

An Introduction to the *REMP* Package

Yinan Zheng

December 2, 2017

Contents

1	Introduction	1
2	Installation	1
3	REMP: Repetitive Element Methylation Prediction	1
3.1	Groom methylation data	2
3.2	Prepare annotation data	2
3.3	Run prediction	3
3.4	Plot prediction	9
4	Extract RE-CpG methylation profiled by Illumina BeadChip array	10
5	Session Information	11

1 Introduction

REMP predicts DNA methylation of locus-specific repetitive elements (RE) by learning surrounding genetic and epigenetic information. *REMP* provides genomewide single-base resolution of DNA methylation on RE that are difficult to measure using array-based or sequencing-based platforms, which enables epigenome-wide association study (EWAS) and differentially methylated region (DMR) analysis on RE.

2 Installation

Install *REMP* (release version):

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

To install devel version:

```
> library(devtools)
> install_github("YinanZheng/REMP")
```

Load *REMP* into the workspace:

```
> library(REMP)
```

3 REMP: Repetitive Element Methylation Prediction

Currently *REMP* supports Human (hg19, build 37) Alu and LINE-1 (L1) repetitive element (RE) methylation prediction using Illumina 450k or EPIC array.

3.1 Groom methylation data

Appropriate data preprocessing including quality control and normalization of methylation data are recommended before running *REMP*. Many packages are available to carry out these data preprocessing steps, for example, *minfi*, *wateRmelon*, and *methylumi*.

REMP is trying to minimize the requirement of the methylation data format. User can maintain the methylation data in *RatioSet* or *GenomicRatioSet* object offered by *minfi*, *data.table*, *data.frame*, *DataFrame*, or *matrix*. User can input either beta value or M-value. There are only two basic requirements of the methylation data:

1. Each row should represent CpG probe and each column should represent sample.
2. The row names should indicate Illumina probe ID (i.e. cg000000029).

However, there are some other common data issues that may prevent *REMP* from running correctly. For example, if the methylation data are in beta value and contain zero methylation values, logit transformation (to create M-value) will create negative infinite value; or the methylation data contain NA, Inf, or NaN data. To tackle these potential issues, *REMP* includes a handy function `groomMethy` which can help detect and fix these issues. We highly recommend to take advantage of this function:

```
> # Get GM12878 methylation data (450k array)
> GM12878_450k <- getGM12878('450k')
> GM12878_450k <- groomMethy(GM12878_450k, verbose = TRUE)
> GM12878_450k
```

```
class: RatioSet
dim: 482421 1
metadata(0):
assays(2): Beta M
rownames(482421): cg000000029 cg000000108 ...
               cg27666046 cg27666123
rowData names(0):
colnames(1): GM12878
colData names(0):
Annotation
  array: IlluminaHumanMethylation450k
  annotation: ilmn12.hg19
Preprocessing
  Method: NA
  minfi version: NA
  Manifest version: NA
```

For zero beta values, `groomMethy` will replace them with smallest non-zero beta value. For NA/NaN/Inf values, `groomMethy` will treat them as missing values and then apply KNN-imputation to complete the dataset. If the imputed value is out of the original range (which is possible when `imputebyrow = FALSE`), mean value will be used instead. Warning: imputed values for multimodal distributed CpGs (across samples) may not be correct. Please check package *ENmix* to identify the CpGs with multimodal distribution.

3.2 Prepare annotation data

To run *REMP* for RE methylation prediction, user first needs to prepare some annotation datasets. The function `initREMP` is designed to do the job.

Suppose user will predict Alu methylation using Illumina 450k array data:

```
> data(Alu.demo)
> remparcel <- initREMP(arrayType = "450k", REtype = "Alu",
+                       RE = Alu.demo, ncore = 1)
> remparcel
```

```
REMPParcel object
RE type: Alu
Illumina platform: 450k
Valid (max) Alu-CpG flanking window size: 1200
Number of RE: 500
Number of Alu-CpG: 5039
```

For demonstration, we only use 500 selected Alu sequence dataset which comes along with the package (`Alu.demo`). We specify `RE = Alu.demo`, so that the annotation dataset will be generated for the 500 selected Alu sequences. Most of the time, specifying `RE` is not necessary, as the function will fetch the complete RE sequence dataset from package *AnnotationHub* using `fetchRMSK`. User can also use this argument `RE` to provide customized RE dataset.

