methylTFRAnnotationMm10

November 13, 2026

`availableMotifSets` *List the motif sets provided by this package*

Description

Returns the names accepted by the `motifSet` argument of `getTFbindsites` and `getGCfreq`.

Usage

```
availableMotifSets()
```

Value

A character vector of motif set names.

See Also

`methylTFRAnnotationMm10` for an overview of the package.

Examples

```
availableMotifSets()
```

`genomewide_GC` *Genome-wide GC distribution*

Description

GC content of the mm10 genome in non-overlapping 30 nt windows. Returned by `getGenomeGC`; the AnnotationHub record title is `genomewide_GC_mm10.rds`.

Format

A `GRanges`, one range per 30 nt window on the primary chromosomes, with metadata columns

`GC_bias` Numeric, GC fraction of the window (0 to 1); `NaN` for windows made of `N`.

`GC_bin` Integer, genome-wide GC quintile of the window, 1 (lowest) to 5 (highest).

and `S4Vectors::metadata()` entries `gc_breaks` (the six quintile boundaries), `bin_scope` (`"genome"`: quantiles taken genome-wide) and `step` (window step, 30).

Details

methylTFR overlaps each methylation call with these windows to assign it a GC bin; the motif tables in `motif_gcfreq` use the same boundaries.

Value

This page documents a data resource rather than a function; the object is returned by `getGenomeGC`.

Source

Computed from the UCSC mm10 BSgenome package; see `system.file("scripts", "make-data.R", package = "methylTFRAnnotationMm10")`.

See Also

`getGenomeGC`, `methylTFRAnnotationMm10`

Examples

```
# AnnotationHub records of this resource and their descriptions
md <- utils::read.csv(system.file("extdata", "metadata.csv",
    package = "methylTFRAnnotationMm10"
))
md[grepl("genomewide_GC", md$Title), c("Title", "RDataClass")]

# Loading the resource downloads it from AnnotationHub on first use:
# gc <- getGenomeGC()
# table(gc$GC_bin); S4Vectors::metadata(gc)$gc_breaks
```

`getGCfreq` *Load motif GC frequency tables*

Description

Load the motif GC frequency tables for a motif set. Downloaded from AnnotationHub on first use and cached locally thereafter.

Usage

```
getGCfreq(motifSet = "altius")
```

Arguments

`motifSet`	Motif set to load. One of `availableMotifSets()`: `altius`, `cisbpv2`, `jaspar2020`. Case-insensitive.

Details

Each table describes the GC composition around one motif's binding sites. Column j corresponds to the j-th 30 nt window along the (extended) binding site; row i gives the fraction of binding sites whose window at that position falls into genome-wide GC quintile i (row 1 lowest, row 5 highest GC). Columns therefore sum to one. **methylTFR** combines these fractions with the observed methylation per GC bin to obtain the methylation a motif would be expected to show from sequence composition alone. See `motif_gcfreq` for the data format.

Value

A named `list` of numeric matrices with five rows (GC quintiles), one matrix per motif.

See Also

`motif_gcfreq`, `getTFbindsites`, `getGenomeGC`

Examples

```
# Point the package at a temporary directory holding a small mock
# table so the example runs without downloading from AnnotationHub.
mock_dir <- file.path(tempdir(), "gcfreq_example")
dir.create(mock_dir, showWarnings = FALSE)
freq <- matrix(c(0.1, 0.2, 0.4, 0.2, 0.1), nrow = 5, ncol = 3)
saveRDS(list(MA0001.1 = freq), file.path(mock_dir,
"altius_motif_gcfreq.rds"))
old_opt <- options(methylTFRAnnotationMm10.datadir = mock_dir)

gcfreqs <- getGCfreq("altius")
gcfreqs$MA0001.1
colSums(gcfreqs$MA0001.1)

options(old_opt)
unlink(mock_dir, recursive = TRUE)
```

`getGenomeGC` *Load the genome-wide GC distribution*

Description

Load the genome-wide GC distribution for mm10. Downloaded from AnnotationHub on first use and cached locally thereafter.

Usage

```
getGenomeGC(assembly = "mm10")
```

Arguments

`assembly`	Genome assembly. Only `"mm10"` (the default) is available in this package; case-insensitive.

Details

The genome is tiled into non-overlapping 30 nt windows. Each window carries its GC fraction and its genome-wide GC quintile. **methylTFR** uses the table to assign every methylation call to a GC bin. See `genomewide_GC` for the data format.

The package serves a single assembly, so the argument can be left out. It is kept so that code written as `getGenomeGC("mm10")` works; any other assembly is an error.

