{epistack}

An R/Bioconductor package to visualise stack profiles of epigenomic signals

Safia Saci & Guillaume Devailly

R-Toulouse

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

INRAE

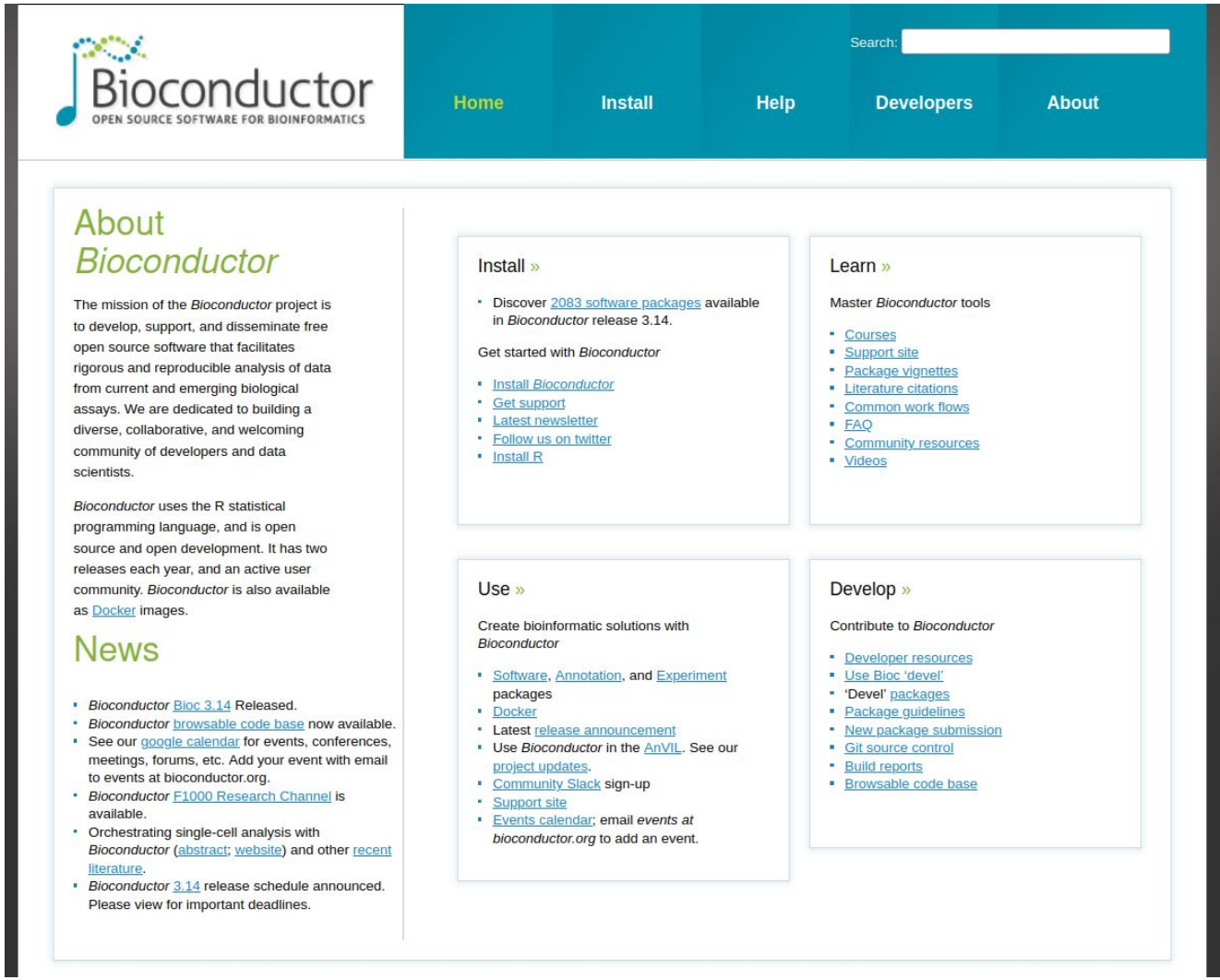
GenPhySE
Génétique, Physiologie et Systèmes d'Elevage

anr agence nationale de la recherche
AU SERVICE DE LA SCIENCE

@G_Devailly

Bioconductor www.bioconductor.org

- An alternative R package repository
- Dedicated to bioinformatics, the data science of biology.
- More opinionated than CRAN:
  - package reviewing
  - BiocCheck()

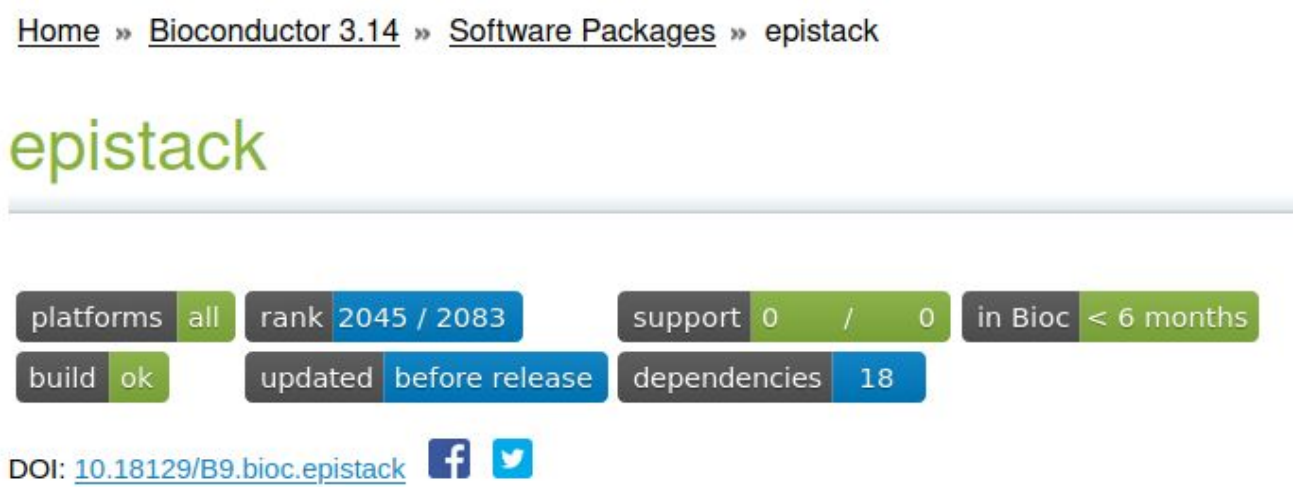


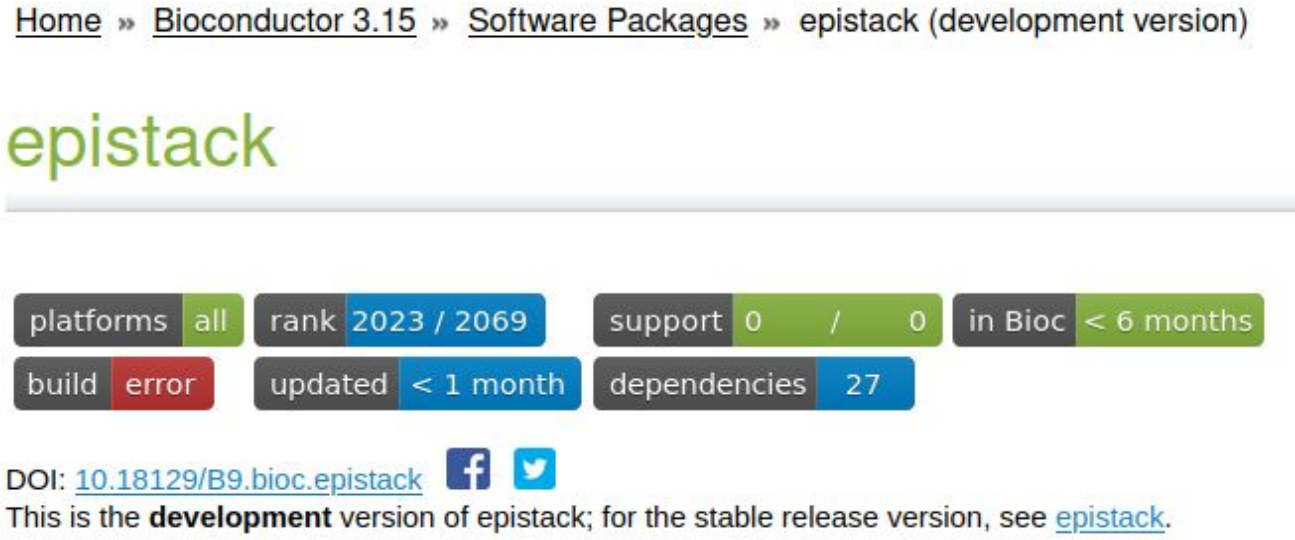
Bioconductor

- Release cycle every 6 months
- Everything should work in *release*
- Some things may be broken in *devel*
- Package are tested, vignettes are built, nightly
- Install packages using:

```
BiocManager::install("packagename")
```

- A bioconductor release will work only (mostly) with the latest version of R





Bioconductor in numbers

Release 3.14 (for R 4.1.0)

- 2.083 software packages
- 408 data packages
- 904 annotation packages

{DESeq2} in 2020: 370.000 downloads from 124.000 distinct IPs.

Bioconductor strength 1

A core set of methods / parser / classed to work with biological data

Common Bioconductor Methods and Classes

We strongly recommend reusing existing methods for importing data, and reusing established classes for representing data. Here are some suggestions for importing different file types and commonly used *Bioconductor* classes. For more classes and functionality also try searching in BiocViews for your data type.

Importing

- GTF, GFF, BED, BigWig, etc., – rtracklayer::import()
- VCF – VariantAnnotation::readVcf()
- SAM / BAM – Rsamtools::scanBam(), GenomicAlignments::readGAlignment*()
- FASTA – Biostrings::readDNAStringSet()
- FASTQ – ShortRead::readFastq()
- MS data (XML-based and mgf formats) – Spectra::Spectra(), MSnbase::readMSData(), Spectra::Spectra(source = MsBackendMgf::MsBackendMgf()), MSnbase::readMgfData()

