

# Package ‘GUIDEseq’

April 12, 2022

**Type** Package

**Title** GUIDE-seq analysis pipeline

**Version** 1.24.0

**Date** 2020-04-22

**Encoding** UTF-8

**Author** Lihua Julie Zhu, Michael Lawrence, Ankit Gupta,  
Hervé Pagès , Alper Kucukural, Manuel Garber, Scot A. Wolfe

**Maintainer** Lihua Julie Zhu <julie.zhu@umassmed.edu>

**Depends** R (>= 3.2.0), GenomicRanges, BiocGenerics

**Imports** BiocParallel, Biostrings, CRISPRseek, ChIPpeakAnno,  
data.table, matrixStats, BSgenome, parallel, IRanges (>=  
2.5.5), S4Vectors (>= 0.9.6), GenomicAlignments (>= 1.7.3),  
GenomeInfoDb, Rsamtools, hash, limma,dplyr

**biocViews** ImmunoOncology, GeneRegulation, Sequencing, WorkflowStep,  
CRISPR

**Suggests** knitr, RUnit, BiocStyle, BSgenome.Hsapiens.UCSC.hg19,  
TxDb.Hsapiens.UCSC.hg19.knownGene, org.Hs.eg.db

**VignetteBuilder** knitr

**Description** The package implements GUIDE-seq analysis workflow  
including functions for obtaining unique insertion  
sites (proxy of cleavage sites), estimating the locations  
of the insertion sites, aka, peaks, merging estimated insertion  
sites from plus and minus strand, and performing off target  
search of the extended regions around insertion sites.

**License** GPL (>= 2)

**LazyLoad** yes

**NeedsCompilation** no

**git\_url** <https://git.bioconductor.org/packages/GUIDEseq>

**git\_branch** RELEASE\_3\_14

**git\_last\_commit** 17cac6b

**git\_last\_commit\_date** 2021-10-26

**Date/Publication** 2022-04-12

## R topics documented:

GUIDEseq-package . . . . .	2
annotateOffTargets . . . . .	3
combineOfftargets . . . . .	4
createBarcodeFasta . . . . .	6
getPeaks . . . . .	7
getUniqueCleavageEvents . . . . .	8
getUsedBarcodes . . . . .	12
GUIDEseqAnalysis . . . . .	13
mergePlusMinusPeaks . . . . .	19
offTargetAnalysisOfPeakRegions . . . . .	21
peaks.gr . . . . .	24
uniqueCleavageEvents . . . . .	25
<b>Index</b>	<b>27</b>

---

GUIDEseq-package	<i>Analysis of GUIDE-seq</i>
------------------	------------------------------

---

## Description

The package includes functions to retain one read per unique molecular identifier (UMI), filter reads lacking integration oligo sequence, identify peak locations (cleavage sites) and heights, merge peaks, perform off-target search using the input gRNA. This package leverages CRISPRseek and ChIPpeakAnno packages.

## Details

Package: GUIDEseq  
 Type: Package  
 Version: 1.0  
 Date: 2015-09-04  
 License: GPL (>= 2)

Function GUIDEseqAnalysis integrates all steps of GUIDE-seq analysis into one function call

## Author(s)

Lihua Julie Zhu Maintainer:julie.zhu@umassmed.edu

## References

Shengdar Q Tsai and J Keith Joung et al. GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases. *Nature Biotechnology* 33, 187 to 197 (2015)

**See Also**

GUIDEseqAnalysis

**Examples**

```

if(interactive())
{
  library("BSgenome.Hsapiens.UCSC.hg19")
  umiFile <- system.file("extdata", "UMI-HEK293_site4_chr13.txt",
    package = "GUIDEseq")
  alignFile <- system.file("extdata","bowtie2.HEK293_site4_chr13.sort.bam" ,
    package = "GUIDEseq")
  grNA.file <- system.file("extdata","grNA.fa", package = "GUIDEseq")
  guideSeqRes <- GUIDEseqAnalysis(
    alignment.inputfile = alignFile,
    umi.inputfile = umiFile, grNA.file = grNA.file,
    orderOfftargetsBy = "peak_score",
    descending = TRUE,
    keepTopOfftargetsBy = "predicted_cleavage_score",
    scoring.method = "CFDscore",
    BSgenomeName = Hsapiens, min.reads = 80, n.cores.max = 1)
  guideSeqRes$offTargets
}

```

---

annotateOffTargets      *Annotate offtargets with gene name*

---

**Description**

Annotate offtargets with gene name and whether it is inside an exon

**Usage**

annotateOffTargets(thePeaks, txdb, orgAnn)

**Arguments**

- thePeaks      Output from offTargetAnalysisOfPeakRegions
- txdb      TxDb object, for creating and using TxDb object, please refer to GenomicFeatures package. For a list of existing TxDb object, please search for annotation package starting with Txdb at [http://www.bioconductor.org/packages/release/BiocViews.html#\\_\\_\\_Annotation](http://www.bioconductor.org/packages/release/BiocViews.html#___Annotation) such as TxDb.Rnorvegicus.UCSC.rn5.refGene for rat, TxDb.Mmusculus.UCSC.mm10.knownGene for mouse, TxDb.Hsapiens.UCSC.hg19.knownGene for human, TxDb.Dmelanogaster.UCSC.dm3.ensGene for Drosophila and TxDb.Celegans.UCSC.ce6.ensGene for C.elegans
- orgAnn      organism annotation mapping such as org.Hs.egSYMBOL in org.Hs.eg.db package for human

**Value**

A data frame and a tab-delimited file `offTargetsInPeakRegions.xls`, containing all input offtargets with potential gRNA binding sites, mismatch number and positions, alignment to the input gRNA and predicted cleavage score, and whether the offtargets are inside an exon and associated gene name.