All data are stored in the *REMPParcel* object:

```
> saveParcel(remparcel)
```

It is recommended to specify a working directory using argument `work.dir` so that the data generated can be preserved for later use. Without specifying working directory, the annotation dataset will be created under the temporal directory `tempdir()` by default. User can also turn on the `export` parameter in `initREMP` to save the data automatically.

3.3 Run prediction

Once the annotation data are ready, user can pass the annotation data parcel to `remp` for prediction:

```
> remp.res <- remp(GM12878_450k, REtype = 'Alu',
+                  parcel = remparcel, ncore = 1, seed = 777)
```

If `parcel` is missing, `remp` will then try to search the *REMPParcel* data file in the directory indicated by `work.dir`. If `work.dir` is also missing, `remp` will try to search the *REMPParcel* data file in the temporal directory `tempdir()`.

By default, `remp` uses Random Forest (`method = 'rf'`) model (package *randomForest*) for prediction. Random Forest model is recommended because it offers more accurate prediction results and it automatically enables Quantile Regression Forest (Nicolai Meinshausen, 2006) for prediction reliability evaluation. `remp` constructs 19 predictors to carry out the prediction. For Random Forest model, the tuning parameter `param = 6` (i.e. `mtry` in *randomForest*) indicates how many predictors will be randomly selected for building the individual trees. The performance of random forest model is often relatively insensitive to the choice of `mtry`. Therefore, auto-tune will be turned off using random forest and `mtry` will be set to one third of the total number of predictors. It is recommended to specify a seed for reproducible prediction results.

`remp` will return a *REMPset* object, which inherits Bioconductor's *RangedSummarizedExperiment* class:

```
> remp.res

class: REMProduct
dim: 4808 1
metadata(8): REannotation RECPG ... GeneStats Seed
assays(3): rempB rempM rempQC
rownames: NULL
rowData names(1): RE.Index
colnames(1): GM12878
colData names(1): mtry

> # Display more detailed information
> details(remp.res)
```

```
RE type: Alu
Methylation profiling platform: 450k
Flanking window size: 1000
```

Prediction model: Random Forest
 QC model: Quantile Regression Forest
 Seed: 777
 Covered 4808 CpG sites in 500 Alu

Number of Alu-CpGs by chromosome:

chr1	chr2	chr3	chr4	chr5	chr6	chr7	chr8
449	276	293	131	179	397	292	102

chr9	chr10	chr11	chr12	chr13	chr14	chr15	chr16
98	148	254	310	66	127	133	333

chr17	chr18	chr19	chr20	chr21	chr22
295	81	674	66	37	67

Training information:

500 profiled Alu by Illumina array are used for model training.
 481 Alu-CpGs that have at least 2 neighboring profiled CpGs are used for model training.

Coverage information:

The data cover 500 Alu (4808 Alu-CpG).
 Gene coverage by Alu (out of total refSeq Gene):
 492 (1.96%) total genes;
 413 (2.15%) protein-coding genes;
 117 (1.59%) non-coding RNA genes.

Distribution of methylation value (beta value):

Min.	1st Qu.	Median	Mean	3rd Qu.
0.02704267	0.47479653	0.66164191	0.59629296	0.75054362
Max.				
0.92317298				

Distribution of reliability score:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.6266425	1.4569065	1.7416376	1.7866886	2.0532293	5.5405940

Prediction results can be obtained by accessors:

```
> # Predicted RE-CpG methylation value (Beta value)
> rempB(remp.res)
```

