DMRcate for bisulfite sequencing

Peters TJ

December 19, 2024

Summary

Worked example to find DMRs from whole genome bisulfite sequencing data.

```
if (!require("BiocManager"))
        install.packages("BiocManager")
BiocManager::install("DMRcate")
```

Load DMRcate into the workspace:

```
library(DMRcate)
```

Bisulfite sequencing assays are fundamentally different to arrays, because methylation is represented as a pair of methylated and unmethylated reads per sample, instead of a single beta value. Although we could simply take the logit-transformed fraction of methylated reads per CpG, this removes the effect of varying read depth across the genome. For example, a sampling depth of 30 methylated reads and 10 unmethylated reads is a much more precise estimate of the methylation level of a given CpG site than 3 methylated and 1 unmethylated. Hence, we take advantage of the fact that the overall effect can be expressed as an interaction between the coefficient of interest and a two-level factor representing methylated and unmethylated reads [1].

The example shown here will be performed on a BSseq object containing bisulfite sequencing of regulatory T cells from various tissues as part of the tissueTreg package[2], imported using ExperimentHub. First, we will import the data:

```
library(ExperimentHub)
eh <- ExperimentHub()
bis_1072 <- eh[["EH1072"]]
bis_1072

## An object of type 'BSseq' with
##   21867550 methylation loci
```

```
##   15 samples
## has been smoothed with
##   BSmooth (ns = 70, h = 1000, maxGap = 100000000)
## All assays are in-memory

colnames(bis_1072)

##  [1] "Fat-Treg-R1"     "Fat-Treg-R2"     "Fat-Treg-R3"     "Liver-Treg-R1"
##  [5] "Liver-Treg-R2"   "Liver-Treg-R3"   "Skin-Treg-R1"    "Skin-Treg-R2"
##  [9] "Skin-Treg-R3"    "Lymph-N-Tcon-R1" "Lymph-N-Tcon-R2" "Lymph-N-Tcon-R3"
## [13] "Lymph-N-Treg-R1" "Lymph-N-Treg-R2" "Lymph-N-Treg-R3"
```

The data contains 15 samples: 3 (unmatched) replicates of mouse Tregs from fat, liver, skin and lymph node, plus a group of 3 CD4+ conventional lymph node T cells (Tcon). We will annotate the BSseq object to reflect this phenotypic information:

```
bsseq::pData(bis_1072) <- data.frame(replicate=gsub(".*-", "", colnames(bis_1072)),
                            tissue=substr(colnames(bis_1072), 1,
                                          nchar(colnames(bis_1072))-3),
                            row.names=colnames(bis_1072))
colData(bis_1072)$tissue <- gsub("-", "_", colData(bis_1072)$tissue)
as.data.frame(colData(bis_1072))

##                  replicate       tissue
## Fat-Treg-R1             R1     Fat_Treg
## Fat-Treg-R2             R2     Fat_Treg
## Fat-Treg-R3             R3     Fat_Treg
## Liver-Treg-R1           R1   Liver_Treg
## Liver-Treg-R2           R2   Liver_Treg
## Liver-Treg-R3           R3   Liver_Treg
## Skin-Treg-R1            R1    Skin_Treg
## Skin-Treg-R2            R2    Skin_Treg
## Skin-Treg-R3            R3    Skin_Treg
## Lymph-N-Tcon-R1         R1 Lymph_N_Tcon
## Lymph-N-Tcon-R2         R2 Lymph_N_Tcon
## Lymph-N-Tcon-R3         R3 Lymph_N_Tcon
## Lymph-N-Treg-R1         R1 Lymph_N_Treg
## Lymph-N-Treg-R2         R2 Lymph_N_Treg
## Lymph-N-Treg-R3         R3 Lymph_N_Treg
```

For standardisation purposes (and for `DMR.plot` to recognise the genome) we will change the chromosome naming convention to UCSC:

```
bis_1072 <- renameSeqlevels(bis_1072, mapSeqlevels(seqlevels(bis_1072), "UCSC"))
```

