Introduction to RBM package

Dongmei Li

October 13, 2015

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] 54
```

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

 [1]  11  17  30  36  48  50  63  93 106 126 158 167 178 181 183 207 227 234 267
[20] 295 303 318 319 325 329 334 363 416 418 436 454 470 509 524 529 534 607 614
[39] 633 641 653 689 706 719 765 783 797 824 836 841 843 857 908 912

> sum(myresult$bootstrap_p<=0.05)

[1] 7

> which(myresult$bootstrap_p<=0.05)

[1]  66 191 248 385 463 466 691

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

[1] 9

> 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] 25

> which(myresult2$bootstrap_p<=0.05)

 [1]  30  39  48  81 115 134 155 161 304 318 324 370 463 468 475 486 543 580 595
[20] 615 747 773 824 929 942

> 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] 55
```

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

```
[1] 55
```

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

```
[1] 53
```

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

```
 [1]  36  37  65 101 129 139 150 240 270 288 291 294 299 376 389 391 410 414 424
[20] 434 435 480 498 517 531 535 545 555 562 606 608 637 673 675 704 706 719 752
[39] 769 774 787 809 820 849 864 913 917 919 939 940 941 965 968 969 973
```

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

```
 [1]  34  36  37  42  65  73 101 129 139 144 240 255 270 275 288 291 295 299 317
[20] 375 376 387 410 424 434 435 493 498 517 531 535 555 606 608 618 637 675 704
[39] 706 719 752 769 774 804 809 820 849 864 913 917 919 939 941 965 973
```

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

```
 [1]  37  65  73 101 139 188 240 255 270 288 291 295 299 356 379 389 391 410 424
[20] 434 435 493 517 531 535 555 562 606 608 619 673 675 704 706 719 752 769 774
[39] 787 796 804 809 820 864 874 913 917 919 939 941 965 969 973
```

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

```
[1] 16
```

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

[1] 14

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

[1] 10

> which(con2_adjp<=0.05/3)

 [1]  37 291 299 435 517 608 637 706 719 809 820 864 917 919

> which(con3_adjp<=0.05/3)

 [1]  73 291 299 424 434 517 608 719 809 913

> 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] 40

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

[1] 52

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

[1] 36

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

 [1]   2  10  14  18  41  95 130 131 152 231 254 271 281 314 386 426 434 444 460
[20] 484 489 492 498 594 598 602 685 768 804 813 901 943 959 963 969 973 974 984
[39] 990 994
```

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

 [1]   2  10  14  18  35  41  56  78  95  97 130 131 152 194 200 231 271 311 312
[20] 314 386 388 434 444 460 484 489 498 509 549 556 575 594 598 641 656 738 768
[39] 813 839 840 872 901 928 950 959 963 969 973 974 984 990

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

 [1]   2  10  14  31  35  41  95 131 152 194 200 231 279 301 386 444 460 484 489
[20] 492 498 594 598 641 768 795 813 840 901 959 963 969 973 974 984 990

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

[1] 3

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

[1] 1

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

[1] 0
```

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 gemonewide 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:/biocbld/bbs-3.2-bioc/tmpdir/Rtmpch0Abf/Rinst26646cca5ca7/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] 69
```

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

```
[1] 53

> 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] 5

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

[1] 5

> 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_
> print(sig_results_perm)

        IlmnID       Beta exmdata2[, 2] exmdata3[, 2] exmdata4[, 2]
83  cg00072216 0.04505377    0.04598964    0.04000674    0.03231534
103 cg00094319 0.73784280    0.73532960    0.75574900    0.73830220
106 cg00095674 0.07076291    0.05045181    0.03861991    0.03337576
848 cg00826384 0.05721674    0.05612171    0.06644259    0.06358381
851 cg00830029 0.58362500    0.59397870    0.64739610    0.67269640
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
83     0.04965089    0.04833366    0.03466159    0.04390894
103    0.67349260    0.73510200    0.75715920    0.78981220
106    0.04693030    0.06837343    0.04534005    0.03709488
848    0.05230160    0.06119713    0.06542751    0.06240686
851    0.50820240    0.34657470    0.66276570    0.64634510
    diff_results$ordfit_t[diff_list_perm]
83                                 2.514109
103                               -2.268711
106                                3.100324
848                               -2.314412
851                               -2.841244
    diff_results$permutation_p[diff_list_perm]
83                                             0
103                                            0
106                                            0
848                                            0
851                                            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]
95  cg00081975 0.03633894    0.04975194    0.06024723    0.05598723
106 cg00095674 0.07076291    0.05045181    0.03861991    0.03337576
259 cg00234961 0.04192170    0.04321576    0.05707140    0.05327565
911 cg00888479 0.07388961    0.07361080    0.10149800    0.09985076
928 cg00901493 0.03737166    0.03903724    0.04684618    0.04981432
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
95     0.04561792    0.05115624    0.06068253    0.06168212
106    0.04693030    0.06837343    0.04534005    0.03709488
259    0.04030003    0.03996053    0.05086962    0.05445672
911    0.08633986    0.06765189    0.09070268    0.12417730
928    0.04490690    0.04204062    0.05050039    0.05268215
    diff_results$ordfit_t[diff_list_boot]
95                              -3.252063
106                              3.100324
259                             -4.052697
911                             -3.621731
928                             -2.716443
    diff_results$bootstrap_p[diff_list_boot]
95                                         0
106                                        0
259                                        0
911                                        0
928                                        0
```