Package 'betterChromVAR'

November 11, 2026

Title Improved ChromVAR (Chromatin Variation Across Regions)

Description A much faster analytical implementation of chromVAR, with additional features, used to infer TF activity from (bulk or single-cell) ATAC-seq data and motif annotations (or binding probabilities). The package also includes the CVnorm normalization method based on the chromVAR logic.

Version 1.0.3

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

URL https://github.com/plger/betterChromVAR

BugReports https://github.com/plger/betterChromVAR/issues

biocViews Software, ATACSeq, Normalization, Epigenetics, Sequencing

Depends SummarizedExperiment

Imports BiocParallel, Biostrings, GenomicRanges, IRanges, Matrix, matrixStats, methods, S4Vectors, stats

Suggests BiocStyle, knitr, rmarkdown, sessioninfo, testthat

Config/testthat/edition 3

VignetteBuilder knitr

git_url https://git.bioconductor.org/packages/betterChromVAR

git_branch RELEASE_3_23

git_last_commit 783b8b9

git_last_commit_date 2026-05-07

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Pierre-Luc Germain [aut, cre] (ORCID: <https://orcid.org/0000-0003-3418-4218>)

Maintainer Pierre-Luc Germain <pierre-luc.germain@hest.ethz.ch>

Contents

addGCBias 2
as.list,bcvBackground-method 3
bcvBackground-class 3
betterChromVAR 4
computeBackgrounds 5
computeDeviationsAnalytic 6
CVnorm 8
getBackgroundBins 9
getDummyData 11
getExpectation 12
normalizeDevsForSize 12
sampleBackgroundPeaks 13
shrinkColumnProps 14

Index **15**

`addGCBias` *addGCBias*

Description

Add the `bias` column to the object's rowData, containing the regions' proportion of Gs and Cs.

Usage

```
addGCBias(object, genome)
```

Arguments

`object`	An object inheriting RangedSummarizedExperiment or GRanges.
`genome`	A BSgenome object or any other genome object supported by `getSeq`.

Value

`object` with the GC content in `mcols(object)$bias` (if GRanges) or `rowData(object)$bias`.

Examples

```
# not run:
# se <- addGCBias(se, genome)
```

as.list,bcvBackground-method *Coerce bcvBackground to a list*

Description

Coerce bcvBackground to a list

Show a bcvBackground object

Subsetting a bcvBackground

Usage

```
## S4 method for signature 'bcvBackground'
as.list(x)

## S4 method for signature 'bcvBackground'
show(object)

## S4 method for signature 'bcvBackground,ANY,ANY,ANY'
x[i, j, ..., drop = TRUE]
```

Arguments

x	A bcvBackground object.
object	A bcvBackground object.
i, j	Indices for subsetting (if j is provided, i is ignored).
...	Additional arguments.
drop	Logical, whether to drop dimensions.

Value

A list containing the slots of the object.

Nothing, prints an overview of the object.

An bcvBackground object.

bcvBackground-class *Bin and background data for betterChromVAR (for internal use)*

Description

Bin and background data for betterChromVAR (for internal use)

betterChromVAR *betterChromVAR*

Description

A fast, analytic implementation of chromVAR. This is a wrapper around the getBackgroundBins, computeBackgrounds, and computeDeviationsAnalytic steps. It additionally allows for multi-threading. For more control or optimization, see the individual steps. A fast, analytic implementation of chromVAR's computeDeviations, with additional features.