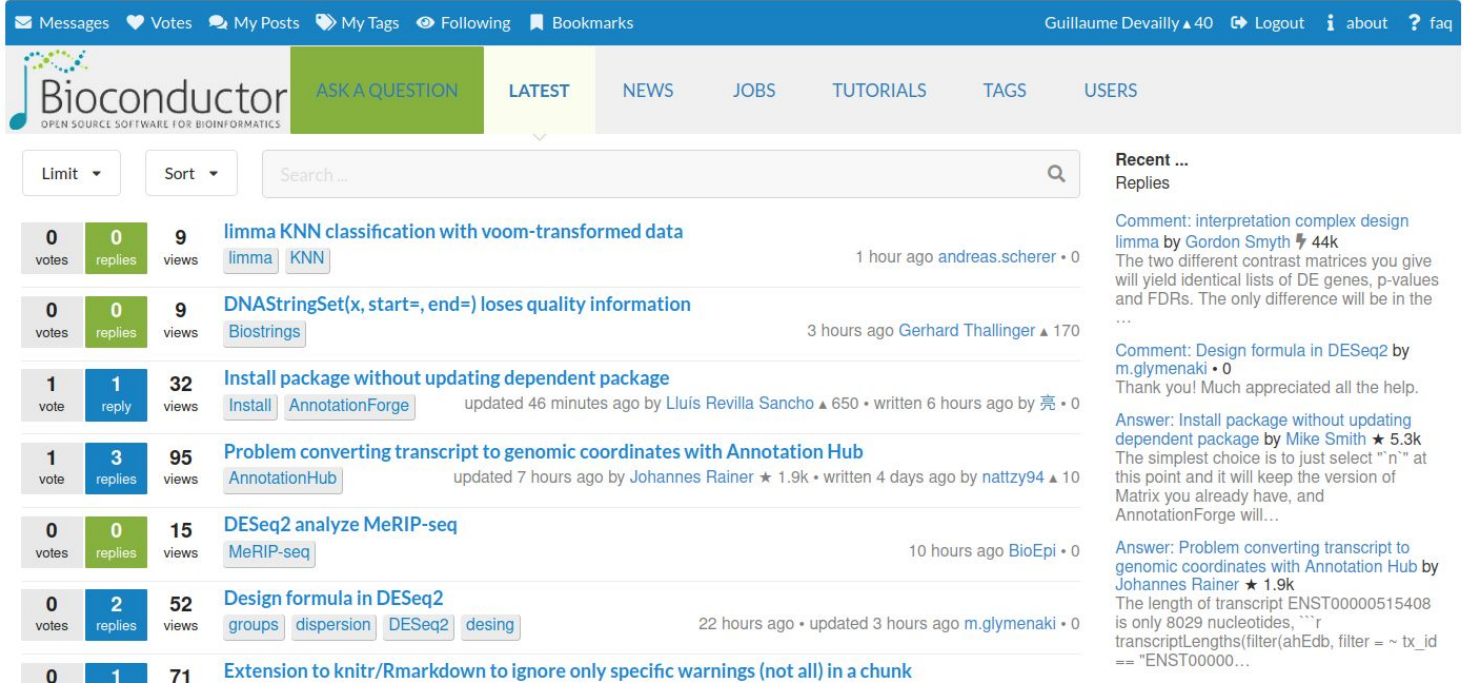
Common Classes

- Rectangular feature x sample data – SummarizedExperiment::SummarizedExperiment() (RNAseq count matrix, microarray, ...)
- Genomic coordinates – GenomicRanges::GRanges() (1-based, closed interval)
- Genomic coordinates from multiple samples – GenomicRanges::GRangesList()
- Ragged genomic coordinates – RaggedExperiment::RaggedExperiment()
- DNA / RNA / AA sequences – Biostrings::*StringSet()
- Gene sets – BiocSet::BiocSet(), GSEABase::GeneSet(), GSEABase::GeneSetCollection()
- Multi-omics data – MultiAssayExperiment::MultiAssayExperiment()
- Single cell data – SingleCellExperiment::SingleCellExperiment()
- Mass spec data – Spectra::Spectra(), MSnbase::MSnExp()

Bioconductor strength 2

Community:

- forum *à la* stackoverflow
- conferences & events
- F1000 Research gateway



This gateway highlights Bioconductor package-based vignettes and cross-package workflows.

How to submit a Bioconductor package?

Raise an issue at github.com/Bioconductor/Contributions

with a link to a GitHub repo of your package

How to submit a Bioconductor package?

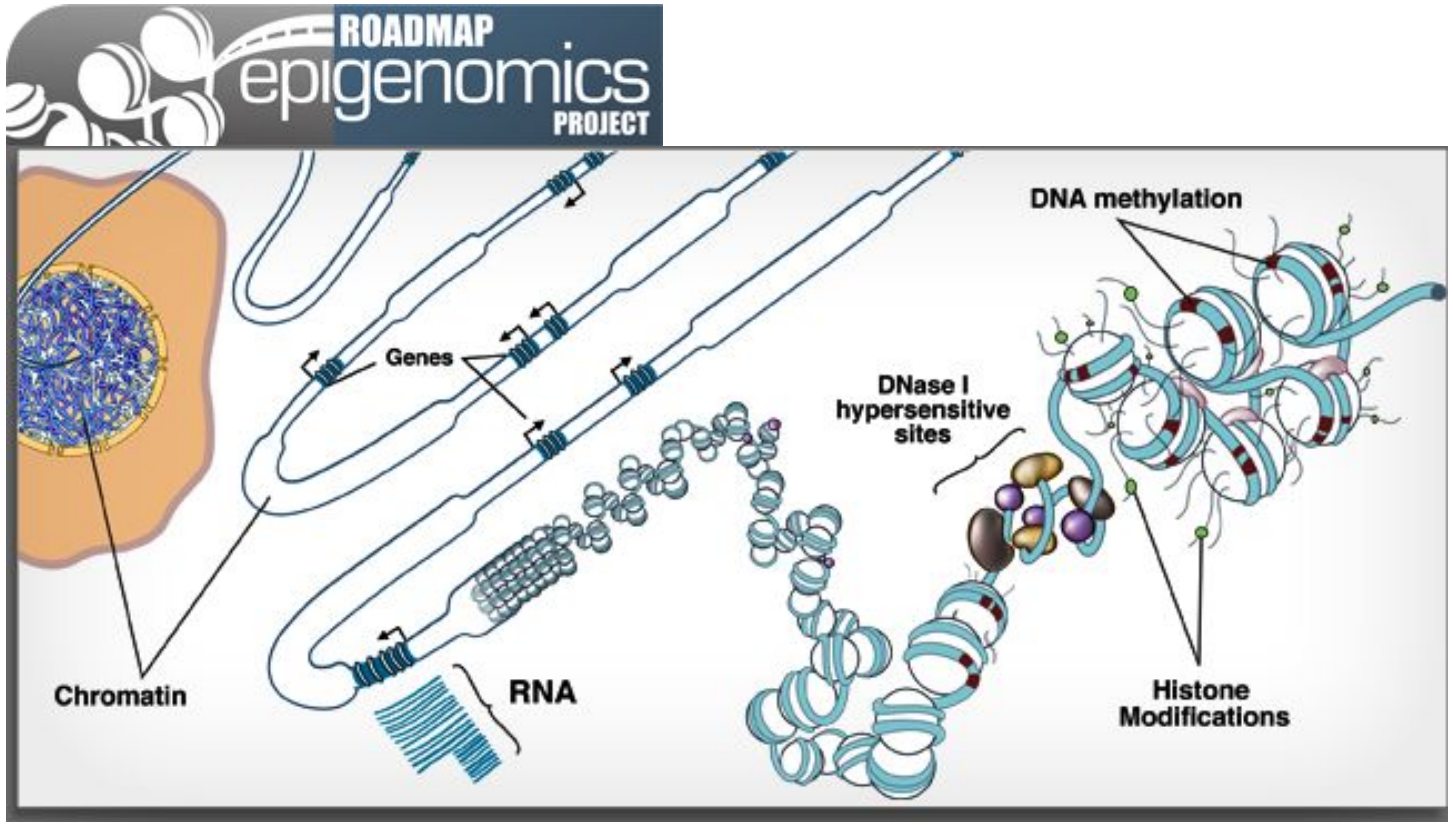
Bioconductor has expectations for your package:

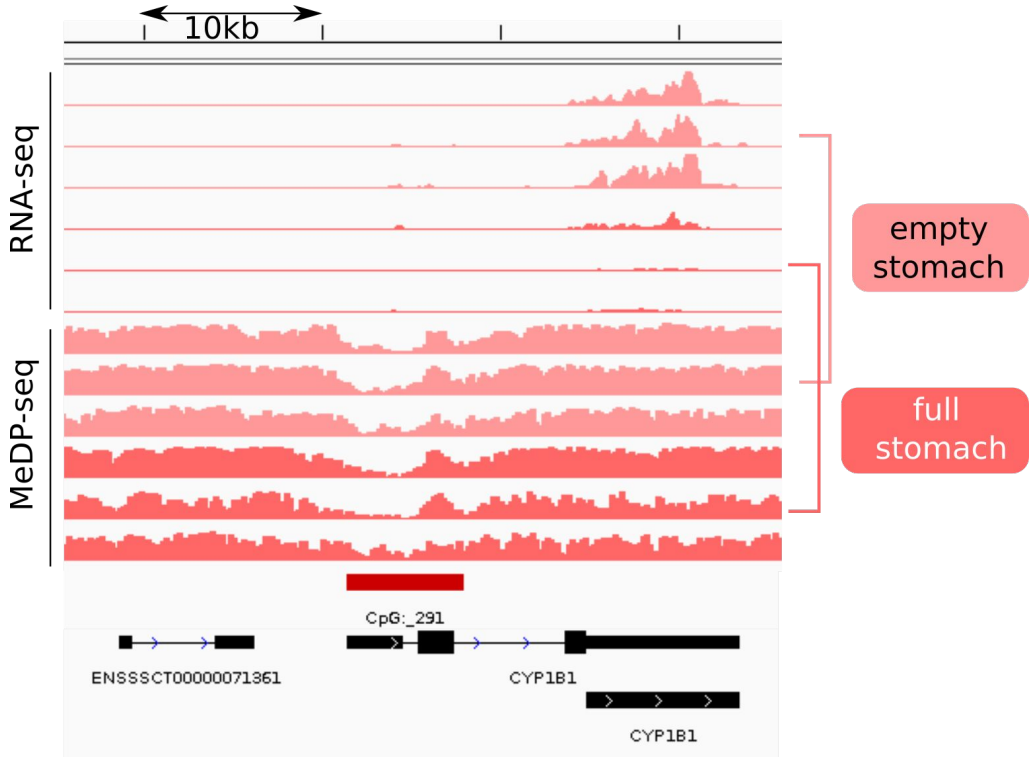
- Mandatory vignette
- Use Bioconductor recommended classes and methods
- 0 error, 0 warning, 0 note in R CMD check 😨
- 0 error, 0 warning, minimal number of notes in BiocCheck::BiocCheck() 😱
  - including mandatory subscription to a daily mailing list storing your password in clear 😤

```
$error
character(0)

$warning
[1] "y of x.y.z version should be even in release"

$note
[1] "Consider adding these automatically suggested biocViews: ChipOnChip"

[2] "The Description field in the DESCRIPTION is made up by less than 3 sentences. Please consider expanding this field, and\nstructure it a
s a full paragraph"
[3] "Recommended function length ⩽ 50 lines."

[4] "Consider shorter lines; 31 lines (1%) are > 80 characters long."

[5] "Consider multiples of 4 spaces for line indents, 184 lines(7%) are not."

[6] "Cannot determine whether maintainer is subscribed to the bioc-devel mailing list (requires admin credentials).\nSubscribe here: http
s://stat.ethz.ch/mailman/listinfo/bioc-devel"
```