Value

A GRanges with metadata columns GC_bias (GC fraction of the window) and GC_bin (quintile, 1 to 5).

See Also

genomewide_GC, getGCfreq

Examples

```
# Point the package at a temporary directory holding a small mock
# object so the example runs without downloading from AnnotationHub.
mock_dir <- file.path(tempdir(), "genomegc_example")
dir.create(mock_dir, showWarnings = FALSE)
gr <- GenomicRanges::GRanges(c("chr1:1-30", "chr1:31-60", "chr1:61-90"))
gr$GC_bias <- c(0.30, 0.43, 0.63)
gr$GC_bin <- c(1L, 3L, 5L)
saveRDS(gr, file.path(mock_dir, "genomewide_GC_mm10.rds"))
old_opt <- options(methylTFRAnnotationMm10.datadir = mock_dir)

getGenomeGC()

options(old_opt)
unlink(mock_dir, recursive = TRUE)
```

---

getTFbindsites *Load transcription factor binding sites*

Description

Load genome-wide transcription factor binding sites for a motif set. Downloaded from AnnotationHub on first use and cached locally thereafter.

Usage

```
getTFbindsites(motifSet = "altius")
```

Arguments

motifSet	Motif set to load. One of availableMotifSets(): altius, cisbpv2, jaspar2020. Case-insensitive.

Details

Each element of the returned list holds the binding sites of one motif. Every range is the motif match extended by 200 bp on either side, the footprint window across which **methylTFR** reads methylation. See tf_bindsites for the data format.

Value

A GRangesList with one GRanges per motif, named by motif ID.

See Also

tf_bindsites, getGCfreq

Examples

```
# Point the package at a temporary directory holding a small mock
# object so the example runs without downloading from AnnotationHub.
mock_dir <- file.path(tempdir(), "tfbs_example")
dir.create(mock_dir, showWarnings = FALSE)
mock_grl <- GenomicRanges::GRangesList(
    MA0001.1 = GenomicRanges::GRanges(
        c("chr1:1001-1411", "chr2:5001-5411"),
        strand = c("+", "-")
    )
)
saveRDS(mock_grl, file.path(mock_dir, "altius_tf_bindsites.rds"))
old_opt <- options(methylTFRAnnotationMm10.datadir = mock_dir)

getTFbindsites("altius")

options(old_opt)
unlink(mock_dir, recursive = TRUE)
```

---

methylTFRAnnotationMm10-package

methylTFRAnnotationMm10: methylTFR annotations for mm10

Description

Genome annotations that **methylTFR** needs to compute GC-bias-corrected transcription factor (TF) deviation scores from DNA methylation data on the Mus musculus mm10 assembly.

Details

The package provides three kinds of resources, each with an accessor:

getTFbindsites Genome-wide TF binding sites, one GRanges per motif, extended to the footprint window (see tf_bindsites).

getGCfreq Motif GC frequency tables: how each motif's binding sites distribute over genome-wide GC quintiles (see motif_gcfreq).

getGenomeGC The genome-wide GC distribution in 30 nt windows (see genomewide_GC).

availableMotifSets lists the motif sets that can be requested. The three objects are passed to methylTFR::run_methyltfr().

The data are too large to ship with the package. They are hosted on AnnotationHub, downloaded on first use and served from the local AnnotationHub cache afterwards. The metadata of every resource is listed in system.file("extdata", "metadata.csv", package = "methylTFRAnnotationMm10"), and the scripts that produced them are in system.file("scripts", package = "methylTFRAnnotationMm10").

To read the .rds files from a local directory instead of AnnotationHub (for example on a machine without internet access), set options(methylTFRAnnotationMm10.datadir = "/path/to/files") or the environment variable METHYL_TFRANNOTATION_Mm10_DIR.

Author(s)

Maintainer: Irem B. Gunduz <irembgunduz@gmail.com> (ORCID)

Other contributors:

- ERA-NET Transcan-Neu III - EPILUNAR (Grant 01KT2409) [funder]
- Saarland University (NanoBioMed Young Investigator Grant) [funder]

See Also

```
vignette("methylTFRAnnotationMm10")
```

`motif_gcfreq` *Motif GC frequency tables*

Description

Per-motif tables describing the GC composition around each motif's binding sites on mm10. Returned by `getGCfreq`; the AnnotationHub record titles are `<motifSet>_motif_gcfreq.rds`.

Format

A named `list` with one numeric matrix per motif (names are motif IDs, matching the names of the corresponding `tf_bindsites` list). Each matrix has

5 rows Genome-wide GC quintiles, from lowest (row 1) to highest (row 5) GC. Quintile boundaries are the ones stored with `genomewide_GC`, so both objects share one scale.

one column per window position A 30 nt window slides one base at a time along the binding site (motif plus 200 bp on either side, padded by a further 65 bp so edge windows are complete). Column j is the j-th window.