For demonstration purposes, we will retain CpGs on chromosome 19 only:

```
bis_1072 <- bis_1072[seqnames(bis_1072)=="chr19",]
bis_1072

## An object of type 'BSseq' with
##   558056 methylation loci
##   15 samples
## has been smoothed with
##   BSmooth (ns = 70, h = 1000, maxGap = 100000000)
## All assays are in-memory
```

Now we can prepare the model to be fit for `sequencing.annotate()`. The arguments are equivalent to `cpg.annotate()` but for a couple of exceptions:

- There is an extra argument `all.cov` giving an option whether to retain only CpGs where *all* samples have non-zero coverage, or whether to retain CpGs with only partial sample representation.
- The design matrix should be constructed to reflect the 2-factor structure of methylated and unmethylated reads. Fortunately, `edgeR::modelMatrixMeth()` can take a regular design matrix and transform is into the appropriate structure ready for model fitting.

```
tissue <- factor(pData(bis_1072)$tissue)
tissue <- relevel(tissue, "Liver_Treg")

#Regular matrix design
design <- model.matrix(~tissue)
colnames(design) <- gsub("tissue", "", colnames(design))
colnames(design)[1] <- "Intercept"
rownames(design) <- colnames(bis_1072)
design

##                 Intercept Fat_Treg Lymph_N_Tcon Lymph_N_Treg Skin_Treg
## Fat-Treg-R1             1        1            0            0         0
## Fat-Treg-R2             1        1            0            0         0
## Fat-Treg-R3             1        1            0            0         0
## Liver-Treg-R1           1        0            0            0         0
## Liver-Treg-R2           1        0            0            0         0
## Liver-Treg-R3           1        0            0            0         0
## Skin-Treg-R1            1        0            0            0         1
## Skin-Treg-R2            1        0            0            0         1
## Skin-Treg-R3            1        0            0            0         1
```

```
## Lymph-N-Tcon-R1         1        0            1            0        0
## Lymph-N-Tcon-R2         1        0            1            0        0
## Lymph-N-Tcon-R3         1        0            1            0        0
## Lymph-N-Treg-R1         1        0            0            1        0
## Lymph-N-Treg-R2         1        0            0            1        0
## Lymph-N-Treg-R3         1        0            0            1        0
## attr(,"assign")
## [1] 0 1 1 1 1
## attr(,"contrasts")
## attr(,"contrasts")$tissue
## [1] "contr.treatment"
```

```
#Methylation matrix design
methdesign <- edgeR::modelMatrixMeth(design)
methdesign
```

```
##    Sample1 Sample2 Sample3 Sample4 Sample5 Sample6 Sample7 Sample8 Sample9
## 1        1       0       0       0       0       0       0       0       0
## 2        1       0       0       0       0       0       0       0       0
## 3        0       1       0       0       0       0       0       0       0
## 4        0       1       0       0       0       0       0       0       0
## 5        0       0       1       0       0       0       0       0       0
## 6        0       0       1       0       0       0       0       0       0
## 7        0       0       0       1       0       0       0       0       0
## 8        0       0       0       1       0       0       0       0       0
## 9        0       0       0       0       1       0       0       0       0
## 10       0       0       0       0       1       0       0       0       0
## 11       0       0       0       0       0       1       0       0       0
## 12       0       0       0       0       0       1       0       0       0
## 13       0       0       0       0       0       0       1       0       0
## 14       0       0       0       0       0       0       1       0       0
## 15       0       0       0       0       0       0       0       1       0
## 16       0       0       0       0       0       0       0       1       0
## 17       0       0       0       0       0       0       0       0       1
## 18       0       0       0       0       0       0       0       0       1
## 19       0       0       0       0       0       0       0       0       0
## 20       0       0       0       0       0       0       0       0       0
## 21       0       0       0       0       0       0       0       0       0
## 22       0       0       0       0       0       0       0       0       0
## 23       0       0       0       0       0       0       0       0       0
## 24       0       0       0       0       0       0       0       0       0
## 25       0       0       0       0       0       0       0       0       0
## 26       0       0       0       0       0       0       0       0       0
## 27       0       0       0       0       0       0       0       0       0
## 28       0       0       0       0       0       0       0       0       0
## 29       0       0       0       0       0       0       0       0       0
```

