistics in the microarray literature that rank genes in evidence of DE, to name a few, fold change (FC), t statistic, SAM (Tusher et al. (2001)). selecting a best statistic or ordering statistics in terms of merit has been problematic. No characterizations of microarray data that indicate desirability of a specific choice exist and, likewise, no comparisons across a sufficiently wide range of benchmark datasets have been undertaken. To avoid making fairly arbitrary choices when deciding which ranking statistic to use and to borrow strength across related measures, we apply a novel ranking scheme that assesses DE via distance synthesis (DEDS) of different related measures. Further details on the packages are given in Yang et al. (2004).

Functionalities in `DEDS`. The `DEDS` package implements the DEDS procedure and several common statisics, such as, FC, t statistics, SAM, F statistics, B statistics (Lönnstedt and Speed (2001)) and moderated F and t statisitcs (Smyth et al. (2003)), for the analysis of DE in microarrys.

Case study. We demonstrate the functionality of the `DEDS` package using two microarray experiments: Affymetrix spike-in (Irizarry et al. (2003)) and ApoA1 (Dudoit et al. (2002b)).

Related packages in Bioconductor. The Bioconductor packages `marrayClasses`, `marrayInput` and `marrayNorm` provide functions for reading and normalizing spotted microarray data. The package `affy` provides functions for reading and normalizing Affymetrix microarray data.

Help files. As with any R package, detailed information on functions, classes and methods can be obtained in the help files. For instance, to view the help file for the function `comp.FC` in a browser, use `help.start()` followed by `?comp.FC`.

2 Case study 1: Affymetrix Spike-in Experiment

We demonstrate the functionality of this package using gene expression data from the Affymetrix spike-in experiment. To load the dataset, use `data(affySpikeIn)`, and to view a description of the experiements and data, type `?affySpikeIn`.

```
> library(DEDS)
> data(affySpikeIn)
```

2.1 Data

The spike-in experiment represents a portion of the data used by Affymetrix to develop their MAS 5.0 preprocessing algorithm. The whole dataset features 14 human genes spiked-in at a series of 14 known concentrations ($0, 2^{-2}, 2^{-1}, \ldots, 2^{10}$ pM) according to a Latin square design among 12612 null genes. Each "row" of the Latin square (given spike-in gene at a given concentration) was replicated (typically 3 times, two rows 12 times, 59 arrays in total for the whole dataset). Further details are available at `http://www.affymetrix.com/analysis/download_center2.affx`. Here we showcase a portion of this dataset that presents a two-group comparison problem with 12 replicates in each group. Therefore, `affySpikeIn` contains the gene expression data for the 24 samples and 12,626 genes retained after RMA probe level summaries. The dataset includes

- `affySpikeIn`: a $12,626 \times 24$ matrix of expression levels;
- `affySpikeIn.gnames`: a vector of gene identifiers of length $12,626$;
- `affySpikeIn.L`: a vector of class labels (0 for class 1, 1 for class 2).
- `spikegene`: a vector that shows that location and identities of the 14 spiked genes.

```
> dim(affySpikeIn)
```

```
[1] 12626    24

> affySpikeIn.L

 [1] 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1

> spikedgene

37777_at   684_at  1597_at 38734_at 39058_at 36311_at 36889_at  1024_at
    7843    12244      658     8810     9137     6363     6946       28
36202_at 36085_at 40322_at   407_at  1091_at  1708_at
    6253     6134    10414    10795      102      780
```

2.2 The deds.stat.linkC and deds.stat functions

The `deds.stat.linkC` and `deds.stat` functions are the main functions that carry out the DEDS procedure. The former wraps around a C function and is therefore quicker than the latter; the latter does the computation solely in R and is slower but is more flexible in fine-tuning parameters for statisital measures. The user is recommended to use `deds.stat.linkC` for efficiency purpose. We describe the most important arguments in the function `deds.stat.linkC` below (see also `?deds.stat.linkC`):

`X`: A matrix, in the case of gene expression data, rows correspond to N genes and columns to p mRNA samples.