Epigenomic data

Tracks of genomic scores





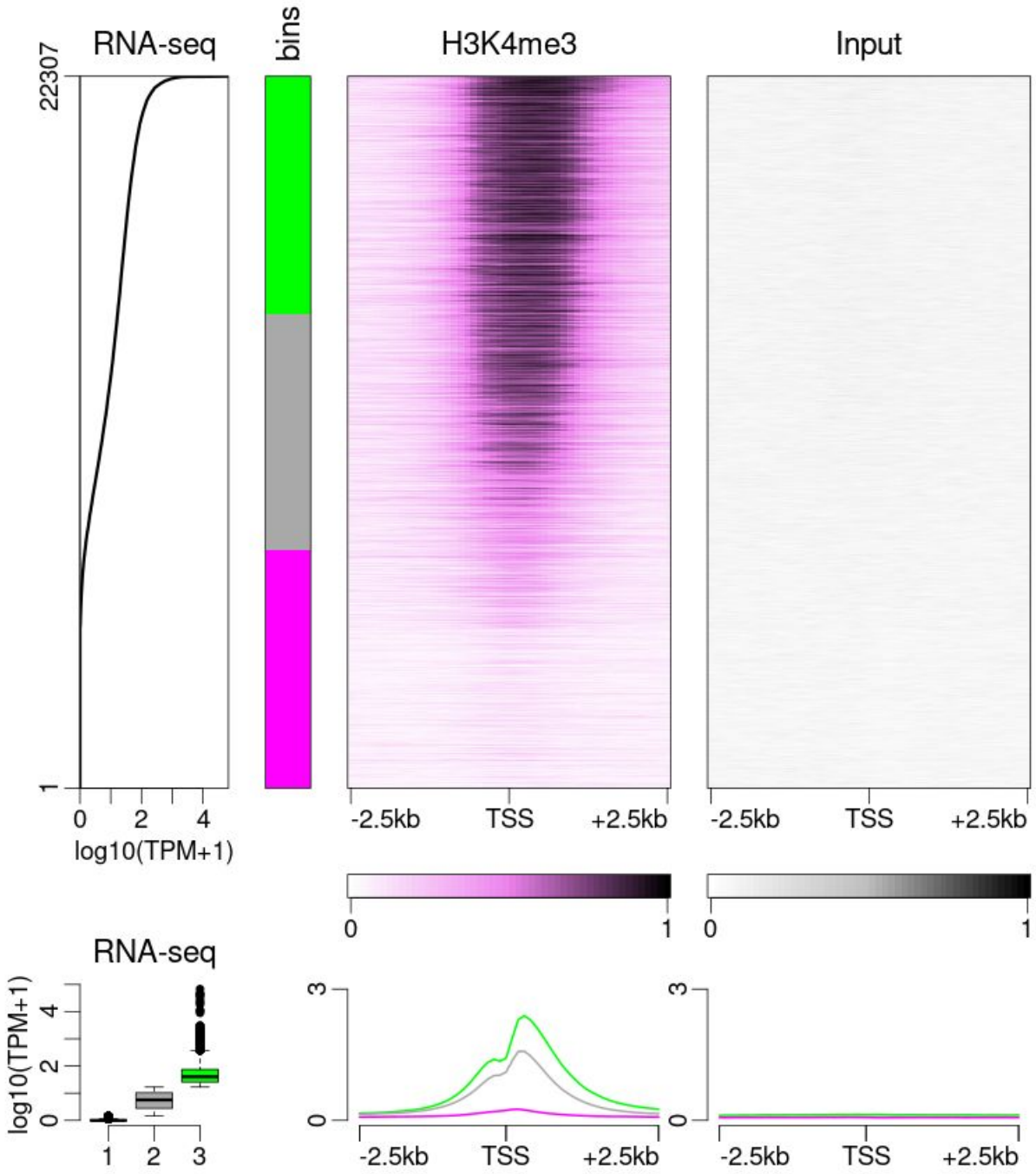
{epistack} epigenetic stacks

Visualise stacks of epigenetic signals on anchor regions:

- Gene starts
- Peak center
- CpG islands
- Differentially methylated regions
- ...

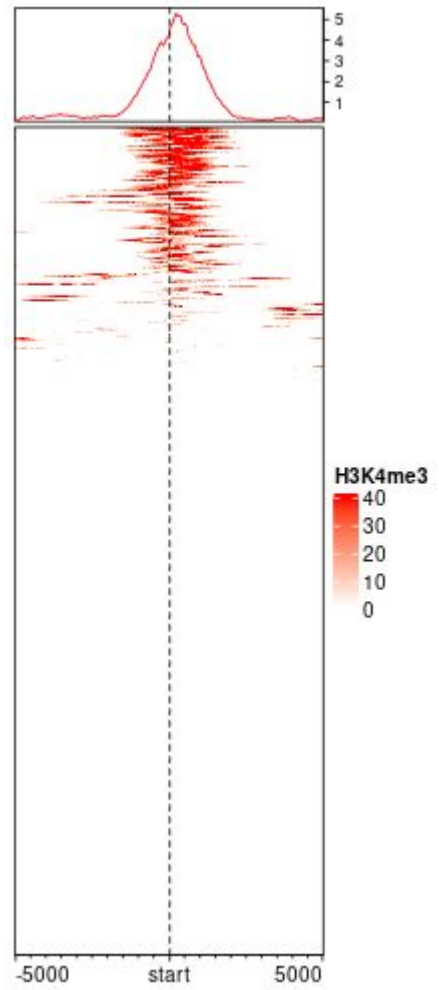
Sort regions:

- Gene expression levels
- P-values
- Fold changes
- Clustering
- Region widths
- Distance to closest TSS
- Gene types
- ...

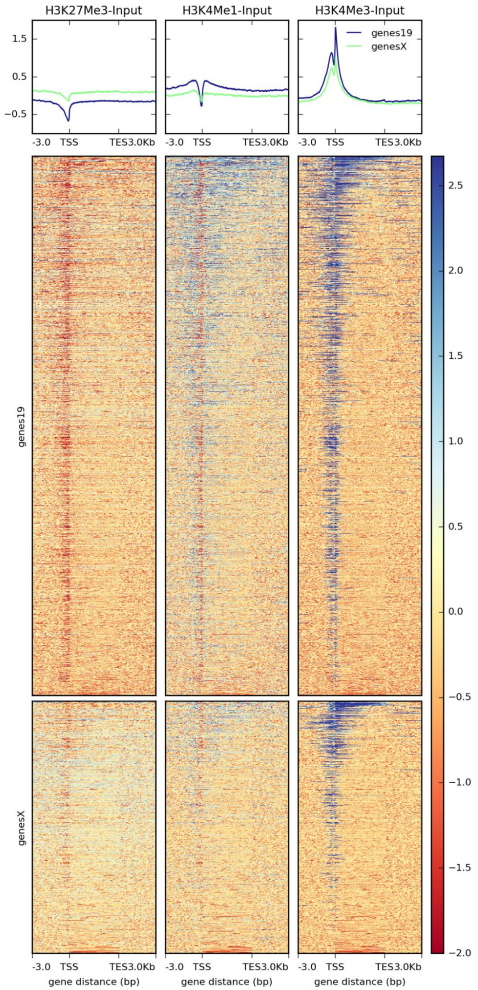


{epistack} alternatives

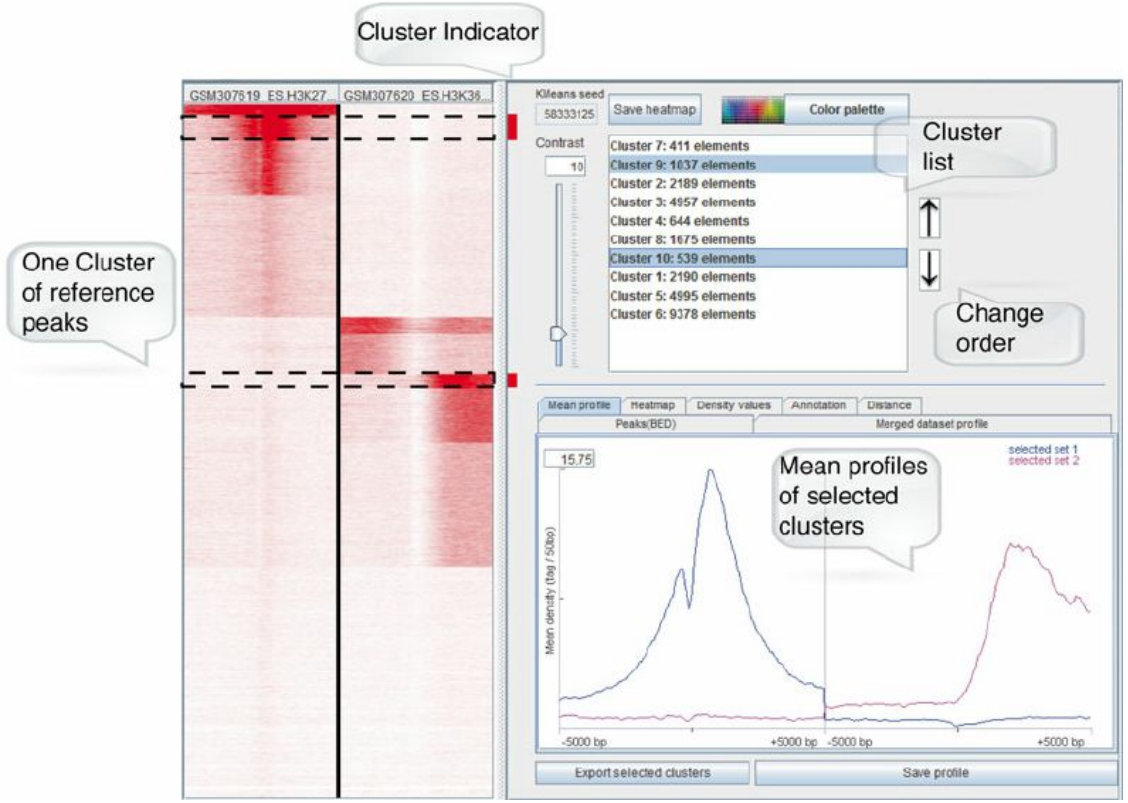
EnrichedHeatmap (bioconductor)



deepTools plotHeatmap (CLI - python)



seqMINER (GUI - java)



And also: seqplots, Repitools, ChIPseeker, ... @Bioconductor

{epistack} is a visualisation package

Input: A SummarizedExperiment object, with signal matrices embedded as assays

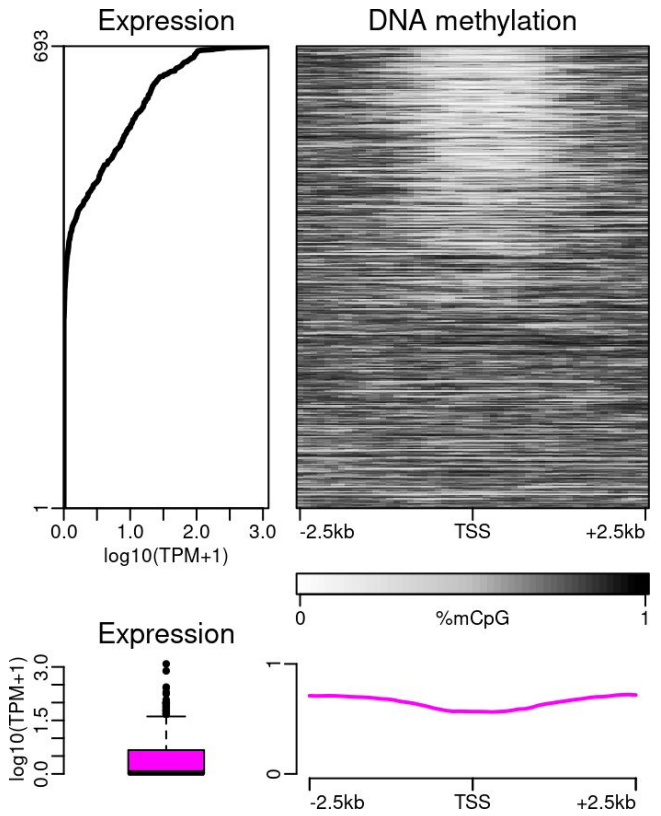
```
library(SummarizedExperiment)
library(epistack)

data("stackepi")
dim(stackepi)
#> [1] 693  51
stackepi
#> class: RangedSummarizedExperiment
#> dim: 693 51
#> metadata(0):
#> assays(1): DNAme
#> rownames(693): ENSSSCG00000016737 ENSSSCG00000036350 ... ENSSSCG00000024209
#>   ENSSSCG00000048227
#> rowData names(3): gene_id exp score
#> colnames(51): window_1 window_2 ... window_50 window_51
#> colData names(0):
```

