Package 'riboSeqR'

November 12, 2026

Type Package

Title Analysis of sequencing data from ribosome profiling experiments

Description Plotting functions, frameshift detection and parsing of sequencing data from ribosome profiling experiments.

Version 1.46.0

Date 2025-07-23

License GPL-3

LazyLoad yes

Depends R (>= 3.0.2), methods, GenomicRanges, abind

Imports Rsamtools, IRanges, S4Vectors, baySeq, Seqinfo, seqLogo

Suggests BiocStyle, RUnit, BiocGenerics

biocViews Sequencing,Genetics,Visualization,RiboSeq

URL https://github.com/samgg/riboSeqR

BugReports https://github.com/samgg/riboSeqR/issues

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

git_branch RELEASE_3_23

git_last_commit ed29901

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Thomas J. Hardcastle [aut],
Samuel Granjeaud [cre] (ORCID: <https://orcid.org/0000-0001-9245-1535>)

Maintainer Samuel Granjeaud <samuel.granjeaud@inserm.fr>

Contents

riboSeqR-package . . . 2
filterHits . . . 3
findCDS . . . 5
frameCounting . . . 6
lengthDist . . . 7
libScales . . . 8
logoContext . . . 9
plotCDS . . . 9
plotTranscript . . . 11
readingFrame . . . 13
readRibodata . . . 14
riboCoding-class . . . 15
riboData-class . . . 16
rnaCounts . . . 16
sliceCounts . . . 18

Index **20**

`riboSeqR-package` *Analysis of sequencing data from ribosome profiling experiments.*

Description

Plotting functions, frameshift detection and parsing of sequencing data from ribosome profiling experiments.

Details

Package:	riboSeqR
Type:	Package
Version:	1.0.0
Date:	2014-05-02
License:	GPL-3
LazyLoad:	yes
Depends:	R (>= 3.0.2), methods, baySeq, GenomicRanges, abind
Imports:	GenomicRanges, methods, abind
Packaged:	2014-05-16 10:49:24 UTC; tjh48
Built:	R 3.0.2; ; 2014-05-16 10:49:44 UTC; unix

Further information is available in the following vignettes:

`riboSeqR` riboSeqR (source, pdf)

Author(s)

Thomas J. Hardcastle

Maintainer: Thomas J. Hardcastle <tjh48@cam.ac.uk>

`filterHits` *Filters framecalled data based on the mean number of hits observed across a replicate group.*

Description

If no ribosomal footprints of the correct lengths (and frames) are seen at a coding sequence in any replicate group, this sequence is unlikely to be translated (and is therefore likely to be uninteresting). This function filters out these coding sequences, leaving only those with a minimum number of hits in at least one replicate group, and a minimum number of unique sequences aligning in at least one replicate group (to exclude single stacks of sequenced reads passing the filtering).

Usage

```
filterHits(fCs, lengths = 27, frames, hitMean = 10, unqhitMean = 1,
ratioCheck = TRUE, fS)
```

Arguments

`fCs`	`riboCoding` object defining the number of ribosome footprint reads observed over a set of coordinates, their lengths, and their frame relative to coding start sites.
`lengths`	The lengths of ribosome footprint reads to be used in filtering.
`frames`	If given, the frames of the ribosome footprint reads to be used in filtering. Should be of equal length to the 'lengths' parameter - see examples.
`hitMean`	The mean number of hits within a replicate group that should be observed to pass filtering.
`unqhitMean`	The mean number of unique sequences within a replicate group that should be observed to pass filtering. This parameter is intended to avoid cases where a coding sequence is deemed to be expressed based on a few highly expressed sequences.
`ratioCheck`	Checks the ratios of expected phase to maximal phase within the putative coding sequence. See Details.
`fS`	The output of 'frameCounting' function. Required if 'ratioCheck = TRUE'.

Details

Frames can be given as a single vector (which specifies the frames used for all lengths of footprints, or as a list, specifying the frame for each length given in 'lengths'.

