Introduction to RBM package

Dongmei Li

May 19, 2021

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] 112

> which(myresult$permutation_p<=0.05)

  [1]   5  12  14  20  31  32  35  41  55  87  94 129 139 141 147 156 160 163
 [19] 166 167 197 204 213 220 222 225 235 238 246 253 266 283 295 301 303 305
 [37] 312 319 321 328 342 354 365 370 388 399 402 408 411 419 433 437 444 446
 [55] 447 479 480 484 485 489 514 524 540 550 562 573 580 584 606 613 622 635
 [73] 650 666 668 672 674 685 699 703 709 713 722 726 729 734 749 758 763 766
 [91] 771 775 779 815 821 860 862 865 866 890 892 907 918 919 941 963 964 973
[109] 980 983 985 990

> sum(myresult$bootstrap_p<=0.05)

[1] 16

> which(myresult$bootstrap_p<=0.05)

 [1]  55  94 163 166 225 246 301 328 342 584 685 729 754 862 941 964

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

[1] 11

> 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] 23

> which(myresult2$bootstrap_p<=0.05)

 [1]   5  64  83  91 136 152 180 240 247 266 324 378 404 491 552 626 628 760 865
[20] 874 888 935 968

> 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] 61

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

[1] 46

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

[1] 45

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

 [1]  12  13  36  63  83  90 106 109 115 128 163 173 181 198 224 236 238 254 265
[20] 272 299 318 332 341 376 406 427 436 442 449 460 473 509 520 530 558 577 601
[39] 619 663 674 685 698 699 706 740 746 749 756 769 816 822 848 855 871 873 931
[58] 964 981 990 993

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

 [1]   6  13  63  83  90 106 109 115 128 163 181 198 224 236 250 254 265 272 299
[20] 332 344 376 390 406 442 460 473 509 558 601 619 663 685 698 699 706 729 740
[39] 746 756 769 816 855 871 873 990

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

 [1]  63  83  90 106 128 181 198 212 224 236 250 272 273 299 332 376 390 436 442
[20] 460 473 509 520 601 619 663 685 698 699 706 729 740 746 749 756 767 769 816
[39] 855 859 871 873 931 981 990
```

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

[1] 12

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

[1] 11

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

[1] 8

> which(con2_adjp<=0.05/3)

 [1]  83 106 128 272 299 663 706 740 816 873 990

> which(con3_adjp<=0.05/3)

[1] 128 272 299 663 706 740 873 990

> 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] 60

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

[1] 49

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

[1] 51
```

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

 [1]   1  54  73 106 107 111 112 123 124 164 165 175 183 184 229 233 237 293 301
[20] 315 319 327 366 397 453 459 473 476 500 514 528 532 542 602 604 611 634 638
[39] 677 705 728 744 760 768 773 818 819 842 863 874 879 883 910 926 934 948 968
[58] 978 981 986

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

 [1]  54 105 106 111 112 124 155 164 165 183 184 293 301 319 327 366 453 459 473
[20] 476 510 514 526 532 542 570 602 611 615 634 705 728 744 760 768 773 818 819
[39] 842 874 879 883 905 910 926 948 968 978 981

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

 [1]  54  73  78 106 111 112 121 124 164 165 183 184 233 237 293 301 315 319 327
[20] 366 397 453 473 500 528 532 542 611 630 634 638 659 705 744 760 768 773 818
[39] 819 842 863 869 874 879 883 910 926 948 968 978 981

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

[1] 11

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

[1] 4

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

[1] 7
```

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/RtmpKMks3I/Rinstd358e3a1c848d/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] 59

> sum(diff_results$bootstrap_p<=0.05)

[1] 73

> 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] 9

> 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]
83  cg00072216 0.04505377    0.04598964    0.04000674    0.03231534
848 cg00826384 0.05721674    0.05612171    0.06644259    0.06358381
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
83     0.04965089    0.04833366    0.03466159    0.04390894
848    0.05230160    0.06119713    0.06542751    0.06240686
    diff_results$ordfit_t[diff_list_perm]
83                               2.514109
848                             -2.314412
    diff_results$permutation_p[diff_list_perm]
83                                       0
848                                      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    0.45999340    0.46787420
252 cg00230502 0.10061390    0.13517870    0.12538510    0.16304920
258 cg00234616 0.06639040    0.14705640    0.18254770    0.19942150
```

```
259 cg00234961 0.04192170    0.04321576    0.05707140    0.05327565
280 cg00260778 0.64319890    0.60488960    0.56735060    0.53150910
346 cg00331237 0.05972383            NA    0.08204769    0.08345662
482 cg00468146 0.11144740    0.15416650    0.19827990    0.18517240
632 cg00615377 0.11265030    0.16140570    0.19404450    0.17468600
979 cg00945507 0.13432250    0.23854600    0.34749760    0.28903340
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
146    0.67191510    0.63137380    0.47929610    0.45428300
252    0.11970870    0.12036160    0.17423730    0.18155480
258    0.10620550    0.11668540    0.12630260    0.19163650
259    0.04030003    0.03996053    0.05086962    0.05445672
280    0.61920530    0.61925200    0.46753250    0.55632410
346    0.05372019    0.06241126    0.06955040    0.09140985
482    0.12285820    0.13271110    0.14196260    0.22159420
632    0.12573100    0.14483660    0.16338240    0.20130510
979    0.11848510    0.16653850    0.30718420    0.26624740
    diff_results$ordfit_t[diff_list_boot]
146                              5.394750
252                             -3.144056
258                             -3.046867
259                             -4.052697
280                              4.170347
346                             -3.767916
482                             -3.212481
632                             -3.661161
979                             -4.750997
    diff_results$bootstrap_p[diff_list_boot]
146                                        0
252                                        0
258                                        0
259                                        0
280                                        0
346                                        0
482                                        0
632                                        0
979                                        0
```