**Author(s)**

Lihua Julie Zhu

**See Also**

GUIDEseqAnalysis

**Examples**

```
if (!interactive()) {
  library("BSgenome.Hsapiens.UCSC.hg19")
  library(TxDb.Hsapiens.UCSC.hg19.knownGene)
  library(org.Hs.eg.db)
  peaks <- system.file("extdata", "T2plus100offTargets.bed",
    package = "CRISPRseek")
  gRNAs <- system.file("extdata", "T2.fa",
    package = "CRISPRseek")
  outputDir = getwd()
  offTargets <- offTargetAnalysisOfPeakRegions(gRNA = gRNAs, peaks = peaks,
    format=c("fasta", "bed"),
    peaks.withHeader = TRUE, BSgenomeName = Hsapiens,
    upstream = 20L, downstream = 20L, PAM.size = 3L, gRNA.size = 20L,
    orderOfftargetsBy = "predicted_cleavage_score",
    PAM = "NGG", PAM.pattern = "(NGG|NAG|NGA)$", max.mismatch = 2L,
    outputDir = outputDir,
    allowed.mismatch.PAM = 3, overwrite = TRUE)
  annotatedOfftargets <- annotateOffTargets(offTargets,
    txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
    orgAnn = org.Hs.egSYMBOL)
}
```

---

combineOfftargets

*Combine Offtargets*

---

**Description**

Merge offtargets from different samples

**Usage**

```
combineOfftargets(offtarget.folder, sample.name,
  remove.common.offtargets = FALSE, control.sample.name,
  offtarget.filename = "offTargetsInPeakRegions.xls",
  common.col = c("offTarget", "predicted_cleavage_score",
    "gRNA.name", "gRNAPlusPAM", "offTarget_sequence",
    "guideAlignment2OffTarget", "offTargetStrand",
    "mismatch.distance2PAM", "n.PAM.mismatch",
    "n.guide.mismatch", "PAM.sequence", "offTarget_Start",
    "offTarget_End", "chromosome"),
  exclude.col,
  outputFileFileName)
```

**Arguments**

offtarget.folder	offtarget summary output folders created in GUIDEseqAnalysis function
sample.name	Sample names to be used as part of the column names in the final output file
remove.common.offtargets	Default to FALSE If set to TRUE, off-targets common to all samples will be removed.
control.sample.name	The name of the control sample for filtering off-targets present in the control sample
offtarget.filename	Default to offTargetsInPeakRegions.xls, generated in GUIDEseqAnalysis function
common.col	common column names used for merge files. Default to c("offTarget", "predicted_cleavage_score", "gRNA.name", "gRNAPlusPAM", "offTarget_sequence", "guideAlignment2OffTarget", "offTargetStrand", "mismatch.distance2PAM", "n.PAM.mismatch", "n.guide.mismatch", "PAM.sequence", "offTarget_Start", "offTarget_End", "chromosome")
exclude.col	columns to be excluded before merging. Please check offTargetsInPeakRegions.xls to choose the desired columns to exclude
outputFileName	The merged offtarget file

**Details**

Please note that by default, merged file will only contain peaks with offtargets found in the genome in GUIDEseqAnalysis function.

**Value**

a tab-delimited file similar to offTargetsInPeakRegions.tsv, containing all peaks from all samples merged by potential gRNA binding sites, mismatch number and positions, alignment to the input gRNA and predicted cleavage score. Sample specific columns have sample.name concatenated to the original column name, e.g., peak\_score becomes sample1.peak\_score.

**Author(s)**

Lihua Julie Zhu

**Examples**

```
offtarget.folder <- system.file("extdata",
  c("sample1-17", "sample2-18", "sample3-19"),
  package = "GUIDEseq")
mergedOfftargets <-
  combineOfftargets(offtarget.folder = offtarget.folder,
  sample.name = c("cas9only", "WT SpCas9", "SpCas9-MT3-ZFP"),
  outputFileName = "TS2offtargets3Constructs.xls")
```

---

createBarcodeFasta      *Create barcode as fasta file format for building bowtie1 index*

---

**Description**

Create barcode as fasta file format for building bowtie1 index to assign reads to each library with different barcodes. The bowtie1 index has been built for the standard GUIDE-seq protocol using the standard p5 and p7 index. It can be downloaded at <http://mccb.umassmed.edu/GUIDE-seq/barcode.bowtie1.index.tar.gz>

**Usage**

```
createBarcodeFasta(p5.index, p7.index, reverse.p7 = TRUE,
  reverse.p5 = FALSE, header = FALSE, outputFile = "barcodes.fa")
```

**Arguments**

p5.index	A text file with one p5 index sequence per line
p7.index	A text file with one p7 index sequence per line
header	Indicate whether there is a header in the p5.index and p7.index files. Default to FALSE
reverse.p7	Indicate whether to reverse p7 index, default to TRUE for standard GUIDE-seq experiments
reverse.p5	Indicate whether to reverse p5 index, default to FALSE for standard GUIDE-seq experiments
outputFile	Give a name to the output file where the generated barcodes are written. This file can be used to build bowtie1 index for binning reads.

**Note**

Create barcode file to be used to bin the reads sequenced in a pooled lane

**Author(s)**

Lihua Julie Zhu

**Examples**

```
p7 <- system.file("extdata", "p7.index",
  package = "GUIDEseq")
p5 <- system.file("extdata", "p5.index",
  package = "GUIDEseq")
outputFile <- "barcodes.fa"
createBarcodeFasta(p5.index = p5, p7.index = p7, reverse.p7 = TRUE,
  reverse.p5 = FALSE, outputFile = outputFile)
```

---

`getPeaks`*Obtain peaks from GUIDE-seq*

---

**Description**

Obtain strand-specific peaks from GUIDE-seq

**Usage**

```
getPeaks(gr, window.size = 20L, step = 20L, bg.window.size = 5000L,
  min.reads = 10L, min.SNratio = 2, maxP = 0.05,
  stats = c("poisson", "nbinom"), p.adjust.methods =
  c("none", "BH", "holm", "hochberg", "hommel", "bonferroni", "BY", "fdr"))
```

**Arguments**

<code>gr</code>	GRanges with cleavage sites, output from <code>getUniqueCleavageEvents</code>
<code>window.size</code>	window size to calculate coverage
<code>step</code>	step size to calculate coverage
<code>bg.window.size</code>	window size to calculate local background
<code>min.reads</code>	minimum number of reads to be considered as a peak
<code>min.SNratio</code>	minimum signal noise ratio, which is the coverage normalized by local background
<code>maxP</code>	Maximum p-value to be considered as significant
<code>stats</code>	Statistical test, default poisson
<code>p.adjust.methods</code>	Adjustment method for multiple comparisons, default none

**Value**

peaks GRanges with count (peak height), bg (local background), SNratio (signal noise ratio), p-value, and option adjusted p-value

summarized.count  
A data frame contains the same information as peaks except that it has all the sites without filtering.

**Author(s)**

Lihua Julie Zhu

**Examples**

```
if (interactive())
{
  data(uniqueCleavageEvents)
  peaks <- getPeaks(uniqueCleavageEvents$cleavage.gr,
    min.reads = 80)
  peaks$peaks
}
```

---

getUniqueCleavageEvents

*Using UMI sequence to obtain the starting sequence library*

---

**Description**

PCR amplification often leads to biased representation of the starting sequence population. To track the sequence tags present in the initial sequence library, a unique molecular identifier (UMI) is added to the 5 prime of each sequence in the starting library. This function uses the UMI sequence plus the first few sequence from R1 reads to obtain the starting sequence library.