{epistack} is a visualisation package

Input: A SummarizedExperiment object, with signal matrices embedded as assays

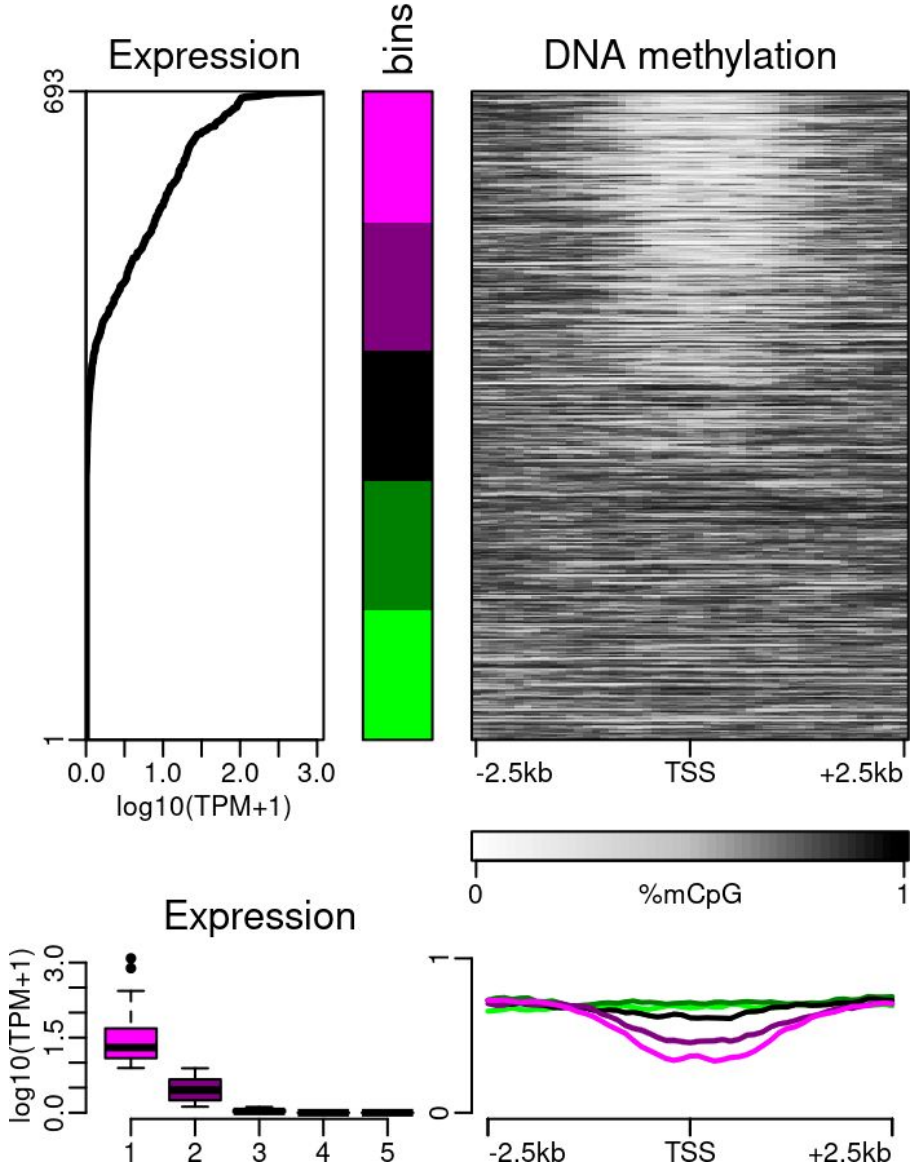
```
library(SummarizedExperiment)
library(epistack)

data("stackepi")
dim(stackepi)
#> [1] 693  51
stackepi
#> class: RangedSummarizedExperiment
#> dim: 693 51
#> metadata(0):
#> assays(1): DNAme
#> rownames(693): ENSSSCG00000016737 ENSSSCG00000036350 ... ENSSSCG00000024209
#>   ENSSSCG00000048227
#> rowData names(3): gene_id exp score
#> colnames(51): window_1 window_2 ... window_50 window_51
#> colData names(0):
```

```
plotEpistack(
  stackepi,
  assays = "DNAme", metric_col = "exp",
  ylim = c(0, 1), zlim = c(0, 1),
  x_labels = c("-2.5kb", "TSS", "+2.5kb"),
  titles = "DNA methylation", legends = "%mCpG",
  metric_title = "Expression", metric_label = "log10(TPM+1)",
  metric_transfunc = function(x) log10(x+1)
)
```



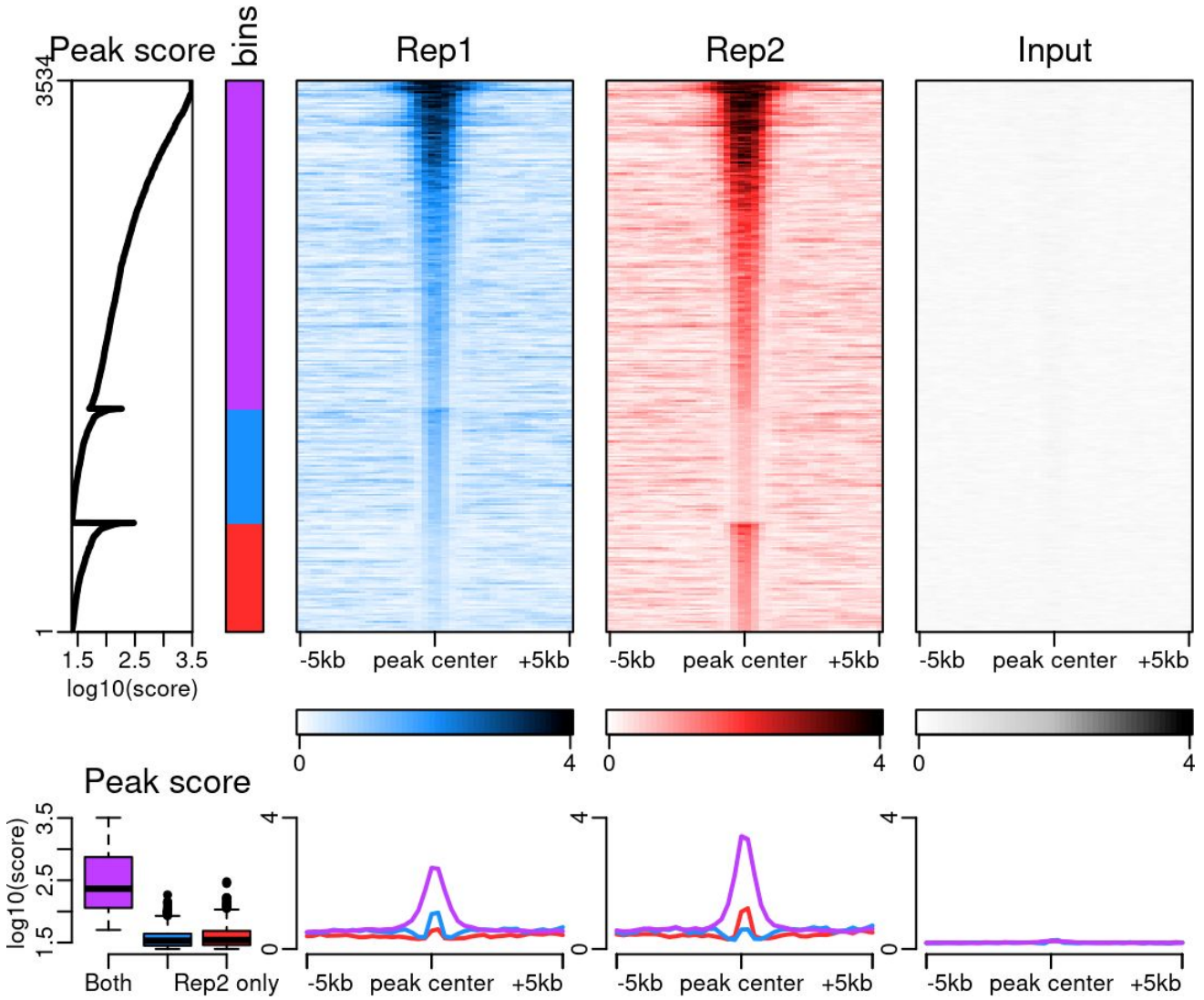
plotEpistack() parameters highlights: bins

```
stackepi <- addBins(stackepi, nbins =  5)

plotEpistack(
  stackepi,
  assays = "DNAme", metric_col = "exp",
  ylim = c(0, 1), zlim = c(0, 1),
  x_labels = c("-2.5kb", "TSS", "+2.5kb"),
  titles = "DNA methylation", legends = "%mCpG",
  metric_title = "Expression", metric_label = "log10(TPM+1)",
  metric_transfunc = function(x) log10(x+1)
)
```



plotEpistack() parameters highlights: several tracks

```
plotEpistack(
  meDP,
  assays = c("Rep1", "Rep2", "input"),
  ...
)
```