Usage

```
betterChromVAR(
  object,
  annotations,
  grouping = NULL,
  nthreads = NULL,
  verbose = FALSE,
  ...
)
```

Arguments

object	A SummarizedExperiment (or SingleCellExperiment) with an assay 'counts', and with a 'bias' column in rowData(object). Note that the regions should have similar widths.
annotations	Peak annotation (sparse) matrix, with motifs as columns, or a SummarizedExperiment containing this in the first assay. Values should be either logical or between 0 and 1.
grouping	An optional factor or vector coercible to a factor indicating the groupings of the columns of object. This is optionally used to compute the base expectation such that rare cell types are given as much weight as abundant ones. In single-cell data, the grouping can for instance be the interaction of samples and cell types. This should either be a vector coercible to factor of length equal to ncol(object), or a character of length 1 specifying a column of colData(object).
nthreads	Either an integer scalar indicating the number of threads to use, or a BiocParallelParam object.
verbose	Logical; whether to output progress messages (default FALSE).
...	Passed to getBackgroundBins.

Details

Contrarily to the original chromVAR, this function is entirely deterministic, and achieves higher precision and much higher efficiency through two changes:

1. working with expected background sampling mean and variances, rather than actual permutations, and 2) computing expectations and variance at the level of bias bins, instead of in the peak-space. The function additionally includes experimental bias shrinkage options, the possibility to handle annotations that are not binary (e.g. probability scores) and a third bias dimension (fragment length bias, which should be stored in `rowData(object)$flbias` see `getBackgroundBins` for details).

Value

A SummarizedExperiment containing the adjusted deviations and z-scores for each motif/sample. The rowData additionally contains the number of motif matches and their variability.

Author(s)

Pierre-Luc Germain

References

Schep A.N., Wu B., Buenrostro J.D., Greenleaf W.J. (2017) chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data, Nature Methods, doi: 10.1038/nmeth.4401

Examples

```
attach(getDummyData())
# if GC content not already in the object, use:
# counts <- addGCBias(counts, genome=YOUR_GENOME)
dev <- betterChromVAR(counts, motifMatches)
dev
# note that this is the exact equivalent of doing:
# bg <- getBackgroundBins(counts)
# bg <- computeBackgrounds(counts, bg)
# dev <- computeDeviationsAnalytic(counts, bg, motifMatches)
```

---

computeBackgrounds *computeBackgrounds*

Description

computeBackgrounds

Usage

```
computeBackgrounds(
  object,
  bins,
  grouping = NULL,
  expectation = NULL,
  shrinkage = c("none", "average", "smooth"),
  sigma = 1,
```

```
  verbose = FALSE
)
```

Arguments

object	A SummarizedExperiment (or SingleCellExperiment) with an assay 'counts', or a (sparse) matrix of counts.
bins	A bcvBackground object, as produced by getBackgroundBins.
grouping	An optional factor or vector coercible to a factor indicating the groupings of the columns of object. This is optionally used to 1) compute the base expectation such that rare cell types are given as much weight as abundant ones, and 2) apply shrinkage (if shrinkage!="none") on a per-grouping fashion. In single-cell data, the grouping can for instance be the interaction of samples and cell types. (The name of a colData column of object can also be provided.)
expectation	Optional vector of length equal to nrow(object) giving the expected counts. If NULL, defaults to mean counts (eventually grouped, see grouping and getExpectation).
shrinkage	The method to use to shrink background (i.e. bias) bin frequencies. Either "average" (shrinks towards the bin's average across cells/samples of the same group), "smooth" (per-sample 2D smoothing over the bin matrix, somewhat redundant with w), or "none" (default).
sigma	Sigma parameter for the 2D smoothing. Ignored unless shrinkage="smooth".
verbose	Logical; whether to output progress messages.

Value

A bcvBackground object with bins*samples slots filled, for use with computeDeviationsAnalytic.

