Introduction to RBM package

Dongmei Li

April 16, 2015

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:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("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] 14
```

```
> which(myresult$permutation_p<=0.05)

 [1]   3  43  57  69 100 105 166 321 460 490 759 891 925 957

> sum(myresult$bootstrap_p<=0.05)

[1] 3

> which(myresult$bootstrap_p<=0.05)

[1]  23 315 851

> 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] 13

> which(myresult2$bootstrap_p<=0.05)

 [1]  50 166 295 498 511 560 710 728 804 874 903 920 931

> 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] 70

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

[1] 69

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

[1] 72

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

 [1]   1   3  14  19  21  34  46  76  79 100 129 139 154 172 174 181 191 197 208
[20] 217 249 252 260 285 324 329 331 356 366 376 383 388 424 465 478 487 496 509
[39] 512 528 564 568 592 608 614 620 659 669 671 709 726 732 741 742 743 767 784
[58] 791 807 813 819 867 875 889 905 913 928 958 979 995

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

 [1]   1   3  14  19  46  74  76  79  86 100 119 129 139 153 154 174 181 197 208
[20] 217 243 249 252 285 288 331 366 376 383 409 424 431 465 487 496 512 528 564
[39] 568 587 592 608 614 620 659 669 671 709 726 732 741 742 743 784 791 807 813
[58] 826 867 875 889 905 906 928 958 963 979 990 995

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

 [1]   1   3  14  19  74  76  79 100 119 120 129 139 141 153 154 174 181 195 197
[20] 208 217 249 252 260 285 331 356 383 424 478 487 496 509 512 528 549 560 564
[39] 568 592 595 608 614 620 659 669 671 726 741 742 743 767 771 782 784 791 807
[58] 812 813 843 867 875 889 905 906 928 932 945 958 979 990 995

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

```
[1] 19

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

[1] 12

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

[1] 17

> which(con2_adjp<=0.05/3)

 [1]   3  19 197 285 331 564 608 659 671 791 807 995

> which(con3_adjp<=0.05/3)

 [1]   1   3  19 197 285 424 564 608 671 791 807 889 905 928 958 990 995

> 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] 56

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

[1] 73

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

[1] 68

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

```
 [1]   8  50  61  65  85  94 119 123 178 191 199 203 206 220 225 227 266 321 325
[20] 347 358 376 394 417 458 503 506 516 519 539 551 558 565 613 633 644 648 664
[39] 679 693 741 745 755 771 806 816 829 840 867 882 902 930 954 964 967 988
```

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

```
 [1]   8  15  50  65  85 101 111 119 123 178 187 191 199 203 206 220 225 235 256
[20] 260 266 321 325 347 349 358 376 394 405 417 458 460 476 503 506 507 510 516
[39] 519 539 545 551 557 558 613 633 644 648 664 666 679 693 741 745 755 771 806
[58] 816 829 837 840 867 878 882 888 902 907 930 950 964 967 988 993
```

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

```
 [1]   5   8  15  50  61  65  71  97 119 123 156 191 199 203 206 220 225 227 235
[20] 242 266 321 323 325 339 347 349 358 367 394 405 417 458 460 469 503 506 511
[39] 516 519 539 545 551 558 613 633 644 648 664 666 679 693 697 741 755 771 806
[58] 813 816 829 867 878 882 902 930 964 967 988
```

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