Building epistack's input RangedSummarizedExperiment

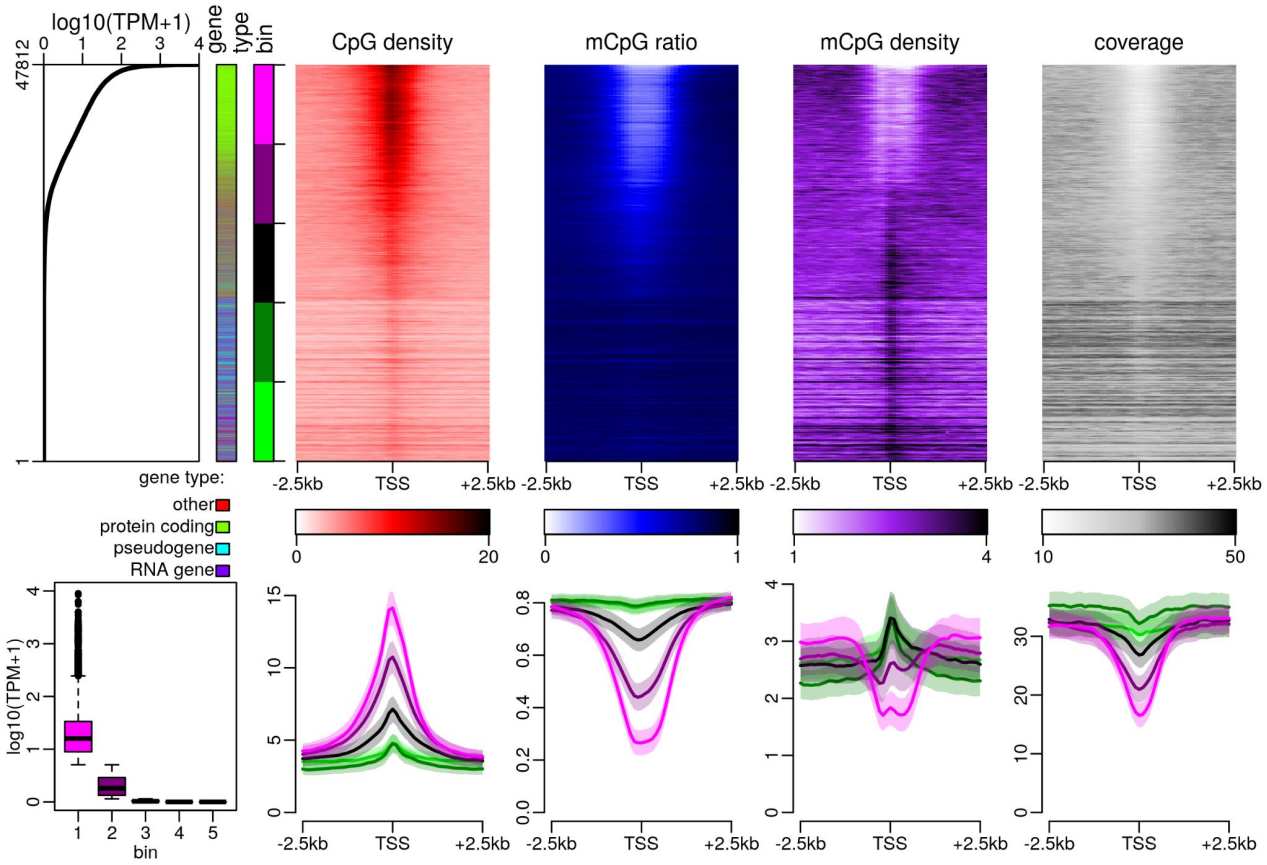
- epigenetic tracks:
  - `.bam`, `.bigwig`
  - load into R with `GenomicAlignment::readGAlignment()` or `rtracklayer::import()`
- anchors:
  - `.bed`, `.gtf`/`.gff`
  - load into R with `rtracklayer::import()`
- get epigenetic stacks matrices
  - `EnrichedHeatmap::normalizeToMatrix()`
  - `Repitools::annotationCounts()`
  - `ChIPseeker::getTagMatrix()`
- an additional ordering vector
  - gene expression, p-value, fold change, etc.
  - `espitack::addMetricAndArrange*()`

Building epistack's input RangedSummarizedExperiment

```
anchors <- rtracklayer::import( "my_peaks.bed" )
signal <- rtracklayer::import( "my_coverage.bw" )

stack <- EnrichedHeatmap::normalizeToMatrix(
  signal,
  anchors,
  extend = 2500, w = 50
)

pack <- SummarizedExperiment(
  rowRanges = anchors,
  assays =  list(stack = stack)
)

plotEpistack(
  pack,
  assays = "stack",
  …
)
```