**Usage**

```
getUniqueCleavageEvents(alignment.inputfile, umi.inputfile,
  alignment.format = c("auto", "bam", "bed"),
  umi.header = FALSE, read.ID.col = 1,
  umi.col = 2, umi.sep = "\t", keep.chrM = FALSE,
  keep.R1only = TRUE, keep.R2only = TRUE,
  concordant.strand = TRUE, max.paired.distance = 1000,
  min.mapping.quality = 30, max.R1.len = 130, max.R2.len = 130,
  apply.both.max.len = FALSE, same.chromosome = TRUE,
  distance.inter.chrom = -1, min.R1.mapped = 20, min.R2.mapped = 20,
  apply.both.min.mapped = FALSE, max.duplicate.distance = 0,
  umi.plus.R1start.unique = TRUE, umi.plus.R2start.unique = TRUE,
  n.cores.max = 6)
```



**Arguments**

<code>alignment.inputfile</code>	The alignment file. Currently supports bed output file with CIGAR information. Suggest run the workflow <code>binReads.sh</code> , which sequentially runs barcode binning, adaptor removal, alignment to genome, alignment quality filtering, and bed file conversion. Please download the workflow function and its helper scripts at <a href="http://mccb.umassmed.edu/GUIDE-seq/binReads/">http://mccb.umassmed.edu/GUIDE-seq/binReads/</a>
<code>umi.inputfile</code>	A text file containing at least two columns, one is the read identifier and the other is the UMI or UMI plus the first few bases of R1 reads. Suggest use <code>getUMI.sh</code> to generate this file. Please download the script and its helper scripts at <a href="http://mccb.umassmed.edu/GUIDE-seq/getUMI/">http://mccb.umassmed.edu/GUIDE-seq/getUMI/</a>
<code>alignment.format</code>	The format of the alignment input file. Currently only bam and bed file format is supported. BED format will be deprecated soon.
<code>umi.header</code>	Indicates whether the umi input file contains a header line or not. Default to FALSE
<code>read.ID.col</code>	The index of the column containing the read identifier in the umi input file, default to 1
<code>umi.col</code>	The index of the column containing the umi or umi plus the first few bases of sequence from the R1 reads, default to 2
<code>umi.sep</code>	column separator in the umi input file, default to tab
<code>keep.chrM</code>	Specify whether to include alignment from chrM. Default FALSE
<code>keep.R1only</code>	Specify whether to include alignment with only R1 without paired R2. Default TRUE
<code>keep.R2only</code>	Specify whether to include alignment with only R2 without paired R1. Default TRUE
<code>concordant.strand</code>	Specify whether the R1 and R2 should be aligned to the same strand or opposite strand. Default opposite.strand (TRUE)
<code>max.paired.distance</code>	Specify the maximum distance allowed between paired R1 and R2 reads. Default 1000 bp
<code>min.mapping.quality</code>	Specify min.mapping.quality of acceptable alignments
<code>max.R1.len</code>	The maximum retained R1 length to be considered for downstream analysis, default 130 bp. Please note that default of 130 works well when the read length 150 bp. Please also note that retained R1 length is not necessarily equal to the mapped R1 length
<code>max.R2.len</code>	The maximum retained R2 length to be considered for downstream analysis, default 130 bp. Please note that default of 130 works well when the read length 150 bp. Please also note that retained R2 length is not necessarily equal to the mapped R2 length
<code>apply.both.max.len</code>	Specify whether to apply maximum length requirement to both R1 and R2 reads, default FALSE

<code>same.chromosome</code>	Specify whether the paired reads are required to align to the same chromosome, default TRUE
<code>distance.inter.chrom</code>	Specify the distance value to assign to the paired reads that are aligned to different chromosome, default -1
<code>min.R1.mapped</code>	The maximum mapped R1 length to be considered for downstream analysis, default 30 bp.
<code>min.R2.mapped</code>	The maximum mapped R2 length to be considered for downstream analysis, default 30 bp.
<code>apply.both.min.mapped</code>	Specify whether to apply minimum mapped length requirement to both R1 and R2 reads, default FALSE
<code>max.duplicate.distance</code>	Specify the maximum distance apart for two reads to be considered as duplicates, default 0. Currently only 0 is supported
<code>umi.plus.R1start.unique</code>	To specify whether two mapped reads are considered as unique if both containing the same UMI and same alignment start for R1 read, default TRUE.
<code>umi.plus.R2start.unique</code>	To specify whether two mapped reads are considered as unique if both containing the same UMI and same alignment start for R2 read, default TRUE.
<code>n.cores.max</code>	Indicating maximum number of cores to use in multi core mode, i.e., parallel processing, default 6. Please set it to 1 to disable multicore processing for small dataset.

**Value**

<code>cleavage.gr</code>	Cleavage sites with one site per UMI as GRanges with metadata column total set to 1 for each range
<code>unique.umi.plus.R2</code>	a data frame containing unique cleavage site from R2 reads mapped to plus strand with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) start.y (start of readSide.y/R2 read) end.x (start of readSide.x/R1 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)
<code>unique.umi.minus.R2</code>	a data frame containing unique cleavage site from R2 reads mapped to minus strand with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) end.y (end of readSide.y/R2 read) start.x (start of readSide.x/R1 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)
<code>unique.umi.plus.R1</code>	a data frame containing unique cleavage site from R1 reads mapped to minus strand without corresponding R2 reads mapped to the plus strand, with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome

of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) start.x (start of readSide.x/R1 read) start.y (start of readSide.y/R2 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

unique.umi.minus.R1

a data frame containing unique cleavage site from R1 reads mapped to plus strand without corresponding R2 reads mapped to the minus strand, with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) end.x (end of readSide.x/R1 read) end.y (end of readSide.y/R2 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

all.umi

a data frame containing all the mapped reads with the following columns. readName (read ID), chr.x (chromosome of readSide.x/R1 read), start.x (start of readSide.x/R1 read), end.x (end of readSide.x/R1 read), mapping.qual.x (mapping quality of readSide.x/R1 read), strand.x (strand of readSide.x/R1 read), cigar.x (CIGAR of readSide.x/R1 read) , readSide.x (1/R1), chr.y (chromosome of readSide.y/R2 read) start.y (start of readSide.y/R2 read), end.y (end of readSide.y/R2 read), mapping.qual.y (mapping quality of readSide.y/R2 read), strand.y (strand of readSide.y/R2 read), cigar.y (CIGAR of readSide.y/R2 read), readSide.y (2/R2) R1.base.kept (retained R1 length), R2.base.kept (retained R2 length), distance (distance between mapped R1 and R2), UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