For highly translated coding regions, the small proportion of out-of-phase reads may cause overlapping but out-of-phase putative coding regions to pass this filtering step. If 'ratioCheck = TRUE', filterHits identifies those cases where the phase with the maximum number of reads (the maximal phase) is not the expected phase for that putative coding region. If the ratio of reads in the expected phase to maximal phase does not significantly (Chi-squared test with significance threshold of 0.05) exceed that ratio observed for all coding regions, the putative coding region is filtered out.

Value

`riboCoding` object containing the filtered coding sequences and the associated numbers of ribosome footprint reads.

Author(s)

Thomas J. Hardcastle

See Also

`frameCounting`

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)

# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = fCs, lengths = 27:28)

# filter coding sequences. 27-mers are principally in the 1-frame,
# 28-mers are principally in the 0-frame relative to coding start (see
# readingFrame function).
```

```
ffCs <- filterHits(fCs, lengths = c(27, 28), frames = list(1, 0),
                   hitMean = 50, unqhitMean = 10, fS = fS)
```

---

findCDS *Parses a transcriptome file looking for start/stop codons in frame.*

Description

Looks in the fastaFile for defined start and stop codons in frame with one another. Reports the discovered coordinates as a GRanges object with a 'frame' value.

Usage

```
findCDS(fastaFile, startCodon = c("ATG"), stopCodon = c("TAG", "TAA", "TGA"))
```

Arguments

fastaFile	Fasta file of transcriptome sequences.
startCodon	Vector of possible start codons. Defaults to "ATG".
stopCodon	Vector of possible stop codons. Defaults to c("TAG", "TAA", "TGA").

Value

A GRanges object.

Author(s)

Thomas J. Hardcastle

Examples

```
datadir <- system.file("extdata", package = "riboSeqR")
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))
```

`frameCounting` *Counts aligned reads within coding sequence regions by frame and footprint size, splitting by frame and footprint size.*

Description

Ribosome footprint sequencing reads aligning within coding sequence regions may align in the same frame (relative to start codon) as the coding sequence, or frame shifted by 1 or 2 frames. This function calls the number of aligning reads within the coding sequence, split by frame and footprint size.

Usage

```
frameCounting(riboDat, cds, lengths = 25:30, offset5p = 0, offset3p = 0)
```

Arguments

`riboDat`	A `riboData` object containing the ribosome footprints to be counted.
`cds`	A `GenomicRanges` object containing the coordinates of the coding sequences.
`lengths`	Lengths of ribosome footprints to be included in the `riboData` object.
`offset5p`	An offset to the 5' end of the coding sequence coordinates to include ribosome footprints some distance upstream of coding start.
`offset3p`	An offset to the 3' end of the coding sequence coordinates to exclude reads too close to coding stop.

Value

A `riboCoding` object.

Author(s)

Thomas J. Hardcastle

See Also

`riboData`

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
```

```
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)
```

`lengthDist` *A function plotting the density of lengths in a riboData object.*

Description

riboData object contain data on multiple lengths of sequenced reads. This function plots the density of those lengths.

Usage

```
lengthDist(riboDat, add = FALSE, legend = NULL, ...)
```

Arguments

`riboDat`	riboData object to be plotted
`add`	Should the curve be added to an existing plot, or a new plot drawn?
`legend`	Legend text, if given.
`...`	Additional arguments to be passed to 'plot' or 'legend'.