DataFrame with 4808 rows and 1 column

```
GM12878
<numeric>
1 0.7636445
2 0.7898260
3 0.7952402
4 0.7972490
5 0.7991708
...
4804 0.4460638
4805 0.4902897
4806 0.4837586
4807 0.4998728
4808 0.5074072
```

```

> # Predicted RE-CpG methylation value (M value)
> rempM(remp.res)

DataFrame with 4808 rows and 1 column
      GM12878
      <numeric>
1      1.691942
2      1.909950
3      1.957459
4      1.975321
5      1.992535
...
4804 -0.3124699826
4805 -0.0560429811
4806 -0.0937582629
4807 -0.0007342609
4808  0.0427487349

> # Genomic location information of the predicted RE-CpG
> # Function inherit from class 'RangedSummarizedExperiment'
> rowRanges(remp.res)

GRanges object with 4808 ranges and 1 metadata column:
      seqnames      ranges strand |      RE.Index
      <Rle>      <IRanges> <Rle> |      <Rle>
[1]      chr1      [943927, 943928] - | Alu_0000527
[2]      chr1      [943935, 943936] - | Alu_0000527
[3]      chr1      [943968, 943969] - | Alu_0000527
[4]      chr1      [943974, 943975] - | Alu_0000527
[5]      chr1      [943991, 943992] - | Alu_0000527
...
[4804]      chr22 [42095154, 42095155] - | Alu_1170175
[4805]      chr22 [42095161, 42095162] - | Alu_1170175
[4806]      chr22 [42095170, 42095171] - | Alu_1170175
[4807]      chr22 [42095198, 42095199] - | Alu_1170175
[4808]      chr22 [42095214, 42095215] - | Alu_1170175
-----
seqinfo: 93 sequences from an unspecified genome; no seqlengths

> # Standard error-scaled permutation importance of predictors
> rempImp(remp.res)

DataFrame with 19 rows and 1 column
      GM12878
      <numeric>
RE.score      9.559523
RE.Length      8.542915
RE.CpG.density  5.794904
RE.InNM        3.571228
RE.InNR        1.870291
...
distance.min2  14.391297
Methy.min      29.221280
Methy.min2     11.668607
Methy.mean     11.857837
Methy.std       4.294336

```

```
> # Retrive seed number used for the reesults
> metadata(remp.res)$Seed
```

```
[1] 777
```

Trim off less reliable predicted results:

```
> # Any predicted CpG values with quality score less than
> # threshold (default = 1.7) will be replaced with NA.
> # CpGs contain more than missingRate * 100% (default = 20%)
> # missing rate across samples will be discarded.
> remp.res <- rempTrim(remp.res, threshold = 1.7, missingRate = 0.2)
> details(remp.res)
```

```
RE type: Alu
Methylation profiling platform: 450k
Flanking window size: 1000
Prediction model: Random Forest - trimmed (1.7)
QC model: Quantile Regression Forest
Seed: 777
Covered 2253 CpG sites in 384 Alu
```

Number of Alu-CpGs by chromosome:

```
chr1 chr2 chr3 chr4 chr5 chr6 chr7 chr8
202 138 153 72 64 219 123 56
```

```
chr9 chr10 chr11 chr12 chr13 chr14 chr15 chr16
42 88 84 114 24 102 27 185
```

```
chr17 chr18 chr19 chr20 chr21 chr22
143 42 300 43 7 25
```

Coverage information:

```
The data cover 384 Alu (2253 Alu-CpG).
Gene coverage by Alu (out of total refSeq Gene):
368 (1.47%) total genes;
302 (1.57%) protein-coding genes;
90 (1.22%) non-coding RNA genes.
```

Distribution of methylation value (beta value):

```
Min. 1st Qu. Median Mean 3rd Qu.
0.06450368 0.66316026 0.74097110 0.70317242 0.79073142
Max.
0.90961584
```

Distribution of reliability score:

```
Min. 1st Qu. Median Mean 3rd Qu. Max.
0.6266425 1.3158947 1.4387024 1.4222391 1.5640511 1.6992731
```

Aggregate the predicted methylation of CpGs in RE by averaging them to obtain the RE-specific methylation level:

```
> remp.res <- rempAggregate(remp.res, NCpG = 2)
> details(remp.res)
```

RE type: Alu (aggregated by mean: min # of CpGs: 2)
Methylation profiling platform: 450k
Flanking window size: 1000
Prediction model: Random Forest - trimmed (1.7)
QC model: Quantile Regression Forest
Seed: 777
Covered 295 Alu (aggregated by mean: min # of CpGs: 2)

Number of Alu (aggregated by mean: min # of CpGs: 2) by chromosome:

chr1	chr2	chr3	chr4	chr5	chr6	chr7	chr8
26	17	21	10	7	30	17	7

chr9	chr10	chr11	chr12	chr13	chr14	chr15	chr16
5	13	10	14	4	9	5	23

chr17	chr18	chr19	chr20	chr21	chr22
17	6	41	7	2	4

Coverage information:

The data cover 295 Alu (aggregated by mean: min # of CpGs: 2)
Gene coverage by Alu (aggregated by mean: min # of CpGs: 2) (out of total refSeq Gene):
278 (1.11%) total genes;
228 (1.19%) protein-coding genes;
65 (0.88%) non-coding RNA genes.