Examples

```
attach(getDummyData())
# if GC content not already in the object, use:
# counts <- addGCBias(counts, genome=YOUR_GENOME)

# we fist get the background bins:
bg <- getBackgroundBins(counts)
# then we can compute the backgrounds for each sample:
bg <- computeBackgrounds(counts, bg)
# for use in computeDeviationsAnalytic...
```

---

computeDeviationsAnalytic

computeDeviationsAnalytic

Description

computeDeviationsAnalytic

Usage

```
computeDeviationsAnalytic(
  object,
  background,
  annotations,
  verbose = FALSE,
  retSE = TRUE,
  compute = c("deviations", "z", "variability"),
  denominator = c("global", "local", "none")
)
```

Arguments

object	A SummarizedExperiment (or SingleCellExperiment) with an assay 'counts', and with a 'bias' column in `rowData(object)`. Note that the regions should have similar widths.
background	A bcvBackground object with bins*samples slots filled, as produced by computeBackgrounds.
annotations	Peak annotation (sparse) matrix, with motifs as columns, or a SummarizedExperiment containing this in the first assay. Values should be either logical or between 0 and 1.
verbose	Logical; whether to output progress messages.
retSE	Logical; whether to return a SummarizedExperiment object.
compute	What to compute. Defaults to everything: deviations, z and motif variability.
denominator	The type of denominator to use for the deviations. Either 'global' (default), i.e. the global expectation (same as the original chromVAR), 'local' (background expectation of the cell/sample), or 'none' (denominator of 1). 'global' (default) is recommended.

Value

A SummarizedExperiment (or a list if `retSE=FALSE`).

Examples

```
attach(getDummyData())
# if GC content not already in the object, use:
# counts <- addGCBias(counts, genome=YOUR_GENOME)

# we fist get the background bins:
bg <- getBackgroundBins(counts)
# then we compute the backgrounds for each sample:
bg <- computeBackgrounds(counts, bg)
# then we can compute the deviations:
dev <- computeDeviationsAnalytic(counts, bg, motifMatches)
dev
```

CVnorm *CVnorm: chromVAR-inspired ATAC-seq normalization*

Description

Corrects ATAC peak counts by removing the effects of technical biases (GC/accessibility) using the chromVAR background binning approach and an optional variance-based bias shrinkage (inspired from the qsmooth package) to preserve group biological signal.

Usage

```
CVnorm(
  object,
  bias = NULL,
  grouping = NULL,
  smoothGrouping = grouping,
  shrinkMode = c("dampen", "qsmooth"),
  toAssay = "corrected",
  bs = NULL,
  w = 0.1,
  Z = FALSE,
  useWidthAdj = NULL,
  enforceZeros = TRUE
)
```

Arguments

`object`	A matrix of counts, or a SummarizedExperiment-like object with an assay named 'counts'.
`bias`	A vector of length equal to `ncol(object)` specifying the per-peak bias (i.e. GC content). If omitted, will try to get it from `rowData(object)$bias`.
`grouping`	Optional grouping for the baseline expectation (prevents bias toward more abundant groups). This should either be a vector coercible to factor of length equal to `ncol(object)`, of a character of length 1 specifying a column of `colData(object)` (if `object` is a `SummarizedExperiment`).
`smoothGrouping`	Optional grouping to determine correction strength. If bias is consistent within these groups, correction is reduced. Accepts the same type of inputs as `grouping`, and by default takes the same values.
`shrinkMode`	The way to perform the group-based shrinkage. With `shrinkMode="dampen"` (default), no corrected is applied in bins when the bias is entirely explained by groups. `shrinkMode="qsmooth"` instead reproduces the logic of the `qsmooth` package: if variance in bias is chiefly explained by groups, between group bias will not be corrected, but within-group differences will be. Using this prior to differential analysis however leads to increase Type I error rate, and it should therefore not be used for downstream application.

toAssay	The name of the assay in which to store the corrected data (default 'corrected'). Ignored unless object is a SummarizedExperiment-like object.
bs	Number of bins per dimension (see getBackgroundBins).
w	Standard deviation of the Gaussian kernel for bin smoothing.
Z	Logical; whether to return standardized residuals (Z-scores) instead of the (default) corrected counts.
useWidthAdj	Whether to adjust for the different width of the regions. If omitted, will be TRUE if the average absolute difference to the median width is greater than 10% of the median width. If TRUE, will adjust for pmax(200L,width). If an integer scalar, will adjust for pmax(useWidthAdj,width).
enforceZeros	Logical; whether to enforce that zero counts should remain zeroes after correction (ignored if Z=TRUE).