```
## 30       0       0       0       0       0       0       0       0       0
##    Sample10 Sample11 Sample12 Sample13 Sample14 Sample15 Intercept Fat_Treg
## 1         0        0        0        0        0        0         1        1
## 2         0        0        0        0        0        0         0        0
## 3         0        0        0        0        0        0         1        1
## 4         0        0        0        0        0        0         0        0
## 5         0        0        0        0        0        0         1        1
## 6         0        0        0        0        0        0         0        0
## 7         0        0        0        0        0        0         1        0
## 8         0        0        0        0        0        0         0        0
## 9         0        0        0        0        0        0         1        0
## 10        0        0        0        0        0        0         0        0
## 11        0        0        0        0        0        0         1        0
## 12        0        0        0        0        0        0         0        0
## 13        0        0        0        0        0        0         1        0
## 14        0        0        0        0        0        0         0        0
## 15        0        0        0        0        0        0         1        0
## 16        0        0        0        0        0        0         0        0
## 17        0        0        0        0        0        0         1        0
## 18        0        0        0        0        0        0         0        0
## 19        1        0        0        0        0        0         1        0
## 20        1        0        0        0        0        0         0        0
## 21        0        1        0        0        0        0         1        0
## 22        0        1        0        0        0        0         0        0
## 23        0        0        1        0        0        0         1        0
## 24        0        0        1        0        0        0         0        0
## 25        0        0        0        1        0        0         1        0
## 26        0        0        0        1        0        0         0        0
## 27        0        0        0        0        1        0         1        0
## 28        0        0        0        0        1        0         0        0
## 29        0        0        0        0        0        1         1        0
## 30        0        0        0        0        0        1         0        0
##    Lymph_N_Tcon Lymph_N_Treg Skin_Treg
## 1             0            0         0
## 2             0            0         0
## 3             0            0         0
## 4             0            0         0
## 5             0            0         0
## 6             0            0         0
## 7             0            0         0
## 8             0            0         0
## 9             0            0         0
## 10            0            0         0
## 11            0            0         0
## 12            0            0         0
```

```
## 13            0            0         1
## 14            0            0         0
## 15            0            0         1
## 16            0            0         0
## 17            0            0         1
## 18            0            0         0
## 19            1            0         0
## 20            0            0         0
## 21            1            0         0
## 22            0            0         0
## 23            1            0         0
## 24            0            0         0
## 25            0            1         0
## 26            0            0         0
## 27            0            1         0
## 28            0            0         0
## 29            0            1         0
## 30            0            0         0
```

