Introduction to RBM package

Dongmei Li

April 30, 2018

Clinical and Translational Science Institute, University of Rochester School of Medicine and Dentistry, Rochester, NY 14642-0708

Contents

1 Overview

This document provides an introduction to the `RBM` package. The `RBM` package executes the resampling-based empirical Bayes approach using either permutation or bootstrap tests based on moderated t-statistics through the following steps.

- Firstly, the RBM package computes the moderated t-statistics based on the observed data set for each feature using the lmFit and eBayes function.
- Secondly, the original data are permuted or bootstrapped in a way that matches the null hypothesis to generate permuted or bootstrapped resamples, and the reference distribution is constructed using the resampled moderated t-statistics calculated from permutation or bootstrap resamples.
- Finally, the p-values from permutation or bootstrap tests are calculated based on the proportion of the permuted or bootstrapped moderated t-statistics that are as extreme as, or more extreme than, the observed moderated t-statistics.

Additional detailed information regarding resampling-based empirical Bayes approach can be found elsewhere (Li et al., 2013).

2 Getting started

The RBM package can be installed and loaded through the following R code.
Install the RBM package with:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("RBM")
```

Load the RBM package with:

```
> library(RBM)
```

3 RBM_T and RBM_F functions

There are two functions in the RBM package: RBM_T and RBM_F. Both functions require input data in the matrix format with rows denoting features and columns denoting samples. RBM_T is used for two-group comparisons such as study designs with a treatment group and a control group. RBM_F can be used for more complex study designs such as more than two groups or time-course studies. Both functions need a vector for group notation, i.e., "1" denotes the treatment group and "0" denotes the control group. For the RBM_F function, a contrast vector need to be provided by users to perform pairwise comparisons between groups. For example, if the design has three groups (0, 1, 2), the aContrast parameter will be a vector such as ("X1-X0", "X2-X1", "X2-X0") to denote all pairwise comparisons. Users just need to add an extra "X" before the group labels to do the contrasts.

- Examples using the RBM_T function: normdata simulates a standardized gene expression data and unifdata simulates a methylation microarray data. The p-values from the RBM_T function could be further adjusted using the p.adjust function in the stats package through the Bejamini-Hochberg method.

```
> library(RBM)
> normdata <- matrix(rnorm(1000*6, 0, 1),1000,6)
> mydesign <- c(0,0,0,1,1,1)
> myresult <- RBM_T(normdata,mydesign,100,0.05)
> summary(myresult)

              Length Class  Mode
ordfit_t      1000   -none- numeric
ordfit_pvalue 1000   -none- numeric
ordfit_beta0  1000   -none- numeric
ordfit_beta1  1000   -none- numeric
permutation_p 1000   -none- numeric
bootstrap_p   1000   -none- numeric

> sum(myresult$permutation_p<=0.05)

[1] 45
```

```
> which(myresult$permutation_p<=0.05)

 [1]  26  49  54  57  65  85  86  90 100 116 144 148 157 198 275 296 297 309 336
[20] 380 461 500 504 521 583 599 625 630 632 665 675 684 695 708 711 722 749 820
[39] 823 831 855 875 930 941 975

> sum(myresult$bootstrap_p<=0.05)

[1] 19

> which(myresult$bootstrap_p<=0.05)

 [1]  83 125 148 251 326 387 440 442 454 468 617 632 652 671 787 821 891 958 960

> permutation_adjp <- p.adjust(myresult$permutation_p, "BH")
> sum(permutation_adjp<=0.05)

[1] 3

> bootstrap_adjp <- p.adjust(myresult$bootstrap_p, "BH")
> sum(bootstrap_adjp<=0.05)

[1] 0

> unifdata <- matrix(runif(1000*7,0.10, 0.95), 1000, 7)
> mydesign2 <- c(0,0,0, 1,1,1,1)
> myresult2 <- RBM_T(unifdata,mydesign2,100,0.05)
> sum(myresult2$permutatioin_p<=0.05)

[1] 0

> sum(myresult2$bootstrap_p<=0.05)

[1] 18

> which(myresult2$bootstrap_p<=0.05)

 [1]    3   84  133  207  230  342  381  409  511  541  564  571  610  649  670
[16]  871  926 1000

> bootstrap2_adjp <- p.adjust(myresult2$bootstrap_p, "BH")
> sum(bootstrap2_adjp<=0.05)

[1] 0
```

- Examples using the `RBM_F` function: normdata_F simulates a standardized gene expression data and unifdata_F simulates a methylation microarray data. In both examples, we were interested in pairwise comparisons.