### Author(s)

Lihua Julie Zhu

### References

Shengdar Q Tsai and J Keith Joung et al. GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases. Nature Biotechnology 33, 187 to 197 (2015)

### See Also

getPeaks

### Examples

```
if(interactive())
{
  umiFile <- system.file("extdata", "UMI-HEK293_site4_chr13.txt",
    package = "GUIDEseq")
  alignFile <- system.file("extdata","bowtie2.HEK293_site4_chr13.sort.bam" ,
    package = "GUIDEseq")
  cleavages <- getUniqueCleavageEvents(
    alignment.inputfile = alignFile , umi.inputfile = umiFile,
    n.cores.max = 1)
  names(cleavages)
  #output a summary of duplicate counts for sequencing saturation assessment
```

```

    table(cleavages$umi.count.summary$n)
  }

```

---

getUsedBarcodes	<i>Create barcodes from the p5 and p7 index used for each sequencing lane</i>
-----------------	---

---

### Description

Create barcodes from the p5 and p7 index for assigning reads to each barcode

### Usage

```

getUsedBarcodes(p5.index, p7.index, header = FALSE, reverse.p7 = TRUE,
  reverse.p5 = FALSE, outputFile)

```

### Arguments

p5.index	A text file with one p5 index sequence per line
p7.index	A text file with one p7 index sequence per line
header	Indicate whether there is a header in the p5.index and p7.index files. Default to FALSE
reverse.p7	Indicate whether to reverse p7 index, default to TRUE for standard GUIDE-seq experiments
reverse.p5	Indicate whether to reverse p5 index, default to FALSE for standard GUIDE-seq experiments
outputFile	Give a name to the output file where the generated barcodes are written

### Value

DNAStrngSet

### Note

Create barcode file to be used to bin the reads sequenced in a pooled lane

### Author(s)

Lihua Julie Zhu

### Examples

```

p7 <- system.file("extdata", "p7.index",
  package = "GUIDEseq")
p5 <- system.file("extdata", "p5.index",
  package = "GUIDEseq")
outputFile <- "usedBarcode"
getUsedBarcodes(p5.index = p5, p7.index = p7, reverse.p7 = TRUE,
  reverse.p5 = FALSE, outputFile = outputFile)

```

---

GUIDEseqAnalysis      *Analysis pipeline for GUIDE-seq dataset*

---

## Description

A wrapper function that uses the UMI sequence plus the first few bases of each sequence from R1 reads to estimate the starting sequence library, piles up reads with a user defined window and step size, identify the insertion sites (proxy of cleavage sites), merge insertion sites from plus strand and minus strand, followed by off target analysis of extended regions around the identified insertion sites.

## Usage

```
GUIDEseqAnalysis(alignment.inputfile, umi.inputfile,
  alignment.format = c("auto", "bam", "bed"),
  umi.header = FALSE, read.ID.col = 1L,
  umi.col = 2L, umi.sep = "\t",
  BSgenomeName,
  gRNA.file,
  outputDir,
  n.cores.max = 1L,
  keep.chrM = FALSE,
  keep.R1only = TRUE, keep.R2only = TRUE,
  concordant.strand = TRUE,
  max.paired.distance = 1000L, min.mapping.quality = 30L,
  max.R1.len = 130L, max.R2.len = 130L,
  apply.both.max.len = FALSE, same.chromosome = TRUE,
  distance.inter.chrom = -1L, min.R1.mapped = 20L,
  min.R2.mapped = 20L, apply.both.min.mapped = FALSE,
  max.duplicate.distance = 0L,
  umi.plus.R1start.unique = TRUE, umi.plus.R2start.unique = TRUE,
  window.size = 20L, step = 20L, bg.window.size = 5000L,
  min.reads = 5L, min.reads.per.lib = 1L,
  min.peak.score.1strandOnly = 5L,
  min.SNratio = 2, maxP = 0.01,
  stats = c("poisson", "nbinom"),
  p.adjust.methods =
  c("none", "BH", "holm", "hochberg", "hommel", "bonferroni", "BY", "fdr"),
  distance.threshold = 40L,
  max.overlap.plusSig.minusSig = 30L,
  plus.strand.start.gt.minus.strand.end = TRUE,
  keepPeaksInBothStrandsOnly = TRUE,
  gRNA.format = "fasta",
  overlap.gRNA.positions = c(17,18),
  upstream = 20L, downstream = 20L, PAM.size = 3L, gRNA.size = 20L,
  PAM = "NGG", PAM.pattern = "NNN$", max.mismatch = 6L,
  allowed.mismatch.PAM = 2L, overwrite = TRUE,
```

```

weights = c(0, 0, 0.014, 0, 0, 0.395, 0.317, 0, 0.389, 0.079,
0.445, 0.508, 0.613, 0.851, 0.732, 0.828, 0.615, 0.804, 0.685, 0.583),
orderOfftargetsBy = c("peak_score", "predicted_cleavage_score", "n.mismatch"),
descending = TRUE,
keepTopOfftargetsOnly = TRUE,
keepTopOfftargetsBy = c("predicted_cleavage_score", "n.mismatch"),
scoring.method = c("Hsu-Zhang", "CFDscore"),
subPAM.activity = hash( AA =0,
AC = 0,
AG = 0.259259259,
AT = 0,
CA = 0,
CC = 0,
CG = 0.107142857,
CT = 0,
GA = 0.069444444,
GC = 0.022222222,
GG = 1,
GT = 0.016129032,
TA = 0,
TC = 0,
TG = 0.038961039,
TT = 0),
subPAM.position = c(22, 23),
PAM.location = "3prime",
mismatch.activity.file = system.file("extdata",
"NatureBiot2016SuppTable19DoenchRoot.csv",
package = "CRISPRseek"),
txdb,
orgAnn
)

```

## Arguments

`alignment.inputfile`

The alignment file. Currently supports bam and bed output file with CIGAR information. Suggest run the workflow `binReads.sh`, which sequentially runs barcode binning, adaptor removal, alignment to genome, alignment quality filtering, and bed file conversion. Please download the workflow function and its helper scripts at <http://mccb.umassmed.edu/GUIDE-seq/binReads/>

`umi.inputfile`

A text file containing at least two columns, one is the read identifier and the other is the UMI or UMI plus the first few bases of R1 reads. Suggest use `getUMI.sh` to generate this file. Please download the script and its helper scripts at <http://mccb.umassmed.edu/GUIDE-seq/getUMI/>