Just like for `cpg.annotate()`, we can specify a contrast matrix to find our comparisons of interest.

```
cont.mat <- limma::makeContrasts(treg_vs_tcon=Lymph_N_Treg-Lymph_N_Tcon,
                                 fat_vs_ln=Fat_Treg-Lymph_N_Treg,
                                 skin_vs_ln=Skin_Treg-Lymph_N_Treg,
                                 fat_vs_skin=Fat_Treg-Skin_Treg,
                                 levels=methdesign)
cont.mat

##               Contrasts
## Levels         treg_vs_tcon fat_vs_ln skin_vs_ln fat_vs_skin
##   Sample1                 0         0          0           0
##   Sample2                 0         0          0           0
##   Sample3                 0         0          0           0
##   Sample4                 0         0          0           0
##   Sample5                 0         0          0           0
##   Sample6                 0         0          0           0
##   Sample7                 0         0          0           0
##   Sample8                 0         0          0           0
##   Sample9                 0         0          0           0
##   Sample10                0         0          0           0
##   Sample11                0         0          0           0
##   Sample12                0         0          0           0
##   Sample13                0         0          0           0
##   Sample14                0         0          0           0
##   Sample15                0         0          0           0
```

```
##   Intercept              0         0         0         0
##   Fat_Treg               0         1         0         1
##   Lymph_N_Tcon          -1         0         0         0
##   Lymph_N_Treg           1        -1        -1         0
##   Skin_Treg              0         0         1        -1
```