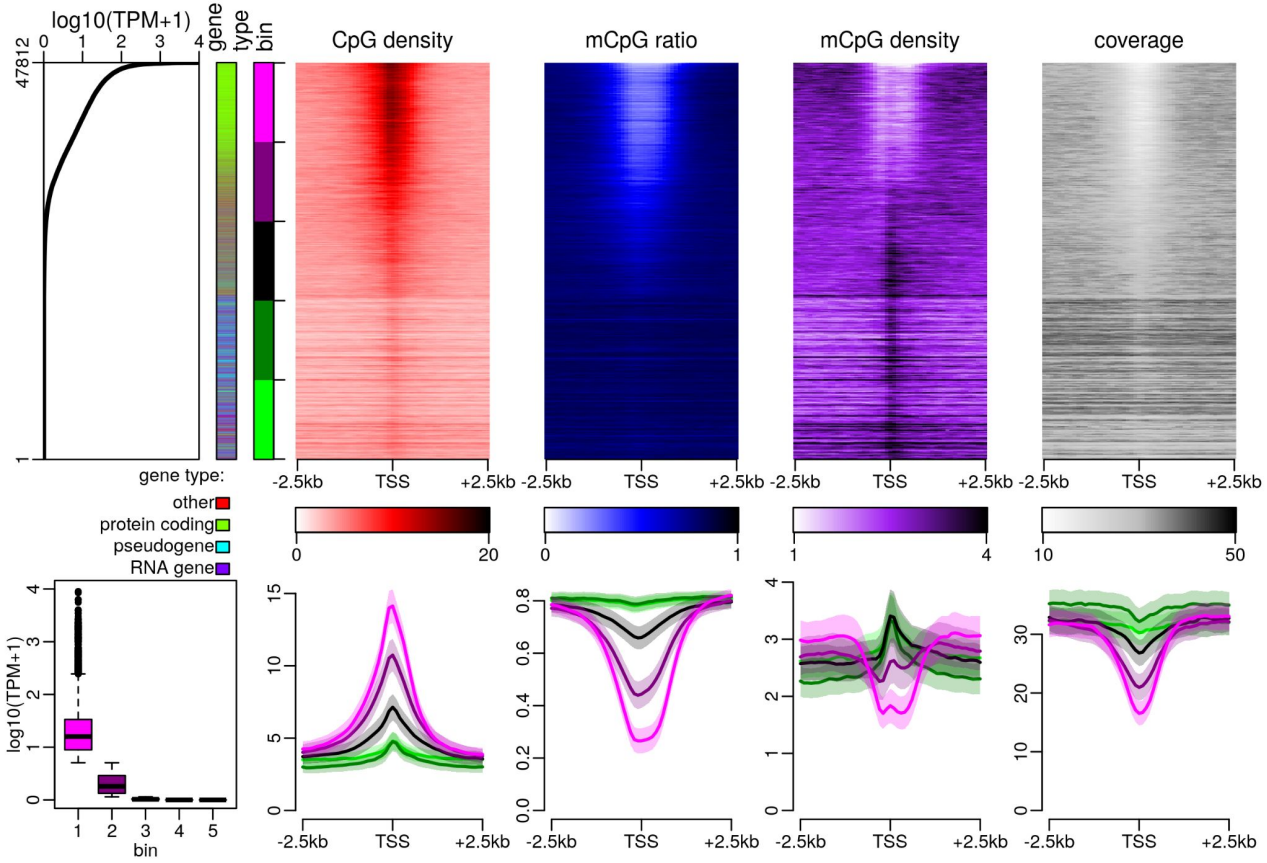
Use case: WGBS coverage at TSS

ROADMAP, H1 cell line

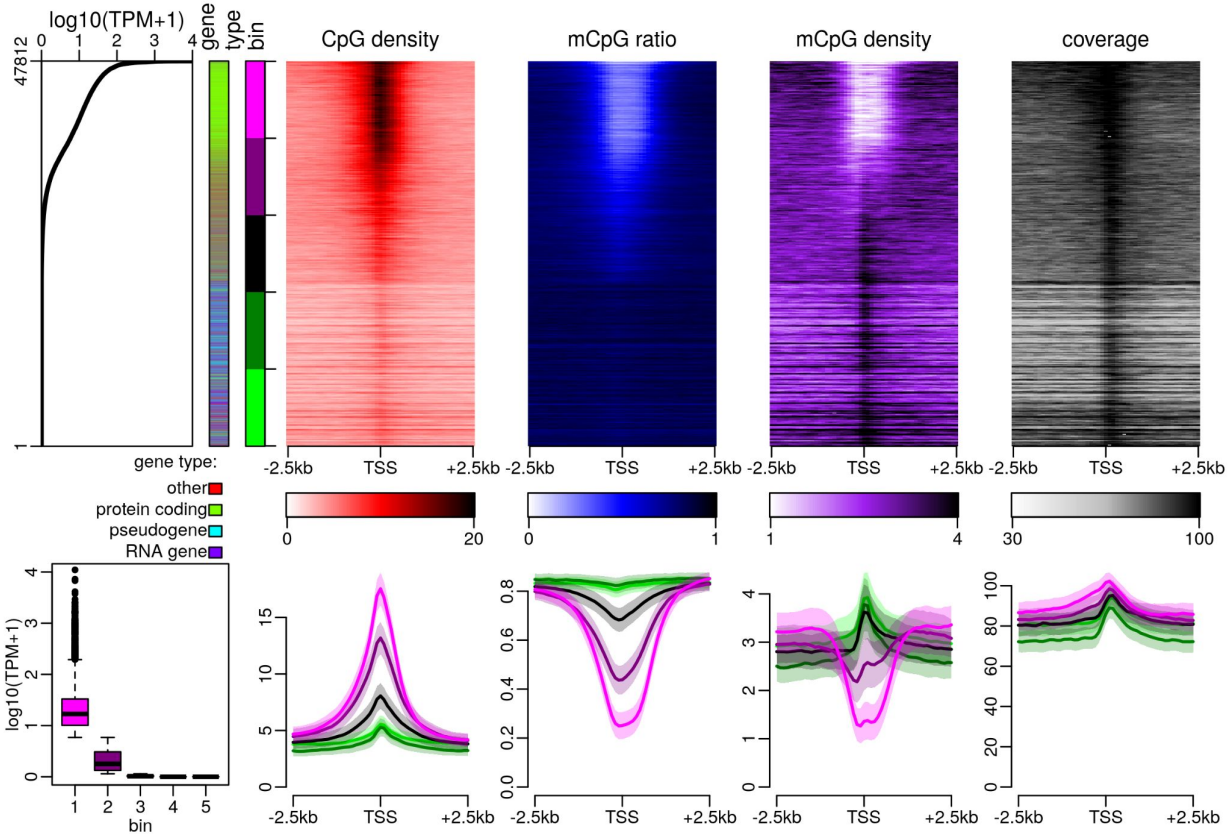


Use case: WGBS coverage at TSS

ROADMAP, H1 cell line

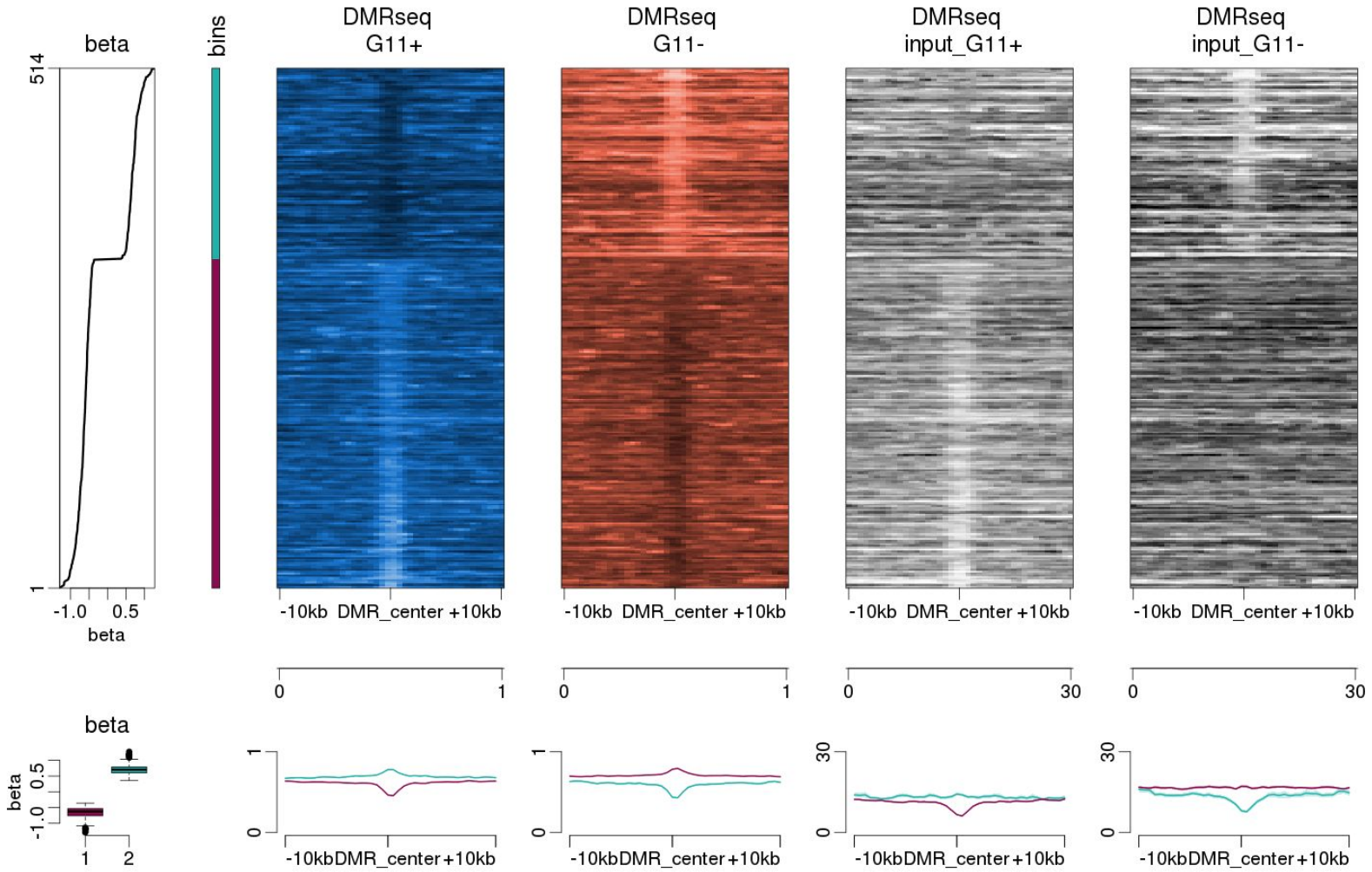


ROADMAP, ESC derived CD56+ Ectoderm

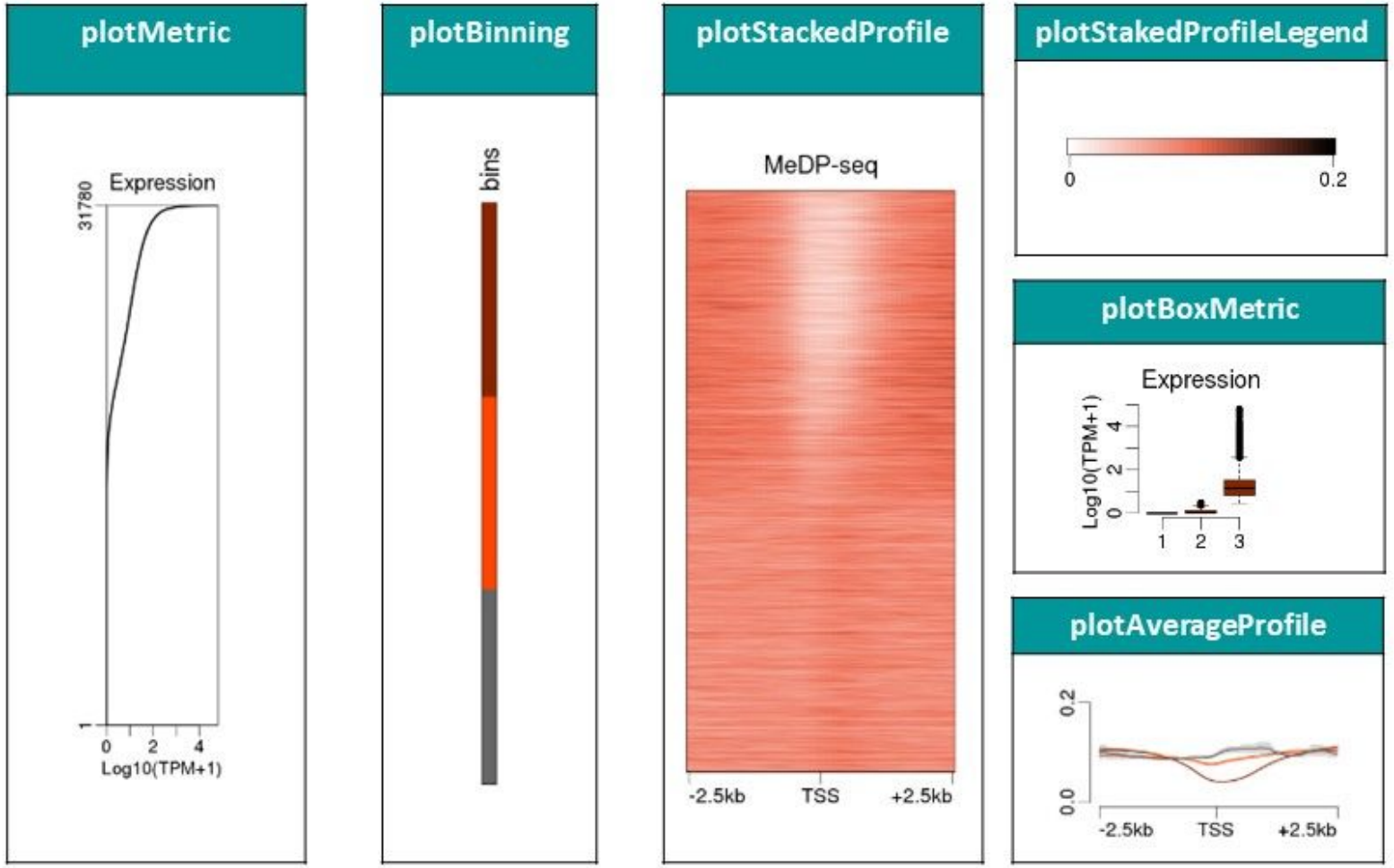


Use case: visualisations of DMR

Differentially Methylated Regions



Individual plotting functions



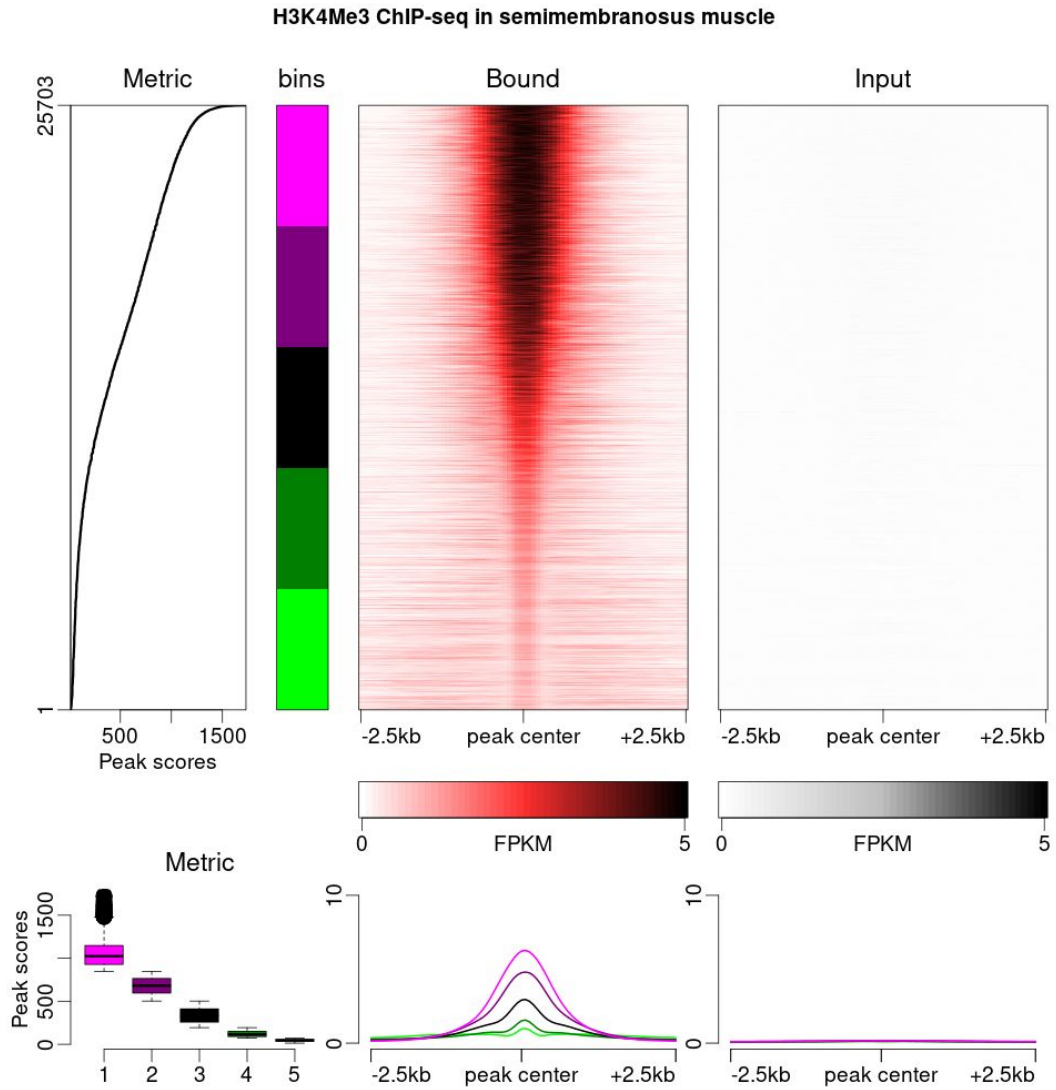
Assemble panels in the order you wish

```
layout()
{gridExtra}
{patchwork}
```