Distribution of methylation value (beta value):

Min.	1st Qu.	Median	Mean	3rd Qu.
0.07892564	0.63379146	0.72447535	0.68432307	0.77716362
Max.				
0.85898677				

Distribution of reliability score:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.86999909	1.3508105	1.4525065	1.4348347	1.5282582	1.6954474

It is recommended to aggregate the predicted CpG methylation by RE to obtain RE-wise methylation level. This is beneficial because 1) it greatly reduces the data dimension for downstream analysis and 2) it produces more robust RE methylation estimation. Note that by default, RE with 2 or more predicted CpG sites will be aggregated. Therefore, the downside of doing this is the reduced coverage of RE. The assumption of doing this is the CpG methylation level within each RE are similar.

To add genomic regions annotation of the predicted REs:

```
> # By default gene symbol annotation will be added
> remp.res <- decodeAnnot(remp.res)
> rempAnnot(remp.res)
```

GRanges object with 295 ranges and 10 metadata columns:

	seqnames	ranges	strand	name
	<Rle>	<IRanges>	<Rle>	<character>
[1]	chr1	[943896, 944203]	-	AluSq2
[2]	chr1	[1197026, 1197260]	-	AluY
[3]	chr1	[6265537, 6265810]	-	AluSz
[4]	chr1	[7914008, 7914123]	+	AluJb
[5]	chr1	[12039040, 12039314]	+	AluJr
...

[291]	chr21	[37506146, 37506436]	-	AluSg
[292]	chr22	[22987350, 22987681]	-	AluY
[293]	chr22	[24058845, 24058989]	+	AluJb
[294]	chr22	[24103787, 24104072]	+	AluSx1
[295]	chr22	[27050687, 27050975]	-	AluY

	score	Index	InNM.symbol	InNR.symbol
	<numeric>	<Rle>	<character>	<character>
[1]	2417	Alu_0000527	<NA>	<NA>
[2]	1697	Alu_0000636	UBE2J2	<NA>
[3]	2092	Alu_0002477	RNF207	<NA>
[4]	1680	Alu_0003481	UTS2	<NA>
[5]	1328	Alu_0007823	MFN2	<NA>

...
[291]	2311	Alu_1146404	CBR3	CBR3-AS1
[292]	2378	Alu_1156182	GGTLC2	POM121L1P
[293]	933	Alu_1156674	<NA>	GUSEBP11
[294]	1980	Alu_1156753	C22orf15	<NA>
[295]	2468	Alu_1158731	<NA>	<NA>

	InTSS.symbol	In5UTR.symbol	InCDS.symbol
	<character>	<character>	<character>
[1]	<NA>	<NA>	<NA>
[2]	<NA>	<NA>	UBE2J2
[3]	RNF207	<NA>	<NA>
[4]	UTS2	<NA>	<NA>
[5]	MFN2	<NA>	<NA>

...
[291]	CBR3	<NA>	<NA>
[292]	POM121L1P GGTLC2	GGTLC2	<NA>
[293]	<NA>	<NA>	<NA>
[294]	C22orf15	<NA>	<NA>
[295]	<NA>	<NA>	<NA>

	InExon.symbol	In3UTR.symbol
	<character>	<character>
[1]	<NA>	<NA>
[2]	<NA>	<NA>
[3]	<NA>	<NA>
[4]	<NA>	<NA>
[5]	<NA>	<NA>

...
[291]	<NA>	<NA>
[292]	<NA>	<NA>
[293]	<NA>	<NA>
[294]	<NA>	<NA>
[295]	<NA>	<NA>

seqinfo: 93 sequences (1 circular) from hg19 genome

Seven genomic region indicators will be added to the annotation data in the input *REMP* object:

- InNM: in protein-coding genes (overlap with refSeq gene's "NM" transcripts + 2000 bp upstream of the transcription start site (TSS))
- InNR: in noncoding RNA genes (overlap with refSeq gene's "NR" transcripts + 2000 bp upstream of the TSS)
- InTSS: in flanking region of 2000 bp upstream of the TSS. Default upstream limit is 2000 bp, which can be modified globally using `remp_options`

- In5UTR: in 5'untranslated regions (UTRs)
- InCDS: in coding DNA sequence regions
- InExon: in exon regions
- In3UTR: in 3'UTRs

Note that intron region and intergenic region information can be derived from the above genomic region indicators: if "InNM" and/or "InNR" is not missing but "InTSS", "In5UTR", "InExon", and "In3UTR" are missing, then the RE is strictly located within intron region; if all indicators are missing, then the RE is strictly located in intergenic region.