`L`: A vector of integers corresponding to observation (column) class labels. For k classes, the labels must be integers between 0 and $k-1$.

`B`: The number of permutations.

`tests`: A character vector specifying the statistics for synthesis of DEDS. `test` could be any of the following: ‘‘`t`’’ (t statistics), ‘‘`f`’’ (F statistics), ‘‘`fc`’’ (FC), ‘‘`sam`’’ (SAM), ‘‘`modt`’’ (moderated t statistics), ‘‘`modF`’’ (moderated F statistics) and ‘‘`B`’’ (B statistics). As a default, DEDS synthesizes t statistics, FC and SAM.

`tail`: A character string specifying the type of rejection region; choices include ‘‘`abs`’’, ‘‘`higher`’’ and ‘‘`lower`’’.

`adj`: A character string specifying the type of multiple testing adjustment; choices include ‘‘`fdr`’’ for returning q values controling False Discovery Rate (FDR; see Benjamini and Hochberg (1995)) and ‘‘`adjp`’’ for adjusted p values (see Dudoit et al. (2002a)) controling family wise type I error rate.

We apply `deds.stat.linkC` on the `affySpikeIn` dataset using 400 permutations and computing associated q values for each gene. Here, as a default, DEDS synthesizes t statistics,FC and SAM. The information of the top 20 genes can be printed out using the function `topgenes`; note that the rankings of genes by DEDS balance among the three measures it synthesizes, t statistics, FC and SAM.

```
> deds.affy <- deds.stat.linkC(affySpikeIn, affySpikeIn.L, B = 400)
```

```
> topgenes(deds.affy, number = 20, genelist = affySpikeIn.gnames)
```

```
                 Name geneOrder       DEDS          t         fc       sam
1              684_at     12244 0.00000000 171.858505  7.2235484 82.349174
2            36085_at      6134 0.01685367 -19.250088 -0.7648563 -8.954151
3            36202_at      6253 0.01685367 -15.757862 -0.8603835 -8.579226
4            33818_at      3845 0.03731885 -16.560453 -0.6846256 -7.866767
5             1091_at       102 0.04430109 -19.858982 -0.4745293 -6.819760
6            36311_at      6363 0.04430109 -14.013592 -0.6919303 -7.278708
7              546_at     12106 0.04430109 -10.181562 -0.8456221 -6.568406
8             1024_at        28 0.04430109 -11.241182 -0.6650023 -6.342763
9            40322_at     10414 0.04430109 -12.258781 -0.5484858 -6.065384
10           38734_at      8810 0.04430109 -10.129580 -0.5154920 -5.337663
11           32660_at      2675 0.04430109 -13.974358 -0.3281164 -4.743865
12          1552_i_at       609 0.04430109  -6.499473 -0.6759853 -4.515819
13           36889_at      6946 0.04430109  -8.767056 -0.5016880 -4.874979
14           38254_at      8325 0.04430109   8.293945  0.3783841  4.144027
15           37777_at      7843 0.04430109  -8.696619 -0.2633858 -3.466853
16           39058_at      9137 0.04430109  -7.176070 -0.2977104 -3.415163
17            1032_at        37 0.04430109   4.684491  0.4109406  3.080278
18 AFFX-YEL021w/URA3_at     12625 0.04430109  -2.130391 -0.6669140 -1.859075
19            1708_at       780 0.04430109  -5.415472 -0.3216362 -3.060909
20             407_at     10795 0.04430109  -6.421535 -0.2044849 -2.637487
```

2.3 The plotting functions `pairs.DEDS` and `hist.deds`

We next illustrate the usage of the function `pairs.DEDS`, which is a S3 method for `pairs`. It displays a scatter matrix plot for individual statistics that DEDS synthesizes and highlights the top genes according to a user specified threshold (`thresh`); see also `?pairs.DEDS`. Plots on the diagonal panels are QQ-plots as the default, but can be set as ‘‘`histogram`’’,‘‘`boxplot`’’, ‘‘`density`’’ or ‘‘`none`’’. To display only qq-plots, the function `qqnorm.DEDS` can be use.

```
> pairs(deds.affy, subset = c(2:12626), thresh = 0.01, legend = F)
```

```
> qqnorm(deds.affy, subset = c(2:12626), thresh = 0.01)
```

