Introduction to RBM package

Dongmei Li

October 27, 2020

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] 19

> which(myresult$permutation_p<=0.05)

 [1]   5 119 140 201 217 253 267 282 338 417 456 607 617 694 735 788 793 817 986

> sum(myresult$bootstrap_p<=0.05)

[1] 6

> which(myresult$bootstrap_p<=0.05)

[1] 217 438 507 522 615 986

> 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] 33

> which(myresult2$bootstrap_p<=0.05)

 [1]  47  53  59  70 146 161 200 345 381 421 472 475 508 521 538 544 572 643 671
[20] 687 688 692 705 715 724 739 750 804 828 832 866 939 976

> 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] 58

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

[1] 64

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

[1] 50

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

 [1]   1   3  34  79  96 100 103 104 156 158 185 258 297 300 320 324 337 359 360
[20] 361 362 381 383 409 457 481 492 529 549 581 619 623 629 630 641 649 666 707
[39] 709 726 740 752 758 768 769 781 794 815 825 833 844 899 940 950 954 980 981
[58] 996

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

 [1]   1   2  34  39  73  77  79  96 100 103 104 156 158 185 238 258 281 297 300
[20] 320 324 337 359 361 365 380 381 383 386 409 481 492 529 533 536 549 579 594
[39] 619 630 641 647 666 709 726 732 740 752 768 769 794 815 825 844 877 899 906
[58] 926 940 950 954 980 981 996

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

 [1]   1   2  21  77 100 103 156 185 258 297 300 324 359 360 361 365 383 386 409
[20] 481 492 529 549 567 579 619 630 641 649 666 709 726 740 752 758 768 769 781
[39] 794 815 826 844 899 906 940 950 954 980 981 996

> 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] 2
```

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

```
[1] 7
```

```
> which(con2_adjp<=0.05/3)
```

```
[1] 386 899
```

```
> which(con3_adjp<=0.05/3)
```

```
[1]   1 258 549 752 899 950 954
```

```
> 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] 69
```

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

```
[1] 61
```

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

```
[1] 67
```

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

```
 [1]  13  14  21  48  56  65  67  94  97 102 103 130 176 202 223 227 228 238 252
[20] 256 258 262 276 285 299 301 313 317 320 324 336 348 353 357 386 418 423 459
[39] 466 470 500 516 518 521 527 530 538 541 543 546 548 550 567 571 580 597 637
[58] 661 677 712 713 737 825 831 886 920 927 957 999
```

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

```
 [1]   2  12  14  21  48  65  67  94 103 176 185 209 211 227 238 252 258 276 285
[20] 299 313 317 320 324 336 338 353 357 386 466 500 514 516 518 521 527 530 534
[39] 541 543 546 567 570 571 580 597 619 637 661 677 685 693 712 721 737 810 831
[58] 870 922 957 999
```

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

```
 [1]   2  13  21  48  56  59  65  67  94  97 103 176 185 211 227 238 252 269 276
[20] 280 285 299 301 313 317 320 324 336 348 353 357 386 418 426 459 466 500 516
[39] 518 521 527 530 534 541 546 550 567 580 597 619 637 661 677 685 693 712 713
[58] 721 724 737 810 831 922 923 942 957 999
```

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

```
[1] 12
```

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

```
[1] 7
```

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

```
[1] 9
```

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] "/tmp/RtmpQleqYl/Rinst23285ec8db65/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] 49

> sum(diff_results$bootstrap_p<=0.05)

[1] 66

> 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] 2

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

[1] 6

> 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]
245 cg00224508 0.04479948    0.04972043    0.04152814    0.04189373
764 cg00730260 0.90471270    0.90542290    0.91002680    0.91258610
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
245    0.04208405    0.05284988    0.03775905    0.03955271
764    0.90575890    0.88760470    0.90756300    0.90946790
    diff_results$ordfit_t[diff_list_perm]
245                              1.962457
764                             -1.808081
    diff_results$permutation_p[diff_list_perm]
245                                          0
764                                          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]
146 cg00134539 0.61101320    0.53321780     0.4599934    0.46787420
259 cg00234961 0.04192170    0.04321576     0.0570714    0.05327565
280 cg00260778 0.64319890    0.60488960     0.5673506    0.53150910
```

```
632 cg00615377 0.11265030    0.16140570     0.1940445    0.17468600
677 cg00651216 0.06825629    0.12529090     0.1440919    0.13907250
979 cg00945507 0.13432250    0.23854600     0.3474976    0.28903340
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
146    0.67191510    0.63137380    0.47929610    0.45428300
259    0.04030003    0.03996053    0.05086962    0.05445672
280    0.61920530    0.61925200    0.46753250    0.55632410
632    0.12573100    0.14483660    0.16338240    0.20130510
677    0.07669587    0.09597587    0.11690440    0.15194540
979    0.11848510    0.16653850    0.30718420    0.26624740
    diff_results$ordfit_t[diff_list_boot]
146                              5.394750
259                             -4.052697
280                              4.170347
632                             -3.661161
677                             -3.387628
979                             -4.750997
    diff_results$bootstrap_p[diff_list_boot]
146                                        0
259                                        0
280                                        0
632                                        0
677                                        0
979                                        0
```