`alignment.format`

The format of the alignment input file. Default bed file format. Currently only bed file format is supported, which is generated from `binReads.sh`

`umi.header`

Indicates whether the umi input file contains a header line or not. Default to

	FALSE
read.ID.col	The index of the column containing the read identifier in the umi input file, default to 1
umi.col	The index of the column containing the umi or umi plus the first few bases of sequence from the R1 reads, default to 2
umi.sep	column separator in the umi input file, default to tab
BSgenomeName	BSgenome object. Please refer to available.genomes in BSgenome package. For example, BSgenome.Hsapiens.UCSC.hg19 for hg19, BSgenome.Mmusculus.UCSC.mm10 for mm10, BSgenome.Celegans.UCSC.ce6 for ce6, BSgenome.Rnorvegicus.UCSC.rn5 for rn5, BSgenome.Drerio.UCSC.danRer7 for Zv9, and BSgenome.Dmelanogaster.UCSC.dm3 for dm3
gRNA.file	gRNA input file path or a DNASTringSet object that contains the target sequence (gRNA plus PAM)
outputDir	the directory where the off target analysis and reports will be written to
n.cores.max	Indicating maximum number of cores to use in multi core mode, i.e., parallel processing, default 1 to disable multicore processing for small dataset.
keep.chrM	Specify whether to include alignment from chrM. Default FALSE
keep.R1only	Specify whether to include alignment with only R1 without paired R2. Default TRUE
keep.R2only	Specify whether to include alignment with only R2 without paired R1. Default TRUE
concordant.strand	Specify whether the R1 and R2 should be aligned to the same strand or opposite strand. Default opposite.strand (TRUE)
max.paired.distance	Specify the maximum distance allowed between paired R1 and R2 reads. Default 1000 bp
min.mapping.quality	Specify min.mapping.quality of acceptable alignments
max.R1.len	The maximum retained R1 length to be considered for downstream analysis, default 130 bp. Please note that default of 130 works well when the read length 150 bp. Please also note that retained R1 length is not necessarily equal to the mapped R1 length
max.R2.len	The maximum retained R2 length to be considered for downstream analysis, default 130 bp. Please note that default of 130 works well when the read length 150 bp. Please also note that retained R2 length is not necessarily equal to the mapped R2 length
apply.both.max.len	Specify whether to apply maximum length requirement to both R1 and R2 reads, default FALSE
same.chromosome	Specify whether the paired reads are required to align to the same chromosome, default TRUE

<code>distance.inter.chrom</code>	Specify the distance value to assign to the paired reads that are aligned to different chromosome, default -1
<code>min.R1.mapped</code>	The maximum mapped R1 length to be considered for downstream analysis, default 30 bp.
<code>min.R2.mapped</code>	The maximum mapped R2 length to be considered for downstream analysis, default 30 bp.
<code>apply.both.min.mapped</code>	Specify whether to apply minimum mapped length requirement to both R1 and R2 reads, default FALSE
<code>max.duplicate.distance</code>	Specify the maximum distance apart for two reads to be considered as duplicates, default 0. Currently only 0 is supported
<code>umi.plus.R1start.unique</code>	To specify whether two mapped reads are considered as unique if both containing the same UMI and same alignment start for R1 read, default TRUE.
<code>umi.plus.R2start.unique</code>	To specify whether two mapped reads are considered as unique if both containing the same UMI and same alignment start for R2 read, default TRUE.
<code>window.size</code>	window size to calculate coverage
<code>step</code>	step size to calculate coverage
<code>bg.window.size</code>	window size to calculate local background
<code>min.reads</code>	minimum number of reads to be considered as a peak
<code>min.reads.per.lib</code>	minimum number of reads in each library (usually two libraries) to be considered as a peak
<code>min.peak.score.1strandOnly</code>	Specify the minimum number of reads for a one-strand only peak to be included in the output. Applicable when set <code>keepPeaksInBothStrandsOnly</code> to FALSE and there is only one library per sample
<code>min.SNratio</code>	Specify the minimum signal noise ratio to be called as peaks, which is the coverage normalized by local background.
<code>maxP</code>	Specify the maximum p-value to be considered as significant
<code>stats</code>	Statistical test, currently only poisson is implemented
<code>p.adjust.methods</code>	Adjustment method for multiple comparisons, default none
<code>distance.threshold</code>	Specify the maximum gap allowed between the plus strand and the negative strand peak, default 40. Suggest set it to twice of <code>window.size</code> used for peak calling.
<code>max.overlap.plusSig.minusSig</code>	Specify the cushion distance to allow sequence error and imprecise integration Default to 30 to allow at most 10 ( $30 - \text{window.size} - 20$ ) bp (half window) of minus-strand peaks on the right side of plus-strand peaks. Only applicable if <code>plus.strand.start.gt.minus.strand.end</code> is set to TRUE.



<code>plus.strand.start.gt.minus.strand.end</code>	Specify whether plus strand peak start greater than the paired negative strand peak end. Default to TRUE
<code>keepPeaksInBothStrandsOnly</code>	Indicate whether only keep peaks present in both strands as specified by <code>plus.strand.start.gt.minus.strand.end</code> , <code>max.overlap.plusSig.minusSig</code> and <code>distance.threshold</code> .
<code>gRNA.format</code>	Format of the gRNA input file. Currently, fasta is supported
<code>PAM.size</code>	PAM length, default 3
<code>gRNA.size</code>	The size of the gRNA, default 20
<code>PAM</code>	PAM sequence after the gRNA, default NGG
<code>overlap.gRNA.positions</code>	The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18 for SpCas9.
<code>max.mismatch</code>	Maximum mismatch to the gRNA (not including mismatch to the PAM) allowed in off target search, default 6
<code>PAM.pattern</code>	Regular expression of protospacer-adjacent motif (PAM), default <code>NNN\$</code> . Alternatively set it to <code>(NAG NGG NGA)\$</code> for off target search
<code>allowed.mismatch.PAM</code>	Maximum number of mismatches allowed for the PAM sequence plus the number of degenerate sequence in the PAM sequence, default to 2 for NGG PAM
<code>upstream</code>	upstream offset from the peak start to search for off targets, default 20 suggest set it to window size
<code>downstream</code>	downstream offset from the peak end to search for off targets, default 20 suggest set it to window size
<code>overwrite</code>	overwrite the existing files in the output directory or not, default FALSE
<code>weights</code>	a numeric vector size of gRNA length, default <code>c(0, 0, 0.014, 0, 0, 0.395, 0.317, 0, 0.389, 0.079, 0.445, 0.508, 0.613, 0.851, 0.732, 0.828, 0.615, 0.804, 0.685, 0.583)</code> for SPCas9 system, which is used in Hsu et al., 2013 cited in the reference section. Please make sure that the number of elements in this vector is the same as the <code>gRNA.size</code> , e.g., pad 0s at the beginning of the vector.
<code>orderOfftargetsBy</code>	Criteria to order the offtargets, which works together with the descending parameter
<code>descending</code>	Indicate the output order of the offtargets, i.e., in the descending or ascending order.
<code>keepTopOfftargetsOnly</code>	Output all offtargets or the top offtarget using the <code>keepOfftargetsBy</code> criteria, default to the top offtarget
<code>keepTopOfftargetsBy</code>	Output the top offtarget for each called peak using the <code>keepTopOfftargetsBy</code> criteria, If set to <code>predicted_cleavage_score</code> , then the offtargets with the highest predicted cleavage score will be retained If set to <code>n.mismatch</code> , then the offtarget with the lowest number of mismatch to the target sequence will be retained