Author(s)

Thomas J. Hardcastle

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

lengthDist(riboDat)
```

`libScales` *Generates library scaling factors from ribosome footprint (and associated RNA) data.*

Description

This function applies the library scaling factor estimation methods implemented in the baySeq package to ribosome footprint data and (if present) the associated RNA-Seq data. The library scaling factors can then be used in downstream analyses.

Usage

```
libScales(rC, riboDat, lengths, frames, method = "edgeR")
```

Arguments

`rC`	A `riboCoding` object containing the coordinates of the coding sequences and the number of ribosomal footprint reads of various classes to be found in each.
`riboDat`	A `riboData` object containing the ribosome footprint and RNA-seq alignments.
`lengths`	Lengths of ribosome footprints to inform count data.
`frames`	Frames of ribosome footprints (relative to coding start site). If omitted, all frames are used.
`method`	Method to be used for scaling factor estimation. See `getLibsizes`

Value

A list object containing

`riboLS`	Ribosome footprint library scaling factors.
`rnaLS`	RNA-Seq library scaling factors.

Author(s)

Thomas J. Hardcastle

logoContext — *Produces a motif of coding start context*

Description

The findCDS function reports the immediate context of coding start. This function uses the 'seqLogo' package to produce a motif of the contexts reported.

Usage

```
logoContext(cds, ...)
```

Arguments

cds	A GRanges object containing a '$context' value which defines the 7 bases defining coding start context.
...	Other values to be passed to the seqLogo function.

Value

A matrix containing counts of the different bases observed at each position of the 7-base context.

Author(s)

Thomas J. Hardcastle

plotCDS — *Plots average ribosome footprint alignment to coding sequences at 5' and 3' ends.*

Description

For each sample, the average (normalised by translation abundance over transcript) of the ribosome footprints of a given length alignments at the 5' and 3' ends of all specified transcripts beginning at each base relative to coding start/end are plotted. The bases are colour coded relative to start codon.

Usage

```
plotCDS(coordinates, riboDat, lengths = 27, min5p = -20, max5p = 200,
min3p = -200, max3p = 20, cap, main = "", plot = TRUE, ...)
```

Arguments

coordinates	Coordinates (as a GRanges object) of the coding sequences.
riboDat	riboData object containing ribosome footprint data.
lengths	Lengths of footprints to be plotted. May be given as a vector, in which case multiple plots will be produced.
min5p	The distance upstream of the translation start to be plotted.
max5p	The distance downstream of the translation start to be plotted.
min3p	The distance upstream of the translation end to be plotted.
max3p	The distance downstream of the translation end to be plotted.
cap	If given, caps the height of plotted values.
main	Title of the plot.
plot	Should the acquired matrix of mean expression be plotted? Defaults to TRUE.
...	Additional arguments to be passed to 'plot' and 'axes'.

Value

Invisibly returns lists of lists of matrices containing weighted averages plotted for each sample/length combination.

Author(s)

Thomas J. Hardcastle

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)

# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = fCs, lengths = 27:28)
```

```
# filter coding sequences. 27-mers are principally in the 1-frame,
# 28-mers are principally in the 0-frame relative to coding start (see
# readingFrame function).

ffCs <- filterHits(fCs, lengths = c(27, 28), frames = list(0, 2),
                   hitMean = 50, unqhitMean = 10, fS = fS)

plotCDS(coordinates = ffCs@CDS, riboDat = riboDat, lengths = 27)
```

`plotTranscript` *Plots ribosome footprint abundance and mRNA coverage (if available) for a specific transcript.*

Description

Abundances of ribosomal footprints of a given size class are plotted on a transcript. The footprints are colour coded according to the first base of the transcript, and not any coding start site, to allow for multiple coding start sites on a given transcript. Coding regions may simultaneously be plotted and colour coded under the same scheme.

Usage

```
plotTranscript(transcript, coordinates, annotation, riboData, length =
27, frameShift = 0, cap, riboScale, rnaScale, xlim, main, note = "",
libScales, ...)
```

Arguments

`transcript`	The name of the transcript to be plotted.
`coordinates`	A GRanges object containing any coding regions on the transcript.
`annotation`	A GRanges object containing annotated coding coordinates to be plotted as bars above the figure.
`riboData`	A `riboData` object containing the ribosome footprint (and optionally, RNA-seq) data.
`length`	Size class of ribosome footprint data to be plotted.
`frameShift`	Frameshift for the ribosome footprint data. See Details.
`cap`	Cap on the largest value that will be plotted as an abundance of the ribosome footprint data.
`riboScale`	Scale to be used on the ribosome footprint axis.
`rnaScale`	Scale to be used on the RNA-seq coverage axis.
`xlim`	Limits of the bases of the transcript to be plotted (i.e., the x-axis). If missing, the full transcript will be plotted.
`main`	Optional title for the plot.

note	Additional note to be added to plot titles (in addition to transcript and sample names).
libScales	If supplied, library scaling factors for normalisation of ribosomal and RNA counts (see libScales).
...	Additional arguments to be passed to plotting function.