```
> normdata_F <- matrix(rnorm(1000*9,0,2), 1000, 9)
> mydesign_F <- c(0, 0, 0, 1, 1, 1, 2, 2, 2)
> aContrast <- c("X1-X0", "X2-X1", "X2-X0")
> myresult_F <- RBM_F(normdata_F, mydesign_F, aContrast, 100, 0.05)
> summary(myresult_F)

             Length Class  Mode
ordfit_t     3000   -none- numeric
ordfit_pvalue 3000   -none- numeric
ordfit_beta1  3000   -none- numeric
permutation_p 3000   -none- numeric
bootstrap_p   3000   -none- numeric

> sum(myresult_F$permutation_p[, 1]<=0.05)

[1] 68

> sum(myresult_F$permutation_p[, 2]<=0.05)

[1] 68

> sum(myresult_F$permutation_p[, 3]<=0.05)

[1] 81

> which(myresult_F$permutation_p[, 1]<=0.05)

 [1]   1  19  25  32  41  49  80  84 135 167 169 181 194 218 232 235 293 303 322
[20] 330 346 353 377 381 382 383 416 460 466 472 476 478 482 501 509 510 534 551
[39] 567 568 583 599 609 621 622 627 628 655 656 698 707 739 750 761 769 779 781
[58] 800 809 819 828 835 844 878 893 894 916 985

> which(myresult_F$permutation_p[, 2]<=0.05)

 [1]   1  32  37  41  84  86  97 135 169 178 181 194 218 232 235 268 293 303 322
[20] 330 346 353 369 377 381 416 460 466 472 476 478 482 501 509 510 515 534 551
[39] 567 568 583 599 609 621 622 628 636 655 656 698 707 711 739 750 761 779 781
[58] 800 819 828 844 878 884 893 894 916 950 965

> which(myresult_F$permutation_p[, 3]<=0.05)

 [1]   1  25  32  41  46  84  86 102 116 121 135 161 169 178 181 194 212 218 232
[20] 235 241 268 293 303 322 330 346 353 372 377 380 381 382 416 460 466 472 476
[39] 478 482 501 509 510 534 567 568 583 599 609 621 622 628 636 655 656 675 681
[58] 698 707 711 735 739 750 761 769 779 781 800 819 828 835 844 878 884 893 894
[77] 913 916 950 965 985
```

```
> con1_adjp <- p.adjust(myresult_F$permutation_p[, 1], "BH")
> sum(con1_adjp<=0.05/3)

[1] 0

> con2_adjp <- p.adjust(myresult_F$permutation_p[, 2], "BH")
> sum(con2_adjp<=0.05/3)

[1] 9

> con3_adjp <- p.adjust(myresult_F$permutation_p[, 3], "BH")
> sum(con3_adjp<=0.05/3)

[1] 9

> which(con2_adjp<=0.05/3)

[1]   1 235 353 377 501 534 628 655 750

> which(con3_adjp<=0.05/3)

[1]   1 235 322 353 377 501 621 750 894

> unifdata_F <- matrix(runif(1000*18, 0.15, 0.98), 1000, 18)
> mydesign2_F <- c(rep(0, 6), rep(1, 6), rep(2, 6))
> aContrast <- c("X1-X0", "X2-X1", "X2-X0")
> myresult2_F <- RBM_F(unifdata_F, mydesign2_F, aContrast, 100, 0.05)
> summary(myresult2_F)

             Length Class  Mode
ordfit_t     3000   -none- numeric
ordfit_pvalue 3000   -none- numeric
ordfit_beta1  3000   -none- numeric
permutation_p 3000   -none- numeric
bootstrap_p   3000   -none- numeric

> sum(myresult2_F$bootstrap_p[, 1]<=0.05)

[1] 56

> sum(myresult2_F$bootstrap_p[, 2]<=0.05)

[1] 59

> sum(myresult2_F$bootstrap_p[, 3]<=0.05)

[1] 57
```

```
> which(myresult2_F$bootstrap_p[, 1]<=0.05)

 [1]   5  11  26  56  58  94 101 128 130 135 138 142 146 206 209 211 251 257 276
[20] 277 280 289 296 333 347 370 382 401 447 457 514 531 538 616 630 633 635 680
[39] 683 693 718 754 759 770 789 810 829 845 871 889 905 921 934 943 948 992