scoring.method	Indicates which method to use for offtarget cleavage rate estimation, currently two methods are supported, Hsu-Zhang and CFDscore
subPAM.activity	Applicable only when scoring.method is set to CFDscore A hash to represent the cleavage rate for each alternative sub PAM sequence relative to preferred PAM sequence
subPAM.position	Applicable only when scoring.method is set to CFDscore The start and end positions of the sub PAM. Default to 22 and 23 for SP with 20bp gRNA and NGG as preferred PAM
PAM.location	PAM location relative to gRNA. For example, default to 3prime for spCas9 PAM. Please set to 5prime for cpf1 PAM since it's PAM is located on the 5 prime end
mismatch.activity.file	Applicable only when scoring.method is set to CFDscore A comma separated (csv) file containing the cleavage rates for all possible types of single nucleotide mismatche at each position of the gRNA. By default, using the supplemental Table 19 from Doench et al., Nature Biotechnology 2016
txdb	TxDb object, for creating and using TxDb object, please refer to GenomicFeatures package. For a list of existing TxDb object, please search for annotation package starting with Txdb at <a href="http://www.bioconductor.org/packages/release/BiocViews.html#___Annotation">http://www.bioconductor.org/packages/release/BiocViews.html#___Annotation</a> such as TxDb.Rnorvegicus.UCSC.rn5.refGene for rat, TxDb.Mmusculus.UCSC.mm10.knownGene for mouse, TxDb.Hsapiens.UCSC.hg19.knownGene for human, TxDb.Dmelanogaster.UCSC.dm3.ensGene for Drosophila and TxDb.Celegans.UCSC.ce6.ensGene for C.elegans
orgAnn	organism annotation mapping such as org.Hs.egSYMBOL in org.Hs.eg.package for human

### Value

offTargets	a data frame, containing all input peaks with potential gRNA binding sites, mismatch number and positions, alignment to the input gRNA and predicted cleavage score.
merged.peaks	merged peaks as GRanges with count (peak height), bg (local background), SNratio (signal noise ratio), p-value, and option adjusted p-value
peaks	GRanges with count (peak height), bg (local background), SNratio (signal noise ratio), p-value, and option adjusted p-value
uniqueCleavages	Cleavage sites with one site per UMI as GRanges with metadata column total set to 1 for each range
read.summary	One table per input mapping file that contains the number of reads for each chromosome location

### Author(s)

Lihua Julie Zhu

## References

Shengdar Q Tsai and J Keith Joung et al. GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases. *Nature Biotechnology* 33, 187 to 197 (2015)

## See Also

getPeaks

## Examples

```
if(!interactive())
{
  library("BSgenome.Hsapiens.UCSC.hg19")
  umiFile <- system.file("extdata", "UMI-HEK293_site4_chr13.txt",
    package = "GUIDEseq")
  alignFile <- system.file("extdata", "bowtie2.HEK293_site4_chr13.sort.bam" ,
    package = "GUIDEseq")
  gRNA.file <- system.file("extdata", "gRNA.fa", package = "GUIDEseq")
  guideSeqRes <- GUIDEseqAnalysis(
    alignment.inputfile = alignFile,
    umi.inputfile = umiFile, gRNA.file = gRNA.file,
    orderOfftargetsBy = "peak_score",
    descending = TRUE,
    keepTopOfftargetsBy = "predicted_cleavage_score",
    scoring.method = "CFDscore",
    BSgenomeName = Hsapiens, min.reads = 80, n.cores.max = 1)
  guideSeqRes$offTargets
  names(guideSeqRes)
}
```

---

mergePlusMinusPeaks     *Merge peaks from plus strand and minus strand*

---

## Description

Merge peaks from plus strand and minus strand with required orientation and within certain distance apart

## Usage

```
mergePlusMinusPeaks(peaks.gr, peak.height.mcol = "count",
  bg.height.mcol = "bg", distance.threshold = 40L,
  max.overlap.plusSig.minusSig = 30L,
  plus.strand.start.gt.minus.strand.end = TRUE, output.bedfile)
```

**Arguments**

- `peaks.gr` Specify the peaks as GRanges object, which should contain peaks from both plus and minus strand. In addition, it should contain peak height metadata column to store peak height and optionally background height.
- `peak.height.mcol` Specify the metadata column containing the peak height, default to `count`
- `bg.height.mcol` Specify the metadata column containing the background height, default to `bg`
- `distance.threshold` Specify the maximum gap allowed between the plus stranded and the negative stranded peak, default 40. Suggest set it to twice of `window.size` used for peak calling.
- `max.overlap.plusSig.minusSig` Specify the cushion distance to allow sequence error and imprecise integration. Default to 30 to allow at most 10 ( $30 - \text{window.size} / 2$ ) bp (half window) of minus-strand peaks on the right side of plus-strand peaks. Only applicable if `plus.strand.start.gt.minus.strand.end` is set to `TRUE`.
- `plus.strand.start.gt.minus.strand.end` Specify whether plus strand peak start greater than the paired negative strand peak end. Default to `TRUE`
- `output.bedfile` Specify the bed output file name, which is used for off target analysis subsequently.

