

Introduction to RBM package

Dongmei Li

October 29, 2025

Clinical and Translational Science Institute, University of Rochester School of Medicine and Dentistry, Rochester, NY 14642-0708

Contents

1 Overview	1
2 Getting started	2
3 RBM_T and RBM_F functions	2
4 Ovarian cancer methylation example using the RBM_T function	6

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 Benjamini-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] 37

> which(myresult$permutation_p<=0.05)

[1] 7 23 32 53 85 143 199 220 256 262 315 342 376 383 396 463 478 480 547
[20] 566 599 664 670 701 737 745 789 806 821 833 839 850 920 955 962 965 993

> sum(myresult$bootstrap_p<=0.05)

[1] 2

> which(myresult$bootstrap_p<=0.05)

[1] 254 547

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

[1] 0

> bootstrap_adj_p <- p.adjust(myresult$bootstrap_p, "BH")
> sum(bootstrap_adj_p<=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] 24

> which(myresult2$bootstrap_p<=0.05)

[1] 8 62 158 195 209 254 260 262 278 306 358 361 388 494 520 531 532 561 605
[20] 617 732 862 881 956

> bootstrap2_adj_p <- p.adjust(myresult2$bootstrap_p, "BH")
> sum(bootstrap2_adj_p<=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] 52

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

[1] 60

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

[1] 49

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

[1]  4  44  94 126 133 147 180 223 232 236 284 289 316 321 339 348 351 366 372
[20] 374 408 413 416 471 498 507 515 518 539 600 601 649 650 698 702 705 711 713
[39] 730 732 760 784 796 805 838 847 848 868 883 894 924 940

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

[1]  4  30  44  53 126 133 147 174 180 215 223 236 284 287 289 296 321 329 339
[20] 348 351 366 371 372 374 408 413 416 454 471 498 507 515 518 539 558 600 601
[39] 625 649 650 698 702 713 730 760 784 796 804 838 847 848 868 883 893 923 924
[58] 940 967 971

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

[1]  4  44 126 133 174 180 215 223 232 284 289 292 316 321 339 348 351 366 372
[20] 374 377 408 413 416 471 498 539 563 600 601 606 649 650 685 698 702 713 730
[39] 760 784 796 805 838 847 848 868 883 894 940

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

[1] 7

```

```

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

[1] 3

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

[1] 6

> which(con2_adjp<=0.05/3)

[1] 348 601 760

> which(con3_adjp<=0.05/3)

[1] 223 339 348 374 416 601

> 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] 59

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

[1] 55

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

[1] 59

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

[1] 12 27 30 98 122 126 139 150 161 176 180 191 208 222 228 233 248 259 307
[20] 366 375 396 398 409 411 416 433 465 485 543 559 571 595 614 635 638 642 646
[39] 703 709 721 724 731 736 760 803 809 835 839 846 893 902 904 910 920 921 942
[58] 950 972

```

```

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

[1] 12 30 98 122 126 150 161 176 180 208 213 222 230 233 248 259 265 307 320
[20] 375 396 409 411 416 422 465 485 512 526 543 559 571 614 623 635 642 646 659
[39] 691 692 703 709 721 736 743 780 809 835 893 902 904 920 921 942 972

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

[1] 12 26 70 95 98 122 126 150 161 180 191 208 222 233 238 248 259 307 320
[20] 366 375 396 411 416 422 465 485 486 512 526 543 559 571 595 614 623 624 635
[39] 642 646 649 691 692 721 724 743 760 803 809 812 835 839 893 902 904 920 921
[58] 972 984

> con21_adj_p <- p.adjust(myresult2_F$bootstrap_p[, 1], "BH")
> sum(con21_adj_p<=0.05/3)

[1] 4

> con22_adj_p <- p.adjust(myresult2_F$bootstrap_p[, 2], "BH")
> sum(con22_adj_p<=0.05/3)

[1] 5

> con23_adj_p <- p.adjust(myresult2_F$bootstrap_p[, 3], "BH")
> sum(con23_adj_p<=0.05/3)

[1] 3

```

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 genome-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 illustration purpose, we chose the first 1000 loci in 8 randomly selected women with 4 ovarian 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/var/folders/r0/l4fjk6cj5xj0j3brt4bplpl40000gt/T/RtmpFikMCY/Rinst11bb54d2b6831/RBM/"

> 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.:0.59031	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] 47

> sum(diff_results$permutation_p<=0.05)

[1] 70

> sum(diff_results$bootstrap_p<=0.05)

```

```
[1] 64
```

```
> 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] 32
```

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

```
[1] 10
```

```
> 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[diff_list_perm, ])
> print(sig_results_perm)
```