```
[1] 8
```

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

```
[1] 9
```

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

```
[1] 15
```

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] "E:/biocbld/bbs-3.1-bioc/tmpdir/Rtmpa2cQH0/Rinst65c898746e/RBM/data"

> data(ovarian_cancer_methylation)
> summary(ovarian_cancer_methylation)

        IlmnID        case1             case2            control1
 cg00000292:  1   Min.   :0.01058   Min.   :0.01138   Min.   :0.009103
 cg00002426:  1   1st Qu.:0.04111   1st Qu.:0.04290   1st Qu.:0.041543
 cg00003994:  1   Median :0.08284   Median :0.10438   Median :0.087042
 cg00005847:  1   Mean   :0.27397   Mean   :0.29086   Mean   :0.283729
 cg00006414:  1   3rd Qu.:0.52135   3rd Qu.:0.54436   3rd Qu.:0.558575
 cg00007981:  1   Max.   :0.97069   Max.   :0.96901   Max.   :0.970155
 (Other)   :994
    control2           case3             case4             control3
 Min.   :0.01019   Min.   :0.01108   Min.   :0.009753   Min.   :0.01278
 1st Qu.:0.04092   1st Qu.:0.04059   1st Qu.:0.041818   1st Qu.:0.04260
 Median :0.09042   Median :0.08527   Median :0.092807   Median :0.09362
 Mean   :0.28508   Mean   :0.28482   Mean   :0.283113   Mean   :0.27563
 3rd Qu.:0.57502   3rd Qu.:0.57300   3rd Qu.:0.558211   3rd Qu.:0.52240
 Max.   :0.96658   Max.   :0.97516   Max.   :0.963620   Max.   :0.95974
                   NA's   :1         NA's   :1
    control4
 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] 31
```

```
> sum(diff_results$permutation_p<=0.05)

[1] 70

> sum(diff_results$bootstrap_p<=0.05)

[1] 41

> 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] 13

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

[1] 5

> 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      case1      case2   control1   control2      case3
66  cg00059424 0.02742616 0.02554150 0.03049395 0.02910234 0.02547771
129 cg00121158 0.03045297 0.02728770 0.03573820 0.03316130 0.02853104
193 cg00177923 0.04979515 0.04607616 0.04060709 0.04200000 0.04504412
245 cg00224508 0.04479948 0.04477529 0.04152814 0.04189373 0.04208405
432 cg00419564 0.03638860 0.03661916 0.04101457 0.04065540 0.03283922
454 cg00436282 0.79300120 0.79687150 0.78143640 0.77336780 0.81890080
460 cg00445824 0.14782870 0.16655800 0.14393210 0.13479670 0.20038750
601 cg00577464 0.02668995 0.04306255 0.02710953 0.01929375 0.02770019
660 cg00634577 0.03182804 0.03432180 0.03525499 0.03612398 0.03384897
690 cg00661202 0.01639344 0.01586308 0.01876500 0.02097005 0.01490915
764 cg00730260 0.90471270 0.90207400 0.91002680 0.91258610 0.90575890
772 cg00743372 0.03922780 0.03499011 0.02187972 0.02568053 0.02796053
815 cg00792849 0.12674200 0.17255180 0.12769620 0.11480600 0.15856800
         case4   control3   control4 diff_results$ordfit_t[diff_list_perm]
66  0.02523713 0.04478458 0.03391813                              -2.203752
129 0.02993313 0.03318921 0.03345607                              -2.118418
193 0.04416582 0.03535173 0.04315951                               2.080942
245 0.05731476 0.03775905 0.03955271                               1.811314
```

```
432 0.03934095 0.04413387 0.04462037                              -2.407553
454 0.81115150 0.67173700 0.77315940                               2.213478
460 0.16185300 0.11630830 0.13912630                               2.993440
601 0.03204772 0.02634007 0.02547999                               1.786901
660 0.03062112 0.03502489 0.03710039                              -1.402274
690 0.01764273 0.01847447 0.01803320                              -1.267423
764 0.90290550 0.90756300 0.90946790                              -2.477349
772 0.03001808 0.02575992 0.02093909                               2.824452
815 0.14693120 0.12092930 0.11909650                               3.209470
    diff_results$permutation_p[diff_list_perm]
66                                          0
129                                         0
193                                         0
245                                         0
432                                         0
454                                         0
460                                         0
601                                         0
660                                         0
690                                         0
764                                         0
772                                         0
815                                         0
```

```
> sig_results_boot <- cbind(ovarian_cancer_methylation[diff_list_boot, ], diff_results$ordfit_t
> print(sig_results_boot)

       IlmnID      case1      case2   control1   control2      case3
76  cg00065408 0.03952223 0.03967472 0.04799694 0.04929252 0.04064262
110 cg00098239 0.02698720 0.02142180 0.01856646 0.01934917 0.02510008
129 cg00121158 0.03045297 0.02728770 0.03573820 0.03316130 0.02853104
593 cg00568792 0.07330291 0.08046910 0.07976878 0.08124407 0.07145583
887 cg00862290 0.43640520 0.50563130 0.60786800 0.56325950 0.50259740
        case4   control3   control4 diff_results$ordfit_t[diff_list_boot]
76  0.03622954 0.04778213 0.04327211                              -3.115264
110 0.02291811 0.01784160 0.02109290                               1.976098
129 0.02993313 0.03318921 0.03345607                              -2.118418
593 0.07302756 0.08203244 0.08172888                              -2.328352
887 0.51368550 0.56646700 0.54552980                              -3.935409
    diff_results$bootstrap_p[diff_list_boot]
76                                        0
110                                       0
129                                       0
593                                       0
887                                       0
```