Introduction to RBM package

Dongmei Li

May 1, 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] 20

> which(myresult$permutation_p<=0.05)

 [1]  34 121 167 170 184 198 199 288 339 414 490 523 568 613 674 756 854 855 882
[20] 946

> sum(myresult$bootstrap_p<=0.05)

[1] 5

> which(myresult$bootstrap_p<=0.05)

[1] 184 249 376 509 688

> 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]  34  95  96 157 211 239 241 278 279 312 313 339 355 358 378 506 514 573 756
[20] 769 770 775 832 851 865 917 938 971 978 987 992 995 996

> 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] 64

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

[1] 58

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

[1] 44

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

 [1]  24  27  36  48  51  66 100 147 190 233 241 248 290 293 307 320 330 335 349
[20] 368 369 382 410 418 419 447 496 501 505 518 519 525 526 546 555 563 580 584
[39] 589 590 602 634 652 663 674 688 718 727 746 777 780 786 817 826 834 867 895
[58] 898 906 916 923 974 982 999

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

 [1]  27  42  48  51  53  66  71 117 147 190 233 241 248 283 290 293 307 330 335
[20] 349 368 369 382 410 419 447 496 500 501 505 525 526 546 571 584 589 590 602
[39] 634 652 663 674 688 727 733 777 780 786 817 826 834 867 898 906 916 923 962
[58] 974

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

 [1]  27  48  51  53  66 117 147 241 290 293 307 320 349 368 369 382 410 419 447
[20] 496 501 505 571 584 589 634 652 663 674 688 718 727 777 780 786 834 842 867
[39] 898 906 916 962 974 999

> 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] 8

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

[1] 9

> which(con2_adjp<=0.05/3)

[1]  27  48 147 349 368 505 589 727

> which(con3_adjp<=0.05/3)

[1]  27  48 369 410 447 505 674 688 974

> 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] 74

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

[1] 62

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

[1] 73

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

```
 [1]   6  15  48  51  59  68 105 121 145 159 166 185 200 210 213 218 266 272 274
[20] 288 300 336 337 352 355 387 392 413 438 457 474 503 525 527 563 587 621 626
[39] 637 640 647 663 672 679 701 710 715 729 746 762 778 784 786 794 812 816 827
[58] 844 845 864 889 903 910 925 933 938 953 957 958 968 981 983 984 988

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

 [1]   6  15  39  48  59  68 145 159 166 185 200 206 210 213 218 231 266 272 273
[20] 289 300 336 352 387 392 413 457 474 503 563 583 621 626 634 637 640 668 679
[39] 691 701 715 729 746 776 778 786 794 816 827 845 864 889 903 910 925 951 957
[58] 968 981 983 984 988

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

 [1]   6  15  48  51  59  68  80  94 113 121 145 159 166 185 192 200 206 210 213
[20] 218 231 241 266 272 274 336 352 355 370 387 413 457 474 503 525 527 539 552
[39] 563 613 621 631 634 637 640 647 679 691 694 701 711 715 729 746 750 762 778
[58] 784 786 794 816 845 864 887 889 903 910 925 938 957 981 983 988

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

[1] 4

> 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] 2
```

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/Rtmp4o9fAI/Rinst11b7f4dee2d14/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] 55

> sum(diff_results$bootstrap_p<=0.05)

[1] NA

> 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] 11

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

[1] NA

> 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]
16  cg00014085 0.05906804    0.04518973    0.04211710    0.03665208
83  cg00072216 0.04505377    0.04598964    0.04000674    0.03231534
146 cg00134539 0.61101320    0.53321780    0.45999340    0.46787420
237 cg00215066 0.94926640    0.95311870    0.94634910    0.94561120
245 cg00224508 0.04479948    0.04972043    0.04152814    0.04189373
280 cg00260778 0.64319890    0.60488960    0.56735060    0.53150910
437 cg00424946 0.04122172    0.04325330    0.03339863    0.02876798
542 cg00520135 0.77510370    0.79688730    0.81833620    0.83043920
627 cg00612467 0.04777553    0.03783457    0.05380982    0.05582291
764 cg00730260 0.90471270    0.90542290    0.91002680    0.91258610
931 cg00901704 0.05734342    0.04812868    0.04478214    0.03878488
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
16     0.04222944    0.05324246    0.03728026    0.04062589
83     0.04965089    0.04833366    0.03466159    0.04390894
146    0.67191510    0.63137380    0.47929610    0.45428300
237    0.94837410    0.94665570    0.94089070    0.94600090
245    0.04208405    0.05284988    0.03775905    0.03955271
280    0.61920530    0.61925200    0.46753250    0.55632410
```

```
437    0.03353116    0.03719167    0.03096761    0.03234779
542    0.83062760    0.55544810    0.83402240    0.89514710
627    0.04740551    0.05332965    0.05775211    0.05579710
764    0.90575890    0.88760470    0.90756300    0.90946790
931    0.04497277    0.05751033    0.03089829    0.04423603
    diff_results$ordfit_t[diff_list_perm]
16                               2.325659
83                               2.514109
146                              5.394750
237                              1.419654
245                              1.962457
280                              4.170347
437                              2.102892
542                             -1.775375
627                             -2.239498
764                             -1.808081
931                              2.464709
    diff_results$permutation_p[diff_list_perm]
16                                          0
83                                          0
146                                         0
237                                         0
245                                         0
280                                         0
437                                         0
542                                         0
627                                         0
764                                         0
931                                         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]
16  cg00014085 0.05906804    0.04518973    0.04211710    0.03665208
95  cg00081975 0.03633894    0.04975194    0.06024723    0.05598723
146 cg00134539 0.61101320    0.53321780    0.45999340    0.46787420
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
833 cg00814580 0.09348613    0.09619816    0.12010440    0.11534240
882 cg00858899 0.11427700    0.11919540    0.07690343    0.08321229
887 cg00862290 0.43640520    0.54047160    0.60786800    0.56325950
928 cg00901493 0.03737166    0.03903724    0.04684618    0.04981432
931 cg00901704 0.05734342    0.04812868    0.04478214    0.03878488
```

```
979 cg00945507 0.13432250    0.23854600    0.34749760    0.28903340
    exmdata5[, 2] exmdata6[, 2] exmdata7[, 2] exmdata8[, 2]
16     0.04222944    0.05324246    0.03728026    0.04062589
95     0.04561792    0.05115624    0.06068253    0.06168212
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
285    0.11693600    0.10650430    0.12281160    0.12310430
743    0.07594936    0.09062161    0.06475791    0.07271878
833    0.09577040    0.11598850    0.12860890    0.14111200
882    0.08961409    0.10730660    0.09203980    0.08726349
887    0.50259740    0.40111730    0.56646700    0.54552980
928    0.04490690    0.04204062    0.05050039    0.05268215
931    0.04497277    0.05751033    0.03089829    0.04423603
979    0.11848510    0.16653850    0.30718420    0.26624740
    diff_results$ordfit_t[diff_list_boot]
16                               2.325659
95                              -3.252063
146                              5.394750
259                             -4.052697
280                              4.170347
285                             -3.093997
743                              3.444684
833                             -3.428319
882                              3.179415
887                             -3.217939
928                             -2.716443
931                              2.464709
979                             -4.750997
    diff_results$bootstrap_p[diff_list_boot]
16                                        0
95                                        0
146                                       0
259                                       0
280                                       0
285                                       0
743                                       0
833                                       0
882                                       0
887                                       0
928                                       0
931                                       0
979                                       0
```