Say we want to find DMRs between the regulatory and conventional T cells from the lymph node. First we would fit the model, where `sequencing.annotate()` transforms counts into log2CPMs (via `limma::voom()`) and uses `limma` under the hood to generate per-CpG t-statistics, indexing the FDR at 0.05:

```
seq_annot <- sequencing.annotate(bis_1072, methdesign, all.cov = TRUE,
                                 contrasts = TRUE, cont.matrix = cont.mat,
                                 coef = "treg_vs_tcon", fdr=0.05)

## Filtering out all CpGs where at least one sample has zero coverage...
## Processing BSseq object...
## Transforming counts...
## Fitting model...
## Your contrast returned 157 individually significant CpGs.  We recommend
the default setting of pcutoff in dmrcate().

seq_annot

## CpGannotated object describing 506908 CpG sites, with independent
## CpG threshold indexed at fdr=0.05 and 157 significant CpG sites.
```

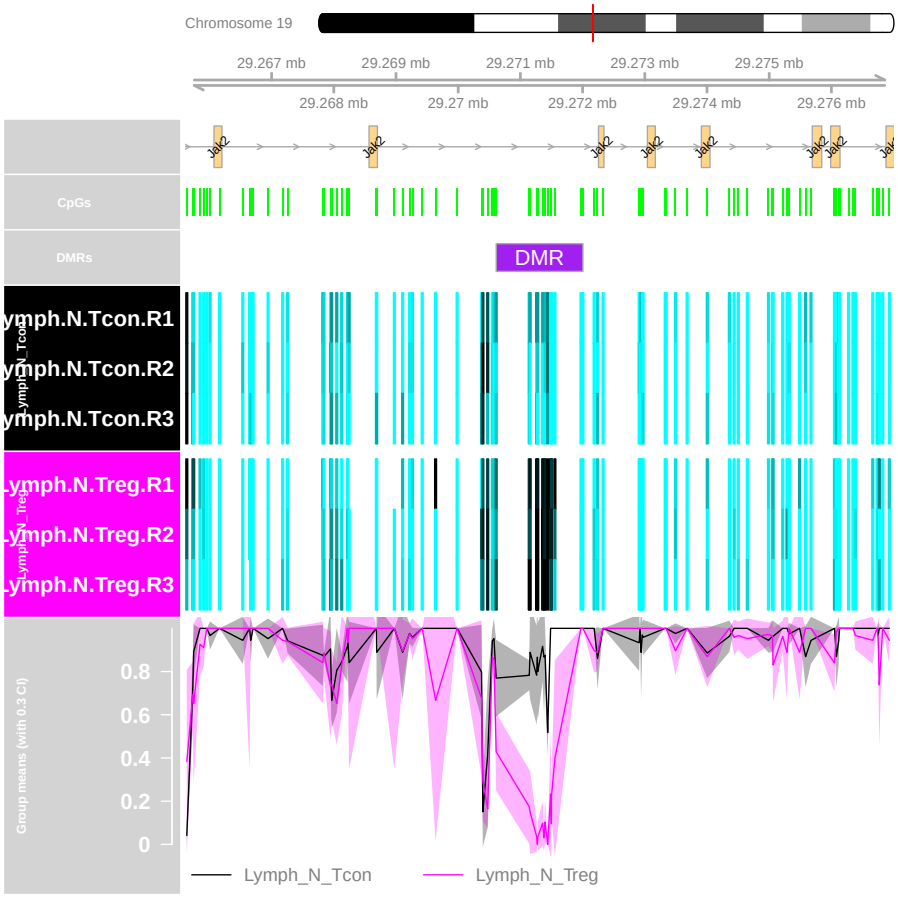
And then, just like before, we can call DMRs with `dmrcate()`:

```
dmrcate.res <- dmrcate(seq_annot, C=2, min.cpgs = 5)

## Fitting chr19...
## Demarcating regions...
## Done!

dmrcate.res

## DMResults object with 9 DMRs.
## Use extractRanges() to produce a GRanges object of these.

treg_vs_tcon.ranges <- extractRanges(dmrcate.res, genome="mm10")

## see ?DMRcatedata and browseVignettes('DMRcatedata') for documentation
## loading from cache

treg_vs_tcon.ranges
```

```
## GRanges object with 9 ranges and 8 metadata columns:
##       seqnames            ranges strand |   no.cpgs min_smoothed_fdr
##          <Rle>         <IRanges>  <Rle> | <integer>        <numeric>
##   [1]    chr19 29270611-29272005      * |        16      4.32351e-94
##   [2]    chr19 36378257-36379597      * |        27      1.53747e-76
##   [3]    chr19 40808208-40809554      * |        26      3.43873e-63
##   [4]    chr19 46653280-46654180      * |        19      3.94008e-59
##   [5]    chr19 26683453-26684174      * |        12      1.77927e-57
##   [6]    chr19 32276886-32278089      * |        13      1.74620e-56
##   [7]    chr19 29374953-29375393      * |        12      1.48257e-54
##   [8]    chr19 41874401-41874895      * |        22      2.75829e-39
##   [9]    chr19 57092365-57092646      * |        10      3.80468e-36
##          Stouffer       HMFDR      Fisher   maxdiff  meandiff overlapping.genes
##         <numeric>   <numeric>   <numeric> <numeric> <numeric>       <character>
##   [1] 7.30413e-50 1.53438e-06 9.06017e-52  -6.40482  -4.22353              Jak2
##   [2] 5.11228e-45 5.32893e-06 6.32480e-49  -6.09625  -3.03550             Pcgf5
##   [3] 1.57979e-50 7.52023e-05 1.44219e-45  -4.83855  -3.07494            Cc2d2b
##   [4] 6.97298e-34 5.32893e-06 1.28153e-35   5.18388   2.93152             Wbp1l
##   [5] 3.23045e-25 3.29036e-07 1.30324e-30  -6.40328  -3.53692           Smarca2
##   [6] 2.50713e-38 5.32893e-06 1.55805e-36   5.81470   3.93201             Sgms1
##   [7] 7.68507e-20 3.13417e-06 3.83714e-26  -6.10902  -3.02083    Cd274, Gm36043
##   [8] 1.15140e-30 6.23899e-05 7.96953e-30   4.57011   2.56520             Rrp12
##   [9] 1.28763e-24 7.32179e-06 6.36706e-23  -4.67645  -3.36472            Ablim1
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Looks like the top DMR is associated with the *Jak2* locus and hypomethylated in the Treg cells (since `meandiff < 0`). We can plot it like so:

```
cols <- as.character(plyr::mapvalues(tissue, unique(tissue),
                                     c("darkorange", "maroon", "blue",
                                       "black", "magenta")))
names(cols) <- tissue

DMR.plot(treg_vs_tcon.ranges, dmr = 1,
         CpGs=bis_1072[,tissue %in% c("Lymph_N_Tcon", "Lymph_N_Treg")],
         phen.col = cols[tissue %in% c("Lymph_N_Tcon", "Lymph_N_Treg")],
         genome="mm10")
```



Now, let's find DMRs between fat and skin Tregs.

```
seq_annot <- sequencing.annotate(bis_1072, methdesign, all.cov = TRUE,
                                 contrasts = TRUE, cont.matrix = cont.mat,
                                 coef = "fat_vs_skin", fdr=0.05)

## Filtering out all CpGs where at least one sample has zero coverage...
## Processing BSseq object...
## Transforming counts...
## Fitting model...
## Your contrast returned 5 individually significant CpGs; a small
but real effect.  Consider increasing the 'fdr' parameter using changeFDR(),
but be warned there is an increased risk of Type I errors.
```

Because this comparison is a bit more subtle, there are very few significantly differential CpGs at this threshold. So we can use `changeFDR()` to relax the

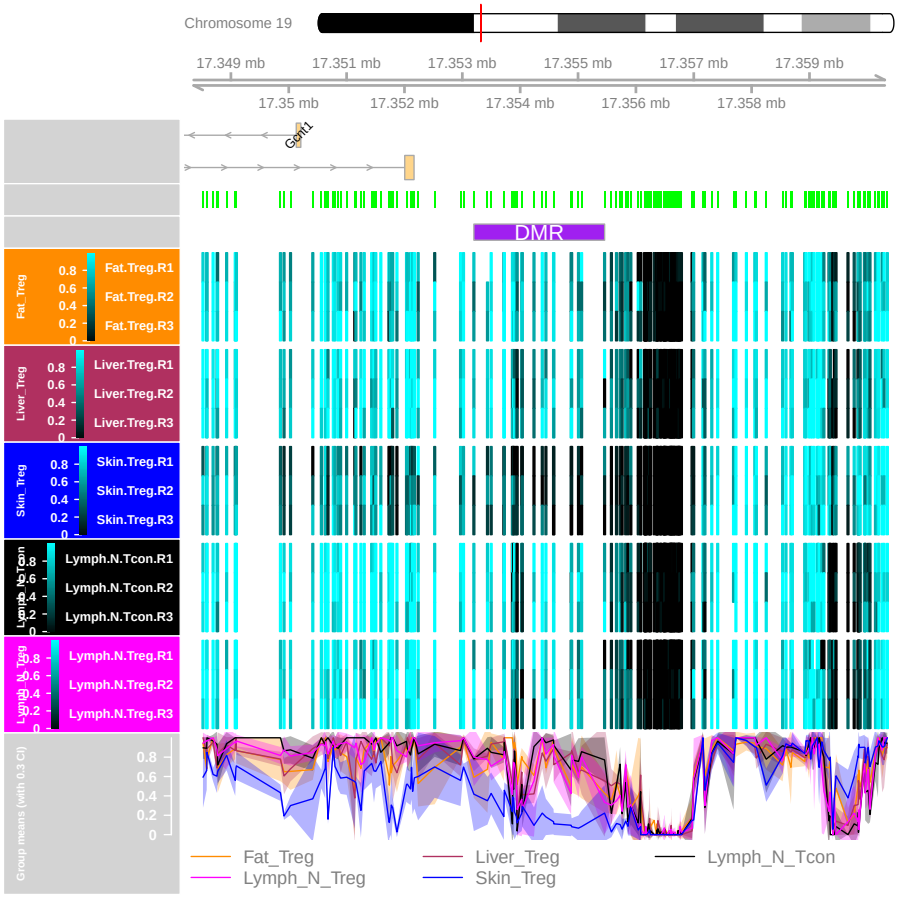
FDR to 0.25, taking into account that there is an increased risk of false positives.

```
seq_annot <- changeFDR(seq_annot, 0.25)

## Threshold is now set at FDR=0.25, resulting in 63 significantly differential CpGs.
```

```
dmrcate.res <- dmrcate(seq_annot, C=2, min.cpgs = 5)

## Fitting chr19...
## Demarcating regions...
## Done!

fat_vs_skin.ranges <- extractRanges(dmrcate.res, genome="mm10")

## see ?DMRcatedata and browseVignettes('DMRcatedata') for documentation
## loading from cache
```

Now let's plot the top DMR with not only fat and skin, but with all samples:

```
cols

##     Fat_Treg     Fat_Treg     Fat_Treg   Liver_Treg   Liver_Treg   Liver_Treg
## "darkorange" "darkorange" "darkorange"     "maroon"     "maroon"     "maroon"
##    Skin_Treg    Skin_Treg    Skin_Treg Lymph_N_Tcon Lymph_N_Tcon Lymph_N_Tcon
##       "blue"       "blue"       "blue"      "black"      "black"      "black"
## Lymph_N_Treg Lymph_N_Treg Lymph_N_Treg
##    "magenta"    "magenta"    "magenta"

DMR.plot(fat_vs_skin.ranges, dmr = 1, CpGs=bis_1072, phen.col = cols, genome="mm10")
```



Here we can see the methylation of skin cells over this region near the *Gcnt1* promoter is hypomethylated not only relative to fat, but to the other tissues as well.