Details

The readingFrame value allows the colour-coding of the ribosome footprints to be shifted so that the colours of the coding sequences match the colours of the ribosome footprint data. E.g., if 28-mers are predominantly in frame 2 relative to coding start, a value of 'readingFrame=2' will ensure that 28-mers in a coding region will take the same colour as that coding region if they are in the correct relative frame.

Value

NULL; plotting function.

Author(s)

Thomas J. Hardcastle

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)


# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = fCs, lengths = 27:28)

# filter coding sequences. 27-mers are principally in the 1-frame,
# 28-mers are principally in the 0-frame relative to coding start (see
# readingFrame function).
```

```
ffCs <- filterHits(fCs, lengths = c(27, 28), frames = list(1, 0),
                   hitMean = 50, unqhitMean = 10, fS = fS)

plotTranscript("CUFF.37930.1", coordinates = ffCs@CDS,
               riboData = riboDat, length = 27, cap = 200)
```

`readingFrame` *Analyses frame called ribosome footprint data within coding sequences and identifies likely frame-shift of different length ribosome footprint reads.*

Description

Ribosome footprint data data can be used to identify the frame-shift relative to start codon of the different n-mers. For each readlength specified, the sum of alignments in the different frames is shown, together with the maximum likelihood frame.

Usage

```
readingFrame(rC, lengths = 26:30)
plotFS(fS, lengths, legend.text = c("Frame 0", "Frame 1", "Frame 2"), ...)
```

Arguments

`rC`	A riboCoding object, produced by the `frameCounting` function.
`lengths`	Lengths of reads to be analysed for frame-shift, or to be plotted. May be omitted in plotting, in which case all lengths will be plotted.
`fS`	The output of the readingFrame function, to be plotted.
`legend.text`	Text for legend.
`...`	Additional arguments to be passed to barplot function.

Value

A matrix giving the number of aligned reads in each frame for each length, and the maximum likelihood frame.

Author(s)

Thomas J. Hardcastle

See Also

`frameCounting`

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
rCs <- frameCounting(riboDat, fastaCDS)

# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = rCs, lengths = 27:28)
plotFS(fS)
```

`readRibodata` *Reads ribosomal and (optionally) rna data from alignment files.*

Description

Reads BAM files, or flat text files (which may be compressed) containing strand, transcript name, start and sequence information for each alignment.

Usage

```
readRibodata(riboFiles, rnaFiles, columns = c(strand = 1, seqname = 2,
start = 3, sequence = 4), zeroIndexed = FALSE, header = FALSE, replicates, seqnames)
```

Arguments

`riboFiles`	Filenames of ribosomal alignments.
`rnaFiles`	Filenames of RNA alignments.
`columns`	Columns of alignment files containing strand, transcript (seqname) name, start of alignment, and sequence. Ignored for BAM files.
`zeroIndexed`	Are the alignments zero-indexed (i.e., the first base in a sequence is 0). Defaults to FALSE.
`header`	Does the (non-bam) alignment file have a header line? Defaults to FALSE.
`replicates`	Replicate information for the files.
`seqnames`	Transcript (seqname) names to be read into the object.

Details

If a filename ends in '.bam' or '.BAM' it will be assumed that the file is a BAM file and processed accordingly. Otherwise, it will be treated as a flat text file.