Experimental: CLI interface

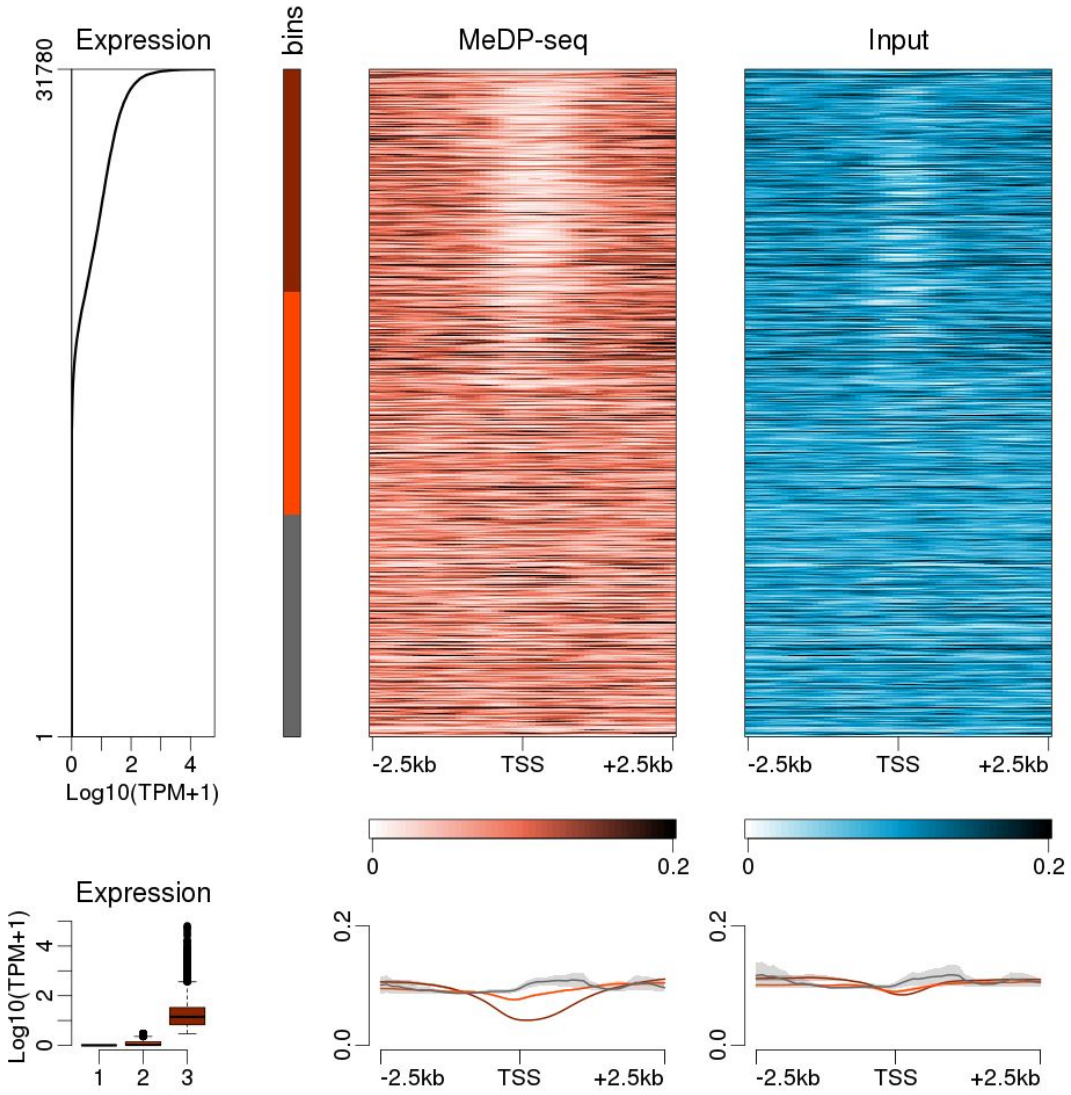
Not feature complete, but may be useful (at least to me!)

```
epistack.R \
     -a ERX5798633_R1_peaks.narrowPeak \
     -b ERX5798633_R1.bigWig \
     -i ERX5798663_R1.bigWig \
     -p ERX5798633.png \
     -t 'H3K4Me3 ChIP-seq in semimembranosus muscle' \
     -r center -y 10 -z 5 -c 2 -v -g 5 -m 99999 -f ci95

> Parsing files... done!
> Processing... done!
> Plotting... done!
> Job completed for file: ERX5798633_R1.bigWig
```

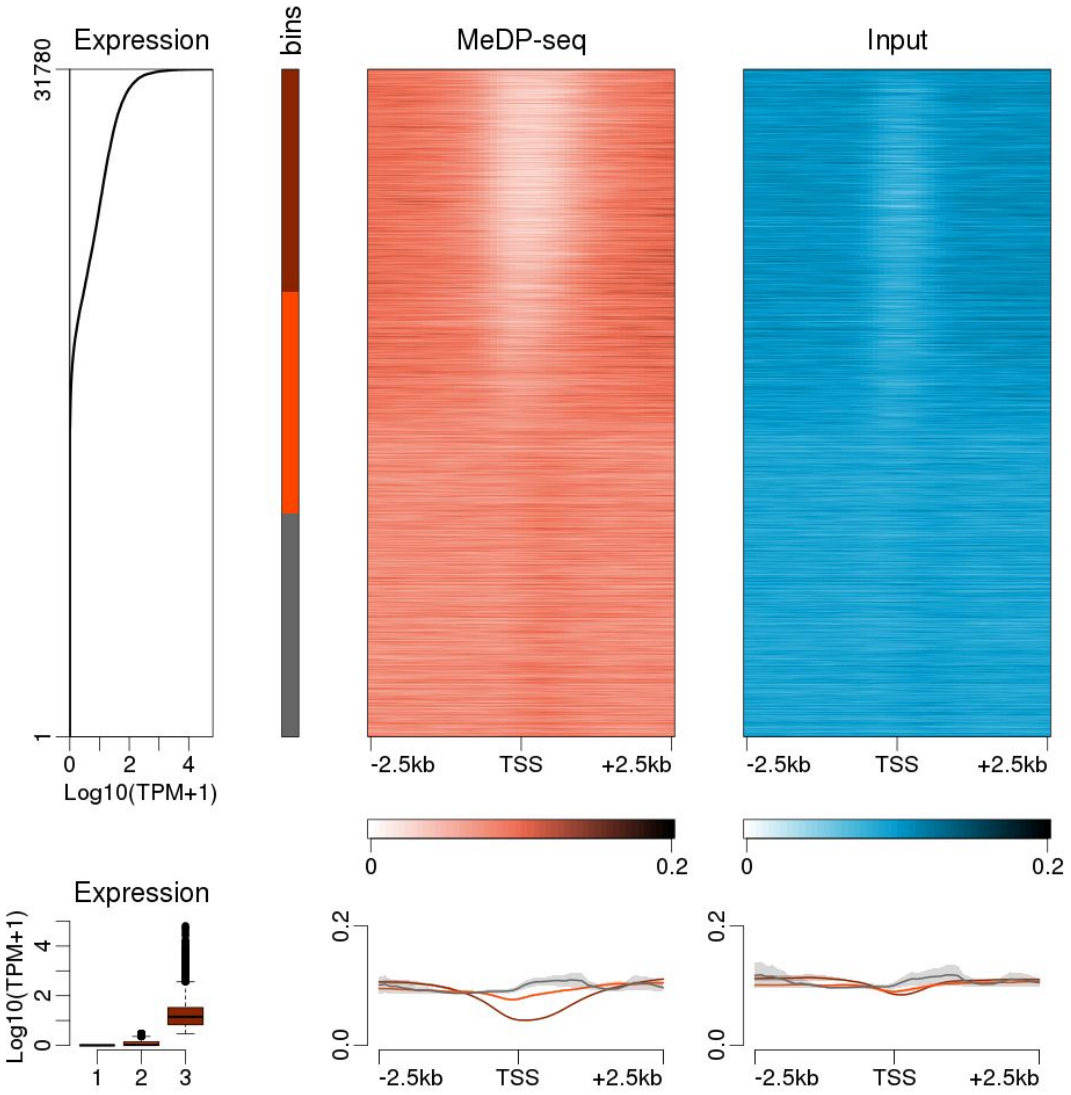


The overplotting issue: more regions than pixels

Default R behaviour

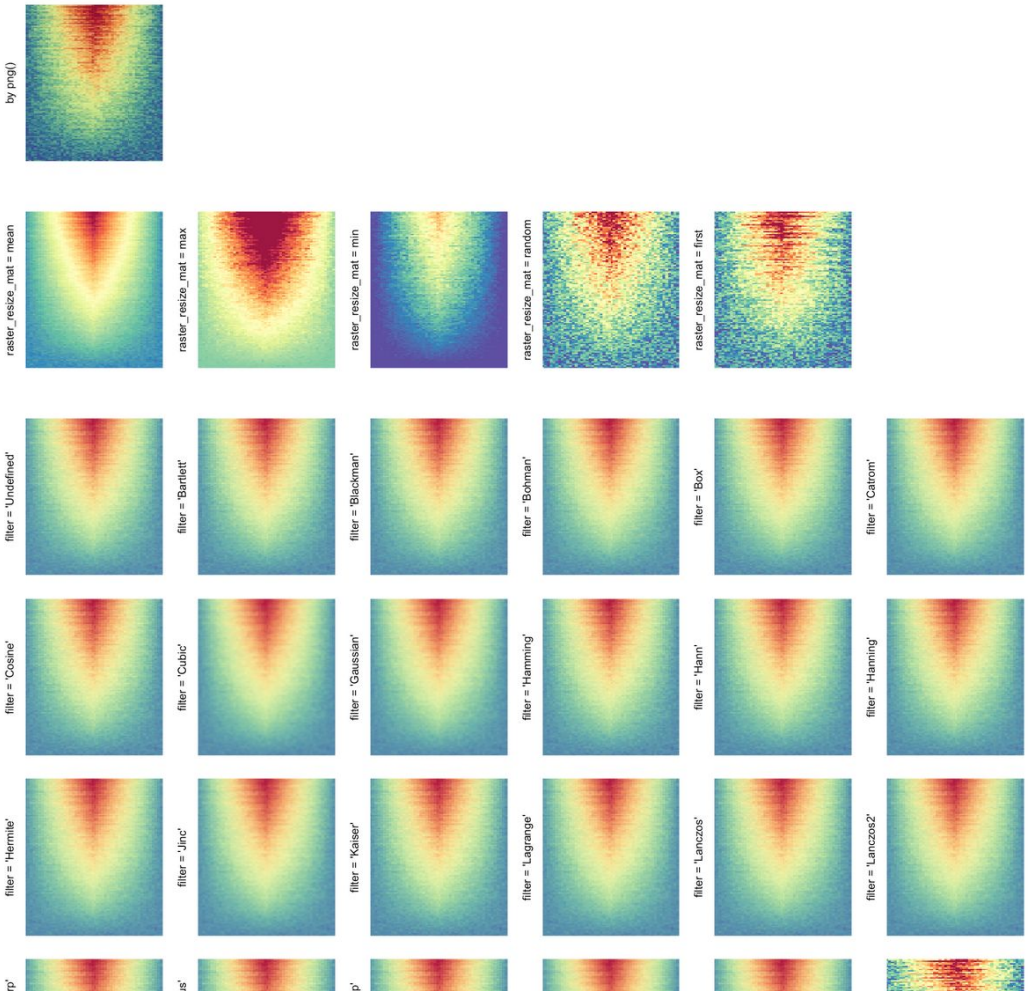


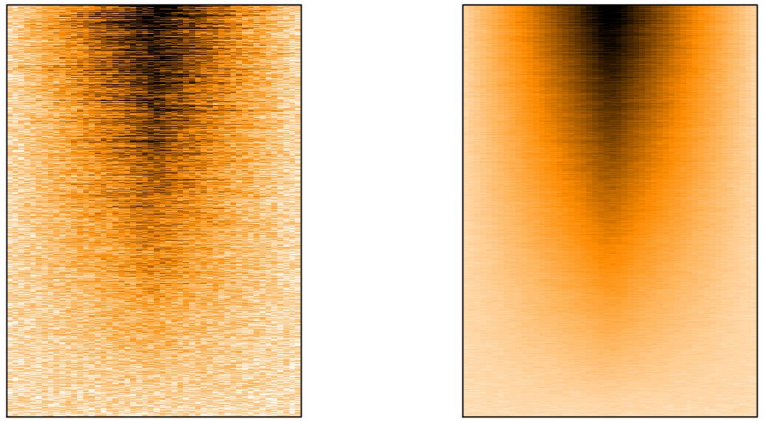
Default {epistack} behaviour



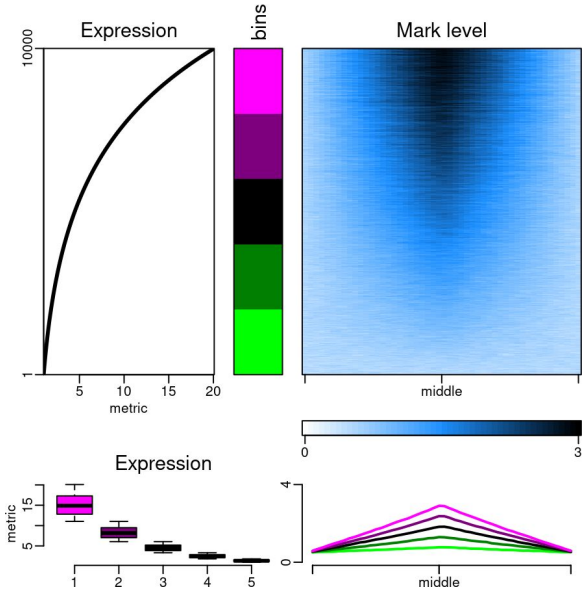
The overplotting issue: more regions than pixels

