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] 17
```

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

 [1]  15  79 149 153 196 289 290 335 358 462 584 620 660 827 884 885 942

> sum(myresult$bootstrap_p<=0.05)

[1] 9

> which(myresult$bootstrap_p<=0.05)

[1]  36 117 374 481 535 601 654 798 963

> 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] 14

> which(myresult2$bootstrap_p<=0.05)

 [1] 151 367 446 485 674 677 713 750 774 777 813 856 886 903

> 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] 50

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

[1] 51

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

[1] 45

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

 [1]   3  25  28  54  81 121 124 134 171 186 194 220 258 280 327 328 354 360 366
[20] 368 385 418 422 425 460 466 487 552 576 600 633 671 686 695 707 735 749 755
[39] 761 768 775 778 788 823 866 878 883 931 932 987

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

 [1]   3  25  28  54  81 121 124 134 171 186 194 220 258 280 327 328 354 360 366
[20] 368 385 401 418 425 460 487 541 552 600 633 643 671 686 707 735 737 749 755
[39] 761 768 775 823 862 866 878 880 883 931 932 987 988

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

 [1]   3  25  28  54  74  81 124 134 171 194 220 258 280 327 328 360 366 368 385
[20] 401 418 422 425 460 487 552 600 671 686 707 749 761 768 775 778 788 798 823
[39] 878 883 925 931 975 987 988

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

[1] 5
```

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

[1] 7

> 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]  28 134 385 749 878 883 931

> which(con3_adjp<=0.05/3)

[1]  54  81 552 749 768 878 883

> 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] 57

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

[1] 51

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

[1] 57

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

 [1]   4  21  97  98 109 128 191 215 260 279 281 323 337 344 362 377 390 394 435
[20] 443 450 465 467 476 478 479 498 504 508 528 584 588 620 644 645 646 679 718
[39] 726 749 763 788 796 817 820 823 827 869 886 888 914 918 930 937 946 989 991
```

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

 [1]   4  21  97  98 125 191 193 215 223 279 281 323 337 344 362 377 390 394 406
[20] 435 443 450 465 467 476 478 479 504 528 584 588 644 645 646 679 718 763 788
[39] 800 817 823 827 843 869 888 914 918 937 946 989 991

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

 [1]   4  26  28  97  98 186 191 193 215 223 260 279 281 323 337 344 362 377 390
[20] 394 435 443 450 465 467 476 478 479 504 584 588 644 645 646 663 679 685 718
[39] 728 749 788 817 823 827 869 888 914 918 930 937 941 946 947 979 987 989 991

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

[1] 7

> 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] 10
```

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/RtmpsWIZAZ/Rinst3efe79af894d/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] 47

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

```
[1] 43

> 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] 2

> 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      case4
66  cg00059424 0.02742616 0.0255415 0.03049395 0.02910234 0.02547771 0.02523713
460 cg00445824 0.14782870 0.1665580 0.14393210 0.13479670 0.20038750 0.16185300
      control3   control4 diff_results$ordfit_t[diff_list_perm]
66  0.04478458 0.03391813                              -2.203752
460 0.11630830 0.13912630                               2.993440
    diff_results$permutation_p[diff_list_perm]
66                                           0
460                                          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      case4
165 cg00151234 0.4920805 0.44188460 0.35210020 0.39177560 0.55194460 0.41179330
488 cg00474209 0.0716709 0.05880237 0.04600708 0.05083778 0.05330264 0.06112561
      control3   control4 diff_results$ordfit_t[diff_list_boot]
165 0.31723050 0.38466600                               3.474724
488 0.05248991 0.04533533                               2.825490
    diff_results$bootstrap_p[diff_list_boot]
165                                        0
488                                        0
```