If using text files these should contain (as a minimum), the sequence of the read, the name of the sequence to which the read aligns, the strand to which it aligns, and the starting position of alignment. A Bowtie alignment (note that Bowtie, rather than Bowtie2, is recommended for short reads, which ribosome footprints are) using the option "–suppress 1,6,7,8" will generate this minimal data. It is by default assumed that the data are generated in this way, and the default columns specification for the default readRibodata function (see below) reflects this.

Value

`riboData` object.

Author(s)

Thomas J. Hardcastle

Examples

```
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT", "M", "M"))
```

---

riboCoding-class *Class "riboCoding"*

Description

The `riboCoding` class contains a set of coordinates defining coding sequences, a set of replicate data for the experimental samples, an array of ribosome footprint abundances for each coding sequence split by size class and frame, and a similar array describing the abundance of unique sequences aligning within each coding sequence.

Slots

`CDS`**:** Object of class `"GRanges"` defining the coordinates of coding sequences.

`hits`**:** An array describing the abundances of ribosome footprints, split by size class and frame (relative to coding start) for each of the coding sequences defined in the 'CDS' slot.

`unqHits`**:** An array describing the abundances of unique sequences of ribosome footprints, split by size class and frame (relative to coding start) for each of the coding sequences defined in the 'CDS' slot.

`replicates`**:** A factor defining the replicate structure of the samples for which data are held. Replicate samples will have the same level in this factor.

Methods

Methods '[' and 'show' are defined for this class.

Author(s)

Thomas J. Hardcastle

`riboData-class` *Class "riboData"*

Description

The `riboData` class contains a list of GRanges objects containing ribosome footprint alignment data, a factor defining the replicate structure of the samples involved, and (optionally) a list of GRanges objects containing RNA-seq alignment data (paired with the ribosome footprint data). It will generally be created by the 'readRibodata' function and not directly by the user.

Slots

`riboGR`**:** List of `"GRanges"` objects (one for each sequenced sample) describing the alignments of ribosomal footprint data to a transcriptome.

`rnaGR`**:** List of `"GRanges"` objects (one for each sequenced sample, paired with 'riboGR' slot) describing the alignments of RNA-seq data to a transcriptome.

`replicates`**:** A factor defining the replicate structure of the samples for which data are held. Replicate samples will have the same level in this factor.

Methods

No methods are currently defined for this class.

Author(s)

Thomas J. Hardcastle

`rnaCounts` *Extracts mRNA counts from a riboDat object for a set of coding sequence coordinates.*

Description

Takes mRNA count data from riboDat object, maps them to coding sequences specified in GRanges object, and counts the total number of hits. This is a crude approach intended to quickly produce comparable data to ribosome footprint counts. More sophisticated alternatives, addressing coverage variation, isoforms, multireads &c. have been widely described in the literature on mRNA-seq analyses.

Usage

```
rnaCounts(riboDat, CDS)
```

Arguments

riboDat	A riboData object containing the RNA-seq alignments.
CDS	A GRanges object defining the coordinates of the coding sequences for which to acquire counts.

Details

The count data thus acquired can be compared to counts of ribosomal footprint data through a beta-binomial analysis (see vignette) to discover differential translation.

Value

A matrix containing count data for the RNA-seq libraries.

Author(s)

Thomas J. Hardcastle

See Also

sliceCounts

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)

# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = fCs, lengths = 27:28)

# filter coding sequences. 27-mers are principally in the 1-frame,
```

```
# 28-mers are principally in the 0-frame relative to coding start (see
# readingFrame function).

ffCs <- filterHits(fCs, lengths = c(27, 28), frames = list(1, 0),
                   hitMean = 50, unqhitMean = 10, fS = fS)

# Extract counts of RNA hits from riboCount data.
rnaCounts <- rnaCounts(riboDat, ffCs@CDS)
```

`sliceCounts` *Slices out count data from riboCoding object for use in differential translation analyses.*

Description

For any given coding sequence, multiple lengths of reads in various frames (relative to coding start) may align. This function extracts specific size-classes and frames of ribosome footprint reads and sums them to give a single value for each coding sequence and each sequencing library, for use in downstream analysis.

Usage