3 Case study 2: ApoA1 Experiment

The next example we demonstrate is a set of cDNA microarray data from a study of a mouse model with very low HDL cholesterol levels described in Dudoit et al. (2002b). To load the dataset, use `data(ApoA1)`, and to view a description of the experiements and data, type `?ApoA1`.

```
> data(ApoA1)
```

3.1 Data

The goal of the ApoA1 experiment to identify DE genes in apolipoprotein A1 (apo A1) knock-out mice. The treatment group consists of eight knock-out mice and the control group consists of eight normal mice. The dataset includes

- ApoA1: a $6,384 \times 16$ matrix of expression levels;
- ApoA1.L: a vector of class labels (0 for the control group, 1 for the treatment group);

```
> dim(ApoA1)
```

```
[1] 6384   16
```

```
> ApoA1.L
```

```
 [1] 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1
```

3.2 Application of the deds.stat.linkC function

We apply deds.stat.linkC on the ApoA1 dataset using 400 permutations and evaluating the adjusted$p$. Here, as a default, DEDS synthesizes t statistics, FC and SAM.

```
> deds.ApoA1 <- deds.stat.linkC(ApoA1, ApoA1.L, B = 400, adj = "adjp")
```

```
> sum(deds.ApoA1$p <= 0.01)
```

```
[1] 7
```

```
> sum(deds.ApoA1$p <= 0.05)
```

```
[1] 9
```

```
> topgenes(deds.ApoA1, number = 9)
```

```
  geneOrder   DEDS         t         fc       sam
1      2149 0.0025 16.500624  3.2440281  8.674891
2       540 0.0025  8.784189  2.9692008  5.761274
3      5356 0.0025  9.256464  1.7848039  4.821543
4      1739 0.0025  9.789541  0.9976147  3.572326
5      2537 0.0050  7.842692  0.9840616  3.249543
6      4139 0.0050  7.880206  0.9779046  3.243980
7         2 0.0100 -6.656223 -0.8907498 -2.862510
8      4941 0.0150  5.909812  0.9214721  2.764874
9      1204 0.0250 -5.156277 -0.9961325 -2.688297
```

```
> pairs(deds.ApoA1, legend = F)
```

4 Statistical functions

4.1 The comp.t and other related functions

The DEDS package provides the following functions, comp.FC, comp.t, comp.SAM, comp.F, comp.B, comp.modt and comp.modF for the computation of FC, t-statistics, SAM, F statistics, B statistics, moderated t- and F- statistics respectively.

There are two steps in applying the above functions to obtain corresponding statistics:

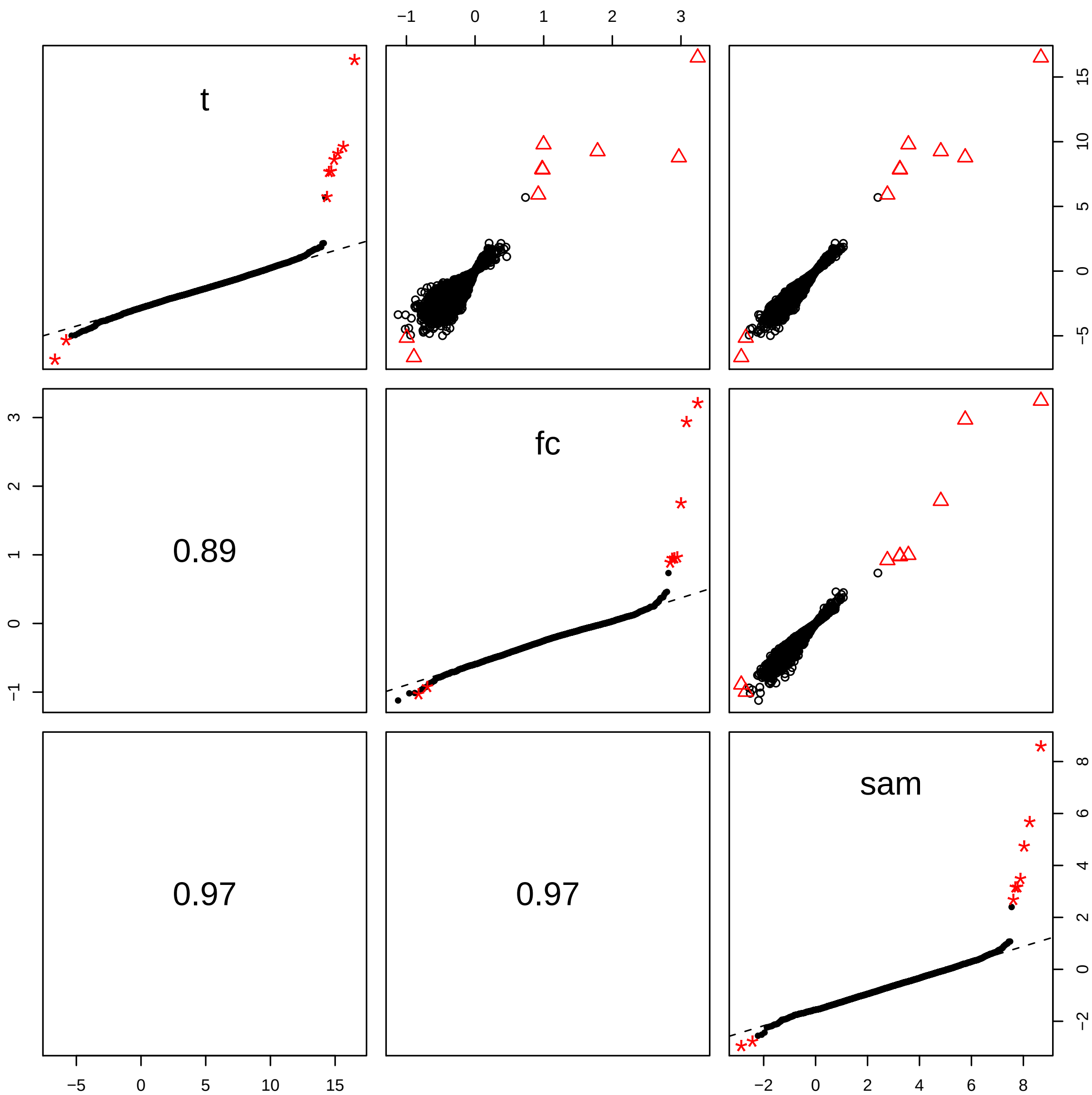

Figure 1: Pairs plots for the ApoA1 data

1. Create the statistic function.

2. Apply the function to the microarray expression matrix.

We illustrate the usage with `comp.t` and other functions follow the same rules. The function `comp.t` has three arguments (see also `?comp.t`):

`L`: A vector of integers corresponding to observation (column) class labels. For k classes, the labels must be integers between 0 and $k-1$.

`mu`: A number indicating the true value of the mean (or difference in means if two sample statistics are calculated; default set at 0.

`var.equal`: a logical variable indicating whether to treat the two variances as being equal.

`comp.t` returns a function of one argument with bindings for `L`, `mu` and `var.equal`. This function accepts a microarray data matrix as its single argment, when evaluated, computes t statistic for each row of the matrix.

```
> t <- comp.t(L = affySpikeIn.L)
> t.affy <- t(affySpikeIn)
```

4.2 The `comp.stat` function

A simple wrapper function `comp.stat` is provided for users interested in applying a standard set of statistical measures using default parameters. The most important arguments for `comp.stat` are elaborated below:

`X`: A matrix, with rows correspond to genes and columns to mRNA samples.

`L`: A vector of integers corresponding to observation (column) class labels. For k classes, the labels must be integers between 0 and $k-1$.