Value

If object is a matrix, then a matrix of corrected counts of the same dimensions. If object is a SummarizedExperiment-like object, then the object is returned with an extra assay named based on toAssay.

Author(s)

Pierre-Luc Germain

References

- Schep A.N., Wu B., Buenrostro J.D., Greenleaf W.J. (2017) chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data, Nature Methods, doi: 10.1038/nmeth.4401
- Hicks SC, Okrah K, Paulson JN, Quackenbush J, Irizarry RA, Corrado Bravo H (2018). "Smooth quantile normalization." Biostatistics 19 (2), doi: 10.1093/biostatistics/kxx028

Examples

```
counts_se <- getDummyData()$counts
# if GC content not already in the object, use:
# counts_se <- addGCBias(counts_se, genome=YOUR_GENOME)
counts_se <- CVnorm(counts_se)
```

---

getBackgroundBins *getBackgroundBins*

Description

Computes chromVAR-like background (i.e. bias) bins, as well as bin-to-bin selection probabilities needed for betterChromVAR.

Usage

```
getBackgroundBins(
  x,
  bias = NULL,
  flbias = NULL,
  w = 0.1,
  bs = NULL,
  pseudo = 0,
  verbose = TRUE
)
```

Arguments

x	A SummarizedExperiment containing a 'counts' assay, or a matrix of counts, or a vector of expected (e.g. mean) counts.
bias	A vector of length equal to `ncol(object)` specifying the per-peak bias (i.e. GC content). If omitted, will try to get it from `rowData(x)$bias`.
flbias	A vector of length equal to `ncol(object)` specifying the per-peak fragment length bias (e.g. log10-transformed median length of fragments overlapping each region). If omitted, will try to get it from `rowData(object)$flbias`. If absent, will use 2-dimensional bias bins (which is already pretty good).
w	Standard deviation of the Gaussian kernel.
bs	Number of bins per dimension. This can be a single integer (total bins = `bs^2`), or an integer vector of length 2 (if `flbias=NULL`) or 3 (in which case there are `prod(bs)` total bins). The values specify the number of bins for, in order: enrichment, GC and fragment length. By default, `bs=50` if `flbias` is not provided (mimicking chromVAR), and `bs=c(30, 30, 6)` if it is.
pseudo	Optional pseudocount to be added. This should not be needed with standard workflows.
verbose	Whether to print processing info.

Details

The procedure underlying this function is the same as in `chromVAR::getBackgroundPeaks`, with the following differences:

- Rather than producing a set of background peaks for each input peak, the function returns peak-to-bin mappings and bin-to-bin background selection probabilities, which enables an analytic background computation. It is, as such, entirely deterministic.
- The function supports the optional use of a third bias dimension, provided through the `flbias` argument, meant for fragment length bias. This is still an experimental feature.

Value

A `bcvBackground` object, to be used with `computeBackgrounds`.

References

Schep A.N., Wu B., Buenrostro J.D., Greenleaf W.J. (2017) chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data, Nature Methods, doi: 10.1038/nmeth.4401

Examples

```
counts_se <- getDummyData()$counts
background <- getBackgroundBins(counts_se)
```

`getDummyData` *Dummy data for testing purposes*

Description

Dummy data for testing purposes

Usage

```
getDummyData(nRegions = 500, nSamples = 10, nMotifs = 5)
```

Arguments

`nRegions`	Number of regions to generate
`nSamples`	Number of samples to generate
`nMotifs`	Number of motifs to generate

Value

A list with the slots `counts` (a peak counts SummarizedExperiment) and `matches` (a sparse matrix of binary motif matches per peaks)

Examples

```
out <- getDummyData()
(counts <- out$counts)
matches <- out$motifMatches
```

getExpectation *getExpectation*

Description

Computes expected counts (a glorified rowMeans)

Usage

```
getExpectation(counts, grouping = NULL, normalize = TRUE)
```

Arguments

counts	A count matrix, or object inheriting SummarizedExperiment with a 'counts' assay.
grouping	An optional vector of length equal to ncol(counts), indicating the grouping of the cells. If provided, cells will be averaged by group before averaging across groups.
normalize	Logical; whether to normalize data between averaging (but after grouping). Default TRUE and highly recommended if providing grouping.