> which(myresult2_F$bootstrap_p[, 2]<=0.05)

 [1]   5  10  11  26  56  58  86  94  96 101 128 138 142 146 149 194 209 247 257
[20] 289 333 347 370 382 401 413 419 447 457 474 481 495 514 531 538 544 616 633
[39] 635 657 680 693 718 753 754 759 770 803 810 829 845 871 889 905 921 934 948
[58] 951 992

> which(myresult2_F$bootstrap_p[, 3]<=0.05)

 [1]  11  26  56  58  86  94 101 128 130 138 142 146 149 206 251 257 274 289 347
[20] 370 382 401 413 419 447 457 481 498 514 531 538 544 616 633 635 657 680 683
[39] 718 724 741 754 759 770 789 810 829 845 871 889 905 921 934 943 948 958 992

> con21_adjp <- p.adjust(myresult2_F$bootstrap_p[, 1], "BH")
> sum(con21_adjp<=0.05/3)

[1] 10

> con22_adjp <- p.adjust(myresult2_F$bootstrap_p[, 2], "BH")
> sum(con22_adjp<=0.05/3)

[1] 9

> con23_adjp <- p.adjust(myresult2_F$bootstrap_p[, 3], "BH")
> sum(con23_adjp<=0.05/3)

[1] 5
```

4 Ovarian cancer methylation example using the RBM_T function

Two-group comparisons are the most common contrast in biological and biomedical field. The ovarian cancer methylation example is used to illustrate the application of RBM_T in identifying differentially methylated loci. The ovarian cancer methylation example is taken from the gemone-wide DNA methylation profiling of United Kingdom Ovarian Cancer Population Study (UKOPS). This study used Illumina Infinium 27k Human DNA methylation Beadchip v1.2 to obtain DNA methylation profiles on over 27,000 CpGs in whole blood cells from 266 ovarian cancer women and 274 age-matched healthy controls. The data are downloaded from the NCBI GEO website with access number GSE19711. For illutration purpose, we chose the first 1000 loci in 8 randomly selected women with 4 ovariance cancer cases (pre-treatment) and 4 healthy controls. The following codes show the process of generating significant differential DNA methylation loci using the RBM_T function and presenting the results for further validation and investigations.

```
> system.file("data", package = "RBM")

[1] "C:/Users/biocbuild/bbs-3.7-bioc/tmpdir/Rtmp0Ouy7N/Rinst1304404451f0/RBM/data"

> data(ovarian_cancer_methylation)
> summary(ovarian_cancer_methylation)

        IlmnID         Beta          exmdata2[, 2]     exmdata3[, 2]
 cg00000292:  1   Min.   :0.01058   Min.   :0.01187   Min.   :0.009103
 cg00002426:  1   1st Qu.:0.04111   1st Qu.:0.04407   1st Qu.:0.041543
 cg00003994:  1   Median :0.08284   Median :0.09531   Median :0.087042
 cg00005847:  1   Mean   :0.27397   Mean   :0.28872   Mean   :0.283729
 cg00006414:  1   3rd Qu.:0.52135   3rd Qu.:0.59032   3rd Qu.:0.558575
 cg00007981:  1   Max.   :0.97069   Max.   :0.96937   Max.   :0.970155
 (Other)   :994                     NA's   :4
 exmdata4[, 2]     exmdata5[, 2]     exmdata6[, 2]     exmdata7[, 2]
 Min.   :0.01019   Min.   :0.01108   Min.   :0.01937   Min.   :0.01278
 1st Qu.:0.04092   1st Qu.:0.04059   1st Qu.:0.05060   1st Qu.:0.04260
 Median :0.09042   Median :0.08527   Median :0.09502   Median :0.09362
 Mean   :0.28508   Mean   :0.28482   Mean   :0.27348   Mean   :0.27563
 3rd Qu.:0.57502   3rd Qu.:0.57300   3rd Qu.:0.52099   3rd Qu.:0.52240
 Max.   :0.96658   Max.   :0.97516   Max.   :0.96681   Max.   :0.95974
                   NA's   :1
 exmdata8[, 2]
 Min.   :0.01357
 1st Qu.:0.04387
 Median :0.09282
 Mean   :0.28679
 3rd Qu.:0.57217
 Max.   :0.96268