```
sliceCounts(rC, lengths = 27, frames)
```

Arguments

`rC`	A `riboCoding` object containing the coordinates of the coding sequences and the number of ribosomal footprint reads of various classes to be found in each.
`lengths`	Lengths of ribosome footprints to inform count data.
`frames`	Frames of ribosome footprints (relative to coding start site). If omitted, all frames are used.

Details

Frames can be given as a single vector (which specifies the frames used for all lengths of footprints, or as a list, specifying the frame for each length given in 'lengths'.

The count data thus acquired can be compared to counts of RNA-seq data through a beta-binomial analysis (see vignette) to discover differential translation.

Value

A matrix containing counts of ribosomal footprint matches to coding sequences specified in riboCoding object 'rC'.

Author(s)

Thomas J. Hardcastle

See Also

rnaCounts

Examples

```
#ribosomal footprint data
datadir <- system.file("extdata", package = "riboSeqR")
ribofiles <- paste(datadir,
                   "/chlamy236_plus_deNovo_plusOnly_Index", c(17,3,5,7), sep = "")
rnafiles <- paste(datadir,
                  "/chlamy236_plus_deNovo_plusOnly_Index", c(10,12,14,16), sep = "")

riboDat <- readRibodata(ribofiles, rnafiles, replicates = c("WT", "WT",
"M", "M"))

# CDS coordinates
chlamyFasta <- paste(datadir, "/rsem_chlamy236_deNovo.transcripts.fa", sep = "")
fastaCDS <- findCDS(fastaFile = chlamyFasta,
                    startCodon = c("ATG"),
                    stopCodon = c("TAG", "TAA", "TGA"))

# frame calling
fCs <- frameCounting(riboDat, fastaCDS)


# analysis of frame shift for 27 and 28-mers.
fS <- readingFrame(rC = fCs, lengths = 27:28)

# filter coding sequences. 27-mers are principally in the 1-frame,
# 28-mers are principally in the 0-frame relative to coding start (see
# readingFrame function).

ffCs <- filterHits(fCs, lengths = c(27, 28), frames = list(1, 0),
                   hitMean = 50, unqhitMean = 10, fS = fS)

# Extract counts of ribosomal footprints from riboCount data.

riboCounts <- sliceCounts(ffCs, lengths = c(27, 28), frames = list(0,
2))
```

Index

∗ **IO**
findCDS, 5
readRibodata, 14
∗ **classes**
riboCoding-class, 15
riboData-class, 16
∗ **hplot**
plotCDS, 9
plotTranscript, 11
∗ **manip**
filterHits, 3
findCDS, 5
frameCounting, 6
libScales, 8
readingFrame, 13
readRibodata, 14
rnaCounts, 16
sliceCounts, 18
∗ **package**
riboSeqR-package, 2
∗ **plot**
lengthDist, 7
logoContext, 9
[,riboCoding,ANY,ANY-method *(riboCoding-class)*, 15
[,riboCoding,ANY-method *(riboCoding-class)*, 15
[,riboCoding-method *(riboCoding-class)*, 15

filterHits, 3
findCDS, 5
frameCounting, 6, *13*

getLibsizes, *8*

lengthDist, 7
libScales, 8, *12*
logoContext, 9

plotCDS, 9
plotFS *(readingFrame)*, 13
plotTranscript, 11

readingFrame, 13
readRibodata, 14
riboCoding, *8*, *18*
riboCoding *(riboCoding-class)*, 15
riboCoding-class, 15
riboData, *8*, *10*, *11*, *17*
riboData *(riboData-class)*, 16
riboData-class, 16
riboSeqR *(riboSeqR-package)*, 2
riboSeqR-package, 2
rnaCounts, 16, *19*

show,riboCoding-method *(riboCoding-class)*, 15
sliceCounts, *17*, 18