Value

A vector of expectation for each row of counts

Examples

```
attach(getDummyData())
e <- getExpectation(counts)
```

normalizeDevsForSize *normalizeDevsForSize*

Description

Normalizes the z-scores assay of a deviations object to make the scores comparable across motifs with different number of matches.

Usage

```
normalizeDevsForSize(dev)
```

Arguments

dev	A SummarizedExperiment object as produced by betterChromVAR or computeDeviationsAnalytic.

Value

The dev object with an additional assay named 'norm'.

Examples

```
attach(getDummyData())
dev <- betterChromVAR(counts, motifMatches)
dev <- normalizeDevsForSize(dev)
dev
```

`sampleBackgroundPeaks` *sampleBackgroundPeaks*

Description

Given a background generated by `getBackgroundBins`, samples background peaks for each input peak.

Usage

```
sampleBackgroundPeaks(background, niterations = 50)
```

Arguments

background	A bcvBackground produced by `getBackgroundBins`.
niterations	Number of background peaks to sample for each target peak.

Details

This function is not used by `betterChromVAR`, which is deterministic, but for other applications requiring an outputs similar to that of the original `getBackgroundPeaks`.

Value

A peaks x niterations matrix of integers representing the indices of the sampled background peaks.

Examples

```
counts_se <- getDummyData()$counts
background <- getBackgroundBins(counts_se)
bg_peaks <- sampleBackgroundPeaks(background, niterations=20)
```

shrinkColumnProps *shrinkColumnProps*

Description

Empirical Bayes shrinkage of a matrix of counts towards a prior proportion (by default the mean across columns).

Usage

```
shrinkColumnProps(x, shrinkTo = NULL, var.theo = FALSE)
```

Arguments

x	A matrix of counts, with features as rows and samples as columns.
shrinkTo	A vector (of length equal to nrow(x)) of sampling probabilities to shrink towards, or a matrix (of the same dimensions as x) of such probabilities. If omitted, will be the weighted mean of the columns' relative frequencies.
var.theo	Logical; whether to use theoretical (i.e. binomial) variances of the proportions, rather than the observed (weighted) variance.

Value

A matrix of the same dimensions as x representing the shrunk column-wise proportions.

Examples

```
# generate a matrix of 5 sampling (with different total counts) of 20
# features based on the same base frequency :
baseFreq <- abs(rnorm(20))
baseFreq <- baseFreq/sum(baseFreq)
mat <- sapply(c(10,20,30,40,50), function(tot){
  rpois(length(baseFreq), baseFreq*tot)
})
# apply shrinkage and confirm that shrunk proportions are better correlated
shrunk_mat <- shrinkColumnProps(mat)
mean(cor(shrunk_mat))>mean(cor(mat))
```

Index

[,bcvBackground,ANY,ANY,ANY-method
(as.list,bcvBackground-method), 3

addGCBias, 2
as.list,bcvBackground-method, 3

bcvBackground-class, 3
betterChromVAR, 4, *9*, *12*, *13*

computeBackgrounds, *4*, 5, *7*, *10*
computeDeviationsAnalytic, *4*, *6*, 6, *12*
CVnorm, 8

getBackgroundBins, *4–6*, *9*, 9, *13*
getBackgroundPeaks, *13*
getDummyData, 11
getExpectation, *6*, 12
getSeq, *2*

normalizeDevsForSize, 12

sampleBackgroundPeaks, 13
show,bcvBackground-method
(as.list,bcvBackground-method), 3
shrinkColumnProps, 14