`test`: A character string specifying the statistic to be applied.
‘‘`t`’’ – t statistics;
‘‘`fc`’’ – FC;
‘‘`sam`’’ – SAM statistics;
‘‘`F`’’ – F statistics;
‘‘`modt`’’ – moderated t statistics;
‘‘`modF`’’ – moderated F statistics;
‘‘`B`’’ – B statistics;

To compute t statistics on the `affySpikeIn` data, instead of using `comp.t`, the users can also use `comp.stat` by specifying the `test` as ‘‘`t`’’. However, `comp.stat` computes t statistics assuming unequal variance (if it is a two-sample comparison); if the user desires to use an equal variance option, the function `comp.t` has to be applied instead.

```
> t.affy <- comp.stat(affySpikeIn, affySpikeIn.L, test = "t")
```

4.3 Computing adjusted p values and q values for associated statistics

Two functions `comp.adjp` and `comp.fdr` are provided to calculate adjusted p values (controling FWER) and q values (controling FDR) respectively. The function `comp.adjp` computes permuation based step-down maxT adjusted p values for a selected test statistic, e.g., one- or two-sample t-statitics, F-statistics, SAM, Fold change, moderated t-statistics and moderated F-statisitcs, for each gene. The procedure is based on codes from the function `mt.maxT` in the Bioconductor package `multtest` and described in Westfall and Young (1993). The function `comp.fdr` computes permuation based q values for the same spectrum of statistics. The following codes provide examples showcasing the usage of these two functions.

```
> adjp.affy <- comp.adjp(affySpikeIn, affySpikeIn.L, B = 500, test = "t")

> adjp.affy[1:10, ]

      order         t unadj.p adj.p
 [1,] 12244 171.85851       0     0
 [2,]   102 -19.85898       0     0
 [3,]  6134 -19.25009       0     0
 [4,]  3845 -16.56045       0     0
 [5,]  6253 -15.75786       0     0
 [6,]  6363 -14.01359       0     0
 [7,]  2675 -13.97436       0     0
 [8,] 10414 -12.25878       0     0
 [9,]    28 -11.24118       0     0
[10,] 12106 -10.18156       0     0

> fdr.affy <- comp.fdr(affySpikeIn, affySpikeIn.L, B = 500, test = "t")

> fdr.affy[1:10, ]

      order         t unadj.p qvalues
 [1,] 12244 171.85851       0       0
 [2,]   102 -19.85898       0       0
 [3,]  6134 -19.25009       0       0
 [4,]  3845 -16.56045       0       0
 [5,]  6253 -15.75786       0       0
 [6,]  6363 -14.01359       0       0
 [7,]  2675 -13.97436       0       0
 [8,] 10414 -12.25878       0       0
 [9,]    28 -11.24118       0       0
[10,] 12106 -10.18156       0       0
```

References

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Statist. Soc. B*, 57:289–300.

Dudoit, S., Shaffer, J. P., and Boldrick, J. C. (2002a). Multiple hypothesis testing in microarray experiments. Submitted.

Dudoit, S., Yang, Y. H., Callow, M. J., and Speed, T. P. (2002b). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, 12(1):111–139.

Irizarry, R. A., Bolstad, B. M., Collin, F., Cope, L., Hobbs, B., and Speed, T. P. (2003). Summaries of affymetrix genechip probe level data. *Nucleic Acids Research*, 31:e15.

Lönnstedt, I. and Speed, T. P. (2001). Replicated microarray data. *Statistica Sinica*, 12(1):31–46.

Smyth, G., Ritchie, M., Wettenhall, J., and Thorne, N. (2003). Linear models for microarray data. `R` package, `http://lib.stat.cmu.edu/R/CRAN/`.

Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to transcriptional responses to ionizing radiation. *Proc. Natl. Acad. Sci.*, 98:5116–5121.

Yang, Y. H., Xiao, Y., and Segal, M. (2004). Indentifying differentially expressed genes from micorray experiments via statistic synthesis. *Manuscript in preparation.*

Westfall, P. H. and Young, S. S. (1993). *Resampling-based multiple testing: Examples and methods for p-value adjustment.* John Wiley & Sons.