3.4 Plot prediction

Make a density plot of the predicted methylation (beta values):

```
> plot(remp.res, main = "Alu methylation (GM12878)", col = "blue")
```

4 Extract RE-CpG methylation profiled by Illumina BeadChip array

REMP offers a handy tool to extract methylation data of CpGs that are located in RE.

```
> # Use Alu.demo for demonstration
> remp.res <- remprofile(GM12878_450k, REtype = "Alu", RE = Alu.demo)
> details(remp.res)
```

```
RE type: Alu
Methylation profiling platform: 450k
Flanking window size: N/A
Prediction model: Profiled
QC model: N/A
Covered 594 CpG sites in 500 Alu
```

Number of Alu-CpGs by chromosome:

chr1	chr2	chr3	chr4	chr5	chr6	chr7	chr8
51	38	39	20	21	61	39	13

chr9	chr10	chr11	chr12	chr13	chr14	chr15	chr16
11	22	25	31	8	15	21	34

chr17	chr18	chr19	chr20	chr21	chr22
28	11	82	11	4	9

Coverage information:

```
The data cover 500 Alu (594 Alu-CpG).
Gene coverage by Alu (out of total refSeq Gene):
  492 (1.96%) total genes;
  413 (2.15%) protein-coding genes;
  117 (1.59%) non-coding RNA genes.
```

Distribution of methylation value (beta value):

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.0010000	0.4232500	0.6695000	0.6052828	0.8280000	0.9710000

```
> # All accessors and utilites for REMProduct are applicable
> remp.res <- rempAggregate(remp.res)
> details(remp.res)
```

```
RE type: Alu (aggregated by mean: min # of CpGs: 2)
Methylation profiling platform: 450k
Flanking window size: N/A
Prediction model: Profiled
QC model: N/A
Covered 75 Alu (aggregated by mean: min # of CpGs: 2)
```

Number of Alu (aggregated by mean: min # of CpGs: 2) by chromosome:

chr1	chr2	chr3	chr4	chr5	chr6	chr7	chr8
5	7	6	3	3	11	5	1

chr10	chr11	chr12	chr13	chr14	chr15	chr16	chr17
1	1	4	1	2	4	3	1

chr18	chr19	chr20	chr22
-------	-------	-------	-------

3 11 2 1

Coverage information:

The data cover 75 Alu (aggregated by mean: min # of CpGs: 2)

Gene coverage by Alu (aggregated by mean: min # of CpGs: 2) (out of total refSeq Gene):

86 (0.34%) total genes;

71 (0.37%) protein-coding genes;

17 (0.23%) non-coding RNA genes.

Distribution of methylation value (beta value):

Min.	1st Qu.	Median	Mean	3rd Qu.
0.04713393	0.42243309	0.66087749	0.59936950	0.83849455
Max.				
0.92886423				

5 Session Information

R version 3.4.2 (2017-09-28)

Platform: x86_64-apple-darwin15.6.0 (64-bit)

Running under: OS X El Capitan 10.11.6

Matrix products: default

BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib

LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib

locale:

[1] C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:

[1] parallel stats4 stats graphics grDevices
[6] utils datasets methods base

other attached packages:

[1] REMP_1.2.5
[2] IlluminaHumanMethylationEPICanno.ilm10b2.hg19_0.6.0
[3] IlluminaHumanMethylation450kanno.ilmn12.hg19_0.6.0
[4] minfi_1.24.0
[5] bumpHunter_1.20.0
[6] locfit_1.5-9.1
[7] iterators_1.0.8
[8] foreach_1.4.3
[9] Biostrings_2.46.0
[10] XVector_0.18.0
[11] SummarizedExperiment_1.8.0
[12] DelayedArray_0.4.1
[13] matrixStats_0.52.2
[14] Biobase_2.38.0
[15] GenomicRanges_1.30.0
[16] GenomeInfoDb_1.14.0
[17] IRanges_2.12.0
[18] S4Vectors_0.16.0
[19] BiocGenerics_0.24.0
[20] knitr_1.17

loaded via a namespace (and not attached):

