Introduction to RBM package

Dongmei Li

May 2, 2019

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:

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
> BiocManager::install("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] 37

> which(myresult$permutation_p<=0.05)

 [1]  32 116 121 188 191 205 223 243 262 282 287 293 306 353 376 383 396 423 450
[20] 489 509 525 550 558 570 629 635 654 669 712 837 860 879 914 951 976 997

> sum(myresult$bootstrap_p<=0.05)

[1] 12

> which(myresult$bootstrap_p<=0.05)

 [1] 223 236 287 295 376 469 581 627 657 818 837 976

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

[1] 0

> 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] 4

> which(myresult2$bootstrap_p<=0.05)

[1]  37 152 374 477

> 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] 51

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

[1] 59

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

[1] 62

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

 [1]   7  20  29  59 107 121 122 142 146 165 180 184 186 190 215 221 245 271 340
[20] 352 360 386 392 445 478 480 511 514 529 532 565 569 616 635 667 668 703 725
[39] 739 755 765 771 783 804 898 902 910 947 966 976 982

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

 [1]   7  16  20  29  59  75  95 107 116 121 122 142 146 158 165 180 184 186 190
[20] 215 221 226 227 245 248 255 271 296 340 352 360 440 445 478 480 511 514 529
[39] 532 537 549 565 569 572 667 668 703 725 739 765 771 783 804 893 898 902 947
[58] 976 982

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

 [1]   7  16  29  54  59 107 113 121 122 142 146 165 180 184 186 190 215 221 226
[20] 227 245 248 255 296 316 340 352 360 389 440 445 478 480 511 514 529 532 537
[39] 549 558 562 565 569 589 616 620 635 668 697 703 725 739 765 771 783 804 898
[58] 902 910 947 976 982

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

```
[1] 7

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

[1] 15

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

[1] 13

> which(con2_adjp<=0.05/3)

 [1]  29 107 121 142 184 215 340 360 511 529 725 739 771 783 902

> which(con3_adjp<=0.05/3)

 [1]  29 142 184 215 340 360 668 703 725 771 783 804 898

> 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] 52

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

[1] 49

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

[1] 53

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

```
 [1]  30  32  39  51  98 111 120 155 174 175 209 250 251 261 295 322 326 340 350
[20] 353 365 379 380 390 391 392 403 412 434 458 479 499 536 577 583 603 661 667
[39] 768 773 803 804 883 889 896 897 906 908 945 946 954 976

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

 [1]  30  32  39  56  82  98 120 155 174 175 209 250 251 261 267 295 322 335 350
[20] 353 365 379 380 390 391 392 397 403 412 434 479 499 655 661 667 773 780 803
[39] 827 847 889 896 897 906 908 945 954 961 976

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

 [1]  30  32  39  51  60  82  98 120 158 174 175 199 209 250 251 261 322 337 350
[20] 353 365 379 380 390 391 392 403 412 433 434 479 499 577 583 587 606 619 667
[39] 748 773 780 803 847 865 889 892 896 897 906 945 946 954 976

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

[1] 6

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

[1] 8

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

[1] 11
```

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-3.9-bioc/tmpdir/RtmpyuvsFZ/Rinst24843ecd317/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] 56

> sum(diff_results$bootstrap_p<=0.05)

[1] 60

> 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] 1

> 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] exmdata5[, 2]
851 cg00830029 0.583625     0.5939787     0.6473961     0.6726964     0.5082024
    exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
851     0.3465747     0.6627657     0.6463451
    diff_results$ordfit_t[diff_list_perm]
851                             -2.841244
    diff_results$permutation_p[diff_list_perm]
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]
252 cg00230502 0.1006139     0.1351787     0.1253851     0.1630492
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
252     0.1197087     0.1203616     0.1742373     0.1815548
    diff_results$ordfit_t[diff_list_boot]
252                             -3.144056
    diff_results$bootstrap_p[diff_list_boot]
252                                       0
```