```
sessionInfo()

## R version 4.4.2 (2024-10-31)
## Platform: x86_64-pc-linux-gnu
## Running under: Ubuntu 24.04.1 LTS
##
## Matrix products: default
## BLAS:   /home/biocbuild/bbs-3.20-bioc/R/lib/libRblas.so
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.12.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
```

```
##  [3] LC_TIME=en_GB              LC_COLLATE=C
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## time zone: America/New_York
## tzcode source: system (glibc)
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] bsseq_1.42.0
##  [2] tissueTreg_1.26.0
##  [3] IlluminaHumanMethylationEPICv2anno.20a1.hg38_1.0.0
##  [4] DMRcatedata_2.24.0
##  [5] IlluminaHumanMethylationEPICanno.ilm10b4.hg19_0.6.0
##  [6] IlluminaHumanMethylationEPICmanifest_0.3.0
##  [7] FlowSorted.Blood.EPIC_2.10.0
##  [8] minfi_1.52.1
##  [9] bumphunter_1.48.0
## [10] locfit_1.5-9.10
## [11] iterators_1.0.14
## [12] foreach_1.5.2
## [13] Biostrings_2.74.1
## [14] XVector_0.46.0
## [15] SummarizedExperiment_1.36.0
## [16] Biobase_2.66.0
## [17] MatrixGenerics_1.18.0
## [18] matrixStats_1.4.1
## [19] GenomicRanges_1.58.0
## [20] GenomeInfoDb_1.42.1
## [21] IRanges_2.40.1
## [22] S4Vectors_0.44.0
## [23] ExperimentHub_2.14.0
## [24] AnnotationHub_3.14.0
## [25] BiocFileCache_2.14.0
## [26] dbplyr_2.5.0
## [27] BiocGenerics_0.52.0
## [28] DMRcate_3.2.1
##
## loaded via a namespace (and not attached):
##   [1] splines_4.4.2
```

```
##   [2] BiocIO_1.16.0
##   [3] bitops_1.0-9
##   [4] filelock_1.0.3
##   [5] cellranger_1.1.0
##   [6] tibble_3.2.1
##   [7] R.oo_1.27.0
##   [8] preprocessCore_1.68.0
##   [9] XML_3.99-0.17
##  [10] rpart_4.1.23
##  [11] lifecycle_1.0.4
##  [12] httr2_1.0.7
##  [13] edgeR_4.4.1
##  [14] base64_2.0.2
##  [15] lattice_0.22-6
##  [16] ensembldb_2.30.0
##  [17] MASS_7.3-61
##  [18] scrime_1.3.5
##  [19] backports_1.5.0
##  [20] magrittr_2.0.3
##  [21] limma_3.62.1
##  [22] Hmisc_5.2-1
##  [23] rmarkdown_2.29
##  [24] yaml_2.3.10
##  [25] doRNG_1.8.6
##  [26] askpass_1.2.1
##  [27] Gviz_1.50.0
##  [28] DBI_1.2.3
##  [29] RColorBrewer_1.1-3
##  [30] abind_1.4-8
##  [31] zlibbioc_1.52.0
##  [32] quadprog_1.5-8
##  [33] purrr_1.0.2
##  [34] R.utils_2.12.3
##  [35] AnnotationFilter_1.30.0
##  [36] biovizBase_1.54.0
##  [37] RCurl_1.98-1.16
##  [38] nnet_7.3-19
##  [39] VariantAnnotation_1.52.0
##  [40] rappdirs_0.3.3
##  [41] GenomeInfoDbData_1.2.13
##  [42] rentrez_1.2.3
##  [43] genefilter_1.88.0
##  [44] annotate_1.84.0
##  [45] permute_0.9-7
##  [46] DelayedMatrixStats_1.28.0
```