	IlmnID	Beta	exmdata2[, 2]	exmdata3[, 2]	exmdata4[, 2]
16	cg00014085	0.05906804	0.04518973	0.04211710	0.03665208
19	cg00016968	0.80628480	NA	0.81440820	0.83623180
83	cg00072216	0.04505377	0.04598964	0.04000674	0.03231534
95	cg00081975	0.03633894	0.04975194	0.06024723	0.05598723
103	cg00094319	0.73784280	0.73532960	0.75574900	0.73830220
106	cg00095674	0.07076291	0.05045181	0.03861991	0.03337576
131	cg00121904	0.15449580	0.17949750	0.23608110	0.24354150
146	cg00134539	0.61101320	0.53321780	0.45999340	0.46787420
219	cg00202702	0.04104248	0.04685628	0.03793627	0.03529329
237	cg00215066	0.94926640	0.95311870	0.94634910	0.94561120
245	cg00224508	0.04479948	0.04972043	0.04152814	0.04189373
259	cg00234961	0.04192170	0.04321576	0.05707140	0.05327565
280	cg00260778	0.64319890	0.60488960	0.56735060	0.53150910
283	cg00262415	0.03850601	0.04621248	0.03579758	0.03765227
285	cg00263760	0.09050395	0.10197760	0.14801710	0.12242400
349	cg00332745	0.04703361	0.04634372	0.03676908	0.04518837
437	cg00424946	0.04122172	0.04325330	0.03339863	0.02876798
520	cg00502442	0.03163993	0.03581662	0.02785063	0.02549502
627	cg00612467	0.04777553	0.03783457	0.05380982	0.05582291
632	cg00615377	0.11265030	0.16140570	0.19404450	0.17468600
743	cg00717862	0.07999436	0.07873347	0.06089359	0.06171374
764	cg00730260	0.90471270	0.90542290	0.91002680	0.91258610
772	cg00743372	0.03922780	0.02919634	0.02187972	0.02568053
804	cg00777121	0.04540701	0.05430304	0.04154242	0.04221162

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
911	cg00888479	0.07388961	0.07361080	0.10149800	0.09985076
928	cg00901493	0.03737166	0.03903724	0.04684618	0.04981432
931	cg00901704	0.05734342	0.04812868	0.04478214	0.03878488
939	cg00906183	0.03949030	0.04365079	0.03720015	0.03575748
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	
19	0.80831380	0.73306440	0.82968340	0.84917800	
83	0.04965089	0.04833366	0.03466159	0.04390894	
95	0.04561792	0.05115624	0.06068253	0.06168212	
103	0.67349260	0.73510200	0.75715920	0.78981220	
106	0.04693030	0.06837343	0.04534005	0.03709488	
131	0.17352980	0.12564280	0.18193170	0.20847670	
146	0.67191510	0.63137380	0.47929610	0.45428300	
219	0.04074652	0.05125000	0.03908795	0.04075583	
237	0.94837410	0.94665570	0.94089070	0.94600090	
245	0.04208405	0.05284988	0.03775905	0.03955271	
259	0.04030003	0.03996053	0.05086962	0.05445672	
280	0.61920530	0.61925200	0.46753250	0.55632410	
283	0.03746915	0.04200230	0.03014699	0.02903290	
285	0.11693600	0.10650430	0.12281160	0.12310430	
349	0.04975075	0.05253778	0.04444665	0.03717721	
437	0.03353116	0.03719167	0.03096761	0.03234779	
520	0.03111720	0.03189393	0.02415307	0.02941176	
627	0.04740551	0.05332965	0.05775211	0.05579710	
632	0.12573100	0.14483660	0.16338240	0.20130510	
743	0.07594936	0.09062161	0.06475791	0.07271878	
764	0.90575890	0.88760470	0.90756300	0.90946790	
772	0.02796053	0.03512214	0.02575992	0.02093909	
804	0.04911277	0.04872797	0.04261405	0.04474881	
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	
911	0.08633986	0.06765189	0.09070268	0.12417730	
928	0.04490690	0.04204062	0.05050039	0.05268215	
931	0.04497277	0.05751033	0.03089829	0.04423603	
939	0.03856975	0.06024309	0.03594439	0.03502819	
979	0.11848510	0.16653850	0.30718420	0.26624740	
	diff_results\$ordfit_t[diff_list_perm]				
16		1.954876			
19		-2.547097			
83		1.947226			

95	-2.654324
103	-2.343784
106	2.887876
131	-3.562745
146	5.636263
219	1.375603
237	1.021426
245	1.494678
259	-2.833203
280	4.337628
283	1.601804
285	-2.993292
349	1.659117
437	1.574598
520	1.319602
627	-1.797392
632	-3.722206
743	2.918806
764	-1.560713
772	1.885560
804	1.445572
848	-1.687144
851	-2.986319
887	-3.368752
911	-3.490240
928	-1.982308
931	2.127264
939	1.558215
979	-4.968792
diff_results\$permutation_p[diff_list_perm]	
16	0
19	0
83	0
95	0
103	0
106	0
131	0
146	0
219	0
237	0
245	0
259	0
280	0
283	0
285	0

```

349          0
437          0
520          0
627          0
632          0
743          0
764          0
772          0
804          0
848          0
851          0
887          0
911          0
928          0
931          0
939          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]
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
346	cg00331237	0.05972383	NA	0.08204769	0.08345662
351	cg00333528	0.08219988	0.16874890	0.14585930	0.17866410
632	cg00615377	0.11265030	0.16140570	0.19404450	0.17468600
677	cg00651216	0.06825629	0.12529090	0.14409190	0.13907250
743	cg00717862	0.07999436	0.07873347	0.06089359	0.06171374
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]	
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	
346	0.05372019	0.06241126	0.06955040	0.09140985	
351	0.09031025	0.11959060	0.15422700	0.20188360	
632	0.12573100	0.14483660	0.16338240	0.20130510	
677	0.07669587	0.09597587	0.11690440	0.15194540	
743	0.07594936	0.09062161	0.06475791	0.07271878	
887	0.50259740	0.40111730	0.56646700	0.54552980	
979	0.11848510	0.16653850	0.30718420	0.26624740	
	diff_results\$ordfit_t[diff_list_boot]				
146	5.636263				
259	-2.833203				

280	4.337628
346	-3.328798
351	-2.664069
632	-3.722206
677	-3.457874
743	2.918806
887	-3.368752
979	-4.968792
diff_results\$bootstrap_p[diff_list_boot]	
146	0
259	0
280	0
346	0
351	0
632	0
677	0
743	0
887	0
979	0