Introduction to RBM package

Dongmei Li

May 5, 2024

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] 48

> which(myresult$permutation_p<=0.05)

 [1]  31  45  68  89  91 104 110 111 113 183 197 228 249 265 266 280 282 293 312
[20] 322 326 344 360 432 442 530 549 597 635 646 655 703 762 765 798 803 811 837
[39] 843 869 871 878 909 910 927 928 956 960

> sum(myresult$bootstrap_p<=0.05)

[1] 10

> which(myresult$bootstrap_p<=0.05)

 [1] 149 252 409 429 563 703 869 871 876 909

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

[1] 6

> 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] 20

> which(myresult2$bootstrap_p<=0.05)

 [1]  31 218 244 281 349 366 378 392 406 534 562 628 701 747 783 835 856 939 951
[20] 964

> 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] 79

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

[1] 57

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

 [1]  20  33  46  60  89 124 127 185 194 228 231 270 272 274 300 308 348 349 353
[20] 354 364 368 369 400 401 434 454 461 468 491 526 542 562 604 606 608 612 634
[39] 643 646 658 671 683 711 722 727 760 762 764 771 776 781 823 856 860 879 891
[58] 899 900 926 937 940 948 969 970 985 986 996

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

 [1]  20  33  46  53  60  89  93 119 124 127 156 185 194 231 270 272 274 300 308
[20] 348 349 353 354 363 364 368 369 400 401 405 434 439 446 454 461 468 491 504
[39] 526 542 551 562 582 600 604 606 608 612 634 643 646 658 671 683 696 711 722
[58] 727 760 776 781 809 823 829 856 860 891 893 899 926 930 937 940 948 969 982
[77] 985 986 996

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

 [1]  33  46  60  89 124 127 185 193 231 270 300 308 349 353 354 364 368 369 400
[20] 434 454 461 468 526 542 591 606 612 634 646 658 674 683 711 722 727 760 762
[39] 781 802 823 835 856 860 881 891 899 915 926 937 940 948 969 982 985 986 996
```

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

[1] 10

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

[1] 21

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

[1] 5

> which(con2_adjp<=0.05/3)

 [1]  33 124 185 270 300 349 353 354 364 369 434 454 634 646 722 760 823 856 926
[20] 937 985

> which(con3_adjp<=0.05/3)

[1] 354 722 760 937 940

> 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] 76

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

[1] 68
```

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

 [1]   11   71   76   87  114  152  165  248  255  270  275  317  336  340  345
[16]  365  368  403  409  415  416  432  439  448  475  479  510  524  550  553
[31]  583  584  597  622  653  698  708  737  762  788  795  807  833  846  858
[46]  901  962  964  967  968  981 1000

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

 [1]    1   11   27   29   71   76   87  114  142  146  149  152  165  197  232
[16]  236  248  255  265  270  274  275  317  336  345  357  368  384  394  396
[31]  403  409  415  416  439  448  457  467  475  479  510  519  524  550  553
[46]  570  583  584  597  609  643  648  653  698  708  711  737  762  788  792
[61]  833  846  858  867  879  901  937  962  964  967  968  979  981  983  998
[76] 1000

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

 [1]    1   11   12   27   32   71   76  114  142  144  152  165  197  237  248
[16]  255  265  270  274  275  317  336  345  368  396  403  409  410  415  416
[31]  432  439  448  475  479  510  519  524  583  584  597  609  622  643  653
[46]  698  708  711  734  737  762  788  792  795  833  858  867  901  927  962
[61]  964  967  968  979  981  983  994 1000

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

[1] 5

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

[1] 10

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

[1] 14
```

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] "/private/tmp/Rtmp3eMrek/Rinst127206d39123b/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] 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] 10

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

[1] 8

> 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]
19  cg00016968 0.80628480            NA    0.81440820    0.83623180
103 cg00094319 0.73784280    0.73532960    0.75574900    0.73830220
131 cg00121904 0.15449580    0.17949750    0.23608110    0.24354150
245 cg00224508 0.04479948    0.04972043    0.04152814    0.04189373
627 cg00612467 0.04777553    0.03783457    0.05380982    0.05582291
764 cg00730260 0.90471270    0.90542290    0.91002680    0.91258610
848 cg00826384 0.05721674    0.05612171    0.06644259    0.06358381
851 cg00830029 0.58362500    0.59397870    0.64739610    0.67269640
887 cg00862290 0.43640520    0.54047160    0.60786800    0.56325950
979 cg00945507 0.13432250    0.23854600    0.34749760    0.28903340
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
19     0.80831380    0.73306440    0.82968340    0.84917800
```

```
103    0.67349260    0.73510200    0.75715920    0.78981220
131    0.17352980    0.12564280    0.18193170    0.20847670
245    0.04208405    0.05284988    0.03775905    0.03955271
627    0.04740551    0.05332965    0.05775211    0.05579710
764    0.90575890    0.88760470    0.90756300    0.90946790
848    0.05230160    0.06119713    0.06542751    0.06240686
851    0.50820240    0.34657470    0.66276570    0.64634510
887    0.50259740    0.40111730    0.56646700    0.54552980
979    0.11848510    0.16653850    0.30718420    0.26624740
    diff_results$ordfit_t[diff_list_perm]
19                                -2.446404
103                               -2.268711
131                               -3.451679
245                                1.962457
627                               -2.239498
764                               -1.808081
848                               -2.314412
851                               -2.841244
887                               -3.217939
979                               -4.750997
    diff_results$permutation_p[diff_list_perm]
19                                            0
103                                           0
131                                           0
245                                           0
627                                           0
764                                           0
848                                           0
851                                           0
887                                           0
979                                           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
200 cg00183916 0.03525946    0.03984548    0.02765822    0.02789838
259 cg00234961 0.04192170    0.04321576    0.05707140    0.05327565
280 cg00260778 0.64319890    0.60488960    0.56735060    0.53150910
285 cg00263760 0.09050395    0.10197760    0.14801710    0.12242400
743 cg00717862 0.07999436    0.07873347    0.06089359    0.06171374
882 cg00858899 0.11427700    0.11919540    0.07690343    0.08321229
911 cg00888479 0.07388961    0.07361080    0.10149800    0.09985076
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
95     0.04561792    0.05115624    0.06068253    0.06168212
```

```
200    0.03034811    0.04302129    0.02753873    0.03067437
259    0.04030003    0.03996053    0.05086962    0.05445672
280    0.61920530    0.61925200    0.46753250    0.55632410
285    0.11693600    0.10650430    0.12281160    0.12310430
743    0.07594936    0.09062161    0.06475791    0.07271878
882    0.08961409    0.10730660    0.09203980    0.08726349
911    0.08633986    0.06765189    0.09070268    0.12417730
    diff_results$ordfit_t[diff_list_boot]
95                              -3.252063
200                              2.272449
259                             -4.052697
280                              4.170347
285                             -3.093997
743                              3.444684
882                              3.179415
911                             -3.621731
    diff_results$bootstrap_p[diff_list_boot]
95                                         0
200                                        0
259                                        0
280                                        0
285                                        0
743                                        0
882                                        0
911                                        0
```