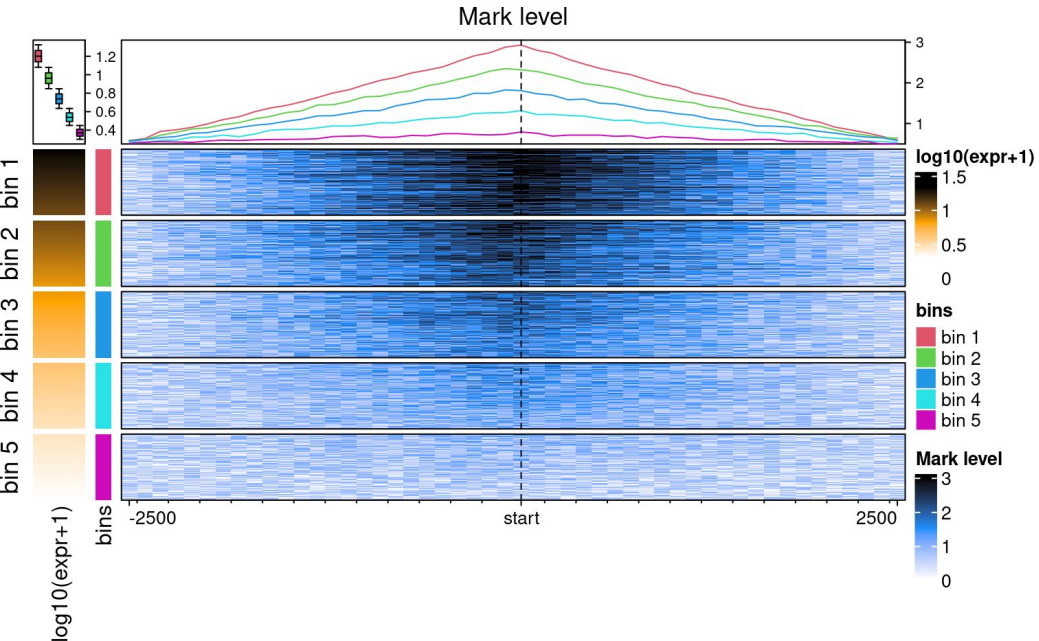
- `episTack::redimMatrix(mat, target_height, target_width, summary_func)`
- bioinfo-fr.net/creer-des-heatmaps-a-partir-de-grosses-matrices-en-r
- jokergoo.github.io/2020/06/30/rasterization-in-complexheatmap/

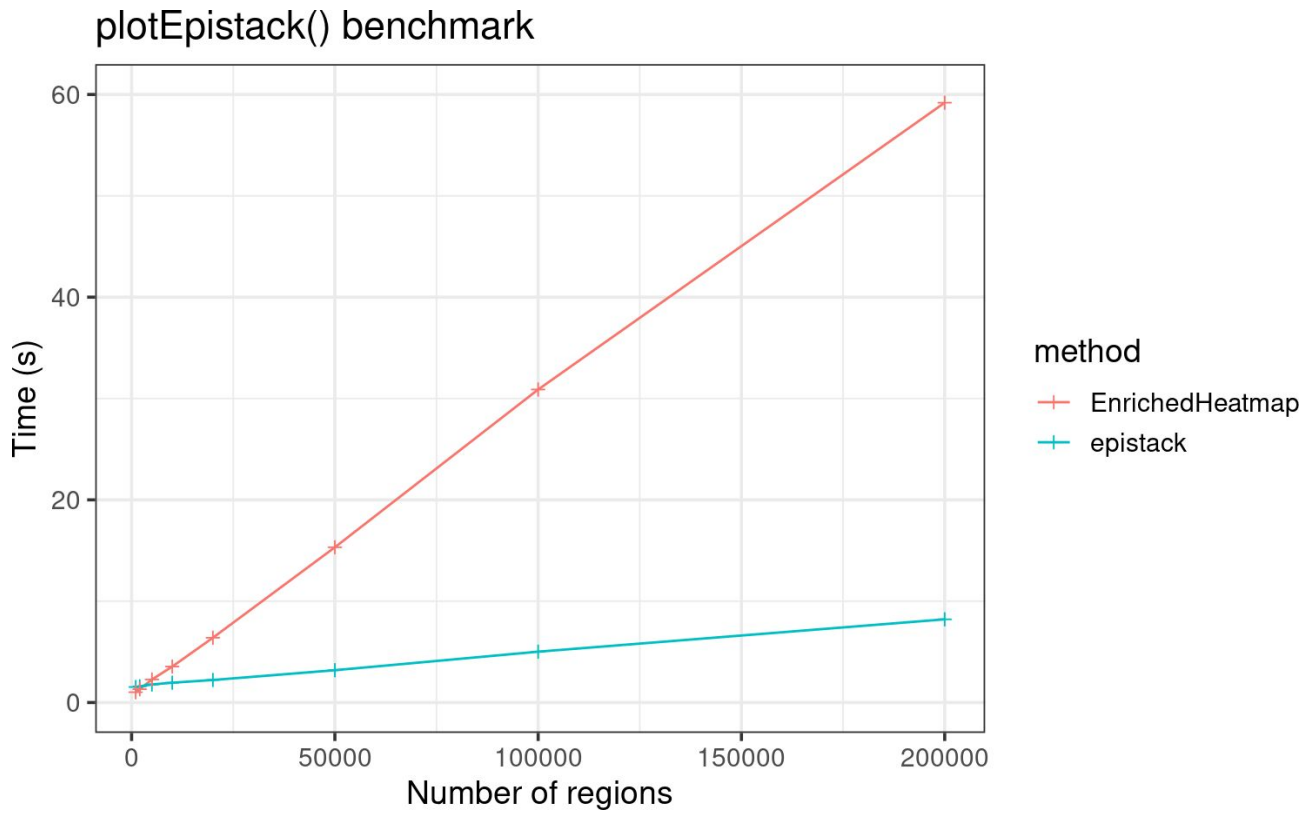




{epistack} vs {EnrichedHeatmap} benchmark







{epistack} in the real world

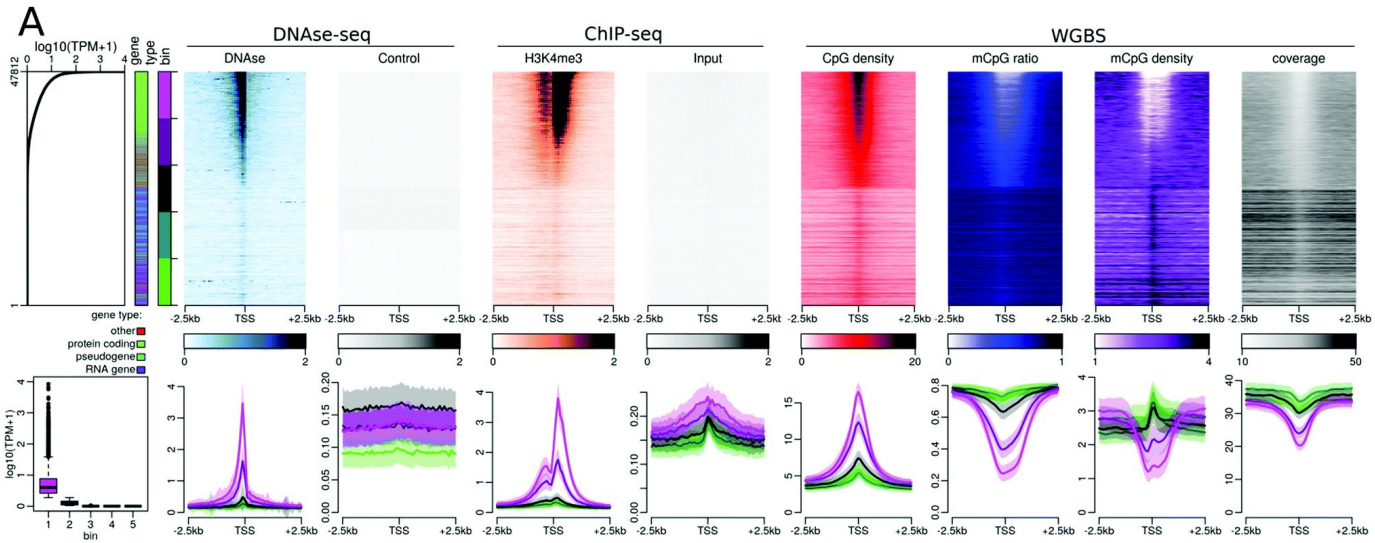
- inspired by our work on ROADMAP data

[doi.org/10.1039/d0mo00130a](doi.org/10.1039/d0mo00130a)

joshiapps.cbu.uib.no/perepigenomics_app/

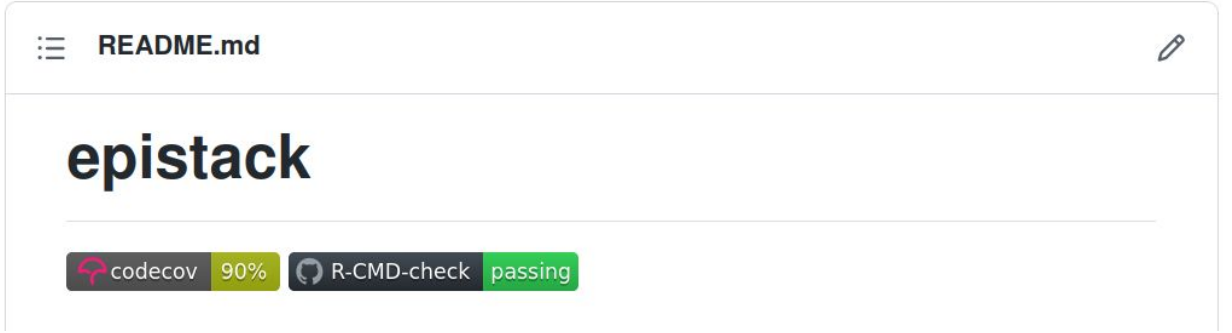
- Used internally to visualise DMR in various team projects

- To be applied on ALL FAANG ChIP-seq / ATAC-seq / WGBS data in the ANR VizFaDa



FAANG
Functional Annotation of Animal Genomes

{epistack}



Available on github:

github.com/GenEpi-GenPhySE/epistack

```
remotes::install_github("GenEpi-GenPhySE/epistack")
```

Available on Bioconductor:

bioconductor.org/packages/epistack/

Vignette:

gdevailly.github.io/using_epistack.html

Built using {devtools}, {testthat}, {roxygen2}, Github Actions