```
##  [47] codetools_0.2-20
##  [48] DelayedArray_0.32.0
##  [49] xml2_1.3.6
##  [50] tidyselect_1.2.1
##  [51] UCSC.utils_1.2.0
##  [52] beanplot_1.3.1
##  [53] base64enc_0.1-3
##  [54] illuminaio_0.48.0
##  [55] GenomicAlignments_1.42.0
##  [56] jsonlite_1.8.9
##  [57] multtest_2.62.0
##  [58] Formula_1.2-5
##  [59] survival_3.8-3
##  [60] missMethyl_1.40.0
##  [61] tools_4.4.2
##  [62] progress_1.2.3
##  [63] Rcpp_1.0.13-1
##  [64] glue_1.8.0
##  [65] gridExtra_2.3
##  [66] SparseArray_1.6.0
##  [67] xfun_0.49
##  [68] dplyr_1.1.4
##  [69] HDF5Array_1.34.0
##  [70] withr_3.0.2
##  [71] IlluminaHumanMethylation450kanno.ilmn12.hg19_0.6.1
##  [72] BiocManager_1.30.25
##  [73] fastmap_1.2.0
##  [74] latticeExtra_0.6-30
##  [75] rhdf5filters_1.18.0
##  [76] openssl_2.3.0
##  [77] digest_0.6.37
##  [78] mime_0.12
##  [79] R6_2.5.1
##  [80] colorspace_2.1-1
##  [81] gtools_3.9.5
##  [82] jpeg_0.1-10
##  [83] dichromat_2.0-0.1
##  [84] biomaRt_2.62.0
##  [85] RSQLite_2.3.9
##  [86] R.methodsS3_1.8.2
##  [87] tidyr_1.3.1
##  [88] generics_0.1.3
##  [89] data.table_1.16.4
##  [90] rtracklayer_1.66.0
##  [91] prettyunits_1.2.0
```

```
##  [92] httr_1.4.7
##  [93] htmlwidgets_1.6.4
##  [94] S4Arrays_1.6.0
##  [95] pkgconfig_2.0.3
##  [96] gtable_0.3.6
##  [97] blob_1.2.4
##  [98] siggenes_1.80.0
##  [99] htmltools_0.5.8.1
## [100] ProtGenerics_1.38.0
## [101] scales_1.3.0
## [102] png_0.1-8
## [103] knitr_1.49
## [104] rstudioapi_0.17.1
## [105] tzdb_0.4.0
## [106] rjson_0.2.23
## [107] nlme_3.1-166
## [108] checkmate_2.3.2
## [109] curl_6.0.1
## [110] org.Hs.eg.db_3.20.0
## [111] cachem_1.1.0
## [112] rhdf5_2.50.1
## [113] stringr_1.5.1
## [114] BiocVersion_3.20.0
## [115] foreign_0.8-87
## [116] AnnotationDbi_1.68.0
## [117] restfulr_0.0.15
## [118] GEOquery_2.74.0
## [119] pillar_1.10.0
## [120] grid_4.4.2
## [121] reshape_0.8.9
## [122] vctrs_0.6.5
## [123] xtable_1.8-4
## [124] cluster_2.1.8
## [125] htmlTable_2.4.3
## [126] evaluate_1.0.1
## [127] readr_2.1.5
## [128] GenomicFeatures_1.58.0
## [129] cli_3.6.3
## [130] compiler_4.4.2
## [131] Rsamtools_2.22.0
## [132] rngtools_1.5.2
## [133] rlang_1.1.4
## [134] crayon_1.5.3
## [135] nor1mix_1.3-3
## [136] mclust_6.1.1
```

```
## [137] interp_1.1-6
## [138] plyr_1.8.9
## [139] stringi_1.8.4
## [140] deldir_2.0-4
## [141] BiocParallel_1.40.0
## [142] munsell_0.5.1
## [143] lazyeval_0.2.2
## [144] Matrix_1.7-1
## [145] BSgenome_1.74.0
## [146] hms_1.1.3
## [147] sparseMatrixStats_1.18.0
## [148] bit64_4.5.2
## [149] ggplot2_3.5.1
## [150] Rhdf5lib_1.28.0
## [151] KEGGREST_1.46.0
## [152] statmod_1.5.0
## [153] highr_0.11
## [154] memoise_2.0.1
## [155] bit_4.5.0.1
## [156] readxl_1.4.3
```

References

[1] Chen Y, Pal B, Visvader JE, Smyth GK. Differential methylation analysis of reduced representation bisulfite sequencing experiments using edgeR. *F1000Research*. 2017 **6**, 2055.

[2] Delacher M, Imbusch CD, Weichenhan D, Breiling A, Hotz-Wagenblatt A, Trager U, ... Feuerer M. (2017). Genome-wide DNA-methylation landscape defines specialization of regulatory T cells in tissues. *Nature Immunology*. 2017 **18**(10), 1160-1172.