**Value**

output a list and a bed file containing the merged peaks a data frame of the bed format

- `mergedPeaks.gr` merged peaks as GRanges
- `mergedPeaks.bed` merged peaks in bed format

**Author(s)**

Lihua Julie Zhu

**References**

Zhu L.J. et al. (2010) ChIPpeakAnno: a Bioconductor package to annotate ChIP-seq and ChIP-chip data. BMC Bioinformatics 2010, 11:237doi:10.1186/1471-2105-11-237. Zhu L.J. (2013) Integrative analysis of ChIP-chip and ChIP-seq dataset. Methods Mol Biol. 2013;1067:105-24. doi: 10.1007/978-1-62703-607-8\_8.

**Examples**

```
if (interactive())
{
  data(peaks.gr)
  mergedPeaks <- mergePlusMinusPeaks(peaks.gr = peaks.gr,
    output.bedfile = "mergedPeaks.bed")
}
```

```

mergedPeaks$mergedPeaks.gr
head(mergedPeaks$mergedPeaks.bed)
}

```

---

offTargetAnalysisOfPeakRegions

*Offtarget Analysis of GUIDE-seq peaks*


---

## Description

Finding offtargets around peaks from GUIDE-seq or around any given genomic regions

## Usage

```

offTargetAnalysisOfPeakRegions(gRNA, peaks,
  format=c("fasta", "bed"),
  peaks.withHeader = FALSE, BSgenomeName, overlap.gRNA.positions = c(17,18),
  upstream = 20L, downstream = 20L, PAM.size = 3L, gRNA.size = 20L,
  PAM = "NGG", PAM.pattern = "NNN$", max.mismatch = 6L,
  outputDir, allowed.mismatch.PAM = 2L, overwrite = TRUE,
  weights = c(0, 0, 0.014, 0, 0, 0.395,
  0.317, 0, 0.389, 0.079, 0.445, 0.508, 0.613, 0.851, 0.732, 0.828, 0.615,
  0.804, 0.685, 0.583),
  orderOfftargetsBy = c("predicted_cleavage_score", "n.mismatch"),
  descending = TRUE,
  keepTopOfftargetsOnly = TRUE,
  scoring.method = c("Hsu-Zhang", "CFDscore"),
  subPAM.activity = hash( AA =0,
    AC = 0,
    AG = 0.259259259,
    AT = 0,
    CA = 0,
    CC = 0,
    CG = 0.107142857,
    CT = 0,
    GA = 0.069444444,
    GC = 0.022222222,
    GG = 1,
    GT = 0.016129032,
    TA = 0,
    TC = 0,
    TG = 0.038961039,
    TT = 0),
  subPAM.position = c(22, 23),
  PAM.location = "3prime",
  mismatch.activity.file = system.file("extdata",
    "NatureBiot2016SuppTable19DoenchRoot.csv",

```

```

        package = "CRISPRseek"),
    n.cores.max = 1
)

```

### Arguments

gRNA	gRNA input file path or a DNASTringSet object that contains gRNA plus PAM sequences used for genome editing
peaks	peak input file path or a GenomicRanges object that contains genomic regions to be searched for potential offtargets
format	Format of the gRNA and peak input file. Currently, fasta and bed are supported for gRNA and peak input file respectively
peaks.withHeader	Indicate whether the peak input file contains header, default FALSE
PAM.size	PAM length, default 3
gRNA.size	The size of the gRNA, default 20
PAM	PAM sequence after the gRNA, default NGG
BSgenomeName	BSgenome object. Please refer to available.genomes in BSgenome package. For example, BSgenome.Hsapiens.UCSC.hg19 for hg19, BSgenome.Mmusculus.UCSC.mm10 for mm10, BSgenome.Celegans.UCSC.ce6 for ce6, BSgenome.Rnorvegicus.UCSC.rn5 for rn5, BSgenome.Drerio.UCSC.danRer7 for Zv9, and BSgenome.Dmelanogaster.UCSC.dm3 for dm3
overlap.gRNA.positions	The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18 for SpCas9.
max.mismatch	Maximum mismatch allowed in off target search, default 6
PAM.pattern	Regular expression of protospacer-adjacent motif (PAM), default to any NNN\$. Set it to (NAGINGGINGA)\$ if only outputs offtargets with NAG, NGA or NGG PAM
allowed.mismatch.PAM	Number of degenerative bases in the PAM.pattern sequence, default to 2
outputDir	the directory where the off target analysis and reports will be written to
upstream	upstream offset from the peak start to search for off targets, default 20
downstream	downstream offset from the peak end to search for off targets, default 20
overwrite	overwrite the existing files in the output directory or not, default FALSE
weights	a numeric vector size of gRNA length, default c(0, 0, 0.014, 0, 0, 0.395, 0.317, 0, 0.389, 0.079, 0.445, 0.508, 0.613, 0.851, 0.732, 0.828, 0.615, 0.804, 0.685, 0.583) for SPCas9 system, which is used in Hsu et al., 2013 cited in the reference section. Please make sure that the number of elements in this vector is the same as the gRNA.size, e.g., pad 0s at the beginning of the vector.
orderOfftargetsBy	criteria to order the offtargets by and the top one will be kept if keepTopOfftargetsOnly is set to TRUE. If set to predicted_cleavage_score (descending order), the offtarget with the highest predicted cleavage score for each peak will be kept. If set to n.mismatch (ascending order), the offtarget with the smallest number of mismatch to the target sequence for each peak will be kept.

descending	No longer used. In the descending or ascending order. Default to order by predicted cleavage score in descending order and number of mismatch in ascending order When altering orderOfftargetsBy order, please also modify descending accordingly
keepTopOfftargetsOnly	Output all offtargets or the top offtarget per peak using the orderOfftargetsBy criteria, default to the top offtarget
scoring.method	Indicates which method to use for offtarget cleavage rate estimation, currently two methods are supported, Hsu-Zhang and CFDscore
subPAM.activity	Applicable only when scoring.method is set to CFDscore A hash to represent the cleavage rate for each alternative sub PAM sequence relative to preferred PAM sequence
subPAM.position	Applicable only when scoring.method is set to CFDscore The start and end positions of the sub PAM. Default to 22 and 23 for SP with 20bp gRNA and NGG as preferred PAM
PAM.location	PAM location relative to gRNA. For example, default to 3prime for spCas9 PAM. Please set to 5prime for cpf1 PAM since it's PAM is located on the 5 prime end
mismatch.activity.file	Applicable only when scoring.method is set to CFDscore A comma separated (csv) file containing the cleavage rates for all possible types of single nucleotide mismatches at each position of the gRNA. By default, using the supplemental Table 19 from Doench et al., Nature Biotechnology 2016
n.cores.max	Indicating maximum number of cores to use in multi core mode, i.e., parallel processing, default 1 to disable multicore processing for small dataset.