> ovarian_cancer_data <- ovarian_cancer_methylation[, -1]
> label <- c(1, 1, 0, 0, 1, 1, 0, 0)
> diff_results <- RBM_T(aData=ovarian_cancer_data, vec_trt=label, repetition=100, alpha=0.05)
> summary(diff_results)

             Length Class  Mode
ordfit_t     1000   -none- numeric
ordfit_pvalue 1000   -none- numeric
ordfit_beta0  1000   -none- numeric
ordfit_beta1  1000   -none- numeric
permutation_p 1000   -none- numeric
bootstrap_p   1000   -none- numeric

> sum(diff_results$ordfit_pvalue<=0.05)

[1] 45
```

```
> sum(diff_results$permutation_p<=0.05)

[1] 39

> sum(diff_results$bootstrap_p<=0.05)

[1] NA

> ordfit_adjp <- p.adjust(diff_results$ordfit_pvalue, "BH")
> sum(ordfit_adjp<=0.05)

[1] 0

> perm_adjp <- p.adjust(diff_results$permutation_p, "BH")
> sum(perm_adjp<=0.05)

[1] 1

> boot_adjp <- p.adjust(diff_results$bootstrap_p, "BH")
> sum(boot_adjp<=0.05)

[1] NA

> diff_list_perm <- which(perm_adjp<=0.05)
> diff_list_boot <- which(boot_adjp<=0.05)
> sig_results_perm <- cbind(ovarian_cancer_methylation[diff_list_perm, ], diff_results$ordfit_t
> print(sig_results_perm)

        IlmnID       Beta exmdata2[, 2] exmdata3[, 2] exmdata4[, 2]
450 cg00432979 0.03681359      0.045157    0.04374394    0.03683598
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
450    0.04419125    0.04409653    0.02839263     0.0341002
    diff_results$ordfit_t[diff_list_perm]
450                              1.546114
    diff_results$permutation_p[diff_list_perm]
450                                      0

> sig_results_boot <- cbind(ovarian_cancer_methylation[diff_list_boot, ], diff_results$ordfit_t
> print(sig_results_boot)

        IlmnID       Beta exmdata2[, 2] exmdata3[, 2] exmdata4[, 2]
83  cg00072216 0.04505377    0.04598964    0.04000674    0.03231534
95  cg00081975 0.03633894    0.04975194    0.06024723    0.05598723
146 cg00134539 0.61101320    0.53321780    0.45999340    0.46787420
200 cg00183916 0.03525946    0.03984548    0.02765822    0.02789838
280 cg00260778 0.64319890    0.60488960    0.56735060    0.53150910
397 cg00394658 0.27940900    0.40410330    0.40262320    0.44339290
517 cg00499822 0.09723835    0.13925420    0.12969170    0.15998260
```

```
520 cg00502442 0.03163993    0.03581662    0.02785063    0.02549502
743 cg00717862 0.07999436    0.07873347    0.06089359    0.06171374
804 cg00777121 0.04540701    0.05430304    0.04154242    0.04221162
833 cg00814580 0.09348613    0.09619816    0.12010440    0.11534240
979 cg00945507 0.13432250    0.23854600    0.34749760    0.28903340
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
83     0.04965089    0.04833366    0.03466159    0.04390894
95     0.04561792    0.05115624    0.06068253    0.06168212
146    0.67191510    0.63137380    0.47929610    0.45428300
200    0.03034811    0.04302129    0.02753873    0.03067437
280    0.61920530    0.61925200    0.46753250    0.55632410
397    0.35626060    0.23388380    0.41974630    0.45806880
517    0.11009170    0.08752679    0.15305730    0.21607890
520    0.03111720    0.03189393    0.02415307    0.02941176
743    0.07594936    0.09062161    0.06475791    0.07271878
804    0.04911277    0.04872797    0.04261405    0.04474881
833    0.09577040    0.11598850    0.12860890    0.14111200
979    0.11848510    0.16653850    0.30718420    0.26624740
    diff_results$ordfit_t[diff_list_boot]
83                               2.514109
95                              -3.252063
146                              5.394750
200                              2.272449
280                              4.170347
397                             -3.070559
517                             -2.837883
520                              1.873471
743                              3.444684
804                              1.995220
833                             -3.428319
979                             -4.750997
    diff_results$bootstrap_p[diff_list_boot]
83                                        0
95                                        0
146                                       0
200                                       0
280                                       0
397                                       0
517                                       0
520                                       0
743                                       0
804                                       0
833                                       0
979                                       0
```