Entries are the fraction of the motif's binding sites whose window at that position falls into each quintile; every column sums to one. Windows containing `N` are ignored.

Details

The tables are what make the methylTFR deviation score bias-corrected. A motif whose sites sit in GC-rich sequence overlaps hypomethylated CpG islands more often; combining these fractions with the observed methylation per GC bin gives the methylation expected from sequence composition alone, which is then subtracted.

Available motif sets:

`jaspar2020` JASPAR2020 CORE matrices, scanned genome-wide.

`cisbpv2` CIS-BP v2 PWMs (`pwms_v2` from **chromVARmotifs**), scanned genome-wide.

`altius` Vierstra motif archetypes v1.0.

Value

This page documents a data resource rather than a function; the object is returned by `getGCfreq`.

Source

Built with methylTFRAnnotationBuilder (https://github.com/EpigenomeInformatics/methylTFRAnnotationBui from BSgenome sequences; see `system.file("scripts", "make-data.R", package = "methylTFRAnnotationMm10"` and `system.file("extdata", "metadata.csv", package = "methylTFRAnnotationMm10")` for the exact sources and versions.

See Also

`getGCfreq`, `methylTFRAnnotationMm10`

Examples

```
# AnnotationHub records of this resource and their descriptions
md <- utils::read.csv(system.file("extdata", "metadata.csv",
    package = "methylTFRAnnotationMm10"
))
md[grepl("_motif_gcfreq", md$Title), c("Title", "RDataClass")]

# Loading the resource downloads it from AnnotationHub on first use:
# gcfreqs <- getGCfreq("jaspar2020")
# length(gcfreqs); dim(gcfreqs[[1]]); colSums(gcfreqs[[1]])
```

`tf_bindsites` *Transcription factor binding sites*

Description

Genome-wide predicted transcription factor binding sites on mm10. Returned by `getTFbindsites`; the AnnotationHub record titles are `<motifSet>_tf_bindsites.rds`.

Format

A `GRangesList` with one `GRanges` per motif, named by motif ID. Each range is a motif match on the primary chromosomes (chr1-19, X, Y; chrM excluded) extended by 200 bp on either side of the match, so all ranges of a motif have the same width (motif width + 400 bp). The strand of the match is kept; there are no metadata columns. Coordinates are 1-based.

Details

Available motif sets:

`jaspar2020` JASPAR2020 CORE matrices, scanned genome-wide.

`cisbpv2` CIS-BP v2 PWMs (`pwms_v2` from **chromVARmotifs**), scanned genome-wide.

`altius` Vierstra motif archetypes v1.0.

Value

This page documents a data resource rather than a function; the object is returned by `getTFbindsites`.

Source

See `motif_gcfreq`.

See Also

getTFbindsites, methylTFRAnnotationMm10

Examples

```
# AnnotationHub records of this resource and their descriptions
md <- utils::read.csv(system.file("extdata", "metadata.csv",
    package = "methylTFRAnnotationMm10"
))
md[grepl("_tf_bindsites", md$Title), c("Title", "RDataClass")]

# Loading the resource downloads it from AnnotationHub on first use:
# tfbs <- getTFbindsites("jaspar2020")
# length(tfbs); head(lengths(tfbs)); tfbs[[1]]
```

Index

∗ **datasets**
genomewide_GC, 1
motif_gcfreq, 6
tf_bindsites, 7

altius_motif_gcfreq *(*motif_gcfreq*)*, 6
altius_tf_bindsites *(*tf_bindsites*)*, 7
availableMotifSets, 1, *5*

cisbpv2_motif_gcfreq *(*motif_gcfreq*)*, 6
cisbpv2_tf_bindsites *(*tf_bindsites*)*, 7

genomewide_GC, 1, *3–6*
genomewide_GC_mm10 *(*genomewide_GC*)*, 1
getGCfreq, *1*, 2, *4–7*
getGenomeGC, *1–3*, 3, *5*
getTFbindsites, *1*, *3*, 4, *5*, *7, 8*

jaspar2020_motif_gcfreq *(*motif_gcfreq*)*, 6
jaspar2020_tf_bindsites *(*tf_bindsites*)*, 7

methylTFRAnnotationMm10, *1, 2*, *7, 8*
methylTFRAnnotationMm10 *(*methylTFRAnnotationMm10-package*)*, 5
methylTFRAnnotationMm10-package, 5
motif_gcfreq, *2, 3*, *5*, 6, *7*

tf_bindsites, *4–6*, 7