### Value

a tab-delimited file offTargetsInPeakRegions.tsv, containing all input peaks with potential gRNA binding sites, mismatch number and positions, alignment to the input gRNA and predicted cleavage score.

### Author(s)

Lihua Julie Zhu

### References

Patrick D Hsu, David A Scott, Joshua A Weinstein, F Ann Ran, Silvana Konermann, Vineeta Agarwala, Yinqing Li, Eli J Fine, Xuebing Wu, Ophir Shalem, Thomas J Cradick, Luciano A Marraffini, Gang Bao & Feng Zhang (2013) DNA targeting specificity of rNA-guided Cas9 nucleases. Nature Biotechnology 31:827-834 Lihua Julie Zhu, Benjamin R. Holmes, Neil Aronin and Michael Brodsky. CRISPRseek: a Bioconductor package to identify target-specific guide RNAs for CRISPR-Cas9 genome-editing systems. Plos One Sept 23rd 2014 Lihua Julie Zhu (2015). Overview of guide RNA design tools for CRISPR-Cas9 genome editing technology. Frontiers in Biology August 2015, Volume 10, Issue 4, pp 289-296

**See Also**

GUIDEseq

**Examples**

```
#### the following example is also part of annotateOffTargets.Rd
if (interactive()) {
  library("BSgenome.Hsapiens.UCSC.hg19")
  peaks <- system.file("extdata", "T2plus100offTargets.bed",
    package = "CRISPRseek")
  gRNAs <- system.file("extdata", "T2.fa",
    package = "CRISPRseek")
  outputDir = getwd()
  offTargets <- offTargetAnalysisOfPeakRegions(gRNA = gRNAs, peaks = peaks,
    format=c("fasta", "bed"),
    peaks.withHeader = TRUE, BSgenomeName = Hsapiens,
    upstream = 20L, downstream = 20L, PAM.size = 3L, gRNA.size = 20L,
    orderOfftargetsBy = "predicted_cleavage_score",
    PAM = "NGG", PAM.pattern = "(NGG|NAG|NGA)$", max.mismatch = 2L,
    outputDir = outputDir,
    allowed.mismatch.PAM = 3, overwrite = TRUE
  )
}
```

peaks.gr

*example cleavage sites***Description**

An example data set containing cleavage sites (peaks) from getPeaks

**Usage**

```
data("peaks.gr")
```

**Format**

GRanges with count (peak height), bg (local background), SNratio (signal noise ratio), p-value, and option adjusted p-value

**Value**

peaks.gr GRanges with count (peak height), bg (local background), SNratio (signal noise ratio), p-value, and option adjusted p-value

**Source**

<http://trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR1695644>



**Examples**

```
data(peaks.gr)
names(peaks.gr)
peaks.gr
```

---

uniqueCleavageEvents *example unique cleavage sites*

---

**Description**

An example data set containing cleavage sites with unique UMI, generated from getUniqueCleavageEvents

**Usage**

```
data("uniqueCleavageEvents")
```

**Value**

**cleavage.gr** Cleavage sites with one site per UMI as GRanges with metadata column total set to 1 for each range

**unique.umi.plus.R2** a data frame containing unique cleavage site from R2 reads mapped to plus strand with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) start.y (start of readSide.y/R2 read) end.x (start of readSide.x/R1 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

**unique.umi.minus.R2** a data frame containing unique cleavage site from R2 reads mapped to minus strand with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) end.y (end of readSide.y/R2 read) start.x (start of readSide.x/R1 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

**unique.umi.plus.R1** a data frame containing unique cleavage site from R1 reads mapped to minus strand without corresponding R2 reads mapped to the plus strand, with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) start.x (start of readSide.x/R1 read) start.y (start of readSide.y/R2 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

**unique.umi.minus.R1** a data frame containing unique cleavage site from R1 reads mapped to plus strand without corresponding R2 reads mapped to the minus strand, with the following columns chr.y (chromosome of readSide.y/R2 read) chr.x (chromosome of readSide.x/R1 read) strand.y (strand of readSide.y/R2 read) strand.x (strand of readSide.x/R1 read) end.x (end of readSide.x/R1 read) end.y (end of readSide.y/R2 read) UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

**all.umi** a data frame containing all the mapped reads with the following columns. readName (read ID), chr.x (chromosome of readSide.x/R1 read), start.x (start of readSide.x/R1 read), end.x (end of readSide.x/R1 read), mapping.qual.x (mapping quality of readSide.x/R1 read), strand.x (strand of readSide.x/R1 read), cigar.x (CIGAR of readSide.x/R1 read), readSide.x (1/R1), chr.y (chromosome of readSide.y/R2 read), start.y (start of readSide.y/R2 read), end.y (end of readSide.y/R2 read), mapping.qual.y (mapping quality of readSide.y/R2 read), strand.y (strand of readSide.y/R2 read), cigar.y (CIGAR of readSide.y/R2 read), readSide.y (2/R2), R1.base.kept (retained R1 length), R2.base.kept (retained R2 length), distance (distance between mapped R1 and R2), UMI (unique molecular identifier (umi) or umi with the first few bases of R1 read)

### Source

<http://trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR1695644>

### Examples

```
data(uniqueCleavageEvents)
names(uniqueCleavageEvents)
sapply(uniqueCleavageEvents, class)
uniqueCleavageEvents[[1]] # GRanges object
lapply(uniqueCleavageEvents, dim)
```

# Index

## \* datasets

peaks.gr, [24](#)  
uniqueCleavageEvents, [25](#)

## \* manip

createBarcodeFasta, [6](#)  
getUsedBarcodes, [12](#)

## \* misc

combineOfftargets, [4](#)  
getPeaks, [7](#)  
getUniqueCleavageEvents, [8](#)  
GUIDEseqAnalysis, [13](#)  
mergePlusMinusPeaks, [19](#)  
offTargetAnalysisOfPeakRegions, [21](#)

## \* package

GUIDEseq-package, [2](#)

## \* utilities

annotateOffTargets, [3](#)  
createBarcodeFasta, [6](#)  
getUsedBarcodes, [12](#)

annotateOffTargets, [3](#)

combineOfftargets, [4](#)  
createBarcodeFasta, [6](#)

getPeaks, [7](#)  
getUniqueCleavageEvents, [8](#)  
getUsedBarcodes, [12](#)  
GUIDEseq (GUIDEseq-package), [2](#)  
GUIDEseq-package, [2](#)  
GUIDEseqAnalysis, [13](#)

mergePlusMinusPeaks, [19](#)

offTargetAnalysisOfPeakRegions, [21](#)

peaks.gr, [24](#)

uniqueCleavageEvents, [25](#)