- [1] AnnotationHub_2.10.1
- [2] plyr_1.8.4
- [3] lazyeval_0.2.1
- [4] splines_3.4.2
- [5] BiocParallel_1.12.0
- [6] ggplot2_2.2.1
- [7] digest_0.6.12
- [8] BiocInstaller_1.28.0
- [9] htmltools_0.3.6
- [10] magrittr_1.5
- [11] memoise_1.1.0
- [12] BSgenome_1.46.0
- [13] doParallel_1.0.11
- [14] sfsmisc_1.1-1
- [15] limma_3.34.2
- [16] recipes_0.1.1
- [17] readr_1.1.1
- [18] annotate_1.56.1
- [19] gower_0.1.2
- [20] dimRed_0.1.0
- [21] siggenes_1.52.0
- [22] prettyunits_1.0.2
- [23] colorspace_1.3-2
- [24] blob_1.1.0
- [25] dplyr_0.7.4
- [26] settings_0.2.4
- [27] RCurl_1.95-4.8
- [28] genefilter_1.60.0
- [29] impute_1.52.0
- [30] bindr_0.1
- [31] GEOquery_2.46.10
- [32] survival_2.41-3
- [33] glue_1.2.0
- [34] DRR_0.0.2
- [35] registry_0.3
- [36] gtable_0.2.0
- [37] ipred_0.9-6
- [38] zlibbioc_1.24.0
- [39] kernlab_0.9-25
- [40] ddalpha_1.3.1
- [41] DEoptimR_1.0-8
- [42] scales_0.5.0
- [43] DBI_0.7
- [44] rngtools_1.2.4
- [45] Rcpp_0.12.14
- [46] xtable_1.8-2
- [47] progress_1.1.2
- [48] foreign_0.8-69
- [49] bit_1.1-12
- [50] mclust_5.4
- [51] preprocessCore_1.40.0
- [52] lava_1.5.1

[53] prodlim_1.6.1
[54] httr_1.3.1
[55] RColorBrewer_1.1-2
[56] pkgconfig_2.0.1
[57] reshape_0.8.7
[58] XML_3.98-1.9
[59] nnet_7.3-12
[60] caret_6.0-77
[61] tidymodels_0.2.3
[62] rlang_0.1.4
[63] reshape2_1.4.2
[64] AnnotationDbi_1.40.0
[65] munsell_0.4.3
[66] tools_3.4.2
[67] RSQLite_2.0
[68] ranger_0.8.0
[69] broom_0.4.3
[70] evaluate_0.10.1
[71] stringr_1.2.0
[72] yaml_2.1.15
[73] org.Hs.eg.db_3.5.0
[74] ModelMetrics_1.1.0
[75] bit64_0.9-7
[76] beanplot_1.2
[77] robustbase_0.92-8
[78] purrr_0.2.4
[79] bindrcpp_0.2
[80] nlme_3.1-131
[81] doRNG_1.6.6
[82] mime_0.5
[83] nor1mix_1.2-3
[84] RcppRoll_0.2.2
[85] xml2_1.1.1
[86] biomaRt_2.34.0
[87] compiler_3.4.2
[88] curl_3.0
[89] interactiveDisplayBase_1.16.0
[90] tibble_1.3.4
[91] stringi_1.1.6
[92] GenomicFeatures_1.30.0
[93] lattice_0.20-35
[94] Matrix_1.2-12
[95] psych_1.7.8
[96] multtest_2.34.0
[97] data.table_1.10.4-3
[98] bitops_1.0-6
[99] httpuv_1.3.5
[100] rtracklayer_1.38.0
[101] R6_2.2.2
[102] RMySQL_0.10.13
[103] codetools_0.2-15
[104] MASS_7.3-47
[105] assertthat_0.2.0
[106] CVST_0.2-1

[107] openssl_0.9.9
[108] pkgmaker_0.22
[109] withr_2.1.0
[110] GenomicAlignments_1.14.1
[111] Rsamtools_1.30.0
[112] mnormt_1.5-5
[113] GenomeInfoDbData_0.99.1
[114] hms_0.4.0
[115] quadprog_1.5-5
[116] grid_3.4.2
[117] rpart_4.1-11
[118] timeDate_3042.101
[119] tidyr_0.7.2
[120] base64_2.0
[121] class_7.3-14
[122] BSgenome.Hsapiens.UCSC.hg19_1.4.0
[123] illuminaio_0.20.0
[124] shiny_1.0.5
[125] lubridate_1.7.1